HACETTEPE ÜNİVERSİTESİ
EĞİTİM BİLİMLERİ ENSTİTÜSÜ

Bilgisayar ve Öğretim Teknolojileri Eğitimi Ana Bilim Dalı

NESNEYE YÖNELİK VE ROBOT PROGRAMLAMANIN ÖĞRENCİ BAŞARISINA, SOYUTLAMAYA, PROBLEM ÇÖZMEYE VE MOTİVASYONA ETKİLERİNİN İNCELENMESİ

Murat ÇINAR

Doktora Tezi

Ankara, 2019

Liderlik, araştırma, inovasyon, kaliteli eğitim ve değişim ile

Daha ileriye... En İyiye...

HACETTEPE ÜNİVERSİTESİ
EĞİTİM BİLİMLERİ ENSTİTÜSÜ

Bilgisayar ve Öğretim Teknolojileri Eğitimi Ana Bilim Dalı

NESNEYE YÖNELİK VE ROBOT PROGRAMLAMANIN ÖĞRENCİ BAŞARISINA, SOYUTLAMAYA, PROBLEM ÇÖZMEYE VE MOTİVASYONA ETKİLERİNİN İNCELENMESİ

AN EXAMINATION OF THE EFFECTS OF OBJECT-ORIENTED AND ROBOTICS PROGRAMMING ON STUDENT ACHIEVEMENT, ABSTRACTION, PROBLEM SOLVING AND MOTIVATION

Murat ÇINAR

Doktora Tezi

Ankara, 2019

Kabul ve Onay

Eğitim Bilimleri Enstitüsü Müdürlüğüne,

MURAT ÇINAR'ın hazırladığı "Nesneye Yönelik ve Robot Programlamanın Öğrenci Başarısına, Soyutlamaya, Problem Çözmeye ve Motivasyona Etkilerinin İncelenmesi" başlıklı bu çalışma jürimiz tarafından **Bilgisayar ve Öğretim Teknolojileri Eğitimi Ana Bilim Dalı, Bilgisayar ve Öğretim Teknolojileri Eğitimi Bilim Dalında Doktora Tezi** olarak kabul edilmiştir.

Jüri Başkanı	Prof. Dr. Arif ALTUN	
Jüri Üyesi (Danışman)	Prof. Dr. Hakan TÜZÜN	
Jüri Üyesi	Prof. Dr. Selçuk ÖZDEMİR	
Jüri Üyesi	Prof. Dr. S. Sadi SEFEROĞLU	
Jüri Üyesi	Doç. Dr. Filiz KALELİOĞLU	

Bu tez Hacettepe Üniversitesi Lisansüstü Eğitim, Öğretim ve Sınav Yönetmeliği'nin ilgili maddeleri uyarınca yukarıdaki jüri üyeleri tarafından 17 / 05 / 2019 tarihinde uygun görülmüş ve Enstitü Yönetim Kurulunca ..... / ..... / 2019 tarihinde kabul edilmiştir.

Prof. Dr. Ali Ekber ŞAHİN
Eğitim Bilimleri Enstitüsü Müdürü

Öz

Bu çalışmada nesneye yönelik (karşılaştırma) ve robot (deney) programlamanın öğrenci başarısına, soyutlamaya, problem çözmeye ve motivasyona etkileri karşılaştırmalı olarak incelenmiştir. Katılımcı grubunu 81 onuncu sınıf öğrencisi oluşturmaktadır. Gömülü deneysel araştırma deseninin benimsendiği çalışmada veriler ardışık iki durum çalışmasından elde edilmiştir. On iki hafta süren ilk durum çalışmasına 41 öğrenci (20 deney, 21 karşılaştırma); sekiz hafta süren ikinci durum çalışmasına ise 40 öğrenci (21 deney, 19 karşılaştırma) katılmıştır. Uygulamaların öncesinde katılımcılara programlamaya yönelik başarı testi, problem çözme ölçeği, formal ve betimsel soyutlama testleri uygulanmıştır. Uygulamaların sonrasında ise katılımcılara bu araçlara ek olarak motivasyon ölçeği uygulanmıştır. Çalışmada ders günlükleri, problem çözme taslakları, sesli düşünme ve odak grup görüşme kayıtları gibi nitel veriler de kullanılmıştır. Uygulamaların sonrasında hem deney hem de karşılaştırma gruplarının programlama puanları istatistiksel olarak anlamlı şekilde artmıştır. Ancak bu artış yöntemler arasında anlamlı şekilde farklılaşmamıştır. Uygulamalar sonrası karşılaştırma grupları formal ve betimsel soyutlama puanlarında anlamlı bir değişiklik bulunmazken, deney gruplarının formal (sadece ikinci durum çalışmasında) ve betimsel soyutlama puanlarında anlamlı bir artış gerçekleşmiştir. Yöntemler arası soyutlama puanları karşılaştırıldığında sadece ikinci durum çalışmasında betimsel soyutlama puanları açısından deney grubu lehine istatistiksel olarak anlamlı bir farklılık bulunmuştur. Uygulamalar sonrası grup içi ya da gruplar arası problem çözme puanlarında istatistiksel olarak anlamlı bir farklılık bulunmamıştır. Her iki durum çalışmasında da deney grupları motivasyon puanlarının karşılaştırma gruplarına göre istatistiksel olarak anlamlı şekilde yüksek olduğu bulunmuştur.

Anahtar sözcükler: bilgisayımsal düşünme, programlama eğitimi, eğitsel robotlar, görsel programlama, soyutlama, problem çözme, motivasyon, öğrenci başarısı

Abstract

This study comparatively examined the effects of object-oriented (comparison) and robotic (experimental) programming on student achievement, abstraction, problem solving, and motivation. The study included 81 tenth-grade students. In the study that adopted embedded experimental research design, data were obtained from two consecutive case studies. The first case in which 41 students took part (20 experimental, 21 comparison), lasted twelve weeks; while the latter involving 40 students (21 experimental, 19 comparison) lasted eight weeks. The participants were administered achievement test regarding computer programming, problem-solving scale, and formal and descriptive abstraction tests before the implementations. Following the implementations, the motivation scale was also applied in addition to these instruments. Qualitative data such as course diaries, problem solving sketches, think aloud and focus group interview records were also used in the study. The programming scores of both comparison and experimental groups increased statistically significantly after the implementations. However, this increase did not differ significantly across the methods. While there was no significant difference within the formal or descriptive abstraction scores of the comparison groups, there was a significant increase found in formal (only in case-2) and descriptive abstraction scores of the experimental groups. In examining the abstraction scores between the methods, a statistically significant difference existed in descriptive abstraction scores in favor of the experimental group only in case-2. There was no statistically significant difference in terms of the problem-solving scores between or within the test groups. Motivation scores of the experimental groups were statistically higher than the comparison groups in both cases.

Keywords: computational thinking, programming education, educational robots, visual programming, abstraction, problem solving, motivation, student achievement

Teşekkür

Bu çalışmanın ortaya çıkmasında katkılarını esirgemeyen, her zaman aydınlatıcı ve yol gösterici olan, çalışmanın her aşamasında bilgi ve deneyimlerinden yararlandığım değerli hocam ve tez danışmanım Prof. Dr. Hakan TÜZÜN'e sonsuz teşekkürlerimi ve saygılarımı sunarım.

Çalışmamı inceleyerek değerli katkılarını eksik etmeyen Prof. Dr. Arif ALTUN'a, Prof. Dr. Selçuk ÖZDEMİR'e, Prof. Dr. Süleyman Sadi SEFEROĞLU'na ve Doç. Dr. Filiz KALELİOĞLU'na teşekkür eder, saygılarımı sunarım. Ayrıca veri analizi sürecinde yol gösterici tavsiyeleriyle çalışmama destek sağlayan Prof. Dr. Halil YURDUGÜL'e ve Prof. Dr. Özgül YILMAZ TÜZÜN'e minnettar olduğumu belirtmek isterim.

Yüksek lisans ve doktora eğitimim boyunca manevi desteğini hiç esirgemeyen aileme sonsuz sevgi ve teşekkürlerimi sunarım.

İçindekiler

Öz ii

Abstract iii

Teşekkür iv

İçindekiler v

Tablolar Dizini viii

Şekiller Dizini xii

Simgeler ve Kısaltmalar Dizini xiii

Bölüm 1 Giriş 1

Problem Durumu 3

Araştırmanın Amacı ve Önemi 5

Araştırma Problemi 6

Sayıltılar 6

Sınırlılıklar 7

Tanımlar 8

Bölüm 2 Araştırmanın Kuramsal Temeli ve İlgili Araştırmalar 10

Bilgisayımsal Düşünme 10

Etimolojik Bir Karmaşa: Bilgisayım ve Bilgisayımsal Düşünme 15

Bilgisayımsal Okuryazarlık, Bilgisayımsal Düşünme, Programlama ve Kodlama Kavramları Arasındaki Farklılıklar 16

Bilgisayımın Eğitim Ortamlarına Entegrasyonu 17

Bilgisayımsal Düşünme için Önerilen Çerçeveler 19

Ülkemizde Bilgisayımın Eğitime Entegrasyonu 22

Bilgisayımsal Düşünme Araçları 23

Bilgisayımsal Düşünme Teknikleri 26

Robotik Etkinlikler ve Robot Programlama 32

Yapa-Boza Öğrenme 37

Soyutlama 38
Problem Çözme 39
Motivasyon 50
Bilgisayımsal Düşünme ile İlgili Araştırmalar 56
Bölüm 3 Yöntem 73
Araştırmanın Evreni ve Örneklemi 78
Veri Toplama Süreci 81
Veri Toplama Araçları 87
Verilerin Analizi 96
Araştırmacının Rolü 100
İç ve Dış Geçerlik 100
Bölüm 4 Bulgular ve Yorumlar 104
Bilgisayımsal Düşünme Etkinliklerinin Öğrencilerin Programlama Başarısı Üzerine Etkileri 104
Bilgisayımsal Düşünme Etkinliklerinin Öğrencilerin Soyutlama Becerisi Üzerine Etkileri 110
Bilgisayımsal Düşünme Etkinliklerinin Öğrencilerin Problem Çözme Algıları Üzerine Etkileri 120
Bilgisayımsal Düşünme Becerilerinin Geliştirilmesinde Kullanılan Farklı Programlama Etkinliklerinin Öğrenci Motivasyonu Üzerine Etkileri 127
Programlama Başarısı, Problem Çözme, Soyutlama ve Motivasyon Arasındaki İlişki 129
Bilgisayımsal Problem Çözme Sürecinde Öne Çıkan Bilişsel İşlemler 131
Öğrencilerin Bilgisayımsal Problem Çözme Etkinlikleri Esnasında Yaşadıkları Deneyimleri ve Karşılaştıkları Engeller 138
Bölüm 5 Sonuç, Tartışma ve Öneriler 197
Sonuç ve Tartışma 197
Öneriler 209

Kaynaklar 213
EK-A: Problem Çözme Etkinlik Örnekleri 239
EK-B: Programlama Başarı Testi Soru Örnekleri 241
EK-C: Odak Grup Görüşme Soruları 243
EK-Ç: Ders Günlüğü Örnek Formu 245
EK-D: Nitel Veri Analizi Sonucunda Ortaya Çıkan Serbest Kodlar ve Kodlama Sıklıkları 246
EK-E: Nitel Veri Analizi Sonucunda Ortaya Çıkan Temalar (🏛), Kategoriler (☂) ve Kodlar (🔑) 250
EK-F: Etik Komisyonu Onay Bildirimi 259
EK-G: Etik Beyanı 260
EK-H: Doktora Tez Çalışması Orijinallik Raporu 261
EK-I: Dissertation Originality Report 262
EK-İ: Yayımlama ve Fikrî Mülkiyet Hakları Beyanı 263

Tablolar Dizini

Tablo 1 *Alanyazında "Computational Thinking" Kavramı İçin Önerilen Türkçe Kavram Karşılıkları* 15
Tablo 2 *Brennan ve Resnick (2012) Tarafından Önerilen Bilgisayımsal Düşünme Çerçevesi* 19
Tablo 3 *Bilgisayımsal Düşünme İçin Kullanılan Ortam ve Araçlar* 24
Tablo 4 *Polya'nın Problem Çözme Aşamaları* 47
Tablo 5 *Akademik Veri Tabanı Taraması İçin Kullanılan Bilimsel Çalışma Dâhil Etme ve Dışlama Kriterleri* 57
Tablo 6 *Yıllara Göre Yayımlanan Makale Sayıları* 58
Tablo 7 *Makalelerin Dergilere Göre Dağılımı* 58
Tablo 8 *Ülkelere Göre Yayımlanan Makale Sayısı* 59
Tablo 9 *Araştırma Türüne Göre Yayın İstatistikleri* 59
Tablo 10 *Makalelerin Araştırma Yöntemine Göre Dağılımı* 60
Tablo 11 *Araştırma Desenlerine Göre Çalışma Sayıları* 60
Tablo 12 *Öğrenme Bağlamına Göre Çalışma Sayıları* 60
Tablo 13 *Çalışmaların Farklı Akademik Disiplinlere Göre Dağılımı* 61
Tablo 14 *Çalışmaların Altında Yatan Kuramsal Temel (Perspektif, Öğrenme Kuramları ve Modelleri) Açısından İncelenmesi* 62
Tablo 15 *Çalışmalarda Kullanılan Öğretim Yöntem ya da Tekniklerinin Dağılımı*. 62
Tablo 16 *Makalelerin Araştırma Süreleri Açısından İncelenmesi* 63
Tablo 17 *Makalelerin Katılımcı Düzeyine Göre Dağılımı* 63
Tablo 18 *Makalelerin Örneklem Boyutu Açısından İncelenmesi*..... 64
Tablo 19 *Makalelerin Örneklem Seçim Yöntemine Göre Dağılımı* 64
Tablo 20 *Makalelerde İncelenen Değişkenlerin Dağılımı* 65
Tablo 21 *Makalelerin Veri Toplama Araçları Yönünden İncelenmesi* 66
Tablo 22 *Makalelerde Kullanılan Veri Analiz Teknikleri*..... 66
Tablo 23 *Makalelerin Öğretim Verilme Yöntemi Açısından İncelenmesi* 67
Tablo 24 *Makalelerin Bilgisayımsal Düşünme Bağlamı Açısından İncelenmesi* ... 67
Tablo 25 *Katılımcıların Programlama Deneyimleri* 67
Tablo 26 *Çalışmalarda Bilgisayımsal Düşünmenin Geliştirilmesi İçin Kullanılan Teknik ve Stratejilerin İncelenmesi*..... 68
Tablo 27 *Makalelerde Kullanılan Teknolojik Ortam Modalitesi* 68

Tablo 28 *Bilgisayımsal Düşünme İçin Kullanılan Teknolojik Ortam Arayüzü* 69
Tablo 29 *Çalışmalarda Kullanılan Bilgisayımsal Düşünme ya da Programlama Etkinlik Biçimi* 69
Tablo 30 *Çalışmalarda Kullanılan Araç-Gereç ve BİT Donanımları* 69
Tablo 31 *Çalışmalarda Kullanılan Yazılım Araçları* 70
Tablo 32 *Makalelerin Katılımcı Çalışma Biçimi Açısından İncelenmesi* 71
Tablo 33 *Makalelerin Bilgisayımsal Düşünme Değerlendirme Yaklaşımları Açısından İncelenmesi* 71
Tablo 34 *Durum Çalışmalarına göre Deney ve Karşılaştırma Grubuna Seçkisiz Olarak Atanan Katılımcıların Sayı, Cinsiyet ve Akademik Başarı Ortalaması Açısından Dağılımı* 78
Tablo 35 *Katılımcılara ait Bazı Betimsel İstatistikler* 80
Tablo 36 *Uygulamanın Gerçekleştiği Ortama İlişkin Bilgiler* 82
Tablo 37 *Durum Çalışması 1'deki Haftalık Ders İçerikleri* 84
Tablo 38 *Durum Çalışması 2'deki Haftalık Ders İçerikleri* 86
Tablo 39 *Programlama Başarısı Testinin Kapsam Geçerliği için Görüşüne Başvurulan Uzmanların Demografik Özellikleri* 88
Tablo 40 *Programlama Başarısı Testine İlişkin Madde İstatistikleri* 89
Tablo 41 *Programlama Başarısı Testine İlişkin Test İstatistikleri* 90
Tablo 42 *Soyutlama Envanterinin Türkçe Formunun Uygunluğuna İlişkin Görüşüne Başvurulan Uzmanlara Ait Demografik Bilgiler* 91
Tablo 43 *Veri Kaynaklarının Nitel Analizi Sonucundaki Kodlanma İstatistikleri* 98
Tablo 44 *Programlama Başarısı Ön Test ve Son Test Puanlarına İlişkin Çarpıklık, Basıklık ve Normallik Testi İstatistikleri* 105
Tablo 45 *Uygulama Öncesi Gruplar arası Programlama Başarısı Ön Test Puanlarının Karşılaştırılması* 106
Tablo 46 *Uygulama Öncesi ve Sonrası Grup içi Programlama Başarı Test Puanlarının Değişimi* 106
Tablo 47 *Akademik Başarı Ortalaması, Programlama Başarısı Ön Test ve Programlama Başarısı Son Test Puanları Arasındaki İlişkilerin İncelenmesi* 108
Tablo 48 *Programlama Başarısı Son Test Puanlarına İlişkin Grup Ortalama ve Düzeltilmiş Ortalama Puanları* 109
Tablo 49 *Deney ve Karşılaştırma Grubu Öğrencilerinin Programlama Başarısı Son Test Puanlarının ANCOVA ile Karşılaştırılması* 110

Tablo 50 *Formal Soyutlama Ön Test ve Son Test Puanlarına İlişkin Çarpıklık, Basıklık ve Normallik Testi İstatistikleri* 111
Tablo 51 *Uygulama Öncesi Gruplar Arası Formal Soyutlama Ön Test Puanlarının Karşılaştırılması* 112
Tablo 52 *Uygulama Öncesi ve Sonrası Grup İçi Formal Soyutlama Test Puanlarının Değişiminin Wilcoxon İşaretli Sıralar Testi ile Analizi* 113
Tablo 53 *Akademik Başarı Ortalaması, Formal Soyutlama Ön Test ve Formal Soyutlama Son Test Puanları Arasındaki İlişkilerin İncelenmesi* 114
Tablo 54 *Uygulama Sonrası Gruplar Arası Formal Soyutlama Son Test Puanlarının Karşılaştırılması* 115
Tablo 55 *Betimsel Soyutlama Ön Test ve Son Test Puanlarına İlişkin Çarpıklık, Basıklık ve Normallik Testi İstatistikleri* 116
Tablo 56 *Öğrencilerin Betimsel Soyutlama Ön Test Puanlarının Mann-Whitney U Testi ile Karşılaştırılması* 117
Tablo 57 *Uygulama Öncesi ve Sonrası Grup İçi Betimsel Soyutlama Test Puanlarının Değişiminin Wilcoxon İşaretli Sıralar Testi ile Analizi* 117
Tablo 58 *Akademik Başarı Ortalaması, Betimsel Soyutlama Ön Test ve Betimsel Soyutlama Son Test Puanları Arasındaki İlişkilerin İncelenmesi* 119
Tablo 59 *Deney ve Karşılaştırma Grubu Öğrencilerinin Betimsel Soyutlama Son Test Puanlarının Mann-Whitney U Testi ile Karşılaştırılması* 119
Tablo 60 *Problem Çözme Ölçeği Ön Test ve Son Test Puanlarına İlişkin Çarpıklık, Basıklık ve Normallik Testi İstatistikleri* 121
Tablo 61 *Uygulama Öncesi Gruplar arası Problem Çözme Ölçeği Ön Test Puanlarının Karşılaştırılması* 122
Tablo 62 *Problem Çözme Ölçeği Faktör Puanlarına İlişkin Ön Test Sonuçlarına İlişkin Betimsel İstatistikler* 123
Tablo 63 *Gruplara Göre Öğrenenlerin Uygulama Öncesi ve Sonrası Problem Çözme Becerisi Algılarının Değişimi* 124
Tablo 64 *Problem Çözme Ölçeği Son Test Puanlarına İlişkin Grup Ortalama ve Düzeltilmiş Ortalama Puanları* 124
Tablo 65 *Deney ve Karşılaştırma Grubu Öğrencilerinin Problem Çözme Ölçeği Son Test Puanlarının ANCOVA ile Karşılaştırılması* 125
Tablo 66 *Problem Çözme Ölçeği Faktör Puanlarına İlişkin Son Test Sonuçlarına İlişkin Betimsel İstatistikler* 126

Tablo 67 *Öğretim Materyalleri Motivasyon Ölçeği Puanlarına İlişkin Çarpıklık, Basıklık ve Normallik Testi İstatistikleri*..... 127
Tablo 68 *Uygulama Sonrası Gruplar Arası Motivasyon Puanlarının t Testiyle Karşılaştırılması*..... 128
Tablo 69 *Öğretim Materyalleri Motivasyon Ölçeği Puanlarının Alt Faktörlere Göre İncelenmesi* 129
Tablo 70 *Durum çalışmalarına göre Programlama Başarısı, Problem Çözme Becerisi Algısı, Formal ve Betimsel Soyutlama Becerisi ve Motivasyon Arasındaki İlişkiler* 130

Şekiller Dizini

Şekil 1. Programlama aracılığıyla özünde birden çok düzeyde soyutlamalar barındıran bir problem çözme yöntemi olan bilgisayımsal düşünme süreçlerinin teşvik edilmesi 3
Şekil 2. Bilgisayımsal düşünme ile ilgili alanyazın inceleme süreci 56
Şekil 3. Karma araştırma desenleri 76
Şekil 4. Durum çalışmalarına katılan deney (a, c) ve karşılaştırma (b, d) grubu öğrencileri..... 79
Şekil 5. Durum Çalışması-1'de gerçekleştirilen araştırma yöntemi ve veri toplama süreçleri..... 83
Şekil 6. Durum Çalışması-2'de gerçekleştirilen araştırma yöntemi ve veri toplama süreçleri..... 83
Şekil 7. Eğitsel robot hareketlerinin matematiksel esaslara göre modellenmesi. 133
Şekil 8. Öğrencilerin koşul ve döngü işlemlerini karıştırması..... 154

Simgeler ve Kısaltmalar Dizini

BİT: Bilgi ve İletişim Teknolojileri

BD: Bilgisayımsal Düşünme

BT: Bilişim Teknolojileri

CSTA: Computer Science Teachers Association (Bilgisayar Bilimi Öğretmenleri Birliği)

ISTE: International Society for Technology in Education (Uluslararası Eğitim Teknolojileri Birliği)

NGSS: Next Generation Science Standards (Gelecek Nesil Bilim Standartları)

η^2: Eta-kare

p: Anlamlılık Düzeyi

sd: Serbestlik Derecesi

SS: Standart Sapma

$\overline{X}$: Ortalama

2B: İki boyutlu

3B: Üç boyutlu

Bölüm 1
Giriş

Bilgisayarların insanın düşünme ve öğrenmesini nasıl etkileyebileceği konusunu ilk tartışmaya açanlardan olan Seymour Papert, bilgisayarların bilgiye erişim kalıplarını değiştirmesi ile düşünmeyi geliştirmesi arasındaki ayrıma dikkat çekmiştir. Papert (1980) bilgisayarların fiziksel bağlantılarından oldukça uzaklaştırıldığında dahi insanların nasıl düşündüğünü etkilemek suretiyle düşünme süreçlerine kavramsal açıdan katkıda bulunabileceğini ileri sürmüştür. Yenilikçi olmasına karşın 1980'li yıllarda hâkim olan geleneksel okul kültürü nedeniyle geniş ölçüde yayılamayan bilgisayar bilimi kavramlarının öğrencilerin entelektüel düşünme becerilerinin geliştirilmesinde kullanılabileceği görüşü Wing'in (2006, 2008) çalışmaları sayesinde 2000'li yılların ortalarından itibaren bilgisayımsal düşünme (computational thinking) kavramı ile yeniden popülerlik kazanmıştır.

Analitik bir düşünme türü olan Bilgisayımsal Düşünme (BD), bilgisayar bilimlerinin temel kavramlarından yararlanarak problemler çözmeyi, sistemler tasarlamayı ve insan davranışlarını anlamayı kapsamaktadır (Wing, 2006, 2008). BD'nin özü bir problem ile karşılaşıldığında bilgisayar bilimcisi gibi düşünmektir (Grover & Pea, 2013). Bununla birlikte BD sadece bilgisayar uzmanlarının değil, herkesin öğrenmek ve kullanmak için istekli olabileceği evrensel olarak uygulanabilir bir tutum ve beceri seti sunmaktadır (Wing, 2006). Aslında bir problem çözme stratejisi olan BD ile kastedilen şey bilgisayar programını yazan programcı gibi düşünmek ve bu düşünme becerisini farklı alanlarda problem çözmek amacıyla kullanmaktır. Wing (2006), BD'nin her çocuğun analitik düşünme yeteneğine eklenmesi gerektiğini savunmaktadır. BD tıpkı okuma, yazma ve aritmetik gibi 21.yy'da öğrenciler için kritik bir beceri olarak düşünülmektedir (Qualls & Sherrell, 2010; Wing, 2006).

BD, Bilgisayar Bilimleri alanının genişliğini yansıtan bir dizi zihinsel araç içermektedir. BD görünüşte zor bir probleme ilişkin uygun temsillerin oluşturulması, problemin sadeleştirme, içe yerleştirme, dönüşüm ya da simülasyon gibi yollarla izlenebilir hale getirilmesi ve böylece nasıl çözüleceğinin bilindiği bir forma kavuşturulmasına katkı sağlayabilir. Belirli bir problemi çözmek zorunda olduğumuzda şunu sorabiliriz: "Bu problem ne kadar zor?" ve "bu problemi çözmek

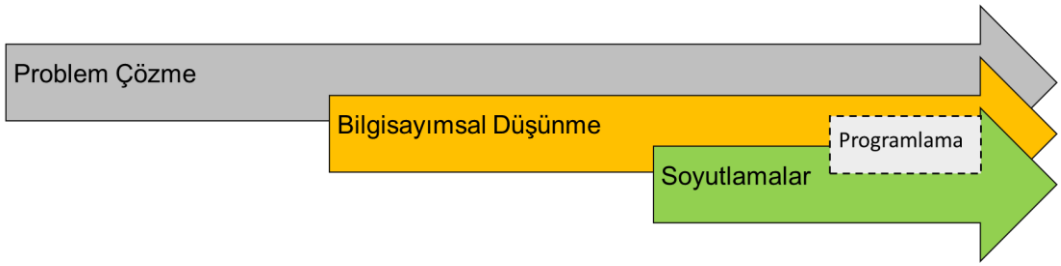
için en iyi yol nedir?". Bilgisayar bilimleri bu tür sorulara kesin ve net yanıtlar vermek için sağlam kuramsal temellere sahiptir. İster zor bir problemin üstesinden gelmek ister büyük ve karmaşık bir sistem tasarlamak için işe koşulsun, bilgisayımsal modellerin özünde soyutlamalar bulunmaktadır. Soyutlama genel anlamda gereksiz ya da ilişkisiz ayrıntıların geri plana atılarak temel karakteristikler üzerine kurulu genelleyici çıkarımların elde edildiği kavramsal bir süreçtir.

Soyutlama, BD'nin temel taşı olup, onu diğer düşünme türlerinden ayıran en önemli farklılıktır. Özünde bir model oluşturma olan soyutlama süreci mantıksal ve fiziksel perspektifleri ayırmak suretiyle karmaşık problemlerin çözümüne olanak sağlar. Bu nedenle soyutlama karmaşıklıkla başa çıkmak için bir anahtardır. Soyutlama aslında günlük hayatta rutin olarak işe koşulan bir beceridir. Ayrıca birçok alanın özünde çeşitli düzeylerde soyutlamalar bulunmaktadır (Colburn & Shute, 2007). Soyutlama kavramı matematik, bilişsel bilimler, yapay zekâ, sanat, felsefe, karmaşık sistemler ve bilgisayar bilimlerini kapsayacak şekilde çok geniş bir yelpazedeki disiplinlerde formal ya da informal olarak kullanılmaktadır. Örneğin coğrafya dersini ele aldığımızda paralel, meridyen, ekliptik düzlem vb. konular soyutlamalar üzerine kuruludur. Bu soyut temsiller özellikle denizcilik alanındaki seyir hesaplamalarının esasını oluşturmaktadır. Hemen her disiplin içinde kendine yer edinmiş soyutlama için genel bir tanım yapmak oldukça zordur. Çünkü soyutlama farklı disiplinlerde farklı şekillerde tanımlanmaktadır. Görsel sanatlarda soyutlama gerçekliğin imgelem yoluyla yeniden şekillendirilmesi ve sembollerle dışa vurulması, matematikte ise nesnelerin özellikleri ve/veya aralarındaki ilişkilerin kendilerinden ayrıştırılması ve matematiksel bir yapının geliştirilmesi için organize edilmesi şeklinde ifade edilmektedir. Bu çalışmada soyutlama Hill, Houle, Merritt ve Stix (2008) tarafından önerildiği gibi "sadeleştirme yapmak için detayları ortadan kaldırmak ve ana hatları gösteren genellemeler türetmek" olarak ele alınmıştır. Bilgisayar bilimleriyle ilişki olan bu tanımdan hareketle soyutlamanın birbirini tamamlayan iki boyuttan oluşan bir yapı olduğu görülmektedir. Soyutlama her ne kadar gelişimsel aşamalara oldukça bağlı olsa da öğretimsel süreçler yoluyla geliştirilebilecek bir beceridir.

Bilgisayar bilimleri soyutlama olarak karakterize edilebilecek etkinlikler yönünden zengin referanslara sahiptir. BD'nin öneminin anlaşılması ve eğitim ortamlarına entegrasyonu üzerine geniş bir fikir birliğini takiben son zamanlarda BD

gelişiminin nasıl destekleneceği ve değerlendirileceğine ilişkin daha pratik sorular ön plana çıkmıştır. Programlama sadece BD'nin temel bir becerisi olmayıp, BD yeterliklerinin sergilenmesini de sağlayan önemli bir etkinliktir. Birçok farklı beceriyi gerektiren BD için programlama becerileri anahtar roldedir. Çünkü programlama insanın zihinsel gücü ve bilgi işleme kapasitesinin bir kombinasyonunu gerektiren problemlerin çözümü için anahtar niteliğinde yeni bir düşünme yolunun geliştirilmesini teşvik etmektedir.

Bu çalışmada bir problem çözme biçimi olan BD süreçlerinin ortaya çıkarılması için programlama bir yol olarak düşünülmüştür (Lye & Koh, 2014). Böylece öğrencilerin soyutlamalardan yararlanarak karmaşıklıkla başa çıkmaları ve problem çözme becerilerini geliştirmeleri amaçlanmıştır (Şekil-1).



Şekil 1. Programlama aracılığıyla özünde birden çok düzeyde soyutlamalar barındıran bir problem çözme yöntemi olan bilgisayımsal düşünme süreçlerinin teşvik edilmesi

Öğrenenlere bilgisayımsal kavram, pratik ve perspektiflerin etkili bir biçimde nasıl kazandırılacağı noktasında farklı araç, ortam ve pedagojiler kullanılmaktadır. Çalışmada eğitsel robotlar bilgisayımsal problem çözme ve programlama etkinlikleri için hem eğitsel bir yenilik hem de geleneksel programlama eğitimine bir alternatif olarak önerilmiştir. Bu alternatif programlama etkinliğinin programlama başarısı, soyutlama, problem çözme ve motivasyon üzerine olan etkileri incelenmiştir.

Problem Durumu

Bilgisayar programlama öğretiminde karşılaşılan güçlüklerin çoğu, geleneksel bilgisayar bilimi öğretimindeki görevlerin yapısı ve bunların temsil biçiminden kaynaklanmaktadır (Ben-Ari, 2001; diSessa, 2001; Guzdial & Forte, 2005). Programlama öğretiminin geleneksel yolu öğrencilerden basit bir program oluşturmak için belli bir programlama diline özgü komutları ezberlemelerinin ve bu dile ilişkin sözdizimsel kurallara uymalarının istenmesidir (Chang, 2014).

Ülkemizde mesleki ve teknik ortaöğretim kurumlarının bilişim teknolojileri alanında verilen programlamaya giriş derslerinde C# ya da Visual Basic gibi profesyonel yazılım geliştirme araçları kullanılmaktadır. Bu durum özellikle programlamaya yeni başlayan öğrenciler için birçok zorluğu beraberinde getirmektedir. Programlama ortamının karmaşık arayüzü, nesne ya da kod bloklarının fazlalığı ve oldukça prosedürel kod yazım kuralları programlamanın anlamsal ve sözdizimsel ilişkisinin kurulmasını güçleştirmekte, dolayısıyla bilgisayımsal becerilerin sergilenmesini sınırlamaktadır. Bu durum öğrencilerin programlama başarısı, öğrenme uğraşı (katılım), motivasyonu ve bilgisayar bilimleriyle ilgili kariyer seçimlerini olumsuz etkilemektedir (Felleisen, Findler, Flatt & Krishnamurthi, 2004). Geleneksel programlama öğretimi öğrencilerin bilişsel rolünü göz ardı etmekte ve bu yönüyle de yenilikçi ve yaratıcı düşünme becerilerinin gelişimini desteklemekte yetersiz kalmaktadır (Fang, Chen, Cai, Cui & Harn, 2017).

Bilgisayar bilimleri ve programlama eğitimine ilişkin geçtiğimiz yarım asrı aşkın bir dönemden kalan oldukça geniş bir alanyazın bulunmaktadır. Bununla birlikte bu araştırmaların daha ziyade lisans sınıfları bağlamında olduğu görülmektedir. Son zamanlarda dünya genelinde BD (Bilgisayımsal Düşünme) ve programlama etkinlikleri dâhil olmak üzere bilgisayar bilimlerinin K12 müfredatına entegre edilmesine ilişkin artan girişimler bulunmaktadır. Bununla birlikte BD süreçlerinin nasıl yönlendirilmesi gerektiği, süreç sonunda elde edilecek çıktıların neler olduğu ve bu çıktıların nasıl değerlendirileceğine ilişkin araştırmalar oldukça sınırlıdır (Grover & Pea, 2013; Lye & Koh, 2014).

Bu çalışmada eğitsel robot programlama alternatif bir programlama etkinliği olarak önerilmiştir. Eğitsel robotlar oldukça kavramsal olan programlamayı otantik dünya düzenlemeleri içerisinde somutlaştırmakta ve program kodlarının anlamsal ve sözdizimsel ilişkisinin kurulmasını kolaylaştırmaktadır. Mevcut çalışmada öğrencilerin bilgisayımsal problemlere çözüm üretmek amacıyla gerçekleştirdikleri programlama etkinlikleri aracılığıyla yaşadıkları BD deneyimleri derinlemesine incelenerek bu süreç içerisinde öne çıkan bilişsel unsurlar ve karşılaşılan zorluklar derinlemesine incelenmiştir. BD becerilerinin daha etkili bir şekilde sergilenmesi amacıyla öğrencilerin problem çözme süreçlerini Polya (1957, 1973) tarafından önerilen problem çözme yöntemi rehberliğinde sürdürmesi teşvik edilmiştir. Böylece problem çözme sürecinin desteklenmesi sağlanmıştır. Çalışma alanyazında

nispeten daha az temsil edilen bir katılımcı grubu olan mesleki ve teknik ortaöğretim öğrencileriyle gerçekleştirilmiştir.

Araştırmanın Amacı ve Önemi

Bu çalışmanın amacı K12 düzeyindeki bilişim derslerinde eğitsel robot ve nesne yönelimli görsel programlama etkinliklerinin programlama başarısı, soyutlama, problem çözme ve motivasyon üzerindeki etkisini karşılaştırmalı olarak incelemektir. Programlama eğitimi boyunca öğrenenlerin BD süreçlerinin derinlemesine incelenmesi, bu süreçte öne çıkan bilişsel unsurların ve karşılaşılan zorlukların programlama eğitiminin veriliş yöntemine göre farklılaşan ve benzeşen yönlerinin tespit edilmesi de amaçlanmaktadır. Çalışmada ayrıca etkili bir problem çözme yaklaşımı olarak BD'nin öğretimsel pratikler içerisine nasıl entegre edilebileceğine ilişkin çıkarımların da elde edilmesi hedeflenmektedir.

BD diğer konu alanlarına transfer edilip uygulanabilecek bir problem çözme yöntemidir (Barr & Stephenson, 2011). Programlama aracılığıyla yürütülen bilgisayımsal etkinlikler öğrencilerin entelektüel düşünme ve problem çözme becerilerini geliştirmek amacıyla kullanılabilir (Feurzeig, Papert & Lawler, 2011; Papert, 1993). Bununla birlikte tıpkı diğer disiplinlerde olduğu gibi bilgisayar bilimlerinde de konu alanına ilişkin akıl yürütmenin ön koşulu, özünde soyutlamaların bulunduğu uygun zihinsel modeller kurmaktır. Son zamanlarda başta BD olmak üzere bilgisayar bilimi öğretimine ilişkin artan bir eğilim olmasına rağmen bu düşünme türünün temel bileşenlerinden olan soyutlama üzerine oldukça az sayıda çalışma bulunmaktadır. Ayrıca çalışmaların büyük bölümünün nispeten kısa süreli öğretimsel etkinlikler ve dar kapsamlı bilgisayımsal kavramlar etrafında şekillendirilen tek bir kesitsel incelemeye dayalı olduğu görülmektedir. Bu araştırma başta soyutlama olmak üzere BD ve bilgisayımsal problem çözme süreçlerinin doğasına ilişkin mevcut anlayışlara katkı sağlaması bakımından önemlidir. Ayrıca yürütülen ardışık iki durum çalışması elde edilen çıkarımların tutarlılığını inceleme ve durumsallıkları geniş bir perspektifte betimleme fırsatı vermektedir. Çalışmanın BD'ye katkı sağlayacak stratejiler geliştirilmesine yardımcı olacağı düşünülmektedir. Programlamaya giriş dersleri için müfredat ve öğretim tasarım pratiklerini içeren mevcut araştırmanın ne yapılacağından ziyade nasıl yapılacağı bilgisini sunarak öğretimsel pratiklere rehberlik sağlaması beklenmektedir. Ek olarak, mevcut çalışma öğrenci yansımalarından yola çıkarak BD etkinliklerinde öğrencilerin ihtiyaç

duydukları destek unsurlarının programlama modalitesine göre adreslenmesi bakımından önemlidir. Her ne kadar K12 seviyesindeki mesleki öğretim bağlamında gerçekleştirilmiş olsa da araştırma sonuçlarının diğer okul türleri, eğitim kademeleri ve öğrenme disiplinleri için de değerli çıkarımlar sağlayacağı düşünülmektedir.

Araştırma Problemi

Nesneye yönelik ve robot programlama etkinliklerinin programlama başarısı, soyutlama, problem çözme ve motivasyon üzerindeki etkisi nedir?

Alt problemler. Araştırma problemine ait alt problemler şu şekilde sıralanmıştır:

- Nesne yönelimli görsel programlama ile karşılaştırıldığında eğitsel robot programlama etkinlikleri programlama başarısı, soyutlama, problem çözme ve motivasyon üzerinde anlamlı bir farklılık yaratmakta mıdır?
- Bilgisayımsal problem çözme sürecinde öne çıkan bilişsel işlemler veya unsurlar nelerdir?
- Öğrencilerin programlama aracılığıyla bilgisayımsal problem çözme etkinlikleri esnasında karşılaştıkları engeller nelerdir?

Sayıltılar

Katılımcıların öz raporlamaya dayalı veri toplama araçlarını (ders günlükleri ve ölçekler) doğru ve samimi şekilde doldurduğu kabul edilmiştir. Öğretim materyalleri motivasyon anketinin programlama araçlarına karşı motivasyon algısını, problem çözme ölçeğinin günlük yaşamdaki problemlere yaklaşım şeklini, programlama başarı testinin programlama bilgisini, formal ve betimsel soyutlama testlerinin sırasıyla formal ve betimsel soyutlama becerilerini ölçtüğü kabul edilmiştir.

Çalışmada sınıfların her birinden test gruplarına seçkisiz olarak yapılan atamalar neticesinde oluşturulan deney ve karşılaştırma gruplarının ders materyallerine karşı motivasyonlarının uygulama öncesinde aynı seviyede olduğu kabul edilmiştir. Uygulama sürecinde yaşanan katılımcı kaybının deney ve karşılaştırma gruplarında önemli bir farklılık oluşturmadığı kabul edilmiştir.

Sınırlılıklar

Karma araştırma yönteminin benimsendiği çalışmalarda nicel ve nitel veri kümesi birleştirilirken farklı katılımcı özelliklerine ve örneklem büyüklüğüne sahip olmanın sonuçları dikkate alınmalıdır. Bu çalışmada birbirinden ayrık olarak yürütülen iki durum çalışması arasında örneklem büyüklüğü açısından farklılık bulunmaktadır. Nicel veriler daha genellenebilir sonuçların üretilmesi; nitel veriler ise üzerinde çalışılan olgunun daha derinlemesine incelenmesi için toplanmıştır.

Bir araştırmanın katılımcı grubu ne kadar dikkatli ve özenli seçilirse seçilsin çalışma ilerledikçe bazı katılımcıların çalışmadan ayrılması olasıdır. Bu durum özellikle de uygulama süresi uzun olan deneysel çalışmalar için geçerlidir, çünkü zaman içinde çeşitli nedenlerden ötürü bazı katılımcıların çalışmadan ayrılması olasıdır. Katılımcı kaybı da denilen bu durum iç güvenliği tehdit eden önemli bir araştırma olgusudur. Katılımcı kaybı iç geçerliğe yönelik tehditler arasında muhtemelen kontrol edilmesi en zor olanıdır (Fraenkel & Wallen, 2006). Katılımcı kaybının bertaraf edilmesi için en yaygın yanlış anlayış çalışmadan ayrılan katılımcıların yerine çalışmaya yeni katılımcıların dâhil edilmesi önlemidir. Bu çalışmada çalışmadan ayrılan katılımcılar bir sınırlılık oluşturmaktadır. Gerçekleştirilen ilk uygulamada katılımcıların %14'ü; ikinci uygulamada %13'ü çeşitli gerekçelerle çalışmadan çıkarılmıştır. Bu katılımcıların çıkarılma gerekçeleri şu şekildedir:

Durum çalışması 1'de;

- 1 öğrenciye özel eğitim gereksinimi nedeniyle bireyselleştirilmiş eğitim planı uygulandığı için bu katılımcıdan elde edilen veriler analiz dışında tutulmuştur.
- 3 katılımcı farklı okullara nakil olmaları nedeniyle çalışmadan çıkarılmıştır.
- Devamsızlık süresini aşan 3 öğrenci uygulamayı yarıda bırakmıştır.

Durum çalışması 2'de;

- 1 öğrenciye özel eğitim gereksinimi nedeniyle bireyselleştirilmiş eğitim planı uygulandığı için bu katılımcıdan elde edilen veriler analiz dışında tutulmuştur.

- 2 katılımcı farklı okullara nakil olmaları nedeniyle veri kümesinden çıkarılmıştır.
- Devamsızlık süresini aşan 3 öğrenci uygulamayı yarıda bırakmıştır.

Ortak sınav tarihlerinde sınav yapılması durumunda programlama etkinliklerine ilgili ders saatinde ara verilmiştir. Ayrıca bazı öğrencilerin okul nöbetçisi olduğu günlerde uygulamaya katılamamaları da bir sınırlılık oluşturmaktadır.

Tanımlar

Bilgisayımsal düşünme: BD, bilgisayar biliminin temel kavramlarından yararlanarak problem çözmeyi, sistemleri tasarlamayı ve insan davranışlarını anlamayı kapsayan bir problem çözme sürecidir (Wing, 2006).

Programlama: Programlama bilgisayarın bir görevi yerine getirmesini sağlayan bir dizi komuttan oluşan ve yeniden kullanılabilir bir ürünün geliştirilmesine yönelik olan faaliyetler kümesi olarak tanımlanabilir (Pea & Kurland, 1984). Bilgisayar programlama, bir anlamda bilgisayara bir dizi talimatla ne yapılacağını anlatma sanatıdır. Bilgisayarı programlamak BD'yi gerektiren otantik bir öğrenme etkinliğidir (Daily & Eugene, 2013).

Öğrenci başarısı: Öğrenci başarısı belirli bir alandaki mevcut öğrenmelerinin düzeyiyle ilişkilidir. Başarının nasıl ele alındığı ve karakterize edildiği neyin öğretileceği ve bunun nasıl gerçekleştirileceğinin en önemli belirleyicisidir. Mevcut çalışmada öğrenci başarısı programlama başarı testinden alınan puan olarak tanımlanmıştır.

Soyutlama: Soyutlama birbirini tamamlayan iki boyuttan oluşan bir yetenektir. Bunlar sadeleştirme yapmak için detayları ortadan kaldırmak ve ana hatları gösteren genellemeler türetmektir (Hill vd., 2008). Soyutlama bir anlamda karmaşıklık karşısında basitliği sezme (fark etme) ve oluşturma becerisidir. Soyutlama üst düzey düşünme (high-order thinking) becerileri için gerekli bilişsel bir beceridir.

Problem çözme: Kişinin günlük yaşamda karşılaştığı sorunlu durumlarda etkili başa çıkma yolunu bulabilmek için geliştirdiği ve ürettiği bilişsel, duyuşsal ve davranışsal bir süreçtir.

Motivasyon: Motivasyon bireyin bir amaca yönelik gösterdiği çabanın sürekli olma eğilimidir (Ngan & Law, 2015). Bu çalışmada hipotetik bir yapı olan motivasyon, öğretimsel etkinliklere katılma ve bu etkinlikleri sürdürme konusunda istekli olma eğilimi olarak ele alınmıştır. Çalışmada öğrencilerden bilgisayar bilimi kavramlarına aşina olmaları için eğitsel etkinliklere katılmaları, sunulan programlama ortamları ve araçları aracılığıyla BD problemlerine çözüm üretmeleri ve bu yolla gerçekleştirdikleri BD pratiklerinde BD perspektiflerini yansıtmaları beklenmiştir. Motivasyon bilinçli ve istemli çaba gerektiren bir süreç olan öğrenme ile ilişkilidir (Bruner, 1960; Tüzün, 2004; Wlodkowski, 1978). Bu nedenle öğrencilerin öğrenme süreçleri üzerinde motivasyonun önemli bir etkisi olacağı düşünülerek çalışmaya dâhil edilmesi uygun bulunmuştur.

Bölüm 2
Araştırmanın Kuramsal Temeli ve İlgili Araştırmalar

Bilgisayımsal Düşünme

Papert (1980) bilgisayar programlamanın entelektüel modeller tarafından temellendirilebilecek koşullar sağlama potansiyeline dikkat çekerek; erken yaştaki çocukların programlama sayesinde gelişim aşamaları için oldukça soyut olduğu düşünülen kavramlar üzerinde hâkimiyet kurabileceklerini ileri sürmüştür. Papert'a göre öğrenciler programlama aracılığıyla fiziksel varlığına gerek olmaksızın bazı niteliksel özelliklere ilişkin sezgisel bir nicelik algısı kazanıp, soyutu somutlaştırabilir. Bork (1981) bilgisayar programlamanın bir eğitim aracı ya da eğitsel bir etkinlik olabileceğini, öğrencilerin programlama aracılığıyla geliştirdikleri analitik düşünce becerilerinin geniş yelpazedeki problemlere uygulanabileceğini belirtmiştir. En yararlı öğrenmenin "öğretilmeksizin öğrenme" olduğunu vurgulayan Papert (1980) bilgisayarların çocukların düşünme süreçlerini bilinçli hale getirmelerine olanak sağlayıp, daha etkili öğrenmelerine yardımcı olabileceğini belirtmiştir. Çocuklar yapmak istediklerini gerçekleştirmek için bilgisayarı programlarken, kendi başına nasıl bir şey yapabileceklerini düşünür ve böylece nasıl düşündüklerine ilişkin içsel bir yansıma geliştirir. Özetle, ne düşünülmeli yerine nasıl düşünülmesi gerektiğini teşvik eden programlama ortamları bilişsel süreçler için etkili bir öğretim aracı olma potansiyeli taşımaktadır.

Bilgisayımsal olarak düşünmeyi öğrenmek bilgisayar bilimleri alanında uzun zamandır önemli bir tema olarak kabul edilmektedir (Pellas & Peroutseas, 2016). Bilgisayımsal temsillerin bilgiyle yeni ilişkiler kurulması ve güçlü fikirlerin ortaya çıkarılması noktasında sağlayabileceklerine vurgu yapan Papert (1996) bilgisayımsal düşünme (BD) terimini kullanan ilk kişidir. Aslında öğrencilerin entelektüel düşünme becerilerini bilgisayar bilimi kavramlarıyla geliştirme çabaları matematik öğretimi için geliştirilen Logo programlamanın ortaya çıktığı 1960'lı yıllara dayanmaktadır (Feurzeig vd., 2011; Papert, 1980). Logo programlama dili, hareket etme ve dönme gibi eylemleri karmaşık yollarla birleştirmek suretiyle ekranda bulunan bir kaplumbağa nesnesine komutlar vererek karmaşık geometrik şekillerin üretilmesine olanak tanımaktadır (Papert, 1980). Gorman ve Bourne (1983) haftada 1 saat Logo programlama diliyle meşgul olan ilköğretim 3. sınıf öğrencilerinin haftada yarım saat programlama yapan öğrencilere kıyasla kural öğrenme testinde

daha üstün bir performans sergilediğini raporlamıştır. Öte yandan Pea ve Kurland (1984) Logo araştırmacılarının anekdotlardan biraz daha fazlasını temel aldıklarını işaret ederek Logo programlama deneyimlerinin planlama performansları üzerinde anlamlı bir etkisi olamayacağını öne sürmüştür. Pea ve Kurland (1984) bilgisayar programlama öğreniminin yüksek zihinsel işlevlerin geliştirilmesini teşvik edip etmediği konusundaki mevcut anlayışları tartışmaya açmıştır. Araştırmacılar çoğu kez programlama öğretiminin temel düşüncesi olan şeyin belirli bir programlama dilinin semantiğinin ve sözdiziminin öğrenilmesi olduğuna dikkat çekmiştir. Bu durumun yetişkin programcılar arasında bile yanlış anlaşılmalara yol açtığını, bunun iyi programcıların ne bildiğini ve nasıl yaptığını gösteren bilişsel çalışmalarla çelişebileceğini, sonuç olarak bu yaklaşımın kabul edilemez bir yanılgı olduğunu belirtmiştir. Ayrıca programlama öğrenimi ve transfer çıktılarına yönelik araştırmaların halen uzman programcıların ne yaptığına yönelik bilinenlerden yararlanılacak şekilde problem çözme ve görev analizi gibi geleneksel olmayan yöntemleri konu edinmediğinin altı çizilmiştir. Araştırmacılar tarafından eleştirilen ikinci yaygın beklenti ise programlama öğreniminden kazanılan yüksek bilişsel becerilerin kendiliğinden transferidir. Buna göre kendiliğinden transferi öğrenmede programlama becerisi düzeyinin ötesinde programlama ile transfer türleri arasında bağlantılar kurulmasında oluşabilecek dirençlerin dikkate alınması gerektiği belirtilmiştir. Mayer, Dyck ve Vilberg (1986) bir kişinin düşünme becerileri ile BASIC programlamayı öğrenme becerileri arasındaki ilişkiye dair sınırlı bir kanıt bulmuştur. İlgili çalışmada programlama öğreniminin daha ziyade programlama dili ile doğrudan ilişkili düşünme becerileri üzerinde olumlu etkileri olduğu belirtilmiştir. Araştırmacılar ayrıca sunulan programlama probleminin ya da prosedürünün kavranması ile genellenebilir beceri transferi arasında kurulan ilişkinin sorunsal doğasına dikkat çekmiştir. Düşünme becerileri ile programlama arasındaki ilişkileri araştırmanın en verimli yolunun, genel entelektüel yeteneğe odaklanmak yerine bilişsel görev analizleri aracılığıyla programlamaya özgü düşünme bileşenlerine odaklanmak olduğu belirtilmiştir (Mayer vd., 1986).

Logo programlama dilinin temel amacı öğrencilerin matematiksel ve mantıksal düşünme becerilerini geliştirmelerine katkı sağlamaktır. Logo programının arkasında yani kuramsal temelinde yapılandırmacılığın (constructivism) bir yorumu olarak kurmacılık (constructionism) ya da bir diğer ifadeyle inşacılık bulunmaktadır.

Yapılandırmacılık anlayışına göre bireyler bilgiyi deneyim yoluyla aktif olarak inşa etmektedir. Piaget (1954) fikirlerini şu şekilde açıklamıştır: "Bilmek nesneyi değiştirmek, dönüştürmek, bu dönüşüm sürecini ve sonunda bu nesnenin nasıl inşa edildiğini anlamaktır". Papert (1993) öğrencilerin aktif olarak bir ürün inşa etmek için mevcut araçları kullandığı kültürün sağladığı zengin öğrenme ortamlarının önemine vurgu yapmıştır. Öğrenciler bu ürünlerini paylaşabilir ve kendi açıklama ve tanımlamalarıyla başkalarıyla iletişim kurabilirler. Papert'a göre öğrenciler dışarıda olanı içselleştirmekte ve daha sonra inşa sürecinde içsel olanı dışa vurmaktadır. Köklerini yapılandırmacılıktan alan kurmacılık çocuğun bilgiyi deneyim yoluyla aktif olarak yapılandırdığı fikrini ortaya koymakta ve eğitimde yaparak öğrenme yaklaşımını savunmaktadır. Piaget'nin (1954) kuramı bilginin bir kişinin zihninde nasıl oluşturulduğunu açıklamak için geliştirilmişken, Papert (1980) bunu bilgisayar ve robotik kullanımı da dâhil olmak üzere dünyadaki nesneler tarafından desteklenen iç yapılar oluşturma yolları üzerine odaklanarak genişletmiştir. Jonassen (1995) teknolojinin bir bilgi inşa aracı olarak kullanılmasını önermiş ve öğrenenlerin ondan öğrenmek yerine teknoloji ile öğrenmesi gerektiğinin altını çizmiştir. Bununla birlikte Logo programlama dili, ortaya çıktığı yıllardaki geleneksel davranışçı okul kültürü nedeniyle gerekli ilgiyi görmemiştir (Agalianos, Noss & Whitty, 2001). Yenilikçi olmasına karşın geniş ölçüde yayılmayan BD, 2006 yılından itibaren yeniden popülerlik kazanmıştır. BD'nin, ister bir insan ister bir makine tarafından işletilsin, bilgisayımsal süreçlerin güç ve sınırları üzerine kurulu olduğunu belirten Wing (2006) bilgisayımsal yöntem ve modellerin insanların tek başına başa çıkamayacağı problemleri çözme ve sistemleri tasarlama olanağı sağladığına dikkati çekmiştir. diSessa (2001) bilgisayarların konular, bağlamlar ve akademik disiplinlere nüfuz etme potansiyeli olan yeni ve güçlü bir okuryazarlık formu için temel oluşturabileceğini ileri sürmüştür.

BD üzerine günümüze kadar yapılmış çalışmalarda ağırlıklı olarak BD'ye ilişkin tanımlayıcı hususlara yer verildiği görülmektedir (Kalelioğlu, Gülbahar & Kukul, 2016). Buna rağmen BD'nin üzerinde uzlaşılan genel bir tanımı bulunmamakta, çeşitli araştırmacılar tarafından farklı şekillerde tanımlanmaktadır (Grover & Pea, 2013; Leonard vd., 2016). Bu yüzden BD'nin ne olduğuna dair getirilen farklı yorumlamalar sıklıkla eleştirilmekte ve bu durum BD'nin bir disiplin olarak eğitimciler tarafından belirgin olarak anlaşılmasında sıkıntılara yol

açmaktadır (Grover & Pea, 2013). BD kavramının yeniden ilgi odağı olmasını sağlayan öncü çalışmasında Wing (2006) BD'yi bilgisayar bilimleri için temel kavramlardan yararlanarak problemler çözmek, sistemler tasarlamak ve insan davranışlarını anlamak olarak tanımlamıştır. Wing (2006), BD'nin ne olup ne olmadığına ilişkin olarak onun karakteristik özelliklerini şu şekilde listelemiştir:

- *Kavramsallaştırmadır, programlama değildir*. BD sanılanın aksine teknik bir jargon olmayıp, bir kavramsallaştırma biçimidir. BD bir bilgisayarı programlamaktan fazlasıdır. Birden fazla soyutlama düzeyinde düşünmeyi gerektirir.
- *Temeldir, ezber becerisi değildir.* BD mekanik bir rutinin izlenmesi değildir. Her insanın modern toplumda işlev göstermesi için bilmesi gereken temel bir beceridir.
- *Bilgisayarlara değil insanlara özgü bir düşünme biçimidir*. BD insanların problemleri çözdüğü bir yoldur; insanların bilgisayar gibi düşünmeye çalışması değildir.
- *Matematiksel ve mühendislik odaklı düşünme üzerine kuruludur*.
- *Düşüncelerdir, ürünler (donanım ya da yazılım araçları) değildir.* Problemlere yaklaşmak ve çözmek, günlük hayatı yönetmek ve diğer insanlarla iletişim ve etkileşim kurmak için kullanılan bilgisayımsal kavramlardır.
- *Herkes içindir, yani her alanda uygulanabilir.*

Wing (2006), BD'nin yalnızca programlama ya da kodlama olmadığını dolayısıyla sözdizimsel olarak belirli mekaniksel rutinlerin içerisine sıkıştırılamayacağını, aksine onun temelinde çok düzeyli soyutlamalar olan bir kavramsallaştırma süreci olduğunu belirtmektedir. Ayrıca onun matematik ve mühendislik alanlarıyla ilişkili olduğunu, üretime dönük ancak programlama ürünlerinin ötesinde düşüncelere odaklanan ve her alanda kullanılabilecek ve diğer insanlarla iletişimi ve işbirliğini teşvik eden bir problem çözme yaklaşımı olduğunu belirtmektedir. Wing (2011) daha sonra BD'nin tanımını daha operasyonel olacak şekilde güncellemiştir. Güncellenen tanıma göre BD, problemlerin ve çözümlerinin bir bilgi işlem ajanı tarafından etkin bir şekilde işletilebilecek bir temsil formuna kavuşmasını sağlamak üzere formüle edilmesinde yer alan düşünce süreçlerini

kapsamaktadır. Benzer şekilde Aho'ya (2012) göre BD, problemlerin ve çözümlerinin bilgisayımsal adımlar ve algoritmalar şeklinde temsilini içeren düşünme süreçleridir. Uluslararası Eğitim Teknolojileri Birliği (ISTE) ve Bilgisayar Bilimi Öğretmenleri Birliği (CSTA) operasyonel bir tanım geliştirmek üzere BD'yi altı karakteristik özellik etrafında çerçevelendirmiştir. Buna göre BD problemlerin bilgisayar ya da diğer araçların çözebileceği şekilde formüle edilmesini (1); verilerin mantıksal organizasyonu ve analizini (2), modeller ve simülasyonlar gibi soyutlamalar yoluyla verilerin temsilini (3); çözümün algoritmik düşünme yoluyla otomatikleştirilmesini (4); çözüm adımlarının ve kaynaklarının en etkili ve verimli kombinasyonlarını bir araya getirmek suretiyle olası çözümlerin analizini ve gerçekleştirilmesini (5), problem çözme sürecinin genelleştirmesini ve transferini (6) içeren bir problem çözme süreci olarak tanımlanmıştır (Barr, Harrison & Conery, 2011; Kalelioğlu vd., 2016). Israel, Pearson, Tapia, Wherfel ve Reese'e (2015) göre BD, öğrencilerin bilgisayarları kullanarak düşüncelerini modellediği ve programlar geliştirdiği bir süreçtir. BD, insanların zor problemleri kolayca çözülebilen alt görevlere indirmelerini, problemleri uygun şekilde temsil etmelerini, verileri yorumlamalarını, bir makine tarafından çalıştırılabilen algoritmalar oluşturmalarını, ortaya çıkan sonucun doğruluğunu, verimliliğini ve hatta estetikliğini göz önünde bulundurmalarını sağlayan bir dizi zihinsel işlem gerektirir (Chen vd., 2017).

ABD Ulusal Araştırma Konseyi, BD'nin etrafındaki tartışmanın çeşitliliğine ve kapsayıcı doğasına atıfta bulunarak 2010 yılında BD'nin kapsamı ve niteliğine ilişkin bir toplantı düzenlemiştir. Bu etkinliğin sonucunda BD'nin içerdiği 20'den fazla üst düzey beceri ya da etkinlik setini (problemi soyutlama ve ayrıştırma, sezgisel akıl yürütme, araştırma/arama stratejileri, özyineleme, makine öğrenmesi ve paralel işleme gibi bilgisayar bilimi kavram bilgisi vs.) listeleyen bir rapor üretilmiştir (US National Research Council, 2010). BD farklı hiyerarşik seviyelerde algoritmik düşünme, otomasyon, ayrıştırma, modelleme, örüntüleme, özyineleme, ölçekleme ve sembolik gösterim gibi bilgisayar bilimi yapılarının ve problem çözme stratejilerinin yaygınlığını kapsayan şemsiye bir kavram olarak kabul edilmektedir (Denning, 2009; Grover & Pea, 2013; Guzdial, 2008). Furber'e (2012) göre BD, bilgisayar bilimindeki araç ve teknikleri doğal ve yapay sistemleri/süreçleri anlamak ve onlar hakkında akıl yürütmek için kullanmaktır. Barr ve Stephenson'a (2011) göre BD, bir problemi uygun bir yol seçerek ifade etme, onun için bir model yaratma ve

en sonunda problemi çözmek için en uygun stratejiyi seçme prosedürüdür. Zhong, Wang, Chen ve Li (2016) BD'nin en öz tanımlamayla bir problem çözme süreci olduğunu belirterek, yapılan tüm tanımlamaların BD ile ilgili bakış açılarını genişletirken bu çekirdek vurguyu değiştirmediğini belirtmiştir.

Bu tanımlamalar ışığında BD'nin Bilgisayar Bilimleri (kavram, teknik ve yaklaşım) üzerine temellendirilen özünde çok düzeyli soyutlamalar barındıran düşünme süreçlerini kapsadığı ve her alanda kullanılabilecek bir problem çözme yaklaşımı olduğu söylenebilir.

Etimolojik Bir Karmaşa: Bilgisayım ve Bilgisayımsal Düşünme

İngilizcesi "computational thinking" olan kavramın Türkçe karşılığı için çok farklı terimsel karşılıklar önerilmiştir (Tablo 1).

Tablo 1

Alanyazında "Computational Thinking" Kavramı İçin Önerilen Türkçe Kavram Karşılıkları

Önerilen Türkçe kavramsal karşılıklar	Yazarlar
Bilgi-işlemsel düşünme	Demir ve Seferoğlu (2017); Gülbahar (2018); Numanoğlu ve Keser (2017); Uslu, Mumcu ve Eğin (2018)
Bilişimsel düşünme	Yıldız, Çiftçi ve Karal (2017)
Bilgisayarca düşünme	Korkmaz, Çakır ve Özden (2016); Korkmaz, Çakır, Özden, Oluk ve Sarıoğlu (2015)
Hesaplamalı düşünme	Özçınar (2017); Özçınar ve Öztürk (2018)
Kompütasyonel düşünme	Şahiner ve Kert (2016)

Bilgi-işlemsel düşünme, bilişimsel düşünme, bilgisayarca düşünme, hesaplamalı düşünme, kompütasyonel düşünme bunlardan en bilinenleridir. Benzer bir Türkçe isim karmaşası 1960'lı yıllarda "computer" kelimesinin Türkçe karşılığı için yaşanmıştır. "Computer" kelimesinin Türkçe karşılığı için kompüter ve elektronik beyin gibi ifadeler önerilmiştir. Bilgisayar kelimesi ilk kez 1969 yılında Hacettepe Üniversitesi tarafından verilen bir gazete ilanında kullanılmıştır. Prof. Dr. Aydın Köksal tarafından önerilen bu ifade yaygın olarak kabul görmüş ve computer kelimesinin karşılığı olarak Türkçeye yerleşmiştir (Köksal, 1981, 2010). Computer kelimesinin bilgisayar olarak dilimize yerleştiği göz önünde bulundurularak etimolojik açıdan daha verimli sonuçlar elde etmek amacıyla compute eyleminin bilgisaymak,

computation isminin bilgisayım ve computational sıfatının da bilgisayımsal olarak kullanılmasının uygun olacağı düşünülmüştür. Böylece mevcut tez çalışmasında computational thinking kavramının Türkçe karşılığı olarak bilgisayımsal düşünme ifadesi kullanılmıştır.

Bilgisayımsal Okuryazarlık, Bilgisayımsal Düşünme, Programlama ve Kodlama Kavramları Arasındaki Farklılıklar

Çok sayıda araştırmacı, politika yapıcı ve uluslararası sivil toplum kuruluşu bilgisayımsal okuryazarlığın 21. yy. öğrenme becerileri içinde hayati bir rolü olduğunu belirtse de bu kavramın ne olduğu konusunda halen bir belirsizlik bulunmaktadır (Lye & Koh, 2014). Bilgisayımsal okuryazarlığın önemli bir yönü iterasyon, özyineleme, sistematik prosedürler ve modüleştirme gibi programlamanın temel kavramlarına ilişkin sağlam bir anlayış içerisinde bilgisayarı bir araç olarak kullanabilme becerisidir (Berland, Martin, Benton, Smith & Davis, 2013). diSessa (2001) bilgisayımsal okuryazarlığı materyal, sosyal ve bilişsel olmak üzere üç boyutta ele almaktadır. Materyal boyutu bilgisayımsal ürünler ortaya çıkarma olarak tanımlanmaktadır. Karmaşık bir Excel tablosunun hazırlanması ya da bir algoritmanın hazırlanması gibi birçok biçimde ortaya çıkan bu boyut basitçe bilgisayımsal ürünler inşa etmek için program yazma becerisi olarak ifade edilebilir (Berland, 2016). Bilgisayımsal okuryazarlığın sosyal boyutu bilgisayımsal ürünlere neden ihtiyaç duyulduğunu anlama ve açıklama yeteneğidir. Bu boyut bir bilgisayımsal ürünün kimin için önemli olduğu ve nasıl işletileceğiyle ilgilidir. Bilgisayar programının düzgün bir biçimde ifade edilmesi öğretim, konuşma ve teknik becerilerin örtüşmesini gerektirir. Çoğunlukla BD olarak da adlandırılan bilişsel boyut ise kısmen bilgisayımsal okuryazarlığın materyal ve sosyal boyutunun ilişkisinden ortaya çıkmaktadır. Bilişsel boyut bilgisayımın yapabildikleri ve yapamadıkları, niçin ve nerede yararlı olduğu ve bir araç olarak nasıl kullanılabileceğine dayalı olarak problemler, bağlamlar ve dünya hakkında düşünme yeteneği olarak tanımlanabilir.

Alanyazın incelendiğinde temellerini bilgisayar bilimlerinden alan bu akademik disiplini tanımlamak için bilgisayım, BD, kodlama ve bilgisayar programlama dâhil olmak üzere birçok terimin kullanıldığı görülmektedir (Snodgrass, Israel & Reese, 2016). Terminoloji ile ilgili yaygın bir fikir birliği bulunmamasına rağmen, BD genellikle problem-çözme, sistemler tasarlama ve

bilgisayımsal yollarla düşünerek çözümler üretmeyi içeren bir dizi bilgisayımsal deneyimleri izah etmek için kullanılmaktadır (Mouza, Yang, Pan, Ozden & Pollock, 2017; Snodgrass vd., 2016; Wing, 2006). Tıpkı okuryazarlığın yazmadan fazlası olduğu gibi BD de programlamadan fazlasıdır.

BD'nin popüler hale gelmesiyle birlikte programlama ve kodlama kavramlarının iç içe geçtiği ve söz konusu kavramların bilgisayar bilimlerindeki terim karşılıklarının da karmaşık bir hal aldığı görülmektedir. Son dönemde bilgisayar bilimlerine ve programlamaya yeni başlayanlara programlamanın zor ve korkutucu değil; aksine eğlenceli bir etkinlik olduğu izlenimini vermek adına programlama yerine "kodlama" teriminin kullanıldığı görülmektedir.

Bir program belirli bir amaç için bir araya getirilen ve makineler tarafından işletilen bir kod bloğudur. Diğer bir ifadeyle bilgisayarın programcı tarafından verilen kodları belirtilen bir sıraya göre yerine getirdiği bir yönergeler dizisidir (Stair & Reynolds, 2013). Programlama bir problemin derinlemesine analiz edilmesini, ona ilişkin bir mantık geliştirilmesini; daha sonra matematiksel ve mantıksal ilkeler kullanılarak problemin çözülmesini gerektirir. Bu süreçte programcılar kara tahtaları, kâğıtları, taslakları ya da krokileri kullanarak geliştirdiği mantığı ortaya koyar, tasavvur ettiklerinin nasıl yürütüleceğine ilişkin bir yordama yapar. Öte yandan kodlama, geliştirilen mantığın gerçek koda çevrilmesi sürecidir. Kodlama daha ziyade dil (programlama dili) odaklıdır. Bu açıdan bakıldığında programlama kodlamadan fazlasıdır. Çünkü programlama öğrencilerin soyutlama ve ayrıştırma gibi bileşenleri kapsayan BD becerilerinin sergilenmesini gerektirir.

Bilgisayımın Eğitim Ortamlarına Entegrasyonu

Bilgisayım günlük hayatta hemen her alana yayılmakta ve küreselleşen dünyada ekonomik rekabeti sürdürmek için önemli görülmektedir (Bundy, 2007). Dahası bilgisayım doğada kendiliğinden gerçekleşen bilgi işleme süreçlerini anlamak ve kontrol etmek için gereklidir. Öğrencilerin yaratıcılık, eleştirel düşünme ve problem çözme dâhil olmak üzere 21. yy becerilerinin geliştirilmesi küreselleşen ve hiper bağlantılı günümüz toplumlarının önde gelen kaygılarındandır (Gretter & Yadav, 2016). BD, 21. yy'da öğrenciler için kritik becerilerden birisi olarak kabul edilmektedir. Medya teknolojileri ve endüstrideki son gelişmelerle birlikte, programlama yapma herkes için daha yaygın bir beceri olma yolunda ilerlerken,

algoritmaları tanıma ve anlama 21. yy. okuryazarlığı için dördüncü "r" (reading, ‘riting, ‘rithmetic, ve ‘rithms) olma yolunda küresel olarak ivme kazanmıştır (Grover & Pea, 2013). Alanyazın incelendiğinde özellikle Wing’in 2006 yılındaki çalışmasından sonra bilgisayımsal beceriler üzerine artan bir ilgi olduğu görülmektedir. Bununla birlikte BD kavramı yaygınlaştıkça bu düşünme becerisinin nasıl geliştiğine, nasıl destekleneceğine ve değerlendirileceğine ilişkin daha pratik sorular ön plana çıkmaya başlamıştır. BD’ye ilişkin tanımlamaların ve anlayışların genişletilmesini takiben BD üzerine son çalışmaların pedagojik çıktılar üzerine odaklandığı görülmektedir. ABD Ulusal Araştırma Konseyi bu durumu takiben BD'nin pedagojik yönlerine odaklanan ikinci bir toplantı gerçekleştirerek BD'nin mevcut alanlarla nasıl ilişkilendirileceği, bir BD sürecinin neye benzeyebileceği, BD pratiğinde öğretmenlerin nasıl eğitileceği ve BD'nin en iyi nasıl değerlendirileceği gibi sorulara cevap bulmaya çalışmıştır (US National Research Council, 2011). Öte yandan dünya genelinde BD ve programlama etkinlikleri dâhil olmak üzere bilgisayar bilimlerinin müfredata entegrasyonuna ilişkin artan girişimler bulunmaktadır. Özellikle K12 düzeyindeki öğrencilerin farklı disiplinlerde olduğu kadar gerçek yaşamda da karşılaştıkları problemlerin çözümü için bilgisayımsal ilke ve araçları kullanmalarının teşvik edilmesine yönelik çağrılar artmaktadır (Wilkerson-Jerde, 2014).

BD’nin K12 öğretim programlarına entegrasyonuna CSTA, ISTE ve Gelecek Nesil Bilim Standartları (Next Generation Science Standards-NGSS) gibi bazı uluslararası sivil toplum kuruluşları da önemli katkılar sağlamaktadır. K12 düzeyindeki bilgisayım dersleri dünya genelinde giderek popüler hale gelmektedir. İngiltere'de bilgisayım dersleri 5-16 yaş arası için zorunlu hale gelmiştir. Ayrıca, bilgisayar programlama ile ilgili dersler yaklaşık 12 Avrupa ülkesinde (Türkiye, Bulgaristan, Kıbrıs, Çek Cumhuriyeti, Danimarka, Estonya, Yunanistan, İrlanda, İtalya, Litvanya, Polonya ve Portekiz) K12 müfredatının bir parçası olmuştur (Balanskat & Engelhardt, 2014). K12 seviyesindeki müfredat değişikliklerine ek olarak, programlamayı teşvik etmek için bazı uluslararası etkinliklere oldukça geniş katılımlar sağlanmıştır. Örneğin, Kodlama Saati (Hour of Code) etkinliği 180'den fazla ülkede on milyonlarca öğrenciye ulaşmıştır. Bununla birlikte BD becerisi için yürütülen çok çeşitli etkinlik ve uygulamaların etkisinin belirlenmesine ilişkin ölçme araçlarının yetersizliği söz konusu öğretimsel uygulamaların pedagojik ve pratik

çıktılarının değerlendirilmesinde zorluklar yaratmaktadır. Uygun bir değerlendirilmeye tabi tutulmaksızın BD becerisine yönelik herhangi bir öğretimsel pratiğin başarılı olduğuna dair beklentiler varsayımdan öteye geçememektedir. Bu yüzden şu ana kadar gelinen nokta halen bilgisayımın okul müfredatının zorunlu bir parçası olma hedefinden uzaktadır (Grover & Pea, 2013). BD topluluğunun öğrencilerin ne öğrenmesi gerektiğine karar vermesi için BD'nin tam olarak neyi kastettiğini tanımlaması, öğrenenlerin ne öğrendiğini değerlendirmesi için onların ne yaptığını ve BD'ye ilişkin tutum ve yeteneklerinin ne olduğunu bilmesine ihtiyaç bulunmaktadır. BD'nin değerlendirilmesine yönelik bazı çalışmalar gerçekleştirilmiştir (Brennan & Resnick, 2012; Fields, Searle, Kafai & Min, 2012). Ancak BD'yi kapsamlı ve etkin bir şekilde değerlendirmek için işlevsel çerçeve ve yaklaşımlar halen oldukça sınırlıdır.

Bilgisayımsal Düşünme için Önerilen Çerçeveler

Brennan ve Resnick (2012) BD'yi, aynı zamanda ilgili araştırma çalışmaları için de çerçeve görevi görebilecek, üç boyutta ele almıştır. Bunlar: bilgisayımsal kavramlar, pratikler ve perspektiflerdir (Tablo 2).

Tablo 2

Brennan ve Resnick (2012) Tarafından Önerilen Bilgisayımsal Düşünme Çerçevesi

BD Bağlamı	Tanım	Örnekler
Bilgisayımsal Kavramlar	Dizi, yineleme (döngü), koşul ifadeleri, metotlar (paralel işleme), olay işleyicisi, kullanıcı arayüz tasarımı, klavye girdileri gibi programcının kullandığı kavramlar	Değişkenler
Bilgisayımsal Pratikler	Programlama sürecinde gerçekleştirilen problem çözme uygulamaları	Döngüler, Artımlı ve tekrarlayıcı olma, test etme ve hata ayıklama, yeniden kullanma ve birleştirme, soyutlama ve ayrıştırma
Bilgisayımsal Perspektifler	Öğrencilerin kendileri, başkalarıyla olan ilişkileri ve çevrelerindeki teknolojik dünya hakkındaki anlayışları	Teknoloji dünyası hakkındaki açıklamalar ve sorgulamalar

BD kavramları döngüler, olaylar ve tetikleyiciler, koşul ifadeleri, değişkenler vb. yapı bloklarıyla (kodlarıyla) ilgili temel bilgilerin anlaşılmasını içerir. BD pratikleri ise bu kavramların nasıl ele alınacağına odaklanır. Bunlar, kodlama ya da

programlama yaparken öğrenciler tarafından uygulanan strateji ve tekniklerdir. Bilgisayımsal problem çözme bu süreçte yer alır. Pratikler test etme ve sorun giderme, kodları birleştirme ve tekrar kullanma, programı modüler hale getirme (ayrıştırma), yineleyici ve artırımlı olarak çalıştırmayı kapsamaktadır. Öte yandan BD perspektifleri ilgili bileşenlerle kodlama yapılırken sergilenir. Bunlar kendini yaratıcı biçimde ifade etme, başkalarıyla çalışma ve işbirliği yapmanın avantajlarını açıklama, teknolojiler hakkında soru sorma ve onları sorgulama kapasitesinin gelişimini içermektedir. Brennan ve Resnick'in (2012) çerçevesi öğrencilerin kod oluştururken öncelikli olarak kullandıkları bilgilerin türüne ve bu bilgileri nasıl kullandıklarına odaklanır.

BD'nin K12 bağlamlarına etkili bir şekilde entegre edilmesini sağlama çabalarına rehberlik etmesi için ISTE ve CSTA tarafından belirlenen BD kavramları ve yeterlikler şu şekilde sıralanmıştır (Barr & Stephenson, 2011; Mouza vd., 2017):

a) **Problemin ayrıştırılması.** Karmaşık problemlerin daha yönetilebilir parçalara bölünmesi

b) **Algoritmik düşünme.** Problemleri çözmek için belirgin bir dizi adım ve komutların kullanılması

c) **Soyutlama.** Temel bir fikir edinerek karmaşıklığı indirgemek. Soyutlamalar yoluyla doğal ve yapay olgulara ilişkin zihinsel modeller geliştirmek

d) **Verilerin toplanması, analiz edilmesi ve temsili.** Verilere ulaşma, onları değerlendirme ve kelimeler, görseller veya modeller kullanarak onları temsil etme

e) **Otomasyon:** Otomatikleştirilmiş çözümler için dijital araçlar kullanma

f) **Paralel işleme:** İşleri/süreçleri aynı zamanda yapmak ve ortak bir hedefin başarılmasına yardımcı olması için görevleri eş zamanlı olarak gerçekleştirecek kaynakları organize etmek

g) **Simülasyon:** Bir sürecin ya da sistemin temsilini yaratmak

Temel BD yetkinlikleri aynı zamanda BD'nin önemli boyutları olan birtakım eğilimler veya tutumlarla da arttırılabilir. Bu tutumlar, karmaşıklıkla başa çıkmaya yönelik özgüven, zor problemlerle çalışmanın sürekliliği, belirsizlik toleransı, açık

uçlu sorunlarla başa çıkabilme yeteneği, iletişim kurma ve başkalarıyla çalışma becerisidir. Weintrop vd. (2016) matematik ve fen için BD pratiklerinin taksonomisini geliştirmek üzere gerçekleştirdikleri çalışmada öğrenci ihtiyaçlarını ve matematik ve fen disiplinindeki uzmanların çalışma biçimini en iyi yansıtan BD karakteristiklerini tanımlamıştır. Araştırmacılar süreç boyunca diğer STEM araştırmacıları, öğretmenler ve program geliştiricilerle yakından çalışmıştır. Çalışmada ABD Ulusal Araştırma Konseyi tarafından yayımlanan iki rapor kullanılmış ve bu raporlardan hareketle bir alanyazın incelemesi gerçekleştirilmiştir. Bu inceleme sonunda BD taksonomisi için bir başlangıç noktası olarak kullanılan on temel özellik belirlenmiştir. Bunlar;

- Açık uçlu problemlerle baş edebilme becerisi
- Zorlu problemlerle çalışırken süreklilik (ısrar)
- Karmaşıklıkla başa çıkma konusunda güven
- Fikirleri bilgisayımsal olarak anlamlı yollarla sunma
- Büyük problemleri daha küçük problemlere bölme
- Eldeki problemin farklı yönleri için soyutlamalar oluşturma
- Problemi daha tanımlanabilir bir problem haline getirme
- Veri gösteriminin ya da temsil sisteminin güçlü ve zayıf yönlerini değerlendirme
- Algoritmik çözümler üretme
- Algoritmalarda belirsizliği teşhis etme ve adresleme

BD'yi değerlendirmek için gittikçe yaygınlaşan bir diğer etkinlik de Bebras Uluslararası Enformatik ve Bilgi İşlemsel Düşünme Etkinliğidir (www.bebras.org). Çeşitli yaş gruplarındaki öğrenciler arasında BD ve bilişimi teşvik etmeyi amaçlayan Bebras etkinlikleri, içerisinde çoklu zorluklar bulunan kısa görevler aracılığıyla gerçekleştirilen problem çözme etkinliklerini kapsamaktadır (Dagiene & Stupuriene, 2016). Bebras etkinlikleri Kasım ayının 2.-3. haftasında gerçekleştirilmektedir. Bu etkinlik esnasında öğrenciler öğretmenlerinin rehberliğinde bilgisayar ve mobil cihazlar aracılığıyla bilgisayımsal problemlerle uğraşmaktadır. Bebras, 60'ın üzerinde ülkede ulusal düzeyde yarışmalar ya da bireysel/grup katılımlarıyla sürdürülmekte olan uluslararası bir etkinliktir. Ülkemizde Bilge Kunduz (www.bilgekunduz.org) etkinliği altında benzer enformatik ve BD etkinlikleri gerçekleştirilmektedir. Bebras, bilgisayımla ilgili sınırlı bilgi birikimi olan öğrencilerin

BD becerilerinin değerlendirilmesine olanak tanımaktadır. Bununla birlikte Bebras etkinliklerinde BD kavramlarının yeterince adreslenmediği görülmektedir.

BD'nin K12 düzeyindeki öğretim programlarına entegrasyonunun ya başlı başına ayrı bir öğretim programı olarak bilişim/bilgisayar bilimleri dersleri (BİT, BT ve Yazılım) bünyesinde ya da farklı disiplinlere (örn. matematik, fen, müzik vs.) ait derslerin öğretim programlarına entegre edilmesi olmak üzere iki şekilde gerçekleştirildiği görülmektedir. Özellikle, ayrı bir disiplin olarak verilecek BD müfredatının otantik yaşam problemlerine dayalı otantik koşulları barındıracak şekilde tasarlanması konusunda artan bir fikir birliği bulunmaktadır.

Ülkemizde Bilgisayımın Eğitime Entegrasyonu

Birçok ülke (örn. Türkiye, İngiltere, ABD, Yunanistan), K12 müfredatlarındaki Bilgi ve İletişim Teknolojileri derslerini, öğrencilerin nasıl programlama (kodlama) yapacaklarını ve BD becerilerini nasıl kullanabileceklerini öğrendikleri bir biçime evirmektedir. Ülkemizde Talim ve Terbiye Kurulunun 19/02/2018 tarih ve 56 sayılı kararıyla kabul edilen ve 2018-2019 eğitim öğretim yılından itibaren uygulamaya konulan Ortaöğretim Kurumları Haftalık Ders Çizelgesine göre bilişim eğitimi lise düzeyindeki müfredat programlarında seçmeli ders statüsünde olup, Bilgisayar Bilimi ve Proje Hazırlama dersleri aracılığıyla haftalık 1 ya da 2 saat verilmektedir. Yine Talim ve Terbiye Kurulunun 20/02/2018 tarih ve 10 sayılı kararıyla kabul edilen İlköğretim Kurumları Haftalık Ders Çizelgesinde bilişim dersinin Bilişim Teknolojileri ve Yazılım dersi altında 5. ve 6. sınıflarda zorunlu, 6. ve 7. sınıflarda seçmeli olarak haftalık 2 saat okutulmasına karar verilmiştir. Tıpkı gelişmekte olan birçok ülke gibi (örn. Güney Kore) Türkiye de Bilişim Teknolojileri müfredatında uygulama yazılımlarının (örn. MS Office yazılımları) ağırlığını azaltmak, algoritma ve programlamayı kullanarak bilgisayar bilimleri ilkelerinin ve problem çözme stratejilerinin üzerinde daha çok durmak için yeniden müfredat tasarımı sürecine girmiştir (Jun, Han, Kim & Lee, 2014). Ancak müfredat düzenlemelerine getirilen eleştirilerin başında öğrencilerin düşünme becerilerinin yeterince desteklenmemesi ve bilgisayarların (daha geniş çerçevede BİT araçlarının) problem çözümünde araç olarak kullanılmasının yeterince teşvik edilmemesi gelmektedir. Müfredat düzenlemelerinde öğrenenlerin bilgisayımsal okuryazarlıklarının geliştirilmesi kadar, bilgisayımsal yetkinliklerini farklı alanlardaki problemlerin çözümünde kullanmalarına odaklanılması gerekmektedir. Çok sayıda ülkede kodlamanın okul

müfredatına girmesi ya da bilgisayımın başta matematik ya da fen olmak üzere diğer öğretim programlarının yükselen bir parçası olması için somut adımlar atılmaktadır (Falloon, 2016). Buna rağmen BD girişimlerinin önemli bir bölümünün, sadece kodlama gibi faaliyetler aracılığıyla BD'nin geliştirilmesine odaklandığı görülmektedir.

Bazı araştırmacılar okul müfredatına kodlama gibi BD etkinliklerini dâhil etmenin bilgisayımsal sorunların anlaşılması ve çözülmesinin ötesinde daha geniş faydalarının olduğunun altını çizmektedir. Lye ve Koh (2014), BD'nin sistematik düşünebilme yeteneğini besleyerek, matematik ve fen uzmanlığının gelişmesine yardımcı olduğunu belirtmektedir. Öte yandan bilgisayım müfredatı tasarlanırken bilgisayar bilimleri ve teknoloji eğitimiyle nasıl ilişkilendirileceği açıklığa kavuşturulmalıdır. Ayrıca somut müfredat standartları geliştirilmeli ve bu alanlarda öğretmenlerin niteliklerinin geliştirilmesi için mesleki gelişim konuları göz ardı edilmemelidir.

Bilgisayımsal Düşünme Araçları

Sınıf içerisinde kullanılacak bir teknolojinin öğrenme verimini arttırması, iletişimi ve işbirliğini desteklemesi ve içerik sunumuna olanak sağlaması beklenmektedir. BD'nin başta K12 olmak üzere öğretimsel düzenlemeler içerisinde yer alabilmesine yönelik birçok araç bulunmaktadır. Bu araçların büyük bölümü "düşük zemin, yüksek tavan" ilkesiyle çocukların yazılım araçlarına sınırlı bir anlayışla bile erişebilecekleri, anlayışları ve yeterlikleri arttıkça daha ileri programlar geliştirmelerine olanak tanıyacak şekilde tasarlanmıştır. Bu araçlar, genellikle sürükle ve bırak arabirimleri aracılığıyla kural tabanlı programlamayı sağlayarak sözdizimine olan bağlılığı en aza indirgemektedir (Kelleher, 2006). Yapılan çalışmalar bu araçların bilgisayar bilimlerine karşı öğrencileri motive etmede başarılı olduğunu göstermektedir (Kelleher, 2006; Squire, 2003). Bu araçlar oldukça karmaşık olan bilgisayar bilimi kavramlarının K12 seviyesinde yaygınlaşmasında kolaylaştırıcı bir rol üstlenmektedir. Alanyazın incelendiğinde BD becerisinin desteklenmesine yönelik olarak blok temelli grafiksel programlama ortamları, web-tabanlı 2B ve 3B simülasyon ve yazarlık araçları, 3B video oyun ortamları, simülasyon ve modelleme araçları, eğitsel robotik ve elektronik devre setleri gibi uyarlanabilir ortamlar, bilimsel sorgulama ve sınıflama araçları gibi çok çeşitli ortam ve araçların önerildiği görülmektedir (Tablo 3).

Tablo 3

Bilgisayımsal Düşünme İçin Kullanılan Ortam ve Araçlar

BD Araç türü	BD Aracı
Blok Temelli Görsel Programlama	Scratch (Resnick vd., 2009)
	Alice (Cooper, Dann & Pausch, 2000)
	Blocky (Fraser, 2015)
2B ve 3B Animasyon ve Oyun tasarım ve yazarlık araçları	Kodu Game Lab (Fowler, Fristce & MacLauren, 2012; Stolee & Fristoe, 2011)
	Stagecast Creator (Smith, Cypher & Tesler, 2000)
	ToonTalk (Kahn, 1996)
	Agentsheets ve Agentcubes (Repenning vd., 2014)
	RoboBuilder (Weintrop & Wilensky, 2012)
	Dragon Architect (Bauer, Butler & Popović, 2015)
	GameMaker (Hoganson, 2010)
	Greenfoot (Kölling, 2010)
3B Video Oyunlar ve Oyun benzeri ortamlar	Second Life (Pellas & Peroutseas, 2016)
	PlayLOGO 3D (Paliokas, Arapidis & Mpimpitsos, 2011)
	OpenSim (Pellas, 2014)
Simülasyon ve Modelleme Araçları	NetLogo (Tisue & Wilensky, 2004)
	iLab (Jona & Vondracek, 2013)
	Minecraft (Roscoe, Fearn & Posey, 2014)
Eğitsel Robotlar	Lego Mindstorms (Sullivan, 2008)
	Lego We Do Robotik seti (Jaipal-Jamani & Angeli, 2017)
Elektronik devre yapım setleri	Aurdino (Brock, Bruce & Reiser, 2009)
	Gogo Boards (Sipitakiat, Blikstein & Cavallo, 2004)
	PicoCricket (Rusk, Resnick, Berg & Pezalla-Granlund, 2008)
	littleBits (Bdeir, 2009)
	Elektronik tekstil üretim setleri (Arduino LilyPads) (Buechley & Eisenberg, 2008; Litts, Kafai, Lui, Walker & Widman, 2017)
	KIBO robotik (Sullivan, Bers & Mihm, 2017)
Bilimsel Sorgulama Araçları	WISE (Linn, Clark & Slotta, 2003; Slotta, 2004)
	Genscope (Horwitz, 1995)
	WorldWatcher (Edelson, Gordin & Pea, 1999)
Sınıflama (Taksonomi) Araçları	Categorizer (Wilkerson-Jerde, 2014)
	BioKIDS CyberTracker (Parr, Jones & Songer, 2002)

Kullanımı kolay görsel programlama araçlarının (Scratch, Toontalk, Stagecast Creator ve Alice gibi) uygunluğu K12 seviyesindeki okullarda BD girişimlerinin yaygınlaşmasına önemli katkılar sağlamıştır. Küçük yaşlardaki öğrenenler için geliştirilen programlama araçları (E-toys, Scratch J, Scratch)

programlama bilgisine bağlığı azaltarak BD becerilerinin sergilenmesini kolaylaştırmaktadır. Giderek artan sayıda ortam öğrenenlerin etkinliklerini gerçek ürünler (modeller, hikâyeler, oyunlar, animasyonlar, robotik uygulamalar vb.) haline getirerek programlama çıktılarının sahipliğine olanak sağlamaktadır. Burke ve Kafai (2010) ortaokul öğrencilerinin kodlama ve yazma becerilerinin gelişmesi için Scratch kullanımını araştırdığı çalışmasında öğrencilerden Scratch kullanarak dijital bir hikâye oluşturmalarını istemiştir. Çalışmada öğrencilerin dijital hikâyeler oluşturmak için temel programlama kavramlarını (döngü, olay oluşturma ve işleme, paralel yürütme vb.) kullanabildikleri bulunmuştur. Kalelioğlu, Gülbahar, Akçay ve Doğan (2014) Scratch programlamanın ilköğretim öğrencilerinin problem çözme becerileri ve tutumları üzerindeki etkisini incelemişlerdir. Araştırmacılar Scratch programlamanın problem çözme becerisine ilişkin önemli kazanımlar sağlamadığını bulmuştur. Ancak, Scratch kullanımı sonrası öğrencilerin programlamaya karşı olumlu tutum geliştirdiği raporlanmıştır.

Blok temelli programlama araçları programlamaya karşı öğrenci ilgisinin uyandırılmasına, öğrencilerin programlama hakkında bilgilendirilmesine, Java ve C gibi metin tabanlı dillere kolay bir geçiş yapılmasına ve risk altındaki bilgisayar bilimleri bölümlerinin sürekliliğinin korunmasına katkıda bulunabilir. Sürükle-bırak yöntemiyle kullanılan kodlama blokları sözdizimsel kuralları asgari seviyeye indirerek programlama için zorluk eşiğini düşürürken, aynı zamanda öğrencilerin programlama mantığıyla da meşgul olmasına olanak tanımaktadır. Öte yandan bu araçlarla ilgili alanyazında bir takım uyarı ve eleştiriler bulunmaktadır. Kalelioğlu (2015) operasyonel dizilim eksikliği, geçersiz sözdizimi kullanımı ve problem çözme stratejilerinin yetersiz temsilinin daha üst düzey programlama etkinliklerinde kaotik bir çabaya dönüşebileceğini belirtmiştir. Ayrıca farklı araç türleri vasıtasıyla gerçekleştirilen etkinlikleri öğrencilerin programlamadan ziyade işe koşulan diğer etkinliğe (örneğin tasarım ya da multimedya özellikleri) odaklanmasına yol açabilmektedir. Pellas ve Peroutseas (2016), Scratch, Alice, Kodu veya AgentSheets gibi görsel programlama ortamlarına rağmen, kullanıcıların anlamlı bir şey yaratmak için bilgisayar programlama kavramları ile yeterince meşgul olmadıklarını belirtmiştir. Araştırmacılar bu durumun özellikle problem temelli görevlerde öğrenci katılımını kolaylaştıramayan bir “dik öğrenme eğrisi” yarattığına dikkati çekmiştir. Programlamaya yeni başlayan öğrenciler için BD becerilerini nasıl

kullanacaklarını anlamada zorluk yaratan bu eğri verimsiz sonuçlara yol açabilmektedir. 2B veya 3B görsel ortamlar aracılığıyla uygulanan çeşitli programlama etkinliklerinde, öğrenciler belirli bir hikâye (senaryo) veya öğretim içeriği olmaksızın karmaşık problem durumlarıyla karşı karşıya bırakılmaktadır. Sonuç olarak, öğrenciler nasıl programlanacağını öğrenmek yerine, bir şeyi kişiselleştirmek ve değiştirmek için daha fazla zaman harcamak zorunda kalmaktadır (Pellas & Peroutseas, 2016). Bu durum öğrencilerin temel programlama derslerine katılımını olumsuz yönde etkilemektedir.

BD'yi anlama ve değerlendirme için çerçeve oluşturma çabalarına paralel olarak, bilgisayımsal yetkinlikleri K12 öğrenme bağlamlarına entegre etmek ve bu yetkinliklerin gelişimini desteklemek için sürekli olarak yeni öğrenme ortamları, araçları ve etkinlikleri tasarlanmaktadır. Sayısal teknolojilerin temel günlük yaşam pratiklerini yerine getirmede önemli bir rol üstlendiği günümüz dünyasında bireylerin teknolojik sistemleri kullanmalarının yanı sıra işler ters gittiğinde söz konusu teknolojiyi eleştirel bir biçimde ele almaları, sorunları tespit etmeleri ve gidermeleri için gerekli eğitim, bilgi ve becerilere sahip olmaları zorunludur (Angeli vd., 2016). Wing (2006), bu vizyona erişmek için bilgisayar ve diğer bilişim araçlarının BD gelişimini ve yayılımını kolaylaştırabileceğini ileri sürmüştür. Bununla birlikte bilgisayarsız bilgisayar bilimleri (CS Unplugged) gibi bilgisayara gerek duyulmaksızın bilgisayım kavramlarını tanıtan ve öğrenenlere bilgisayar biliminin doğası hakkında zengin deneyimler sunan çabalar da bulunmaktadır. Bilgisayarsız bilgisayar bilimleri etkinliklerinde kalem ve kâğıt kullanarak, yani priz kullanmaksızın yapılan etkinliklerle algoritmik düşünme, soyutlama, ayrıştırma, genelleme ve örüntü bulma/yaratma, mantıksal neden-sonuç ilişkisi kurma gibi BD'nin temel bileşenlerinin geliştirilmesi hedeflenmektedir. Bilgisayar bilimleri, belirli bir yazılıma veya donanıma bağlı tek bir beceri kümesine sahip olanların hızla geride kalabileceği oldukça devingen bir alandır (Chen vd., 2017). Bu nedenle BD'nin teknolojik araçlara bağlı olmaksızın geliştirilmesi, onun günlük hayatın bir parçası olması ve hayatın her alanında karşılaşılan geniş ölçekteki problemlere uygulanması için son derece önemlidir.

Bilgisayımsal Düşünme Teknikleri

Toplumda ve gündelik hayatta yeniliklere ön ayak olmasına karşın bilgisayar bilimlerine karşı ilgi giderek azalmaktadır (Carter, 2006; Denning & McGettrick,

2005). Birçok araştırmacı bunun temel nedenini öğrencilerin geleneksel pedagojiye ve bilgisayar bilimleri yöntemlerine yetersiz ilgisi olarak açıklamaktadır (Patterson, 2005; Zendler & Klaudt, 2012). Toplumda çok sayıda kişi programlamanın bilgisayar bilimleri uzmanlarının işi olduğunu düşünmektedir. Öğrenenlerin bilgisayar bilimlerine karşı ilgisinin yetersiz olmasının önemli bir nedeni de bilgisayar bilimlerinin temel esasları olarak sadece programlama sürecine odaklanılmasıdır.

BD ile ortaya konulan “herkes için programlama” ilkesi ve her öğrenim kademesinde yeni başlayanlar için programlamaya elverişli ve düşük kullanım zorluğuna sahip programlama araçları sayesinde bilgisayar bilimlerine karşı azalan ilgi tekrar canlanmıştır. Eğitimciler öğrencilerin dikkatini çekmek ve algılarını değiştirmek için bu alanda salt programlama dışında çeşitli etkinlik türleri önermiştir. Bunlardan bazıları;

- Bilgisayarsız bilgisayar bilimi (Bell, Alexander, Freeman & Grimley, 2009; Taub, Armoni & Ben-Ari, 2012),
- Fiziksel ya da somut öğeli programlama (robotik uygulamalar ve elektronik devre tasarımları dâhil) (Bers, Flannery, Kazakoff & Sullivan, 2014; Chen vd., 2017; Flowers & Gossett, 2002; Jaipal-Jamani & Angeli, 2017; Leonard vd., 2016; Litts vd., 2017; McNerney, 2004),
- Dijital hikâye anlatımı (Daily & Eugene, 2013; Zhang, Liu, de Pablos & She, 2014; Zhong vd., 2016),
- Sistem modelleme/geliştirme ve simülasyon (Basawapatna, 2016; Berland & Wilensky, 2015; Fang vd., 2017),
- Örüntü bulma, kural oluşturma ve sınıflama (Wilkerson-Jerde, 2014)
- Problem tabanlı öğrenme (Koschmann, Feltovich, Myers & Barrows, 1992),
- Bulmaca/araştırma tabanlı öğrenme (Merrick, 2010)
- Oyun temelli ya da oyunlaştırarak öğrenme (Berland vd., 2013; Berland & Wilensky, 2015; Chang, 2014; Greitzer, Kuchar & Huston, 2007; Leonard vd., 2016)

Bazı uygulamalarda bu etkinliklerden bir ya da birkaçının birlikte işe koşulduğu görülmektedir. Bununla birlikte alanyazında BD öğretimine yönelik baskın stratejinin halen programlama olduğu görülmektedir. Ayrıca söz konusu etkinlikler genellikle programlamayla birleştirilmektedir.

Programlama. Papert (1980) programlama etkinliklerinin çocukların bilişsel becerileri üzerinde olumlu bir etkiye sahip olabileceğini ve onların öğrenme biçimlerini etkileyebileceğini savunmaktadır. Programlama eğitiminin başta BD olmak üzere eleştirel düşünme ve yaratıcı düşünme gibi üst düzey düşünme becerilerini ve problem çözme yetkinliğini arttırdığına dair çalışmalar bulunmaktadır (Carbonaro, Szafron, Cutumisu & Schaeffer, 2010; Gross & Kelleher, 2010).

Programlama, her ne kadar kökenlerini Bilgisayar Bilimlerinden alsa da BD'nin birçok formundan sadece birisidir (Voogt, Fisser, Good, Mishra & Yadav, 2015). Programlamayı öğrenmek oldukça zorlu bir görevdir. Özellikle yeni başlayanlar için programlama etkinlikleri çok çeşitli zorluklar ve gereksinimler barındırmaktadır. Programlama dersleri bu nedenle zor olarak kabul edilir ve genellikle yüksek ders bırakma oranlarına sahiptir (Robins, Rountree & Rountree, 2003). Bununla birlikte BD bilgisayar kullanımının ötesinde, çoğu zaman programlama yapmayı gerektirmeksizin, çoklu bağlamlarda ve farklı konu alanlarında geliştirilebilir.

Bilgisayarsız bilgisayar bilimleri. Öğretmen adaylarının birçoğu BD'nin sınıf içerisine entegrasyonunun yalnızca teknolojiyle sağlanabileceği gibi hatalı bir anlayışa sahiptir (Mouza vd., 2017). Bilgisayarsız bilgisayar bilimleri etkinlikleri fiziksel materyalleri ve hareketleri temel alan bir dizi etkinlikler (örneğin oyunlar, kalem-kâğıt etkinlikleri) yoluyla BD eğitimine odaklanmaktadır.

Dijital hikâye anlatımı. Yaratıcı ve yenilikçi düşünme, eleştirel düşünme ve problem çözme gibi kritik becerilerin; ortak çalışma ve işbirliği gibi çalışma biçimlerinin bilgi, medya ve BİT okuryazarlığı gibi yetkinliklerin önemi giderek artarken geliştirilmesi zor bir beceri seti de öğrenci ve eğitimcileri beklemektedir. Yirmi birinci yy. öğrenme becerilerinin eğitim kurumlarına yaptığı en önemli çağrı okulların daha etkin öğretim ve öğrenme biçimleri araması gerektiğidir.

Dijital hikâye anlatımı, hikâye anlatılarının etkileşimli bir ortamda multimedya (metin, ses, video, grafik) bileşenlerinden yararlanılarak oluşturulduğu pratiğe dönük

eğitsel bir uygulamadır. Bilindiği gibi insanların çoğu hikâyeleri sever; ancak bilgisayar bilimlerinden herkes hoşlanmaz (Hartley & McWilliam, 2009). Özellikle blok temelli programlama çevreleri (Scratch ve Alice gibi) sundukları ses, görüntü ve hareket bileşenleriyle öğrencilerin dijital hikâye oluşturmasına son derece elverişli ortamlar haline gelmektedir. Dijital hikâye anlatımıyla öğrenciler tarafından genellikle sıkıcı bulunan programlama süreci daha zevkli hale getirilip, akış deneyiminin daha sık yaşanması sağlanabilir. Böylelikle hem soyutlama, algoritmik düşünme ve mantıksal akıl yürütme gibi BD'nin hem de özfarkındalık, empati, yaratıcı düşünme, kendini ifade etme becerisi gibi dijital hikaye anlatımlarının desteklediği beceriler ortak bir noktada buluşarak bir sinerji yaratılabilir. Zhang vd. (2014), öğrencilerin Alice'te animasyon ve video oyunları oluştururken aynı zamanda temel programlama kavramlarını öğrenebildiklerini raporlamıştır.

Hikaye anlatımları öğrenenlerin duygusal özfarkındalık ve empati gelişimini destekleme potansiyeline sahiptir (Daily & Eugene, 2013). Dijital anlatımlar öğrencilerin kolaylıkla açığa vuramadığı kültürel özellikleri sergilemeleri ya da yansıtmaları için de fırsatlar yaratmaktadır. Daily ve Eugene (2013), British Petroleum Deepwater Horizon petrol sızıntısı felaketinin yakından etkilediği körfez bölgelerindeki öğrencilerle yürüttüğü çalışmada öğrencilerin hikâyelerini geliştirdikçe ve çevreleriyle etkileşimde bulundukça, kendi duygularını ve başkalarının duygularını hikâyelerine yansıttıklarını raporlamıştır. Araştırmacılar aynı zamanda öğrencilerin Scratch ortamında düşüncelerini ve fikirlerini ifade edebilmek için BD becerilerini de sergilediklerini gözlemlemiştir. Çalışmada BD ortamlarının dijital hikaye anlatımı bağlamında kullanılmasının öğrencilerin duygusal farkındalık ve empati becerilerinin geliştirilmesi ve onların farklı ortamlarda etkili işbirlikleri kurma çabalarının desteklenmesi açısından oldukça verimli sonuçlar verdiği belirtilmiştir. Ek olarak, araştırmacılar BD etkinliklerinin dijital hikâyeleme tekniğiyle birleştirilmesinin öğrencilerin bilgisayar bilimleri ve STEM alanlarında kariyer seçimlerine yönelik olumlu tutum geliştirmesine katkı sağladığını belirtmiştir.

Sistem modelleme ve simülasyon. Sistem, sınırları belirlenmiş birbiriyle ilişkili elemanların kümesidir. En açık bir ifade ile sistem belirli bir amacı gerçekleştirmek için birlikte çalışan ve birbirlerini etkileyen parçalardan oluşan bir bütündür (Şenaras & Sezen, 2017). Sistem yaklaşımı ise bir probleme ilişkin tüm özellikleri bütünsel bir bakış açısıyla düşünerek, bileşenler arasındaki ilişkileri ortaya

çıkaran bir problem çözme yaklaşımıdır. Karmaşık sistemsel düşünme, genellikle bilgisayımsal sistemleri modelleme ve bireysel unsurların etkileşiminden kaynaklanan davranışları ortaya çıkarmayla ilişkili olarak tanımlanır (Jacobson & Wilensky, 2006). Görüleceği gibi bu yaklaşımda temel husus, sistemin elemanları arasındaki etkileşimlerin oluşturduğu ağı görebilmek ve anlayabilmektir. Aristo'nun da belirttiği üzere "bütün; bütünü oluşturan parçaların ayrı ayrı etkisinden çok daha büyük bir etkiye sahiptir". Dolayısıyla bugün karşılaştığımız problemlerin daha iyi anlaşılabilmesi için yaklaşımlarımızda doğrusal nedenselliğin yerine, karşılıklı bağımlılığın yer alması daha makul gözükmektedir.

Karmaşık sistemsel düşünme becerisinin gelişimi için modelleme etkinlikleri (kavram haritası gibi) anahtar roldedir. Modelleme ve simülasyon araçları öğrencilere programlamaksızın öğe ya da nesneleri benzersiz konfigürasyonlara yerleştirme ve diğer öğelerle etkileşimlerini belirleme yoluyla bir problem üzerinde deney yapma ya da bir olguyu açıklama fırsatı vermektedir (Basawapatna, 2016). Burada ajanlar ve bunların etkileşimi için oluşturulan koşulsal durumların tanımlanması oldukça önemlidir. Hiç programlama deneyimi olmayan öğrenciler modelleme ve simülasyon araçlarını kullanarak analojiler yoluyla gerçek dünya olgularını simüle edip soyutlama süreçlerini deneyimleyebilmektedir.

Oyun tasarımı/programlama ve oyunlaştırma. Dijital oyunlar belirli bir senaryo ya da hikâye çerçevesinde oyuncuların hedef belirlemesine ve kurallar eşliğinde oyunda ilerlemesine olanak tanıyan, onlarda merak duygusu uyandıran ve onlara akış deneyimi yaşatan, sunulan görece zorluklara göre içinde bir puanlama ya da ödül-ceza sistemi barındıran, rekabetçi ve çekici etkileşimli teknolojiler olarak tanımlanabilir. Dijital oyun pedagojisi, geleneksel araç-temelli yaklaşımdan farklı dinamikleri (amaç, mücadele, uğraşı, eğlence vb.) sayesinde belirli bir öğrenme konusuna yönelik tutum, bağlılık ve motivasyonu arttırma potansiyeline sahiptir. Oyuncular dijital bir oyunu zorluklarla uğraşmak, sosyalleşmek, dinamik özerk deneyimler elde etmek, popüler olmak ve övünmek, duygusal deneyimler yaşamak, keşfetmek, hayal kurmak ve etkileşimlere girmek için oynar (Rouse, 2005).

Oyun tasarımı ve programlama karmaşık bir eser yaratmak üzere geniş yelpazede bir etkinlik setini (oyun kuralları oluşturmak, karakter ve diyalog, görsel tasarım ve programlama) içeren bir süreçtir. Bu süreç BD becerilerinin geliştirilmesi ve sergilenmesi için oldukça elverişli fırsatlar sunmaktadır (Robertson & Howells,

2008). Süreç sonunda ortaya çıkan ürünler popüler kültürde oldukça ses getirmekte ve çok geniş kullanıcı kitlelerine ulaşabilmektedir. Bu yönüyle oyun tasarımı otantik, özerk ve aktif öğrenme deneyimi sunmaktadır (Barab, Thomas, Dodge, Carteaux & Tuzun, 2005). Oyun tasarımı aynı zamanda iletişim ve işbirliğine oldukça açık bir süreçtir. Alanyazın incelendiğinde öğrencilerin BD becerilerini geliştirmek için oyun tasarımı ve programlama ya da oyunlaştırma gibi çeşitli stratejilerin yaygın olarak işe koşulduğu görülmektedir (Garneli & Chorianopoulos, 2018). Berland ve Wilensky (2015) fiziksel veya sanal robotik araçları çok kullanıcılı bir futbol oyununun programlanması bağlamında kullanarak bu entegrasyonu BD ve karmaşık sistem modelleme bağlamında incelemiştir. Garneli ve Chorianopoulos (2018), elektron hareketi ve elektrik akımı başlıklı bir fen bilgisi konusuna ilişkin olarak ortaokul öğrencilerinden oluşan iki katılımcı grubundan birine Scratch aracıyla bir simülasyon, diğerine ise dijital bir oyun tasarlatmak koşuluyla iki farklı etkinlik türünün BD becerisi üzerindeki etkisini incelemiştir. Sonuçlar her iki etkinlik türünün de öğrencilerin BD becerilerini geliştirdiğini göstermekle birlikte video oyunu grubunun simülasyon grubuyla karşılaştırıldığında daha yüksek başarı elde ettiği görülmüştür. Dahası, video oyunlarının öğrenci motivasyonunu daha olumlu etkilediği raporlanmıştır.

Örüntü bulma, kural oluşturma ve sınıflama. Belirli bir olguyla ilgili örnekleri gözlemlemeyi, bu örnekler arasındaki benzerlik ve farklılıkların farkına varmayı, bu farklılıkların nedenlerini açıklamayı ve en nihayetinde kural oluşturarak onları sınıflamayı içerir. Bir konunun (örneğin matematiksel ya da doğa bilimlerindeki konular) muhtelif örnekleri ya da ürünleri öğrencilere çoklu fikirleri birleştirici bir süreçle bir araya getirme, anlayışlarını yeni yollarla organize etme, umulmadık sonuçlar karşısında anlayışları üzerinde düşünüp, onları gözden geçirme fırsatı vermektedir (Wilkerson-Jerde, 2014). Bu aynı zamanda öğrencilerin öğrenmenin altındaki yapıları keşfetmelerine, neden-sonuç ilişkileri kurmalarına yardımcı olmaktadır.

Wilkerson-Jerde (2014), matematik öğretimine yönelik olarak gerçekleştirdiği çalışmada ortaokul öğrencilerinin (6-8. sınıf) bilgisayar destekli etkileşimli bir fotoğraf galerisi aracını (Categorizer) kullanarak çeşitli fraktalları yapısal ve matematiksel özelliklerine göre analiz etmesini, kullanıcı tanımlı kategoriler içerisine sınıflamasını, belirli alan küçültme oranlarına sahip fraktallar için algoritmalar

geliştirerek ortak bir koleksiyon oluşturmasını ve oluşturdukları bu bilgisayımsal ürünleri paylaşarak üzerinde tartışmasını sağlamıştır. Çalışmada kurmacı öğrenme anlayışını temele alan işbirlikli bir ortam aracılığıyla oluşturulan bir öğrenme topluluğu içerisinde öğrenenlerin oluşturdukları ürün koleksiyonlarını (algoritmalar ve fraktal örnekleri) kategorize etmesi ve kurdukları bağlantıları keşfetmeleri sağlanmıştır. Anket sonuçlarından ve ortam kullanımından elde edilen bulgular öğrencilerin fraktallar inşa etmek için kullandıkları bilgisayımsal kurallar ile bu fraktalları matematiksel olarak organize etme yolları arasında bağlantılar kurabildiklerini ve bu sayede konu alanına dair daha derinlemesine bir anlayış geliştirdiklerini göstermiştir. Analizler öğrenci etkileşimlerinin ürün-konu alanı arasında bağlantılar kurulmasını ya da yakalanmasını teşvik ettiğini göstermiştir.

Özetle, ürünler öğrenilenlerin uygulanmasına olanak tanıyacak biçimde algoritmik çıkarımlara, neden sonuç ilişkilerine ve bu yolla genellemelere ulaşılmasını sağlayacak önemli örüntüleri temsil etmektedir. Bu aynı zamanda öğrencilerin nesnelerin yüzeysel özelliklerinden ziyade derin yapısal özelliklerine odaklanmalarına, ortak ya da işbirlikli çalışmalar yoluyla örüntüleri keşfetmelerine ve bunlara ilişkin kural geliştirmelerine yardımcı olmaktadır.

Robotik Etkinlikler ve Robot Programlama

Papert'ın öne sürdüğü kurmacılığın ana esası öğrencilerin etkileşimli ve gerçekçi nesneler oluşturma sürecine aktif olarak katıldığında bir öğrenme sürecinin daha etkili olacağı düşüncesidir. Kurmacılığın kuramsal bir temeli de deneyimsel öğrenmeyle yakından ilişkilidir. Teknoloji aracılı ürünler tasarlamak ve kullanmak kendi başına bir disiplinde öğrenme, katılım ve bilgi üretmenin merkezi bir yönüdür.

Birçok insanın robotlarla ilk tanışması muhtemelen oyuncaklar ya da bilimkurgu filmleri (Yıldız Savaşları film serisindeki R2-D2 ve BB-8 droidleri gibi) vasıtasıyla gerçekleşmiştir. Günümüzde ise robotlar sadece çocuklar için eğlenceli bir araç değil, hayatımızın her alanının içinde olan ve her noktasını etkileyen (akıllı ev eşyaları, arabalar ve kıyafetler) ürün ve uygulamalardır. Aslında robot denilince akla ilk olarak insan gibi hareket edebilen, insansı tepkiler veren makineler gelmektedir. Ancak bu kalıplaşmış imajlar yanıltıcı olabilmektedir. Yürüme ve konuşma sadece bir tür robotik özelliktir. Robotik, programlanabilen her tür makineyi kapsar. Robot, özünde dış dünyadaki verileri sensörler (dokunma, sıcaklık, uzaklık,

sıvı, nem, alev-duman, ışık ve ısı sensörü gibi) vasıtasıyla analog ya da dijital bir kanaldan toplayan, bu verileri işleyen ve buna göre tepkiler üreten elektronik bir sistemdir. Otomobillerin yakıt deposundaki sensörün yakıt seviyesini ölçüp yakıtın belli bir değerin altına inmesi durumunda uyarı lambasını yakması buna örnek olarak gösterilebilir.

BD becerilerinin geliştirilmesi için önerilen programlama çevrelerinin büyük bölümünün ekran bağımlı olması çocukların ortak projeler üzerinde işbirliği yapma potansiyelini sınırlamaktadır. Bilgisayar üzerinden çalışma genellikle tek bir kullanıcı veya bir çift kullanıcıyla sınırlıdır. Birden çok kullanıcı bir bilgisayarı kullanmayı denediğinde, genellikle bazı kullanıcılar etrafta gezinmekte veya klavyeyi ve fareyi kimin kontrol edeceği konusunda pazarlığa girişmektedir (Burleson vd., 2018). Özellikle küçük yaşlardakiler olmak üzere öğrencilerin genellikle oyun oynayarak, sosyalleşerek ve çevrelerini manipüle ederek öğrendikleri bilinmektedir (Wyeth, 2008; Zuckerman & Resnick, 2003). BD için çok çeşitli eğitim teknolojilerine odaklanılmasına rağmen, bunların büyük çoğunluğu hala bir çocuk başına bir cihaz paradigmasına dayanmaktadır. Ayrıca yeni pedagojik teknolojilerin öğretim entegrasyonu ağırlıklı olarak ekran temelli bağlamlarda gerçekleşmektedir. Robotik teknolojiler öğrencilerin fiziksel nesnelerle etkileşim kurmasını gerektirmektedir. Bu aynı zamanda programlama öğreniminde sağlanan güç ve esneklikle birlikte, öğrenenlere BD becerilerini geliştirmelerinin yeni ve etkili yollarını sunmaktadır. Öğrenciler robotik teknolojiler sayesinde otantik dünya düzenlemeleri içerisinde nesnelerle etkileşim kurarken, aynı zamanda akranlarıyla daha geniş bir iletişim ve işbirliği fırsatı yakalamaktadır.

Okul öncesi ve K12 düzeyindeki öğrenenlerin teknolojiyle etkileşimleri yönünden farklılıklar bulunmaktadır. Okul öncesi öğrencileri programlama simgelerini daha sezgisel bir yolla bir araya getirmeye çalışırken; daha üst sınıflardaki öğrenciler programlama bloklarını okuyup, bunlar üzerine değer girişi yapabilmektedir. Özellikle bazı dilsel ve matematiksel ön gereksinimler erken yaştaki çocukların programlama öğrenmesinin önünde bariyer oluşturmaktadır. Her ne kadar Scratch Jr gibi çocuklara yönelik bazı programlama ortamları sadece şekil ve renk öğeleri ile sembolik olarak temsil edilen kodlar sağlasa da özellikle erken yaştaki kullanıcılar için uygun programlama ortamlarının sayısı oldukça sınırlıdır. Robotik etkinlikler küçük çocukların hem fiziksel beceri kazanması hem de somut

öğe ve çıktılarla (görsel ve sesli uyarılar) uzamsal programlama yaparak BD becerileri geliştirmeleri için eşsiz fırsatlar sunmaktadır (Burleson vd., 2018).

Tolman'a (1999) göre insanlar kendilerini yarattıkları şeylerle tanımlamaktadır. Robotik yapı blokları ve elektronik devre setleri öğrenciler için son derece manipülatif öğrenme araçlarıdır. Robotik etkinlikler öğrenci özerkliğinin desteklenmesi, yaratıcı, stratejik ve üstbilişsel düşünme becerilerinin sergilenmesi için oldukça elverişlidir. Robotik teknolojiler öğrencileri soyut fikirler hakkında düşünmeye teşvik etmektedir. Çünkü öğrenciler robotların hareketlerini programlamadan önce zihinlerinde tasavvur etmektedir. Bu yönüyle gerçek materyallerin aktif manipülasyonuna dayanan robotik teknolojiler soyut kavramların gözlemlenebilir ve hayal edilebilir bir şeye bağlanmasına olanak tanır (Burleson vd., 2018).

Eğitsel robotlar endüstri ve araştırmalardaki muadillerine nazaran daha uygun maliyetli ve daha kolay kullanımlı hale geldikçe K12 sınıflarında daha görünür olmaya başlamıştır (Ludi, 2012). Günümüzde uygun maliyetli robotik setler ve kullanımı kolay programlama araçları oldukça yaygınlaşmıştır. Buna paralel olarak öğrencilerden robotları mucizevi şeyler gibi görmeyi bırakıp, onların çalışma esaslarını kavramaları, belirledikleri amaçlar doğrultusunda robotları tasarlamaları ve kodlamaları; böylece üretim etkinlikleriyle uğraşmaları beklenmektedir. Tıpkı dünyada olduğu gibi (First, RoboCup Junior, BotBall) ülkemizde de K12 düzeyinde birçok robot etkinliği (MEB uluslararası robot yarışması, Hackidhon vb.) düzenlenmektedir. Giderek artan sayıda eğitim kurumu, öğrencilerin programlanabilir yapı setlerini kullanarak farklı ihtiyaçlara cevap verecek robotik ürünler oluşturmalarını teşvik etmektedir.

Robotik teknolojiler geliştikçe, robotların öğretme ve öğrenmeyi destekleme amaçlı kullanımı daha popüler hale gelmiştir. Robot tasarım ve programlama sürecinde, öğrenciler mühendislik, matematik ve bilgisayar bilimi kavramlarıyla meşgul olmaktadır (Chih-Wei, Jih-Hsien, Po-Yao, Chin-Yeh & Gwo-Dong, 2010). Araştırmalar, robotiğin ilkokul öğrencilerinin bile beklenenden daha erken dönemde STEM kavramlarını öğrenmesi için fırsatlar sunabileceğini ve STEM alanlarındaki bilgi ve becerileri geliştirebileceğini göstermektedir (Nugent vd., 2015; Park, 2015; Rogers & Portsmore, 2004). Ayrıca birçok çalışma eğitsel robotlarla çalışmanın problem çözme stratejilerinin geliştirilmesi için güçlü bir motivasyon sağladığını

raporlamıştır (Bergen, 2001; Mauch, 2001; Norton, McRobbie & Ginns, 2007). Robotik faaliyetlerin eğitsel çekiciliğinin bir nedeni de birden fazla tasarım türünü içermeleridir. Bunlar;

- **Fiziksel tasarım:** Yapı parçalarının ve mekanizmaların tasarımı
- **Bilgisayımsal tasarım:** Bir yapı bloğunun nasıl hareket edeceğini ve yanıt vereceğini belirleyen bilgisayar programının yazılmasını içeren tasarımdır.

Robot yapım setleri farklı eğitim alanında çok çeşitli öğrenme stillerini desteklemek için kullanılabilir. Bununla birlikte robotik uygulamaların eğitim ortamlarında hâlihazırda tanıtıldığı yolun oldukça dar kapsamlı olduğu görülmektedir. Robotik eğitimleri çoğu zaman katılımcıların motor komutlarıyla robotu hareket ettirdiği ve basit sensör uygulamaları gerçekleştirdiği etkinliklerle sınırlı kalmaktadır. Daha geniş bir uygulama yelpazesi geliştirmenin daha farklı ilgi alanlarına hitap etme potansiyelini elinde bulunduracağı unutulmamalıdır. Rusk vd. (2008) öğrencilerin robotik teknolojilere daha geniş bir bağlamda katılım sağlaması amacıyla eğitimcilere 4 temel strateji önermiştir. Bunlar;

- **Zorluklardan ziyade temalara odaklanma.** Tek bir tasarım zorluğuna (örn. bir parkur üstündeki engellere göre manevra yapma) odaklanmak yerine ortak bir tema (örn. yeni programlanabilir müzik aletleri yaratma; çeşitli giyilebilir eserler üretme; bir kitaba veya filme dayanarak robotik karakterler yaratma gibi) belirleyip, çalışma gruplarının bu tema çerçevesinde çalışmasını sağlamak.
- **Sanat ve mühendisliği birleştirme.** Öğrenciler sanat ve mühendisliği birleştiren disiplinler arası projeler yaratma sürecinde de mühendislik kavramlarını öğrenebilir. Örneğin yürürken renk değiştirmeye programlanmış ışıklar ile donatılmış bir ayakkabı tasarımı, bir müzik aleti tasarımı, programlanabilir bir su çeşmesinin yapılması gibi.
- **Hikâyelemeyi teşvik etmek.** Bazı çocuklar nesnelerin mekanik özellikleri üzerine odaklanmayı tercih ederken, bazıları nesneleri rol yapma ve sosyal etkileşim aracı olarak görmektedir.

- **Yarışmalardan ziyade sergilerin organize edilmesi.** Birçok robotik etkinlik yarışma biçiminde yani öğrencilerin belirli kurallar içerisinde tasarım ve programlama yaptığı etkinlikler yoluyla gerçekleştirilmektedir. Sergi formatının açık uçlu yapısı daha geniş bir yelpazedeki yetenekleri destekleyebilir. Böylece hem yaratıcı yeteneklerin ortaya konulmasına olanak tanınıp, hem de projelerin herkese açık gösteriminin motivasyonel faydaları korunabilir (Turbak & Berg, 2002).

BD becerilerinin geliştirilmesi giderek daha önemli hale geldikçe, bu becerilerin geliştirilmesine fırsat sağlayan öğretimsel uygulamaların hazırlanmasına olan ihtiyaç da artmaktadır (Burleson vd., 2018). Bu konuda özellikle öğretmen eğitimleri ve atölye çalışmaları büyük önem taşımaktadır. Avrupa'daki bazı öğretmen eğitimi kurumları tarafından 2006-2009 yılları arasında yürütülen TERECoP projesinde robotik eğitimi ve yapılandırmacı pedagojik yöntemler temel alınarak hizmet içindeki öğretmenlere yönelik pilot atölye çalışmaları, kurslar ve uluslararası çalıştaylar düzenlemiştir (Alimisis, 2009). Geniş katılımlı (6 Avrupa ülkesinden 8 eğitim kurumu) bu proje sonunda ortaokul öğrencilerine yönelik çeşitli müfredat tasarımları, yapılandırmacı öğrenme yöntemleri (temel öğrenme ilkeleri, hedef ve stratejiler) ve robotik eğitimine ilişkin materyaller üretilmiştir. Ülkemizde de benzer atölye çalışmalarına rastlamak mümkün olsa da bu çalışmaların genellikle kısa süreli (1 haftadan kısa) ve bölgesel nitelikte olduğu görülmektedir.

Mevcut tez çalışmasında robotik uygulamalar için Lego Mindstorms NXT 2.0 eğitsel robot seti tercih edilmiştir. MIT Media Laboratuvarlarındaki araştırmalardan esinlenilerek ortaya çıkarılan Lego Mindstorms robotik seti özellikle mobil robotlar geliştirmek için oldukça uygundur. Lego Mindstorms seti oldukça fazla sayıdaki fiziksel yapı parçaları (yapı blokları, servo motorlar, çok tipli sensörler vs.) sayesinde öğrenenlere oldukça manipülatif bir öğrenme etkinliği sunarken aynı zamanda kullanımı basit çeşitli programlama araçlarıyla bu yapı bloklarının işlevsel hale getirilmesine de olanak sağlamaktadır. Lego Mindstorms dünya çapında birçok robot etkinliğinin düzenlenmesinde öncü olmuştur. Yapılan araştırmalar Lego eğitsel robot setlerinin okulöncesinden lisans seviyesine kadar hemen her öğrenim kademesindeki öğrenenler için BD kavramlarının öğrenilmesi, BD ve problem çözme stratejilerinin geliştirilmesi ve bu stratejilerin gerçek yaşam düzenlemeleri içerisinde uygulanması noktasında etkili bir öğrenme çevresi sunduğunu

belirtmektedir (Bers vd., 2014; Flórez vd., 2017). Ayrıca doğrudan bilgisayar kullanımını gerektirmeyen Lego Mindstorms seti programlama ile ilgili kaygıların azaltılmasını ve daha geniş bir kullanıcı kitlesine ulaşılmasını sağlamaktadır. Bununla birlikte bilgisayımsal yetkinliklerinin geliştirilmesinde robotik araçların fiziksel varlıklarından ziyade pedagojik açıdan sağladıklarına odaklanılması gerektiği unutulmamalıdır.

Yapa-Boza Öğrenme

Yapıp bozma (tinkering), hem acemilerin hem de uzmanların programlamayı öğrenirken ya da programlama yaparken gerçekleştirdikleri otantik bir süreçtir. Eğitsel robot programlama, yapa boza öğrenmeyi destekleyen ve teşvik eden keşfedici bir etkinliktir. Yapa-boza öğrenmek hem bir yönelimi hem de bir dizi etkinliği tanımlamak için kullanılmaktadır. Yapa-boza öğrenme yönelimi esasen hiçbir kurama dayanmayan, bir bilgisayar programının yaratılması ya da değiştirilmesi sürecinde hiçbir plana sahip olmama ya da buna ihtiyaç duymamayı betimler. Yapa-boza öğrenme olarak tanımlanan etkinlikler, deneme-yanılma, kurcalama, geri bildirim mekanizmalarını (test gibi) bulma ve kullanma gibi faaliyetlerin kombinasyonlarını içerir. Yapa-boza öğrenmenin iteratif ve değişken doğası yeni başlayanların programlamayı öğrenmesinde yardımcı olabilir. Yapa-boza öğrenmede yer alan aşağıdan-yukarıya ya da tahmin et-kontrol et süreçleri (Burke & Kafai, 2010; Resnick vd., 2009) programlama dilindeki veya ortamındaki anahtar öğelerin keşfini destekler (Perkins, Hancock, Hobbs, Martin & Simmons, 1986). Birçok araştırmacı yapa-boza öğrenmenin programlamaya yeni başlayan öğrenenler için kritik bir öneme sahip olduğunu ileri sürmektedir. Bu anlamda yeni başlayanların program geliştirirken yararlandığı düşünce ve stratejiler üzerine yapılan araştırmalar kuramcılara ve eğitimcilere BD ile ilgili öğrenme ve öğretim pratiklerinin geliştirilmesinde ve uygulanmasında bir çekiş gücü sağlayabilir (Berland vd., 2013).

Yapılan çalışmalar yoğun ve sistematik bir şekilde (gelişigüzel değil) yapıp bozan acemilerin daha fazla öğrenme kazanımı elde ettiğini göstermektedir (Perkins vd., 1986). Bazı öğrenenler bir çıkmazla karşılaştıklarında, bir çözüme ulaşmak için sürekli olarak kodları değiştirip, yeniden işletip, planlama becerilerini işe koşarak programlama hakkında daha fazla çıkarım elde ederken; bazıları sorunlar karşısında programlamadan vazgeçme eğilimi sergilemektedir. Yapa-boza

öğrenme keşif ve bilgi yapımını kolaylaştırmasının yanında öğrenenleri diğer yollarla da destekleyebilir. Örnek olarak, yapa-boza öğrenme iyi bir programcı olmak için gerekli planlama becerilerinin gelişimini ve yeni öğrenme yörüngelerinin keşfini destekleyebilir. Perkins vd. (1986) gelişigüzel yapıp bozmanın, öğrencilerin ilerlemesini olumsuz etkilediğini, bununla birlikte dikkatli (ya da bilinçli) yapıp bozmanın programlama beceri repertuarının geliştirilmesi için faydalı olduğunu ortaya koymuştur.

Özetle, yapa-boza öğrenme, keşfetme ve tam zamanında planlamayı içeren keyifli bir etkinlik olup, istenilen sonucu üreten bir eylemin nedenlerinin sorgulanmasını teşvik eden ve çevreden geri bildirim üreten bir süreçtir (Berland vd., 2013). Eğitsel robot programlama bu bağlamda yapa-boza öğrenme için oldukça elverişli koşullar sunan bir öğrenme etkinliği olarak ön plana çıkmaktadır.

Soyutlama

Wing (2006), BD üzerine öncü kabul edilen çalışmasında BD'nin özünde soyutlama ve ayrıştırmanın olduğunu belirtmiştir. Soyutlama becerisi ya da bir soyutlama yaratmak, daha az önemli özellikleri geri plana atarken, fikrin önemli yönlerini ön plana çıkarma, fikri ya da süreci daha genel tanımlamalarla kavramsallaştırma ve temsil etme yeteneği gerektirir (Weintrop vd., 2016). Soyutlamalar oluşturmak ve kullanmak farklı disiplinlerde karmaşıkla başa çıkmak için sıklıkla işe koşulmaktadır. Soyutlama bir program yazarken, bir fikir ya da bulguyu iletmek için görselleştirmelerden yararlanırken, bir problemin amacını ya da kapsamını tanımlarken ve verilen problemi ya da olguyu daha fazla irdelemek veya anlamak için modeller yaratırken sıklıkla işe koşulan bir beceridir. Sullivan ve Heffernan'a (2016) göre soyutlama, modeller ve simülasyonlar yoluyla bir problemin yeni bir temsilini yaratmaktadır. Örnek olarak bir algoritma, girdileri kullanarak bir dizi adımın işletilmesiyle istenen sonucu elde etmeye çalışan işlem dizisinin bir soyutlamasıdır.

BD aynı zamanda algoritmik düşünme süreçlerini de kapsamaktadır. Algoritmik düşünme, aşamalı bir düşünme biçimidir (stepwise thinking). Algoritmaya ilişkin birçok tanımlama olmasına rağmen, genel olarak bazı girdilerin ve başlangıç durumlarının nasıl ele alınacağına ve buna ilişkin çıktıların ya da sonuç durumlarının nasıl üretileceğine ilişkin bir dizi kural seti olarak karakterize edilmektedir (US

National Research Council, 2010). Algoritmalar bilgisayar bilimlerinde ve matematikteki temel düşünme araçlarından birisidir (Wilkerson-Jerde, 2014). Deneyimli bilgisayar bilimcileri bilgisayımsal problemleri herhangi bir özel programlama dilinden bağımsız olarak belirli bir soyutlama düzeyinde analiz eder ve çözer. Algoritmaları anlayabilmenin bir başka önemli yönü belirli bir algoritmanın verilen bir girdi için ne tür bir çıktısının olacağını tahmin edebilmektir. Yani programcı aynı zamanda hazırladığı programa ait zihinsel bir model geliştirip, programın nasıl çalışacağını yordamalıdır (Robins vd., 2003). Bu da hangi ayrıntıların önemsenmesi ve hangilerinin göz ardı edilmesi gerektiğine karar verme olan soyutlama süreci ile yakından ilintilidir. Soyutlamaların temelinde analoji, modelleme ve simülasyonlar yatmaktadır. Otantik dünya olgularına ilişkin temsili sistemler yaratırken, kullanıcılar hangi problemi çözmek ya da derinlemesine anlamak istediklerine dayalı olarak otantik dünyanın hangi yönlerinin modelleneceğini de seçer (Basawapatna, 2016).

Problem Çözme

BD, bilgisayar gibi düşünme olmayıp bir problemin tüm yönlerini anlamayı, karmaşıklık içinden sadelikleri sezmeyi ve yeterli kaynaklarla başarılabilecek en uygun çözümleri bulmayı içeren bir yaklaşımdır. BD'nin her şeyden önce bir problem çözme yöntemi olduğu unutulmamalıdır. Dolayısıyla BD sürecinin yönlendirilmesinde problem çözme süreçlerini temele alan öğrenme yaklaşımlarından yararlanmak oldukça makul bir yaklaşım gibi gözükmektedir.

Problem çözme ve probleme dayalı öğrenme. Problem çözme ve probleme dayalı öğrenme (PDÖ) kavramlarının sıklıkla birbiri yerine kullanıldığı görülmektedir. Aslında PDÖ eğitimde problem çözme süreci kadar uzun bir geçmişe sahip değildir. Problem çözmenin tanımı öğrenme kuramları perspektiflerine göre değişmektedir. PDÖ'nün kökleri tıp uzmanlığı üzerine yapılan ve varsayımsal-tümdengelimli bir akıl yürütme sürecini vurgulayan araştırmalara dayanmaktadır (Barrows & Tamblyn, 1980). PDÖ bir problemin çözümünü anlamaya yönelik çalışma sürecinden doğan öğrenmedir. PDÖ sıklıkla sorgulamaya dayalı diğer öğrenme yöntemleriyle karıştırılmaktadır. Sorgulamaya dayalı öğrenme ve PDÖ gibi eğitimdeki çoğu yenilikçi yaklaşım öğrenmeyi problem çözme ya da karmaşık olgu soruşturma içerisine yerleştirmektedir (Hmelo-Silver, Duncan & Chinn, 2007). Alanyazındaki PDÖ ve sorgulamaya dayalı öğrenmenin çok çeşitli örneklerini incelendiğinde, bu iki yaklaşımı açık bir biçimde birbirinden ayıran herhangi bir

boyuta rastlanmamıştır. Hem PDÖ hem de sorgulayıcı öğrenme otantik problemler ve sorular etrafında organize edilmektedir. Her ikisi de işbirlikli öğrenme ve etkinliklere ağırlık vermektedir. İkisinde de öğrenciler bilişsel olarak sezme, kanıta dayalı açıklamalar geliştirme ve sunmaya çalışmaktadır. Öğretmen öğrenme sürecini kolaylaştırmak için önemli bir rol üstlenir ve tam zamanında içerik bilgisi sağlar. PDÖ ve sorgulayıcı öğrenme arasında algıladığımız temel ayrım onların kökenleridir. PDÖ, genellikle problem verisini kullanır. Öte yandan sorgulayıcı öğrenme bilimsel sorgulama yöntemine yaslanarak soru sorma, veri toplama, verileri analiz etme ve kanıt temelli argümanlar oluşturma üzerine yoğun bir vurgu yapar (Krajcik & Shin, 2014; Kuhn, Black, Keselman & Kaplan, 2000). Öte yandan, PDÖ'yü diğer sorgulama tabanlı öğrenmelerden ayıran karakteristik özelliklerinden birisi problemin öğrenme sürecinin başında yani müfredat girdilerinden önce verilmesidir.

PDÖ, öğrencilerin problem çözme yoluyla öğrendikleri bir öğretim yöntemidir. Öğrenciler problem çözme deneyimleriyle hem konu alanı öğrenmelerini hem de düşünme stratejilerini öğrenebilir (Hmelo-Silver, 2004). PDÖ'de öğrenci öğrenmesi tek bir doğru cevaba sahip olmayan karmaşık bir problem üzerine kuruludur. Öğrenciler bir problemi çözmek için nelere ihtiyaç duyduklarını tanımlamak için genellikle işbirlikli olarak çalışırlar. PDÖ özyönelimli öğrenmeyi teşvik eder. Öğrenenler edindikleri bilgileri problemlere uygulayıp öğrendikleri ve kullandıkları stratejilerin etkililiğine ilişkin yansımaları iletir. PDÖ, öğrenenleri otantik problemlerle uğraşmaya ve öğrenmelerinin sorumluluğunu almaya teşvik etmesi yönüyle onların aktif öğrenen haline getirilmesi için oldukça uygun bir pedagojidir. PDÖ'nün öğrencilerin strateji geliştirme ve bilgi inşa etmelerine yardımcı olmak üzere iki temel vurgusu vardır.

Okul öncesinden lisans sonrasına kadar eğitim kurumlarının temel görevi öğrencilere bilişsel beceriler kazandırmaktır. En önemli bilişsel becerilerden birisi hiç şüphesiz problem çözme yeteneğidir. Problem çözme öğretiminde problemler genellikle iyi yapılandırılmış (well-defined), yani problemin çözümü için gerekli tüm bilgileri barındıracak şekilde sunulur ve doğru cevabı garanti eden açık bir prosedür ile çözüme kavuşturulur (Frederiksen, 1983). Fakat gerçek yaşamda karşılaştığımız sosyal, politik, ekonomik ve bilimsel problemlerin tamamına yakını kötü yapılandırılmıştır (ill-defined) (Simon, 1973). Kötü yapılandırılmış problemler genel

olarak daha karmaşık olup, sorunun ne zaman çözüldüğünü belirlemek için daha az kesin kritere sahiptir. Öğrenciler bu tür problemlerin çözümü için başlangıçta tüm bilgilere sahip değildir. Problem çözme süreçlerinin iyi ve kötü yapılandırılmış problemler için temelde benzer olduğu ancak bilgi işleme perspektifinden bakıldığında kötü yapılandırılmış problemler için uzun süreli bellekten veya dış kaynaklardan farklı unsurlar ortaya çıktıkça kişinin daha geniş bir zihinsel tanılama repertuarı geliştirdiği görülmektedir. Bu yönden bakıldığında kötü yapılandırılmış problem çözümünün daha dinamik bir süreç olduğu söylenebilir. Simon (1973), kötü yapılandırılmış problemlerin çoğu zaman bir dizi iyi yapılandırılmış alt problemlere ayrıştırılarak çözüldüğüne dikkati çekmiştir. Yani problem, alt problemlerde iyi yapılandırılmış, ancak büyük ölçekte kötü yapılandırılmış olabilmektedir.

PDÖ karmaşık görevlerde konumlanmaktadır. Bu tarz görevler öğrencilerin anlamdırmalarla meşgul olmasına, soruşturma ve problem çözme süreçlerini yönetmelerine fırsat verip, düşüncelerini ifade etme ve öğrendiklerini yansıtma için onları cesaretlendirmektedir (Quintana vd., 2004). PDÖ görevleri öğrenciler için birçok yönden zorlayıcıdır. Öğrencilerin süreçte karşılaştıkları kavramsal ve pratik engelleri aşmalarına yardımcı olunması bakımından uygulanacak destek stratejileri oldukça önem arz etmektedir (Hmelo-Silver vd., 2007). Araştırmacılar problem çözme sürecinde sağlanan desteklerin (iskelelerin) bilişsel yükü azaltabileceğini, uzman rehberliği sağlayabileceğini ve öğrencilerin disiplinli düşünme ve hareket etme yolları edinmelerine yardımcı olabileceğini belirtmektedir. Yapı iskelesi yakınsak gelişim alanı içerisinde karmaşık ve zor görevleri daha erişilebilir ve yönetilebilir olanlarla değiştirerek öğrenmeyi öğrenciler için daha izlenebilir ve yerine getirilebilir hale getirir (Vygotsky, 1978). Yapı iskelesinin önemli bir özelliği öğrencilerin hem görevin nasıl yapılacağını hem de görevin bu şekilde yapılmasının gerekliliğini öğrenmelerini desteklemesidir. İskele, öğrencilere sadece görevlerin karmaşıklıkları boyunca rehberlik etmekle kalmaz, aynı zamanda öğrencileri önemli disipliner çerçeve ve stratejilerle ilişki kurmaya zorlayacak şekilde çalışmalarının önemli yönlerini sorunsallaştırabilir (Reiser, 2004). Öğrenme iskeleleri eğitsel yazılım ya da müfredata özgü materyaller karşısında sıklıkla öğrenme çevresine, öğretmenlere, akranlara ve öğrenenlerin kendilerine dağıtılmaktadır (Puntambekar & Kolodner, 2005). Öğretmenler öğrencilerin akranlarıyla birlikte görevlerle mantıklı ve üretken bir biçimde meşgul olmasını sağlar. Bununla birlikte alanyazında

öğretimsel rehberliğin etkisine dair anlaşmazlıklara da rastlanmaktadır (Hmelo-Silver vd., 2007; Kirschner, Sweller & Clark, 2006). Kirschner vd. (2006) yönlendirmesiz ya da en asgari düzeyde yönlendirmelere dayalı keşifsel öğrenmelerin daha üstün öğretimsel çıktılar ürettiğini destekleyen deneysel çalışmaların bulunmadığına dikkati çekerek özellikle acemi öğrenciler için en asgari düzeyde yönlendirmelerin kavram yanılgıları, eksik ya da dağınık bilgi edinilmesi gibi olumsuz sonuçlara yol açabileceğini belirtmiştir. Buna göre doğrudan ve güçlü öğretimsel rehberliğin daha verimli sonuçlar üreteceği ileri sürülmüştür. Ne var ki araştırmacılar keşfedici öğrenme, PDÖ, sorgulamaya dayalı öğrenme, deneyimsel (experiential) öğrenme ve yapılandırmacı öğrenmeyi yönlendirmesiz ya da en asgari düzeyde yönlendirmeye dayalı yaklaşımlar altında gruplamıştır. Yani doğrudan öğretimsel rehberlik sağlanan yaklaşımlarla asgari yönlendirmeli öğretimsel yaklaşımları aynı çatı altında birleştirerek bu yaklaşımların etkisiz ve verimsiz olduğunu iddia etmişlerdir. Hmelo-Silver vd. (2007) ise söz konusu çalışmanın pedagojik ve dayandırıldığı ampirik kanıtlar yönünden hatalı olduğunu belirtmiştir. Hmelo-Silver vd. (2007), pedagojik perspektiften bakıldığında Kirschner vd'nin (2006) en asgari düzey (minimum) rehberlikli öğretim kategorisi altında, farklı pedagojik yaklaşımları rastgele bir araya getirdiğini belirtmiştir. Bu yaklaşımlardan büyük bölümünün özellikle de sorgulayıcı öğrenme ve PDÖ'nün en az yönlendirmeli yaklaşımlar olmadığını, aksine öğrenci öğrenmesini kolaylaştırmak için kapsamlı bir iskele ve rehberlik sağlanan yaklaşımlar olduğunu savunmuştur. Hmelo-Silver vd. (2007) ayrıca sorgulayıcı öğrenme ve PDÖ'nün öğretimsel etkinliği desteklediğine ilişkin birçok ampirik kanıt olduğunu belirtmiştir. Bu kanıtların standartlaştırılmış testlerdeki öğrenci başarısının yanında derin ve anlamlı öğrenmeyi desteklediğini ileri sürmüştür.

PDÖ, salt öğrenci keşfine dayalı bir yaklaşım değildir. Aksine bu yaklaşımlar öğrencilere önemli yönlendirme ve destekler sağlamaktadır. PDÖ'nün işe koştuğu stratejiler doğrudan eğitimi bile içerebilir. Bununla birlikte PDÖ bağlamında, doğrudan eğitim tam zamanında sağlanmalı ve genellikle öğrenciler sunulan bilgileri bilme ihtiyacını yaşamalıdır (Edelson, 2001). Öğrencilere önemli bilgiler sunan yani hedef odaklı küçük bir ders, öğrencilerin sunulan bilgilerin gerekliliği kadar aynı zamanda problem çözme ve araştırma uygulamalarıyla ilişkisini anlamasına

yardımcı olabilir. Bu türden doğrudan yönelimli talimatlar bilginin gelecekte ilgili bağlamlarda kullanılmasını teşvik edebilir.

Problem çözme sürecine kılavuzluk edilmesi ya da bu sürecin desteklenmesi bilişsel yükü azaltıp, öğrencilerin karmaşık durumlardan öğrenmesine olanak tanır. Bu nedenle problem çözme sürecini bir çerçeve (yöntem ve stratejiler) kılavuzluğunda sürdürmek oldukça verimli sonuçların elde edilmesini sağlayabilir.

Problem çözme yöntem ve stratejileri. Problem çözme, öğrenme sürecinde sıklıkla ortaya çıkan en önemli bilişsel işlem türlerinden biridir. Problem, bir hedefe ulaşmaya çalışılan bir durum olduğunda ve oraya ulaşmak için bir yol bulunması gerektiğinde ortaya çıkar. Bununla birlikte her öğrenme problem çözme değildir. Beceriler amaçlara ulaşmak için otomatik olarak işletilecek şekilde düzenlendiğinde problem çözme süreci gerçekleşmez. Ayrıca problem çözme öğrencilerin ne yapacaklarını bildikleri düşük düzeyli öğrenmelerde kullanılamaz. Problem çözme, bireylerin hâlihazırda bir çözümün olmadığı durumda hedefe ulaşma çabalarının bütünüdür (Schunk, 2012). İçinde sürpriz, yani belirsizlik barındırmayan ve ne kadar zor olursa olsun rutin veya belli prosedürlerle çözülen problem sadece bir alıştırmadır (Psycharis & Kallia, 2017).

Çok geniş yelpazedeki durumlar problem olarak nitelendirilebilir; bir soruya cevap vermek, bir değeri hesaplamak, bir nesneyi yerleştirmek, bir öğretim süreci gerçekleştirmek, bir işi ya da nesneyi güvenceye almak gibi. Görüldüğü gibi problem belirli bir akademik disipline, mesleki alana ya da günlük yaşama ilişkin olabilir. İçerik alanı ve karmaşıklığı ne olursa olsun tüm problemlerin bazı ortak özellikleri vardır. Bunlar; başlangıç durumu (problem çözenin mevcut durumu ve bilgi düzeyi), amaca ulaşma (alt amaçlar dâhil) odaklı olma, başlangıç durumları ve alt amaçlar üzerinde işlemler (bilişsel ya da davranışsal etkinlikler) gerçekleştirme gereksinimidir (Anderson, 2009).

Tarihsel açıdan bakıldığında problem çözme süreç ve yöntemlerinin öğrenme kuramları etrafında şekillendirildiği görülmektedir. Örneğin Thorndike (1913) problem çözmeyi deneme-yanılma olarak ele almıştır. Deneme yanılmada yapılan girişimler sonuca bağlı olarak şekillenir. Bununla birlikte problem çözme girişiminin sayısının artması bireylerde çaresizliğe yol açabilir. Deneme-yanılma zaman alıcı, güvenilir ve etkin olmayan ve en uygun sonuca ulaşmayı garanti etmeyen bir

problem çözme yöntemi olmasına rağmen günlük hayatta sorunları çözmek için sıklıkla başvurduğumuz yöntemlerden birisidir.

Bazı araştırmacılar problem çözmeyi genellikle bir problem karşısında olası bir çözümün ani farkındalığı (içgörü) olarak açıklamıştır. Wallas (1926) problem çözme sürecinin hazırlık (bilinçli ve istemli olarak problemi anlama ve çözüme ilişkin bilgi toplama), kuluçka (bazen problemi bir yana koymayı da içerecek şekilde problem üzerinde örtük düşünme), aydınlanma (olası bir çözümün farkındalığına aniden ulaşma yani içgörü süreci) ve doğrulama (çözümün işe yarayıp yaramadığının test edilmesi) aşamalarından oluştuğunu belirtmiştir. Birçok Gestalt psikoloğu problem çözmenin içgörüsel bir algı değişikliği sonucu gerçekleştiğini ileri sürmektedir. Öğrenciler tarafından sıklıkla kullanılmasına rağmen alışılagelmiş ezberin okul dışındaki yaşamda verimsiz ve nadiren kullanıldığını belirten Gestalt psikologları bir olayın anlamını kavramanın ya da performansın temelini oluşturan kural ya da yönlendirmeleri kavramanın önemini vurgulamaktadır.

Problem çözmenin bir başka yolu genellikle bir çözüme ulaşmaya ön ayak olan ilke ya da temel kuralları işe koşan genel problem çözme yöntemlerini yani buluşsal yolları kullanmaktır (Anderson, 2009). Polya (1957) bir problemi anlamak için buluşa yönelik bir veri listesi sunmuş ve bunu öğrencileri bir problemin çözümüne kavuşturmak için bir plan ve rehber sorular dâhilinde tasarlamıştır. Polya'nın önerdiği bu yöntem problemi anlama, plan geliştirme, planı uygulama ve çözümü gözden geçirme olmak üzere ardışık dört aşamadan oluşan zihinsel süreçleri kapsamaktadır.

Bransford ve Stein (1984) Polya'nın önerdiği yönteme benzer bir buluşsal çerçeve önermiştir. IDEAL olarak adlandırılan bu model problemin belirlenmesi (1), problemin açıklanması ve temsil edilmesi (2), muhtemel çözüm stratejilerinin araştırılması ve keşfedilmesi (3), stratejilerin eyleme dökülmesi (4) ve geri bakma ve yapılanların etkisini değerlendirme (5) adımlarını içermektedir.

Yaratıcı problem çözme modeli genel nitelikli bir diğer problem çözme çerçevesidir (Treffinger, 1995; Treffinger & Isaksen, 2005). Bu model üç temel bileşenden oluşmaktadır: Zorluğu anlamak, fikirler üretmek ve eylem için hazırlanmak. Üstbilişsel süreçler (planlama, izleme ve değerlendirme gibi) süreç boyunca sergilenir (Schunk, 2012). Problem çözmek genel bir amaç ya da yönelimle

başlar. Önemli veriler elde edildikten sonra (olay, görüş ya da kaygılar) özel bir amaç ya da soru formüle edilir. Fikir üretmenin ayırt edici yanı amacı gerçekleştirmeye yönelik seçenekler üretmek için ayrık düşünmedir. Eylem için hazırlanmak; öne çıkan seçenekleri incelemek, dirençlerin üstesinden gelme yollarını bulmak ve destekleyici kaynaklara ulaşmaktır.

İnsanlar alana ya da disipline özgü beceriler edinmek adına konu alanına aşina oldukları içerikle çalışırken genellikle prosedürel bilgileri kullanır. Bu nedenle genel nitelikli buluşsal problem çözme yöntemleri içerik alanına aşina olunmadığında oldukça faydalıdır. Bu yöntemler problem çözme yöntemini sistematik bir yolla destekler. Buluşsal yöntemlerin çoğu ardışık adımlardan oluşsa da bu adımların izlenmesinde esneklik bulunmaktadır.

Newell ve Simon (1972) problem çözme konusunda bir başlangıç durumu, amaç durumu ve işlemlerin gerçekleştirilmesini gerektiren ve sıklıkla alt amaçların üretilmesine yol açan olası çözüm yollarından oluşan bir bilgi işleme modeli önermiştir. Bu modele göre öğrenciler problemin zihinsel bir temsilini oluşturduktan sonra başlangıç ve hedef durumları arasındaki tutarsızlığı azaltmak için işlemler gerçekleştirir. Araç-sonuç analizi olarak adlandırılan bu yöntem problem çözücünün sürekli olarak bulunduğu durumla istenilen durumu karşılaştırdığı bir sezgiye dayanmaktadır. Problem çözme süreci "Şimdi olduğum yer ile nerede olmak istediğim arasındaki fark nedir?" ve "Bu farkı azaltmak için ne yapabilirim?" şeklindeki rehber soruları eşliğinde yürütülür (Frederiksen, 1983). Bir çözüm bulmak için temsil üzerinde çalışma süreci, arama olarak adlandırılır (Andre, 1986). Bu yöntem bilgi işleme stratejilerini temele alır. Problem çözmede ilk adım, zihinsel bir temsil oluşturmaktır. Polya'nın ilk adımı olan problemi anlamaya benzer olarak temsil etme, bilinen bilgiyi bellekte bir modele çevirmeyi gerektirir. İçsel temsil çalışan bellekteki önermelerden ve imgelerden oluşur. Problemin zihinsel temsili çeşitli araçlar (kâğıt ya da dijital araçlar) ile dışa vurulabilir. Çalışan bellekteki bilgiler uzun süreli bellekteki (USB) ilgili bilgileri aktif hale getirir ve problem çözen sonunda bir problem çözme stratejisi seçer. Stratejiler uygulandıkça başlangıçtaki zihinsel temsil değişebilirken özellikle problem çözme girişimi başarısız olduğunda yeni bilgiler aktifleştirilir. Bu durumda amaçtaki ilerleme de değerlendirilir. Problem temsili, bellekte hangi bilginin aktifleştirildiğini ve sorunun çözümünün ne kadar kolay olduğunu belirler. Eğer kişi problemi çok yönlü ele almazsa ya da çok fazla

kısıtlayıcı kullanarak zihninde problemin yanlış bir temsilini oluştururursa arama sürecinin doğru bir çözüm yolu belirlemesi pek olası değildir (Chi & Glaser, 1985).

Alanyazın incelendiğinde tıpkı problem çözme yöntemleri gibi çeşitli problem çözme stratejilerinin de önerildiği görülmektedir. Bu stratejilerden bazıları: üret ve test et, araç (yol)-sonuç analizi, analojik akıl yürütme ve beyin fırtınasıdır. Üret-test et stratejisi sınırlı sayıda çözüm önerisinin amaca ulaşmadaki etkililiğinin sırayla test edilmesini kapsamaktadır. Örneğin eve geldiğinizde düğmeye dokunduğunuz halde lambanın yanmaması durumuyla karşılaştığınızda olası nedenleri düşünüp, stratejilerinizi teker teker uygular ve çözüm üretmeye çalışırsınız. Araç-sonuç analizinde mevcut durumu hedefle karşılaştırılır ve bu iki durum arasındaki farklar tanımlanır (Resnick, 1985). Alt hedefler, mevcut durum ile hedef arasındaki farklılıkları azaltmak için kurulur. Alt hedefleri gerçekleştirmek için işlemler gerçekleştirilir ve bu işlemler asıl hedefe ulaşılana kadar devam eder. Araç-sonuç analizi güçlü bir problem çözme sezgiselidir. Alt hedefler uygun bir şekilde tanımlandığında araç sonuç analizinin problemi çözmesi oldukça olasıdır. Bir dezavantajı, çok sayıda alt amaç içeren karmaşık problemlerin çalışan belleğe ek yük bindirmesidir. Ayrıca tek bir alt amacın gözden kaçırılması problemin çözümüne engel olabilir.

Bir diğer genel problem stratejisi olan analojik akıl yürütme problem durumuyla aşina olunan bir başka durum arasında bir analoji kurulmasını içermektedir. Aşina olunan bir durum ile üzerinde çalışılan durum daha sonra problemin çözümüyle ilişkilendirilir. Analojik akıl yürütme USB'de bilinen bir alanın ağına erişerek bu ağı çalışan bellekteki problem durumuna aşinalık (bilinirlik) yaratmak için kullanmayı içerir. Başarılı bir anolojik akıl yürütme bilindik durumun problem durumuyla benzer yapısal özelliklere sahip olmasını gerektirir (örneğin güneş sistemiyle moleküler yapılar arasında benzerlik kurulması gibi). Gick ve Holyoak (1980, 1983) analojik problem çözmenin gücünü ortaya koymuştur. Öğrencilerin zor bir tıbbi sorunu askeri bir problemle analoji kurarak çözmesini sağlamıştır. Araştırmacıların açık bir yönerge kullanmaksızın tıbbi sorunun çözülmesi için askeri sorunun kullanılmasına yönelik basit bir ipucu vermesi problem çözme sürecini oldukça geliştirmiştir. Bu stratejinin etkili olması için aşina olunan alan ile problem alanlarının iyi bilinmesi gerekir. Çünkü yetersiz bilgi olması

durumunda öğrencilerin problem ile analoji kurulan durum arasındaki ilişkiyi görmesi olası değildir.

Beyin fırtınası olası problem çözümlerini formüle etmek için yararlı olan genel bir problem çözme stratejisidir (Isaksen & Gaulin, 2005). Beyin fırtınası aşamaları sorunun tanımlanması, herhangi bir değerlendirme yapılmaksızın olabildiğince çok çözüm üretilmesi, potansiyel çözümleri yargılama kriterlerine karar verilmesi ve en iyi çözümü seçmek için bu kriterlerin uygulanmasıdır.

Polya'nın problem çözme çerçevesi. Birçok öğrenci için buluşsal çerçeveler problem çözme stratejilerinden daha sistematik olup daha uygun çözümlere ön ayak olmaktadır (Schunk, 2012). Programlama problemlerinin çözümü esnasında, öğrenciler problem çözme, tasarım ve düşünme süreçlerini deneyimler (Soloway, 1993). Unutulmamalıdır ki BD her şeyden önce bir problem çözme sürecidir. Bir probleme çözüm aramak için sezgisel (buluşsal) akıl yürütmeyi kullanan BD yenilikler geliştirmek için önemli bir potansiyele sahiptir (Fang vd., 2017). Bu çalışmada BD sürecinde öğrencilere kılavuzluk sağlaması ve onların problem çözme sürecini desteklemesi amacıyla Polya'nın problem çözme adımlarının kullanılması teşvik edilmiştir. Polya (1957) tarafından ortaya konulan problem çözme yöntemi iyi bilinen problem çözme çerçevelerinden birisidir. Oldukça tanınan bu yöntem aynı zamanda birçok problem çözme yöntemine de ilham vermiştir. Bilişsel Akademik Dil Öğrenme Yaklaşımı (Cognitive Academic Language Learning Approach-CALLA) (Chamot, Dale, O'Malley & Spanos, 1992) ve IDEAL (Bransford & Stein, 1984) bu yöntemlerden sadece birkaçıdır. Polya'nın problem çözme yöntemi 4 temel aşamadan oluşmaktadır. Bunlar problemi anlama, plan geliştirme, planı uygulama ve gözden geçirmedir. Bu aşamalardan her biri problem çözme sürecine kılavuzluk sağlayacak birtakım rehber sorular barındırmaktadır. Ayrıca her aşamanın ne derece başarıldığının değerlendirilmesi adına gösterge niteliğinde bazı davranışlar bulunmaktadır (Tablo 4).

Tablo 4

Polya'nın Problem Çözme Aşamaları

Aşama (Faz)	Göstergeler	Rehber Sorular
1. Aşama: Problemi Anlamak	Problemi kendi cümleleriyle ifade edebilme	- Bilinmeyen nedir? - Hangi veriler bulunmaktadır? - Durum ya da Koşul (şart) nedir?

		- Bu şartı yerine getirmek mümkün müdür? - Şartlar bilinmeyeni belirlemek için yeterli midir? Yetersiz midir? Gereksiz midir? Çelişkili midir? - Bir şekil çizebilir misin? - Uygun gösterimler (formül ya da simgeleri) yapılabilir mi? - Durumu çeşitli alt bölümlere ayırıp, yazabilir misin?
2. Aşama: Plan ya da strateji tasarlama ya da geliştirme	İlgili problemleri incelemek ve bir model (desen) ya da eşitlik (denklem) aramak	- Daha önce bu problemle karşılaştın mı? Ya da aynı sorunun biraz daha farklı bir haliyle karşılaştın mı? - Benzer bir problemin çözümünü biliyor musun? - Faydalı olabilecek bir teorem biliyor musun? - Bilinmeyene bak ve aşina olduğun benzer bir bilinmeyene sahip bir problemi düşünmeye çalış. Seninkiyle ilişkili olan ve daha önce çözdüğün problemi düşün. Bunu kullanabilir misin? Sonuçlarını kullanabilir misin? Yöntemini kullanabilir misin? - Kullanımını mümkün kılmak için bazı ilave bileşenler eklemeli misin? - Problemi yeniden ifade edebilir misin? - Onu hala farklı olarak mı görüyorsun? O zaman tanımlamalara geri dön. - Sunulan problemi çözemezsen, önce ilgili bazı problemleri çözmeye çalış. Daha erişilebilir bir problem hayal edebiliyor musun? Daha genel bir problem mi? Daha özel bir problem mi? Benzer bir problem mi? - Sorunun bir kısmını çözebilir misin? Şartın/durumun sadece bir bölümünü ele alıp, diğer bölümleri at; bilinmeyenin ne kadar uzaklıkta olduğunu sonra nasıl değişebileceğini belirle. Verilerden faydalı bir şey türetebilir misin? Bilinmeyenleri

		belirlemek için uygun diğer verileri düşünebilir misin? - Bilinmeyeni veya verileri (gerekirse her ikisini) değiştirebilir misin? Yeni bilinmeyen ve yeni veriler birbirine daha yakınlaştı mı? - Tüm verileri kullandın mı? - Bütün koşulu kullandın mı? - Problemle ilgili tüm önemli kavramları dikkate aldın mı?
3. Aşama: Planı ya da stratejiyi yürütmek	Bir strateji uygulamak. İşlemleri, operatörleri ve bağlantıları kontrol etmek	- Plan adımlarının ya da aşamalarının doğru olduğunu açıkça görebiliyor musun? - Doğru olduğunu kanıtlayabilir misin?
4. Aşama: Geriye bakma ve çözüm planını gözden geçirme	Elde edilen çözümü incelemek	- Sonucu kontrol edebilir misin? - Çıkarımını kontrol edebilir misin? - Sonuçları farklı bir şekilde türetebilir misin? Bir bakışta görebiliyor musun? - Yöntemin farklı problemler için de kullanılabilir mi?

Problemi çözme sürecinin ilk aşaması öğrenenin problemi anlamak zorunda olduğu aşamadır. Bu aşamanın temel göstergesi öğrenenin problemi kendi cümleleriyle ifade etmesidir. Bunun için bilinmeyenin, eldeki verilerin, koşul ve şartların anlaşılması gerekmektedir. Çözüm planı ya da stratejisi geliştirme aşaması veriler ile bilinmeyen arasında bir bağlantının kurulduğu aşamadır. Eğer öğrenen doğrudan (hızlı) bir bağlantı bulmazsa, ek (yardımcı) problemleri düşünmek zorunda kalabilir. Ancak en sonunda çözümün bir planını elde etmelidir. Öğrenen daha önce karşılaştığı benzer nitelikteki problemlerden yola çıkarak, ancak farklılıkları da gözeterek, karşısındaki problemi yeniden ele alır. Bu aşamada cevap verilmesi gereken temel soru problemin nasıl bir yol izlenerek çözüleceğidir. Adından da anlaşılacağı gibi üçüncü adım çözüm planının işe koşulduğu aşamadır. Geliştirilen stratejinin her adımı kontrol edilir. Bunun yanı sıra işlemler ve bağlantılar da test edilir. Bu aşama aynı zamanda geliştirilen plana yönelik sonuçların elde edildiği aşamadır. Son aşama olan geriye bakma problem çözümünün gözden geçirilip, ne kadar etkili olduğunun değerlendirildiği aşamadadır. Bu aşamada geliştirilen stratejinin farklı hangi tür problemler için kullanılabileceği düşünülür. Bu çalışmada

problem çözme süreçleri Polya'nın önerdiği rehber sorular aracılığıyla iskelenmeye çalışılmıştır.

Motivasyon

Motivasyon bireyin bir amaca yönelik gösterdiği çabanın sürekli olma eğilimi olarak tanımlanabilir. Bu tanımdan hareketle öğrenme motivasyonu, öğrenmeye yönelik çabaların süreklilik göstermesi olarak ele alınabilir (Ngan & Law, 2015). Vroom'a (1964) göre motivasyon üç yapının kolektif bir işlevidir. Bunlar beklenti (bir kişinin kendi kabiliyetine dair beklenti ve güven seviyesi), araçsallık (arzulananı gerçekleştirip gerçekleştiremeyeceği algısı) ve valanstır (sonuçlara veya ödüllere ilişkin sahip olunan tutum ya da duygusal yönelimler). Tıpkı problem çözme gibi motivasyon kavramı da öğrenme kuramlarına göre farklı şekillerde tanımlanmaktadır. Davranışsal bakış açısından bir görevle meşgul olma isteği dışsal motivasyona (açık yönergeler, ödül ve takdir, ceza, sosyal baskı ya da kabul alma isteği ve rekabet) dayalıdır (Greeno, Collins & Resnick, 1996). Bir bireyin belirli bir duruma tepki verme eğilimi ilgili bağlamda mevcut olan pekiştireç ve ceza gibi öğelere ve bireyin sonuçlarla ilgili ihtiyaç ve iç hedeflerine dayanmaktadır. Tüm organizmalar, kendi davranışları tarafından üretilen sonuçlardan büyük ölçüde etkilenmektedir (Carpenter, 1974). Bu bakış açısından motivasyon başlatma, yönetme ve sürdürme olmak üzere üç psikolojik işleve sahiptir. Kurallar ve tavsiyeler gibi sosyo bilişsel kuramın kültürün kritik unsurları olarak adreslediği aktarımlar davranışsal perspektifte de motivasyonun önemli bileşenlerinden sayılır. Çünkü bireyler ortak ya da benzer durumlar yaşamış olup bunu çeşitli (yararlı/zararlı) yollarla tanımlayanların deneyimlerinden faydalanırlar (Burton, Moore & Magliaro, 2004).

Sosyal bilişsel kuram perspektifinden hedef, beklenti, değerler ve öz yeterlik motivasyon unsurları olarak dikkat çeker (Schunk, 2012). Kişinin hedefe inanması öz yeterlik algısını yükseltir ve onu hedefe motive eder. Mevcut performansın hedefle karşılaştırılması ve ilerleme kaydedilmesiyle, insanlar iyileştirme için bir öz yeterlik duygusu yaşar. Motivasyon, sergilenen davranıştan istenen sonuçların elde edileceğine (olumlu sonuç beklentileri) ve bu davranışları gerçekleştirilebileceğine ya da öğrenebileceğine (yüksek öz yeterlik) olan inanca dayanır (Schunk, 2012). Sosyal etmenler (toplulukla hareket etme gibi) motivasyon sürecinde önemli rol oynar. Model davranışların sonuçları ya da sosyal karşılaştırmalar kişiye işlevsel

değer, sonuç ve davranışların etkililiğiyle ilgili bilgi sağlar. Sonuçlar beklenti ve öz yeterlik yoluyla motivasyonu etkiler.

Bilgi-işleme kuramı öğrenmede motivasyonun rolüne daha az vurgu yapmaktadır. Bilgi işleme perspektifinden bakıldığında sistem üzerinde bilgi akışını yönlendiren kontrol süreçleri motivasyonel işlevlere sahip gibi düşünülebilir. Öğrenenler hedeflerine ulaşmayı desteklemek için bilişsel süreçleri işe koşar. Amaçlar ve öz yeterlik gibi motivasyonel işlemler bilgi ağlarında gömülü önermeler olarak bellekte temsil edilir (Schunk, 2012).

Yapılandırmacı öğrenme anlayışına göre motivasyon bağlamsal faktörlerden, örtük kuramlardan ve öğretmen beklentilerinden etkilenmektedir. Bağlamsal faktörler öğrenme ortamlarının yapısı ve organizasyonuyla ilgili olup görev yapısı, öğrenci özerkliği ve çalışma biçimi, öğrenme etkinliklerinin çok-boyutluluğu ve değerlendirme ölçütleri gibi değişkenleri kapsamaktadır. Örtük kuramlar öğrenenlerin kendi öğrenme süreçleri ve akademik yetkinlikleri hakkında sahip olduğu anlayışlarla ilgilidir. Yapılandırmacı anlayışa göre öğrenciler öğrenmenin kendi kontrolleri altında olduğunu düşündüklerinde daha fazla çaba sarf etmekte, daha fazla öğrenme pratiği yapıp daha etkili öğrenme stratejileri kullanmaktadır. Öğrenciler öğretmenlerinin kendilerinden ne beklediklerine ilişkin olarak da birtakım inançlara sahiptir. Bu inançlar öğrenci-öğretmen etkileşimini ve öğrenci başarısını etkilemektedir.

Motivasyonla ilgili en baskın ve yaygın tanımlamalar bilişsel temellidir (Turner, 2001). Motivasyon hedefe yönelik davranışların harekete geçirilmesi ve sürdürülmesidir. Bu bakış açısı öğrenenlerin hedeflerine ulaşmak için bilişsel süreçleri (planlama, izleme ve düzenleme gibi içsel kontrol süreçleri) ve davranışları (çaba ve kararlılık) kullandığını varsayar. Görüldüğü gibi bilişsel kuramlar öğrenme ve motivasyonda zihinsel süreçlere vurgu yapar. Bilişsel bakış açısını yansıtan motivasyon çalışmalarında ortak bir vurgu bilişsel ve duygusal değişkenler üzerindedir (Tüzün, 2004). Motivasyon daha ziyade dil ve yapı iskelesi gibi destek biçimlerini de içerecek şekilde sosyo-kültürel ve öğretimsel faktörler ile bu faktörlerle etkileşim içinde bulunan bilişsel etkinliklere bağlıdır (Sivan, 1986). Bilgi nesneler, ürünler, teknolojik araç ve ortamlar ve sosyal bağlamlar dâhil olmak üzere insanlar ve çevreleri arasında dağıtıktır. Uğraşın bu yerleşik doğası motivasyonu hem bağlamı hem de bireyi önemli değişkenler olarak konumlandıran bir anlayış içerisine

yerleştirir. Buradan hareketle bu çalışmada öğrencilere sunulan farklı öğretimsel etkinliklerin (eğitsel robot programlama ve nesneye yönelik görsel programlama) motivasyona etkisinin de incelenmesine karar verilmiştir.

Motivasyon kuram ve modelleri. Geçmişten günümüze çok sayıda motivasyon kuramı ve modeli önerilmiştir. Bu kuramlardan bazıları dürtü, koşullanma, bilişsel tutarlılık ve hümanistik kuramlardır. Dürtü kuramı fizyolojik temelli bir kuramdır. Organizmayı davranışa iten ve davranışı sürekli bir çabaya dönüştüren şeyin vücut iç (homestatik) dengelerini korumak adına fizyolojik ihtiyaçlardan kaynaklandığını ileri sürer. Yönelim, ihtiyaçlar karşılanıncaya kadar sürer. Koşullanma kuramı motivasyonu bir uyarana karşı ortaya çıkan ya da uyaranların varlığında yayılan yanıtlar açısından tanımlar. Birçok kanıt pekiştiricilerin insanların ne yaptığını etkileyebileceğini göstermektedir. Buna göre davranışın ortaya çıkmasını ve sürdürülmesini tetikleyen şey bireylerin pekiştirileceklerine olan inanç ve buna verdikleri değerdir (Bandura, 1986). Bilişsel tutarlılık kuramı motivasyonun biliş ve davranışların etkileşimden kaynaklandığını ileri sürer. Bu kuram bir tür iç dengeye dayalıdır. Çünkü bilişsel yapılar arasındaki tutarsızlıklar içsel bir gerilime yol açar. Bu geriliminden kaynaklanan sorunların birbiriyle tutarlı bilişsel işlemler ve davranışlar yoluyla çözülmesi gerekir. Hümanistik motivasyon kuramları bireylerin öz farkındalığına önem verir. Motivasyon temel ihtiyaçlardan başlayarak kişinin potansiyelini en üst düzeye çıkarmasını sağlamak adına geniş ölçekteki isteklerine göre şekillenir. En bilindik hümanistik kuramlar kişinin tam potansiyelini geliştirmeye yönelik motivasyonu vurgulayan ihtiyaçlar hiyerarşisi (Maslow, 1970) ve bu süreçte öğrenmeye ve öğretime vurgu yapan kendini gerçekleştirme eğilimidir (Rogers, 1963). Güncel motivasyon kuramları bilişsel süreçlere daha çok vurgu yapmaktadır. Schunk'un (2008) motive edilmiş öğrenme modeli motivasyonun öğrenme öncesinde, sırasında ve sonrasında işlev gördüğünü ileri sürer:

- Öğrenme öncesindeki amaçlar, beklentiler (öz yeterlik ve beklenen çıktı), değerler, duygular, ihtiyaçlar ve sosyal destek
- Öğrenme sırasındaki öğretimsel değişkenler (öğretmenler, dönüt, materyal ve gereçler), bağlamsal değişkenler (akran ve çevre), kişisel değişkenler (bilgi inşası, beceri edinimi, öz-düzenleme, etkinliklerin seçimi, çaba ve ısrar)

- Öğrenme sonrasında algılanan öznitelikler, amaçlar, duygular, değerler, ihtiyaçlar ve sosyal destekler önemli motivasyon unsurlarıdır.

Motivasyonla ilişkili bir diğer kuram amaç (amaç yönelimi) kuramıdır. Bu kuram esasen hedefin içsel motivasyon üzerindeki etkisini vurgular. Amaç kuramı kişinin hedefleri, beklentileri, yüklediği anlamlar (attributions), yetenek algıları, motivasyonel yönelimleri, sosyal ve öz karşılaştırmaları ve kazanım davranışları arasında önemli bağlantılar kurar (Anderman & Wolters, 2006; Blumenfeld, 1992; Schunk, 2012). Amaç kuramındaki önemli bir merkezi yapı bireyin kazanım faaliyetlerine katılımının amacını ve odağını ifade eden hedef yönelimidir. Amaç kuramı, amaca yönelik davranışları açıklarken, bazıları doğrudan herhangi bir hedefi içermeyen (örneğin, başkalarıyla karşılaştırmalar) geniş bir değişkenler dizisini de göz önünde bulundurur. Amaç kuramı performans ve öğrenme amaçları arasındaki farka dikkat çeker. Amaç kuramı öğrenme hedeflerinin öğrenme için gerekli olan becerilere ve yeterliklere daha fazla odaklandığını ve öğrencilerin ilerlemeyi algıladıkça, öz yeterliklerinin ve motivasyonlarının arttığını ileri sürer. Bunun aksine performans amaçları ilerlemede ek bir takım bileşenlere (sosyal ve öz karşılaştırma gibi) odaklanarak motivasyon artışı sağlamayabilir (Schunk, 2012).

Amaç kuramı sıklıkla amaç belirleme (goal-setting) (Latham & Locke, 1979; Locke & Latham, 1990, 2002) kuramıyla karıştırılmaktadır. Ancak bu iki kuram arasında önemli farklılıklar bulunmaktadır. Amaç belirleme kuramına göre kişinin amaçlarının niteliği (açıklık, zorluk, kabul ya da bağlılık, geri bildirim, görev karmaşıklığı) onların motivasyonlarını ve dolayısıyla performanslarını etkiler. Amaçlar ne kadar belirgin ve zorluk seviyesi ne kadar yüksek ise kişi daha fazla görev performansı sergiler. Bir kişi hedefe bağlı olduğu sürece, onu elde etmek için gerekli becerilere sahipse ve bu hedefle çatışan başka hedefleri olmadığı sürece, amaç zorluğu ve görev performansı arasında olumlu ve doğrusal bir ilişki vardır. Amaç belirleme kuramı başta yönetim olmak üzere sosyal psikoloji, sağlık bilimleri gibi çok farklı disiplinlerden köken almıştır. Amaç kuramı ise eğitim ortamlarında öğrencilerin başarı davranışlarını açıklar ve tahmin eder. Amaç belirleme kuramı amaçların nasıl kurulduğu, değiştiği ve amaç özelliklerinin (zorluk, açıklık gibi) davranışları (çaba seviyesini) nasıl arttırdığı ve yönlendirdiği üzerine daha çok odaklanmaktadır. Bundan farklı olarak amaç kuramı bireyin başarı etkinliklerindeki uğraşı üzerine odaklanmaktadır. Amaç belirleme kuramı amaç kuramıyla

karşılaştırıldığında davranış üzerinde daha sınırlı bir değişken dizisinin etkisini inceler. Bunun aksine amaç kuramı doğrudan hedefle ilişkili olmayan bazı değişkenleri de içerecek şekilde daha geniş bir yelpazedeki etkileri inceler.

Bir diğer kuram olan başarı motivasyon kuramı çaba gerektiren faaliyetlerde yetkin olmaya çalışmayı ifade eder. Bu kuram ustalık hedefleri ve performans hedefleri arasında ayrım yapmaktadır (Ames & Archer, 1988). Ustalık hedefleri bir kişinin yetkinliklerini geliştirmesine odaklanırken, performans hedefleri kişinin yetkinliklerini ve üstünlüklerini başkalarından daha iyi bir performans göstererek sergilemesine odaklanmaktadır. Bu aynı zamanda görevlere katılım biçimi açısından görev ve ego odaklı şeklinde ayrıştırılabilir. Bu iki katılım biçimi yetenek ve çaba hakkında farklı inançları yansıtır (Nicholls, 1984). Görev yönelimli öğrenciler öğrenmeyi amaç olarak görür ve görev gereksinimlerine daha çok odaklanır. Bu öğrenciler daha fazla çabanın yeteneği arttıracağını düşünür. Öğrencilerin mevcut performansı önceki performansa göre bir gelişme olarak görüldüğünde, yeterlik duyguları ortaya çıkar. Öğrenme, yetkinliği (ustalığı) geliştirme için en önemli amaçtır. Yetkinliğini geliştirme ve gösterme sadece kişinin kendisine bağlıdır. Buna karşın ego yönelimli katılımın özünde bir tür öz-yargılama süreci yatmaktadır. Çabaların rolü sınırlıdır; çaba, yalnızca yeteneğin belirlediği limite göre performansı arttırabilir. Ego yönelimli öğrenciler, beceriksiz görünmekten kaçınmak ister. Öğrenmenin tek başına bir önemi yoktur. Öğrenme ve görev başarımı sadece başkalarından aşağı ya da yukarı görünmek için bir araçtır.

Alanyazında motivasyonun duyuşsal etkilerine odaklanan kuramlar da bulunmaktadır. Csikszentmihalyi'nin (1990) ileri sürdüğü akış kuramı bir etkinlikten alınan yüksek zevk ve haz ile tam olarak odaklanma hissini ilişkilendirmektedir. Aslında akış kuramı içsel motivasyonun nasıl sağlandığına bir cevap niteliğindedir (Tüzün, 2004). Csikszentmihalyi (1990) kaygı ya da can sıkıntısı olmaksızın, akışın görev zorlukları ve kişinin beceri düzeyleriyle örtüştüğünde akış deneyiminin gerçekleşeceğini ileri sürmektedir. Csikszentmihalyi (1975) önceleri akış için kontrol, dikkat, merak ve içsel ilgi olmak üzere 4 temel bileşeni adreslese de daha sonra akış deneyiminin gerçekleşmesi için 8 şartın oluşması gerektiğini belirtmiştir (Csikszentmihalyi, 1993). Bunlar;

- Görevin zorluklarıyla bireyin yetkinlikleri arasında bir dengenin olması,

- Eylem ile farkındalığın birleşmesi,
- Açık amaçlar ve eylemlerin sonuçları hakkında bir dönütün sağlanması,
- Üzerinde çalışılan göreve yoğunlaşılması,
- Eylemler üzerinde kontrol hissinin oluşması,
- Öz-bilincin azalması,
- Zaman algısının bozulması (zamanın akıp gitmesi),
- İçinde kendi amacını taşıyan (autotelic) bir deneyimin oluşması. Bu şart kendisinden önceki 7 boyutun bilinçte oluşması durumunda, üstlenilen görevin kendi başına yapılmaya değer olmasını beraberinde getirir.

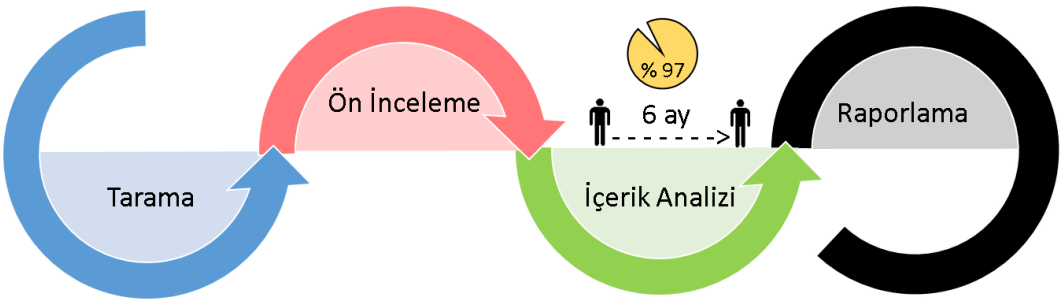
Öğretim ortamlarının tasarımı için tanınmış ve yaygın olarak uygulanan motivasyon tasarım modellerinden birisi de Keller (1983, 1987a, 1987b) tarafından geliştirilen ARCS modelidir. Sade ve güçlü bir model olan ARCS modeli, temellerini bir dizi motivasyon kuram ve kavramlarından alsa da bunlardan en göze çarpanı beklenti-değer kuramıdır (Keller, 1983; Porter & Lawler, 1968; Vroom, 1964). Beklenti-değer kuramında çaba temel ölçülebilir motivasyonel çıktı olarak tanımlanır. Çabanın ortaya çıkması için gerekli iki ön koşul bulunmaktadır. Bunlar kişinin göreve değer vermesi ve kişinin görevde başarılı olabileceğine inanmasıdır. Bu nedenle bir öğretimsel durumda öğrenme görevinin öğrenci için ilgi çekici ve anlamlı ve aynı zamanda öğrenme hedeflerinin başarılı bir şekilde gerçekleştirilmesi için olumlu beklentileri destekleyecek bir yolla sunulması gerekmektedir. ARCS modeli öğrenmeyi motive etmek için 4 temel strateji bileşeni tanımlamaktadır:

- Dikkat (Attention): Dikkat aşamasında öğrencinin derse karşı merakını uyandırıp ilgisi çekilir ve bu ilgi dersin sonuna kadar sürdürülür.
- Uygunluk ya da alaka (Relevance): Uygunluk aşamasında öğrenciye konunun kendi kişisel ihtiyaç ve amaçlarına uygunluğu fark ettirilir.
- Güven (Confidence): Güven aşamasında öğrencilerin kişisel çaba ve kontrol ile başarının yakalanabileceğini fark etmeleri sağlanır.
- Tatmin (Satisfaction): Tatmin aşamasında ise öğrencilerin başarıları çeşitli pekiştireçlerle ödüllendirilerek öğrencilerin içsel tatmin duymaları sağlanır.

Öğrenci motivasyonu neyin nasıl öğrenildiğini etkileyebilir. Buna karşılık, öğrenciler daha becerikli olduklarını öğrendikçe ya da buna inandıkça, öğrenmeye devam etmek için motive olur. Özetle, motivasyon öğrencilerin öğrenmeyi sağlayan etkinliklerle meşgul olmasını sağlamaktadır. Eğitmenler öğrenme için motivasyonun önemini anlamalı ve öğrenme süreci boyunca öğrenci motivasyonunu geliştirmeye çabalamalıdır. Motivasyon doğrudan ölçülemeyen, daha ziyade sözel anlatımlar, görev seçimleri ve hedefe yönelik faaliyetler gibi davranışsal göstergelerden çıkarım yapılarak değerlendirilen psikolojik bir yapıdır. Motivasyon insanların niçin yaptığı gibi davranışlarının nedenlerini anlamamıza yardımcı olan açıklayıcı bir kavramdır. Bu tez çalışmasında yapılan öğretim uygulamalarının ve kullanılan öğretimsel materyallerin öğrenci motivasyonunu olumlu yönde etkilemesi amacıyla Keller tarafından geliştirilen ARCS modeli temel alınmıştır.

Bilgisayımsal Düşünme ile İlgili Araştırmalar

Alanyazın inceleme süreci Şekil-2'de betimlenmiştir. Alanyazın incelemesi sırasıyla tarama, ön-inceleme, içerik analizi (tematik gruplama), sonuçların sayısallaştırılması ve raporlama aşamalarından oluşmaktadır. Tarama aşamasında öncelikle akademik veri tabanı seçimi ve arama kriterleri belirlenmiştir.



Şekil 2. Bilgisayımsal düşünme ile ilgili alanyazın inceleme süreci

Bu çalışmada konu alanına giren makalelere erişmek için ISI Web of Science (WoS) veri tabanı seçilmiştir. WoS birincil kaynakları saptamak için araştırmacılar tarafından en yaygın olarak kullanılan genel referans araçları arasındadır (Fraenkel & Wallen, 2006). Konu alanıyla ilgili nitelikli ve güncel kaynaklara erişmek amacıyla "computational thinking" anahtar kelimelerini kullanarak 2006-2017 yılları arasında Sosyal Bilimler Atıf İndeksi (SSCI) tarafından indekslenen hakemli dergilerde yayımlanan ve Eğitim-Eğitsel Araştırmalar kategorisine giren makalelere ulaşılmıştır. Tarama sonrası ulaşılan 38 makalenin tamamına tam metin olarak

erişim sağlanmıştır. Ön İnceleme aşamasında tüm çalışmalar gözden geçirilmiştir. Buna göre 2 derleme, 1 alanyazın incelemesi, 3 kuramsal perspektif/çerçeve çalışması, 2 öğretim modeli/müfredat geliştirme çalışması olmak üzere toplam 8 makale alanyazın taramasından çıkarılmıştır. Böylece geriye kalan 30 makale üzerinde içerik analizi gerçekleştirilmiştir. Alanyazın incelemesi için kullanılan akademik çalışma dâhil etme ve dışlama kriterleri Tablo 5'te sunulmuştur.

Tablo 5

Akademik Veri Tabanı Taraması İçin Kullanılan Bilimsel Çalışma Dâhil Etme ve Dışlama Kriterleri

Dâhil etme kriterleri	Dışlama kriterleri
Bilimsel çalışmanın özgün araştırma makalesi olması	Derleme, kitap, kitap bölümü, kitap incelemesi ve kongre bildiri/sözlü sunum çalışmaları
Makalelere tam erişim	Makaleye kısıtlı erişim
Makale dilinin İngilizce olması	Makalenin dilinin İngilizceden farklı olması
Makalenin konu olarak eğitim ve eğitimle ilgili araştırma kategorisine girmesi	Konusunun doğrudan eğitim içerikli olmaması
SSCI'de taranması	SCI, SCI-e, A&HCI, CPCI-S, CPCI-SSH, BKCI-S, BKCI-SSH, ESCI'de taranması
2006-2017 arasında yayımlanması	2006 yılından önce ya da 2017 yılından sonra yayımlanması

Çalışmalar yayım yılı, yayımlandığı dergi ve ülke, araştırma tipi (kesitsel ya da boylamsal), araştırma yöntemi (nicel, nitel ya da karma), araştırma deseni, öğrenme bağlamı (formal ya da informal), akademik disiplin, kuramsal perspektif (öğrenme kuramı/modeli), öğretim yöntemi/tekniği, araştırma süresi, katılımcı düzeyi, örneklem boyutu, örneklem seçim yöntemi, incelenen değişkenler, veri toplama aracı/yöntemi, veri analiz tekniği, öğretim veriliş yöntemi (yüz yüze, uzaktan eğitim ya da harmanlanmış), BD bağlamı (BD kavram, pratik ve perspektifleri), katılımcıların BD deneyimi, BD geliştirme stratejisi ya da etkinliği, ortam özellikleri (2B ya da 3B), BD araç arayüzü (görsel ya da sözdizimsel), programlama tipi (blok temelli, ajan temelli vs.), kullanılan BİT donanımları, yazılım araçları, katılımcıların çalışma biçimi (bireysel, işbirlikli ya da eşli) ve değerlendirilme yaklaşımı (özetleyici ya da biçimlendirici) açısından incelenmiştir. Toplam 28 kategorideki tematik analiz aynı araştırmacı tarafından altı ay arayla iki kez gerçekleştirilmiştir. İki inceleme arasındaki uyumun %97 olduğu saptanmıştır. Yıllara göre yayımlanan makale sayıları incelendiğinde BD konusunda son yıllarda artan bir araştırma eğiliminin

olduğu anlaşılmaktadır. Özellikle 2015-2016 yılları arasında BD araştırmalarının sayısında iki kat artış olduğu göze çarpmaktadır (Tablo 6). BD'nin popülerleşmeye başladığı 2006 yılı ile 2012 yılları arasının ise BD'nin araştırmacılar tarafından yeniden fark edildiği bir kuluçka periyodu olduğu söylenebilir.

Tablo 6

Yıllara Göre Yayımlanan Makale Sayıları

Yayın Yılı	Frekans(f)	Yüzde (%)
2013	5	16,67
2014	4	13,33
2015	4	13,33
2016	8	26,67
2017	9	30,00
Toplam	30	100

Yayınların dergilere göre dağılımı incelendiğinde BD ile ilgili en fazla yayın yapılan dergilerin sırasıyla "Computers & Education", "Journal of Educational Computing Research" ve "Journal of Science Education and Technology" dergilerinin olduğu görülmektedir (Tablo 7).

Tablo 7

Makalelerin Dergilere Göre Dağılımı

Dergi Adı	Frekans(f)	Yüzde (%)
Computers & Education	7	23,33
Journal of Educational Computing Research	5	16,67
Journal of Science Education and Technology	5	16,67
Educational Technology & Society	2	6,67
Asia Pacific Education Researcher	1	3,33
Australasian Journal of Educational Technology	1	3,33
Cultura Y Educacion	1	3,33
Educational Assessment Evaluation and Accountability	1	3,33
Educational Technology Research and Development (ETR&D)	1	3,33
Eurasia Journal of Mathematics Science and Technology Education	1	3,33
Instructional Science	1	3,33
Interactive Learning Environments	1	3,33
Journal of Computer Assisted Learning	1	3,33
Journal of the Learning Sciences	1	3,33
Urban Education	1	3,33
Toplam	30	100

Ülkelere göre yayımlanan makale sayısı incelendiğinde BD ile ilgili araştırma konusunda ABD'nin oldukça öne çıktığı görülmektedir. Onu Çin Halk Cumhuriyeti izlemektedir (Tablo 8).

Tablo 8

Ülkelere Göre Yayımlanan Makale Sayısı

Ülke/Bölge	Frekans(f)	Yüzde (%)
ABD	15	42,86
Çin Halk Cumhuriyeti	5	14,29
Yunanistan	3	8,57
Singapur	2	5,71
Güney Kore	2	5,71
İspanya	2	5,71
Tayvan	2	5,71
Kanada	1	2,86
G. Kıbrıs	1	2,86
Yeni Zelanda	1	2,86
Türkiye	1	2,86
Toplam	35	100

Yayımlanan makalelerin tamamına yakınının kesitsel (%96,67) olduğu belirlenmiştir (Tablo 9). Kesitsel araştırmaların birisinde eş zamanlı olarak yürütülen birden fazla kesitsel durum karşılaştırılmıştır. Bu alanda özellikle öğrencilerin gelişiminin daha iyi gözlemlenebileceği boylamsal araştırmalara ihtiyaç bulunmaktadır. En azından yapılan öğretimsel müdahalelerin etkililiğinin tek bir kesitsel araştırmadan ziyade birden fazla kesitsel araştırma ile değerlendirilmesi elde edilen bulguların karşılaştırılması ve tutarlıklarının incelenmesine olanak tanıyabilir.

Tablo 9

Araştırma Türüne Göre Yayın İstatistikleri

Araştırma Tipi	Frekans(f)	Yüzde (%)
Kesitsel	29	96,67
Boylamsal	1	3,33
Toplam	30	100

Araştırma yöntemi açısından incelendiğinde araştırmalarda nicel araştırma ve karma araştırma desenlerinin daha sık işe koşulduğu görülmektedir (Tablo 10).

Tablo 10

Makalelerin Araştırma Yöntemine Göre Dağılımı

Araştırma Yöntemi	Frekans(f)	Yüzde (%)
Nicel	15	50,00
Nitel	5	16,67
Karma	10	33,33
Toplam	30	100

Deneysel araştırma ve durum çalışmalarının yoğun olarak kullanılan araştırma desenleri olduğu göze çarpmaktadır (Tablo 11). Özellikle karma araştırma desenlerinde deneysel ve durum çalışması araştırma desenlerinin sıkça birlikte kullanıldığı tespit edilmiştir.

Tablo 11

Araştırma Desenlerine Göre Çalışma Sayıları

Araştırma Deseni	Frekans(f)	Yüzde (%)
Deneysel Araştırma	16	43,24
Durum Çalışması	11	29,73
Tasarım Tabanlı Araştırma	3	8,11
Nedensel-Karşılaştırmalı Araştırma	2	5,41
İlişkisel Araştırma	1	2,70
Fenomenografik Araştırma	1	2,70
Öğrenme Analitiği/Veri Madenciliği	1	2,70
Tarama Araştırması	1	2,70
Biçimlendirici Araştırma	1	2,70
Toplam	37	100

Not: Birden fazla araştırma deseninin kullanıldığı karma desenler araştırma deseni frekans toplamını arttırmıştır.

Çalışmalar beklenildiği üzere çoğunlukla formal öğretim düzenlemeleri (%83,88) içerisinde gerçekleştirilmiştir (Tablo12).

Tablo 12

Öğrenme Bağlamına Göre Çalışma Sayıları

Öğrenme Bağlamı	Frekans(f)	Yüzde (%)
Formal	25	83,33
İnformal	2	6,67
Formal/İnformal	2	6,67
Belirtilmemiş	1	3,33
Toplam	30	100

BD etkinlikleri çoğunlukla herhangi bir akademik disipline (bilgisayar bilimleri hariç) bağlı olmaksızın gerçekleştirilmiştir. Diğer taraftan sırasıyla Matematik, STEM, Fen/Doğa bilimleri, Sanat ve Sosyal Bilimler BD'nin en fazla entegre edildiği alanlar olarak dikkat çekmektedir (Tablo 13).

Tablo 13

Çalışmaların Farklı Akademik Disiplinlere Göre Dağılımı

Akademik Disiplin Alanı	Frekans(*f*)	Yüzde (%)
Matematik/Geometri	6	15,00
STEM	5	12,50
Fen/Doğa Bilimleri (Fizik, Kimya, Biyoloji vb.)	4	10,00
Sanat	3	7,50
Sosyal Bilimler (Tarih, Coğrafya vb.)	2	5,00
Teknoloji/Mühendislik	2	5,00
İngilizce	1	2,50
Kütüphane/Medya	1	2,50
Geliştirme (Enrichment)	1	2,50
Dil Becerileri (Okuma)	1	2,50
Uygulanamaz	14	35,00
Toplam	40	100

Not: Bazı çalışmalar birden fazla akademik disiplin çerçevesinde gerçekleştirilmiştir. STEM ayrı bir disiplinler arası alan olarak kabul edilmiştir. Bilgisayar bilimleriyle (programlama gibi) ilgili dersler BD becerilerinin doğal olarak işe koşulduğu alanlar olduğu için tabloya dâhil edilmemiştir.

Hangi teknolojik araç ya da ortam kullanılırsa kullanılsın öğretimsel amaçlı her pratiğin bir öğrenme anlayışı altyapısına sahip olması gerektiği unutulmamalıdır. Yani özünde öğrenme etkinlikleri olan her araştırma, önerilen ya da etkisi araştırılan teknolojinin ve pedagojinin temel kabullerini hangi kuramdan aldığını yani tanımlamalarının ve varsayımlarının hangi kabule dayandığını belirtmek durumundadır. İlginç olarak, kuramsal temel ya da perspektif açısından incelendiğinde çok sayıda çalışmanın belirgin bir öğrenme yaklaşımı (perspektifi), kuramı ya da modeli üzerine kurulu olmadığı tespit edilmiştir. Bununla birlikte kurmacılık (%23,5) ve yapılandırmacılığın (%11,8) çalışmalarda nispeten en sık kullanılan kuramsal temeller olduğu görülmektedir (Tablo 14). Tüm bu sonuçlar çalışmaların daha teknoloji odaklı olduğunu ve bu çalışmalarda öğrenme kuramlarının eğitsel teknolojiye yönelik pedagojiler için bir dayanak olarak sorgulanmadığını göstermektedir.

Tablo 14

Çalışmaların Altında Yatan Kuramsal Temel (Perspektif, Öğrenme Kuramları ve Modelleri) Açısından İncelenmesi

Kuramsal Perspektif/Temel	Frekans(*f*)	Yüzde (%)
Kurmacılık (Constructionism)	8	23,53
Yapılandırmacılık (Constructivism)	4	11,76
Bloom'un Öğrenme Taksonomisi	2	5,88
Fenomenolojik Öğrenme Yaklaşımı	1	2,94
Uygulama (Implementation) Bilimi Perspektifi	1	2,94
Düzeltici (Corrective) Öğretim	1	2,94
Öğrenme İçin Evrensel Tasarım Pedagojisi (Universally-design Pedagogy for Learning)	1	2,94
Teknoloji, Pedagoji ve İçerik Bilgisi (TPACK)	1	2,94
Anlamlı Öğrenme	1	2,94
Açıkça Belirtilmemiş	14	41,18
Toplam	34	100

Çalışmalarda en sık kullanılan öğretim yönteminin problem çözme (görev tamamlama), anlatım ve proje temelli öğretim olduğu belirlenmiştir. Özellikle programlama deneyimi düşük öğrenenlerle gerçekleştirilen çalışmalarda anlatım yönteminin sıklıkla işe koşulduğu saptanmıştır (Tablo 15).

Tablo 15

Çalışmalarda Kullanılan Öğretim Yöntem ya da Tekniklerinin Dağılımı

Öğretim Yöntemi/Tekniği	Frekans(*f*)	Yüzde (%)
Problem Çözme/Görev başarma	14	26,42
Anlatım (Tüm sınıf öğretim)	8	15,09
Proje (Ürün) Tabanlı Öğrenme	7	13,21
İşbirlikli Öğrenme/Takım çalışması	7	13,21
Oyun/Oyun Tasarım tabanlı öğrenme	3	5,66
Tartışma	3	5,66
Akran Desteği	2	3,77
Bire-bir öğrenme desteği (Özel eğitim)	1	1,89
Saha Gözlemleri	1	1,89
Öğrenci Seminerleri	1	1,89
Öğrenme Topluluğu	1	1,89
Uygulanamaz	5	9,43
Toplam	53	100

Not: Bir çalışma içerisinde farklı öğretim yöntem ve teknikleri kullanılabilmektedir.

Yapılan çalışmalar araştırma süresi açısından incelendiğinde çalışmaların büyük bölümünün (%80) 1 dönem ya da daha kısa sürede gerçekleştirildiği görülmektedir (Tablo 16).

Tablo 16

Makalelerin Araştırma Süreleri Açısından İncelenmesi

Araştırma Süresi	Frekans(f)	Yüzde (%)
1 dönemden az (<10 hafta)	11	36,67
1 dönem (10 ile 18 hafta arası)	13	43,33
1 dönem ile 1 yıl arası	1	3,33
1 yıldan uzun	2	6,67
Uygulanamaz	3	10
Toplam	30	100

Çalışmalar katılımcı düzeyi bakımından incelendiğinde sırasıyla lisans (%22,26), ilkokul (%22,58) ve ortaokul (%16,13) düzeyinde katılımcıların çalışmalara nispeten daha sık dâhil edildiği görülmektedir (Tablo 17). Bir çalışmada hem ortaokul hem de lisans öğrencileriyle iki ayrı uygulama gerçekleştirilmiştir. Araştırmaların çoğunlukla lisans düzeyindeki katılımcılarla gerçekleştirilmesinin bir nedeni de bu grubunun araştırmacılar tarafından nispeten kolay erişilebilir olmasıdır. Son dönemde kullanımı nispeten kolay programlama araçlarının yaygınlaşmasıyla birlikte ilkokul ve ortaokul düzeyindeki öğrencileri konu edinen araştırmaların sayısında artış yaşanmıştır. Öğrenim düzeyine göre lise, lisansüstü ve okul öncesi seviyelerindeki çalışmaların nispeten sınırlı sayıda olduğu görülmektedir.

Tablo 17

Makalelerin Katılımcı Düzeyine Göre Dağılımı

Katılımcı Düzeyi	Frekans(f)	Yüzde (%)
Lisans	10	32,26
İlkokul	7	22,58
Ortaokul	5	16,13
Lise	4	12,90
Hizmet içi Öğretmen/Yöneticiler	2	6,45
Okul Öncesi	1	3,23
Lisans Sonrası Eğitim (Pedagojik Formasyon)	1	3,23
Özel Eğitim	1	3,23
Toplam	31	100

Çalışmalardaki örneklem boyutlarının nispeten küçük (50'den az) ya da orta ölçekte (50 ile 100 arasında) olduğu görülmektedir (Tablo 18). Araştırmaların daha ziyade deneysel nitelikli ya da incelenen olgunun otantik koşullarda daha derinlemesine sorgulandığı durum çalışmalarına dayalı olmasının örneklem büyüklüğü dağılımını etkilediği düşünülmektedir.

Tablo 18

Makalelerin Örneklem Boyutu Açısından İncelenmesi

Örneklem Boyutu	Frekans(f)	Yüzde (%)
50'den az	10	33,33
50 ile 100 arası	8	26,67
100 ile 200 arası	8	26,67
200'den büyük	4	13,33
Toplam	30	100

Katılımcıların seçimi bilimsel araştırma sürecinin en önemli aşamalarından birisidir. Çalışmalar örneklem seçim yöntemi açısından incelendiğinde seçkisiz olmayan örnekleme yöntemlerinin (elverişli ve amaçlı gibi) daha yoğun (%76,67) kullanıldığı görülmektedir (Tablo 19). Eğitim araştırmalarına konu edinilen katılımcı grubu genellikle belirli özelliklere sahip bireylerden oluşmaktadır. Bununla birlikte araştırmacılar genellikle sonuçlarını genelleştirmek istedikleri popülasyona nadiren erişmektedir. Öte yandan, örneklem boyutu ve örneklem seçim yöntemi sonuçların genellenebilirliği yani dış geçerliği üzerinde önemli bir rol oynamaktadır. Bu nedenle alanyazında seçkisiz seçim yöntemlerini işe koşan çalışmalara ihtiyaç bulunmaktadır.

Tablo 19

Makalelerin Örneklem Seçim Yöntemine Göre Dağılımı

Örneklem Seçim Yöntemi	Frekans(f)	Yüzde (%)
Seçkisiz	7	23,33
Seçkisiz olmayan	23	76,67
Toplam	30	100

Çalışmalarda incelenen değişkenler oldukça çeşitlilik göstermektedir. Tablo 20 incelendiğinde çalışmalarda en sık incelenen değişkenlerin sırasıyla BD ya da programlama başarısı ya da yetkinliği, öz yeterlik, ürün niteliği, motivasyon, katılım,

eğlenebilirlik, tutum, akıl yürütme becerisi, BD öğretimi/entegrasyonu olduğu görülmektedir.

Tablo 20

Makalelerde İncelenen Değişkenlerin Dağılımı

İncelenen Değişkenler	Frekans(f)	Yüzde (%)
Başarı (BD/Programlama Bilgi, Beceri ve Stratejileri)	17	23,29
Öz yeterlik	5	6,85
Ürün (Program gibi) Niteliği	4	5,48
Motivasyon	3	4,11
Katılım (Bağlılık)	3	4,11
Eğlenebilirlik	3	4,11
Tutum	3	4,11
Akıl Yürütme Becerisi (Neden-Sonuç İlişkisi)	3	4,11
BD Öğretimi/Entegrasyon Pratikleri	3	4,11
BD'ye İlişkin Öğretmen/Öğrenci Görüşleri	2	2,74
STEM Kariyer Tercihi/Tutumu	2	2,74
Diğer Disiplinlere İlişkin Öğrenme Çıktıları	2	2,74
Öğrenci Etkinliği (Aktif Öğrenme)	2	2,74
BD'ye Karşı İlgi	2	2,74
Empati	1	1,37
Duygusal Öz-farkındalık	1	1,37
Mantıksal Düşünme	1	1,37
Kişisel Beklentiler	1	1,37
Öğrenci Profil Dağılımı	1	1,37
Öğrenme Kaygısı	1	1,37
BİT Okuryazarlığı	1	1,37
Bilgisayımsal Okuryazarlık	1	1,37
Örüntü Bulma ve Sınıflama Becerisi	1	1,37
Karmaşık Sistemleri Düşünme Becerisi	1	1,37
Ortam Etkililiği	1	1,37
Olgusal Etkileşim Tasarlama	1	1,37
Genel Bilişsel Düşünme/Öğrenme Türleri	1	1,37
Öğrenci İşbirliği	1	1,37
Algılanan Yarar	1	1,37
Öğrenci Destek ve İhtiyaçları	1	1,37
Öğrenci Memnuniyeti	1	1,37
Problem Çözme Becerisi	1	1,37
BD Kalıpları (Kavram, Pratik ve Perspektifler)	1	1,37
Toplam	73	100

Çalışmalarda en sık kullanılan veri toplama araçlarının test ve ölçek/anket olduğu saptanmıştır (Tablo 21).

Tablo 21

Makalelerin Veri Toplama Araçları Yönünden İncelenmesi

Veri Toplama Araçları/Yöntemleri	Frekans(f)	Yüzde (%)
Test	14	21,21
Ölçek/Anket	14	21,21
Görüşme Formu	9	13,64
Ekran/Görüntü/Ses ve Sohbet Kaydı	5	7,58
Öğrenci Ürünleri (Taslak, Materyal ve Ürün Tasarımları)	5	7,58
Alan Notları	4	6,06
Log (Sistem) Kayıtları	4	6,06
Rubrik/Puanlama Dereceleri	4	6,06
Gözlem	4	6,06
Yansıma raporları	2	3,03
Ulusal İstatistikler	1	1,52
Toplam	66	100

Çalışmalarda en sık kullanılan veri analiz tekniklerinin sırasıyla ortalama puan fark analizi, betimsel istatistik ve içerik analizi olduğu belirlenmiştir. Karmaşık veri analizi tekniklerinin nadir olarak işe koşulduğu saptanmıştır (Tablo 22).

Tablo 22

Makalelerde Kullanılan Veri Analiz Teknikleri

Veri Analiz Teknikleri	Frekans(f)	Yüzde (%)
Ortalama puan fark analizleri (T testleri, ANOVA, MANOVA, Mann-Whitney U, Wilcoxon gibi)	18	31,58
Betimsel analiz (Frekans, Yüzde, Ortalama Puan gibi)	14	24,56
İçerik Analizi/Temellendirilmiş Kuram	10	17,54
Kümeleme Analizi	3	5,26
Faktör Analizi	3	5,26
Regresyon Analizleri	2	3,51
Madde Analizleri (Ayırıcılık ve Güçlük İndeksi)	2	3,51
Güvenirlik Analizleri	2	3,51
Bilişsel Görev Analizi	1	1,75
Korelasyon Analizi	1	1,75
Uzman Görüşü	1	1,75
Toplam	57	100

Çalışmalar beklenildiği üzere çoğunlukla yüz yüze öğretim (%70) aracılığıyla gerçekleştirilmiştir (Tablo 23).

Tablo 23

Makalelerin Öğretim Verilme Yöntemi Açısından İncelenmesi

Öğretim Veriliş Yöntemi	Frekans(f)	Yüzde (%)
Yüz Yüze Eğitim	21	70,00
Uzaktan Eğitim	1	3,33
Harmanlanmış Öğretim	4	13,33
Uygulanamaz	4	13,33
Toplam	30	100

Çalışmalar Brennan ve Resnick (2012) tarafından önerilen BD çerçevesi (BD kavram, pratik ve perspektifleri) açısından da analiz edilmiştir. Buna göre çalışmaların BD perspektiflerinden (%8,16) ziyade BD kavram (%34,69) ve pratikleri (%48,98) üzerine yoğunlaştığı görülmektedir (Tablo 24).

Tablo 24

Makalelerin Bilgisayımsal Düşünme Bağlamı Açısından İncelenmesi

Bilgisayımsal Düşünme (BD) Bağlamı	Frekans(f)	Yüzde (%)
BD Kavramları	17	34,69
BD Pratikleri	24	48,98
BD Perspektifleri	4	8,16
Uygulanamaz	4	8,16
Toplam	49	100

Çalışmaların genellikle BD ve programlama açısından deneyimsiz katılımcılarla gerçekleştirildiği belirlenmiştir (Tablo 25). BD süreçlerinde kullandıkları stratejilerin açıklığa kavuşturulması adına uzman programcıları konu edinen daha çok çalışmaya ihtiyaç bulunmaktadır.

Tablo 25

Katılımcıların Programlama Deneyimleri

Katılımcıların BD/Programlama Deneyimi	Frekans(f)	Yüzde (%)
Tecrübesiz/Yeni başlayan	17	56,67
Deneyimli	5	16,67
Karma (Deneyimli ve Deneyimsiz)	4	13,33
Belirtilmemiş/Uygulanamaz	4	13,33
Toplam	30	100

Programlama, BD becerilerinin otantik olarak işe koşulduğu bir etkinliktir. Alanyazında BD becerisinin geliştirilmesi için kullanılan baskın stratejinin programlama (%48,78) olduğu görülmektedir. Bununla birlikte BD için kullanılan çok geniş yelpazede öğretimsel etkinliklerin olduğu dikkati çekmektedir. Bunların içinde oyun tasarımı, modelleme/simülasyon/animasyon oluşturma, dijital hikaye anlatımı, bilgisayarsız etkinlikler, karmaşık sistem düşünme/tasarlama, örüntü tanıma ve sınıflama, paket program etkinlikleri yer almaktadır (Tablo 26). Öte yandan programlama başta oyun tasarımı ya da programlama ve dijital hikâye anlatımı olmak üzere bazı BD etkinliklerinde kendisine yer bulmaktadır.

Tablo 26

Çalışmalarda Bilgisayımsal Düşünmenin Geliştirilmesi İçin Kullanılan Teknik ve Stratejilerin İncelenmesi

BD Geliştirme Teknikleri/Stratejileri	Frekans(f)	Yüzde (%)
Programlama (bilgisayar ve robot programlama)	20	48,78
Oyun tasarımı ve programlama	4	9,76
Modelleme, simülasyon, animasyon ve çoklu ortam içerikleri	4	9,76
Dijital hikâye anlatımı	3	7,32
Bilgisayarsız etkinlikler (kalem-kâğıt etkinlikler))	2	4,88
Karmaşık sistem düşünme ve tasarlama	2	4,88
Örüntü tanıma ve sınıflama	1	2,44
Paket program uygulamaları	1	2,44
Uygulanamaz	4	9,76
Toplam	41	100

Çalışmalarda kullanılan teknolojik ortam modaliteleri incelendiğinde 2B ortamların (%66,67) daha yoğun olarak kullanıldığı görülmektedir (Tablo 27).

Tablo 27

Makalelerde Kullanılan Teknolojik Ortam Modalitesi

BD Ortam Özellikleri	Frekans(f)	Yüzde (%)
2B	20	66,67
3B	2	6,67
Karma (2B ve 3B)	4	13,33
Uygulanamaz	4	13,33
Toplam	30	100

BD etkinlikleri için kullanılan araç arayüzlerinin çoğunlukla görsel/grafiksel olduğu görülmektedir (Tablo 28).

Tablo 28

Bilgisayımsal Düşünme İçin Kullanılan Teknolojik Ortam Arayüzü

BD/Programlama İçin Kullanılan Ortam Ara yüzü	Frekans(f)	Yüzde (%)
Görsel (Grafiksel)	17	56,67
Metin Tabanlı	3	10,00
Karma (Görsel/Metin Tabanlı)	2	6,67
Uygulanamaz/Açıkça Belirtilmemiş	8	26,67
Toplam	30	100

BD ya da programlama etkinlikleri incelendiğinde bu etkinliklerde genellikle blok tabanlı araçlarının kullanıldığı görülmektedir (Tablo 29). Blok tabanlı araçları, ajan temelli ve sözdizimsel programlamaya dayalı ortamlar takip etmektedir.

Tablo 29

Çalışmalarda Kullanılan Bilgisayımsal Düşünme ya da Programlama Etkinlik Biçimi

BD/Programlama Biçimi	Frekans(f)	Yüzde (%)
Blok Tabanlı	17	48,57
Ajan Temelli	4	11,43
Sözdizimsel	4	11,43
Kalem/Kâğıt Aracılı	2	5,71
Uygulanamaz/Belirtilmemiş	8	22,86
Toplam	35	100

Çalışmalarda kullanılan başlıca BİT donanım araçlarının bilgisayar, eğitsel robot ya da robotik setleri, avuç içi cihazlar (tablet, akıllı telefon ve Kişisel Dijital Asistan-PDA) olduğu belirlenmiştir (Tablo 30).

Tablo 30

Çalışmalarda Kullanılan Araç-Gereç ve BİT Donanımları

Kullanılan araç-gereçler/BİT donanımları	Frekans(f)	Yüzde (%)
Bilgisayar	23	56,10
Eğitsel Robot/Robotik Devre Setleri	7	17,07
Avuç İçi Cihazlar (Tablet, Akıllı Telefon, PDA)	3	7,32
Kalem/Kâğıt	2	4,88
Fiziksel Bloklar (Kod Tahtaları)	1	2,44
Etkileşimli Tahta	1	2,44
Uygulanamaz	4	9,76
Toplam	41	100

Çalışmalarda en yoğun kullanılan yazılım araçlarının sırasıyla Scratch, Alice, e-öğrenme sistemleri, Code.org, C++, AgentSheet/AgentCubes ve Lego robotik yazılımları olduğu belirlenmiştir (Tablo 31).

Tablo 31

Çalışmalarda Kullanılan Yazılım Araçları

Yazılım Araçları/Uygulamaları	Frekans(*f*)	Yüzde (%)
Scratch/Scratch J	10	24,39
Alice	3	7,32
E-Öğrenme Sistemleri (PASS, Moodle gibi)	3	7,32
Code.org	2	4,88
C++	2	4,88
AgentSheets ve AgentCubes	2	4,88
Lego Yazılımları (Mindstorms, WeDo)	2	4,88
Categorizer	1	2,44
VBOT (NetLogo)	1	2,44
E-Toys	1	2,44
E-kitap uygulaması	1	2,44
Multimedya İçerik/Dijital Hikâye araçları (Gloglster, Voicethread, Storybird)	1	2,44
Paket Programlar	1	2,44
Choregraphe Robot Programlama ve Simülasyon Aracı	1	2,44
Second Life	1	2,44
Arduino IDE Software	1	2,44
MatLab	1	2,44
CHERP	1	2,44
Logo	1	2,44
IPRO	1	2,44
Uygulanamaz	4	9,76
Toplam	41	100

Katılımcıların BD ya da programlama etkinliklerini genellikle tekli çalışma (%46,67) biçiminde gerçekleştirdiği görülmektedir (Tablo 32). Bu durumun programlama öğretiminin geleneksel yapısından ve benimsenen birey başına bir cihaz paradigmasından kaynaklandığı düşünülmektedir. Ayrıca BD etkinliklerinin genellikle ekran bağımlı olarak sürdürülmesi katılımcıların birbiriyle etkileşimini azaltmakta ve ortak çalışma alanlarını sınırlamaktadır. Araştırmalar işbirlikli çalışmaların öğrenme çıktıları açısından daha verimli sonuçlar ürettiğini göstermektedir. Öte yandan BD süreçlerine yönelik işbirlikli öğrenci çalışmalarının

nispeten az sayıda olduğu görülmektedir. Bu yüzden alanyazında işbirlikli ve eşli programlama etkinlikleri üzerine daha çok çalışmaya ihtiyaç bulunmaktadır.

Tablo 32

Makalelerin Katılımcı Çalışma Biçimi Açısından İncelenmesi

Katılımcıların Çalışma Biçimi	Frekans(f)	Yüzde (%)
Tekli (Eşsiz)	14	46,67
İşbirlikli	5	16,67
Eşli	1	3,33
Karma (Tekli-Eşli-İşbirlikli)	2	6,67
Uygulanamaz	8	26,67
Toplam	30	100

Çalışmalarda BD becerisinin gelişimi için seçilen değerlendirme yaklaşımlarının genellikle özetleyici (sonuç odaklı) olduğu görülmektedir. Bu anlamda BD gelişiminin izlenmesi ve bu süreçte öne çıkan bilişsel ve duyuşsal faktörlerin derinlemesine incelenmesi için süreç odaklı değerlendirme yaklaşımlarına ihtiyaç bulunduğu söylenebilir (Tablo 33).

Tablo 33

Makalelerin Bilgisayımsal Düşünme Değerlendirme Yaklaşımları Açısından İncelenmesi

Değerlendirme Yaklaşımı	Frekans(f)	Yüzde (%)
Özetleyici (Sonuç Odaklı)	16	53,33
Biçimlendirici (Süreç Odaklı)	1	3,33
Karma	7	23,33
Uygulanamaz	6	20,00
Toplam	30	100

Sonuç olarak son yıllarda giderek artan bir araştırma ilgisi bulunan BD ile ilgili ülkemizde yapılan makale sayısı kısıtlı sayıdadır. BD üzerine yapılan çalışmaların büyük bölümü tek bir kesitsel incelemeye dayalıdır. Bu durum öğretimsel müdahalelerin etkililiğine ve bunun sonucunda elde edilen eğitsel çıktıların tutarlığına ilişkin soru işaretleri yaratmaktadır. Çalışmalarda BD'ye yönelik olarak ele alınan değişkenler ve kullanılan veri analiz teknikleri oldukça farklılık göstermektedir. Bununla birlikte çalışmalarda ele alınan değişkenler incelendiğinde soyutlama ve problem çözme üzerine yapılan çalışmaların oldukça nadir olduğu görülmektedir. BD ile ilgili erken dönem araştırmacıları bilgisayımsal görevlerden

edinilen problem çözme becerilerinin gündelik ve diğer alanlardaki genel problem çözme yaklaşımları üzerine etkilerini araştıran yeterli sayıda empirik çalışma olmadığına dikkati çekmiştir (Mayer vd., 1986; Pea & Kurland, 1984). Bu çalışmalardaki eleştiriler halen güncelliğini korumaktadır. Katılımcıların genellikle lisans, ilk ve ortaokul öğrencilerinden seçildiği, lise öğrencilerine yönelik çalışmaların yetersiz olduğu görülmektedir. BD için yapılan çalışmalar daha ziyade BD kavram ve pratiklerine odaklanırken BD perspektiflerine yönelik sınırlı sayıda çalışma bulunmaktadır. Öğretimsel uygulamaların etkililiğinin değerlendirilmesinde ise genellikle sonuç ya da ürün odaklı bir yaklaşımın tercih edildiği görülmektedir.

Mevcut tez çalışması BD becerilerinin sergilenmesi için işe koşulan nesneye yönelik ve eğitsel robot programlama etkinliklerinin programlama başarısı, soyutlama becerisi, problem çözme yaklaşımı ve motivasyon üzerindeki etkisini incelemektedir. Bu çalışmanın katılımcıları alanyazındaki diğer çalışmalardan farklı olarak düşük akademik başarıya sahip mesleki ortaöğretim öğrencileridir. Bilimsel araştırmalarda benimsenen araştırma yöntemi genellikle elde edilen veri kümesinin de niteliğini belirler. Bu çalışmada sadece süreç sonunda değil, BD etkinlikleri esnasında deneyimlenen bilişsel ve duyuşsal süreçler incelenmiştir. Ayrıca gerçekleştirilen deneysel uygulamanın etkililiği ve sonuçların tutarlılığı iki kesitsel çalışmayla (durum-1 ve durum-2) değerlendirilmiştir.

Bölüm 3
Yöntem

Araştırma desenleri bilimsel çalışmalarda veri toplama, toplanan verileri analiz etme, yorumlama ve raporlama biçimleridir. Bu desenler araştırma yapmak için farklı çerçeveler sunar. Araştırma desenleri araştırmacıların çalışmaları süresince gerekli yöntemsel kararları almalarına ve çalışma sonunda çıkarımlarına temel oluşturacak mantığı belirlemelerine yardımcı olması bakımından oldukça yararlıdır (Creswell & Clark, 2007). Buna ek olarak seçilecek araştırma deseni aynı zamanda çalışmanın felsefi ve kuramsal temellerini yansıtan problem durumuna ve araştırma sorularına uygun olmalıdır.

Bu çalışmada araştırma yöntemi olarak karma araştırma deseni kullanılmıştır. Bir araştırma yöntemi olarak karma araştırma desenleri hem nicel hem de nitel verileri tek ya da bir dizi çalışmada toplamaya, analiz etmeye ve birleştirmeye odaklanır. Karma araştırma desenlerinin temel dayanağı nicel ve nitel yaklaşımların birlikte kullanılmasının bu yaklaşımlardan herhangi biriyle karşılaştırıldığında araştırma problemlerinin daha iyi incelenmesini sağlayacağı düşüncesidir. Karma desenler nitel ve nicel aşamaların araştırma sürecinin başında belirlendiği sabit (fixed) karma desenler ve süreç içerisinde gelişen durumlardan dolayı nicel ya da nitel aşamanın halihâzırdaki araştırmaya sonradan dâhil edilmesiyle oluşturulan gelişen (emergent) karma desenler olmak üzere ikiye ayrılabilir. Bu çalışma nitel ve nicel yöntemler çalışmanın başında tanımlandığı ve yöntemsel prosedürler planlandığı akışta sürdürüldüğü için sabit karma desenler sınıfına girmektedir.

Creswell ve Clark (2007) çalışmalarda karma araştırma yönteminin işe koşulma nedeninin araştırmacılar tarafından açıkça belirtilmesinin önemli ve gerekli olduğunun altını çizmektedir. Bu çalışmada ilgilenilen sorgulama alanına dair daha kapsamlı açıklamalar getirmek, psikometrik yapılara dair bir ölçme işleminin yanında katılımcıların deneyimlediği süreci derinlemesine incelemek, nicel ve nitel verileri kullanarak bulguların bütünlüğünü sağlamak ve veri çeşitlemesi yoluyla daha yüksek geçerlik elde etmek amacıyla karma araştırma yönteminin kullanılması uygun görülmüştür. Buna ek olarak, nicel araştırma yoluyla değişkenler arasındaki ilişkileri ortaya çıkarırken, nitel araştırma yoluyla katılımcılardaki anlamları ortaya çıkarmak, böylece iki farklı mantık üzerine kurulu olarak perspektif çeşitliliğini

sağlamak da mevcut çalışmada karma araştırmanın kullanılmasının altındaki bir diğer önemli gerekçedir.

Genel olarak karma araştırma tipolojisinin seçilmesi için karar verilmesi gereken dört temel unsur bulunmaktadır (Bryman, 2006; Creswell & Clark, 2007). Bunlar;

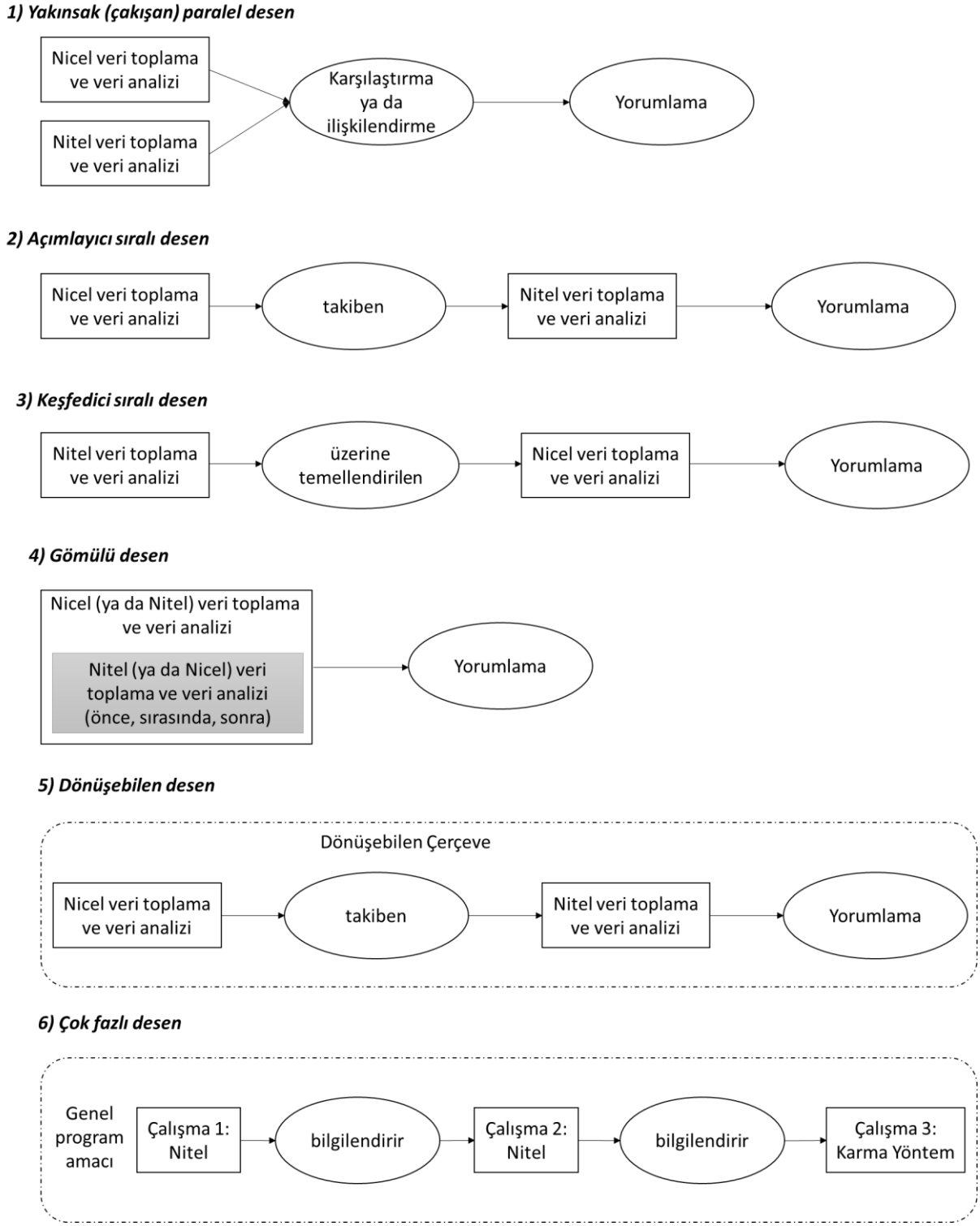
- Nicel ve nitel aşamalar arasındaki etkileşim düzeyi (bağımsız, etkileşimli)
- Aşamalar arası görece öncelik (nicel öncelikli, nitel öncelikli, eşit öncelikli)
- Aşamaların zamanlaması (kesişen, ardışık, çok fazlı kombinasyon)
- Aşamaların uygulanmasında izlenecek prosedür (iki veri setinin birleştirilmesi, bir veri setinin analizinin diğer bir veri setinin toplanması için kullanılması, bir veri setinin daha geniş bir desen ya da prosedür içerisine gömülmesi, iki veri setini birbirine bağlamak için kuramsal bir çerçevenin kullanılması)

Etkileşim düzeyi; iki veri toplama fazının birbiriyle ne ölçüde bağlantılı olduğuyla ilgilidir. Bağımsız düzeyde nicel ve nitel araştırma soruları, veri toplama ve analiz süreçleri birbirinden ayrıktır. Araştırmacı yalnızca çalışmanın sonunda yapılan kapsamlı yorumlama sırasında sonuçları ortaya koyarken nitel ve nicel araştırma fazlarını birleştirir. Etkileşimli düzeyde ise iki veri seti sonuçların yorumlanmasından önce birleştirilir. Bu etkileşim araştırma süresince farklı noktalarda farklı yollarla meydana gelebilir. Örneğin bir araştırma türünün (nitel ya da nicel) tasarımı ve yürütülmesi diğer bir araştırma türünden elde edilen verilerin analiz sonuçlarına bağlı olarak gerçekleşebilir. Bir araştırma türünden elde edilen veriler diğer bir araştırma türüne dönüştürülüp, veriler birlikte analiz edilebilir. Ya da bir araştırma türü diğer araştırma türünü temel alan bir çerçeve içerisinde gerçekleştirilebilir. Öncelik düzeyi; araştırma deseni içerisinde nicel ve nitel aşamalara görece ne kadar önem ve öncelik verildiğiyle ilgilidir. Araştırmacı nicel öncelikli, nitel öncelikli ya da her iki araştırma yaklaşımının eşit öneme ve önceliğe sahip olduğu bir yaklaşım benimseyebilir. Araştırmacı çalışmadaki nicel ve nitel aşamaların zamanlaması konusunda da karar vermelidir. Zamanlama, sıklıkla veri kümelerinin toplandığı zamanla ilgili bir şey olarak anlaşılsa da aynı zamanda iki veri setinin sonuçlarının araştırma içerisinde nasıl bir sıralamayla kullanacağıyla da ilişkilidir. Kesişen ya da çakışık zamanlamada araştırmacı tek fazlı bir araştırma

içerisinde hem nicel hem de nitel verileri eş zamanlı olarak toplar. Ardışık zamanlamada araştırmacı nicel ve nitel verileri ayrı zamanlarda toplar, analiz eder ve sonuçlara ulaşır. Çok fazlı kombinasyonel zamanlamada ise araştırmacı bir çalışma planı dâhilinde veri toplama ve analiz süreçlerini hem eş zamanlı hem de sıralı fazlarda gerçekleştirir. Verilerin birleştirilme stratejileri; sonuçları yorumlama sırasında birleştirme, veri analizi sırasında birleştirme, veri toplama sırasında birleştirme ve tasarım sırasında birleştirme olmak üzere dört stratejiden oluşur.

Creswell ve Clark (2007) karma desen araştırmacılarına yararlı bir çerçeve sunan altı temel karma araştırma tipi önermektedir (Şekil 3). Bunlar yakınsak (convergent) paralel desen, açımlayıcı (explanatory) sıralı desen, keşfedici (exploratory) sıralı desen, gömülü (embedded) desen, dönüşebilen (transformative) desen ve çok fazlı (multiphase) desendir. Yakınsak paralel desen nicel ve nitel yöntemin araştırma sürecinin aynı döneminde zamansal olarak çakışık ya da birbirine çok yakın olarak gerçekleştiği, eşit önceliğe ya da öneme sahip olduğu, analizlerinin birbirinden bağımsız gerçekleştiği fakat sonuçların genel yorumlama esnasında birleştirilerek ele alındığı bir karma araştırma desenidir. Açımlayıcı sıralı desen zamansal anlamda birbirinden ayrık ancak birbiriyle etkileşimli iki farklı fazdan oluşur. Bu desen nicel araştırma safhasıyla başlayıp nitel araştırmayla devam eder. Nicel verilerin toplanması, analiz edilmesi ve sonuçlarının yorumlanması nitel araştırmaya yol gösterici ve ışık tutucu niteliktedir. Nicel safha, araştırma sorularının adreslenmesinde kritik bir öneme sahiptir. Araştırmacı nitel sonuçların başlangıçtaki nicel sonuçların açıklanmasına nasıl katkı sağladığını yorumlar. Açımlayıcı desenden farklı olarak keşfedici desen, başlangıçta nitel verilerin toplanmasına, analiz edilmesine ve yorumlanmasına öncelik verir. Bu ilk ve keşfedici nitel sonuçların üzerine araştırmacı ilk bulguları test etmek ve genellemek üzere ikincil nicel araştırma fazını yürütür. Gömülü desenler araştırmacının tipik bir nicel ya da nitel araştırma içerisinde destekleyici bir faz olarak nitel ya da nicel bir araştırma yürütmesidir. Araştırmacı nitel bir araştırma yöntemini izleyerek yürüttüğü çalışmasında ek bir nicel araştırma (örneğin deneysel araştırma) ya da nicel araştırma yöntemi içerisine ek bir nitel araştırma (örneğin durum çalışması) ekleyebilir. Gömülü desende, yardımcı olarak yürütülen nicel ya da nitel faz genel araştırma deseninin bazı açılardan geliştirilmesi için eklenir. Dönüşebilen desen araştırmacının değişebilen kuramsal bir çerçeve (feminist, farklı etnik ya da radikal

görüşler, engelli kişiler vb.) içinde yapılandırdığı bir desendir. Nicel ve nitel araştırma yöntemlerinin (nicel ve nitel) etkileşimi, önceliği, zamanlaması ve birleştirilmesi değişebilen bu kuramsal çerçeve içerisinde şekillendirilir.



Şekil 3. Karma araştırma desenleri

Dönüşebilen kuramsal çerçeve bilginin tarafsız olmadığını, insan ilgilerinden etkilenen dünya görüşleri ve örtülü ya da açık değer varsayımları içerisinde şekillendiğini öne sürer (Sweetman, Badiee & Creswell, 2010). Araştırma pratikleriyle sorgulama felsefesini (paradigmalar) köprüleyen çalışmalarıyla tanınan Mertens (2003) feministler, farklı etnik / ırksal grup üyeleri ve engelli insanlar dâhil olmak üzere araştırmalardaki görüş ve değerlerin yeri hakkında düşünmeyi teşvik ederek dönüşebilen karma araştırma alanyazınına önemli katkılar sağlamıştır. Çok fazlı tasarım, araştırmacının genel araştırma planı içerisinde nicel ve nitel çalışmaları araştırma sorularıyla uyumlu olacak şekilde hem sıralı hem de eşzamanlı olarak işe koştuğu bir karma araştırma desenidir. Bu desen nispeten daha zamana yayılan çalışmalar (program geliştirme çalışmaları gibi) için uygundur.

Bu çalışmada araştırma deseni olarak gömülü desen benimsenmiştir. Benimsenen gömülü desen, deneysel araştırma çerçevesinde gerçekleştirildiği için gömülü deneysel desen olarak da adlandırılabilir. Gömülü tasarımlar adreslenmesi gereken farklı araştırma soruları için tek veri setinin yeterli olmaması durumunda her bir soru tipi farklı veri türlerinin kullanıldığı karma araştırma desenleridir. Gömülü deneysel desende araştırmacılar deneysel temelli nicel bir araştırma içinde ikincil bir araştırma sorusuna cevap vermek için nitel verileri kullanır (Creswell & Clark, 2007). Araştırmacılar çalışmanın kapsamını geliştirmek, deneysel müdahale sürecini incelemek ya da deneysel uygulamadaki katılımcı tepkilerini açıklamak gibi çeşitli amaçlar için nitel verileri nicel çalışma içerisine dâhil edebilir.

Gömülü araştırma deseni ile yakınsak (bitişik) paralel desen birbirlerine benzemektedir. Bununla birlikte aralarında önemli farklılıklar bulunmaktadır. Bitişik paralel desende nicel ve nitel araştırma genellikle eşit önceliğe ve öneme sahiptir. Ayrıca nicel ve nitel faz genellikle aynı araştırma sorusuna yönelik yürütülmektedir. Buna karşın gömülü desenlerde nicel ya da nitel araştırma yönteminden birisi daha yüksek önceliğe ya da daha geniş bir kapsama sahip olup, destekleyici ya da yardımcı amaçlarla çalışmaya ikincil olarak dâhil edilen diğer araştırma yöntemi farklı araştırma sorularının adreslenmesine yöneliktir.

Alanyazında en sık rastlanan gömülü desen bir deneysel desen içerisine nitel verilerin yerleştirilmesidir. Bu çalışmada da farklı zaman diliminde gerçekleştirilen (bundan sonra durum çalışması olarak bahsedilecek) iki deneysel desen içerisine de nitel araştırma safhaları yerleştirilmiştir.

Araştırmanın Evreni ve Örneklemi

Çalışma evrenini Mesleki ve Teknik Anadolu Liselerinde (MTAL) öğrenim gören onuncu sınıf öğrencileri oluşturmaktadır. Çalışma evreninden örneklem seçimi elverişli örnekleme tekniği ile gerçekleştirilmiştir. Buna göre çalışma grubunu Adana ilindeki bir MTAL'nin Bilişim Teknolojileri alanında öğrenim gören toplam 81 onuncu sınıf öğrencisi oluşturmuştur. Veriler birbirinden bağımsız iki durum çalışmasından elde edilmiştir. 2015-2016 eğitim-öğretim yılının 2. döneminde gerçekleştirilen durum çalışması-1'e 41 öğrenci; 2016-2017 eğitim-öğretim yılının 2. döneminde gerçekleştirilen durum çalışması-2'ye 40 öğrenci katılmıştır. BT alanında programlamaya ilişkin daha önce bir ders almamaları nedeniyle çalışmaya sadece onuncu sınıf öğrencileri dâhil edilmiştir. Her iki durum çalışmasında da ikişer şube kendi içinde deney ve karşılaştırma gruplarına kura yöntemiyle seçkisiz olarak atanmıştır. Katılımcıların, gruplara atanmaları sonrasında akademik başarı ortalaması ve cinsiyet açısından deney ve kontrol gruplarına dengeli bir şekilde dağıldığı görülmüştür (Tablo 34). Böylece gruplara ek bir müdahalede bulunulmasına gerek duyulmamıştır.

Tablo 34

Durum Çalışmalarına göre Deney ve Karşılaştırma Grubuna Seçkisiz Olarak Atanan Katılımcıların Sayı, Cinsiyet ve Akademik Başarı Ortalaması Açısından Dağılımı

Durum Çalışması	Şube	Grup	Öğrenci Sayısı		Akademik Başarı	
			Kadın (f)	Erkek (f)	X	SS
Durum Çalışması-1	Şube-1	Deney	2	9	50,7	8,5
		Karşılaştırma	3	7	50,5	7,9
	Şube-2	Deney	3	6	57,3	6,4
		Karşılaştırma	1	10	57,6	10,5
Durum Çalışması-2	Şube-1	Deney	2	9	69,6	7,2
		Karşılaştırma	1	9	68,9	8,4
	Şube-2	Deney	2	8	59,7	9,0
		Karşılaştırma	2	7	60,8	6,5

Deney ve karşılaştırma grubundaki öğrencilerin sayısı durum-1'de sırasıyla 20 ve 21; Durum-2'de ise 21 ve 19'dur. Her iki durum çalışmasında da katılımcı sayılarının test gruplarına göre dengeli dağıldığı görülmektedir. Bununla birlikte her iki durum çalışmasında da deney ve karşılaştırma gruplarındaki kadın sayısı

erkeklerden düşüktür. Durum-1'de deney ve karşılaştırma grubundaki kadın katılımcıların toplam sayısı sırasıyla 5 (%25) ve 4 (%19) iken; Durum-2'de 4 (%19) ve 3 (%16)'tür. Bu durum, çalışmanın Endüstri Meslek ve Anadolu Teknik Lisesi Programlarında öğrenim gören öğrencilerle gerçekleştirilmesinin bir doğurgası sayılabilir. Türkiye İstatistik Kurumunun katkılarıyla Milli Eğitim Bakanlığına bağlı Strateji Geliştirme Başkanlığı tarafından 2016-2017 eğitim-öğretim yılına ait örgün eğitim istatistikleri incelendiğinde Endüstri Meslek ve Anadolu Teknik Lisesinde okuyan öğrencilerin sırasıyla %88,5 ve %82'sinin erkek olduğu görülmektedir (Millî Eğitim Bakanlığı, 2017). Deney ve karşılaştırma gruplarında gerçekleştirilen etkinliklerle ilgili uygulamalar Şekil-4'te sunulmuştur. Ayrıca, katılımcılara ait yaş, bilgisayar kullanım deneyimi, günlük bilgisayar kullanım süresi ve daha önce programlama dersi alıp almama durumu bilgileri Tablo 35'te sunulmuştur.



Şekil 4. Durum çalışmalarına katılan deney (a, c) ve karşılaştırma (b, d) grubu öğrencileri

Tablo 35

Katılımcılara ait Bazı Betimsel İstatistikler

Durum Çalışması	Değişken	Kategori	Frekans (*f*)	Yüzde (%)
Durum Çalışması-1	Yaş aralığı	15-16 arası	34	83
		17-18 arası	7	17
	Bilgisayar kullanma deneyimi	Hiç	4	10
		0-1 yıl arası	2	5
		1-3 yıl arası	5	12
		3-6 yıl arası	19	46
		6-10 yıl arası	9	22
		10 yıl ve üzeri	2	5
	Evde kişisel bilgisayara sahip olma durumu	Evet	27	66
		Hayır	14	34
	Günlük bilgisayar kullanım süresi	Hiç	8	19,5
		0-1 saat	12	29,3
		1-3 saat	13	31,7
		3-6 saat	8	19,5
	Daha önce programlama eğitimi alma durumu	Evet	2	5
		Hayır	39	95
Durum Çalışması-2	Yaş aralığı	15-16 arası	31	77,5
		17-18 arası	9	22,5
	Bilgisayar kullanma deneyimi	Hiç	3	7,5
		0-1 yıl arası	2	5
		1-3 yıl arası	5	12,5
		3-6 yıl arası	13	32,5
		6-10 yıl arası	13	32,5
		10 yıl ve üzeri	4	10
	Evde kişisel bilgisayara sahip olma durumu	Evet	28	70
		Hayır	12	30
	Günlük bilgisayar kullanım süresi	Hiç	9	22,5
		0-1 saat	10	25
		1-3 saat	14	35
		3-6 saat	4	10
		6-10 saat	2	5
		10 saat ve üzeri	1	2,5
	Daha önce programlama eğitimi alma durumu	Evet	5	12,5
		Hayır	35	87,5

Durum çalışması 1'e katılan katılımcıların yaş ortalaması 16'dır. Katılımcıların %46'sı 3-6 yıl arası bilgisayar kullanma deneyimine sahiptir. Katılımcılardan %32'si günlük ortalama 1-3 saat; %29'u 0-1 saat bilgisayar kullanım süresine sahiptir. Katılımcıların %66'sının evlerinde bilgisayar erişimi bulunmaktadır. Ayrıca katılımcıların %96'sının daha önce bir programlama deneyimi bulunmamaktadır. Benzer şekilde, durum çalışması-2'ye katılan katılımcıların yaş ortalaması da 16'dır. Katılımcıların %32,5'i 3-6 yıl, yine %32,5'i 6-9 yıl arası bilgisayar kullanma deneyimine sahiptir. Durum çalışması-2'deki bu durumun bilgisayar ile ilk tanışma yaşının her yıl giderek düşmesinden kaynaklandığı düşünülmektedir. Katılımcıların %35'i günlük ortalama 1-3 saat; %47,5'inin 0-1 saat ortalama bilgisayar kullanım süresi bulunmaktadır. Katılımcıların %70'inin evlerinde kişisel bilgisayar erişimi bulunmaktadır. Öte yandan %87,5'nin (n=35) daha önce bir programlama eğitimi deneyimi bulunmamaktadır.

Veri Toplama Süreci

Veri toplama süreci, Hacettepe Üniversitesi Etik Komisyonunun inceleme ve kurul onayından sonra gerçekleştirilmiştir (EK-F). Uygulama öncesi tüm katılımcılar çalışma hakkında bilgilendirilmiştir. Çalışmaya katılım için gönüllü olan öğrenciler, velilerine onay formu imzalatıldıktan sonra çalışmaya dâhil edilmiştir.

Çalışma verileri 2015-2016 ve 2016-2017 eğitim öğretim yıllarının ikinci dönemlerinde toplanmıştır. Uygulamaların yapıldığı yer ve uygulama ortamına ilişkin bilgiler Tablo 36'da özetlenmiştir. Uygulamalar ses yalıtımının olduğu fiziksel açıdan uygun ışık ve havalandırma koşullarına sahip bir ortamda gerçekleştirilmiştir. Ayrıca uygulama ortamının bilgisayar, etkileşimli tahta, yazıcı ve yüksek hızlı İnternet alt yapısı gibi temel BİT araçları ve teknolojileri yönünden yeterli bir ortam olduğu görülmektedir.

Çalışma verileri BT alanının 10. sınıfında alan ortak dersi olarak verilen Programlama Temelleri dersi kapsamında toplanmıştır. Programlamaya giriş seviyesindeki bu ders problem temelli düzenlemeler içerisinde temel bilgisayımsal kavramların anlaşılmasının ve başkalarıyla paylaşılmasının (diğer öğrenciler ve öğretmenler) istendiği ve sonunda bir çözüm önerilmesi için bilgisayımsal düşünme (BD) becerilerinin işe koşulmasını gerektiren özel bilgisayımsal kuralları barındırmaktadır.

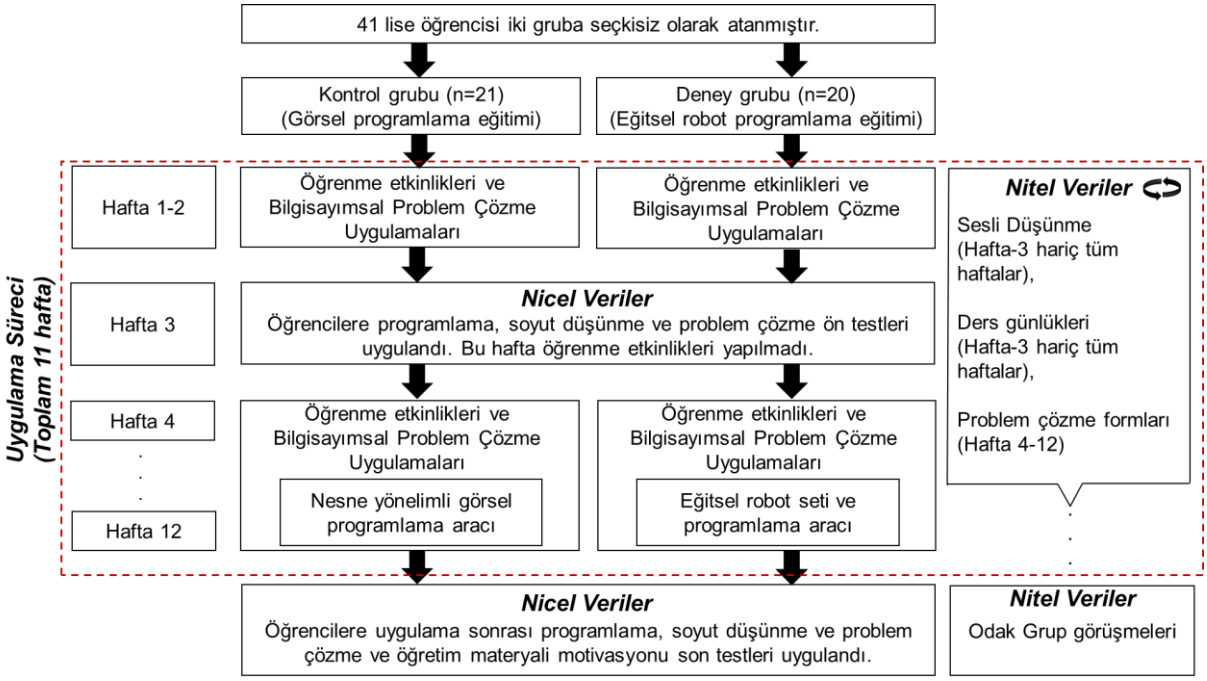
Tablo 36

Uygulamanın Gerçekleştiği Ortama İlişkin Bilgiler

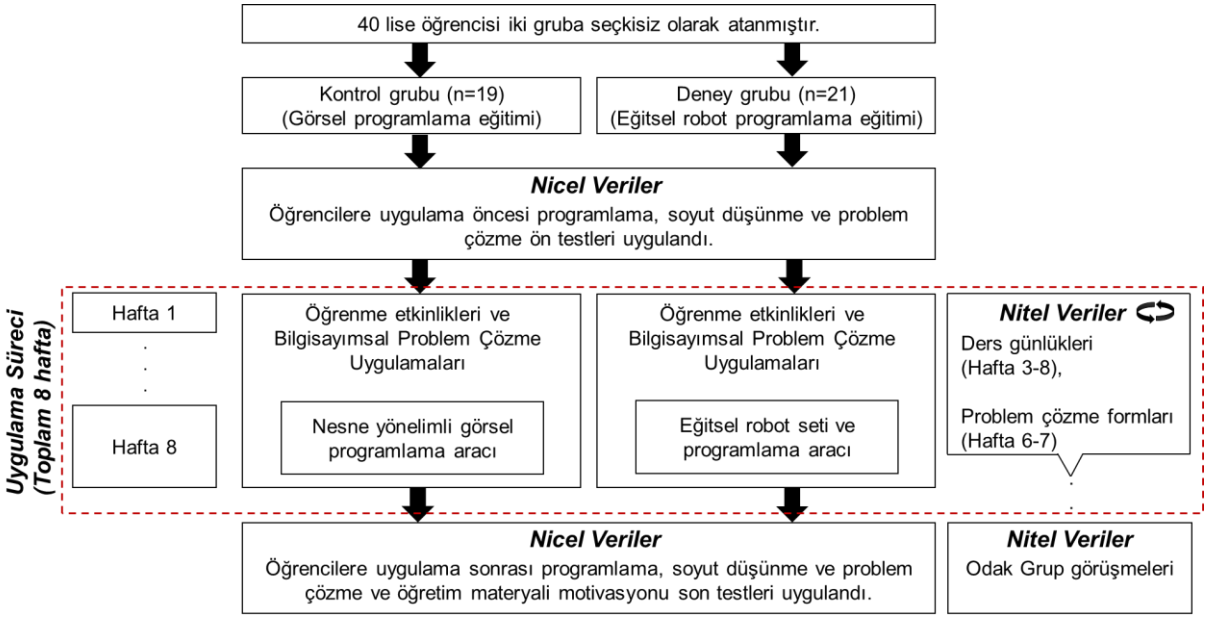
Uygulama yeri	MTAL
Statü	Devlet Okulu
Okul Türü	MTAL
Öğretim Düzeyi	Lise (9-12.sınıf)
Toplam Öğrenci Sayısı	1200
Öğrenci Yaş Aralığı	15-19
Toplam Öğretmen Sayısı	100
Toplam İdareci Sayısı	6
Toplam Destek Personeli Sayısı	4
Sınıflardaki Ortalama Öğrenci Sayısı	30
Bilgisayar Laboratuvarı Sayısı	5
Okuldaki Toplam Bilgisayar Sayısı	160
Uygulama Ortamı	Bilgisayar Laboratuvarı
Uygulama Ortamındaki (Sınıf içi) BİT donanımları	Etkileşimli Tahta, Öğrenci Bilgisayarı (15 adet), Öğretmen Bilgisayarı (2 adet), Yazıcı (1 adet) Yüksek Hızlı İnternet Altyapısı (Aktif değil)

Her iki durum çalışmasında da test gruplarındaki öğrenme etkinlikleri aynı öğretim elemanı rehberliğinde sürdürülmüştür. Böylece gruplar arasında öğretici etkisinden kaynaklanabilecek farklılıkların asgari düzeyde tutulması amaçlanmıştır. Aynı zamanda mevcut çalışmanın yürütücüsü olan dersin öğretim elemanı tarafından hazırlanan haftalık ders içerikleri ve etkinlik planları uygulama öncesinde BÖTE alanındaki bir uzman tarafından gözden geçirilmiştir. Ayrıca gruplardaki öğretim koşullarında oluşabilecek herhangi bir farklılığı gidermek için her dersten sonra gerçekleştirilen tüm öğrenme etkinlikleri bu iki kişi tarafından tartışılmıştır. Buna göre özellikle durum çalışmalarının ilk haftalarında gerçekleştirilen problem çözme etkinliklerinde öğrencilere yöneltilen problem sayısının ilerleyen haftalarda azaltılmasına, bununla birlikte problemlerin karmaşıklık seviyelerinin arttırılmasına karar verilmiştir.

Nicel ve nitel veri toplama fazlarının birbiri içerisine geçtiği durum çalışmalarında gerçekleştirilen araştırma prosedürü ve veri toplama adımları sırasıyla Şekil 5 ve 6'da şemalaştırılmıştır.



Şekil 5. Durum Çalışması-1'de gerçekleştirilen araştırma yöntemi ve veri toplama süreçleri



Şekil 6. Durum Çalışması-2'de gerçekleştirilen araştırma yöntemi ve veri toplama süreçleri

Şekil 5'te ve Şekil 6'da görüldüğü gibi durum çalışması 1 ve 2'de toplanan nicel veri kaynakları programlama başarısı ve soyut düşünme becerisi testleri ile problem çözme becerisi ve öğretim materyallerine yönelik motivasyon anketleri iken nitel veri kaynakları ders günlükleri, problem çözme formları, sesli düşünme (sadece durum çalışması-1'de) ve odak grup görüşmesi kayıtlarıdır. Ayrıca her iki durum çalışmasından önce katılımcılara yaş, cinsiyet, bilgisayar kullanım deneyimi, evde kişisel bilgisayara sahip olup olmama durumu, günlük bilgisayar kullanım süresi ve programlama deneyimi ile ilgili soruların yer aldığı bir veri anketi uygulanmıştır.

Durum çalışmalarında uygulama öncesi öğrenciler iki gruba ayrılarak BD'yle ilgili öğretimsel etkinlikler bir grupta (karşılaştırma grubu) nesne yönelimli görsel programlama araçları, diğer grupta (deney grubu) ise eğitsel robot kiti ve yazılım araçları aracılığıyla gerçekleştirilmiştir. Eğitsel robotlarla çalışmanın başlangıç aşamasında öğrenenler robotları monte etmek (tasarım) için kullanmıştır. Daha sonra programlamaya ilişkin bilgi ve becerilerini ilerletmeleri için onlardan robotları işlevsel olarak manipüle etmeleri beklenmiştir. Aslında dış dünyayla iletişim kurmak için bir araç olarak görülen eğitsel robot programlama etkinlikleri BD becerilerinin sergilenmesi için geleneksel programlama koşullarına bir alternatif olarak düşünülmüştür.

Her iki durum çalışmasında da her hafta dersin ilk bölümünde öğrencilere programlama kavramlarıyla ilgili temel bilgileri içeren bir sunum yapılmıştır. Durum çalışmalarında haftalık olarak işlenen ders içerikleri Tablo 37 ve 38'de sunulmuştur.

Tablo 37

Durum Çalışması 1'deki Haftalık Ders İçerikleri

Hafta No	Eğitsel Robot Programlama	Nesne Yönelimli Görsel Programlama
#1	• Robot ve lego parçalarının tanıtımı • Robot montajı ve robotun çalışır hale getirilmesi	• Visual Studio C# Programının Tanıtımı ve Gelişmiş Özellikleri
#2	• Lego Mindstorms NXT yazılımının Tanıtımı ve Örnek Robot Programlama Etkinlikleri • Değişken kavramı ve türleri	• Değişken kavramı ve türleri • Atama işlemleri • Giriş/Çıkış İşlemleri • Açıklama Satırları
#4	• POLYA'nın Problem Çözme Aşamalarının tanıtımı	• POLYA'nın Problem Çözme Aşamalarının tanıtımı • İşlem Önceliği

	• Robot C yazılımın tanıtımı ve Örnek programlama Etkinlikleri	• İlişkisel ve Mantıksal Operatörler
#5	• Veri Türleri • Atama İşlemleri • Operatörler (Aritmetiksel, Karşılaştırma, Mantıksal ve String) • Formatlı ekran çıkış işlemleri • Random Fonksiyonu	• Karar Yapıları o (if… else) o (İç içe if yapısı)
#6	• Karar Yapıları o if...else o İç-içe o Switch ...Case • Sensörlerden değer okuma	• İşlem Önceliği (Devam) • Karar Yapılarıyla (if… else, iç içe if ve switch case) ilgili örnek Soru çözümleri
#7	• Hazır Fonksiyonlar o Matematiksel Fonksiyonlar o String Fonksiyonları	• Hazır Fonksiyonlar o Matematiksel Fonksiyonlar o String Fonksiyonları
#8	• Döngü Kavramı • Döngü Yapıları o While Döngüsü o Do…While Döngüsü o For döngüsü	• Döngü Kavramı • Döngü Yapıları o While Döngüsü o Do…While Döngüsü o For döngüsü
#9	• Döngü Kavramı • Döngü Yapıları o While Döngüsü o Do…While Döngüsü o For döngüsü	• Döngü Kavramı • Döngü Yapıları o While Döngüsü o Do…While Döngüsü o For döngüsü
#10	• Diziler o Tek boyutlu diziler o İki boyutlu diziler	• Diziler o Tek boyutlu diziler o İki boyutlu diziler
#11	• Alt programlar (yordam ve fonksiyonlar) o Gelişmiş (parametreli) fonksiyonlar o Geriye değer döndüren fonksiyonlar	• Alt programlar (yordam ve fonksiyonlar) o Gelişmiş (parametreli) fonksiyonlar o Geriye değer döndüren fonksiyonlar
#12	• Hata tahmini ve hata ayıklama o Hata türleri o Hata mesajları	• Hata tahmini ve hata ayıklama o Hata türleri o Hata mesajları

Durum çalışması 1'deki deneysel uygulama toplam 12 hafta sürerken; öğrencilerin üzerindeki haftalık yükü azaltmak ve bazı temel kavramların kazanılmasını (karar yapıları ve döngüler gibi) kolaylaştırmak adına Durum çalışması 2'de konu yoğunluğu azaltılmıştır. Ayrıca deneysel uygulama sırasında meydana gelebilecek olumsuzlukları azaltmak adına uygulama süresi de kısaltılmıştır.

Tablo 38

Durum Çalışması 2'deki Haftalık Ders İçerikleri

Hafta No	Eğitsel Robot Programlama	Nesne Yönelimli Görsel Programlama
#1	• Robot ve lego parçalarının tanıtımı • Robotun çalışır hale getirilmesi • Polya'nın Problem Çözme Adımları • Değişkenler	• Visual Studio C# Programının Tanıtımı ve Gelişmiş Özellikleri • Polya'nın Problem Çözme Adımları • Değişkenler
#2	• Atama İşlemleri • Operatörler (Aritmetiksel, Karşılaştırma, Mantıksal ve String) • Formatlı ekran çıkış işlemleri • Random Fonksiyonu	• Atama İşlemleri • Operatörler (Aritmetiksel, Karşılaştırma, Mantıksal ve String)
#3	• Karar Yapıları o if...else • Sensörlerden değer okuma	• Karar Yapıları o if...else
#4	• Karar Yapıları o İç-içe if o Switch ...Case	• Karar Yapıları o İç-içe if o Switch ...Case
#5	• Karar Yapıları o İç-içe if o Switch…Case	• Karar Yapıları o İç-içe if o Switch ...Case
#6	• Döngü kavramı • Döngü yapıları (While, Do.While, For döngüsü)	• Döngü kavramı • Döngü yapıları (While, Do.While, For döngüsü)
#7	• Döngü kavramı • Döngü yapıları (While, Do.While, For döngüsü)	• Döngü kavramı • Döngü yapıları (While, Do.While, For döngüsü)
#8	• Döngü kavramı • Döngü yapıları (While, Do.While, For döngüsü)	• Döngü kavramı • Döngü yapıları (While, Do.While, For döngüsü)

Dersin uygulama bölümü öğrencilerden bilgisayımsal problemlere çözüm üretmelerinin istendiği aşamadır. Bu aşamada problem çözme süreçlerinin iskelenmesi amacıyla öğrencilere problem çözme formları sağlanmıştır. Eğitsel robot programlama grubundaki öğrenciler uygulama boyunca takım halinde çalışmıştır. Öte yandan, görsel programlama grubundaki öğrencilere de bireysel ya da takım halinde çalışma konusunda esneklik sağlanmıştır. Bilgisayımsal problem çözme uygulama örnekleri EK-A'da sunulmuştur. Problem çözme sürecinde yaşanan deneyimleri adreslemek adına haftalık olarak dersin uygulama oturumu sonunda ders günlükleri aracılığıyla öğrenci yansımaları alınmıştır. Problem çözme sürecinde öne çıkan bilişsel unsurları ortaya çıkarmak adına dersin uygulama oturumu öncesi dağıtılan problem çözme formları uygulama sonunda toplanmıştır. Ayrıca durum çalışması-1'de her hafta eğitsel robot programlama grubundaki bir takımdan problem çözme süreçleri esnasındaki düşüncelerini sesli düşünme tekniği ile dışa vurmaları istenmiştir. Öğrencilerin problem çözme süreçleri boyunca yaşadıkları deneyimler ses kayıt cihazı ile kaydedilmiştir. Böylece bilgisayımsal düşünme sürecinde ortaya çıkan bilişsel unsurlara ilişkin çıkarımların elde edilmesi amaçlanmıştır.

Veri Toplama Araçları

Bu bölümde veri toplama sürecinde kullanılan araçlar tanıtılmaktadır. Mevcut araştırmada veri toplamak için nitel ve nicel olmak üzere iki farklı türde araç seti kullanılmıştır. Nicel veriler programlama başarısı testi, soyutlama becerisi envanteri, problem çözme ölçeği, öğretim materyalleri motivasyon anketi araçlarıyla; nitel veriler demografik veri anketi, odak grup görüşme formu, ders günlükleri/yansıma raporları, sesli düşünme protokolü ve problem çözme formları aracılığıyla elde edilmiştir.

Programlama başarısı testi. Çalışma kapsamında temel programlama kavramları ve programlama mantığı kazanımının sorgulanması amacıyla araştırmacı tarafından geliştirilen bir başarı testi kullanılmıştır. Başarı testinin hazırlanması öncesinde alanyazında yer alan programlama başarısı testleri incelenmiştir. Test maddelerin hazırlanmasında Saygıner (2017) tarafından geliştirilen ve 18 maddeden oluşan programlama erişi testinden yararlanılmıştır. Bu testin Pascal programlama dili sözdizimine göre hazırlanan maddeleri sözdizimsel kurallardan arındırılarak kullanılmıştır. Sonuç olarak belirli bir programlama diline ya

da sözdizimine bağlı kalmaksızın hazırlanan 5 açık uçlu ve 20 çoktan seçmeli toplam 25 soruluk bir başarı testi hazırlanmıştır.

Programlama başarısı testinin kapsam geçerliği için uzman görüşüne başvurulmuştur. Beş yıl ve üzeri programlama öğretimi deneyimine sahip 3 öğretmen ve 1 akademisyen olmak üzere toplam 4 konu alanı uzmanından alınan dönütler doğrultusunda 4 madde (7., 12., 16. ve 20. sorular) yeniden düzenlenerek ölçme aracına son şekli verilmiştir. Görüşüne başvurulan uzmanlara ait demografik özellikler Tablo 39'da sunulmuştur.

Tablo 39

Programlama Başarısı Testinin Kapsam Geçerliği için Görüşüne Başvurulan Uzmanların Demografik Özellikleri

Cinsiyet	Meslek	Öğrenim Düzeyi	Branş	Akademik Ünvan	Mesleki Deneyim	Programlama Eğitimi Deneyimi
K	Öğretmen	Yüksek Lis.	Bilişim Tek.	Bilim Uzm.	10-15 yıl	5-10 yıl
E	Öğretmen	Lisans	Bilişim Tek.	-	5-10 yıl	5-10 yıl
E	Öğretmen	Lisans	Bilişim Tek.	-	10-15 yıl	10-15 yıl
E	Akademisyen	Doktora	BÖTE	Doç. Dr.	15-20 yıl	10-15 yıl

Hazırlanan başarı testi, madde analizleri için hâlihazırda bir mesleki ve teknik ortaöğretim kurumuna kayıtlı olan ve daha önce programlama eğitimi almış toplam 206 öğrenciye uygulanmıştır. Programlama başarısı test puanlarının puanlayıcı yanlılığından arınması ve araştırmanın daha rasyonel bir yapı kazanması adına açık uçlu sorulara verilen yanıtlar iki konu alanı uzmanı tarafından değerlendirilip, %100 fikir birliği sağlandığı durumda puanlama yapılmıştır. Fikir birliğinin sağlanmadığı durumlarda üçüncü bir konu uzmanının görüşüne başvurulmuştur. Toplam Dört MTAL'de öğrencilere basılı formda dağıtılan başarı testi örnek formuna ilişkin öğrenci yanıtları daha sonra araştırmacı tarafından sayısal ortama aktarılarak veri analizine hazır hale getirilmiştir.

Madde analizleri için alt-üst grup tekniğinden yararlanılmıştır. Öğrenciler testten aldıkları puanlara göre sıralanmış, %27'lik alt ve %27'lik üst grup seçilerek madde analizleri yapılmıştır. İstikrarın sağlanması için bir yandan her bir grupta olabildiğince çok öğrenci olması gerekirken diğer yandan iki grup arasındaki farkın belirginleşmesi için iki grubun olabildiğince farklı olması beklenmektedir (Matlock-

Hetzel, 1997). Analiz için alt ve üst gruplardan %27'lik dilimin seçilmesinin nedeni bu oranın madde analizleri için yeterli vaka (örnek durum) sağlarken normal dağılım içerisinde kalacak şekilde farklılıkları yüksek düzeye çıkarabilmesidir (Wiersma & Jurs, 1990). Test maddelerine ait güçlük indeksi, ayırt edicilik indeksi, varyans, standart sapma ve güvenirlik indeksi değerleri Tablo 40'ta sunulmuştur.

Tablo 40

Programlama Başarısı Testine İlişkin Madde İstatistikleri

Madde No	Madde Güçlük İndeksi	Madde Ayırt Edicilik İndeksi	Varyans	Standart Sapma	Güvenirlik İndeksi
1	0,482	0,491	0,250	0,500	0,245
2	0,600	0,582	0,240	0,490	0,285
3	0,545	0,400	0,248	0,498	0,199
4	0,555	0,564	0,247	0,497	0,280
5	0,300	0,455	0,210	0,458	0,208
6	0,745	0,291	0,190	0,436	0,127
7	0,418	0,618	0,243	0,493	0,305
8	0,764	0,327	0,180	0,425	0,139
9	0,509	0,364	0,250	0,500	0,182
10	0,609	0,491	0,238	0,488	0,240
11	0,400	0,582	0,240	0,490	0,285
12	0,227	0,273	0,176	0,419	0,114
13	0,455	0,582	0,248	0,498	0,290
14	0,545	0,400	0,248	0,498	0,199
15	0,464	0,491	0,249	0,499	0,245
16*	0,164	0,145	0,137	0,370	0,054
17	0,518	0,600	0,250	0,500	0,300
18	0,264	0,455	0,194	0,441	0,200
19	0,282	0,382	0,202	0,450	0,172
20	0,345	0,473	0,226	0,476	0,225
21*	0,109	0,109	0,097	0,312	0,034
22	0,436	0,509	0,246	0,496	0,252
23	0,573	0,673	0,245	0,495	0,333
24	0,436	0,473	0,246	0,496	0,234
25	0,473	0,400	0,249	0,499	0,200

*Not: * işaretli maddeler testten çıkarılmıştır.*

Ayırt edicilik indeksi 0,19'dan küçük olan maddeler ayırt ediciliği düşük ya da zayıf olarak nitelendirilmektedir (Ebel, 1972). Herhangi bir düzeltme yapılmadığı takdirde bu maddelerin testten çıkarılması önerilmektedir. Buna göre madde analizi

sonrasında ayırt edicilik indeksi 0,19'dan küçük olan 16 ve 21. maddeler testten çıkarılmıştır. Programlama başarısı testinin son haline ait test istatistikleri Tablo 41'de sunulmuştur. Başarı testinde yer alan soru örnekleri EK-B'de sunulmuştur.

Tablo 41

Programlama Başarısı Testine İlişkin Test İstatistikleri

N		110
Kuder-Richardson (KR) 20		0,878
Ortalama güçlük indeksi		0,476
Ortalama ayırıcılık indeksi		0,473
X		10,945
Medyan		9,5
Mod		7
Standart sapma		5,794
Varyans		33,57
Max		23
Min		2
Ranj		21
Çarpıklık katsayısı	Çarpıklık	0,193
	Çarpıklık std. hata	0,230
Basıklık katsayısı	Basıklık	-1,480
	Basıklık std. hata	0,457

Tablo 41 incelendiğinde testin KR-20 iç tutarlık katsayısının 0,878 olduğu görülmektedir. Bu değer test güvenirliğinin yüksek olduğunu göstermektedir. Testin ortalama güçlük indeksi 0,476 olarak tespit edilmiştir. Bu değer testin orta derecede güç olduğunu göstermektedir. Ayrıca 0,473 olan ortalama ayırıcılık indeksi testin yüksek düzeyde ayırt edici olduğuna işaret etmektedir. Testin çarpıklık katsayısının kendi standart hatasına bölünmesiyle elde edilen standart değer 0,84 olarak hesaplanmıştır. Bu değer 0,05 anlamlılık düzeyindeki Z değerleri arasında (-1,96, +1,96) olduğu için verilerin normal dağılım gösterdiği söylenebilir.

Soyutlama becerisi envanteri. Soyutlama becerisinin değerlendirilmesi için alanyazında yer alan testlerin sayısı oldukça kısıtlıdır. Mevcut testlerin çoğu temelde mantıksal çıkarım üzerine odaklanan formal işlemlere yöneliktir. Öte yandan soyutlama tek bir monolitik kavram olmayıp, daha ziyade bir dereceye kadar benzer zihinsel süreçlerden oluşan farklı işlem setleri gerektirmektedir. Ayrıca sadece bilgisayar bilimi kavramlarını kullanarak geliştirilen araçlarla öğrencilerin soyut düşünme becerilerini belirlemek, bu becerinin farklı disiplinlerdeki öğrencilere uygulanmasını oldukça zorlaştırabilir. Hill vd. (2008) bilgisayar bilimleri dışında farklı

disiplinlerdeki öğrencilere uygulanabilecek temelde kavramsal, formal ve betimsel soyutlama olmak üzere 3 bağımsız bilişsel beceriye yönelik bir soyutlama envanteri geliştirmiştir. Geliştirilen bu envanter kavramsal soyutlama algısına yönelik 5'li Likert türündeki 3 maddeden oluşan bir ölçek, formal ve betimsel soyutlama becerisine yönelik çoktan seçmeli sırasıyla 6 ve 3 soruluk testlerden oluşmaktadır. Hill vd. (2008) soyutlama envanterinin geliştirilmesinde Hazzan ve Kramer'in (2007) soyutlamaya uygun görev türlerinin belirlenmesi konusundaki önerilerinden yararlanmıştır. Bu çalışma kapsamında ilgili soyutlama envanterinin formal ve betimsel soyutlama testleri kullanılmıştır. Soyut düşünme envanterinin Türkçeye uyarlama çalışması için öncelikle özgün envanter 3 yabancı dil uzmanı tarafından Türkçeye çevrilmiştir. Bu üç farklı çeviri formu ilgili maddelere göre bir araya getirildikten sonra BÖTE alan uzmanı 11 öğretim elemanından madde çevirilerini İngilizce-Türkçe uyumluluğu açısından değerlendirmeleri istenmiştir. Uzmanların ölçme aracının Türkçeye uygunluğu hakkındaki görüşleri 2'li derecelendirme ölçeği (0=Çeviri uygun değil, 1=Çeviri uygun) kullanılarak alınmıştır. Ayrıca uzmanlardan her bir madde için yapılan üç farklı çeviriden dil ve ifade olarak en uygun olanını seçmesi istenmiştir. Uzmanlar tarafından uygun bulunan bir çeviri önerisi bulunmadığı durumlar için onların görüş ve önerilerini belirtmelerine fırsat vermek adına her bir maddenin altına bir açıklama satırı eklenmiştir. Görüşüne başvurulan alan uzmanlarına ait demografik veriler Tablo 42'de sunulmuştur.

Tablo 42

Soyutlama Envanterinin Türkçe Formunun Uygunluğuna İlişkin Görüşüne Başvurulan Uzmanlara Ait Demografik Bilgiler

Cinsiyet	Akademik Ünvan	Deneyim
E	Prof. Dr.	30 yıl ve üzeri
E	Doç. Dr.	20-25 yıl
K	Prof. Dr.	25-30 yıl
E	Arş. Gör.	5-10 yıl
E	Arş. Gör.	0-5 yıl
E	Arş. Gör.	5-10 yıl
K	Doç. Dr.	15-20 yıl
K	Prof. Dr.	15-20 yıl
E	Prof. Dr.	20-25 yıl
K	Arş. Gör.	5-10 yıl
E	Dr. Öğr. Üyesi	10-15 yıl

Uzmanlar arası uyum için hesaplanan Fleiss Kappa istatistiği (k=0,35) orta derece bir uyuma işaret ederken (Fleiss, 1981), uzmanlar arası gözlenen uyum %71 olarak belirlenmiştir. Uzmanlar tarafından yapılan önerilerin de dikkate alındığı çalışmada, uzmanlar arasında en yüksek uygunluk oranına sahip maddeler seçilerek formal ve betimsel soyutlama test formlarına son şekli verilmiştir. Seçilen maddelerin uzmanlar tarafından uygun bulunma yüzdelerinin formal ve betimsel soyutlama test maddeleri için sırasıyla % 55-73 ve %73-91 arasında değiştiği belirlenmiştir. Formal ve betimsel soyutlama test sorularının ölçülmek istenen hedef yapıyı ne derece yordadığına karar vermek için testler soyutlama becerilerinin yüksek olacağı öngörülen farklı disiplinlerdeki 7 alan uzmanına (BÖTE, bilgisayar mühendisliği ve matematik gibi) uygulanarak sonuçlar değerlendirilmiştir. Uzmanların sorulara yüksek oranda benzer yanıtlar verdiği saptanmıştır.

Problem çözme ölçeği (envanteri). Orjinali Heppner (1988) ve Heppner ve Petersen (1982) tarafından geliştirilen Problem Çözme Ölçeği (PÇÖ), kişilerin problem çözme yetenekleri ve problem çözme stilleri hakkındaki algılarını değerlendirmek için tasarlanmış 35 maddelik Likert türünde bir ölçme aracıdır. Ölçeğe verilebilecek yanıtlar 1 (kesinlikle katılıyorum) ile 6 (kesinlikle katılmıyorum) arasında değişmektedir. Ölçekte 9., 22. ve 29. maddeler puanlama dışı tutulurken, 1., 2., 3., 4., 11., 13., 14., 15., 17., 21., 25., 26., 30., 32. ve 34. maddeler ters olarak puanlanmaktadır. Ölçekten alınabilecek en düşük puan 32, en yüksek puan ise 192'dir. Ölçekten alınan toplam puanın yüksekliği, bireyin problem çözme konusunda kendini yetersiz olarak algıladığını göstermektedir (Savaşır & Şahin, 1997). Ölçeğin Cronbach alfa katsayısı 0,90 olarak bulunmuştur (Heppner, 1988). Problem-çözmeye karşı güven (Problem-Solving Confidence), yaklaşma-kaçınma (Approach-Avoidance) ve kişisel kontrol (Personal Control) olmak üzere üç alt ölçek için elde edilen Cronbach alfa katsayıları sırasıyla 0,85, 0,84 ve 0,72 olarak hesaplanmıştır (Heppner, 1988). Ölçek Şahin, Şahin ve Heppner (1993) tarafından Türkçeye uyarlanmıştır. Ölçeğin Türkçe formunun Cronbach alfa değeri 0,88 olarak bulunmuştur. Bununla birlikte Türkçe uyarlama çalışmasında yürütülen faktör analizi sonuçlarının, ölçeğin orijinaliyle paralellik taşımadığı gözlemlenmiştir. Türkçe formunda ölçek maddeleri 6 faktör etrafında toplanmıştır. Bunlar sırasıyla "aceleci" (13., 14., 15., 17., 21., 25., 26., 30. ve 32. madde), "düşünen" (18., 20., 31., 33. ve 35. madde), "kendine güvenli" (5., 23., 24., 27., 28. ve 34. madde), "kaçıngan" (1.,

2., 3. ve 4. madde), “değerlendirici” (6., 7. ve 8. madde), ve “planlı” (10., 12., 16. ve 19. madde) yaklaşımlardır. Altı alt faktöre ait Cronbach alfa katsayıları sırasıyla 0,78, 0,76, 0,74, 0,69, 0,64 ve 0,59 olarak hesaplanmıştır. Ayrıca, alt ölçekler düzeyinde 11. madde dikkate alınmamaktadır. Araştırmalarda ölçekten elde edilen toplam puanlar ya da her bir alt ölçekten elde edilen puanlar ayrı ayrı kullanılabilir. Bu çalışmada ölçek toplam puanının dikkate alınmasının daha uygun olacağı düşünülmüştür. Bunun temel nedeni ölçeğin Türkçe formunun faktöriyel geçerliğinin orijinal formuna göre önemli farklılıklar arz etmesidir. Çalışmada öğrencilerin PÇÖ ön test ve son test puanlarına ait Cronbach alfa değerleri durum çalışması-1’de sırasıyla 0,758 ve 0,836; durum çalışması-2’de ise 0,826 ve 0,876 olarak hesaplanmıştır.

Öğretim materyalleri motivasyon anketi. Öğrenme ve motivasyon araştırmacıları endüstride etkili olduğu bulunan kuram ve kavramların bazılarını eğitim-öğretim ortamını iyileştirmek adına motivasyon modellerinin geliştirilmesine uygulamıştır. Bunlardan birisi de Keller tarafından ortaya konan ARCS motivasyon tasarım modelidir. ARCS modeli öğretimi motive etmek için 4 temel stratejik bileşen tanımlamaktadır. Adını da baş harflerinden aldığı bu bileşenler dikkat (attention), uygunluk/ilgi (relevance), güven (confidence) ve tatmindir (satisfaction) (Keller, 1983, 1987a, 1987b). ARCS motivasyon tasarım modeli tanınmış ve öğretim tasarımında yaygın olarak uygulanan bir modeldir (Small, 1999). Sade ve etkili bir model olan ARCS modeli temellerini bir dizi motivasyon kuramından alsa da bunlardan en göze çarpanı beklenti-değer kuramıdır (Keller, 1983; Porter & Lawler, 1968; Vroom, 1964). Beklenti-değer kuramında çaba ana ölçülebilir motivasyonel çıktı olarak tanımlanır. Çabanın ortaya çıkması için gerekli iki ön koşul bulunmaktadır. Bunlar kişinin göreve değer vermesi (1) ve kişinin görevde başarılı olabileceğine inanmasıdır (2). Bu nedenle bir öğretim uygulamasında öğrenme görevlerinin öğrenciler için ilgi çekici ve anlamlı olması kadar öğrenme hedeflerinin başarılı bir şekilde gerçekleştirilmesi adına onlarda olumlu beklentiler oluşmasını destekleyecek bir yolla sunulması gerekmektedir.

Çalışmada uygulanan farklı programlama modalitelerinin öğrenci motivasyonuna olan etkisini değerlendirmek için orijinali Keller tarafından geliştirilen Öğretim Materyalleri Motivasyon Anketi (Instructional Materials Motivation Survey) kullanılmıştır. Anketin amacı öğrencilerin öğrenmeye karşı genel motivasyon

düzeylerini ölçmek değil, belirli bir öğretim materyali ile öğrencilerin nasıl motive olduğunu ya da olunmasını beklediğini belirlemektir. Öğretim Materyalleri Motivasyon Anketi (ÖMMA) 4 faktör (dikkat, uygunluk, güven, tatmin) altında toplanan 5'li Likert türünde 36 maddeden oluşmaktadır (Keller, 1987a, 1987b). Faktörlerdeki madde sayısı incelendiğinde dikkat faktörünün 12 madde, uygunluk faktörünün 9 madde, güven faktörünün 9 madde ve tatmin faktörünün 6 madde etrafında şekillendiği görülmektedir. Anketten alınan yüksek puanlar öğrenenlerin öğretim materyaline karşı motivasyonunun yüksek olduğuna işaret etmektedir. Anketin orijinal formunda 10 madde olumsuz ifade içermektedir. Anket maddelerine verilebilecek yanıtlar 1'den (kesinlikle katılmıyorum) 5'e (kesinlikle katılıyorum) kadar derecelendirilmiştir. Anketin Cronbach alfa güvenirlik katsayısı 0,93 olarak hesaplanmıştır (Keller, 2010). ÖMMA'nin Türkçe uyarlaması Kutu ve Sözbilir (2011) tarafından yapılmıştır. Anketin Türkçe formu dikkat-uygunluk ve güven-tatmin olmak üzere 2 faktör ve toplam 24 maddeden oluşmakta olup, Cronbach alfa değeri 0,83 olarak hesaplanmıştır (Kutu & Sözbilir, 2011). Faktörlerin güvenirlik katsayıları dikkat-uygunluk için (1., 2., 3., 4., 5., 6., 7., 8., 9., 10. ve 11. maddeler) 0,79; güven-tatmin için (12, 13., 14., 15., 16., 17., 18., 19., 20., 21., 22., 23. ve 24. maddeler) 0,69 olarak bulunmuştur. ÖMMA'nın Türkçe formunda 3., 12., 14., 16. ve 18. maddeler olumsuz ifadeler içerdiği için ters puanlanmaktadır. Mevcut çalışmada Türkçe formu kullanılan ÖMMA'nın Cronbach alfa güvenirlik katsayısı durum çalışması-1 için 0,838; durum çalışması-2 için 0,872 olarak hesaplanmıştır. Dikkat-uygunluk ve güven-tatmin faktörlerine ait Cronbach alfa değerleri durum çalışması-1 için sırasıyla 0,672 ve 0,825 iken durum çalışması-2 için 0,776 ve 0,834 olarak bulunmuştur.

Demografik veri anketi. Demografik veri anketinde cinsiyet, yaş, bilgisayar kullanım deneyimi, evde kişisel bilgisayara sahip olma durumu, günlük ortalama bilgisayar kullanım süresi, bilgisayar programlamayla ilgili daha önce bir derse ya da kursa katılma durumu ve yapmaktan hoşlanılan bazı etkinlik türleri (problem çözme, kompozisyon yazma, resim yapma, sınıfta konuşma yapma gibi) ile ilgili toplam 7 adet madde bulunmaktadır.

Odak grup görüşme formu. Her iki durum çalışmasında da uygulama sonrası hem eğitsel robot hem de nesne yönelimli görsel programlama gruplarıyla yarı yapılandırılmış bir görüşme formu aracılığıyla yaklaşık 15 dk. süren odak grup

görüşmeleri gerçekleştirilmiştir. Görüşme soruları uygulamanın hoşa giden ve gitmeyen yönleri, bireysel ya da takım olarak yapılan etkinliklerin görev başarısı ve öğrenme sürecine etkileri, etkinlikler esnasında karşılaşılan sorunlar ve bu sorunlara yönelik olarak geliştirilen çözüm stratejileri, programlama etkinliklerinin bilişsel becerilere ve diğer derslere etkisine yönelik sorulardan oluşmaktadır. Araştırmacı tarafından hazırlanan görüşme soruları BÖTE alan uzmanı farklı bir araştırmacı tarafından gözden geçirilmiştir. Gelen dönütler doğrultusunda görüşme formuna son şekli verilmiştir. Odak grup görüşme formu EK-C'de sunulmuştur.

Her iki durum çalışmasında da test grupları kendi içerisinde 10-12 kişilik iki alt gruba ayrılarak odak grup görüşmelerine alınmıştır. Görüşmelere başlamadan önce öğrencilere çalışmanın amaç ve kapsamını belirten bir bilgilendirme metni okunmuştur. Bu metinde odak grup görüşmesinden elde edilecek verilerin katılımcıların kişisel bilgileriyle eşleştirilmeyeceği ve ders başarılarını etkilemeyeceği de belirtilmiştir. Odak grup görüşmeleri bir ses kayıt cihazı aracılığıyla kayıt altına alınmıştır. Bu ses kayıtları daha sonra araştırmacı tarafından metne dönüştürülmüştür. Transkript sürecinde kod isimleri (K1-K81) kullanılmıştır. Ayrıca kod isimlerinin başına eğitsel robot programlama grubu için "R", görsel programlama grubu için "C" harfi eklenmiştir. Odak grup görüşmeleri sonunda durum çalışması-1 için eğitsel robot ve nesne yönelimli görsel programlama gruplarından sırasıyla toplam 35 dk. 30 sn. ve 31 dk. 30sn.'lik; durum çalışması-2 için ise yine sırasıyla toplam 24dk. 02 sn. ve 20 dk. 36 sn.'lik ses kayıtları elde edilmiştir. Görüşmelerin tamamının metne dönüştürülmesi sonucunda A4 boyutunda tek satır aralıklı toplam 32 sayfalık bir belge elde edilmiştir.

Ders günlükleri/yansıma raporları. Çalışmada bilgisayımsal etkinliklere yönelik öğrenci görüşlerini almak amacıyla ders günlüğü uygulaması gerçekleştirilmiştir. Bu amaçla her hafta öğrencilerden yansıtıcı düşünmeyi destekleyici rehber sorular eşliğinde ders sürecindeki deneyimlerini yansıtmaları istenmiştir. Ders günlüğü şablonunun hazırlanmasında Brockbank ve McGill (2007) ile Bayrak ve Usluel'in (2011) rehber sorularından yararlanılmıştır. Sorular öğrencilerin derste neler öğrendiğine, gerçekleştirilen eğitsel etkinlik hakkındaki duygu ve düşüncelerine, programlama esnasında yolunda giden ve yolunda gitmeyen işlerin neler olduğuna yöneliktir (EK-Ç). Ders günlüklerinin tamamının bilgisayar ortamına aktarılması sonucunda durum çalışması-1 için eğitsel robot ve

nesne yönelimli görsel programlama grupları için A4 boyutunda sırasıyla 49 ve 43; durum çalışması-2'de ise 17 ve 13 sayfalık belgeler elde edilmiştir.

Sesli düşünme protokolü. Öğrencilerin bilgisayımsal problem çözme sürecinde ortaya çıkan bilişsel ve duyuşsal süreçlerinin daha derinlemesine incelenmesi amacıyla eğitsel robot uygulamasının yapıldığı grupta her hafta rastgele seçilen bir takımdan programlama etkinlikleriyle uğraşırken düşüncelerini sesli olarak ifade etmesi istenmiştir. Sesli düşünme olarak da adlandırılan ve bir tür bilişsel görev analizi olan bu teknik, sözelleştirme aracılığıyla örtük bilgileri açık hale getirmektedir.

Problem çözme formları. Öğrencilerin BD becerilerini sergilemelerini desteklemek amacıyla problem çözme sürecinde Polya'nın dört aşamalı problem çözme yöntemini benimsemeleri teşvik edilmiştir. Bu amaçla öğrencilerden programlama problemlerini Polya'nın rehber soruları eşliğinde çözmesi istenmiştir. Rehber sorular problemi anlama, plan/strateji geliştirme, stratejiyi uygulama ve çözümü geliştirme olmak üzere 4 bölüm etrafında toplanan 11 sorudan oluşmaktadır. Öğrencilere her bir bilgisayımsal problem için Polya'nın rehber sorularını içeren A4 formatında bir problem çözme formu dağıtılmıştır. Dağıtılan problem çözme formları bilgisayımsal problem çözme etkinlikleri sonunda toplanmıştır. Toplanan formlar tarayıcı aracılığıyla bilgisayar ortamına aktarılarak pdf formatına dönüştürülmüştür. Eğitsel robot ve nesne yönelimli görsel programlama grubu için sırasıyla 122 ve 143 sayfalık belgeler elde edilmiştir.

Verilerin Analizi

Veri analizi nicel ve nitel veri analizi olmak üzere iki safhada gerçekleştirilmiştir. Veri analizi öncesi öğrencilerden toplanan tüm veriler bilgisayar ortamına aktarılmıştır.

Nicel verilerin analizi. Çalışmada öğrencilerin programlama, formal ve betimsel soyutlama testleri ile problem çözme ve motivasyon ölçeklerinden elde ettikleri puanlar nicel veri analizi teknikleriyle analiz edilmiştir. Verilerin analizinde IBM SPSS (v. 21) programından yararlanılmıştır. Nicel analizlerde istatistiksel anlamlılık (p) değeri 0,05 olarak kabul edilmiştir. Nicel veri analizine geçmeden önce verilerin normal dağılım gösterip göstermediği incelenmiştir. Normallik sınamasında Q-Q ve P-P grafikleri, Shapiro-Wilk normal dağılım uygunluk testi, çarpıklık ve

basıklık değerlerine ilişkin Z istatistikleri kullanılmıştır. Normallik varsayımının karşılanması durumunda parametrik, karşılanmaması durumunda ise parametrik olmayan istatistiksel tekniklerden yararlanılmıştır. Uygulama öncesi gruplar arası puanlar bağımsız örneklem t ya da Mann-Whitney U testi ile karşılaştırılmıştır. Uygulama öncesi ve sonrası grup içi puan farklılıkları eşleştirilmiş örneklem t testi ya da onun parametrik olmayan karşılığı Wilcoxon işaretli sıralar testi ile incelenmiştir. Çalışmada değişkenler arası ilişkiler korelasyon analiziyle sınanmıştır.

Uygulama sonrası gruplar arası son test puanları, ön test-son test puanları arasında anlamlı bir ilişki olup olmadığına bağlı olarak ortalama fark analizi ya da kovaryans analiziyle incelenmiştir. Programlama başarısı test sonuçlarının değerlendirilmesinde ayrıca öğrencilerin akademik başarı ortalamaları da dikkate alınmıştır. İlişkilerin istatistiksel olarak anlamlı olduğu durumlarda gruplar arası son test puanları, ön test puanlarının ortak değişken olarak alındığı tek yönlü kovaryans analizi (normal dağılım gösteren durumlarda) ya da Quade sıra farkları kovaryans analizinden (normal dağılım göstermeyen durumlarda) yararlanılarak karşılaştırılmıştır. İlişkilerin anlamlı olmadığı durumlarda gruplar arası son test puanları bağımsız örneklem t ya da Mann-Whitney U testinden yararlanılarak karşılaştırılmıştır. Öğrencilerin öğretim materyallerine yönelik motivasyon ölçeği puanları, bu değişkene ait ön test uygulaması yapılmadığından bağımsız örneklem t testi ya da Mann Whitney U testi ile analiz edilmiştir. Etki büyüklüğü ölçümleri, grup ortalamaları farkı (Cohen d) ya da varyansa dayalı ilişki gücü (R-kare ve eta-kare) olmak üzere iki esasa göre hesaplanmıştır.

Nitel verilerin analizi. Veri analizi Nvivo 10 (QSR, ABD) nitel analiz yazılımı kullanılarak gerçekleştirilmiştir. Çalışmada toplanan veriler Glaser ve Strauss (1967) tarafından önerilen temellendirilmiş kuram (grounded theory) yaklaşımının sürekli karşılaştırma yöntemine benzer bir yolla analiz edilmiştir. Analiz sonucunda ulaşılan anlamlandırmalar yerden bitme (ground-up), yani veri temelli olarak keşfedici bir biçimde ortaya konulmuştur. Verilere aşina olmak adına analiz öncesi tüm veri kaynakları (odak grup görüşmelerine ait transkriptler, ders günlükleri, sesli düşünme kayıtları, problem çözme formları) dikkatlice gözden geçirilmiştir. Bu gözden geçirme sırasında, verilere karşı yüksek düzeyde genel bir izlenim elde edilmiştir. Daha zengin içerikli olması nedeniyle nitel veri analizine sırasıyla odak grup görüşmeleri ve ders günlükleriyle başlanmıştır. Bu veri kaynaklarının kodlanmasına

ilişkin istatistikler Tablo 43'te sunulmuştur. Daha sonra problem çözme formları ve sesli düşünme kayıtları analiz edilmiştir. Problem çözme formları görüntü analizi ile; sesli düşünme kayıtları ise ham ses kayıtları şeklinde analiz edilmiştir.

Tablo 43

Veri Kaynaklarının Nitel Analizi Sonucundaki Kodlanma İstatistikleri

Veri Kümesi	Durum Çalışması	Hafta No	Grup	Yeni Kod Sayısı	Kod Sayısı	Kodlanma sıklığı (f)
Odak Grup Görüşmeleri	Durum-1	-	Eğitsel Robot P.	136	136	408
		-	Görsel Prog.	57	108	379
	Durum-2	-	Eğitsel Robot P.	66	137	367
		-	Görsel Prog.	32	109	263
Ders Günlükleri	Durum-1	#1	Eğitsel Robot P.	50	130	405
			Görsel Prog.	22	92	207
		#2	Eğitsel Robot P.	13	101	312
			Görsel Prog.	8	113	323
		#4	Eğitsel Robot P.	11	108	339
			Görsel Prog.	4	77	236
		#5	Eğitsel Robot P.	3	85	207
			Görsel Prog.	4	77	185
		#6	Eğitsel Robot P.	2	83	190
			Görsel Prog.	1	35	70
		#7	Eğitsel Robot P.	2	90	188
			Görsel Prog.	2	72	158
		#8	Eğitsel Robot P.	6	105	250
			Görsel Prog.	2	61	122
		#9	Eğitsel Robot P.	4	90	193
			Görsel Prog.	2	58	160
		#10	Eğitsel Robot P.	3	67	159
			Görsel Prog.	4	82	193
		#11	Eğitsel Robot P.	2	104	224
			Görsel Prog.	1	91	215
		#12	Eğitsel Robot P.	4	95	235
			Görsel Prog.	0	66	186
	Durum-2	#3	Eğitsel Robot P.	3	86	203
			Görsel Prog.	1	46	82
		#4	Eğitsel Robot P.	1	65	139
			Görsel Prog.	0	64	121
		#5	Eğitsel Robot P.	1	97	257
			Görsel Prog.	0	70	138
		#6	Eğitsel Robot P.	1	94	246
			Görsel Prog.	1	75	168
		#7	Eğitsel Robot P.	1	64	174
			Görsel Prog.	0	51	116
		#8	Eğitsel Robot P.	0	53	105
			Görsel Prog.	0	30	56
			Toplam	450	3167	7979

Veri analizi sürecinde ilk olarak nitel kayıtlarda özel bir düşünceyi ya da ifadeyi adresleyen kelime ya da ifadeler kodlanmıştır (serbest kodlama). Böylece serbest kodlar oluşturulmuştur. Her bir veri kaynağının analizi sonrasında ortaya çıkan yeni kod sayısı incelenmiştir. Ortaya çıkan yeni kod sayısının önemli ölçüde azaldığı durumda serbest kodlar birbiriyle ilişkilendirilerek kategoriler ve temalar

oluşturulmuştur (aksiyel kodlama). Sonuç olarak, kategorilerin ve temaların bir listesi üretilmiştir. Ortaya çıkan kategori ve temalar, ortak ya da farklı noktaları tanımlamak için birbirleriyle ilişkili olarak incelenmiştir (seçici kodlama). Nitel analizde ortaya çıkan kod, kategori ve temalar, frekans ve yüzdeleriyle birlikte tablolar halinde sunulmuştur. Ayrıca doğrudan alıntılar ve temsiller şeklinde her bir kategorik yapıya yönelik olarak geniş tanımlamalara (thick descriptions) da yer verilmiştir. Nitel bulgular sunulurken katılımcılara K1'den, K81'e kadar kod isimleri verilmiştir. Güvenirliği arttırmak amacıyla araştırmacı tarafından ulaşılan kod, kategori ve temalar başka bir uzman tarafından daha kontrol edilerek kodlayıcılar arası tutarlık incelenmiştir.

Odak grup görüşmelerinin analiz edilmesinin ardından elde edilen serbest kodlar ders günlüklerinin analizi öncesinde birbiriyle ilişkilendirilmiştir. Bunun neticesinde 217 serbest kod, 17 kategori ve 7 tema altında toplanmıştır. Odak grup görüşmelerinden elde edilen kod şeması sonraki veri kümeleri için de kullanılıp, her veri kaynağında zenginleştirilmiştir. Ayrıca her bir veri kümesinin analizi sonrasında kod listesindeki kodlar yeniden incelenerek birbiriyle çakışan kodlar birleştirilmiştir. Bu esnada toplam 9 kod silinmiş ve 23 kod çifti birbiriyle birleştirilmiştir. Sonuç olarak veri kaynaklarının analizi sonucunda elde edilen toplam 450 serbest kod, 15 kategori ve 7 tema altında toplanmıştır. Bu temalar şunlardır:

1. Düşünme Biçimleri ve Bilişsel Unsurlar
2. Dersin Genel Çerçevesi ve Pedagojik Unsurlar
3. Problem Çözme ve BD Etkinlikleri
4. Öğrenci Çalışmaları
5. Destekler ve Engeller
6. Duyuşsal Boyut
7. Beklentiler ve Diğer Çıktılar

Serbest kodlara ait kodlama istatistikleri EK-D'de listelenmiştir. Ayrıca nitel veri analizi sonucunda ulaşılan kod, kategori ve temalar EK-E'de sunulmuştur.

Araştırmacının Rolü

Araştırmacı dersin öğretim elemanı olarak tüm gruplarda programlama etkinliklerine katılıp, bilgisayımsal problem çözme sürecinde öğrencilere rehberlik etmiştir. Ayrıca her iki durum çalışmasında da birer ders görevlisi çalışma boyunca araştırmacıya destek olmuştur. Nicel veriler deney ve kontrol gruplarında eş zamanlı olarak toplanmıştır. Nitel veriler de benzer bir anlayışla her iki grup için çok yakın zaman aralıklarında (en fazla 1-2 gün arayla) toplanmıştır.

İç ve Dış Geçerlik

Geçerlik, bir araştırmacının yaptığı çıkarımların uygunluğu, anlamlılığı, doğruluğu ve kullanışlılığını ifade eder (Fraenkel & Wallen, 2006). Çalışmada karma araştırma deseni benimsendiği için alınan geçerlik tedbirleri nicel ve nitel çalışmalar için ayrı şekilde sunulmuştur.

Nicel çalışmalarda iç geçerlik. Araştırmalarda iç geçerliği tehdit eden unsurların başında katılımcı özelliklerinin (yaş, cinsiyet, akademik başarı, sosyo-ekonomik düzey vb.) test grupları arasında farklılık göstermesi gelmektedir. Çalışmada katılımcıların seçilme ve test gruplarına atanma biçimi incelenecek olan değişkenler ile ilgili olarak katılımcıların birbirinden farklılaşmasına neden olabilir. Bu bazen katılımcı seçimi yanlılığı veya katılımcı karakteristiği tehdidi olarak anılır. Çalışmada katılımcıların amaçlı örnekleme yöntemiyle belirlenmiş olması iç geçerlik adına bir tehdit oluşturma potansiyeline sahiptir. Seçkisiz atama iç geçerliğe yönelik tehdit unsurlarını kontrol etmek için güçlü bir tekniktir (Fraenkel & Wallen, 2006). Çalışmada iç geçerliğe yönelik tehdit unsurlarını asgari düzeye indirmek adına uygulama süreci öncesinde katılımcılar deney ve karşılaştırma gruplarına kura yöntemiyle seçkisiz olarak atanmıştır. Ayrıca test gruplarındaki katılımcı özelliklerine ait betimsel istatistikler çalışmada ayrıntılı bir biçimde belirtilmiştir.

Bilimsel çalışmalarda, araştırmacılar tarafından kontrol edilemeyen bazı iç geçerlik tehditleri bulunmaktadır. Bunlardan en önemlisi katılımcı (denek) kaybıdır. Katılımcı kaybı, test gruplarının katılımcı karakteristikleri yönünden farklılaşmasına ve bu yolla test sonuçlarını etkilemesine yol açabilir. Çalışmada uygulama süresince her gruptan kaç katılımcının ayrıldığı ayrıntılı olarak belirtilmiştir. Ayrıca çalışmadan ayrılan katılımcılara ait veriler veri setinden çıkarılmıştır.

Uygulama ortamından ya da koşullarından kaynaklanabilecek farklılıkları gidermek adına tüm test gruplarında eğitsel etkinlikler bilgisayar laboratuvarında gerçekleştirilmiş ve bu etkinliklere aynı öğretim elemanı tarafından rehberlik edilmiştir. Ayrıca verilerin tamamı bilgisayar laboratuvarında toplanmıştır.

Nicel çalışmalarda dış geçerlik. Bu çalışma, bir MTAL'nin BT alanında öğrenim gören toplam 81 onuncu sınıf öğrencisiyle gerçekleştirilmiştir. Araştırmaya katılan katılımcıların gruplara atanmasında, yansızlık kuralına uyulması ve katılımcı sayısının nispeten fazla olması örneklemin evreni temsil edebilme yeteneğini arttırmaktadır. Araştırmalarda incelenen durum sayısı arttıkça bulguların ikna ediciliğinin ve genellenebilirliğinin arttığı belirtilmektedir (Merriam, 1998). Sonuçların genellenebilirliği ile ilgili bir kavram olan dış geçerliğin geliştirilmesi adına çalışmada çoklu deneysel çalışmalar farklı katılımcılarla tekrarlanmıştır.

Nitel çalışmalarda geçerlik. Nitel araştırmalarda geçerlik araştırmacının incelediği olguyu otantik koşullar altında ve olabildiğince yansız olarak gözlemlemesi anlamına gelmektedir (Kirk & Miller, 1986). Nicel araştırmalar geçerlik ve güvenirlik kavramları üzerinde daha fazla dururken, nitel araştırmalar daha ziyade itimat (trustworthiness) kavramı üzerinde durmaktadır (Bassey, 1999; Lincoln & Guba, 1985). İtimat kavramı bir araştırma çalışmasının, okuyucuları çalışma bulgularının dikkate değer olduğuna ikna edebilmesi şeklinde tanımlanmaktadır.

Çeşitleme (triangulation) nitel araştırmalarda itimadı sağlamak için kullanılan en yaygın yöntemlerdendir (Patton, 1980). Çeşitleme farklı örneklem, veri kaynakları ve analiz yöntemlerinin kullanılarak ulaşılan sonuçlara itimadı arttırmaya yönelik çabalar olarak tanımlanmaktadır (Yıldırım & Şimşek, 2008). Mevcut çalışmadan elde edilen çıkarımlara itimadı arttırmak amacıyla gerçekleştirilen müdahaleye ilişkin farklı veri toplama yöntemleri (görüşme, bilişsel görev analizi ve doküman inceleme) ve veri kaynakları (odak grup ve sesli düşünme kayıtları, öğrenci yansımaları, problem çözme formları) kullanılmıştır. Bu sayede veri kaynaklarının birleştirilmesi ve karşılıklı olarak birbirlerini ne ölçüde desteklediğinin ortaya çıkarılması amaçlanmıştır. Ek olarak, veri analiz sürecine farklı bir araştırmacı dâhil edilerek kodlayıcılar arası tutarlık incelenmiştir.

Bir bilimsel çalışmanın değerlendirilmesinde genellikle dört temel özellik dikkate alınmaktadır. Bunlar, doğruluk (truth value), uygulanabilirlik (applicability), kararlılık (consistency) ve tarafsızlıktır (neutrality). Bununla birlikte nicel ve nitel paradigmanın dünya görüşlerinin farklı olması önemlerinin değerlendirilmesi için farklı ölçütler gerektirmektedir. Lincoln ve Guba (1985) nitel çalışmaların değerlendirilmesinde doğruluk için inandırıcılık (credibility), uygulanabilirlik için aktarılabilirlik (transferability), kararlılık için tutarlılık (dependability) ve tarafsızlık için teyit edilebilirlik (confirmability) ölçütlerini önermiştir. Mevcut çalışmada inandırıcılığı sağlamak adına araştırmacı veri kaynaklarıyla (katılımcılar, incelenen durumlar) uzun süreli bir etkileşim içerisinde olmuştur. Katılımcıların öğretimsel müdahale karşısında reaksiyonlarını daha açık ve samimi olarak paylaşmaları için veri toplama sürecinde samimi bir hava oluşturulmaya çalışılmıştır. Ayrıca katılımcıların hiçbir tesir altında kalmadan düşüncelerini rahatça ifade etmesi için veri toplama süreci başında, elde edilecek niteliksel verilerin programlama temelleri dersi değerlendirme ölçütleriyle ilişkilendirilmeyeceği açıkça ifade edilmiştir. Araştırmacı, üzerinde çalıştığı olgulara ilişkin katılımcı bakış açılarını daha derinlemesine incelemek için araştırma alanına uzun süreli katılım (2-3 ay arası), haftalık olarak toplanan veriler yoluyla sürekli izleme ve çeşitleme (yöntem, kaynak ve araştırmacı) yöntemlerini kullanmıştır.

Aktarılabilirlik, incelenen duruma ait detaylı ve ayrıntılı betimlemelere yer vermeyi gerektirir (Lincoln & Guba, 1985; Merriam, 1998). Çalışmada uygulama ortamı, katılımcılar, veri toplama, işleme, analiz etme ve yorumlama süreçleri detaylı olarak ortaya konularak, sonuçlar ulaşılan verilerle ilişkilendirilerek sunulmuştur. Ayrıca çalışmada araştırmacının kendi çıkarımları ve araştırma içinde hangi rolleri aldığı açıkça belirtilmiştir. Elde edilen sonuçlar veri kaynaklarından yapılan alıntılarla desteklenmiştir. Araştırma raporunun iyi yazılmış olması yani araştırmanın bulgularının net bir şekilde ifade edilmesi araştırmanın başarısını etkileyen önemli bir faktördür (Baş & Akturan, 2008). Araştırmanın sonuç bölümünde öncelikle üretilen kategori ve temalara yer verilmiştir. Araştırma sonuçları okuyucunun kendi deneyimleriyle ilişkilendirebileceği şekilde raporlanmıştır. Ayrıca veri toplama sürecinde ulaşılan ham veriler farklı araştırmacılara sunulmak üzere kayıt altına alınmıştır.

Tutarlılık nitel bir çalışmanın kararlılığını yansıtır. Çoklu gerçekliklerin varlığı ve araştırma alanının değişen doğası, geleneksel güvenilirlik tekniklerinin nitel araştırmalarda uygulanmasını engelleyen tutarsızlıklar yaratmaktadır. Reigeluth ve Frick (1999) sonuçların genellenebilirliğinin arttırılması için durumsallıkları ya da koşulları tanımlama ve çalışmayı tekrarlama olmak üzere iki ölçütten bahsetmektedir. Bu çalışmada iki örnek durum ardışık olarak gerçekleşmiştir. Böylece durum çalışmalarından elde edilen çıkarımların tutarlılığı incelenip, durumsallıklar geniş bir perspektifte betimlenmiştir.

Teyit edilebilirlik nitel çalışmanın tarafsızlığına atıfta bulunmaktadır. Araştırmanın varsayım, yanlılık ve kurumsal yönlendirmelerden arındırılmasını sağlamak adına veri toplama ve analiz süreci bir uzman gözetiminde gerçekleştirilmiştir. Ayrıca veri analizi sürecinde ikinci bir araştırmacıdan destek alınarak araştırmacılar tarafından elde edilen çıkarımların tutarlılığı incelenmiştir. Bu amaçla iki veri kaynağı (durum çalışması-1'deki odak grup görüşmeleri) farklı bir araştırmacı tarafından da kodlanarak kodlayıcılar arası uzlaşı yüzdesi hesaplanmıştır. Buna göre kodlayıcılar arası uzlaşı yüzdesi %98,6 olarak bulunmuştur. Çalışmada verilerin toplanmasının ardından verilere ilişkin üye kontrollerinin yapılması sağlanmıştır. Ayrıca her hafta toplanan veriler (ders günlüğü gibi) aracılığıyla kanıt zincirinin oluşturulması sağlanmıştır. Durumlara özgü bağlamsal anlayışlar sağlayan nitel araştırma bulgularının, genellenebilirliği yani dış geçerliği nispeten yüksek olan ve değişkenler arası çoklu ilişkiler sağlayan nicel çalışma bulgularıyla birleştirilmesinin araştırma bulgularına daha rasyonel bir zemin hazırladığı düşünülmektedir.

Bölüm 4

Bulgular ve Yorumlar

Bu bölümde araştırma bulguları ve bu bulgularla ilgili değerlendirmeler yer almaktadır. Bulgular araştırma problemlerine göre organize edilmiştir. Bu kapsamda öncelikle başarı, formal ve betimsel soyutlama, problem çözme ve motivasyon verilerine ilişkin istatistiksel çözümlemelerden elde edilen bulgular sunulmaktadır. Sonrasında ise katılımcı deneyimlerinden yola çıkarak bilgisayımsal problem çözme sürecinde öne çıkan bilişsel unsurlara ve bu süreçte karşılaşılan güçlük ve engellere ilişkin nitel bulgular yer almaktadır.

Bilgisayımsal Düşünme Etkinliklerinin Öğrencilerin Programlama Başarısı Üzerine Etkileri

Çalışmada nesneye yönelik (karşılaştırma grubu) ve eğitsel robot (deney grubu) programlama grubundaki öğrencilerin programlamaya ilişkin ön bilgi düzeylerini saptamak ve uygulama öncesi gruplar arasında programlama bilgisi bakımından anlamlı bir farklılığın olup olmadığını belirlemek amacıyla ön test uygulaması yapılmıştır. Eğitim alanında yapılan araştırmalara getirilen eleştirilerden birisi kullanılacak istatistiksel yöntem ve tekniklere karar verilmesi sürecinde gerekli olan normallik testlerine çalışmada yer verilmemesidir. Demir, Saatçioğlu ve İmrol (2016) uluslararası 5 hakemli dergide 2013-2015 yılları arasında yayımlanan toplam 788 makale içerisinde normallik sınamalarının yapılmasının gerektiği 340 makaleden yalnızca 133 tanesinde (yaklaşık %40'ı) normallik varsayımının sınandığını bildirmiştir. Çalışmada programlama başarısı test puanlarının analizi öncesinde veriler normallik sınamasına tabi tutulmuştur. Ön test toplam puanlarının normallik varsayımları Q-Q ve P-P grafik testleri, Shapiro-Wilk normal dağılım uygunluk testi ve çarpıklık ve basıklık değerlerine ilişkin Z istatistikleriyle incelenmiştir. Tablo 44'te programlama test puanlarına ait çarpıklık ve basıklık değerleri ile normal dağılım uygunluk testine ilişkin test istatistikleri sunulmuştur. Verilere ilişkin P-P ve Q-Q grafiklerinin incelenmesi sonucunda, genel olarak gözlemsel değerler ile kuramsal değerlerin uyum içinde olduğu görülmüştür. Bununla birlikte Durum-2'de deney grubu ön test puanlarında uç değerlerin diğer gruplara nazaran nispeten fazla olduğu dikkati çekmiştir. Uç değerlerin ortalamaya göre dağılımlarının dengeli olduğu görülmüştür. Çarpıklık ve basıklık Z istatistiklerinin tamamına yakınının (Durum-2'de deney grubu son test puanları

hariç) %5 anlamlılık düzeyi için kritik değeri aşmadığı (±1,96) görülmüştür (Hair, Anderson, Tatham & Black, 1995).

Tablo 44

Programlama Başarısı Ön Test ve Son Test Puanlarına İlişkin Çarpıklık, Basıklık ve Normallik Testi İstatistikleri

		Grup	Çarpıklık		Basıklık		Saphiro-Wilk testi	
			İstatistik	Std. Hata	İstatistik	Std. Hata	W istatistiği	p değeri
Ön Test	Durum-1	Görsel P.	,906	,501	1,069	,972	,924	,106
		Eğ. Robot P.	,590	,512	,064	,992	,947	,318
	Durum-2	Görsel P.	,450	,524	-,102	1,014	,960	,567
		Eğ. Robot P.	,613	,501	1,241	,972	,920	,087
Son Test	Durum-1	Görsel P.	,461	,501	1,134	,972	,964	,594
		Eğ. Robot P.	,539	,512	,986	,992	,947	,325
	Durum-2	Görsel P.	,323	,524	,033	1,014	,962	,615
		Eğ. Robot P.	,810	,501	2,094	,972	,933	,155

Çalışmada örneklem boyutunun küçük (n<50) olmasından dolayı normal dağılım uygunluk testlerinden Saphiro-Wilk testi kullanılmıştır. Saphiro-Wilk testinden elde edilen W istatistiği verilerle ideal normal veriler arasındaki korelasyon olup, 0-1 arasında değer almaktadır. Bu istatistiğin 1'e yakın olması verilerin normal dağılıma yaklaştığını; 0'a yakın olması normallikten uzaklaştığını göstermektedir. Değişken dağılımı normalden farklı değilse W testi anlamlılık değeri ,05'ten büyüktür. Çalışmada programlama başarısı ön test puanlarına ilişkin W testi değeri Durum-1'de deney ve karşılaştırma grubu için sırasıyla ,947 (p>,05) ve ,924 (p>,05); Durum-2'de ise ,920 (p>,05) ve ,960 (p>,05) bulunurken, programlama başarısı son test puanlarına ilişkin W testi değeri Durum-1 için deney ve karşılaştırma grubu için sırasıyla ,947 (p>,05) ve ,964 (p>,05); Durum-2 için ise ,933 (p>,05) ve ,962 (p>,05) olarak bulunmuştur. Bu sonuçlar programlama başarısı durum çalışmaları ve veri setinin gruplara göre normal dağılıma uyum sağladığını göstermektedir.

Tablo 45'te durum-1 ve durum-2 için uygulama öncesi öğrenenlerin programlama başarısı ön test sonuçlarına ilişkin bağımsız t testi sonuçları sunulmuştur. Buna göre her iki durum çalışmasında da uygulama öncesi deney ve karşılaştırma gruplarının programlama bilgisinin düşük olduğu ve gruplar arasında

programlama başarısı test puanları açısından anlamlı bir farklılık olmadığı belirlenmiştir (durum-1: t_{39}=,350, p>,05; durum-2: t_{38}=-,052, p>,05).

Tablo 45

Uygulama Öncesi Gruplar arası Programlama Başarısı Ön Test Puanlarının Karşılaştırılması

	Test	Gruplar	N	$\bar{X}$	SS	sd	t	p
Durum çalışması-1	Prog. Baş. Ön test	Görsel Prog	21	5,952	3,074	39	,350	,728
		Eğ. Robot Prog.	20	5,650	2,390			
Durum çalışması-2	Prog. Baş. Ön test	Görsel Prog	19	6,579	2,411	38	-,052	,958
		Eğ. Robot Prog.	21	6,619	2,418			

Not: Programlama başarı testinden alınabilecek en yüksek puan 23'tür.

Her iki durum çalışmasında da programlama eğitimi aracılığıyla gerçekleştirilen bilgisayımsal düşünme (BD) etkinlikleri sonrasında grup içi programlama başarısı son test puanlarında anlamlı bir artış görülmüştür (p<,001) (Tablo 46).

Tablo 46

Uygulama Öncesi ve Sonrası Grup içi Programlama Başarı Test Puanlarının Değişimi

	Grup	Testler	N	$\bar{X}$	SS	sd	t	p	Cohen d	η^2
Durum Çalışması-1	Görsel Prog.	Ön test	21	5,952	3,074	20	6,921	,000	1,510	,705
		Son test	21	9,904	4,380					
	Eğ. Robot Prog.	Ön test	20	5,650	2,390	19	8,027	,000	1,795	,772
		Son test	20	10,250	3,385					
Durum Çalışması-2	Görsel Prog.	Ön test	19	6,579	2,411	18	6,040	,000	1,386	,669
		Son test	19	10,579	3,906					
	Eğ. Robot Prog.	Ön test	21	6,619	2,418	20	7,147	,000	1,559	,718
		Son test	21	11,524	2,676					

Her iki grupta da uygulama sonrası öğrenenlerin programlama test puanlarının yaklaşık 2 kat artarak düşük seviyeden orta seviyeye yaklaştığı görülmektedir. Karşılaştırma grubunda BD etkinlikleri sonrası öğrenenlerin son test puan ortalaması durum-1'de 5,952'ten 9,904'e (t_{20}=6,921, p<,001); durum-2'de

6,579'dan 10,579'a (t_{18}=6,040, p<,001) yükselmiştir. Benzer şekilde deney grubunda da BD etkinlikleri sonrası öğrenenlerin test puanlarında anlamlı artış olmuştur. Buna göre deney grubu son test puanları durum-1'de 5,650'den 10,250'ye (t_{19}=8,027, p<,001) ; durum-2'de 6,619'dan 11,524'e (t_{20}=7,147, p<,001) yükselmiştir. Ancak artışların etkililiğini belirlemek için sadece t testi sonuçlarına bakmak, yani anlamlılık düzeyini incelemek yeterli değildir. Etki büyüklüğü örneklem büyüklüğüne duyarlı olan istatistiksel anlamlılık değerinin nicel olarak desteklenmesinde raporlanması tavsiye edilen istatistiksel değerlerden birisidir (Cohen, 1990, 1994; Kirk, 1996, 2001; Özsoy & Özsoy, 2013; Vacha-Haase, 2001). Etki büyüklüğü, iki karşılaştırma arasındaki standartlaştırılmış ortalama farktır. Bir anlamda iki gruptan elde edilen sonuçlar arasındaki farkın yani ortalama puan farklılıklarının boyutunun (büyüklüğünün) belirlenmesidir. Etki büyüklüğü özellikle yokluk hipotezinin reddedildiği durumlarda belirli öğretimsel koşulların ya da müdahalelerin etkililiğinin belirlenmesinde oldukça değerlidir. Etki büyüklüğü ölçümleri genellikle grup ortalamaları farkı (Cohen d, Glass g, Hedge d) ya da varyansa dayalı ilişki gücü (R-kare (R^2), eta-kare (η^2)) olmak üzere iki esasa göre hesaplanmaktadır. Grup içi ön test ve son test ortalama puanlarının karşılaştırılmasında hem cohen d hem de η2 istatistikleri kullanılmıştır. Karşılaştırma grubu ön test-son test puanlarına ilişkin hesaplanan cohen *d* ve η^2 değerleri durum çalışması-1 için sırasıyla 1,510 ve ,705; durum çalışması 2 için sırasıyla 1,386 ve ,669 olarak bulunmuştur. Bu sonuçlar (cohen d>0,8 ve η^2>0,14) nesne tabanlı görsel programlama etkinliklerinin öğrenenlerin programlama başarılarının yükselmesinde oldukça büyük (geniş) bir etkiye sahip olduğunu ve öğrenenlerin programlama başarısı test puanlarında (ön test-son test) gözlenen varyansın durum çalışması-1 için %70,5'inin; durum çalışması-2 için yaklaşık %67'sinin bu etkinliklere bağlı olarak ortaya çıktığını göstermektedir. Eğitsel robot programlama etkinliklerine katılan gruplarda ön test-son test puan farklılıklarına ilişkin cohen *d* ve η^2 değerleri durum çalışması-1 için sırasıyla 1,795 ve ,772; durum çalışması-2 için sırasıyla 1,559 ve ,718 olarak hesaplanmıştır. Nesne tabanlı programlama grubuna benzer şekilde oldukça yüksek bir etki büyüklüğüne vurgu yapan bu sonuçlara göre öğrenenlerin programlama başarısı test puanlarında gözlenen varyansın durum çalışması-1 için %77'si; durum çalışması-2 için ise yaklaşık %72'si eğitsel robotlarla yapılan programlama etkinliklerine bağlı olarak ortaya çıkmıştır. Tablo 46 incelendiğinde her iki durum çalışmasında da görsel programlama grubuyla

karşılaştırıldığında eğitsel robot programlama grubundaki öğrenenlerin son test puanlarının daha yüksek olduğu görülmektedir.

Programlama eğitimi ve BD etkinlikleri için kullanılan yöntemin programlama başarısına olan etkisini incelemek için ön test puanları ve akademik başarı ortalamalarının ortak değişken, öğrenme yöntemi/öğrenme aracının sabit faktör olarak alındığı tek yönlü kovaryans (ANCOVA) analizi kullanılmıştır. ANCOVA, tarihsel açıdan bakıldığında farklı araştırma alanlarında farklı soruları adreslemek için geliştirilmiş olan regresyon ve varyans analizlerinin (ANOVA) birleştirilmesiyle oluşan bir analizdir (Rutherford, 2001). Kovaryans analizi gruplar arasındaki regresyonları eşitlemesi sayesinde hata varyansını düşürerek F değerinin ve model gücünün artmasına katkı sağlar (Keskin, 2010). Bu durum özellikle örneklem boyutunun veya etki büyüklüğünün küçük olduğu durumlarda oldukça yararlıdır. Varsayımların sağlanması durumunda ANCOVA güçlü bir istatistiksel yöntemdir. Ortak değişken ile bağımlı değişken arasında doğrusal bir ilişki bulunması ANCOVA testinin temel varsayımlarındandır. ANCOVA özellikle ortak değişken ile bağımlı değişken arasındaki ilişkinin yüksek olduğu (r>,30) durumlar için etkilidir. Söz konusu ilişkinin konu alanı için belirli bir kuramsal dayanağının olması esastır. Ayrıca birden fazla ortak değişken bulunuyorsa bu değişkenler arası ilişkinin yüksek düzeyde (r>,80) olmaması gerekmektedir (Keskin, 2010). Analiz öncesi ortak değişkenlerin her biri ile bağımlı değişken arasındaki ilişkinin doğrusallığı saçılım grafiği ile kontrol edilmiştir. Ayrıca bağımlı değişken ile ortak değişkenler arasındaki ilişkinin gücü korelasyon analiziyle belirlenmiştir (Tablo 47).

Tablo 47

Akademik Başarı Ortalaması, Programlama Başarısı Ön Test ve Programlama Başarısı Son Test Puanları Arasındaki İlişkilerin İncelenmesi

			Durum-1		
			Akademik Başarı	Programlama Ön test puanları	Programlama Son test puanları
Durum-2	Akademik Başarı	r	-	,577**	,692**
		p		,00	,00
		N		41	41
	Programlama Ön test puanları	r	,221	-	,749**
		p	,171		,00
		N	40		41
	Programlama Son test puanları	r	,433**	,475**	-
		p	,00	,00	
		N	40	40	

*Not: **p<.01*

Korelasyon analizi sonuçları incelendiğinde akademik başarı ile programlama son test puanları arasında durum-1'de yükseğe yakın düzeyde (r=,692, p<,01), durum-2'de ise orta düzeyde (r=,433, p<,01) pozitif ve anlamlı bir ilişki olduğu belirlenmiştir. Benzer şekilde programlama ön test ile son test puanları arasında durum-1'de yüksek düzeyde (r=,749, p<,01); durum-2'de ise orta düzeyde (r=,475; p<,01) pozitif ve anlamlı bir ilişki saptanmıştır. Ayrıca iki ortak değişken arasındaki ilişkinin durum-1'de orta düzeyde pozitif (r=,577, p<,00); durum-2'de ise ilişkisiz (r=,221, p>,05) olduğu görülmektedir. Ortak değişkenler arasında çok yüksek bir ilişkinin olmadığını (r<,80) gösteren bu sonuçtan yola çıkılarak ortak değişkenlerin modelde kalması uygun görülmüştür.

ANCOVA öncesi grup varyanslarının homojenliği Levene testi ile incelenerek, varyansların homojen olduğu doğrulanmıştır (Durum-1 için F=,938, p>,05; Durum-2 için F=1,584, p>,05). Ayrıca iki grup için de regresyon doğrularının eğimlerinin aynı olduğu yani regresyon homojenliği varsayımının da sağlandığı tespit edilmiştir. Öğrenenlerin programlama başarısı test puanlarına ilişkin düzeltilmiş ortalama sonuçları ve ANCOVA test istatistikleri sırasıyla Tablo 48 ve Tablo 49'da sunulmuştur.

Tablo 48

Programlama Başarısı Son Test Puanlarına İlişkin Grup Ortalama ve Düzeltilmiş Ortalama Puanları

	Grup	N	Ortalama	Düzeltilmiş Ortalama
Durum-1	Görsel Prog.	21	9,9048	9,740
	Eğ. Robot Prog.	20	10,2500	10,423
Durum-2	Görsel Prog.	19	10,5789	10,530
	Eğ. Robot Prog.	21	11,5238	11,568

Ön test puanları ve akademik başarı ortalamalarına göre düzeltilmiş son test puanları incelendiğinde her iki durum çalışmasında da deney grubundaki öğrencilerin daha başarılı olduğu görülmektedir. Buna karşın, her iki durum çalışması için de kovaryans analizinden elde edilen sonuçlar, programlama başarısı son test ortalama puanları arasında programlama yöntemine göre istatistiksel olarak anlamlı bir farklığın olmadığını göstermiştir (Durum-1 için $F_{1\text{-}37}$=,884, p>,05; Durum-

2 için F_{1-36}=1,422, p>,05). Etki büyüklüğünün düşük düzeyde olduğu görülmüştür (Durum-1'de Ŋ² =,023; Durum-2'de Ŋ² =,038).

Tablo 49

Deney ve Karşılaştırma Grubu Öğrencilerinin Programlama Başarısı Son Test Puanlarının ANCOVA ile Karşılaştırılması

	Varyans Kaynağı	Kareler Toplamı	sd	Kareler Ortalaması	F	p	Etki Büyük. (η^2)
Durum-1	Prog_Ontest (Reg)	112,367	1	112,367	20,875	,000	,361
	Akademik B. (Reg)	61,089	1	61,089	11,349	,002	,235
	Grup	4,760	1	4,760	,884	,353	,023
	Hata	199,160	37	5,383			
	Toplam	602,780	40				
Durum-2	Prog_Ontest (Reg)	63,621	1	63,621	8,439	,006	,190
	Akademik B. (Reg)	50,516	1	50,516	6,701	,014	,157
	Grup	10,716	1	10,716	1,422	,241	,038
	Hata	271,390	36	7,539			
	Toplam	426,775	39				

Not: Durum 1 için R^2 = ,670 (Düzeltilmiş R^2= ,643); Durum 2 için R^2= ,364 (Düzeltilmiş R^2= ,311)

Özetle; her iki durum çalışmasında da uygulama öncesinde test grupları arasında programlama başarısı ön test puanları açısından anlamlı bir farklılık bulunmamaktadır. Her iki durum çalışmasında da programlama aracılığıyla gerçekleştirilen BD etkinlikleri sonrasında grup içi programlama başarısı son test puanlarında anlamlı bir artış bulunmuştur. Programlama başarısı son test puanları gruplar arasında yani programlama modalitesine göre anlamlı bir farklılık göstermemiştir.

Bilgisayımsal Düşünme Etkinliklerinin Öğrencilerin Soyutlama Becerisi Üzerine Etkileri

Bir diğer araştırma sorusu programlama yoluyla gerçekleştirilen BD etkinliklerinin öğrenenlerin soyutlama becerisine olan etkilerinin neler olduğudur. Çalışmada programlama eğitiminde kullanılan eğitsel robotların görsel programlama aracıyla karşılaştırıldığında öğrencilerin soyutlama becerisi üzerinde anlamlı bir farklılık yaratıp yaratmadığı da sorgulanmıştır. Daha önce de belirtildiği gibi bu çalışmada soyutlama formal ve betimsel olmak üzere iki alt boyutta ele alınmıştır. Bulgular bu iki alt boyuta göre paylaşılmıştır. İlk olarak formal soyutlamaya ilişkin bulgular paylaşılmıştır.

Bilgisayımsal düşünme etkinliklerinin formal soyutlama becerisine etkisi. Bilgisayar bilimleri soyut olarak karakterize edilen çeşitli öğeler ve etkinlikler yönünden zengin referanslar içermektedir (Colburn & Shute, 2007). Programlama farklılaşan seviyelerde veri soyutlaması ve prosedürel soyutlamaya olanak tanıyan bir etkinliktir. Formal soyutlama; temelde yatanı kavramak ya da çıkarmak ve daha doğrudan ya da basitleştirilmiş bir bakış açısı yakalamak amacıyla sembolik durumlar ve yapılar hakkında akıl yürütme yeteneğidir (Hill vd., 2008). Bu süreç/işlem sadeleştirme yapmak ve dikkati bir yöne toplamak için ayrıntıları ortadan kaldırmayı içermektedir. Formal soyutlama, test maddeleri alakasızlıklar altında görülebilecek bir yapı gerektiren problem çözme maddelerinden oluşan bir araçtır. Bu nedenle test soruları daha kesin ve daha az kesin seçenekler barındırmaktadır.

Öğrencilerin formal soyutlama test puanlarını analiz etmeden önce verilerin normal dağılım sergileyip sergilemediği kontrol edilmiştir. Normallik sınaması Q-Q ve P-P grafikleri, çarpıklık ve basıklık katsayıları ve bunlara ilişkin Z istatistikleri ve normal dağılım uyum iyiliği testi aracılığıyla gerçekleştirilmiştir. Q-Q ve P-P saçılım grafikleri incelendiğinde gözlenen değerle beklenen değerler arasında önemli sapmalar olmadığı tespit edilmiştir. Çarpıklık ve basıklık katsayılarının ±2 arasında değiştiği görülmüştür (Tablo 50).

Tablo 50

Formal Soyutlama Ön Test ve Son Test Puanlarına İlişkin Çarpıklık, Basıklık ve Normallik Testi İstatistikleri

			Çarpıklık		Basıklık		Saphiro-Wilk testi	
		Grup	İstatistik	Std. Hata	İstatistik	Std. Hata	W istatistiği	p değeri
Ön Test	Durum-1	Görsel P.	,806	,501	,087	,972	,864	,007
		Eğ. Robot P.	,725	,512	,190	,992	,882	,020
	Durum-2	Görsel P.	-,076	,524	-1,161	1,014	,881	,022
		Eğ. Robot P.	,876	,501	1,827	,972	,843	,003
Son Test	Durum-1	Görsel P.	,528	,501	,590	,972	,898	,032
		Eğ. Robot P.	,321	,512	-,577	,992	,876	,015
	Durum-2	Görsel P.	-,075	,524	-,012	1,014	,862	,011
		Eğ. Robot P.	,996	,501	1,192	,972	,803	,001

Çarpıklık ve basıklık göstergelerine bağlı Z istatistiklerinin de tamamına yakınının kritik değer olan ±1,96 sınırları içerisinde olduğu saptanmıştır (Durum-2 eğitsel robot programlama grubu formal soyutlama son testi çarpıklık değerine ilişkin Z istatistiği hariç). Bununla birlikte Saphiro-Wilk testi anlamlılık değerleri incelendiğinde formal soyutlama ön test ve son test puanlarının gruplara göre normal dağılım sergilemediği saptanmıştır (p<,05). Bu nedenle test puanlarının analizinde parametrik olmayan testler kullanılmıştır. Grup içi karşılaştırmalar için Wilcoxon İşaretli Sıralar testi, gruplar arası karşılaştırmalar için ise Mann-Whitney U testi kullanılmıştır. Ayrıca son test puanlarının karşılaştırılmasında Quade (1967) tarafından kovaryans analizinin parametrik olmayan karşılığı olarak önerilen kovaryans sıra analizinden de yararlanılmıştır. Gruplar arasında anlamlı bir farklılık olması durumunda etki büyüklüğü de hesaplanmıştır.

Nesne yönelimli görsel programlama ve eğitsel robot programlama eğitimine katılan öğrencilerin uygulama öncesi formal soyutlama testinden aldıkları puanlara ilişkin test istatistikleri Tablo 51'de sunulmuştur. Uygulama öncesi gruplar arasında formal soyutlama becerisi açısından anlamlı bir farklılığın olmadığı belirlenmiştir (durum-1: U=196,5, p>,05; durum-2: U=194,5, p>,05).

Tablo 51

Uygulama Öncesi Gruplar Arası Formal Soyutlama Ön Test Puanlarının Karşılaştırılması

	Test	Gruplar	N	Sıra Ortalaması	Sıra Toplamı	U	p
Durum çalışması-1	Formal Soyutlama Ön Testi	Görsel P.	21	20,36	427,50	196,5	,712
		Eğ. Robot P.	20	21,68	433,50		
Durum çalışması-2	Formal Soyutlama Ön Testi	Görsel P.	19	20,24	384,50	194,5	,887
		Eğ. Robot P.	21	20,74	435,50		

Öğrencilerin uygulama sonrası formal soyutlama puanlarındaki değişim eşleştirilmiş t testinin parametrik olmayan karşılığı olan Wilcoxon işaretli sıralar testi ile analiz edilmiştir (Tablo 52). Buna göre, BD etkinlikleri sonrası ön test puanlarıyla karşılaştırıldığında genellikle formal soyutlama son test puanını arttıran öğrenci sayısı azaltan öğrenci sayısından fazla olmuştur (Durum-1 için deney grubu hariç). Ancak, nispeten düşük de olsa pozitif puanlar yönündeki bu farklılığın sadece

Durum-2'deki deney grubunun ön test-son test puanları arasında anlamlı olduğu belirlenmiştir (Z=2,00, p<,05).

Tablo 52

Uygulama Öncesi ve Sonrası Grup İçi Formal Soyutlama Test Puanlarının Değişiminin Wilcoxon İşaretli Sıralar Testi ile Analizi

	Grup	Son Test-Ön Test	n	Sıra Ortalaması	Sıra Toplamı	Z	p	r
Durum Çalışması-1	Görsel Prog.	Negatif Sıra	7	8,00	56	,688	,491	,106
		Pozitif Sıra	9	8,89	80			
		Eşit	5					
	Eğitsel Robot Prog.	Negatif Sıra	7	7,93	55,50	-,193	,847	-,030
		Pozitif Sıra	7	7,07	49,50			
		Eşit	6					
Durum Çalışması-2	Görsel Prog.	Negatif Sıra	3	2,50	7,50	,649	,516	,105
		Pozitif Sıra	3	4,50	13,50			
		Eşit	13					
	Eğitsel Robot Prog.	Negatif Sıra	0	,00	,00	2,000	,046*	,309
		Pozitif Sıra	4	2,50	10,00			
		Eşit	17					

*Not: *p<,05*

Etki büyüklüğünün raporlanması, özellikle de istatistiksel anlamlılığın olduğu durumlarda, parametrik testler için ne kadar önemliyse parametrik olmayan testler için de bir o kadar önemlidir (Leech & Onwuegbuzie, 2002). Cohen d etki büyüklüğü ölçümleri, normallik ve varyansların heterojenliği ihlallerinden olumsuz etkilendiğinden, bu varsayımları ihlal eden verilerle birlikte kullanılması tavsiye edilmemektedir. Bu nedenle etki büyüklüğü için sadece r istatistiği kullanılmıştır. Bu gruba ait etki büyüklüğünün de orta derecede olduğu görülmektedir (r=,309). Bu sonucun ortaya çıkmasında durum-1 ile karşılaştırıldığında durum-2'de daha yüksek akademik başarıya sahip öğrencilerin bulunmasının etkisi olabileceği düşünülmektedir. Öte yandan, teste ilişkin hesaplanan istatistiksel güç değeri (power=0,58<0,80) düşük örneklem boyutundan (n=21) dolayı bir sınırlılık yaratmaktadır. Durum-2'de deney grubu ön test-son test puanları arasında yapılan karşılaştırmanın etki büyüklüğüne dayanarak (d=0,4294) 0,80'lik bir istatistiksel güç elde etmek için örneklem boyutunun 37 olması gerekmektedir. Her ne kadar sıra farklılığının anlamlı olmadığı durumlarda önemli bir anlam ifade etmese de diğer

gruplar için etki büyüklüğü istatistikleri incelendiğinde etki büyüklüğünün oldukça düşük olduğu, yani yapılan öğretimsel müdahalelerin öğrencilerin formal soyutlama puanlarının geliştirilmesinde oldukça düşük bir etkiye sahip olduğu göze çarpmaktadır.

Son olarak BD etkinlikleri sonrası deney ve karşılaştırma grubundaki öğrencilerin formal soyutlama son test puanları karşılaştırılmıştır. Karşılaştırma öncesinde akademik başarı, formal soyutlama ön test ve son test puanları arasındaki ilişkiler Spearman sıra farkları korelasyon analiziyle incelenmiştir (Tablo 53).

Tablo 53

Akademik Başarı Ortalaması, Formal Soyutlama Ön Test ve Formal Soyutlama Son Test Puanları Arasındaki İlişkilerin İncelenmesi

			Akademik Başarı	Durum-1 Formal Soyutlama Ön Test Puanları	Formal Soyutlama Son Test Puanları
Durum-2	Akademik Başarı	r_s		-,147	,239
		p	-	,361	,133
		N		41	41
	Formal Soyutlama Ön Test Puanları	r_s	,167		,216
		p	,302	-	,175
		N	40		41
	Formal Soyutlama Son Test Puanları	r_s	,074	,624**	
		p	,649	,000	-
		N	40	40	

*Not: **p<,01*

Tablo 53 incelendiğinde Durum-1'de formal soyutlama ön ve son test puanları arasında ilişki olmadığı belirlenmiştir (r_s=,216, p>,05). Buna karşın Durum-2'de formal soyutlama ön test ve son test puanları arasında orta derecede pozitif ve anlamlı bir ilişki olduğu görülmektedir (r_s=,624, p<,01). Her iki durum çalışmasında da akademik başarının formal soyutlama ön test (Durum-1: r_s=-,147, p>,05; Durum-2: r_s=,167, p>,05) ve son test puanlarıyla (Durum-1: r_s=,239, p>,05; Durum-2: r_s=,074, p>,05) anlamlı bir ilişkisinin olmadığı belirlenmiştir. Dolayısıyla akademik başarı değişkeninin formal soyutlama test puanlarının analizinde ortak değişken olarak kullanılmasına gerek duyulmamıştır. Bununla birlikte formal soyutlama ön test puanları ortak değişken olarak parametrik olmayan analizlere dâhil edilmiştir. Formal soyutlama son test puanları karşılaştırıldığında hem durum-1 hem de durum-2'de deney ve karşılaştırma grubu arasında anlamlı bir farklılığın olmadığı görülmüştür (durum-1: U=184,00, p>,05; durum-2: U=183,00, p>,05) (Tablo 54).

Tablo 54

Uygulama Sonrası Gruplar Arası Formal Soyutlama Son Test Puanlarının Karşılaştırılması

	Test	Gruplar	N	Sıra Ortalaması	Sıra Toplamı	U	p
Durum çalışması-1	Formal Soyutlama Son Testi	Görsel P.	21	22,24	467,00	184,00	,473
		Eğ. Robot P.	20	19,70	394,00		
Durum çalışması-2	Formal Soyutlama Son Testi	Görsel P.	19	19,63	373,00	183,00	,626
		Eğ. Robot P.	21	21,29	447,00		

Sıra ortalamalarına bakıldığında Durum-1'de uygulama sonrası formal soyutlama test puanlarının karşılaştırma grubu lehine; Durum-2'de ise deney grubu lehine yüksek olduğu görülmektedir. Ancak bu farklılıklar istatistiksel olarak anlamlı değildir. Ön test puanlarının ortak değişken olarak analize dâhil edildiği Quade sıra farkları kovaryans analiz sonuçlarına göre de formal soyutlama son test puanları açısından gruplar arasında anlamlı bir farklılık olmadığı bulunmuştur (Durum-1 için Quade F=0,661, p=,421>,05; Durum-2 için F=0,255, p=,616>,05).

Özetle; her iki durum çalışmasında da deneysel uygulama öncesi test grupları arasında formal soyutlama puanları açısından anlamlı bir farklılık yoktur. Durum-1'de programlama eğitimi aracılığıyla gerçekleştirilen BD etkinlikleri sonrasında grup içi formal soyutlama beceri puanları (ön-son test arası) arasında anlamlı bir farklılık görülmezken; Durum-2'de deney grubundaki öğrencilerin formal soyutlamaya ilişkin ön ve son test puanlarında son test lehine anlamlı bir artış bulunmuştur. Durum çalışması 1 ve 2'de formal soyutlama son test puanlarına ilişkin olarak test grupları arasında anlamlı bir farklılık bulunmamıştır.

Bilgisayımsal düşünme etkinliklerinin betimsel soyutlama becerisine etkisi. Betimsel soyutlama ortak temeli ya da esası tanımlamak için genelleme sürecine tekabül etmektedir. Betimleyici soyutlama en önemli karakteristiklerin farkına varma, bu karakteristiklerden yola çıkarak anlamlı ancak kısaltılmış genelleyici modeller oluşturma yeteneğidir. Bu; önemli noktaları belirlemeyi, açık ve doğrudan analiz üzerine odaklanmayı, merkeze inmeyi ve temel esasları algılamayı içerir. Benzerlikler ve karşıtlıklar nedeniyle bir varlık topluluğu gruplandırıldığında betimleyici soyutlama görünür birlikteliklerin ve/veya farklılıkların görünür olmasını

sağlar. Formal soyutlama daha çok problem çözme ile ilgiliyken betimsel soyutlama dil manipülasyonundaki uzmanlaşmalardır. Çalışmada kullanılan betimsel soyutlama testi en anlamlı niteliklerin farkına varılmasının ve açıklanmasının gerektiği sorulardan oluşmaktadır. Öğrencilerin betimsel soyutlama test puanları incelenmeden önce veri setinin normal dağılım varsayımını sağlayıp sağlamadığı sınanmıştır (Tablo 55).

Tablo 55

Betimsel Soyutlama Ön Test ve Son Test Puanlarına İlişkin Çarpıklık, Basıklık ve Normallik Testi İstatistikleri

			Çarpıklık		Basıklık		Saphiro-Wilk testi	
		Grup	İstatistik	Std. Hata	İstatistik	Std. Hata	W istatistiği	p değeri
Ön Test	Durum-1	Görsel P.	1,270	,501	,170	,972	,655	,000
		Eğ. Robot P.	1,283	,512	,542	,992	,675	,000
	Durum-2	Görsel P.	,336	,524	-,821	1,014	,802	,001
		Eğ. Robot P.	,145	,501	-,434	,972	,792	,000
Son Test	Durum-1	Görsel P.	,453	,501	-1,095	,972	,791	,000
		Eğ. Robot P.	,186	,512	-1,308	,992	,809	,001
	Durum-2	Görsel P.	,314	,524	-,506	1,014	,784	,001
		Eğ. Robot P.	,660	,501	,417	,972	,833	,002

Çarpıklık ve basıklık katsayılarının ±2 arasında değiştiği görülmüştür. Ayrıca çarpıklık ve basıklık göstergelerine bağlı Z istatistiklerinin de tamamına yakınının kritik değer olan ±1,96 sınırları içerisinde olduğu saptanmıştır (Durum-1'de her iki grup betimsel soyutlama ön testi çarpıklık katsayılarına ilişkin Z istatistikleri hariç). Bununla birlikte Saphiro-Wilk testi anlamlılık değerleri incelendiğinde betimsel soyutlama ön test ve son test puanlarının gruplara göre normal dağılım sergilemediği saptanmıştır ($p<,05$). Bu nedenle test puanlarının analizinde parametrik olmayan testler kullanılmıştır. Grup içi karşılaştırmalar için Wilcoxon İşaretli Sıralar testi; gruplar arası karşılaştırmalar için ise Mann-Whitney U testi kullanılmıştır. Ayrıca son test puanlarının karşılaştırılmasında kovaryans analizinin parametrik olmayan karşılığı olarak önerilen kovaryans sıra analizinden de yararlanılmıştır. Gruplar arasında anlamlı bir farklılık olması durumunda etki büyüklüğü de hesaplanmıştır. Sonuçlar her iki durum çalışmasında da uygulama

öncesi gruplar arasında anlamlı bir farklılık olmadığını ortaya koymuştur (durum-1 için U=210,00, p>,05; durum-2 için U=187,50, p>,05) (Tablo 56). Durum-1'de deney ve karşılaştırma grubundaki öğrencilerin sıra ortalaması eşit seviyedeyken; Durum-2'de deney grubu öğrencilerinin sıra ortalaması kontrol grubundan daha yüksek seviyededir.

Tablo 56

Öğrencilerin Betimsel Soyutlama Ön Test Puanlarının Mann-Whitney U Testi ile Karşılaştırılması

	Test	Gruplar	N	Sıra Ortalaması	Sıra Toplamı	U	p
Durum-1	Betimsel Soyutlama Ön Testi	Görsel P.	21	21,00	441,00	210,00	1,00
		Eğ. Robot P.	20	21,00	420,00		
Durum-2	Betimsel Soyutlama Ön Testi	Görsel P.	19	19,87	377,50	187,50	,719
		Eğ. Robot P.	21	21,07	442,50		

Test gruplarının betimsel soyutlama ön test-son test puanları arasındaki sıra farklılıkları Tablo 57'de özetlenmiştir.

Tablo 57

Uygulama Öncesi ve Sonrası Grup İçi Betimsel Soyutlama Test Puanlarının Değişiminin Wilcoxon İşaretli Sıralar Testi ile Analizi

	Grup	Son Test-Ön Test	n	Sıra Ortalaması	Sıra Toplamı	Z	p	r
Durum Çalışması-1	Görsel Prog.	Negatif Sıra	5	6,50	32,50	1,355	,175	,209
		Pozitif Sıra	9	8,06	8,06			
		Eşit	7					
	Eğitsel Robot Prog.	Negatif Sıra	2	4,00	8,00	2,070*	,038	,327
		Pozitif Sıra	8	5,88	47,00			
		Eşit	10					
Durum Çalışması-2	Görsel Prog.	Negatif Sıra	3	3,00	9,00	-,447	,655	-,073
		Pozitif Sıra	2	3,00	6,00			
		Eşit	14					
	Eğitsel Robot Prog.	Negatif Sıra	0	0,00	0,00	2,530*	,011	,390
		Pozitif Sıra	7	4,00	28,00			
		Eşit	14					

*Not: *p<,05*

Her iki durum çalışmasında da deney grubundaki öğrencilerden ön test puanlarına kıyasla betimsel soyutlama son test puanını arttıran öğrenci sayısı azaltan öğrenci sayısından anlamlı şekilde fazla olmuştur (durum-1 için Z=2,070, p<,05; durum-2 için Z=2,530, p<,05). Öte yandan her iki durum çalışmasında da karşılaştırma grubu öğrencilerinin son test puanlarında pozitif puanlar yönünde anlamlı bir farklılığa rastlanmamıştır (durum-1 için Z=1,355, p>,05; durum-2 için Z=-,447, p>,05). Deney grubu ön test-son test puan farklılıklarının büyüklüğünü değerlendirmek için kullanılacak r etki büyüklüğü katsayısı durum-1'de 0,327; durum-2'de 0,390 olarak hesaplanmıştır. Bu durum deney grubunda yapılan öğretimsel müdahalenin ön test-son test puan farklılıkları üzerinde orta derecede bir etkiye sahip olduğunu göstermektedir. Durum-1'de deney grubu ön test-son test puanlarının karşılaştırılması için yapılan istatistiksel analize ilişkin güç değeri 0,68 olarak bulunmuştur. Bu grup için ortalama puanlara dayalı olarak hesaplanan etki büyüklüğü (d=0,503) dikkate alındığında 0,80 düzeyinde bir istatistiksel güç için katılımcı sayısının en az 28 olması gerekmektedir. Bu gruptaki sınırlı katılımcı sayısı (n=20) grup içi karşılaştırmalardaki istatistiksel anlamlılığının sınırlanmasına yol açabilir. Öte yandan Durum-2'de deney grubu test puanlarının değişimi için yapılan analize ilişkin istatistiksel güç değeri 0,90 olarak bulunmuştur. Bu sonuç gerçekleştirilen testin istatistiksel açıdan güçlü olduğunu göstermektedir. Ayrıca bu test için hesaplanan etki büyüklüğüne (d=0,677) göre 0,80'lik bir istatistiksel güç için gerekli katılımcı sayısının 16 olması gerektiği düşünüldüğünde mevcut katılımcı sayısı (n=21) yeterlidir. Bu sonuçlar eğitsel robot programlama aracılığıyla gerçekleştirilen BD ve problem çözme etkinliklerinin öğrencilerin betimsel soyutlama becerisinin desteklenmesinde kullanılabileceğini göstermektedir.

Gruplar arası betimsel soyutlama son test puanları karşılaştırılmadan akademik başarı, betimsel soyutlama ön test ve son test puanları arasındaki ilişkiler Spearman sıra farkları korelasyon analiziyle incelenmiştir (Tablo 58). Buna göre Durum-1'de betimsel soyutlama ön ve son test puanları arasında ilişki olmadığı belirlenmiştir (r_s=,207, p>,05). Buna karşın Durum-2'de formal soyutlama ön test ve son test puanları arasında orta-yüksek derecede pozitif ve anlamlı bir ilişki olduğu görülmektedir (r_s=,697, p<,01). Her iki durum çalışmasında da akademik başarının betimsel soyutlama ön test (Durum-1: r_s=-,162, p>,05; Durum-2: r_s=-,137, p>,05) ya da son test puanlarıyla (Durum-1: r_s=,178, p>,05; Durum-2: r_s=-,149, p>,05) anlamlı

bir ilişkisinin olmadığı belirlenmiştir. Dolayısıyla akademik başarı değişkeninin betimsel soyutlama test puanlarının analizinde ortak değişken olarak kullanılmasına gerek duyulmamıştır. Bununla birlikte betimsel soyutlama ön test puanları ortak değişken olarak parametrik olmayan analizlere dâhil edilmiştir.

Tablo 58

Akademik Başarı Ortalaması, Betimsel Soyutlama Ön Test ve Betimsel Soyutlama Son Test Puanları Arasındaki İlişkilerin İncelenmesi

			Durum-1		
			Akademik Başarı	Betimsel Soyutlama Ön Test Puanları	Betimsel Soyutlama Son Test Puanları
Durum-2	Akademik Başarı	r_s		-,162	,178
		p	-	,310	,267
		N		41	41
	Betimsel Soyutlama Ön Test Puanları	r_s	-,137		,207
		p	,398	-	,195
		N	40		41
	Betimsel Soyutlama Son Test Puanları	r_s	-,149	,697**	
		p	,360	,000	-
		N	40	40	

*Not: **p<.01*

Eğitsel robot programlama ve nesne yönelimli görsel programlama etkinliklerinin betimsel soyutlama puanlarına olan etkilerini karşılaştırmak amacıyla gerçekleştirilen Mann-Whitney U testi sonuçları incelendiğinde her iki durumda da eğitsel robot programlama grubu sıra ortalamalarının daha yüksek olduğu görülmektedir (Tablo 59). Bununla birlikte sıra farklarının gruplar arasında anlamlı farklılık göstermediği görülmektedir. Buna rağmen Durum-2'deki gruplar arası sıra farklılığının istatistiksel anlamlılık sınırına oldukça yakın olduğu görülmektedir (p=0,051).

Tablo 59

Deney ve Karşılaştırma Grubu Öğrencilerinin Betimsel Soyutlama Son Test Puanlarının Mann-Whitney U Testi ile Karşılaştırılması

	Test	Gruplar	N	Sıra Ortalaması	Sıra Toplamı	U	p
Durum-1	Betimsel Soyutlama Son Testi	Görsel P.	21	20,02	420,50	189,50	,567
		Eğ. Robot P.	20	22,03	440,50		
Durum-2	Betimsel Soyutlama Son Testi	Görsel P.	19	17,08	324,50	134,50	,051
		Eğ. Robot P.	21	23,60	495,50		

Ön test puanları kontrol edildiğinde Durum-1'de gruplar arası betimsel soyutlama son test puanları arasında istatistiksel açıdan anlamlı bir farklılık bulunmamıştır (Durum-1 için Quade F=0,337, p=,565>,05). Buna karşın, Durum-2'de gruplar arası sıra farklılığının anlamlı olduğu tespit edilmiştir (Durum-2 için Quade F=6,401, p=,016<,05).

Özetle; her iki durum çalışmasında da uygulama öncesinde test grupları arasında betimsel soyutlama beceri puanları açısından anlamlı bir farklılık bulunmamaktadır. Her iki durum çalışmasında da BD etkinlikleri sonrasında sadece deney grubu öğrencilerinin betimsel soyutlama beceri puanlarında anlamlı bir artış olmuştur. Durum çalışması-1'de betimsel soyutlama son test puanları gruplar arasında anlamlı bir farklılık göstermezken; Durum çalışması-2'de betimsel soyutlama ön test puanları kontrol edildiğinde betimsel soyutlama son test puanlarının deney grubu lehine daha yüksek olduğu bulunmuştur.

Bilgisayımsal Düşünme Etkinliklerinin Öğrencilerin Problem Çözme Algıları Üzerine Etkileri

Çalışma kapsamında sorgulanan bir diğer araştırma sorusu BD etkinliklerinin öğrencilerin problem çözme algılarını nasıl etkilediği ve bu etkinin programlama modalitesine göre herhangi bir farklılık gösterip göstermediğidir. Çalışmada öğrencilerin problem çözme yaklaşımlarının belirlenmesi için Heppner ve Petersen (1982) tarafından geliştirilen ve Şahin vd. (1993) tarafından Türkçeye uyarlanan Problem Çözme Ölçeği (PÇÖ) kullanılmıştır. Öğrencilerin günlük yaşantılarındaki sorunlara nasıl tepki gösterdiğini ya da bu sorunlara karşı nasıl bir yaklaşım sergilediğini belirlemeye yönelik olan ölçeğin Türkçe formu 6 faktörden oluşmaktadır. Bu faktörler; aceleci yaklaşım, düşünen yaklaşım, kaçıngan yaklaşım, değerlendirici yaklaşım, kendine güvenli yaklaşım ve planlı yaklaşımdır. Ölçekten elde edilen sonuçlar faktör puanlarına ya da toplam puana göre kullanılabilmektedir. Bununla birlikte madde sayıları eşit olmasına rağmen, ölçeğin orijinal formu ile Türkçe formunun faktör yapıları (faktör sayıları, faktör-madde ilişkileri vs.) farklılık göstermektedir. Bu nedenle faktör puanlarının ayrı ayrı değerlendirilmesi yerine ölçek toplam puanlarının kullanılması tavsiye edilmektedir. PÇÖ'den alınan toplam puanın yüksek olması, öğrencinin problem çözme becerileri konusunda kendisini yetersiz olarak algıladığını göstermektedir.

PÇÖ ön test sonuçlarına ilişkin hipotez testlerine geçmeden önce veri setinin normal dağılım gösterip göstermediği sınanmıştır. Ön test ve son test toplam puanlarının normallik varsayımları Q-Q ve P-P grafik testleri, Shapiro-Wilk normal dağılım uygunluk testi ve çarpıklık ve basıklık katsayılarına ilişkin Z istatistikleriyle incelenmiştir. Verilere ilişkin P-P ve Q-Q grafikleri incelendiğinde gözlemsel değerler ile kuramsal değerlerin uyum içinde olduğu yani bu değerler arasında önemli sapmaların olmadığı belirlenmiştir. Çarpıklık ve basıklık Z istatistiklerinin hemen hepsinin (Durum-2 için deney grubu PÇÖ son test puanı hariç) %5 anlamlılık düzeyi için kritik değeri aşmadığı (±1,96) görülmüştür. Çalışmada PÇÖ ön test toplam puanlarına ilişkin Shapiro-Wilk W istatistiği değeri Durum-1'de deney ve karşılaştırma grubu için sırasıyla ,938 (p>,05) ve ,986 (p>,05); Durum-2'de ise ,966 (p>,05) ve ,971 (p>,05) olarak bulunmuştur. PÇÖ son test toplam puanlarına ilişkin W değerleri Durum-1'de deney ve karşılaştırma grubu için sırasıyla ,934 (p>,05) ve ,960 (p>05); Durum-2'de ise ,939 (p>,05) ve ,958 (p>,05) olarak hesaplanmıştır (Tablo 60). Bu sonuçlar PÇÖ'den elde edilen veri setinin gruplara göre normal dağılıma uyum sağladığını göstermektedir.

Tablo 60

Problem Çözme Ölçeği Ön Test ve Son Test Puanlarına İlişkin Çarpıklık, Basıklık ve Normallik Testi İstatistikleri

			Çarpıklık		Basıklık		Saphiro-Wilk testi	
		Grup	İstatistik	Std. Hata	İstatistik	Std. Hata	W istatistiği	p değeri
Ön Test	Durum-1	Görsel P.	-,099	,501	-,144	,972	,986	,986
		Eğ. Robot P.	,241	,512	-1,282	,992	,938	,222
	Durum-2	Görsel P.	-,248	,524	-,667	1,014	,971	,803
		Eğ. Robot P.	,325	,501	,488	,972	,966	,640
Son Test	Durum-1	Görsel P.	,629	,501	1,292	,972	,960	,517
		Eğ. Robot P.	,744	,512	-,233	,992	,934	,185
	Durum-2	Görsel P.	-,210	,524	-,530	1,014	,958	,525
		Eğ. Robot P.	,458	,501	2,005	,972	,939	,206

Normallik varsayımının sağlanması sonrası gerçekleştirilen parametrik analizler sonucunda her iki durum çalışmasında da PÇÖ ön test sonuçlarının test grupları arasında anlamlı bir farklılık göstermediği saptanmıştır (durum-1 için t(39)= 1,313, p>,05; durum-2 için t(30,179)= -,166, p>,05) (Tablo 61). Bu sonuçlar uygulama öncesi test gruplarının gündelik problemlere karşı benzer yaklaşımlar sergiledikleri şeklinde yorumlanabilir. Test gruplarına ilişkin PÇÖ ön test puanları incelendiğinde genel olarak öğrencilerin problem çözme konusunda orta düzeyin üstünde bir yeterlik algısına sahip olduğu anlaşılmaktadır. Durum 2'deki öğrencilerin Durum1'dekilerle karşılaştırıldığında uygulama öncesi problem çözme algılarının daha yüksek düzeyde olduğu görülmektedir. Ayrıca her ne kadar istatistiksel açıdan anlamlı olmasa da Durum-1'de deney grubunun (X=93,70) uygulama öncesi problem çözme becerisi algıları karşılaştırma grubuna (X=100,86) göre daha yüksek düzeydedir.

Tablo 61

Uygulama Öncesi Gruplar arası Problem Çözme Ölçeği Ön Test Puanlarının Karşılaştırılması

	Test	Gruplar	N	$\bar{X}$	SS	sd	t	p
Durum-1	Problem Çözme Ölçeği	Görsel Prog.	21	100,857	15,58	39	1,313	,197
		Eğ. Robot Prog.	20	93,700	19,22			
Durum-2	Problem Çözme Ölçeği	Görsel Prog.	19	88,632	12,18	30,179	-,166	,870
		Eğ. Robot Prog.	21	89,619	24,15			

Not: Problem Çözme Ölçeğinden alınabilecek toplam puan 32-192 arasında değişmektedir.*

Öğrencilerin PÇÖ alt faktörlerine ilişkin ön test puanları da incelenmiştir. Bu puanlara ilişkin betimsel istatistikler Tablo 62'de sunulmaktadır. PÇÖ maddeleri faktör yapılarına göre analiz edildiğinde 34. madde hariç hiçbir madde ters kodlanmamaktadır. Ayrıca 11. madde istatistik dışı tutulmaktadır. Faktör puanlarına bağlı ön test sonuçları incelendiğinde her iki durum çalışmasında da gruplar arası PÇÖ faktör puanlarının birbirine oldukça yakın olduğu görülmektedir (Tablo 62).

Tablo 62

Problem Çözme Ölçeği Faktör Puanlarına İlişkin Ön Test Sonuçlarına İlişkin Betimsel İstatistikler

Durum	Faktör	Grup	N	$\bar{X}$	SS
Durum-1	Aceleci Yaklaşım	Görsel P.	21	32,71	5,84
		Eğ. Robot P.	20	34,70	7,89
	Düşünen Yaklaşım	Görsel P.	21	15,81	5,92
		Eğ. Robot P.	20	14,45	5,57
	Kaçıngan Yaklaşım	Görsel P.	21	16,67	4,08
		Eğ. Robot P.	20	18,50	2,35
	Değerlendirici Yaklaşım	Görsel P.	21	9,24	3,02
		Eğ. Robot P.	20	9,30	2,60
	Kendine-Güvenli Yaklaşım	Görsel P.	21	17,10	5,30
		Eğ. Robot P.	20	17,60	6,98
	Planlı Yaklaşım	Görsel P.	21	12,57	4,32
		Eğ. Robot P.	20	11,00	4,12
Durum-2	Aceleci Yaklaşım	Görsel P.	19	34,53	5,83
		Eğ. Robot P.	21	33,81	10,66
	Düşünen Yaklaşım	Görsel P.	19	11,79	2,62
		Eğ. Robot P.	21	12,48	5,32
	Kaçıngan Yaklaşım	Görsel P.	19	15,47	3,56
		Eğ. Robot P.	21	17,24	3,56
	Değerlendirici Yaklaşım	Görsel P.	19	7,58	2,34
		Eğ. Robot P.	21	9,43	4,14
	Kendine-Güvenli Yaklaşım	Görsel P.	19	15,74	4,53
		Eğ. Robot P.	21	15,05	6,31
	Planlı Yaklaşım	Görsel P.	19	9,00	3,09
		Eğ. Robot P.	21	9,67	4,94

Uygulama sonrası hem deney hem de karşılaştırma grubundaki öğrencilerin problem çözme becerisi algılarında anlamlı bir değişim olmamıştır (p>,05) (Tablo 63). İlginç olarak, karşılaştırma grubundaki öğrencilerin problem çözme becerisi algılarında uygulama sonrası hem durum-1'de hem de durum-2'de düşüş görülmüştür. Buna karşın deney grubundaki öğrencilerin PÇÖ puan değişimlerinde tutarlı bir eğilime rastlanmamıştır. Ayrıca deney grubunun ön test ve son test puanları arasındaki farklılıkların karşılaştırma grubuna göre nispeten daha az olduğu belirlenmiştir.

Tablo 63

Gruplara Göre Öğrenenlerin Uygulama Öncesi ve Sonrası Problem Çözme Becerisi Algılarının Değişimi

	Grup	Testler	N	$\bar{X}$	SS	sd	t	p
Durum-1	Görsel Prog.	Ön test	21	100,857	15,58	20	,564	,579
		Son test	21	103,381	19,02			
	Eğ. Robot Prog.	Ön test	20	93,700	19,22	19	-,455	,654
		Son test	20	92,050	19,97			
Durum-2	Görsel Prog.	Ön test	19	88,6316	12,18	18	1,882	,076
		Son test	19	94,1579	17,02			
	Eğ. Robot Prog.	Ön test	21	89,619	24,15	20	,903	,377
		Son test	21	91,7619	23,48			

Öğrencilerin PÇÖ son test puanlarının incelenmesi öncesinde PÇÖ ön test ve son test puanları arasındaki ilişkiler korelasyon analiziyle incelenmiştir. Buna göre PÇÖ ön test ve son test puanları arasında durum-1 için orta düzeyde pozitif ve anlamlı bir ilişki (r=,528, p<,01); durum-2 için yüksek düzeyde pozitif ve anlamlı bir ilişki olduğu belirlenmiştir (r=,824, p<,01). Bu nedenle BD etkinlikleri sonrasında gruplar arası PÇÖ son test puanları ön test puanlarının ortak değişken olarak alındığı kovaryans analizi ile incelenmiştir. ANCOVA öncesi grup varyanslarının homojenliği Levene testi ile kontrol edilmiş ve varyansların homojen olduğu görülmüştür (Durum-1 için F=,185; p>,05; Durum-2 için F=1,170; p>,05). Ayrıca her iki durum çalışmasında da test gruplarına ilişkin regresyon doğrularının eğimlerinin eşit olduğu tespit edilmiştir. Öğrenenlerin programlama başarısı test puanlarına ilişkin düzeltilmiş ortalama sonuçları Tablo 64'te sunulmuştur.

Tablo 64

Problem Çözme Ölçeği Son Test Puanlarına İlişkin Grup Ortalama ve Düzeltilmiş Ortalama Puanları

	Grup	N	Ortalama	Düzeltilmiş Ortalama
Durum-1	Görsel Prog.	21	103,381	101,429
	Eğ. Robot Prog.	20	92,050	94,099
Durum-2	Görsel Prog.	19	94,158	94,615
	Eğ. Robot Prog.	21	91,762	91,349

Tablo 64 incelendiğinde durum-1 ve durum 2'de deney grubundaki öğrencilerin düzeltilmiş PÇÖ son test puanları sırasıyla 94,099 ve 91,349;

karşılaştırma grubundakilerin ise sırasıyla 101,429 ve 94,615 olduğu görülmektedir. Buna göre ön test puanlarına göre düzeltilmiş son test puanları incelendiğinde her iki durum çalışmasında da deney grubundaki öğrencilerin problem çözme becerisi algılarının karşılaştırma grubuna göre daha yüksek olduğu görülmektedir. Bununla birlikte kovaryans analizinden elde edilen sonuçlar programlama modalitesinin ya da yönteminin istatistiksel olarak anlamlı bir farklılık yaratmadığını göstermiştir (Durum 1 için $F_{1\text{-}38}$=1,805, p>,05; Durum 2 için $F_{1\text{-}37}$=,770, p>,05) (Tablo 65).

Tablo 65

Deney ve Karşılaştırma Grubu Öğrencilerinin Problem Çözme Ölçeği Son Test Puanlarının ANCOVA ile Karşılaştırılması

	Varyans Kaynağı	Kareler Toplamı	sd	Kareler Ortalaması	F	p	Etki Büyük. (η^2)
Durum-1	Düzeltilmiş Model	5025,240	2	2512,620	8,603	,001	,312
	PÇÖ Ön test	3710,021	1	3710,021	12,703	,001	,251
	Grup (Prog. Yöntemi)	527,165	1	527,165	1,805	,187	,045
	Hata	11097,882	38	292,050			
	Toplam	408712,00	41				
	Düzeltilmiş Toplam	16123,122	40				
Durum-2	Düzeltilmiş Model	11184,910	2	5592,455	40,472	,000	,686
	PÇÖ Ön test	11127,646	1	11127,646	80,530	,000	,685
	Grup (Prog. Yöntemi)	106,332	1	106,332	,770	,386	,020
	Hata	5112,690	37	138,181			
	Toplam	361514,000	40				
	Düzeltilmiş Toplam	16297,600	39				

Not: Durum 1 için R^2 = ,312 (Düzeltilmiş R^2= ,275); Durum 2 için R^2= ,686 (Düzeltilmiş R^2= ,669)

Uygulama sonrası öğrencilerin PÇÖ son test puanları alt faktörlere göre de incelenmiştir. PÇÖ faktör puanlarına ilişkin son test sonuçları Tablo 66'da sunulmuştur. Alt faktörlere ilişkin betimsel istatistikler incelendiğinde her iki durum çalışmasında da deney grubundaki öğrencilerin problemlere karşı daha özgüvenli, daha düşünen ve daha değerlendirici bir yaklaşım sergilerken; kontrol grubundaki öğrencilerin daha kaçıngan bir yaklaşım sergiledikleri görülmektedir.

Tablo 66

Problem Çözme Ölçeği Faktör Puanlarına İlişkin Son Test Sonuçlarına İlişkin Betimsel İstatistikler

Durum	Faktör	Grup	N	$\bar{X}$	SS
Durum-1	Aceleci Yaklaşım	Görsel P.	21	30,43	7,19
		Eğ. Robot P.	20	35,80	6,44
	Düşünen Yaklaşım	Görsel P.	21	15,71	5,13
		Eğ. Robot P.	20	13,85	4,82
	Kaçıngan Yaklaşım	Görsel P.	21	18,10	3,13
		Eğ. Robot P.	20	18,40	4,96
	Değerlendirici Yaklaşım	Görsel P.	21	9,52	3,93
		Eğ. Robot P.	20	9,15	3,34
	Kendine-Güvenli Yaklaşım	Görsel P.	21	20,14	6,27
		Eğ. Robot P.	20	16,65	6,05
	Planlı Yaklaşım	Görsel P.	21	12,00	3,48
		Eğ. Robot P.	20	12,05	4,36
Durum-2	Aceleci Yaklaşım	Görsel P.	19	33,32	6,01
		Eğ. Robot P.	21	31,38	9,63
	Düşünen Yaklaşım	Görsel P.	19	12,89	3,75
		Eğ. Robot P.	21	12,48	5,04
	Kaçıngan Yaklaşım	Görsel P.	19	16,58	3,55
		Eğ. Robot P.	21	17,29	3,44
	Değerlendirici Yaklaşım	Görsel P.	19	9,05	3,03
		Eğ. Robot P.	21	7,95	2,13
	Kendine-Güvenli Yaklaşım	Görsel P.	19	17,16	4,74
		Eğ. Robot P.	21	15,76	5,64
	Planlı Yaklaşım	Görsel P.	19	10,21	3,74
		Eğ. Robot P.	21	9,57	3,78

Özet olarak; her iki durum çalışmasında da uygulama öncesinde deney ve kontrol grupları arasında PÇÖ puanları açısından anlamlı bir farklılık bulunmamaktadır. Her iki durum çalışmasında da programlama eğitimi aracılığıyla gerçekleştirilen BD etkinlikleri sonrasında grup içi PÇÖ puanlarında anlamlı bir artış yaşanmamıştır. Her iki durum çalışmasında da ön test puanları kontrol edildiğinde PÇÖ son test puanları gruplar arasında anlamlı bir farklılık göstermemiştir.

Bilgisayımsal Düşünme Becerilerinin Geliştirilmesinde Kullanılan Farklı Programlama Etkinliklerinin Öğrenci Motivasyonu Üzerine Etkileri

Çalışmada programlama etkinlikleri için kullanılan araç ve ortamların motivasyona ilişkin test sonuçları üzerinde yaratacağı yenilik etkisini azaltmak için motivasyon ölçeğinin öğretimsel müdahale sonrasında uygulanmasına karar verilmiştir. Bu amaçla mevcut çalışmada öğrencilerin ders materyallerine karşı motivasyonunun belirlenmesinde son test kontrol gruplu deneysel desen modeli benimsenmiştir. Çalışmada motivasyon puanlarının belirlenmesinde Kutu ve Sözbilir (2011) tarafından Türkçeye uyarlanan ve 2 faktör altında toplam 24 maddeden oluşan 5'li Likert türündeki Öğretim Materyalleri Motivasyon Anketi kullanılmıştır. Anketten alınabilecek puanlar 24-120 arasında değişmektedir. Anketten alınan toplam puanın artması öğrenenin ders materyaline karşı motivasyonunun daha yüksek olduğu anlamına gelmektedir.

Programlama modalitesinin öğrencilerin motivasyonu üzerinde anlamlı bir farklılık yaratıp yaratmadığını analiz etmeden önce deney ve karşılaştırma grubunun motivasyon puanları için normallik sınaması yapılmıştır. Normallik sınaması için Q-Q ve P-P grafikleri, Shapiro-Wilk normal dağılım uygunluk testi ve çarpıklık ve basıklık değerlerine ilişkin Z istatistiklerinden yararlanılmıştır. Q-Q ve P-P grafikleri her iki durum çalışması için de motivasyon puanlarının görsel olarak normal dağılımla uyumlu olduğunu göstermiştir. Öğrencilerin motivasyon puanlarına ilişki Saphiro-Wilk W testi değeri Durum-1'de deney ve karşılaştırma grubu için sırasıyla ,940 (p>,05) ve ,925 (p>,05); Durum-2'de ise ,983 (p>,05) ve ,909 (p>,05) olarak bulunmuştur (Tablo 67).

Tablo 67

Öğretim Materyalleri Motivasyon Ölçeği Puanlarına İlişkin Çarpıklık, Basıklık ve Normallik Testi İstatistikleri

		Grup	Çarpıklık		Basıklık		Saphiro-Wilk testi	
			İstatistik	Std. Hata	İstatistik	Std. Hata	W istatistiği	p değeri
Motivasyon	Durum-1	Görsel P.	-,684	,501	-,535	,972	,925	,111
		Eğ. Robot P.	-,882	,512	,547	,992	,940	,239
	Durum-2	Görsel P.	-1,178	,524	2,669	1,014	,909	,070
		Eğ. Robot P.	-,120	,501	,336	,972	,983	,961

Bu sonuçlar veri setinin gruplara göre normal dağılıma uyum sağladığını göstermektedir. Bununla birlikte Durum-2'de karşılaştırma grubunun çarpıklık ve basıklık katsayılarına ilişkin Z istatistiklerinin %5 anlamlılık düzeyini temsil eden ±1,96 değerini aştığı görülmüştür. Fakat bu gruba ait Saphiro Wilk testi istatistiği ,05'ten büyük olduğu için verilerin normal dağılım gösterdiği kabul edilmiştir. Normallik sınamasının ardından motivasyon puanlarının deney ve karşılaştırma grupları arasında anlamlı bir farklılık gösterip göstermediği bağımsız örneklem t testi ile sınanmıştır. Her iki durum çalışmasında da öğrenenlerin motivasyon puanları deney grubu lehine anlamlı bir farklılık göstermiştir (Durum-1 için t_{39}=-2,295, p<,05; Durum-2 için t_{38}=-2,211, p<,05) (Tablo 68). Cohen d değerinin 0,7-0,8 arasında olması orta dereceli yüksek bir etkinin olduğunu göstermektedir. Ek olarak, motivasyon puanlarında gözlenen varyansın yaklaşık %11-12'sinin öğretim teknolojisine bağlı olarak ortaya çıktığı söylenebilir.

Tablo 68

Uygulama Sonrası Gruplar Arası Motivasyon Puanlarının t Testiyle Karşılaştırılması

	Test	Grup	N	$\bar{X}$	SS	sd	t	p	Cohen d	η^2
Durum-1	Motivasyon Ölçeği	Görsel Prog.	21	85,52	12,53	39	-2,295	,027	0,717	0,114
		Eğitsel Robot Prog.	20	93,65	9,92					
Durum-2	Motivasyon Ölçeği	Görsel Prog.	19	80,79	15,50	38	-2,211	,033	0,700	0,113
		Eğitsel Robot Prog.	21	90,05	10,78					

Öğrencilerin motivasyon puanlarındaki farklılıklar faktör düzeyinde de analiz edilmiştir. Böylelikle motivasyon puanlarındaki farklılığın hangi faktörden kaynaklandığının belirlenmesi amaçlanmıştır. Bu analizlerde normallik varsayımının karşılandığı durumlarda bağımsız örneklem t testi, karşılanmadığı durumlarda Mann-Whitney U testi kullanılmıştır.

Öğretim materyalleri motivasyon ölçeği Türkçe formu Dikkat-Uygunluk ve Güven-Tatmin faktörlerinden oluşmaktadır. Ölçek puanları faktörlere göre incelendiğinde her iki durum çalışmasında da eğitsel robot programlama grubundaki öğrencilerin hem dikkat-uygunluk hem de güven-tatmin alt boyutlarındaki faktör

puanlarının nesne tabanlı görsel programlama grubundakilerden yüksek olduğu görülmektedir (Tablo 69). Önemli olarak durum-2'de eğitsel robot programlama grubundaki öğrencilerin dikkat-uygunluk alt boyutuna ait faktör puanlarının nesne tabanlı görsel programlama gurubundakilerden anlamlı şekilde yüksek olduğu saptanmıştır (U=91,50, p<,01). Bu sonucun ortaya çıkmasında durum-2'deki deneysel uygulama sürecinin daha kısa olmasının bir etkisinin olabileceği düşünülmektedir.

Tablo 69

Öğretim Materyalleri Motivasyon Ölçeği Puanlarının Alt Faktörlere Göre İncelenmesi

Durum	Faktör	Grup	N	$\bar{X}$	SS	p
Durum-1	Dikkat-Uygunluk	Görsel P.	21	42,19	5,77	,055[a]
		Eğ. Robot P.	20	45,55	5,05	
	Güven-Tatmin	Görsel P.	21	43,33	8,31	,059[a]
		Eğ. Robot P.	20	48,10	7,31	
Durum-2	Dikkat-Uygunluk	Görsel P.	19	38,00	7,66	,003[b]
		Eğ. Robot P.	21	44,14	4,54	
	Güven-Tatmin	Görsel P.	19	42,79	9,76	,267[a]
		Eğ. Robot P.	21	45,90	7,70	

Not: [a] Normal dağılım; [b] Normal olmayan dağılım

Programlama Başarısı, Problem Çözme, Soyutlama ve Motivasyon Arasındaki İlişki

Çalışmada başarı, problem çözme algısı, soyutlama becerisi (formal ve betimsel soyutlama) ve motivasyon arasındaki ilişkiler son teste dayalı olarak incelenmiştir. Değişkenlere ait veriler gruplara ayrılmaksızın yani deney ve karşılaştırma grupları birleştirilerek analiz edilmiştir. Değişkenler arası ilişkiler incelenmeden önce toplam puanların normal dağılım sergileyip sergilemediği Q-Q ve P-P grafik testleri, Saphiro-Wilk normal dağılım uyum iyiliği testi, çarpıklık ve basıklık katsayılarına ilişkin Z istatistikleri aracılığıyla incelenmiştir. Buna göre motivasyon, formal ve betimleme soyutlama puanlarının normal dağılım sergilemediği görülmüştür. Normal dağılım sergilemeyen değişkenlerin birbirleriyle ve diğer değişkenlerle ilişkisi Spearman sıra farkları korelasyon katsayısı ile analiz edilirken; normal dağılım sergileyen değişkenler arasındaki ilişki Pearson momentler çarpımı korelasyon katsayısı kullanılarak analiz edilmiştir.

Tablo 70

Durum çalışmalarına göre Programlama Başarısı, Problem Çözme Becerisi Algısı, Formal ve Betimsel Soyutlama Becerisi ve Motivasyon Arasındaki İlişkiler

			Programlama Başarısı	Formal Soyutlama Becerisi	Durum-1 Betimsel Soyutlama Becerisi	Problem Çözme Becerisi Algısı	Motivasyon
Durum-2	Programlama Başarısı	r		,384[b]	,353[b]	-,208[a]	,258[b]
		p	-	,013*	,024*	,193	,104
		N		41	41	41	41
	Formal Soyutlama Becerisi	r	,064[b]		,352[b]	,065[b]	,080[b]
		p	,693	-	,024*	,684	,620
		N	40		41	41	41
	Betimsel Soyutlama Becerisi	r	,208[b]	-,191[b]		,169[b]	,181[b]
		p	,198	,238	-	,292	,258
		N	40	40		41	41
	Problem Çözme Becerisi Algısı	r	-,180[a]	,255[b]	-,219[b]		-,475[b]
		p	,266	,162	,175	-	,002**
		N	40	40	40		41
	Motivasyon	r	,152[b]	-,035[b]	,252[b]	-,542[b]	
		p	,351	,829	,117	,000**	-
		N	40	40	40	40	

*Not: *p<,05, **p<.01, [a]Pearson korelasyon katsayısı, [b]Spearman korelasyon katsayısı*

On iki haftalık bir uygulama sürecine sahip Durum-1'de öğrencilerin programlama başarı puanları ile formal soyutlama puanları (rs=,384, p<,05); yine programlama başarı puanları ve betimsel soyutlama puanları (rs=,353, p<,05); son olarak formal ve betimsel soyutlama puanları (rs=,352, p<,05) arasında pozitif, orta düzeyde ve anlamlı bir ilişki olduğu görülmektedir. Ancak Durum-1'de bu üç değişken arasında görülen bu ilişkiler uygulama sürecinin sekiz hafta sürdüğü Durum-2'de anlamlı çıkmamıştır (p>,05). Her iki durum çalışmasında da öğrencilerin problem çözme becerisi ve motivasyon puanları arasında negatif, orta düzeyde ve anlamlı bir ilişki olduğu saptanmıştır (Durum-1 için rs=-,475, p<,01; Durum-2 için rs=-,542, p<,01). Daha önce de bahsedildiği gibi problem çözme ölçeğinde daha düşük puanlar daha yüksek bir problem çözme beceri algısına işaret etmektedir. Yani problem çözme ve motivasyon arasındaki ilişkinin negatif yönlü olması aslında bu iki değişken arasındaki pozitifliği temsil etmektedir. Diğer bir ifadeyle öğrencilerin problem çözme becerisi algıları yükseldikçe motivasyonları da yükselmektedir. Problem çözme becerisi ve öğretim materyallerine karşı motivasyon değişkenlerinin ikisi de öz-raporlamaya dayalıdır. Bu iki değişkenin birbiriyle ilişkili olmasında her iki değişkene ait verilerin öğrenci algılarına dayalı olarak toplanmasının katkısının olduğu düşünülmektedir.

Bilgisayımsal Problem Çözme Sürecinde Öne Çıkan Bilişsel İşlemler

Algoritmik düşünme. Algoritmik düşünme her iki grupta da öne çıkan düşünme biçimlerinden birisi olmuştur. Algoritmik düşünmenin soru bütünlüğünü kavrama ve hızlı bir şekilde çözüm aşamalarına geçmede faydalı olduğu belirlenmiştir. Bu süreçte problemin anlaşılması, önerilen çözüm adımları arasında mantıksal ilişkilerin kurulması, adımların birbirini tamamlaması ve çözüm rotasındaki konumlarının farkında olunmasının önemli olduğu görülmüştür.

> *R_K19: İlk yaptığım aşamalara göre diğer aşamaları iyice bir üstüne basarak, yani tekrar tekrar ona bakarak diğer aşamaları da başardım. Yani adım adım ilerledim, bu da benim hiçbir sıkıntıya uğramamamı sağladı.*
>
> *R_K8: Kâğıt üzerinde uygulama yapmamız. Sonra sınıf numaralarımızı yazıp, algoritmayı bulmamız. Tekerleğin yarıçapını hesaplamamız. Float'ı çözümledim.*
>
> *C_K29: Polya ilkelerini izlerim. Adım adım doğruyu yapmaya çalışırım. Arasındaki bağı çözüp ufak detayları değiştirip etkin hale getiriyorum.*
>
> *C_K24: İlk başta soruyu anlamalı ve tek tek başlamalı ve sonra konuyu ve soruyu kafanda canlandırmalı ve sonra soruyu çözmelisin.*
>
> *C_K63: En önce soru/soruların mantığını en iyi şekilde okumak ve anlamak, sonra da tek tek işleme, çözüme doğru gitmektir. Hemen yapamadım diye bırakmamak gerekir.*
>
> *C_K22: Hocam ilk başta planlı yapmamız gerekiyor her şeyi. Planlamazsak çünkü o problemi çözemeyiz. Hocam zaten adımlarla yaptığımız zaman uygulamaya geçiriyoruz kolaylıkla.*

Zihindeki mantıksal ilişkilerin kâğıt üzerinde somutlaştırılmasının algoritmik düşünme sürecinin operasyonel hale gelmesinde kolaylık sağladığı ve akranlar arasında ortak çalışma olanakları yarattığı görülmüştür. Algoritmik düşünmenin soruları analiz etme, işlem basamaklarını belirleme ve algoritmayı izleme olmak üzere üç aşamada gerçekleştiği belirlenmiştir.

> *R_K57: Nasıl yola çıkacağım, işlem basamaklarını öğrendim [sistematik bir yol ya da model takibi] ve daha sonra karşıma çıkan problemde bunu kullandım, baya yardımcı oluyor.*

C_K65: PC kodlarını aklımızda tutma çok sıkıntı yaşattı, ama bu kodları istediğim şeye ve nasıl olacağına göre sıralayınca çok kolayca kodları girmeye başladık.

Matematiksel düşünme. Bu düşünme biçimi özellikle eğitsel robot programlama dersinde robot hareketlerinin yönlendirilmesi sürecinde deneme yanılma biçiminde yapılan çözüm girişimlerinin işe yaramaması sonucunda ortaya çıkmıştır; örneğin robotun dikdörtgen oluşturacak şekilde hareket ettirilmesi gibi. Öğrenciler ilk olarak belirli bir motor gücünde robotun hareket süresini ya da belirli bir hareket süresi içerisinde tekerleklere bağlı motorların gücünü değiştirerek robotun belirli bir mesafe kat etmesini (örneğin 1m) sağlamaya çalışmıştır. Bu girişimler çoğu zaman başarısız sonuçlar vermiştir. Öğrencilerin zorluk yaşadığı bir diğer husus dönme açısının ayarlanması olmuştur.

R_K2: Hesaplamalarda, ayar vermelerde problem yaşadık. Yanlış ya da göz kararı yaptığım ayarlar.

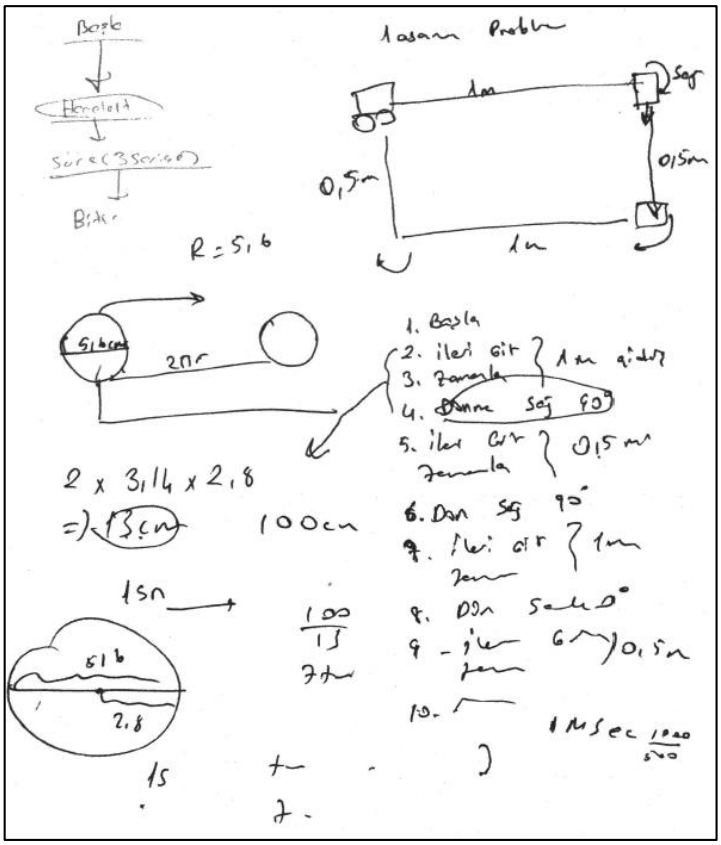
R_K12: Bugünkü robot programlamada kötü olan, robotu 1 metre ileri gönderme sorusundaki matematiksel işlemdi.

R_K14: NXT programlama robotun derecesini [dönüş açısı] filan yapmakta zorlandım.

R_K8: Robotu programlarken 1 m [aralıklarla] kare şeklinde ve "S" şeklinde dönmesini öğrendim. Tekerleğin yarıçapının kaç olduğunu öğrendim.

R_K9: Öncelikle matematiksel yolla gittim ama ilk başlarda deneme yanılma yoluyla gittim. En son matematiksel bir yolla sonuca ulaştım.

Birçok öğrenci bu deneme yanılma süreçleri sonrasında ya görevle uğraşmayı bırakmış ya da görevler için kendilerine verilen süreleri tüketmiştir. Buna karşılık bazı öğrenciler robotun hareketlerini matematiksel bir temele oturtmuştur (Şekil-7).



Şekil 7. Eğitsel robot hareketlerinin matematiksel esaslara göre modellenmesi

Mantıksal düşünme. Genellikle formal işlemler açısından ele alınsa da mantıksal düşünme, bileşenler arasındaki ilişkileri ve nedensellikleri fark etme, sınıflama, genelleme ve bunlara ilişkin çıkarımlara ulaşmayı kapsamaktadır. Bu beceri, bireyin çeşitli akıl yürütmeler aracılığıyla bir sorunu çözmesi veya birtakım soyutlama ve genellemelerden hareketle kurallara ulaşması olarak tanımlanabilir (Korkmaz, 2002). Mantıksal düşünme sebep-sonuç ilişkisi içerisinde bir sonuca ulaşmayı amaçlar. Temelinde ardışık düşünmenin yattığı bu düşünme bir problem durumuna ait tüm verileri elde etmeyi ve bunları zincirleme bir biçimde düzene koymayı gerektirir (Kılıç & Sağlam, 2009). Mantıksal düşünme bir anlamda problem çözme sürecinde işe koşulan bilişsel bir beceri olarak düşünülebilir. Çalışmada mantıksal düşünme ile ilgili hususların deney grubunda daha çok adreslendiği belirlenmiştir.

> *R_K9: Memnun olmadığım; zekâmızı çalıştıramadık. Verilmiş olanlardan sonucu bulamadık.*
>
> *R_K12: Daha çok tartışarak ve fikir [mantık] yürüterek, Polya'nın adımını ve algoritmayı oluşturarak daha rahat yapabildik.*

C_K21: Soruyu anlamaya çalışırım. Veya mantıksal olarak yapmaya çalışırım ya da eski yaptığım problemlere bakarım.

Bilgisayımsal etkinliklerin ilk haftalarında öğrencilerin problem durumunda var olan ilişkilerden ziyade yaptıklarıyla bunların sonuçları arasında nedensellik ilişkisi kurduğu görülmüştür.

R_K14: Bir programda problemi çözmeye çalışırken yaptığım yanlışa bakarım ve doğru yapmak için fikir yürütürüm [akıl yürütürüm].

Soyutlama. Her iki gruptaki öğrencilerin de problem çözme sürecinde zamanla ayrıntılara daha çok dikkat ettiği gözlemlenmiştir.

R_K4: [Problemi] daha iyi okuma ve daha dikkatli problem çözme, problemi en kısa sürede çözmemi sağlıyor.

C_K32: Dersi çok dikkatli dinlemelerini ve ayrıntılara takılmalarını [dikkat etmelerini] tavsiye ederim. Koşullara, sayının [değişken değerlerinin] arttırılabileceğine, koşulların tekrarlanacak işlemlerine, sayaç bilgisine, sıra döngüsüne dikkat edilmelidir.

C_K29: Herhangi bir sorunu daha detaylı düşünüp farklı yollarla çözüm bulabiliyorum. Daha detaylı bakıyoruz bir sonraki probleme.

Bu ayrıntılar sıklıkla öğrencilerin karmaşıklık hissine kapılmalarına yol açmıştır. Öğrenciler haftalar ilerledikçe konulara daha bütünsel olarak yaklaşmak ve daha büyük konu kümesi üzerinde çalışmak zorunda kalmıştır. Öğrenciler tarafından yeni öğrenilecek konuların anlaşılmasını güçleştirdiği belirtilen bu durumun onlarda bilişsel yüklenmeye neden olduğu düşünülmektedir.

C_K31: Hocam kodlamanın bize yararı vardı. Ama çok da kafamızı karıştırıyordu. Çünkü birden fazla şeyi görüyorduk. Ama kafamızı da baya karıştırdı.

R_K57: Konular çok zordu ve üst üste birikiyor. Bu da derste sıkılmama yol açıyor. Tüm ders saatini konuyu anlamakla geçiriyorum. Robotu programlama çok zor hal aldı artık.

Bazı öğrencilerin karmaşıklıkla başa çıkmak için analojilere başvurduğu görülmüştür.

> *C_K31: Onlar robot olduğu için fazla bir şey diyemiyorum, ama verileri soruyu çözerken kavanozun içine almalarını öneririm ve soruyu kolaylaştırıyor.*

Robot programlama grubundaki öğrencilerin kodlama sonuçlarını robot üzerinde görmeleri, yazdıklarının nasıl bir işlev gördüğü konusunda çıkarım yapmalarını kolaylaştırmıştır. Görsel programlama grubundaki bir öğrenci (C_K70) bunun kendileri için önemli bir eksiklik teşkil ettiğini bildirmiştir.

> *R_K52: Yaptığımız soyutsal işlevler somutlaşınca daha mutlu oluyoruz.*
>
> *R_K58: Bir de yazdığımız programları göz önüne seriyoruz.*
>
> *C_K70: Hocam yaptığınız programı robota yüklüyorsunuz ya öyle yapıyorlardır [deney grubu] böylece onun robotta nasıl bir işlem [işlev] gördüğünü de gözlemleme şansımız olabilirdi, bence görsel olarak görebiliriz. Öyle daha iyi olurdu bence, yazdığımız programları görürdük [somutlaşması].*

Olasılıklı düşünme. BD etkinlikleri esnasında öğrencilerin başvurdukları bir diğer düşünme biçimi de olasılıklı düşünmedir. Olasılıklı düşünmenin genellikle öğrencilerin strateji geliştirme/uygulama aşamalarında ve hata giderme çabalarında başvurdukları bir düşünme biçimi olarak ortaya çıktığı görülmüştür.

> *R_K14: Takımımla birlikte düşünerek ve tahmin ederek yaptık ve başardık.*
>
> *R_K18: Bazı yerlerde dikkatli bakmam, dikkat etmem ve tahminde bulunmam.*
>
> *C_K31: … birkaç soruyu yarısına göre tahmin edebildim. Programlama mantığı yavaş yavaş oturuyor.*
>
> *C_K68: Hocam mesela diyoruz ya bu burada olmayabilir. Süslü parantezin yerleri, noktalı virgülün yerleri.*

Öz-düzenleyici düşünme. Her iki gruptaki öğrencilerin de üst bilişsel bir farkındalık içerisinde neyi bilip neyi bilmediklerini sorguladıkları ve eksikliklerini gidermek için çeşitli stratejiler geliştirdikleri görülmüştür.

R_K1: Daha iyi çalışmam gerekli. Bilemedim soruları. Defterden çözerek yapmam gerek, yaptığımız soruları tekrarlamam gerek. Aynı sorunun bir benzerini kendim yazıp çözmem gerek.

R_K8: Daha çok gayret göstermem lazım. Bir programlama problemi sorusunu çözmek için algoritma oluşturmak lazım.

C_K31: Polya'nın adımlarını yaptık. Her şey yolundaydı ama kodlama sıkıntılarım var. Kodlamamı geliştirmem lazım.

C_K32: Daha çok çalışmam ve ek ders almam lazım.

Ders yansımalarının (ders günlüklerinin) öğrencilerin öz-izleme ve öz-denetim yapmalarına önemli bir katkı sağladığı düşünülmektedir.

Bilişsel unsurlar. Öğrencilerin problem çözme sürecinde zaman zaman odaklanma güçlüğü yaşadığı, özellikle içerisinde birden fazla BD kavramı barındıran programlama yapılarını anlamakta ve bunları bilgisayımsal problemlere uygulamakta zorlandığı belirlenmiştir. Öğrenciler, BD etkinliklerinin algılama ve odaklanma kapasiteleri üzerinde olumlu etkisi olduğunu belirtmiştir.

R_K1: [problem çözme etkinlikleri] Algılama odaklanma filan onları geliştiriyor.

C_K24: Evet çok iyi süreçte yaptım. Düşündüğüm şey sadece daha iyi algımı ve bilgimi veremiyordum ve bu hafta daha iyi yaptım ve soruları iyi cevapladım.

C_K57: Konu çok zordu ve anlamam çok zordu. Konular gittikçe çoğalıyor ve zorlaşıyor.

Deney grubundaki öğrenciler BD görevlerinin kendilerini yorucu bir bilişsel çaba içerisine soktuğunu sıkça belirtmiştir. Öğrencilerin çözüm stratejilerini koda dökememeleri zihinsel çabalarının artmasına yol açmıştır.

R_K10: Hep uygulama yapıyoruz Bu da beynimizi yoruyor.

R_K11: Robot [programlama] biraz düşünceli geçmesine rağmen kötü yönleri yoktu.

R_K14: … sorulara grup arkadaşlarımla bakış açımızı ilerletmemiz gerektiğini anladım ve soruları artık düşünüp kafamda toparlayıp öyle yapmaya karar verdim [zihinsel süreçler-muhakeme].

C_K24: … soruları anlıyorum ama kodları ekrana yazamıyorum bu benim kötü yaptığım çözüm oluyor, bundan dolayı soruyu daha geç anlıyorum ama çok iyi çabalıyorum.

Her iki gruptaki öğrenciler de problem çözme sürecinde dikkat becerilerinin önemli bir rol oynadığını belirtmiştir. Özellikle sözdizim kurallarına uygun olarak kodları yazma çabasında öğrencilerin dikkat kaynaklarını yoğun bir şekilde meşgul ettiği görülmüştür.

R_K17: [Bir programlama problemini çözerken] Çalışılmalı, dikkat vermeli ve mantık yürütmeli diye düşünüyorum.

R_K52: [eğitsel programlama etkinliklerinin olumlu yanı] Dikkatimizin artması hocam her şeyde.

C_K33: Dikkat becerimiz arttı. Yazım aşamalarında çok dikkatli olmamız lazım ve en ufak yanlışta çalışmayabilir.

Kişisel sıkıntılar, olumsuz fiziksel koşullar (gürültü gibi), önceki konuların anlaşılamaması ve takım içi uyumsuzluk öğrencilerin üzerinde çalıştıkları işlere yeterince konsantre olamamasına neden olmuştur. Ayrıca yapılan kod hatalarının da öğrencilerin dikkatlerini dağıtıcı bir unsur olduğu görülmüştür. Dikkat dağınıklığı da sözdizimsel hataların artmasına yol açmıştır. Odaklanma güçlüğü yaşayan öğrencilerin sıklıkla notlarına başvurduğu belirlenmiştir.

R_K4: [yolunda gitmeyen işler] Algoritma ve Polya'nın adımları. Soruları artık fazla anlayamıyorum. Çok fazla dikkatim dağılıyor.

R_K18: … dikkatim dağıldığında yani defterden yardım aldık.

C_K33: Dikkatin dağıldığı zamanlarda birçok yazım yanlışı oluyor.

Her iki gruptaki öğrenciler de problemleri ve çözüm stratejilerini zihinde tasavvur etmenin (canlandırmanın) oldukça işe yaradığını belirtmiştir. Ayrıca deney

grubundaki öğrencilerin robot hareketlerini kontrol etmek için sıklıkla zihinsel manipülasyonlara başvurdukları belirlenmiştir.

> *C_K36: Kodlama yaparken hata aldım ve düzeltemedim. Soruyu zihnimde canlandıramadığım için çözemedim. Soruyu zihinde canlandırmalarını öneririm. Gerçekten işe yarıyor.*

> *R_K60: [Robotu] Kafamızda canlandırmayı [manipülasyonu] öğrendik.*

Deney grubundaki öğrenciler zaman geçtikçe robotların ütopik nesneler olmadığının farkına varıp, onların gerçek hayatla bağlantısını kurmaya başlamıştır. Robot programlama etkinliklerinin öğrencilerin hayal güçlerini kullanmalarını teşvik ettiği belirlenmiştir.

> *R_K9: ... dünya çapında mesela ilk olan bir şey yapabiliriz, mesela kimsede olmayan bir şey. Kimsenin yapmamış olduğu bir şey yapabiliyoruz. Yeni bir şey ortaya çıkarabiliyoruz. Mesela hayal gücümüzü kullanabiliyoruz.*

Öğrencilerin Bilgisayımsal Problem Çözme Etkinlikleri Esnasında Yaşadıkları Deneyimleri ve Karşılaştıkları Engeller

Bu bölüm dersin genel çerçevesi ve pedagojik unsurlar, problem çözme ve BD etkinlikleri, öğrenci çalışmaları, duyuşsal boyut, destekler ve engeller ile beklentiler ve diğer çıktılar olmak üzere altı başlık çerçevesinde incelenmiştir.

Dersin genel çerçevesi ve pedagojik unsurlar. Bu bölüme ait bulgular öğretimsel çerçeve ve öğrenme süreçleri olmak üzere 2 alt başlıkta sunulmuştur.

Öğretimsel çerçeve. Her iki grupta da programlama görevlerinin ve problem çözme etkinliklerinin (dersin uygulama kısmı) derse katılımı arttırdığı, öğrenme sürecini ve ders başarısını olumlu yönde etkilediği belirlenmiştir.

> *R_K3: Bu haftaki derste hocamız bize robotun görev kâğıtlarını verdi ve biz de o görev kâğıtlarına bakarak robotu hareket ettiriyoruz ve her hafta bunu geliştiriyoruz.*

> *C_K32: Eğer bu etkinliklere [problem çözme etkinliklerine] katılmazsam başarısız olacağımı biliyorum. Benim için önemli ve gerekli.*

Özellikle karşılaştırma grubundaki öğrenciler komutları ve sözdizimsel kuralları ezberlemenin ders başarılarını yükselteceğini belirtmiştir. Bu öğrencilerin

ders başarısı kaygılarını sıklıkla dışa vurdukları belirlenmiştir. Hatta bazı öğrenciler başarısız olacaklarını düşündükleri için dersi bırakma fikrine kapılmıştır.

> *R_K17: Alışırsak ve yazıları ezberlersek [sözdizimsel kuralları] haftaya daha başarılı olacağımızı düşünüyorum.*
>
> *C_K39: Kodları daha iyi anlamaya çalışmam lazım, ama kodlamalar zor geliyor, hiç yapamıyorum. Başarılı olmam için kodlamayı öğrenmem lazım.*
>
> *C_K29: Dikkatle tahtadakileri doğru ve emin bir şekilde yazmaya ve ezberlemeye çalışıyorum.*
>
> *C_K31: Öğrendiklerimi uygulayamıyorum. Kodlamalarda sıkıntı var. Bölüm değiştireceğim.*

Öğrencilerin dersin uygulama kısmındaki görev performanslarının derse karşı ilgilerini etkilediği görülmüştür. Ayrıca öğrenciler dersin kuramsal kısmından sıkıldıklarını, not almanın derse karşı ilgilerini azalttığını ve dersi sıkıcı hale getirdiğini belirtmiştir.

> *R_K20: İyi yaptığım şeyler daha çok, bu derse ilgi göstermeme neden oluyor.*
>
> *R_K44: Dersi blok yaptık, tüm soruları cevapladık, bu da benim derse isteğimi arttırdı.*
>
> *C_K31: Dinlemek ve yazmak çok avantajlı ama kodlamayı PC üzerinde yapmak iyi, sürekli yazı yazmak dersten soğutuyor.*

Eğitsel robot programlama etkinliklerine katılan öğrenciler, robotların derse karşı ilgilerini arttırdığını sıkça vurgulamıştır. Her iki gruptaki öğrenciler de dersin kuramsal kısmındaki programlama kavramlarının kavranmasının uygulama kısmında başarılı olmak için gerekli olduğunu belirtmiştir. Mesleki kariyerlerini bilişim teknolojileri ve programlama üzerine sürdürmek isteyen öğrencilerin derse karşı daha ilgili olduğu ve dersin kuramsal kısmından daha çok keyif aldıkları belirlenmiştir.

> *C_K28: İlk önce hoca tahtada anlatırken konuyu dikkatli dinlediğim için problemi daha çabuk ve doğru yapıyorum.*

C_K26: Kodları iyi çalışmalıyız ve iyi ve dikkatli dinlemeliyiz ve konuya hâkim olmalıyız.

C_K41: Duygum ve düşüncem iyi dinlersek anlıyoruz neler yapacağımızı. Bugün zorlanmadım.

R_K56: Soruları fazla anlamadan ve hocamızı dinlemeden yapıyorduk. Soruları çözerken biraz zorluk çektik.

R_K57: Programlama Temelleri dersini seviyorum. Çünkü iyi bir programcı olmak istiyorum. Bu yüzden dersi iyi dinliyorum. Dersi dinlerken sıkılmıyorum. Derste yaptığımız şeylerden keyif alıyorum.

Öğrenciler dersin yoğun bir şekilde geçtiğini belirtmiştir. Uygulama süresince programlama temelleri dersi birbirini izleyen kuramsal ve uygulama oturumları şeklinde gerçekleşmiştir. Öğrenciler dersin kuramsal ve uygulama oturumlarının bir arada olmasının programlama kavramlarını daha iyi anlamalarına yardımcı olduğunu belirtmiştir.

R_K5: Çok olumlu, hem ders işledik hem de uygulama yaptık ve bu yüzden olumlu.

R_K7: İlk başta yapamayacağım için ümidi kesmiştim, ama daha sonra hoca anlatınca anladım ve uygulamaya başladık ve ders de çok eğlenceli geçti.

Her iki gruptaki öğrenciler de dersin uygulama oturumunun öğrenmeyi kolaylaştırdığını ve öğrenilenlerin akılda kalıcılığını arttırdığını belirtmiştir. Ayrıca ders saatinin yeterli olmadığı, dersin uygulama kısmına daha çok süre ayrılması ancak dersin kuramsal kısmının da çok sıkıştırılmaması gerektiği belirtilmiştir. Bu husus Anadolu Teknik Lisesi öğretim programlarında Programlama Temelleri ders saatinin 3 saat olmasından dolayı özellikle Durum çalışması-2'de oldukça eleştirilmiştir. Dersin kuramsal ve uygulama kısmı arasında verilen ders araları öğrencilerin uygulama kısmına daha enerjik olarak geçmelerine katkı sağlamıştır. Uygulama oturumunun kesintiye uğramaması da öğrencilerin derse olan ilgilerinin sürekliliğini etkilemiştir. Ancak bu ilgi sürekliliğinin görev performansına oldukça bağlı olduğu görülmüştür.

C_K31: Uygulama yapmak çok eğlenceli ve hem de öğrenmeyi kolaylaştırıyor, [hem de] çabuk öğreniyorsun.

R_K44: Çok eğlenceli bir ders. Komutları uygulama yaparken çok daha iyi anlıyorum, hem soru çözüp hem de uygulama yaparken daha akılda kalıcı oluyor.

Dersin kuramsal kısmında önceki haftalara yönelik yapılan kısa konu tekrarlarının öğrencilerin bilgilerini tazelenmesine yardımcı olduğu ifade edilmiştir. Bu konu tekrarları özellikle ilgili haftada derse katılmayan öğrencilerin dersten kopmamalarına ve derse bağlanmalarına katkı sağlamıştır. Konu tekrarlarının faydaları özellikle karşılaştırma grubu öğrencileri tarafından dile getirilmiştir.

C_K31: Birkaç basit hata dışında iyiydi, operatörleri tekrar ettik. Bilgi yeniledik.

C_K65: Konu tekrarı yaptık. Aklımızda kalmamıştı bazı şeyler onları yaptık, anladık.

Öğrenciler her ders oturumunda yeni programlama kavramlarıyla tanışmalarının kendilerini oldukça yorduğunu ve bu kavramların önceki öğrenmelerle bağlantısının kurulmasını zorlaştırdığını belirtmiştir. Öğrenciler her hafta yeni şeyler öğrenmenin kendilerini meraklandırdığını ve zevkli olduğunu belirtse de ders izlencesindeki yoğunluğun aşırı yüklenmeye neden olduğunu belirtmiştir.

R_K2: Yeni bir program öğrendik. Çok yorucuydu. Ben konulardan şikâyetçi değilim, sadece bir konuyu uzun süre işleyemiyoruz. Her hafta yeni bir konuya geçtiğimiz için kavrayamıyoruz. Sorular gittikçe zorlaşıyor özellikle bu hafta.

R_K3: Bütün programlama temelleri dersinde öğrendiklerim yoktu derste [yeni konular öğrenildi].

R_K5: İyi, zevkli bir dersti; çünkü yeni yeni konular öğrendik.

R_K10: Yeni algoritmalar öğrendik. Çok güzel duyguydu ama yorucu oldu.

C_K30: İyi, hoş, güzel ders ama bu fonksiyon dersini anlayamadım. Geçen haftaki derslere göre biraz farklıydı.

Öğrenciler anlamakta güçlük çektikleri konuların kavranmasında ortak çalışmaların ve akran tartışmalarının etkili sonuçlar verdiğini belirtmiştir.

R_K12: Bu hafta konu ağırdı. Fakat arkadaşlarımla tartışarak üstesinden geldik.

Standart ölçme araçlarının ve geleneksel değerlendirme yöntemlerinin öğrencilerin kaygılarını yükselttiği gözlemlenmiştir. Öğrenciler bildiklerini kâğıda dökmekte zorlandıklarını ifade etmiştir. Öğrenciler süreç temelli değerlendirme yaklaşımlarının ve performans göstergelerinin değerlendirme sürecine dâhil edilmesini istemiştir.

R_K6: Hocam bana bir problem verin, ben uğraşa uğraşa çözerim hocam, bir türlü ama ne bileyim sınavda yapamıyoruz.

C_K37: While konusunda yapabildim [programlama görevi], ama soru sorulup quiz yaptığımızda yapamadım.

C_K78: Hocam bir de sınavda çıktığı zaman heyecanlanıyorum, soruları karıştırıyorum, ondan belki yapamıyorum.

Her iki gruptaki öğrenciler de bilgisayar laboratuvarının fiziksel koşullarından ve sınıf düzeninden (bilgisayar sayısı, çalışma masası sayısı (iş istasyonu), oturma düzeni, sınıf temizliği vb.) memnun olduklarını belirtmişlerdir. Bununla birlikte sınıftaki taburelerin ergonomik açıdan uygun olmadığı, uzun süreli çalışmalarda sırt ve bel ağrısına yol açtığı belirtilmiştir.

R_K4: Masa ve öğrenci sayısı normaldi. Öğrenci sayısına uygun 3-4 masa vardı, yeri iyiydi.

C_K35: Dizilişler ve bilgisayarlar iyi ama sandalye ve oturaklarda yaslanmak için destek olmaması benim ve arkadaşlarım açısından kötü olduğunu düşündüm.

C_K39: Sınıftaki ders düzeni iyi ama sandalyeler yüzünden sırtımız çok bükülüyor ve ağrıyor, bu da iyi bir şey değil tabi ki. Tek sorun bence sandalyeler.

Deney grubunda hem bilgisayar hem de robotik kitler için çalışma masasına ihtiyaç duyulması sınıf ortamının robotik etkinliklere yönelik olarak düzenlenmesini

gerektirmiştir. Bu grupta çalışma istasyonları birbirinden yeterince uzakta kurulmaya çalışılmıştır.

> *R_K16: Robot yapımı için her şey uygundu. Masa sayısı eşit, sınıftaki öğrenci sayısı normaldi.*
>
> *R_K8: Takımların ayrılması iyiydi [fiziksel olarak birbirlerinden yeterince uzak olması]. Masa sayısı iyiydi. Takımlar ayrılmasaydı anlayamazdım.*

Öğrenme süreçleri. Bilgisayımsal problemler aracılığıyla programlama kavramları üzerinde çalışmanın konuların daha iyi anlaşılmasını sağladığı ve kavramların öğrenilmesini kolaylaştırdığı belirtilmiştir. Problem çözme etkinlikleri, bir anlamda bilinenleri ortaya çıkarmak için fırsatlar yaratmıştır. Ayrıca bu etkinlikler öğrencileri kendi öğrenmeleri üzerinde düşünmeye teşvik etmiştir. Öğrencilerin bildiklerini ortaya çıkarmaları ve bunun karşılığını almaları öğrenmeden keyif almalarını sağlamıştır.

> *R_K2: Çok eğlendik, çok iyi anladık, çünkü çok uğraştık.*
>
> *R_K16: 2. ve 3. soruda biraz zorlandım. Galiba konuyu tam anlamamıştım ama genel olarak iyiydim.*
>
> *R_K20: Konuya hâkim değildik, o yüzden kötüydük yani.*
>
> *C_K23: Öğrendiklerim güzel, insana bir şeyler öğrenince keyif veriyor.*
>
> *C_K37: Etkinlikleri yapmazsak rahat anlayamayız. Gereklilik noktasında etkinliği yaptığımızda daha rahat anlıyorum. Etkinlik yapmazsak anlamakta zorlanırım.*

Öğrencilere ihtiyaç duydukları anda sağlanan küçük destekler onların performanslarında önemli artış sağlamıştır.

> *R_K19: Hocam zoru başardık. Yani ilk gördüğümüz zaman şey yapamıyorduk, şöyle diyorduk, hani bunu yapamayız zor filan, ama belirli bir yardımla da olsa küçük ufak bir yardımla da olsa başardık yani zoru.*

Öğrenciler dersin kuramsal kısmında kullanılan materyallerin (görsel, animasyon, kısa film vb.) oldukça işe yaradığını ve öğrenmeyi zenginleştirdiğini belirtmiştir.

R_K8: Programlama Temelleri dersinde dizilerin sırayla dizildiklerini, sanatların [mimari figürlerin] birbiriyle sırayla dizildiklerini öğrendim. Çok eğlenceliydi.

C_K36: Görsellerle çalışmak daha iyi oluyor.

C_K38: Daha çok görsel kullanılabilir. Video izleyebiliriz.

Her iki gruptaki öğrenciler de programlama etkinliklerinin öğrenmeyi eğlenceli hale getirdiğini belirtmiştir.

R_K48: Robot programlama dersimiz hep mutlu geçiyor. Sürekli eğlenerek öğreniyoruz.

C_K68: Ders zevkli geçti, severek öğrendim.

Deney grubundaki öğrenciler robot parçalarını inceleyip, parçaların işlevini anlayıp (mekanizma bilgisi) parçalar arasındaki ilişkileri ve farklılıkları kavrayıp, bu parçaları birbiriyle birleştirerek tasarım yapmıştır. Bu gruptaki öğrencilerin yapa-boza öğrenme süreçlerini oldukça sık deneyimledikleri görülmüştür.

R_K10: Lego robotu parçalanmış şekilde aldık ve topladık.

R_K12: Bu haftaki derste robot yapabilmeyi öğrendik, eğlenceli ve strateji bakımından güzel geçti. Yapboz gibi parçaları birbirine katarak yapabilmeyi denedik, haftaya daha iyisini yapmayı deneyeceğiz.

Deney grubundaki öğrenciler problem çözme etkinlikleri süresince yukarıdan aşağıya (top-down) doğru bir yaklaşım izleyerek yaptıklarını robot üzerinde gözlemlemek suretiyle programlarını iteratif bir şekilde (peyderpey) şekillendirirken; karşılaştırma grubundaki öğrenciler aşağıdan yukarı (bottom-up) bir yaklaşımla program kodlarının işlevini derleme öncesinde anlamaya çalışmıştır.

C_K41: Kodların [programdaki] izinden gidince insan biraz da olsa anlamaya çalışıyor.

R_K46: Robot hareket ettiğinde o hareketleri görüyoruz, daha iyi oluyor. Verdiğimiz komuta göre.

Öğrenciler dersin kuramsal kısmında aldıkları notların oldukça işe yaradığını ve önemli bir öğrenme desteği sağladığını belirtmiştir. Ayrıca notların düzenli bir şekilde tutulması konuların evde tekrar edilmesinin önünü açıp, öğrenilenlerin

kalıcılığının arttırılmasına katkı sağlamıştır. Gerçekleştirilen problem çözme etkinliklerinin sonunda problemlere ilişkin yanıtlar öğrencilerle paylaşılmıştır. Problem çözme girişimlerinin etkililiğine yönelik olarak bunun öğrencilere bir geri bildirim sağladığı görülmüştür.

> *R_K6: Hepimiz defterden bakarak bir sonuca varmaya çalışıyoruz. Hocamız bize yazı yazdırdığı zaman of! filan diyoruz ama o yazdıklarımız belki olmasaydı bu kadar ilerleyemezdik.*
>
> *C_K67: Yazamadığımda defterden bakmayı denedim. Yine yazamadığımda hatalarımı düzeltmeye çalıştım. Bu dersi not almadan anlamak imkânsız. Bugün onu anladım.*

Öğrencilerin programlama kavramlarına aşinalık kazandıkça, özellikle önceki haftaki konuları tekrarladıklarında, bildiklerini daha kolay uygulayabildikleri görülmüştür. Ayrıca programlama kavramlarının problem çözme etkinliklerinde tekrar tekrar kullanılması bu kavramların akılda kalıcılığını arttırmıştır.

> *R_K16: 1. Soruyu yaptım, öğrendiklerim ve aklımda kalanlarla çözmeye çalıştım. Gayret gösterdim ve çözdüğüm soruyu iyi yaptığımı düşünüyorum.*
>
> *C_K24: Avantaj ToolBox güzel yaptım. Gitgide iyi yapmaya çalışıyorum. Şimdi bildiklerimi unutmuyorum. Çünkü çok iyi yapmayı öğreniyorum.*
>
> *C_K26: Kodları iyi yaptım, konuyu hatırladım.*

Her iki gruptaki öğrenciler de yeterince konu tekrar yapmamalarının öğrenmelerini olumsuz etkilediğini belirtmiştir. Başarısızlık durumlarında öğrencilerin sıklıkla başvurdukları yöntem konu tekrarı olmuştur.

> *R_K57: Bu hafta biraz geriden [uygulama sorularında] geldik, sorularda çok zaman kaybettim. Bu da günlük olarak tekrar yapmadığım için oldu sanırım. Bundan sonra günlük tekrar yapacağım.*
>
> *C_K71: Kendimiz program yazmaya çalıştık, ama hiçbirimiz yapamadık, evde hiç tekrar yapmıyorum maalesef.*

Öğrenciler yeni öğrendikleri kavramları öncekilerle birleştirmekte zorlanmıştır. Bu nedenle öğrenciler, içerisinde birden fazla kavram içeren problemleri çözmekte zorlanmıştır.

R_K15: Yani öğrendiğimiz şeyleri, soruları, konuları ya da daha eski konuları aklımıza getiremiyoruz.

C_K36: Dizilerde döngüyü birleştiremedim [önceki öğrendikleriyle bağlantılama].

Deney grubuyla karşılaştırıldığında karşılaştırma grubundaki öğrencilerin öğrenme süreçlerini daha öğretmen bağımlı bir şekilde sürdürdüğü görülmüştür. Bu durum onların öğrenme ortamından ya da araçlarından aldıkları desteğin yetersiz olduğu şeklinde yorumlanmıştır.

C_K23: Hocadan destek aldım. Öğretmenim sayesinde az bir şey öğrendim. Tek sebebi hocam. Yapamadığımda öğretmene sordum.

C_K41: Tek tek hocanın gösterdiği problemlerle gittiğim yaptığım oldu, dinlemediğim kaçırdığım zaman bekledim yapılmasını, öyle işte.

C_K67: Mesela hocam C# ile işlediğimiz zaman hocam bir şey yapıyorduk ya örneğin hoca bize diyordu C#'ı açın tahtadan verdiğini hemen oraya yazıyorduk.

Problem Çözme ve Bilgisayımsal Düşünme Etkinlikleri.

Problem çözme. Polya'nın rehber sorularından yararlanılarak hazırlanan formların problem çözme sürecinde öğrenciler için bir iskele görevi gördüğü ve öğrenci çalışmalarını daha verimli hale getirdiği belirlenmiştir.

R_K10: Polya'nın adımlarını güzel bir şekilde yaptık. Düzenli bir şekilde yapınca hem anlaşılır oluyor hem de uygulamalar sıkıntı çıkmadan çalışıyor.

R_K57: Önce karşımızdakinin bizden ne istediğini anlamamız gerekiyor. Ne istiyor? Bize neler vermiş? Bunları anlamamız lazım, bunları geliştirdi bende.

C_K23: Polya'nın problem çözme yöntemi işimizi kolaylaştırıyor. Bu yol üstünde gidersek işler daha da basitleşiyor.

Polya'nın rehber sorularıyla problem çözme sürecinin daha sistematik bir hal aldığı görülmüştür.

R_K57: Hani daha önceden karşıma bir problem çıktığında ilk olarak problemi çözmeye yöneliyordum. Ama bu programlama dersinden sonra ilk

önce sorunu anlıyorum, soruda benden ne istiyor, bana ne vermiş, benden ne beklentisi var.

C_K27: [işimi] daha çok kolaylaştırdığını düşünüyorum, çünkü Polya'nın aşamalarını yaptığım zaman problem kendiliğinden çözülüyor ve ben de daha çok mutlu oluyorum.

Öğrenciler bu adımların özellikle problemlerin anlaşılmasında oldukça yardımcı olduğunu belirtmiştir. Ayrıca bu yöntemin günlük hayattaki işlerde de sıklıkla kullanıldığı ifade edilmiştir.

C_K30: Ben daha önce bunu hiç duymamıştım ama genellikle günlük hayatta kullanıyormuşum. Bunun farkına vardım. Bir [birincisi], soru için kolaylaştırıyor. Çünkü en başında soruyu anlamamıza yardımcı oluyor.

R_K8: İyi etkiledi. Problemi anlamayı kolaylaştırdı.

Öğrenciler Polya'nın problem çözme aşamalarının problem çözme etkinliklerindeki performanslarını arttığını raporlamıştır.

R_K10: Polya'nın adımlarını kullanarak yaptık, kolay oldu. Geçen hafta fazla soru çözemedik, çünkü bu kurala uymamıştık. Bu hafta 3 soru yaptık.

C_K28: Hocanın verdiği kâğıttan yaptığım için problem çözmek daha kolay oldu. Polya'nın yazım aşamaları sayesinde iç içe if komutu ile ilgili problemi çözdüm.

Öğrenciler Polya'nın aşamalarının kod yazma süreçleri dışındaki diğer süreçleri kolaylaştırdığını belirtmiştir. Polya'nın adımlarından özellikle stratejiyi uygulama aşamasında zorlandıklarını ifade etmişlerdir.

C_K26: Polya'nın problem çözme aşamasında kodları yazmada zorlandım.

C_K38: [Polya'nın problem çözme aşamaları] Kodlama hariç her şeyi kolaylaştırıyor.

C_K36: Polya'nın problem çözme adımlarını yapıyorum. Daha çok kodlama kısmında başarısız oluyorum ve yapamıyorum.

C_K38: Polya'nın yolunu takip ettim ama yine de yarıda kaldım. Planı çözme ve uygulamada sıkıntı yaşadım.

Bu adımlar aynı zamanda öğrencilerin eksikliklerinin nerede olduğunun farkına varmalarına yardımcı olmuştur. Ayrıca probleme ilişkin olası diğer çözümleri de düşünmelerine katkı sağlamıştır. Öğrenciler bu yöntemi diğer derslerine de transfer ettiklerini belirtmiştir. Bu durum Polya'nın problem çözme adımlarının genel bir çerçeve olarak farklı alanlarda da işlevsel olabileceğini göstermektedir.

> *R_K44: Polya'nın adımlarını uyguladım ve değişken uygulamada [kullanımında] büyük [çok] eksik olduğumu düşündüm ama hoca anlatınca anladım.*
>
> *C_K26: Polya'nın problem çözme aşamasında birden fazla cevap buldum.*
>
> *R_K57: Matematikte mesela ilk önce böyle anlamadığım [zaman] böyle bakıyordum soruya, direkt atlıyordum. Yani bir şeyler yapmaya çalışıyordum ama basamaklarına [Polya problem çözme basamakları] göre ilk önce soruyu anlamaya başladım. Daha sonra ben bunu nasıl yapabilirim. Veya hani başka çözüm yolu ne var. Bu şekilde oldu. Fizik dersinde de çok yardımcı oldu.*

Problem çözme etkinliklerinin ilk aşamalarında öğrencilerin problem çözme basamaklarını atlayarak doğrudan çözüme geçtikleri, bu durumun genellikle verimsiz sonuçlar ürettiği saptanmıştır.

> *R_K6: Uğraştığımız halde başaramadık, çünkü çözüme giderken hata yaptık, soruyu anlamadan işleme geçmeye çalışıyoruz.*
>
> *R_K12: Kötü yaptığımı düşündüğüm şeyler kodlara [yönelmem] yani Polya'nın adımlarını izlemeden algoritmayı yapmadan doğrudan çözüme ulaşma isteğine kapılmam. Bu kötü oldu benim için.*
>
> *C_K21: Soruyu tam anlayamıyorum anlamakta güçlük çekiyorum. Soruyu anlamadan doğrudan çözümlemeye geçiyorum. Kodlamalarda kafam karışıyor.*

Problem çözme sürecinin en başında ne yapacağını bilememenin kaotik bir durum olduğu göze çarpmıştır. Öğrencilerin bu durumda en sık başvurduğu yöntem problemi tekrar tekrar okumak olmuştur.

> *R_K10: Tüm yönleriyle kolaylaştırdı. Çünkü öncelikle ne yapacağımızı bilmiyorduk. Yavaş yavaş öğrendik.*

R_K14: Soruları fazla anlamadığımız için iki, üç kez tekrar okumak zorunda kaldık ve zamanımız azaldığı için 2 soru yapamadık.

Soruların uzun ve karmaşık olması öğrencilerin problemleri anlamalarında güçlük yaratmış, bu durum öğrencilerde sıklıkla sorunun güç olduğu algısını oluşturmuştur. Bu durum öğrencilerin sorular üzerinde daha fazla zaman harcamasına ve destek arayışına girmesine sebep olmuştur. Öğrenciler soruların daha ayrıntılı ve ipuçlarıyla birlikte verilmesinin yerinde olacağını belirtmiştir.

R_K1: Sorular çok zordu, anlamadım, kafam çok karıştı, anlamadım hiçbir şey.

R_K4: 2. soruda baya bir kafamız karıştığı için kimse çözemedi. … Soruda daha detay olsa iyi olur. İpucu verilse parantez içinde iyi olur.

C_K80: Sorular biraz karışıktı, kafa karıştırıyordu.

Problemi anlama aşamasının kritik göstergesinin problemin analiz edilmesi, daha açık ifadeyle girdi/çıktı belirleme olduğu saptanmıştır.

R_K18: Adım adım ilerlemeli, soruyu anlamalı, bizden ne istenmiş bilmeliyiz, anlamalıyız.

C_K36: İlk sorunu anlamaya çalışırım. Sonra kaç değişken ve hangi türde olduklarını bulmaya çalışırım.

Problemi anlama aşamasından sonraki aşama plan/strateji geliştirme aşamasıdır. Adından da anlaşılacağı gibi bu aşamanın kritik göstergesi bir çözüm planının ya da taktiğinin geliştirilmesidir. Bu aşama öğrencilerin araştırma yaptığı, çözüm yolu geliştirdiği ve planlama yaptığı bir aşamadır. Araştırma aşamasında öğrenciler genellikle notlarını (defter ve çalışma kâğıtları) gözden geçirip, önceki örnekleri incelemiştir. Önceki örneklerle mevcut soru arasındaki benzerlikleri ve farklılıkları irdelemiştir.

R_K6: İyi yaptığım şeyler çok fazla yok. Ama yine de çözüm için araştırmalar yaptım defterden.

R_K7: Defterlerimize baktık. Defterlerden yardım aldık. Bir de sizin bazı günlerde verdiğiniz kâğıtlar vardı ya böyle onlardan yardım aldık.

R_K10: Hep birlikte yapabiliriz çalışmaları, herkesten bilgi alarak defterden işlediğimiz konuları takip ederek daha mantıklı bir yolda ilerleyebiliriz.

C_K21: Programlar arasındaki farklılıklara baktım ve karşılaştırdım [önceki örnekleri inceleme ve bu örnekleri karşılaştırma]. İpucu olarak eski konulardan yararlandım.

C_K30: Deftere bakma değil, bir sonraki yani kıyaslama yapıyorum. Burada bunu kullansam, burada onu kullansam…

Bu aşamada yaşanan sorunlardan birisi sorulara aynı perspektiften bakmak olmuştur. Öğrenciler sorulara daha geniş perspektiften bakabilmek ve farklı fikir üretmek için sıklıkla akran işbirliğine ihtiyaç duymuştur. Bu aşamada fikir üretme teknikleri oldukça işlevsel bir rol üstlenmiştir.

R_K2: Arkadaşlarımla tartışarak algoritmayı oluşturursak sorunların içinden çıkabiliriz.

R_K19: Hocam dediğiniz gibi tartışarak nasıl desem mesela ben bir şeyi bulamıyorum hocam, aklımda hep tek bir şey olduğundan hani tek bulamıyorum.

C_K35: Toplu çalışmanın bence hiçbir olumsuz yönü yok. Hep birlikte beyin fırtınası yapınca daha basit ve daha anlamlı oluyor ve herkes o problemden ders çıkarıyor ve sonucu buluyor.

C_K37: Bireysel olarak aklımıza fazla bir şey gelmiyor ama takım halinde olursak her birinden bir bilgi aldığımızda sorun olmaz.

Öğrencilerin bir çözüm önerisi geliştirdikten sonra bu çözüm önerisinden yeni fikirler filizlendirdiği gözlenmiştir. Daha sonra öğrenciler çözüm önerilerini algoritmalar ve akış şemalarından yararlanarak bir düzen içerisine sokup kâğıda dökmüştür (planlama).

R_K9: Mesela ilk başta hocam, bir yol bulursun. O yoldan ikinci yol, üçüncü yol diye diye soruları yapabiliriz mesela.

Stratejiyi uygulama aşaması öğrencilerin çözüm stratejilerini program koduna dönüştürdükleri aşamadır. Kod yazmak öğrenciler için bilgisayımsal problem çözme sürecinin tamamlanması adına önemli bir eşiktir. Bu aşamada öğrenciler aklındaki ve kâğıt üzerindeki planlarını koda dökmekte oldukça zorlanmıştır.

C_K28: Aklımdaki cevabı bilgisayarda yapamıyorum.

R_K52: Lab'ta yaparken hangi kavramları kullanacağım kafamda oturttum, fakat kodlama stratejisinde başarısız oldum.

C_K37: Yolunda giden sadece algoritma oluşturmamdı.

C_K65: Aklımdan kod yazamıyorum ve buna yazı olarak not alarak çözüm bulduk.

Sonuçları görme ve geriye bakma aşaması öğrencilerin problem çözme çabalarının karşılıklarını aldıkları aşamadır. Bu süreçte öğrenciler sıklıkla hatalarla mücadele etmek zorunda kalmıştır. Durum çalışmalarının ilk haftalarında öğrenciler karşılarına çıkan program hatalarını deneme-yanılma yöntemiyle gidermeye çalışmıştır. Ancak ilerleyen haftalarda hata mesajlarının daha iyi anlaşılmasıyla hata giderme çabaları daha rasyonel bir çabaya dönüşmüştür. Ortaya çıkan kod hatalarının giderilmesiyle öğrenciler çözüm stratejilerinin beklenen sonuçları üretip üretmediğini değerlendirmeye başlamıştır. Her iki gruptaki öğrencilerin de bir sonuca ulaştıktan sonra genellikle olası daha iyi çözümler üzerinde yeterince durmadıkları gözlemlenmiştir.

R_K44: 4.soruyu çözemedik. Denedik, denedik, izlediğimiz yollar hep hata verdi.

C_K40: Bugün çözüme ulaşamadım. Hatayı bulamadım.

C_K25: Bugün iki soruyu da çözdüm ama çok yoruldum, bir iki yanlış vardı ama onları da çözdüm.

C_K29: Hocam mesela kod yazmada verilen hataları gösteriyor. Orada kendimizi geliştirdik hocam, zaman içerisinde hata yapmamaya başladık.

C_K35: Bir de onu [hatayı] giderdiğin için hataları daha iyi kavrıyorsun bu sefer hata olduğunda gidermeyi daha kolay sağlıyorsun.

Görev listelerinde yer alan problemlerin türü (sayısal ve metin işlemlerini içermesi gibi), görev yapısı (hata bulma gibi) ya da aynı görevin birden fazla programlama kavramını içermesi öğrencilerin görev performanslarını etkilemiştir.

R_K8: Hoca hata kâğıdını [görev listesi] verdi, onları bulduk arkadaşlarımızla. 1. soru kolaydı, 2.sinde biraz zorlandık. Ama 3.sünde

bulduk. Belki de iyice bakmadığımızdan veya algoritma oluşturacak tipte soru olmadıklarından zorlandık.

C_K25: Bu hafta soruları yaptım ama sözlü bir soru vardı, onu yapamadım.

Öğrenciler sıklıkla görev listesindeki soruların zor olmasından yakınmıştır. Ayrıca haftalar ilerledikçe görevlerin zorluk düzeyinin daha da arttığını belirtmişlerdir. Bu durumun problem çözme etkinliklerinde daha fazla çaba sarf edilmesine, ancak bazı problemlere takılıp kalınması durumunda görev listesindeki diğer görevlerin yetiştirilememesine yol açtığı görülmüştür. Öğrencilerin özellikle farklı alanlarda ön bilgi gerektiren (örn. matematik) soruları çözmekte zorlandığı belirlenmiştir.

C_K37: Biraz zorlandım ama yapabildiklerim oluyor. Biraz daha zorlaşıyor sorular.

R_K14: Hata bulma zordu. 2. soruda takıldık. Yapmamıza rağmen else'lerde yanlış butonu çalışıyordu. Ne yaptıysak düzeltemedik.

R_K19: Bu hafta yeni görevlere başladık. Birinci kolay, ikinci zor ama çizgi çizmek kolay oldu, sadece zaman yetmedi.

C_K21: Polya'nın problem çözme basamaklarını öğrendik, ama bununla problem çözemedim. Aşırı zorlaştı problemler. Eskisi gibi kolaylıkla anlayamıyorum problemleri.

R_K48: Aslında kötü bir şey yoktu da zor olan vardı. O da robotun alanı, çevresi ve yarıçapını göstermek.

Robot programlama grubundaki öğrenciler özellikle ilk haftalarda robot montajında ve robota program yüklenmesinde zorluk yaşamıştır.

R_K6: Robot yaparken tabi ki de zorluk yaşadık, çünkü ilk deneyimimizdi.

R_K60: Kötü giden şey robotlara programın zor yüklenmesi.

Her iki gruptaki öğrenciler de problem çözme sürecinin bilgisayar üzerinde gerçekleşmesinin yapılan işi daha zevkli hale getirdiğini, öğrenilenlerin akılda kalıcılığını arttırdığını, kodların semantiğinin daha iyi kavranmasını sağladığını belirtmiştir.

R_K15: Hocam bilgisayarda komutları yazmamız. Komut yazıyoruz. Hangi komutun ne işe yaradığını öğreniyoruz.

C_K21: Uygulama yaparak daha iyi anlıyorum ve aklıma daha iyi giriyor.

C_K37: Deftere yazısını yazınca anlamıyorum. Ama bilgisayarda yazınca çok daha iyi anlıyorum.

C_K78: Soruları bilgisayardan yaptığımız için çok güzel oldu, eğlenceliydi.

Öğrenciler ayrıca bilgisayarda programlama yapmanın problem çözme sürecini kısalttığını ve bilinenleri uygulama imkânı sağladığını belirtmiştir.

R_K19: Aynen günlük gibi farz edin hocam, içimizdekileri aklımızdakileri robota aktardık.

C_K21: Uygulamalı olduğu için hatalarımı gördüm ve daha iyi ilerlemeye çalışıyorum.

C_K39: C# olmasaydı bazı problemleri daha uzun süre düşünecektik, bu da bizim notlarımızı daha da düşürebilirdi.

Öğrenciler problem çözme sürecinde edindikleri deneyimlerin sorunlarla başa çıkma ve hızlı karar alma becerilerini geliştirdiğini belirtmiştir. Ayrıca öğrenciler problem çözme etkinliklerinin öğrenilenlerin gerçek hayata transferini kolaylaştırdığını belirtmiştir. Görevlerin gerçek hayatla ilişkili (gerçek yaşama dönük) olmasının öğrenilenleri gerçek yaşam pratiklerine transfer edilmesini teşvik ettiği görülmüştür. Deney grubundaki öğrenciler özellikle gerçek yaşam problemlerine çözüm üretme noktasında edindikleri deneyimlerden yararlanmak istediklerini sıkça ifade etmiştir.

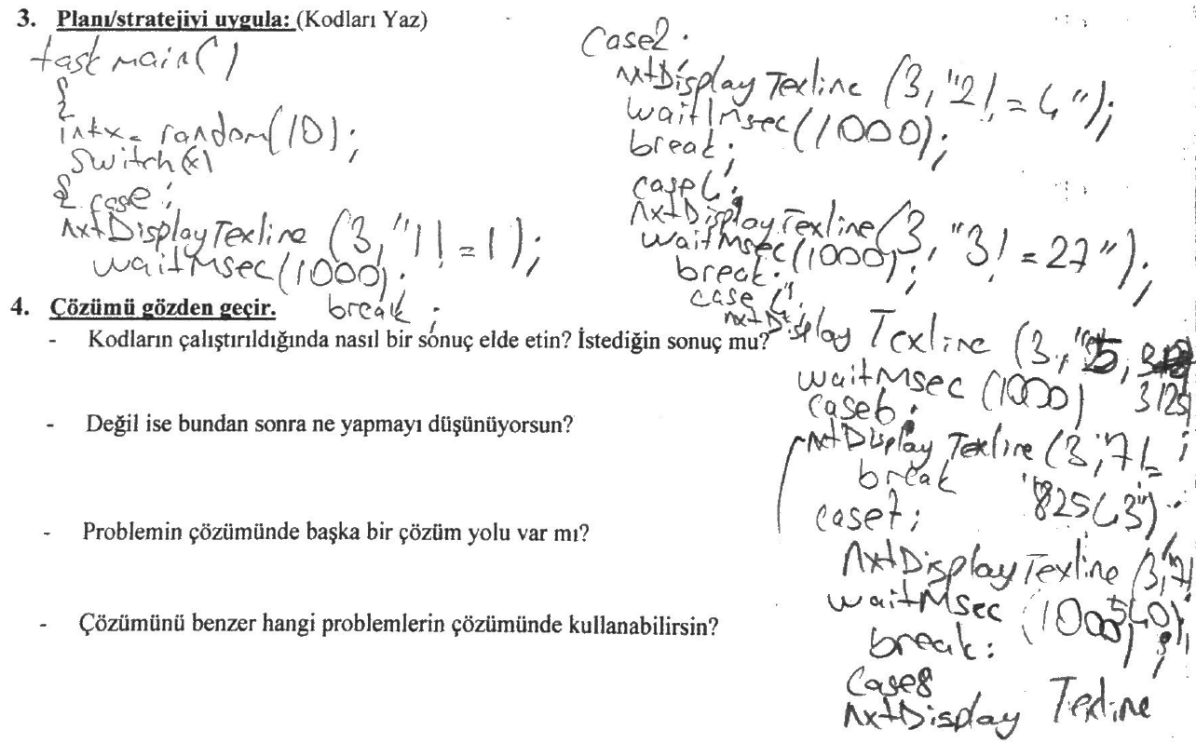
C_K22: Sadece bilgisayarda değil hayatımızda da problem çözüyoruz. Birçok problemleri.

R_K61: Robotun dünyadaki her şeye faydası var. Sensörlerle düşmanı bile görebilirsin.

R_K43: Lab sorularını çözerken geri sayma aleti yaptık, sonunda farklı bir şey istiyor, biz başka bir programlama ile yazdık.

Bilgisayımsal düşünme kavramları ve etkinlikleri. Deneysel uygulama boyunca öğrencilere akış şeması, aritmetiksel işlemler, sayı sistemleri, değişkenler

ve sabitler, veri türleri, tür dönüşümleri, operatörler, koşul işlemleri, döngüler, hazır fonksiyonlar, diziler, alt programlar gibi kavramlar tanıtılmıştır. Öğrencilerin değişkenler ve veri türü dönüşümlerinde aşırı derece zorlandığı gözlemlenmiştir. Aritmetiksel işlemler ve ilişkisel operatörlerin kullanılması esnasında da sıklıkla hata yapılmıştır. Koşul ifadelerinden switch…case yapısıyla döngü komutlarının (while, do…while, for) sıklıkla birbiriyle karıştırıldığı görülmüştür. Şekil 8, sıfır ile on arasında rastgele üretilen bir sayının faktöriyelinin LCD ekranda gösterilmesi görevi için öğrencilerin yazdığı kodları göstermektedir. Şekilde görüldüğü gibi öğrenciler döngü ve koşul yapılarını ayırt edememiştir.



Şekil 8. Öğrencilerin koşul ve döngü işlemlerini karıştırması

Diziler ve alt programların (fonksiyon ve prosedürler) sınıf mantığının gelişmesine katkı sağladığı görülmüştür. Ayrıca alt programların görevi parçala-birleştir yöntemiyle öğrencilerin görev analizi yapmalarını sağladığı ve programlara modüler bir yapı kazandırdığı görülmüştür.

Öğrenciler programlama süreçlerinde bilgisayar, programcı ve son kullanıcı rollerini ayrıştırmakta zorlanmıştır. Problem çözme etkinlikleri dersin uygulama oturumunda gerçekleşmiştir. Dersin uygulama kısmında deney grubundaki öğrencilere Durum Çalışması-1'de ortalama 3,8; Durum çalışması-2'de ortalama 3,7 BD problemi sunulmuştur. Benzer şekilde, karşılaştırma grubundaki öğrencilere de

deney grubuna yöneltilen problemlerin muadilleri yaklaşık olarak aynı sayıda sunulmuştur. Dersin uygulama oturumu yaklaşık 1-2 ders saati (40-60 dk.) sürmüştür. Her iki gruptaki öğrencilere de her hafta farklı görevler verilmiştir.

> *R_K19: Bugün yeni görevler yaptık. Bugün de aynı programı kullandık [robot C] ama değişik görevler yaptık.*

İlk haftalarda gerçekleştirilen problem çözme etkinliklerinde öğrencilere sunulan problem sayısının fazla olması eğitsel robot programlama grubundaki takım çalışmalarında sıkıntılara (görev paylaşımı ve ortak çalışma) yol açmıştır.

> *R_K2: Soruların fazla olması [takım içinde] karışıklığa neden olması olumsuz yönlerden biridir.*

Her iki grupta da problemler genellikle problem çözme süreci sonunda öğrencilerin bir ürün (uygulama) geliştirmesini sağlayacak biçimde hazırlanmıştır.

> *C_K39: Mutluyum, eğer zor hesaplar filan karşımıza çıkarsa hesap makinesi yapıp visual studio'da daha hızlı hesap yapabiliriz.*

> *R_K59: Robotlara rastgele daireler çizdirdik, sensörler sayesinde robotu bir asker yaptık.*

Deney grubundaki öğrenciler görev listelerinde yer alan problemleri genellikle kendilerine verilen süre içerisinde çözmekte zorlanmıştır. Bunun temel nedeni bazı görevlerde takılıp kalmaları olsa da öğrencilerin hazırladıkları programı robota yüklemeleri ve robotun görevleri yerine getirmesini beklemeleri de onların zamanını almıştır.

> *R_K12: Tuğlanın üstüne isimlerimizi, okul numaralarımızı ve ortalamasını yazdırdık. 2.sorumuzda biraz takıldık, fakat zaman yetmediği için fazla yapamadık.*

> *R_K14: Robotu hareket ettirirken ya da robot dersindeyken zamanımız yetmedi, o yüzden bitiremedik.*

Problem çözme etkinliği sonrasında problemlere ilişkin çözümler öğrencilerle paylaşılmıştır. Birçok öğrenci yanıtları görmenin ve tahtadakileri uygulamanın öğrenmelerine yardımcı olduğunu belirtmiştir. Bununla birlikte karşılaştırma grubundaki bazı öğrenciler tahtadakileri uygulamanın mekaniksel bir rutini izlemek gibi olduğunu belirtip, bunun kendileri için pek bir anlam ifade etmediğini belirtmiştir.

R_K13: Değişken türü, karar verme, koşul ifadesi bunları sevmiyorum, yapamıyorum. Ancak sonradan hoca cevapları verdiği için bunları deftere yazarak anlıyorum.

C_K24: ToolBox yapamıyorum ve yazıyorum ve anlayamıyorum ama hocam yaptığımızda anlıyorum ve etkinlikleri yapıyorum.

C_K34: Hocamızın verdiği soruları yapmaya çalışsak da yapamıyoruz. Hocamız yapınca da bir anlamı kalmıyor, ha yapmışız ha yapmamışız bir şey fark etmiyor.

C_K71: Yardım alarak yapabilirim ama tek yapamam, aklım çok karışıyor, bakarak kolayca yapabilirim.

C_K64: Bu derste örnek çözerken hep tahtadan bakıp programı yazıyorduk. Bir kere bakmadan yazmaya çalıştım, yazamadım. Yani hep bakarak olmaz, önemli olan mantığını anlamak.

Öğrencilerin en zorlandıkları aşama kod yazma aşaması olmuştur. Kod yazmak için öğrencilerin en sık kullandığı öğrenme stratejisinin ezber yapmak olduğu görülmüştür. Bu durum kodları unutmalarından dolayı onların sözdizimsel hataları uzun bir süre devam ettirmelerine yol açmıştır.

R_K13: Biraz yazarken zorlandım ama yine de ezberlersek ve öğrenirsek, daha anlaşılır olacağını düşünüyorum.

C_K23: Hata bulma bu hafta çok zorladı beni. Çünkü daha programdaki şeyleri [kodlar ve sözdizimsel kurallar] tam olarak ezberleyemedim.

C_K31: Kodlamayı bilene yararlı ama bilmeyene yararsız, önemli olan kodlamak ama dil olarak bizi zorluyor. Sorunu anlamayı biliyorum, çözüyorum ama bir sıkıntı var ki onu C#'a kodlayamıyorum.

C_K33: Yazım zorluğuyla çok yazım hatası yaptım.

C_K36: Kod yazma yine kötüydü. Yazamıyorum şu kodları.

C_K41: Kodların yerlerini ezberleyemiyorum ve onlarda sıkıntı çekiyorum.

Öğrenciler kodların da nesneler gibi sürükle-bırak şeklinde değil de metin (sözdizimi) şeklinde olmasının sıkıntı çekmelerine yol açtığını belirtmiştir. Ayrıca kod editöründe fare hareketlerinin zaman zaman hatalara yol açtığı belirtilmiştir.

Bununla birlikte kodlama editöründeki kod ipuçlarının ve kod tamamlama özelliğinin kod yazma işini önemli ölçüde hızlandırıp kolaylaştırdığı ve öğrencilere zaman kazandırdığı belirtilmiştir.

R_K10: Yazıyla yazdığımız için sıkıntı.

C_K40: Ben hesap yaparken butonun içinde biraz sıkıştım.

C_K32: Tek harf yazdığımızda kelimenin hepsinin çıkması müthiş bir şekilde kolay oluyor [kod tamamlama özelliği] ve bu benim hoşuma gidiyor.

C_K41: Bu program kolay, kolay gelen yerleri son cümleleri tamamlamadan hangi cümleyi yazacağın çıkıyor ve bana daha iyi zaman kalıyor [zaman tasarrufu].

Deney grubundaki öğrenciler takım çalışmalarında açıklama satırları kullanmanın konu üzerinde işbirlikli çalışmayı kolaylaştırdığını belirtmiştir.

R_K1: Yol izlemek için arkadaşlarımla tartışıp, konuyu anlamamız gerekir. Task main kullanılmalı. Açıklama satırları kullanılmalı.

Öğrenciler, programlama etkinlikleri sayesinde bildiklerini kanıtlayabildiklerini (somutlaştırdıklarını) belirtmiştir. Ayrıca yazılan programın derlenmesinin programdaki eksiklikleri ortaya çıkardığı belirtilmiştir.

R_K49: Bir de doğru yaptığımızı robot ile kanıtlayabiliyoruz.

C_K79: Mesela dışardan birisi şunları şunu biliyor musun derse hemen ona öğretiriz.

R_K42: Programı [robotu] izlerim ve nerde hata yaptığıma bakarım veya bir şey mi unuttum deyip bir daha bakıyorum.

Öğrencilerin haftalar ilerledikçe kodlara aşinalık kazandığı ve kod yazma girişimlerinde daha başarılı sonuçlar aldığı gözlemlenmiştir. Bu durum onların yaptıkları işten keyif almalarına ve derse daha çok bağlanmalarına katkı sağlamıştır. Ayrıca öğrencilerin kodlara aşina oldukça diğer arkadaşlarına akran desteği sağlama girişimlerinin arttığı gözlemlenmiştir.

R_K15: Kodlama yaparken doğru yapmaya çalıştım ve yaptım.

C_K35: Bazen kodlamada zorluk çekiyorum, zorlanmadığım günler de oluyor.

C_K27: Mesela şifrelemeyi öğrendim ve kodlamada iyi olduğumu düşünüyorum.

C_K33: Çözüme yazım hatası almadan ulaşmak.

R_K49: Artık yavaş yavaş öğrendiğimiz için artık kodları daha kolay yazabiliyorum.

C_K70: Kodları doğru yazdığım için programım ilk seferde çalıştı ve ben de yapamayan arkadaşlarıma yardım ettim.

Öğrenciler kodlara aşina oldukça ve yeni kavramlar öğrendikçe (döngü ve alt programlar) hazırladıkları programlardaki kod yoğunluğunu azaltmaya, programların seri ve hızlı bir şekilde çalışmasını sağlamaya yöneldikleri görülmüştür. Kod yoğunluğunu düşük tutarak programlama yapmak öğrencilerin zaman kazanmasını da sağlamıştır.

R_K12: Alt programlarda zorlandım ve mantık yürütemediğim için alt programda daha uzun komutlar yazdığımı anladım. Gerçekte daha kısa komutlar yazılıyormuş.

C_K31: Bu fonksiyonlar yazdığımız kodları en kısa hale getiriyor.

C_K29: Kodlamalar kısaltılabilir, [böylece] daha seri çalışır.

R_K46: Döngüde aynı yazıdan 5 kere yazıyorsak uzatmaya gerek yok, tekrarla 5 diyip yazabiliriz.

Öğrenciler programlama etkinlikleriyle uğraşırken bireysel, eşli ve işbirlikli programlama yöntemlerini kullanmıştır. Robot programlama etkinliklerinde işbirlikli ve eşli; görsel programlama etkinliklerinde ise bireysel ve eşli programlamanın daha sık kullanıldığı gözlemlenmiştir. İşbirlikli programlamada öğrenciler arasında gerçekleşen fikir alışverişinin görevlerin daha kısa sürede tamamlanmasına yardımcı olduğu belirtilmiştir. İşbirlikli programlamanın farklı çözüm önerilerinin geliştirilmesinde ve hata düzeltme süreçlerinde faydalı olduğu belirtilmiştir. Bununla birlikte işbirlikli programlama etkinliklerinde fikir mülkiyeti, farklı fikirlerin karmaşaya yol açması ve bireysel performansların düşmesi gibi olumsuz durumların oluşabildiği görülmüştür.

R_K8: Hocam tek yaptığın [yaptığımız] zaman, zaman kaybedebiliriz. Ama grupla çalıştığımız zaman onlardan fikirlerini alırız.

R_K12: 2. Soruyu da kendim fikir sunarak yaptık ve arkadaşımın birisi PC'de yazdığı için [kod yazdığı için] hepsini kendim yaptım demesi ve siz hiçbir şey yapmadınız demesi kötüydü.

C_K30: Mesela hocam bir yanlışımız olduğu zaman, arkadaşımız söylediğinde bir dahaki yanlışımız olduğunda da arkadaşımız yine söyler deyip umursamıyoruz. Oradaki yanlış aklımızda kalmıyor.

C_K31: Olumlu yönleri bilen birisinin bilmeyene yardım etmesi. Olumsuz yönü bilmeyen kişinin uğraşmayıp kolaya kaçması, dersi dinlemeyip arkadaşına yaptırması. Bireysel olarak bilen kişi bilmeyenden [hiç] yardım alamaz. Ben yapamadığım zaman bilen birine sorsam işim kolaylaşır.

Eşli programlama etkinlikleri esnasında kod yazan (driver) ve gözlemleyen (gözlemci) arasında gerçekleşen etkileşimlerin genellikle görev performansları üzerinde önemli bir katkısının olduğu görülmüştür.

R_K45: Benim bilmediğim mesela hocam arkadaşım bildi, Onun bilmediğini ben bildim. İkimizin birleşiminde hocam ortaya bir şeyler çıktı.

R_K44: Hocam R_K42 hep yazım hatası yapıyordu. İlla birinin ona o hataların yapılmaması gerektiğini söyleyip başında durması gerekiyordu.

Öğrencilerin bireysel programlamada görevlere daha kolay odaklandıkları ve görevi başarmaları halinde daha yüksek bir özgüven duygusu yaşadıkları belirlenmiştir.

C_K34: Tek başıma hesap programı yapmak, bu bana özgüven verdi.

C_K36: Bireysel olmak daha sakin oluyor, fakat daha zor oluyor.

C_K29: Hocam tek kişi kullandığımızda şöyle bir faydası oluyor. Kendi hatamızı kendimiz bulduğumuz zaman hata bulma becerimiz gelişiyor hocam.

C_K72: Hocam yanlışımızı kendimiz görüyoruz.

Hatalar. Bilgisayımsal problem çözme etkinlikleri her iki grup için de öğrencilerin hata yaptıkları, hatalar üzerinde düşündükleri ve hatalarından öğrendikleri bir süreç olmuştur. Ayrıca BD etkinliklerinin problem çözme sürecinde öğrenci özerkliğini arttırdığı, özellikle programlama araçlarının sunduğu anlık hata

denetiminin öğrencilerin çözüm çabalarının kontrolü için öğretmene duyduğu ihtiyacı azalttığı belirlenmiştir.

> *R_K19: 2. soruda çok uğraştım. Çok çabaladım ama yine de cevap yanlış çıktı, sorunlara bakıyorum ya da nedenlerine, her şey düzgündü yani bana göre ama bir yanlış vardı. Bu da problemi çözme aşamasında kötü oldu.*
>
> *C_K64: Break yazmayı unuttuğum programın hatasına baktığımda üstüne çift tıkladım, hatanın nerede olduğunu gösterdi, gösterdiği yer boştu, biraz düşündüm ne eksik diye, diğerlerine baktım, hepsinde break vardı ama sadece onda yoktu, break yazdım hata yeri 0 (sıfır) oldu ve çalıştı.*
>
> *C_K67: Mesela hocam ben buraya gelmeden önce bu soruyu yapmadan önce hiç uğraşmıyordum. Doğrudan hocaya gösteriyordum bu ne diye. Ama burada [Programlama Temelleri dersinde] önce kendin uğraşıyorsun, yapmaya çalışıyorsun, alttan bakıyorsun [hata bulma penceresi] mesela ünlemli [uyarı alanları] yer neresi, nereye neyi katmışım filan, öyle bence çok faydası oluyor.*

Programlama hatalara açık bir süreçtir. Bu süreçte öğrencilerin karşılaştıkları hatalarla başa çıkması gerekmektedir. Öğrencilerin program yazma sürecinde ya da programın derlenmesi sonrasında ortaya çıkan hataları düzeltmeleri söz konusu hataları tespit etmelerine bağlıdır. Programlama araçlarının hata yakalama özelliklerinin öğrencilerin hataları tespit etmesinde oldukça faydalı olduğu anlaşılmıştır. Hem Robot C hem de C# programı gelişmiş hata editörlerine sahiptir. Hata penceresi yazılan program içerisindeki hataların konumunu, hatanın türünü, hatanın neden kaynaklandığına ilişkin açıklamaları ve hatanın düzeltilmesine yönelik ipucu bilgilerini içermektedir. Her iki grupta da hata penceresindeki bilgilerin işlevsel olarak kullanıldığı tespit edilmiştir.

> *R_K8: Robotu bilgisayardan kodlarken nerede yanlışlık yaptığımı öğrendim.*
>
> *C_K25: Hata verdiği için daha kolay oluyor [Flowchart'a göre].*
>
> *C_K26: Hocam, programı çalıştırıyoruz, orda nerede hatamızın olup olmadığını gösteriyor.*

Hata penceresindeki uyarılara tıklanınca imlecin hatanın bulunduğu yere gitmesi de hatanın anlaşılması için önemli bir sağlarlık (affordance) yaratmıştır.

R_K53: Aslında hataları gösteriyordu [hata penceresi] yazarken tıklıyorduk.

C_K77: Altta error yazısı veriyor, ona tıklıyoruz. Oraya mavi ekran oluyor zaten. Düzeltiyoruz onu.

Hata penceresindeki uyarıların yabancı dilde olmasının hataların anlaşılmasının önündeki en önemli engellerden birisi olduğu görülmüştür.

C_K28: Hata mesajları İngilizce olduğundan anlamıyorum.

C_K36: Hata mesajlarını bilmiyorum, anlaşılmıyor.

Robot C ve C# programlama araçlarında kod yazmanın metin-temelli olması öğrencilerin çok fazla sözdizimi hatası yapmasına yol açmıştır. Bununla birlikte C# araç kutusundaki hazır nesneler (buton, metin kutusu vs.) kod yazma işlemini nispeten kolaylaştırmıştır.

C_K23: ToolBox olduğu için kolay oluyor. Bazı hatalar çıkıyor ama onların da üstesinden geliyorum.

Öğrencilerin en sık yaptığı sözdizimi hataları yazım yanlışı (Türkçe karakter, büyük-küçük harf, harf hatası), parantez hataları, değer girişi hataları, değişken hataları (değişken ismi ve veri türü hataları), noktalama işareti hataları ve işaret hatalarıdır (büyüktür, küçüktür vs.).

R_K19: Yani kodlamaları doğru yapıyordum ama ufak bir yerde yanlış çıkıyordu. Bozuluyordu.

C_K21: Boşlukta bile hata oluyor. Boşluk bıraksak bile hata oluyor.

C_K33: Arada sırada yaptım yazım hataları, en ufak yazım hatasında program çalışmıyor ve dikkatim dağılıyor.

C_K36: Hocam nokta koymazsak bile hata oluyordu.

R_K50: Fazla süslü parantez kullandığımız için programın çalışmaması.

Öğrencilerin zaman baskısı ve rekabet gibi nedenlerden ötürü kodları hızlı yazmaları sözdizimi hatalarının artmasına yol açmıştır. Sözdizimi hatalarını gidermeye çalışırken bazı karakterleri (parantez ve noktalama işaretleri gibi) silme ya da yerini değiştirme, hataları daha da arttırmıştır. Ayrıca kod uzunluğunun fazla olduğu ya da karmaşıklık algısının arttığı durumlarda öğrencilerin sıkıldığı ve dikkatlerinin dağıldığı; bunun da hataların artmasına yol açtığı raporlanmıştır.

R_K6: Çok heyecanlıydık, ilk bizim bitirmemiz lazımdı. Çok istiyorduk ama heyecanlıydık, o bizi yanılttı ama yine de iyi yaptığımızı düşünüyorum.

C_K65: Hızlı bir biçimde yapmaya çalıştım, fakat bu hızın mağduru oldum ve çok hata yaptım.

C_K31: 3 tane hata varken o hataları düzeltiyoruz, 6 hata oluyor. 6 taneyi düzeltiyoruz, 12 oluyor.

C_K41: [hata sayısının] 36 çıktığını gördüm ben.

C_K33: Fonksiyonlar ve yazım hataları, çünkü çok karmaşık. Dikkatin dağıldığı zamanlarda birçok yazım yanlışı oluyor.

C_K72: Program yazmak çok güzel ama bazen kodlar uzun oluyor ve ben de açıkçası sıkılıyorum ve kodları yarıda bırakıp teneffüse çıktığım oluyor.

Hataların giderilmemesi ya da bir türlü fark edilememesi zaman zaman öğrencilerin karşılaştıkları hataları gidermek için çaba göstermemesine neden olmuştur.

C_K40: C#, onda sorun yaşadım, kodları yazıyorum, hata verdi mi sorun yaşıyorum, hatalarla uğraşamıyorum.

Durum çalışmalarının başında öğrencilerin hata düzeltme (ayıklama) için kullandıkları temel strateji, kodları silip baştan yazmak olmuştur. İlerleyen zamanlarda ise hata düzeltme araçları yardımıyla hata düzeltmede daha sistematik yollar izlemişlerdir.

R_K55: 3. soruyu 4-5 defa sil, tekrar yaz yaptık. Parantez eksiltip çoğalttık. Sonra sorunun 1 hatası çıktı, o da parantez işaretini eksik katmıştık.

C_K71: Bilgisayardaki hatayı bir türlü çözemedim. Silip bir daha yazmam en kötü şeydi.

Öğrencilerin yaptığı sözdizimi hatalarının zamanla azaldığı saptanmıştır. Bu durumun sözdizimsel kurallara ve program kodlarına aşina olunmasının ve hata giderme çabalarının daha sistematik gerçekleştirilmesinin bir sonucu olduğu düşünülmektedir.

C_K33: Matematikte çok örneğin yanlışım oluyordu harf hataları, şimdi daha az hata yapıyorum.

R_K42: If'in açıklamasını yazdıktan sonra noktalı virgül koyduğum için program hata veriyor ama öğrendim, bir daha if'e noktalı virgül koyulmayacağını öğrendim.

C_K65: Konuları zamanında ilerlettik. Her şey düzgün gidiyor. Mesela hocam bize 3-4 tane ödev verdi hatalarım vardı, ama fark ediyordum hatalarımı çözüm üretiyorum.

Öğrencilerin sıklıkla yaptıkları bir diğer hata türü ise mantık hatalarıdır. Mantık hatalarının giderilmesinde oldukça zorlanıldığı görülmüştür.

R_K9: Yazılım programı bazı hatalar veriyordu. Hocam mesela robotu programlarken o yazdığımız programlarda hata oluyordu bazen. Aslında hata olmasına rağmen orada olmadı olarak gözüküyordu [mantıksal hataların farkına varıyorlar].

C_K31: Hata bulmada biz bir kod yazdığımızda doğruysa çalışıyorsa neden bu programı yapan adam buna bir çare bulmamış, uyarı vermiyor, bizden tam olarak neyi istediğini anlayamıyorum.

Deney grubundaki öğrenciler için kodların robotlar üzerinde işletilmesi ve gerçek hayatta fiziksel çıktılar üretmesi mantık hatalarının daha kolay fark edilmesini sağlamıştır.

R_K57: Hataları bizim gözümüzün önüne seriyordu. Mesela biz yanlış bir değişken kullandığımızda robot bizim verdiğimiz yanlış değişkene göre hareket ediyordu. Ama ben fark ediyordum ki robotun hareket edişine göre demek ki bu değişken olmamalı. Başka bir şey kullanmalıyım. Yani hataları gözümüzün önüne sunuyordu.

R_K61: Aynı kodları yazıyoruz ya hocam, misal bir yanlış olsa hemen robota yapmadan önce hemen hepsini silerdik, ama robotun üstünde denediğimiz zaman o yanlış yeri görüyoruz.

Zaman yönetimi. Problem çözme etkinlikleri esnasında öğrencilere ortalama 40-60 dk. arası bir süre verilmiştir. Bazı öğrenciler zaman kısıtlaması olmasının kendilerini baskı altına soktuğunu ve bunun görev performanslarını etkilediğini belirtmiştir. Zaman baskısını özellikle deney grubu öğrencilerinin hissettiği görülmüştür. Hem programlama yapmanın hem de robot üzerinde deneme

yapmanın bu gruptaki öğrencilere zaman kaybettirmesinden dolayı bu durumun ortaya çıktığı düşünülmektedir.

> *R_K4: Üçüncü soruyu yapıyorduk hocam, zaman yetmedi diye bıraktık [zaman baskısı nedeniyle bırakma]. Normalde başaramadığımız soru yoktu.*
>
> *R_K9: Bugün else ve if kodlamalarını öğrendim. Ama dediğim gibi bugün vakit sıkıntımız olduğu için dersten fazla yararlanamadık.*
>
> *R_K18: 2., 3. ve 4. soruları [yapamadık]. Çünkü zaman ve heyecandan [dolayı] yapamadık.*

Deney grubundaki öğrenciler, takım çalışmalarında görevleri sırayla ve diğer takım üyelerinden bağımsız gerçekleştirmenin kendilerine zaman kaybettirdiğini, aynı görevler içerisinde görev paylaşımı ve işbirliği yapmalarının ise zaman kazandırdığını ve görev başarılarını arttırdığını belirtmiştir.

> *R_K19: Diğer program gibi dikdörtgeni yapamadık, benimle takım arkadaşım. Aslında sıra ondaydı, yani o yapacaktı. Ben yapayım dedim, başaramadım.*
>
> *R_K12: Hocam, grupta işte Polya'nın adımlarını filan yapıyorduk arkadaşlarla. Mesela ben yetiştiremiyorsam bir arkadaş bunu yazıyordu. Ya da algoritmayı oluşturuyordu arkadaş, sonra sıra kimde ise ilk soruyu yapacak arkadaş algoritmayı yapıyordu, sonra diğer arkadaş kodları yazıyordu, işte iş bölümü yapıyorduk. Diğer kişi de işte Polya'nın adımlarını uyguluyordu [yönlendirme] öyle daha iyi oluyordu, süre yetiyordu yani.*

İşbirlikli çalışmalarda takım üyelerinin aylaklığı, zaman kaybettirici unsurlardan birisi olarak göze çarpmıştır. Takımların sık sık yeniden oluşturulması takım üyelerinin her hafta birbirlerine uyum sağlamasını gerektirmektedir. Öğrenciler bu durumun kendilerine zaman kaybettirdiğini ifade etmiştir.

> *R_K15: İyi yani, mutluydum, eğlenceli geçti. Bizim grup biraz konuşmayı sevdiğinden zamanı harcadık doğrusu.*

Problem çözme sürecinde bazı soruların farklı konu alanlarında ön bilgi gerektirmesi bu soruların daha geç cevaplanmasına neden olmuştur.

C_K33: Programlamada çevre daire hesaplamalar[ın]da iyi değilim, ama adım adım öğreniyorum. Soruları çok zaman sürse de yapıyorum. Ama bazen çok zor sorular oluyor ama yapmaya çalışıyorum.

Durum çalışmalarının ilerleyen haftalarında öğrencilerin zamanı daha etkili kullanmaya başladığı görülmüştür. Bilgisayımsal problem çözme sürecinde gerçekleştirilen işbirliklerinin öğrencilere zaman kazandırdığı saptanmıştır. Ek olarak programlama araçlarının kod tamamlama özelliklerinin de öğrencilere zaman kazandıran bir diğer unsur olduğu görülmüştür.

R_K20: Bu haftaki derste zamanla yarışmayı öğrendik.

C_K76: [bireysel çalışma] Daha uzun sürdü. Mesela hocam iki kişi çalışsaydık, mesela ben ile C_K65 aynı yerde otursaydık daha farklı olurdu. Ben tek başıma 10 dk.'da yapıyorsam C_K65 ile belki 5 dk.'da yapardım beraber olsaydık.

Araçlar. Mevcut araştırmada iki farklı programlama modalitesi karşılaştırılmıştır. Deney grubundaki öğrenciler, Lego Mindstorms eğitsel robot seti (9797) ve programlama araçlarını kullanırken; karşılaştırma grubundaki öğrenciler Visual Studio C# görsel programlama dilini kullanmıştır. Lego Mindstorms eğitsel robot seti sökülüp takılabilir lego parçaları, tuğla şeklinde programlanabilir bir mini bilgisayar, 3 adet interaktif servo motor, 4 sensör (1x ultrasonik, 2x dokunma, 1x ses ve 1x ışık sensörü) ve bir batarya dâhil olmak üzere söküp takılabilir toplam 437 parçadan oluşmaktadır. Lego Mindstorms NXT 2.0 öğrenciler için manipülatif bir robotik inşa seti sunmaktadır.

Deneysel uygulamaların ilk haftasında robot parçaları tanıtıldıktan sonra, öğrencilerden kılavuz yardımıyla robot montajı yapmaları istenmiştir. Öğrenciler robot montajını adım adım gerçekleştirmeye çalışmıştır. Genel olarak kılavuzun robot montajını oldukça kolay hale getirdiği ifade edilmiştir. Bununla birlikte öğrenciler kılavuzdaki bazı parçaların renklerinin verilmemesi karşısında robot parçalarını şekilsel olarak karşılaştırdıklarını, bunun zaman zaman hatalara yol açtığını ve kendilerine zaman kaybettirdiğini belirtmiştir. Öğrenciler parçaları bulmak ve birbiriyle birleştirmekten keyif alsalar da bazı montaj parçalarının renk bilgilerinin verilmemesinin önlemez hatalara yol açtığını belirtmiştir. Kısaca öğrenciler

kılavuzdaki parçaları birleştirme yönergelerinin daha açıklayıcı olması gerektiğini belirtmiştir.

> *R_K6: Kılavuz olmasaydı toplamamız çok daha zaman alırdı. Ama kılavuz sayesinde çok büyük zaman kazandık.*
>
> *R_K10: Parçaların yanlış takılması sıkıntı çıkardı, kılavuz anlaşılırdı. Parçaları birleştirirken zorlandık.*
>
> *R_K14: Robot yaparken parçaları birleştirmekte çok zorlandık, robot yapma kılavuzu anlaşılırdı ama bazı parçaların renkleri değişik olduğu için çok şaşırdık.*
>
> *R_K16: Robot yapma kılavuzunda mavi parçayı siyah olarak gösterdiği için zorluk yaşadım. Kılavuzda numara ile gösterilen parçaların olduğu yerlerde numaralandırılmış şekilde gösterilmesi gerekir.*

Robot parçalarının nereye takılacağını bilememe karşında bazı takımların robotları sökerek montaj sürecine ilk adımdan itibaren yeniden başladığı görülmüştür.

> *R_K13: Sadece hepsini yaptıktan sonra bir yeri yanlış olup o kadar emeği bozmak.*
>
> *R_K19: Kılavuzda parçalarda renkleri yazmıyordu, bu da robot yaparken bize dezavantaj oldu, hatta bazı yerlerde renk yüzünden söküp tekrar tasarlamak zorunda kaldık, bu da hem zaman kaybetmeye hem de bozup tekrar tasarlamaya mecbur bıraktı.*

Robot parçalarının küçük ve parça sayısının fazla olması bazı parçaların zor bulunmasına yol açmıştır. Öğrenciler robot parçalarının bir haznenin içinde sunulmasının montaj sırasında parçaların bulunmasını kolaylaştırdığını ve kaybolmasını önlediğini ifade etmiştir.

> *R_K10: Hiçbir sıkıntı yaşanmadı, sadece parçalar düştü.*
>
> *R_K11: Robot parçalarını kolay yaptık, zor olmadı, yani bu parçalar küçüktü, kolay kayboluyordu ama [set içinden] bulunması kolaydı.*

Öğrenciler robot montajı yapmanın el becerilerini geliştirdiğini ifade etmiştir. Montaj işinin tamamlanmasından sonra öğrenciler robotu test etmiştir. Bu esnada robot parçalarının işlevini daha iyi anladıklarını ifade etmişlerdir.

R_K16: Robot tasarlamada iyi olduğumu düşünüyorum. Robot tasarlamada robot parçalarını birleştirerek el becerimi geliştirdim.

R_K19: Eğlenceli tarafı sensör daha yapım aşamasındaydı, biz de deneme turu yaptık, elimizi sensöre attığımızda durmadı, çünkü daha ayarlamıştık bazı yerlerde hata olduğunda birbirimize kızar ama eğlenceli olarak kızardık, bu da dersin eğlenceli geçmesine neden oldu.

R_K45: Robotların işlevsel analizini yaptık.

Robotun montajının tamamlanmasının ve robota program yüklenmesinin ardından öğrenciler test parkurlarını da kendileri tasarlamak zorunda kalmıştır.

R_K2: Robotu programladığımız ortam iyiydi. Sadece deneme parkurumuz yoktu.

Robot sayısının sınırlı olması bazı öğrencilerin aynı robot üzerinde uygulama yapmasını zorunlu kılmıştır. R_K9, öğrenci başına bir robot olmasının çalışma verimini arttırabileceğini ve programlamayı daha keyifli hale getirebileceğini belirtmiştir. Öğrenciler fiziksel robotların yanı sıra yazdıkları kodları Robot C'deki sanal robota yükleyerek ortaya nasıl bir sonuç çıkacağını test etmiştir. Robot programlama etkinlikleri esnasında çok nadir de olsa bazı teknik problemlerle karşılaşılmıştır. Bu problemlerin başında bilgisayar ile robot arasında iletişim kurulamaması ve robotun açılmaması sorunu gelmiştir.

R_K9: Her bireye 1 robot düşerse bence daha verimli ve zevkli olacağını düşünüyorum.

R_K55: Robot C'yi robotsuz da kullanmayı öğrendim.

R_K56: Robotumuz çalışmadı.

Deney grubundaki öğrenciler Robot C programlama aracını kullanmıştır. Bununla birlikte öğrencilere Mindstorms NXT yazılımı (ver. 2.0) da tanıtılmıştır. Mindstorms NXT yazılımı blok-temelli görsel bir programlama aracıdır. Bu araçta programlar kod bloklarının sürükle-bırak esasına göre birbirine kenetlenmesiyle oluşturulmaktadır. Ayrıca kodlar görsel olarak temsil edilmekte ve görseller kodların

işlevleri hakkında kullanıcılara fikir vermektedir. Mindstorms NXT yazılımında robot yapı parçalarının ve kod bloklarının nasıl bir araya getirileceğine ilişkin resim, video ve animasyonların yanı sıra örnek kodlamaların bulunduğu bir panel de bulunmaktadır.

R_K10: İşaretler resimliydi, o yüzden kolaydı.

R_K1: NXT bana göre daha iyi, yazmayla [kod yazmayla] uğraşmıyoruz. Ses ise ses katıyoruz [sürükle bırak]. Bana göre NXT daha kolay.

Öğrenciler Robot C'nin Mindstorms NXT'ye kıyasla daha gelişmiş bir araç olduğunu ve robot hareketlerinin daha ayrıntılı olarak programlanmasına olanak tanıdığını belirtmiştir. Öte yandan, öğrenciler Mindstorms NXT yazılımını Robot C'ye göre daha anlaşılır ve kolay bulduklarını ifade etmiştir. Deney ve karşılaştırma grubu arasındaki kod yazma koşullarından kaynaklanacak (blok ve sözdizimsel temelli gibi) bir farklılığı en asgari düzeye indirgemek adına programlama etkinliklerine metin temelli Robot C aracıyla devam edilmesine karar verilmiştir.

R_K10: [Robot C ile] Robot programlamayı en ince ayrıntıya kadar yapıyoruz.

R_K3: Robot C ona göre biraz daha zordu. Çünkü NXT'de sürükleyip yaptığımız için daha iyi oluyordu.

R_K12: NXT görsel olduğu için biraz zayıf kalıyordu, fakat robot C daha açıklayıcı bir yazılım programı olduğu için Robot C['yi] daha iyi anlayabiliriz.

R_K15: Daha önce hiç böyle şeyler, sorular çözmedik, o yüzden biraz zorlandık ama NXT programı daha iyiydi. Robot C'de zorlanıyoruz.

R_K19: Bence bu program [Robot C] iyi ama karışık, en ufak hata bütün [programı] tekrardan yapmamıza neden oluyor.

Aslında Lego Mindstorms eğitsel robot kitlerinin programlanması için harici bir programlama ortamının kullanılması zorunlu değildir. Robotlar aynı zamanda etkileşimli tuğla üzerinden de programlanabilmektedir. Ancak bu tür bir programlamada LCD ekranda aynı anda tek bir programlama adımı görünmektedir. Bu da özellikle karmaşık programlama görevleri için program adımları arasında bağ kurmakta zorluklar yaratmaktadır.

R_K5: İyi oldu, yoksa robot üzerinde yapsaydık [Lego Mindstorms LCD menüleri] çok zor olurdu.

Robotların hareket yönü ve uzunluğu motorların gücüne ve çalışma süresine göre belirlenmektedir. Robotların dış dünyayla iletişimi sensör, ekran ve ses komutları aracılığıyla gerçekleştirilmektedir. Öğrencilerin özellikle hareket uzunluğunu ve dönüş açısını ayarlama konusunda sıkıntı yaşadığı görülmüştür. Öğrenciler ayrıca ekranın bir koordinat düzlemi mantığıyla haritalandığını kavramakta zorlanmıştır.

R_K3: Uğraştığım halde başaramadığım soru, robotu daire çevresinde döndürmek. Soruları çözme sürecimde ise çok yol denedik ama olmadı.

R_K4: İki problemi yaparken zorlandım. Kare halinde hareket ettirmesi[ni], S halinde hareket ettirilmesini anlamakta zorlandım.

R_K17: 7. ve 8. soruda zorlandım. Kare oluşturamadım. Ayarlayamadım.

R_K15: Bazen anlamını bilmediğimiz kelimeler oldu ama çok iyiydi. 2. görevde biraz zorlandım, çünkü yüz ifadesini yapıyoruz, ama isim çıkmıyor üst tarafta [koordinat sistemi/zamanlama komutu kullanım bilgisi eksikliği]. 7. görevi yaparken çok zorlandım. Arkadaşlarımdan yardım aldım ama olmadı [Robotu kare biçiminde hareket ettirme].

R_K52: Robotun dönüş hareketlerindeki kodlamalarda sıkıntı yaşadık.

Öğrenciler robot hareketlerini gözlemlemelerinin robot hareketlerini kontrol etmede (dönüş açısı ve hareket uzunluğunu) oldukça işe yaradığını belirtmiştir.

R_K10: Ne tarafa doğru gittiği, kaç derecelik açıyla gittiği ve ne kadar hızlı gittiğine baktım.

R_K11: Robotun hızı fazlaydı. Kendisini test ettik. Sonra bitirdik [sorunu çözdük].

R_K15: Nerde ne kullanacağımızı, nerelerde saniye ve kaç saniye kullanacağımızı öğrendim, izledim robotu.

Öğrenciler robotlar için yazdıkları programların otantik dünyada işletilerek bir anlamda somutlaştırılmasının kodların daha iyi anlaşılmasını sağladığını belirtmiştir.

Ayrıca öğrencilerin hazırladıkları programları robot üzerinde uygulamalarının programlama etkinliklerine karşı ilgilerini arttırdığı görülmüştür.

R_K15: Sensör çalıştırmayı öğrendim. Hangi sensörün ne yaptığını öğrendim, çalıştırıp denedim.

R_K57: Dersimiz öyle hani sıkıcı bir ders değil ama yine de sürekli kod yazmak ve yazdığımız kodun bize nasıl bir geri dönüşü [olduğunu] görememiz bize sıkıcı gelebilirdi, ama robotlar hani o robotu bir an önce çalıştırma isteği, içimizdeki heves, dersin daha eğlenceli geçmesini sağladı.

R_K12 zaman içinde robotlara bakış açısının değiştiğini, onların programlanıp işlev kazandırılabilecek araçlar olduğunu kavradığını belirtmiştir.

R_K57: Önceden robotlara bakış açım çok farklıydı. Kendi kendine park eden otomobil duyduğumda çok şaşırırdım, bir araba kendi kendine nasıl park edebilir. Ama ben artık robotlarla çalıştıktan sonra robotu kendi kendine bir arabaymış gibi park edebiliyorum.

Deneysel uygulamaların başında deney grubundaki öğrencilerin ayrıcalık hissi yaşadığı görülmüştür. Bu durumun ortaya çıkmasında yenilik etkisinin önemli bir payı olduğu düşünülmektedir.

R_K12: Diğer gruplara göre, yani anlatılan konulara göre bizim bir ayrıcalığımız gözüküyor. Diğerleri mesela o programda görsel bir şey yapamıyor [fiziksel dünyada demek istiyor], biz görsel bir şey yaptığımız için değişik geliyor, yani güzel bir şey.

Karşılaştırma grubundaki öğrenciler programlama etkinliklerini Visual Studio geliştirme ortamında C# programlama diliyle gerçekleştirmiştir. Öğrenciler araç kutusu sayesinde nesneleri (buton, metin kutusu, liste kutusu, etiket vs.) sürükle-bırak yöntemiyle form ekranına kolayca yerleştirerek form tasarımı yapmıştır. Özellikle programların kullanıcı arayüzünü görselleştirme süreci bu gruptaki öğrencilerin oldukça hoşuna gitmiştir. Gelişmiş hata denetimi özelliğinin öğrencilerin kod hatalarını fark etmelerine ve düzeltmelerine yardımcı olduğu görülmüştür. Ayrıca kod tamamlama özelliği de kod hatalarının azalmasına önemli katkı sağlamıştır.

C_K26: Kolay anlıyorum, resimleri [görsel simgeler] gördüğümüz için.

C_K30: Kullanım kolaylığı, menüler [C#'taki araç kutusu öğeleri] zaten her şey alt alta. Kolay oluyor göz önünde.

C_K38: Bu program çok kolay biz iki kelime yazıyoruz devamını o getiriyor zaten.

Öğrenci çalışmaları. Bu bölüme ait bulgular işbirlikli çalışma ve bireysel çalışma olmak üzere 2 alt başlıkta sunulmuştur.

İşbirlikli çalışma. Robot tasarım ve programlama etkinlikleri işbirlikli çalışma için oldukça elverişlidir. Deney grubunda öğrenciler genellikle 3-4 kişiden oluşan takımlarda çalışmalarını sürdürmüştür. Öte yandan, karşılaştırma grubundaki öğrenciler için de işbirlikli çalışmalara olanak tanınmıştır.

R_K4: İlk parçaları görünce kesin anlarım demiştim, ama iskeleti bitince daha da merak sarmaya başlamıştım, ama orada yanlışlar olduğu için biraz üzüntü duydum, ama takım arkadaşlarımla beraber düzeltip tekrardan yapınca da hoş olduğunu anladım.

C_K31: Grup oluşturduk mesela C_K36'la biz. Ben soruyordum ona, o da bana söylüyordu.

C_K37: Bu derste if-else komutunu öğrendim. C_K31 arkadaşımla birlikte ortak çalışma yaptık.

İşbirlikli çalışmaların en önemli özelliği ortak bir amacın oluşması ve bu ortak amaca ulaşmak için birlikte çalışılmasıdır. Deney grubunda işbirlikli çalışma etkinlikleri karşılaştırma grubundan daha fazla gerçekleştirilmiştir. Bunun bir nedeni ekrana bağlı çalışmaların işbirlikli çalışmalara katılan kişi sayısını sınırlamasıdır. Karşılaştırma grubundaki ortak çalışmalar genellikle iki kişilik takımlar halinde gerçekleşmiştir. Bu çalışmalarda öğrencilerin eşli programlama yaklaşımını temele alan bir çalışma biçimi benimsediği görülmüştür. Ayrıca, karşılaştırma grubunda öğrenciler çalışmalarının büyük kısmını ekran üzerinden gerçekleştirirken, deney grubundaki öğrenciler robotun hazırlanması, programın robota yüklenmesi, robot için test parkurunun hazırlanması ve robotun test edilmesi gibi programlamadan farklı etkinlikler de gerçekleştirmiştir. İşbirlikli çalışmanın en önemli yanlarından birisi de akranların, birbirlerinin öğrenme süreçlerinde iskele vazifesi görmesidir.

R_K15: Mesela benim anlayamadığım yerde [takım arkadaşım] fikir veriyor, yapıyoruz. Onun anlamadığı yerde ben fikrimi belirtiyorum, yapıyoruz.

İşbirlikli çalışmalarda problemin masaya yatırılması ve üzerinde tartışılması takımdaki tüm öğrencilerin probleme odaklanmasını sağlamıştır. Bu durum problem çözme sürecinde çok çeşitli çözüm stratejilerinin üretilmesini, tartışılmasını ve kullanılmasını sağlamıştır.

R_K14: Bugün soruları çözerken birinci soruyu çok iyi yaptığımızı düşünüyorum. Grup arkadaşlarımla ilk önce tartıştık, sonra tahmin yürüttük ve başardık.

R_K18: Birbirimize yardım ediyoruz, fikirleri tartışıyoruz ve karar veriyoruz.

Fikir alışverişi esnasında her öğrencinin kendi fikrini savunmaya çalıştığı saptanmıştır. Zaman zaman takımların üretilen fikirleri irdelemeden uygulamaya geçirdiği görülmüştür.

R_K13: Takım çalışması olduğu için herkes kendi fikrini tartışıyor [savunuyor].

R_K20: İyi yaptığım şey takım arkadaşlarımla kendi fikirlerimi paylaşmak oldu.

R_K15: Memnun oldum, takımım iyiydi. R_K18 fikrini söylüyor, bazı yerlerde deniyorum, olmuyorsa değiştiriyorum, başka fikirler yapıyoruz oluyor.

Problemler karşısında doğrudan kod yazma çabasına girişilmesinin zaman zaman fikir alışverişini kısıtladığı görülmüştür.

R_K12: Hocam şimdi mesela herkes kişisel yapsa, mesela ben biliyorum arkadaşım bilmiyor, biliyor ama o kadar çok bilmiyor mesela o, kendi herkes bireysel yapsa başarabilir belki ama diğeri [kodları] yazarken aynı zamanda yaptığı için hiç fikir alışverişi yapmıyor, yani ben diyorum ki ya biz hiç fikir alışverişi yapmıyoruz.

Problem çözme sürecinde fikir alışverişi bilgilerin bir yerde buluşmasını sağlamıştır. Bu durumun problemin hem daha iyi anlaşılmasını sağladığı hem de çözümüne daha kısa sürede gerçekleştirilmesine olanak tanıdığı belirtilmiştir. Ayrıca işbirlikli çalışma öğrencilerin birbirlerinin hatalarını telafi etmelerine de yardımcı olmuştur.

R_K6: Bütün bilgileri bir yerde buluşturuyoruz. O bilgiler hem zaman tasarrufu sağlıyor hem de [problemi] daha iyi anlıyoruz.

R_K3: Takım arkadaşlarım ile yaptığımız için iyiydi. Çünkü herkes düşünüyor, hepimizin katkısı olunca daha iyi yapabiliyoruz.

R_K19: Olumlu yönleri çoktu, yani hata yaptığım yerlerde arkadaşlarım bana yardım ediyordu, yani hatamı telafi etmeme yardımcı oluyorlardı. Bana her zaman fikir sunuyorlardı, yani şu şöyle olmalı, bu böyle olmalı vs. gibi şeyler.

Akran düşüncesinin dikkate alınmasının ortak çalışmalarda fikirlerin rahatça ifade edilmesini önemli ölçüde etkilediği görülmüştür. Öte yandan, farklı fikirlerin zaman zaman karmaşaya yol açtığı ve bu fikirlerle başa çıkmanın zaman zaman bir gerilim durumu oluşturduğu görülmüştür.

R_K16: Takım olarak çalışmak iyiydi, ama herkes farklı bir şey söylediğinde kafamız karışıyor, robotu yanlış yapıyoruz.

R_K17: Arkadaşlarımla fikirlerimiz uyuşmadığı için robot dersinde çok sıkıldım.

R_K18: [iyi yaptığım şey] Önce soruyu anlamak, bir yol belirlemek ve o yolu takip etmek, bana yanlış yoldasın diyenlerle aynı yolda [çözüm önerilerini] karşılaştırma.

İşbirlikli çalışmalarda zaman zaman fikir mülkiyeti sorunları da yaşanmıştır. Bu durum kod yazan kişinin kendisini problemi çözen kişi gibi görmesinden kaynaklanmıştır.

R_K12: Hiç memnun olmadım. Bu haftaki derste takım halinde yaptığımız bir soruyu arkadaşımız PC'ye yazdığı için "kendim yaptım, siz bir şey bilmiyorsunuz" demesinden memnun değildim.

Deney grubunda takımların oluşturulmasında farklı stratejiler kullanılmıştır. İlk haftalarda öğrencilerin kendi takımlarını oluşturmasına ve takım arkadaşlarını belirlemesine olanak tanınmıştır. Bu durumda genellikle en iyi anlaşılan kişiyi seçme yaklaşımının benimsendiği görülmüştür. İlerleyen haftalarda gruplar görev performanslarına göre heterojen olacak şekilde yeniden oluşturulmuştur. Bazı haftalarda ise takımlar seçkisiz olarak oluşturulmuştur. Öğrencilerin takım

oluşturulmasına dair tercihlerinin birbirinden oldukça farklı olduğu görülmüştür. Bazı öğrenciler en iyi arkadaşlarıyla, bazıları kendi akademik başarı seviyelerine yakın öğrencilerle, bazıları ise farklı alanlarda yetenekleri olan öğrencilerle takım oluşturmak istemiştir.

> *R_K5: Her gruba dersi iyi olan birinin gelmesi bilmeyen arkadaşlarına öğretmesi.*
>
> *R_K9: Bana kalırsa [takımla çalışmanın] olumsuz yönü yok. Sadece ve sadece doğru arkadaşlarınla ve yanına yakın gördüğün [arkadaşlarınla]. Yani [arkadaşların] seninle aynı zekâda ve kapasitede olduğu sürece hiçbir sıkıntı olmaz diye düşünüyorum.*
>
> *R_K10: Eşit seviyede gruplar yapılmalıdır. Zekâ seviyesine göre belirlenmeli.*
>
> *R_K13: Bence herkes istediği [en iyi anlaştığı] kişilerle yaparken en azından eğlenceli yapar diye düşünüyorum.*
>
> *R_K49: Ben hep yapmaya çalışıyordum, çünkü benim takımımdaki arkadaşların bir şey yapmadıklarını düşünüyorum. Bu ders için yeterli bilgilerinin olması gerekir.*

İstenilen takımda yer alamamak (özellikle seçkisiz atama sonrası) bazı öğrencilerin problem çözme sürecinde isteksiz çalışmasına, kendisini çalışmalara verememesine ve derse karşı olumsuz tutum geliştirmesine yol açmıştır.

> *R_K4: İyi karşılarım ama grup değiştirmek istiyorum. Bir sorun olursa ben karışmıyorum, anlayıp bulsunlar, öğrensinler diye kenara çekiliyorum, hiçbir şey yapmıyorsun diyorlar. Cidden grup değiştirmek istiyorum.*
>
> *R_K17: Takım halinde çözmek istemiyorum ve istediğim gruba gidemiyorum. Bu yüzden keyif alamıyorum. Derse kendimi veremiyorum. İsteksiz yapıyorum her şeyi. Bence herkes kendi istediği takımla grup olsun. Bu yüzden de robot programlama dersinden gıcık almaya başlıyorum. Hiçbir şey yapasım gelmiyor.*

Takım üyelerinin çaba göstermemesinin takım değiştirme isteğinin altında yatan en önemli neden olduğu görülmüştür.

R_K19: Zaten belirli kişiler kodlamaları yazıyordu hocam. Diğerleri de bir nebze de olsa biraz yardım etmiyordu hocam bize, bunda da yapacak bir şey yoktu hocam, grubu anca değiştirebildik.

Takım çalışmalarında öğrenciler baş edilmesi gereken birçok sorunla karşılaşmıştır. Bazı takımlarda takım üyelerinden bazılarının problem çözme sürecinde işbirliğine kapalı olması ve görevleri kendi başına gerçekleştirmeye çalışması önemli bir sorun teşkil etmiştir.

R_K7: Kötüydü, çünkü arkadaşlar birbirleriyle konuşmasından dolayı yapamadık. Çünkü R_K9 arkadaşımız kimsenin yapmasına izin vermedi [R_K9 kendisi yaptı].

R_K12: Hocam "ben biliyorum siz karışmayın" falan diyorlar.

R_K60: Soruları çözemeyince anlamıyordum. Kodlamalarda zorlanıyordum. Tek başıma.

Bazı takım üyelerinin görevlerle ilgilenmemesi ve takıma yeterli katkıyı sağlamaması takımın görev performansını olumsuz etkilemiştir. Bu durum takımın diğer üyelerine fazla yük binmesine neden olmuştur. Takım içinde aylaklığın ya da sorumluluk almamanın en önemli nedenlerinden birisi de takımdaki herkesin eşit puan alacağına dair anlayıştan kaynaklandığı görülmüştür.

R_K12: Bugün kötü giden [şeyler], gayret göstermeyen arkadaşlarımız olduğu için soruların çoğu kötü gitti.

R_K15: Bizim grupta bir R_K16 iyiydi. R_K20 hiç katılmadı. Bir de R_K16 ile yaptım. Yaptık ama takım çalışması beraber yapılmalı. Mesela bugün R_K20 hiç yapmadı. Hep ben yaptım. R_K16 fikirlerini söyledi. Benim grubumda ben yapıyorum 70 alıyorum. R_K20 yapmıyor, benimle aynı notu alıyor.

R_K19: Gruptaki bir kişi bana katılmıyor, yani yardım etmiyor, başkalarıyla ilgileniyor, bu da beni ve gruptaki diğer arkadaşlarımı olumsuz yönde etkiliyor.

R_K62: Mesela hocam, R_K60, iki kişiyle hocam, sen yap diyorlar, biz de puan alalım diyorlar.

Takım içindeki bazı üyelerin takıma yeterli katkıyı vermemesi neticesinde görevlerin takım içinde aynı kişiler tarafından yerine getirilmesi bu kişilere fazla yük binmesine yol açmıştır. Sorumluluk yüklenen kişilerde bir baskı oluşmasına neden olan bu durum söz konusu öğrencilerin motivasyonlarının düşmesine, hatta başarısız girişimler sonucunda zaman zaman görevle uğraşmaktan vazgeçmelerine yol açmıştır.

> *R_K4: Ben takımımdan memnun değilim, her şeyi benden bekliyorlar, yapamayınca da bir şeye yaramıyorsun diyorlar [grup içi baskı]. Daha çok konuşmamız, anlaşmamız ve gruptaki herkesin katkı yapması gerektiğini düşünüyorum. Takım arkadaşım çok çabuk sıkıldığı için benim motivasyonum [da] çok az derecede de olsa düşüyor.*
>
> *R_K6: Takım halinde çözmeye çalışıyoruz ama arkadaşlarım sıkılıyor bırakıyor, ben de tek başıma yapmaya çalışıyorum. Yapamayınca sinirleniyorum. Takım çok kötüydü. Bu da benim canımı sıktı, dersi bir süre sonra bıraktım.*

Takım içinde sürekli aynı işi yapmanın bir sorun olduğu görülmüştür. Takım çalışmalarında başarısız girişimlerin de önemli bir gerilim unsuru olduğu saptanmıştır.

> *R_K58: Birisi sadece kabloyu takıyor [bilgisayardan robota program transferi] hocam.*
>
> *R_K1: Ortaklaşa yapmak çok güzel bir şey, arkadaşlarımla tartışıp [yapmak]. Memnun olmadığım şey yapamayınca sinirleniyoruz. Birbirimize kızıyoruz [takım içi gerilim].*

Takım kararlarının bazı bireysel girişimleri kısıtlaması da takım içindeki bir diğer çatışma noktasıdır.

> *R_K17: Robot yaparken gruplaşmak iyi fikir değildi. Bazı arkadaşlarım yapmaya çalışıyordu. Bazıları da yapmıyordu. İstediğim malzemeyi kullanamıyordum. Fikirlerimiz uyuşmadığı için ve takım olduğumuz için kötü performans sergilediğimizi düşünüyorum.*
>
> *R_K19: Kötü yaptığım şey sadece dikdörtgendi, o da takımımdaki arkadaşlarımın bana yardım etmemesinden [desteklememesinden]*

kaynaklandı, ben yapıyorum arkadaşlarım bozuyordu, bu da o programlamayı yapmamama engel oluyordu.

Takım içi yardımlaşma, takım uyumunu etkileyen faktörlerden birisidir. Takım içi yardımlaşmanın en önemli faydaları takım üyelerinin birbirinin eksiğini tamamlaması, yaptığı hataları düzeltmesine yardımcı olması, daha az bildiği ve daha fazla bildiği durumlarda öğrenme iskelesi kurulmasını sağlamasıdır.

R_K3: İyi şeyler vardı. Grup arkadaşlarımla, onların yardımıyla yaptık.

R_K14: Olumlu yönleri, ben bazen yapamadığımda onlardan yardım istiyorum, onlar da yapamadığı zaman benden. Grup halinde çalışmamız daha keyifli.

R_K60: En azından yaptığımız sorularda yapamadığım yerde destek çıkıyorlardı arkadaşlar.

Bir takım çalışmasında öğrencilerin diğer üyelerden beklentileri problem çözme sürecinde çaba göstermeleri, takıma katkı sağlamaları ve tıkanılan yerlerde takıma yol göstermeleridir. Deney grubundaki ortak çalışmalarda da aynı beklentilerin oluştuğu görülmüştür.

R_K5: [diğer üyelerden beklentilerin] Programa yardımcı olma, birkaç yanlışı söyleme ve yapma [düzeltme]. Sensörlerde yardımcı olma. Problem çözme sürecini raporlama.

C_K25: Bugün C_K22 ile dersimi yaptım. C_K22 derste bana hiç yardımcı olmadığı için bir iki yanlışım çıktı ama yaptım.

R_K57: [Takım arkadaşımın] en azından bir fikir yürütmesini isterdim. Ufak tefek de olsa bir katkısının olmasını veya dikkatli olmasını ve onu öğrenmeye çabalamasını isterdim.

R_K59: [Takım arkadaşımın] derste faydalı olmasını veya bize yardımcı olmasını isterdim.

Takımdaki öğrenci sayısının da takım uyumunu etkileyen önemli bir faktör olduğu görülmüştür. Öğrenciler genel olarak takımdaki üye sayısının 3-4 ile sınırlı tutulmasından memnun olduklarını belirtmiştir. Öte yandan üye sayısının 2'ye düştüğü durumlarda öğrencilerin daha fazla çaba harcadığı görülmüştür.

R_K10: Dört kişi takım halinde yapınca daha zevkli oluyor. Birbirimizin hatalarını düzeltiyoruz.

R_K12: Sınıfımız 2'ye bölündüğü için fiziki bakımından iyiydi. Kişi sayısına bakarsak her 3 kişiye 1 robot düşüyordu. 3 farklı takım olduk gayet iyiydi.

R_K17: Ben takımca robot yapma fikrini mantıklı bulmadım. Öğrenci sayısı az olsaydı daha çok anlaşılırdı her şey.

R_K20: Bugün yaptığımız gibi ikili grup 2. kişi olunca ben de yapacağım bari gayret göstereyim dedim.

Takım içinde çalışmaların organize yürümemesi takım uyumunu ve takımın görev performansını olumsuz etkilemiştir.

R_K6: Bugün programlama temelleri dersinde çok sinirlendim. Çünkü biz daha 1. soruyu yapmadan diğer gruplar 2. soruya geçmişti ve hala bizim gruptaki arkadaşlar kafalarına göre davranıyorlardı.

Takım kararlarının takım içi organizasyonunu sağlayan en önemli etmen olduğu görülmüştür. Takım kararları alındığında öğrencilerin organize bir şekilde hareket ettiği ve çalışmalarını daha planlı bir şekilde sürdürdüğü gözlemlenmiştir. Bunun da çalışma verimini olumlu şekilde arttırdığı belirlenmiştir.

R_K4: Memnun olmadığım bir şey yok, görev dağılımı da iyi ve her şeyi danışarak ve konuşarak yaptık.

R_K8: Aramızda toparlanıp işbirliğini iyi yapmamız, ağız birliği yapmamız, daha iyi bilgiler bulmamız ve verimliliğimizi iyi yapmamız gerekli.

Zaman zaman öğrencilerin takım çalışmasına uyum sağlamakta zorlandığı ve görevleri tek başına yürüttüğü görülmüştür. Bu durumun ortaya çıkmasının bir nedeni de takım içindeki rekabettir. Ek olarak, görev listesindeki problemlerin de zaman zaman takım üyelerine paylaştırılıp, ayrık olarak çözülmesinin de takım içi rekabet koşullarının oluşmasına katkı sağladığı görülmüştür.

R_K6: İlk soruyu tek başıma yapacaktım. Kodu yazdım, sadece iki yerde hata vardı. O da çok basit hataydı. Düzeltmeye kalmadı, hocamız bana tek başına yapma dedi, arkadaşlarınla beraber yap dedi. Ben de sinirlendim, yazdığım kodları sildim, çıktım.

R_K20: Var. Soruyu tek çözmeyi de öğrendim. Yani kimseden yardım almadan tek başıma soruyu çözdüm.

R_K6: Bu haftaki derste geçen hafta topladığımız Lego robotlara yazılım yükledik. Çok güzeldi. Eğlenceliydi ve gruptaki arkadaşlarla mücadele içerisindeydik, mükemmeldi.

Takım arkadaşlarını dikkate alma, takım yönetimini etkileyen bir diğer unsurdur. Bazı öğrencilerin diğer arkadaşlarının uyarılarını umursamamasının takım içi organizasyonunu olumsuz etkilediği sıkça ifade edilmiştir.

R_K58: Tabi ki hocam ama [bizi] dinlemiyor ki.

R_K60: [diğer arkadaşlarının uyarısını] Aldı ama uygulamadı.

Takım üyelerinin tutumları ve davranışları da takım uyumunu etkileyen bir diğer unsur olarak göze çarpmıştır.

R_K18: Memnun olduğum beraber ve tartışarak yapmak, memnun olmadığım bilmişlik taslamak, yani başka da bir şey yok.

Öğrencilerin sıklıkla diğer takımları gözlemlediği ve görev listesindeki görevlerde ilerleme durumlarını diğer takımlarla kıyasladıkları görülmüştür. Öğrenciler zaman zaman diğer takımlardan destek istemiştir.

R_K47: 4. soruyu bizim grup dâhil hiç bir grup yapamadı, yani bir tek kötü giden şey 4. soru oldu.

R_K57: Deftere yazdıklarım, notlarımdan yardım aldım ve defterimin yeterli olmadığı yerde grup arkadaşlarımdan, öteki gruptaki arkadaşlardan ve hocadan yardım aldım. Bu yardımların sonucunda robotu programlayabildik.

Takımlarda farklı görev paylaşım örüntüleriyle karşılaşılmıştır. Bunlar; görev listesindeki problemlerin takım üyelerine paylaştırılması (görevin ayrık yürütülmesi), aynı probleme ait işlem adımlarının takım üyelerine paylaştırılması (aynı göreve ait işlerin ayrık yürütülmesi) ve son olarak görevlerin bir kişi öncülüğünde yürütülmesidir. Bununla birlikte her üç görev paylaşım türünde gerektiği yerde takım içi yardımlaşma gerçekleştiği gözlemlenmiştir.

R_K2: Takım destek verdi. Bir kişi yaptı. Diğerleri bilgi verdi.

R_K3: Evet, herkes bu etkinliğe destek verdi, hepimiz farklı yerleri yaptık, karar vere vere.

R_K4: Herkesin ayrı ayrı görevleri vardı. Hata yapan arkadaşlarım birbirini destekliyordu. Örnek vererek düzeltiyorduk.

R_K12: Herkes sırayla yaptığı için, ben kendime gelen soruları doğru yaptığım için mutluyum.

R_K16: İkinci problemi ben yaptım, iyi yaptığımı düşünüyorum, takımdaki arkadaşlarımla tartışarak yaptığım için işim biraz daha kolaylaştı.

Öğrenciler görevlerin ayrık yürütülmesinin kendilerine zaman kaybettirdiğini belirtmiştir. Ayrıca ayrık olarak çözülen problemlerde takım üyelerinin sorumluluğundaki görevleri gerçekleştirememeleri takım performansını olumsuz etkilemiştir. Bu durumda öğrencilerin kendi aralarında tekrardan görev paylaşımı yaptığı görülmüştür.

R_K7: Öyle de olmuyor, yapmıyorlar. Ben bilmiyorum deyip, geçip gidiyorlar.

R_K12: Takım ile çalışırken el birliğiyle yaparak iyi bir iş çıkarttık. Olumsuz olarak sırayla yaptığımız için çok zaman kaybettik.

R_K14: Hocam biz mesela grup arkadaşlarımızla, mesela ben birinci soruyu yapıyorsam, o ikinci soruyu yapıyordu. Üçüncü soruyu başka birisi yapıyor. Öyle soru alışverişi yapıyorduk. Öyle daha iyi anlaşıyorduk.

Aynı göreve ait işlerin farklı kişiler tarafından yürütülmesi esnasında her problem çözme adımı bir önceki adımın üzerine inşa edilmiştir. Bu tür bir görev paylaşımında işbirliğinin diğer görev paylaşma biçimlerinden daha yüksek düzeyde olduğu görülmüştür.

R_K5: Evet takım olarak yaptığımızda herkes bir şeyi yapıyor ve birleştiriyoruz.

Görevlerin tek bir kişi öncülüğünde ilerlemesi zaman zaman bu kişinin farklı fikirlere kendisini kapatmasına ve görevleri kendisinin gerçekleştirmesine yol açmıştır.

R_K17: Grupça yapamıyorum. Takım arkadaşım soru çözemedi, ben çözdüm. Bu yüzden yolunda gitmiyor.

Bireysel çalışma. Bireysel çalışma öğrencilerin tercihleri doğrultusunda gerçekleşmiştir. Deney grubuyla karşılaştırıldığında kontrol grubunda daha fazla görülse de bireysel çalışmalara nispeten nadir rastlanmıştır. Deney grubundaki öğrencilerin zaman zaman görevleri kendi aralarında paylaşarak bireysel çalışma yapmayı tercih ettikleri görülmüştür. Bireysel çalışmayı tercih eden öğrencilerin problemi çözmekte zorlandıkları anda işbirliği arayışına girdiği görülmüştür.

C_K21: Bireysel olarak yapamıyorum. Sorunlarım, yanlışlarım oluyor. Takım halinde yapınca daha iyi oluyor. Sorunlarım ortadan kalkıyor.

C_K39: Bireysel olarak yapsak iyi olmaz. Yapamadığımız yerler olursa arkadaş yardımıyla düzeltebiliriz. Olumlu olarak ise aklımızı daha iyi veririz ve daha iyi anlayabiliriz.

Bireysel çalışmaların daha yüksek düzeyde öğrenci özerkliği sağladığı görülmüştür. Ayrıca bireysel çalışmaların sorulara daha iyi odaklanılmasını sağladığı belirtilmiştir.

C_K28: Bireysel çalışmak daha iyi oluyor. Bir konuyu kendi çabanla yapıyorsun.

C_K35: Bireysel olumlu yönü kendim çabalayıp kendimi zorlayarak yapar isek daha başarılı ve daha yardımcı olur diye düşünüyorum.

R_K20: Bugün derste ilk defa bir soruyu kendim yaptım. Daha iyi olacağıma inanıyorum.

R_K45: Bugün de kendi sorularımı kendim çözdüm.

Evinde bilgisayar olan öğrencilerin özgüvenlerinin ve tek başına çalışma becerilerinin daha yüksek olduğu görülmüştür.

C_K37: Bilgisayardan çalışmama evde oturup çalışıyorum ama yeni programa geçtiğimiz için bunu da yükleyip çalışacağım, kendimden eminim bilgisayar konusunda.

Problem çözme sürecinde destek bulamayan öğrencilerin zorunlu olarak bireysel çalışmaya yöneldiği görülmüştür.

C_K69: Öğretmenden ya da daha iyi anlayabilen birisinden yardım isterim, eğer kimse yardım etmezse kendim çözmeye başlarım.

Bireysel çalışmaların öz-planlama ve bireysel çalışma becerisini geliştirdiği ifade edilmiştir.

C_K67: Bence hocam, bu sizin öğrettiğiniz programlama dersinin bütün derslere faydası var. Hocam şimdi nasıl diyeyim. Planlı çalışma, tek başımıza çalışma öğreniyoruz, ondan sonra nasıl diyeyim bir soru[nla] karşılaştığımızda tek başımıza yapmayı öğreniyoruz.

Destekler ve engeller. Öğrenme desteği öğrenci başarısının en önemli belirleyicilerindendir. Öğrencilerin, başarılı olmak için neyin gerektiğini, bunun önündeki engellerin neler olduğunu ve bu süreçte ihtiyaç duydukları destekleri nereden sağlayabileceklerini bilmeleri öğrenme hedeflerine ulaşmalarında kilit bir öneme sahiptir. Bu bölümde bilgisayımsal problem çözme sürecinde başvurulan destek kaynaklarına ve bilgisayımsal becerilerin sergilenmesini zorlaştıran unsurlara yer verilmektedir.

Destekler. Öğrencilerin üstesinden gelemedikleri zorluklar karşında ilk başvurdukları kaynakların sınıftaki diğer öğrenciler ve ders öğretmeni olduğu görülmüştür. Bazı öğrencilerin problem çözme sürecinde diğer arkadaşlarına rehberlik ettiği görülmüştür. İşbirlikli çalışmalarda öğrencilerin birbirlerinin eksiğini tamamladığı ve öğrenmelerine destek olduğu görülmüştür.

R_K56: Yapamadığım sorularım oldu, ama arkadaşlarım biraz yardım etti, biraz olsun anlıyorum.

C_K67: Mesela bir şeyi tek başımıza yapamadığımızda, bir soruna takıldığımız zaman çoğu kez ya C_K64'e ya da C_K70'e soruyorduk, onlar da yapamıyorsa hocamıza soruyorduk.

R_K52: Hocam, birimizin eksik kaldığı yerde arkadaşımız tamamladı ve onun görmediği hataları onun yanındaki gördü, mesela yazılım hatalarını.

Her iki gruptaki öğrenciler de öğretmen desteği ve yönlendirmesine oldukça sık başvurmuştur. Öğrenciler özellikle uygulamaların ilk haftalarında çalışmalarını yüksek oranda öğretmene bağlı biçimde sürdürmüştür. Öğrenciler genellikle kod

yazma, kod hatalarını düzeltme ve teknik sorunları giderme için öğretmen desteğine başvurmuştur.

C_K75: Birazcık uğraşıyorum, yapamadığım zaman da hocayı çağırıyorum.

C_K78: Hocam mesela bir hatam olursa hocaya danışıyorum. O da hatalarımı düzeltiyor. İleriki zamanlarda karşıma çıktığı zaman daha rahat yapabiliyorum.

C_K79: Bazı yanlışlarım oldu, onu da hocaya gösterip düzelttim.

C_K77: Hocam bilgisayar dondu. Biz hocamızı çağırdık.

Öğrenciler kod yazma aşamasında bilgisayar tarafından yapılan uyarıların da kendilerini yönlendirdiğini belirtmiştir.

C_K24: Hocam biz sizin de dediğiniz gibi kâğıda yazsaydık, birçok hatamız olurdu. Bilgisayar bizi yönlendirdiği için o yönden şanslıyız.

Engeller. Programlama araçlarının yabancı dilde olması öğrencilerin menü seçeneklerini, uyarıları ve hata mesajlarını anlamasını zorlaştırmıştır.

R_K16: Program İngilizce olduğundan zorlandım. Türkçe olsa daha iyi olabilirdi.

C_K36: Programı kullanma deneyimi için [programı] kullanmak biraz zor, Türkçe olsa daha iyi olur, İngilizcede ne yaptığımı bilmiyorum.

C_K39: Anlaşılır değil [Visual] Studio programı, sürekli yabancı kelimelerden oluştuğundan çok iyi anlaşılmıyor.

Programlama konuları birbiriyle bağlantılı olup, belirli bir haftadaki problem çözme etkinlikleri önceki haftalardaki kazanımların da işe koşulmasını gerektirmektedir. Bu durum önceki haftalardaki kavramların yeterince anlaşılmadığı durumlarda öğrenciler açısından zorluk yaratmıştır. Ayrıca bazı bilgisayımsal problemlerin farklı alanlarda (örneğin Matematik) ön bilgi gerektirmesi karşılaşılan bir diğer engeldir.

R_K4: Her şey yolundaydı. 4.soru hariç. Baya zordu. Pi filan zorladı.

R_K12: Programlama yaparken matematik devreye girdiğinde yapamadığım için yardıma ihtiyaç duydum. İpucundan sonra kötü bir şey olmadı.

C_K37: Derste zorlandım, çünkü benden pozitif, negatif ve işaretsiz sayılarla ilgili soru soruldu.

Öğrencilerin bilgisayar okuryazarlık seviyesi ve klavye kullanım becerilerinin (kullanım hızı, ikinci ve üçüncül karakterleri çıkarma ve özel işlevli tuşları bilme) yeterli olmaması programlama etkinliklerini (özellikle kod yazımını) olumsuz etkileyen bir diğer unsur olmuştur. Her iki gruptaki öğrencilerin de zaman içerisinde bilgisayar ve klavye kullanımına aşinalık kazandığı ve klavye kullanma becerilerini geliştirdiği görülmüştür. Ayrıca sözdizimsel hataların da zamanla azaldığı görülmüştür.

C_K23: Klavyede bazı şeyleri bilmediğimden [özel karakterler, ikincil ve üçüncül simgeler] zor oluyor.

R_K15: Önceleri yabancı geliyordu programlar, şu an alıştık.

C_K40: Bana etkileri bilgisayar kullanmasını bilmiyordum. Klavye harflerini bilmiyordum. Proje açmayı bilmiyordum, öğrendim ben bunları.

C_K65: Dediği gibi süslü parantezmiş, noktalı virgülmüş filan, bunların hataları çoğunu gördük mü anlıyoruz artık.

Karşılaşılan teknik sıkıntılar (bilgisayarın yavaşlaması, donması ve robotun açılmaması) da problem çözme sürecini olumsuz etkilemiştir.

R_K47: Kötü giden şey ise ilk soruyu çözerken bizim grubun bilgisayarı yavaşladı, bazen donuyordu, biz de diğer masaya geçtik.

C_K72: Bilgisayarımın klavyesinde space tuşu, yani boşluk tuşu yoktu, kalemimle boşluk tuşuna basıyordum. Biraz zorlandım, ama ders güzeldi.

R_K60: [olumsuz olan] Robotun bazen arıza vermesi.

C_K77: Bazen o kadar uğraşıyoruz ki masa donunca [bilgisayar donunca] insan sinirden deliriyor, aklını veremiyor derse.

Öğrenciler bilgisayar laboratuvarındaki taburelerin birtakım ergonomik sıkıntılara (bel ve sırt ağrısı) yol açtığını belirtmiştir. Ayrıca bilgisayar masalarının U düzeninde olmasından dolayı dersin kuramsal kısmında akıllı tahta-bilgisayar masası arasında yapılan yönelimlerin boyun ağrısına yol açtığı ifade edilmiştir.

C_K32: Arkadaşlarım bazen canımı sıksa da en büyük olumsuzluk bence oturduğumuz taburelerin sağlıklı olmadığını düşünüyorum.

C_K65: Hocam tahtaya bakıyoruz ya böyle ikide bir önümüzü dönüp böyle yazıyoruz, o zaman boynumuz ağrıyor.

Devamsızlık ya da nöbetçi olma nedeniyle derse katılamama, bazı öğrencilerin ilgili haftadaki konuları öğrenmelerinde güçlük oluşturmuştur. Resmi tatil nedeniyle derslerin işlenmemesi öğrencilerin önceki öğrenmeleriyle yeni öğrenmeleri arasında bağ kurmasını zorlaştırmıştır. Yapılan konu tekrarlarının dersi kaçıran öğrencilerin derse bağlanmasına önemli ölçüde katkı sağladığı ifade edilmiştir. Devamsız gün sayısı fazla olan öğrencilerin sınıfta kalma korkusu yaşadığı için derse odaklanamadığı görülmüştür. Ek olarak, kişisel rahatsızlıklar (hastalık, yorgunluk ve uykusuzluk) öğrencilerin derse ilgisini olumsuz etkilemiştir.

R_K7: Ben o hafta nöbetçi olduğum için o uygulamayı bilmiyorum, ama bu haftaki uygulamayı daha iyi buluyorum.

C_K35: If-else dersinde olmadığım için sonradan öğrenmesi zor oldu. Bu derste her hafta çok önemli, bir dersi kaçırdığınızda tüm konular düşüyor.

R_K55: Geçen hafta tatildi, biz de çalışmadan geldik, sonra kötü gitti, eğer tekrar edip gelseydik iyi olurdu.

C_K70: Ben geçen hafta okulda olmadığım için derste fazla bir şey anlamadım, ama hocamız tekrar ettiği için güzel oldu ve anlamama yardımcı oldu.

C_K29: Yaşadığım sorunlar yüzünden derse pek odaklanamadım. Devamsızlık sınıfta kalma korkusu.

R_K9: Bugün çok halsizdim. Fiziksel ve zihinsel olarak çok yorgundum. O yüzden fazla bir şey öğrenemedim.

Duyuşsal boyut. Eğitim teknolojileri alanyazını incelendiğinde çalışmaların büyük bölümünün öğrenmenin bilişsel boyutu üzerinde durduğu görülmektedir. Bununla birlikte yapılan öğretimsel müdahalelere ilişkin toplanan nitel verilerin analizi sonucunda öğrencilerin duyuşsal özelliklerinin de öğrenme süreci ve görev performansı üzerinde etkili olduğu anlaşılmıştır. Bilgisayımsal problem çözme sürecinde öğrencilerin bilişsel desteklerin yanı sıra duyuşsal desteklere de ihtiyaç

duydukları belirlenmiştir. Problem çözme sürecinde her iki gruptaki öğrenciler de sıklıkla başarısızlık korkusu ile baş etmek zorunda kalmıştır. Bazı öğrencilerin problem çözme sürecinin başında görevlerde başarısız olacaklarına dair ön yargı geliştirdikleri görülmüştür. Bu ön yargının özellikle uygulamaların ilk haftalarında deney grubu öğrencilerinde daha sık oluştuğu saptanmıştır.

R_K15: İlk başta yapamayacağız diye korkmuştum, ama sonradan her şey yolunda gitti.

C_K21: Bir şeyleri yapınca kendime güvenim geldi. Güven problemim vardı. Bunları yavaş yavaş yeniyorum. Bunları yendikten sonra daha iyi olacağını düşünüyorum. Ama hala aklımda bir düşünce var bir türlü gitmeyen, ben bunu yapamam, yapmaya çalışsam bile kesin yanlıştır [düşüncesi]. Bu korkumu yenemiyorum. Yenemedikçe başaramıyorum.

Görev performansına dair öz-beklentilerin karşılanamaması öğrencilerde yetersizlik hissinin oluşmasına yol açmıştır. Bu süreçte öğrencilere ihtiyaç duydukları küçük desteklerin verilmesinin oldukça önemli olduğu görülmüştür.

R_K49: Yolunda giden bir şey yoktu. Çünkü güne mutlu ve güzel başladım. Ama sonra da soruyu çözemediğim için hayal kırıklığına uğradım.

C_K66: If…else konuları çok zor. Ben zorlanıyorum, biraz daha gayret etsem çözebilirim diyorum, ama olmuyor.

R_K60: Ya, soruyu da tam anlayamıyordum. Çünkü bir yere kadar öğreniyorum hocam, devamını getiremiyorum. Karşı taraftan bekliyorum bu şeyi [grup arkadaşlarından], yani benim eksik olduğum yerleri tamamlamalarını.

C_K71: int, if…else onları kesin öğrenmiş oldum. Hocamız soru verirken yapamıyorum, kapasitem yetmiyor, az bir ipucu yetiyor, o olsa devamını getiririm.

Görevleri başaramama; asabiyet, can sıkıntısı ve pes etme eğilimi oluşturmuştur. Ayrıca karşılaşılan hataların bir türlü giderilememesi problem çözme sürecini sıkıcı hale getirmiştir.

R_K6: Sadece ilk soruda takıldık, çok ağırıma gitti, başım döndü sinirden. Çünkü 0-10 arası sayı istenmişti. Ne yaptıysak olmadı.

R_K19: Ders güzel geçmedi, çünkü zorlandım bu hafta. Bu haftaki görevler diğerine göre zor geçti. Bir de bana eksi olarak yansıdı. Hem de benim güvenimi düşürdü, yani yapmak istemedim o görevi, çünkü sıkıldım.

C_K31: Programlama temellerinde kodlama her hafta umudum biraz daha tükeniyor.

Programlama dersleri genellikle en yüksek ders bırakma oranına sahip derslerden birisidir. Bu nedenle öğrenme sürecinde ihtiyaç duyulan bilişsel ve duyuşsal desteklerin belirlenmesi ve sağlanması son derece önemlidir. Örnek olarak, C_K31'in olumsuz düşüncelerle başa çıkmakta zorlandığı, bu düşüncelerin kendisini dersin ötesinde okulu bırakma seviyesine getirdiği görülmüştür. Uygulama sonunda yapılan görüşmelerde C_K31 zorluklar karşısında kolay pes etmesinin görevlerin başarısızlıkla sonuçlanmasında önemli bir etmen olduğunu ifade etmiştir.

C_K31: Çok çalışmalı, yapamıyorsan bence okul değiştirmeli benim gibi. Anlamadığım ve yapamadığım için okul değiştiriyorum.

C_K31: Program yazarken, birden fazla değişken olduğu için yanlış yerlerde yanlış değişkenleri kullanıyorduk. Olmuyordu. Çabuk pes ediyorduk. O yüzden başarısız olduğumuz yerler vardı.

Eğitsel robot grubundaki bazı öğrencilerin ilk haftalarda robotu bozma korkusu nedeniyle tedirginlik yaşadıkları, ancak ilerleyen haftalarda bunu üzerinden attıkları belirlenmiştir.

R_K15: İlk başta bozulacak [eğitsel robot] diye korkarak açıyordum. Şimdi rahat rahat açıyorum.

Öğrencilerde başarma inançlarının oluşması problem çözmeye yönelik çabalarının artmasını sağlamıştır. Öğretmenin duyuşsal desteğini arkalarında hissetmeleri öğrencilerin işbirlikli çalışmalarda daha özgüvenli bir şekilde çalışmalarını sürdürmesini sağlamıştır. Öğrencilerin problemleri çözdükçe yeteneklerinin farkına vardıkları, bunun da başarı inançlarını yükselttiği tespit edilmiştir.

R_K1: Kendime inanmamdı. Kendime güveniyorum. Bu düşüncelerim [inanç] ile başardım.

R_K20: Öncelik hocamın beni bu takımda görmesi bana çok güven verdi. Bu güvenle duygu ve düşüncelerimi rahatça paylaşabiliyordum arkadaşlarımla.

R_K8: İyi becerilerimiz olduğunu öğrenmemiz. Buna [problem çözme] katılmasaydım, nasıl becerilerimin olduğunu öğrenemezdim. Eğlenceli olmazdı.

C_K33: Hesap makinesi programlamak, program yapmak isteyince oluyor. Kendisine güveni olması yeter insana.

Özellikle eğitsel robot programlama etkinliklerinde ortaya çıkan fiziksel ürünler/çıktılar öğrencilerin kendilerini takdir etmelerini ve başkalarından takdir görmelerini sağlamıştır.

R_K19: Robotu ilk defa yaptık, bu da bizim kendimizi mucit zannetmemize yardımcı oldu, yani kendimize güvenerek bir iş başardık, bu da kendimizden memnun kaldığımızdan yani güvendiğimizi gösteriyor.

R_K12: Hocam bir de şey vardı, robotu ilk gördüğümüzde yapamayız diye düşünüyorduk, programlarımızı sergide sunduk, gelen arkadaşlarımız nasıl yaptınız filan vay! [şaşkınlık] dedi, çok güzel filan.

Problem çözme oturumu sonrasında yanıtların paylaşılmasının öğrencilerin başarma inançlarına olumlu etkisi olmuştur.

C_K31: Siz tahtada yapınca yapabileceğimizi görüyorduk.

Öğrenci motivasyonunu etkileyen birçok faktörün olduğu belirlenmiştir. Bunlar hoşnutluk, değer verme, cazibe (çekicilik), kullanım kolaylığı, rekabet, amaç belirleme, merak, sürdürme isteği ve uğraştır (katılım). Her iki gruptaki öğrencilerin de uygulamadan genel anlamda memnun olduğu görülmüştür. Öğrencilerin yeni bir şeyler ortaya çıkarması ve zaman içerisinde ilerleme kaydettiklerine yönelik algıları yaptıkları uğraştan hoşnutluk duymasında önemli bir rol oynamıştır.

R_K19: Derste robot tasarlamayı öğrendim hayatımda ilk defa yaptığım için heyecanlıydım. Acaba nasıl bir şey olacak diye çok heyecanlıydım, ayrıca sevinçliydim, çünkü ilk defa kendi robotumu tasarlıyordum, bunun için de kendimden çok ama çok memnundum.

R_K6: Bugün öğrendiklerim güzeldi ve eğlenceliydi. Her geçen gün daha iyi olmaya başladık, o yüzden gururluyuz.

C_K24: Kötü şöyle yaptığım. Yaptığımı eskiden anlamadığımı anladım. Şimdi yapıyorum ve anlıyorum. Artık bu hafta iyiydim ve güzel geçti.

R_K49: Kötü giden bir şey yoktu. Ama dizi ve döngüyü ilk öğrendiğim zaman kodların nasıl yazılacağında sıkıntı çekiyordum. Ama kolay soruları çöze çöze kendimi geliştirdiğimizi düşünüyorum.

Öğrencilerin öğrendiklerine değer vermesi onların derse karşı motivasyonlarını arttıran bir diğer unsurdur. İster bir programlama aracına isterse bir problem çözme stratejisine karşı olsun öğrencilerin değer algılarının, yarar algılarıyla ilişkili olduğu görülmüştür.

R_K9: Hocam dün "Terminatör" adında bir film izledim robotlar üzerine, robotlardan adam yapıyorlar. Yani çok muhteşem bir şey ya.

R_K2: [eğitsel robot programlamanın] bize çok yararı oldu, mekanizma bilgimiz çoğaldı.

C_K34: Duygu ve düşünce olarak öğrendiklerimin bana birçok yerde yararlı olacağını düşünüyorum.

Yazılan programların çalışması ve sonuç üretmesi, özellikle de robotların fiziksel dünyada işlevsel hale getirilmesi (yalnızca eğitsel robot programlama) programlama etkinliklerinin çekici hale gelmesini sağlamıştır. Özellikle deney grubunda eğitsel robotlarla çalışma ve yeni şeyler ortaya çıkarma öğrenmenin bu gruptaki öğrencilere daha çekici hale gelmesini sağlamıştır.

R_K5: Hocam hayatımızda ilk defa gördüğümüz bir şey. [robotlar] İlgimizi çekiyordu.

Öğrencilerin bir programlama ortamını ya da aracını kullanmaya yönelik isteklerini etkileyen en önemli unsurlardan birisinin kullanım kolaylığı olduğu görülmüştür. Programlama aracında kodların görsel, arayüzün basit ve anlaşılır olmasının bu aracın kullanımını kolaylaştırdığı belirtilmiştir. Ayrıca programdaki yönlendirici uyarıların (hata mesajları dâhil) da kullanım kolaylığını etkileyen önemli bir unsur olduğu belirtilmiştir.

R_K5: NXT programı daha kolay, daha anlaşılır ve daha iyi.

R_K18: [Mindstorms NXT yazılımı] Gösterim [görsel] olduğu için kolay.

C_K41: Hata veriyor onları tıkladığında anlıyorum ve bu yönünü seviyorum o yüzden bu programı çalıştırıyorum.

Diğer öğrencilerle ya da takımlarla rekabet etmenin öğrenci motivasyonunu arttırdığı görülmüştür. Hem deney hem de karşılaştırma gruplarındaki öğrencilerin rekabet duygusu yaşadığı ve geride kalmamak için çabalarını arttırdığı görülmüştür. Öğrencilerin zaman zaman zorluklar belirleyip, bu zorluklara meydan okuduğu ve onlarla mücadeleye giriştiği görülmüştür.

R_K12: Bugün programlama temelleri dersinde robot C'nin yeni kodları ve yeni sorularla karşılaştık. Grup halinde biraz yavaş çalışsak da yine de diğer gruplardan öndeydik.

C_K30: Hepsini. İyi yaptığımı düşünüyorum. Zaten sınıfta ilk yapan ben oluyorum.

C_K31: Not tutmak ve sürekli olarak hocanın verdiği örnekleri yaparak ilk defa bugün bu derste fazla hata almadım. Rakibim C_K36. Onun yapamadığı soruyu sınıfta ilk ben yaptım.

C_K65: Hocamız ödev verdi ama bir tek C_K69 yaptı. Bu beni üzdü, çünkü ben niye yapamadım diye.

R_K13: Bence Robot C olursa daha iyi olur. Zoru başarmak iyidir.

Öğrencilerin sıklıkla öğrenme amaçları belirlediği, bunun da onların performansına olumlu yansıdığı görülmüştür.

R_K14: Programlama problemleri ileriki hayatımda çok iyi olmama yardımcı olacak ve sorulara grup arkadaşlarımla bakış açımızı ilerletmemiz gerektiğini anladım.

C_K24: Bugün çok mutlu ve sevinçli bir gündü. Hepsini yaptım ve anladım. Bugüne kadar anlamadığım dersi şimdi bu hafta yaptım ve başardım, daha da yapacağıma inanıyorum.

C_K29: Hata bulma ve giderme. Biraz zor oldu, fakat geliştirmeyi düşünüyorum.

C_K64: Hep tahtaya bakarak olmaz, mantığını anlamaya çalışacağım, bundan sonra tahtaya bakmayacağım. Kendim yapacağım, deneyeceğim, not tutacağım, ona göre yapacağım.

Yapacakları işlerin öğrencilerde merak uyandırması onların işlere karşı motivasyonlarını yükseltmiştir. Özellikle deney grubundaki öğrenciler için robot tasarım ve programlama etkinlikleri onlarda yüksek derecede merak oluşturmuştur. Öğrencilerin yeni konuya geçmeleri de zaman zaman onları meraklandırmıştır.

R_K16: Yaptığımız işe olan merakım. Meraklı olmam beni bu işe daha çok sürüklüyor.

R_K16: Takım halinde olmamız. Sonunda ortaya çıkacak şeyi merak ediyoruz.

C_K63: Yeni bir konuya geçtiğimiz için heyecan ve merak duygusu sardı.

Robotların verilen kodlara göre hareket etmesi bu gruptaki öğrencilerin oldukça hoşuna giderken, programlama süreçleri üzerindeki hâkimiyet duygularını yükseltmiştir.

R_K6: Hocam robotlar bizim elimizdeydi [kontrolümüzdeydi]. Onun için severek yaptığımızı düşünüyorum.

R_K9: Kendimizi sanki bir patron gibi görüyoruz. Biz ne dersek onu yapıyor gibi oluyor.

Öğrencilerin problem çözme sürecinde çalışma kararlılıklarını sürdürmeleri oldukça önemlidir. Etkinliklerin zevkli olması ve öğrencilerin hoşuna gitmesi onların etkinlikleri sürdürme çabalarını önemli ölçüde etkilemiştir.

R_K1: İzlediğim adım sonuna kadar yapmamdı. Yapamadığımda da yine yapmaya çalıştım.

R_K4: Sorularda bayağı azimliydim. Heyecanlıydım. Bir o kadar da yoruldum ama değdi.

R_K11: Güzel şeyler vardı. Heyecanlı geçti, iyiydi. Güzel oldu. Haftaya aynısını genişletmek isterim.

R_K52: İyiydi. Emeğimizi somut bir şekilde görünce daha çok azim artıyor ve eğlenceli hal alıyor.

Öğrenciler başarısız girişimler karşısında hırslandıklarını belirtmiştir. Ancak, bu durum sıklıkla aynı işlemleri ısrarla uygulamaktan dolayı olumsuz sonuçların tekrarlanmasıyla sonuçlanmıştır. Bu noktada öğrencilerin bakış açısını değiştirmeleri gerekmektedir. Öğrenciler kararlılıkları neticesinde öğrendiklerinin daha akılda kalıcı olduğunu belirtmiştir.

C_K38: Hırs yapıyorsun ya yapamayınca.

C_K32: Hırs yapınca da daha çok kötü oluyordu.

C_K35: Bence hırs yaptıkça daha da ilginç hocam. Bir de onu giderdiğin için [hataları gidermek] hataları daha iyi kavrıyorsun.

Öğrencilerin problem çözme uygulamalarında kendi fikirlerini uygulama (kendi tasarımlarını yapma ya da programlarını yazma) olanağı bulması, etkinlikler üzerinde hâkimiyet hissi yaşamaları derse katılımlarını olumlu şekilde etkilemiştir. Deney grubundaki öğrenciler robot hareketlerini kontrol etmelerinin dersi oldukça zevkli hale getirdiğini belirtmişlerdir. Öte yandan, karşılaştırma grubundaki öğrenciler kodlama kısmı zor gelse de form pencerelerini tasarlama işinin keyifli olduğunu belirtmiştir. Her iki gruptaki öğrencilerin de problem çözme sürecinde sıklıkla akış deneyimi yaşadıkları anlaşılmıştır.

R_K9: Hocam şahsen benim hoşuma giden yanı, hani ben yürütüyorum ya robotu, hoşuma giden yanı o. Ben ne dersem onu yapıyor.

C_K31: Bu C# daha zor ama kodlama şeyi zor ama şey tarafı hoşuma gidiyor tasarım tarafı.

R_K55: Çok güzel canım sıkılmıyor. O kadar güzel ders işlediğimizden zaman su gibi akıp geçiyor.

C_K73: İyiydi, güzeldi. Eğlenerek ders yaptık. Zamanın nasıl geçtiğini anlamadık.

Bilgisayara karşı tutumun görev performansını etkilediği görülmüştür. Zaman ilerledikçe öğrencilerin önyargılarının azaldığı ve programlamaya karşı sevgilerinin arttığı anlaşılmıştır.

C_K28: Bilgisayar ile uğraşmayı sevdiğim için konuları çabuk yapıyorum.

R_K6: Hocam bu robotlar sayesinde Pazartesi günlerini çok sevmeye başladım.

R_K42: Programlama dersinde her şey gayet yolundaydı. Benim içi kötü giden bir şey yoktu. Dersi sevmeye başladım.

Öğrencilerin problemlere farklı açılardan bakmasının empatik düşünme becerilerini arttırdığı saptanmıştır.

R_K12: Bir işe başlamadan o işi nasıl çözeceğime bakmam soruyu iyi anlamaya ve o soruyu 2-3 kez okur ve soruyu anladıktan sonra çözüme başlarım. Ve soruda arkadaşlarımın fikrini ve kendi görüşlerimi katarak düzgün bir sonuç elde etmeye bakarım. Hayattaki sorunlarıma da empati kurarak cevap veririm. Daha verimli sonuçlar elde etmek için strateji uygulayarak ve mantık yürüterek gidersek daha sağlıklı olacağını düşünüyorum.

Beklentiler ve çıktılar. Dersten beklentiler başta motivasyon ve katılım olmak üzere öğrenme çabalarının nasıl yönlendirileceğini önemli ölçüde etkilemektedir. Ayrıca öğretimsel süreçler yoluyla ortaya çıkan çıktılar öğrencilerin kariyer tercihlerinde önemli rol oynamaktadır. Bu bölümde öğretimsel uygulamalara ilişkin öğrenci beklentileri ve öğrenme çıktılarına yönelik bulgulara yer verilmektedir.

Beklentiler. Bu bölüm iş ortamı ve müşteriler, mesleki kariyer, toplum ve öğrenci olmak üzere farklı kesimlerdeki beklentilerden yola çıkılarak ele alınmıştır. Öğrencilerin bilgisayımsal becerilerin iş ortamına uygulanması noktasında birbirlerinden farklı bakış açılarına sahip olduğu görülmüştür. Hâlihazırda bir sektörde çalışan R_K6, robot teknolojilerini çalıştığı iş sahasında kullanmayı düşünürken, herhangi bir sektörde çalışmayan C_K30, bu becerilerin iş ortamındaki görevlerle bağlantısının olmayacağını belirtmiştir.

R_K6: Hocam mesela şimdi benim işimin sektörü perde sistemleri, mesela hocam şimdi ben şey düşünüyorum kendim için tasarım hocam, perde tasarımı yani böyle bir program ilerde yapabilirsem benim için çok iyi olur.

C_K30: Ama herhangi bir iş ortamında böyle program sormayacaklar ki? Mesela hastanede eczanede çalışırken sana program yaz demeyecekler.

Öğrenciler bilgisayımsal becerilerin stajda öne çıkma, iş bulma ve mesleki kariyerlerini geliştirme noktasında oldukça işe yarayabileceğini belirtmiştir. Ek olarak, öğrenciler bilgisayımsal etkinliklerin BT kullanma becerilerini geliştirdiğini ve bunun gelecekte teknolojiye ayak uydurmaları açısından kolaylık sağlayacağını belirtmiştir.

C_K38: Bu Visual Studio'da iyi olduğumu düşünüyorum. Bu stajda işime çok yarar.

C_K41: Staj yerlerinde daha kolay yapmak staj yerinin bizi beğenmesi için önemli.

R_K10: Programlamayı iyi yaptım, ilerde meslek sahibi olabiliriz.

C_K35: İleride kendime ait program veya benzer şeyleri yapabilmeyi çok istiyorum.

Öğrenciler çalışma sektöründe müşteri isteklerinin farklılaşacağına dikkati çekerek, müşterilerin bireyselleştirilmiş ihtiyaçlarına cevap verilmesinde bilgisayımsal becerilerin işe koşulması gerektiğini belirtmiştir. Gelişen teknolojinin artık farklı talepleri beraberinde getirdiği (3B tasarım, baskı ve üretim) ifade edilmiştir.

R_K6: Mesela hocam müşteriye göre. Nasıl istiyorsa ona göre baskılayıp göstermemiz lazım, programlarla yapabilirim.

Deney grubundaki öğrencilerin robotları insanlara faydalı olacak, onların ihtiyaçlarına cevap verecek şekilde tasarlama ve programlama beklentisinde olduğu anlaşılmıştır.

R_K61: İnsanlara faydalı olacak, yani insanlara faydalı bir şekilde yararlanılması.

R_K62: Bir toplumdayız ya hocam, sorunlar karşısında çözüm bulma yeteneğimizi geliştiriyor.

Bilgisayımsal etkinliklerin sosyal hayatta ve bir üst akademik seviyede (lisans) işe yarayacak bir beceri seti olduğu belirtilmiştir. Bazı öğrenciler bu becerilerin proje hazırlama süreçlerinde oldukça fayda sağladığına dikkati çekerek, ileride kendilerini geliştirmelerine yardımcı olacağını belirtmiştir.

R_K17: Robot yapmak ilerde daha iyi yerlere gelmemizi sağlayacağını düşünüyorum. Üniversitede gerekeceğini düşünüyorum. Robotu çalıştırmak, onu ayarlamak, komut vermek bence gerekli diye düşünüyorum.

C_K35: Bu derse katılmamın bir sebebi ve bir amacım var, eğer bunlar olmasaydı bu derse katılmazdım. Ders hayatımızda karşımıza sık sık çıkabilecek bir konuya sahip.

R_K19: Aynen hocam. Bir de bu proje hocam, TÜBİTAK projesi hocam, yani daha da çok iyi yüksek yerlere [seviyeye] ulaşmamızı şey yaptı.

Diğer çıktılar. BD etkinliklerinin farklı derslerdeki kavramlarla ilişkilendirilmesinin başta sayısal grubu dersleri olmak üzere diğer derslerdeki birtakım konuların daha iyi anlaşılmasını sağladığı belirtilmiştir. Öğrenciler problem çözme uygulamalarında alan-çevre hesaplamaları, denklem, faktöriyel kavramı, işlem önceliği, karekök hesaplama, üslü sayılar, mod alma, eşitsizlikler, yüzde kavramı ve sayı doğrusu (negatif-pozitif sayı) gibi konuları yeniden ele aldıklarını ve öğrenme eksikliklerini giderme fırsatı bulduklarını belirtmiştir. Deney grubundaki öğrencilerin robotun LCD ekranıyla ilgili yaptıkları işlemler koordinat sistemini daha iyi kavramalarını sağlamıştır. Polya'nın problem çözme stratejisinin bilgisayımsal problemlerin ötesinde farklı alanlardaki problemlerin çözümüne uygulandığı ifade edilmiştir. Öğrencilerin BD etkinlikleri esnasında sıklıkla işe koştukları mantıksal düşünme becerilerinin ve soyutlamaların sözel grubu derslerde özellikle kavramların daha kolay anlaşılmasına ve büyük konu üzerinde çalışmasına katkı sağladığı belirtilmiştir.

R_K17: Ben bu dersin verimli ve sorunlara yardımcı olduğunu düşünüyorum. Özellikle de matematiğe faydasını daha çok buluyorum. Tabi ki matematiğin de bu derse etkisini görüyorum.

C_K27: Geliştirdiğim stratejiler[in] gerçek hayatta değil de derslerde çok faydasını görüyorum. Mesela fizik ve temel elektronik ölçmede çok işimize yarıyor, burada yaptığımız stratejilerin çoğu o derslerde önümüze çıkıyor.

C_K36: Matematik dersinde de soruları daha hızlı ve daha sağlam çözebiliyorum artık.

R_K62: Sözel derslerde de yardımcı oldu hocam. Mantıksal olarak geliştiriyor ya. Soruyu anlamamıza yardımcı oluyor.

Öğrenciler teknik sorunlarla başa çıkarken kullandıkları yöntemlerin diğer bilişim derslerinde ve gündelik hayatta karşılaştıkları bilgisayar sorunlarını çözmede de işe yaradığını belirtmiştir.

C_K26: Matematiğimi biraz daha geliştirdim ve kendi bilgisayarımdaki sorunları da burada öğrenmemle [öğrendiklerimle] çözdüm.

Komutların, programlama aracı menülerinin ve hata uyarılarının İngilizce olması öğrencilerin yabancı dille ilgili bazı kelimeleri ve kavramları öğrenmelerini sağlamıştır.

C_K28: Mesela İngilizce, bu derste hata mesajları İngilizce olduğundan hem İngilizcemiz gelişiyor hem de programlama temellerinde daha iyi bir sonuç elde ediyoruz.

Bölüm 5
Sonuç, Tartışma ve Öneriler

Bu bölümde araştırmanın bulgu ve yorumlarına dayalı olarak ulaşılan sonuçlar tartışılmıştır. Ayrıca araştırma sonuçlarından yola çıkarak programlama yoluyla desteklenecek bilgisayımsal düşünme (BD) etkinliklerine yol göstermesi için çeşitli önerilere yer verilmiştir.

Sonuç ve Tartışma

Çalışma kapsamında yürütülen durum çalışmalarında (durum-1 ve durum-2) gerçekleştirilen öğretimsel müdahale sonrası hem deney hem de karşılaştırma grubu öğrencilerinin programlama başarısında anlamlı bir artış yaşanmıştır. Uygulama sonrası öğrenenlerin programlama testi puanları yaklaşık 2 kat artarak düşük seviyeden orta seviyeye yaklaşmıştır. Ayrıca öğretimsel müdahalelerin öğrencilerin programlama başarısı üzerindeki etkisinin de yüksek olduğu belirlenmiştir. Bu durum düzenlenen programlama eğitimi ve diğer BD etkinliklerinin öğrencilerin programlama becerilerini geliştirdiğini göstermektedir. Her iki durum çalışmasında da deney grubu öğrencilerinin programlama testi kazanım puanlarının kontrol grubundan yüksek olduğu görülmüştür. Bununla birlikte gruplar arası son test puanlarındaki bu farkın iki durum çalışması için de anlamlı olmadığı görülmüştür. Bu sonucun ortaya çıkmasında test gruplarındaki öğrenme etkinliklerinin birbirine oldukça yakın koşullarda gerçekleşmesinin etkisi olduğu düşünülmektedir. Çalışmada deney ve kontrol grubu için programlama modalitesi dışında tüm öğretim koşulları mümkün olduğunca birbirine benzer olarak tasarlanmıştır. Her iki grupta da öğrencilerin BD becerilerinin desteklenmesi için programlama etkinliklerinden yararlanılmıştır. Öğrenme içeriği, temel programlama kavramlarından günlük hayatta kullanılan algoritmik problem çözme etkinliklerine kadar uzanacak şekilde düzenlenmiştir. Tüm gruplarda dersin kuramsal oturumunu takiben bilgisayımsal problem çözme etkinliklerini barından bir uygulama oturumu gerçekleştirilmiştir. Çalışmada deney grubundaki robotik uygulamalar için sökülüp birleştirilebilen robotik yapı parçalarından oluşan Lego Mindstorms NXT 2.0 kiti kullanılmıştır. Robotlara eylem ya da işlev kazandırmak için blok tabanlı yazılımlar mevcut olmasına rağmen (Mindstorms NXT Software v2.1 gibi) hem daha esnek ve işlevsel programlara olanak tanıması hem de karşılaştırma grubundaki öğrenenlerle benzer bir programlama çevresi sunması nedeniyle metin tabanlı “Robot C for Lego

Mindstorms” yazılımı kullanılmıştır. Özetle, hem karşılaştırma grubu öğrencilerinin kullandığı C# hem de deney grubu öğrencilerinin kullandığı Robot C, metin tabanlı programlama dillerinin teknik ve sözdizimi ile ilgili zorluklarını barındırmaktadır.

Alanyazında robotik setlerin ve yazılım araçlarının öğrencilerin programlama becerilerinin desteklenmesindeki olumlu etkilerini raporlayan birçok çalışma bulunmaktadır. Öte yandan, doğrudan eğitsel robot programlama ile geleneksel programlama etkinliklerinin karşılaştırıldığı çalışmalar oldukça az sayıdadır. Özüorçun ve Bicen (2017) robot teknolojilerinin mühendislik öğrencilerinin bilgisayar programlama ve algoritma mantığı üzerinde nasıl bir etkiye sahip olduğunu incelemiştir. Söz konusu çalışmada programlama etkinliklerini fiziksel robotlarla (Lego Mindstorms EV3) gerçekleştiren öğrencilerin programlama başarısında istatistiksel olarak anlamlı bir artış yaşandığı; buna karşın robot eylemlerini oyun benzeri bir program aracılığıyla sanal ortamda gerçekleştiren öğrencilerin puanlarında görülen nispeten küçük artışın ise istatistiksel olarak anlamlı olmadığı raporlanmıştır.

Garcia ve De la Rosa (2016) 6-8. sınıf kademesindeki bir grup ortaokul öğrencisinin web tabanlı bir görsel-blok programlama uygulaması aracılığıyla sanal ortamda gerçekleştirdikleri robotik etkinlikler sonunda programlama ve algoritma kavramlarına ilişkin kazanımlarının önemli ölçüde arttığını (%91 oranında) raporlamıştır. Araştırmacılar ayrıca web tabanlı sanal robotik uygulamasının etkililiğini popüler bir blok programlama aracı olan Scratch ile karşılaştırmıştır. Beş saat uzunluğundaki deneysel girişim sonunda her iki aracın da öğrenci öğrenmeleri üzerinde benzer bir etkiye sahip olduğu belirlenmiştir. Ek olarak, sanal robotik etkinlikler katılımcıların %95’i tarafından ilgi çekici bulunmuştur.

Bilişim teknolojileri öğretmenlerinin yeni teknolojilere hazırlanması bu teknolojilerin okul ortamında etkili ve verimli kullanılmasının ön gerekliliği konumundadır. Majherova ve Kralik (2017) öğretmen adaylarına yönelik olarak yürüttükleri çalışmada programlama eğitiminde farklı robot programlama modalitelerinin etkisini incelemiştir. Araştırmacılar yüksek maliyetli fiziksel robot setlerine alternatif olarak sanal robot çevrelerini kullanmıştır. Çalışmada fiziksel robotlarla çalışan öğrencilerin robot modellemeye (yapımına); sanal robot kullanan öğrencilerin ise robot programlamaya yönelik öz-yeterlik algılarının daha yüksek olduğu görülmüştür.

Litts vd. (2017) e-tekstil ve giyilebilir teknolojik ürünler oluşturmak için kullanılan Lilypad Aurdino aracılığıyla gerçekleştirdiği 3 haftalık bir proje süreci sonunda lise öğrencilerinin işlevsel bir devre tasarlama, hâlihazırda tasarımı ve kodlaması yapılmış bir devreyi yorumlama ve yalnızca kodları yazılmış bir devreyi tasarlama, yorumlama ve yeniden düzenleme yeteneklerinin önemli ölçüde geliştiğini raporlamıştır.

Scott vd. (2015) programlamaya giriş derslerinde web programlama ile robotik uygulamaların programlama süreci üzerindeki etkisi üzerine gerçekleştirdikleri çalışmada robotik etkinliklerle meşgul olan lisans öğrencilerinin programlama için ayırdıkları sürenin daha yüksek olduğunu, işlevsellik ve gelişmişlik açısından daha yüksek nitelikte programlar ürettiklerini raporlamıştır. Ayrıca kod pratikleriyle meşgul olma süresinin programın işlevselliği üzerinde anlamlı bir etkiye sahip olduğu belirtilmiştir. Cinsiyet özellikleri, hazırlanan programların niteliğinde anlamlı bir farklılık yaratmamıştır. Söz konusu çalışmada deney ve karşılaştırma grubu için düzenlenen öğretimsel etkinlikler eşzamanlı olmayıp, farklı dersler kapsamında ardışık olarak (1 dönem arayla) gerçekleştirilmiştir. Ek olarak, deneysel uygulamaların yerleşik olduğu derslerde izlenen öğrenci başarısını değerlendirme teknikleri (ara sınavlar, haftalık görevler) de test gruplarına göre farklılaşmıştır. Özetle, çalışmada değişkenlere ilişkin yapılan manipülasyonlar dışında test grupları için düzenlenen koşulların da farklılaştığı gözükmektedir. Bu tarz öğretimsel müdahaleler çalışmaların iç geçerliği için tehdit oluşturmaktadır. Mevcut çalışmadaki bulgular ve alanyazındaki diğer çalışmalar eğitsel robotların programlama becerilerinin desteklenmesinde en az geleneksel programlama araçları kadar etkili olduğunu göstermektedir. Bununla birlikte bu alanda daha fazla karşılaştırmalı çalışmaya ihtiyaç bulunmaktadır.

Öğrencilerin formal soyutlama test sonuçları incelendiğinde durum çalışması-1'de BD etkinlikleri sonrası hem deney hem de karşılaştırma gruplarının formal soyutlama becerilerinde anlamlı bir gelişme görülmezken; durum-2'de sadece eğitsel robot grubundaki öğrencilerin formal soyutlama puanlarında anlamlı bir artış yaşanmıştır. Her iki durum çalışmasında da deney ve karşılaştırma grupları arasında formal soyutlama son test puanları açısından anlamlı bir farklılık bulunmamıştır. Öte yandan, her iki durum çalışmasında da BD etkinlikleri sonrasında deney grubu öğrencilerinin betimsel soyutlama beceri puanlarında

anlamlı bir artış gözlenirken; karşılaştırma grubu öğrencilerinin puanlarında anlamlı bir farklılık olmamıştır. Durum çalışması-1'de deney ve karşılaştırma gruplarının betimsel soyutlama son test puanları arasında anlamlı bir farklılık gözlenmezken; durum çalışması-2'de betimsel soyutlama puanlarının deney grubu lehine anlamlı şekilde yüksek olduğu görülmüştür. Bu sonuçlar eğitsel robot programlama pratiklerinin geleneksel nesne tabanlı programlama etkinlikleriyle karşılaştırıldığında öğrencilerin soyutlama becerilerinin gelişimini daha fazla desteklediğini göstermektedir.

Bilgisayar bilimlerinde başarı için gerekli anahtar faktör soyutlamalarla çalışma kabiliyetidir (Hazzan, 2002; Hazzan & Kramer, 2007). Bilgisayar bilimcileri farklı çeşitliliklerin yazılım hiyerarşilerini oluştururken gerekli kavramsallaştırmalara soyutlama merdivenini kullanarak ulaşmaktadır. Soyutlama becerisi sergilemek ya da bir soyutlama yaratmak, daha az önemli özellikleri geri plana atıp, fikrin önemli yönlerini ön plana çıkararak bir durumu ya da süreci daha genel hatlarıyla kavramsallaştırma ve temsil etme yeteneği gerektirir (Weintrop vd., 2016). Soyutlamalar karmaşıkla başa çıkmak için sıklıkla işe koşulan bilişsel bir süreçtir. Ayrıca, soyutlama farklı disiplinlerde farklı şekilde tanımlanan oldukça genel bir beceridir. Mevcut çalışmada soyutlama karmaşıklık karşısında basitliği ayırt etme ve oluşturma becerisi olarak ele alınmıştır. Alanyazında soyutlamanın geliştirilebilir bir beceri olduğu konusunda bir fikir birliği bulunmaktadır. Çalışmada soyutlama becerilerinin sergilenmesi ve desteklenmesi için başta problem çözme ve programlama olmak üzere çok çeşitli etkinlikler düzenlenmiştir. Ayrıca, problem çözme formları aracılığıyla öğrencilerin problemin amacını ya da kapsamını belirlemesi, problemi daha fazla irdelemek için modeller yaratması, çözüm fikirlerini ya da bulgularını işlevsel hale getirmek için onları belirli bir formda ifade etmesi ve program kodlarına dönüştürmesi teşvik edilmiştir. Öğrencilerin kod yazmanın prosedürel adımları öncesinde çözümlerini algoritma ve akış şemaları şeklinde ifade etmesinin soyutlama pratiklerine olumlu katkısının olduğu düşünülmektedir. Çünkü algoritma ve akış şemaları zihinsel süreçlerin dışa vurulmasını ve bu yolla onlar üzerinde akıl yürütülmesini sağlarken, çözüm önerilerinin de modellenmesine olanak sağlamaktadır. Bu tür modeller zihinsel temsillerin birbiriyle ilişkili olarak dışa vurulmasını sağlayarak, mevcut bilgilerin ve anlayışların işlevsel hale getirilmesinde iskele vazifesi görmektedir. Yapılan çalışmalar algoritma ve akış şeması gibi kâğıt-

kalemle gerçekleştirilen BD etkinliklerinin programlama araçları kadar etkili sonuçlar verebildiğine işaret etmektedir (Kim, Kim & Kim, 2013). Deney ve karşılaştırma grubundaki öğrenciler bilgisayar bilimleri alanında aynı konuları çalıştıkları halde, BD'nin temel bileşenleri olan soyutlama ve ayrıştırma konusunda farklı deneyimler yaşamıştır. Deney grubu müfredat boyunca her bir etkinlikte tanıtılan BD kavramlarına ek olarak somut robotikler aracılığıyla fiziksel dünyadaki karmaşık işlem ve süreçleri irdeleme, robot ve fiziksel dünya arasındaki ilişkileri düzenleme, robot hareketlerini gözlemleme ve ayrıştırma fırsatı yakalamıştır. Sonuçlar deney grubu öğrencilerinin formal soyutlamayla karşılaştırıldığında betimsel soyutlama performansında daha fazla gelişim gösterdiğini ortaya koymuştur. Bunun bir nedeninin dersin uygulama bölümünde deney grubu öğrencilerine sunulan problem ifadelerinin (EK-A) daha uzun ve karmaşık olarak verilmesine bağlı olarak daha üst düzey betimsel soyutlamalar gerektirmesinin olduğu düşünülmektedir.

Çalışmada BD etkinlikleri öğrencilerin problem çözme becerilerinde anlamlı bir farklılığa yol açmamıştır. Ayrıca durum çalışmalarında problem çözme becerisi açısından test grupları arasında herhangi bir farklılık bulunmamıştır. Alanyazında BD etkinlikleri ve problem çözme yaklaşımları üzerine benzer bulguları ortaya koyan çalışmalar bulunmaktadır. Kalelioğlu ve Gülbahar (2014) Scratch ile programlama etkinliklerinin ilköğretim öğrencilerinin problem çözme becerilerinde önemli bir farklılık yaratmadığını bulmuştur. Kalelioğlu (2015) ilkokul 4.sınıf öğrencileri ile gerçekleştirdiği bir diğer çalışmasında ise çevrim-içi kodlama etkinliklerinin (code.org) öğrencilerin problem çözmeye yönelik yansıtıcı düşünme becerisi üzerinde anlamlı bir farklılık yaratmadığını saptamıştır.

Mevcut tez çalışmasında gerçekleştirilen problem çözme ve BD etkinliklerinin öğrencilerin otantik yaşam problemlerine yaklaşımları üzerinde yeterince etkili olmadığı görülmüştür. Öğrencilere sadece programlama eğitimi ya da programlama ortamı sunmak, etkili öğrenmenin gerçekleşmesi ve öğrenilenlerin farklı alanlara transferi için yeterli olmamaktadır. Bu nedenle BD becerilerine yönelik olarak düzenlenen öğrenme etkinliklerinde kullanılacak problem durumlarının gündelik hayatla ilişkili olacak şekilde seçilmesinin bu becerilerin otantik yaşam problemlerine aktarımını kolaylaştırabileceği düşünülmektedir. Ek olarak, BD becerilerinin yanında problem çözme sürecinde üst bilişsel düşünmeyi teşvik eden pratikler de

öğrencilerin problem çözme yaklaşımlarını geliştirmeleri için önemli fırsatlar sunabilir.

Her iki durum çalışmasında da öğrenen motivasyonu eğitsel robot programlama grubu lehine anlamlı şekilde farklılaşmıştır. ARCS modelini temele alan motivasyon ölçeği alt faktör özelliklerine göre incelendiğinde eğitsel robot programlama etkinliklerinin özellikle dikkat ve uygunluk bakımından öğrenci motivasyonunda pozitif ayrışma yarattığı belirlenmiştir. Çalışmada motivasyon puanları üzerinde eğitim teknolojilerinin yenilik etkisinden kaynaklanabilecek farklılıkları ortadan kaldırmak için veriler 12 ve 8 hafta süren deneysel uygulamaların ardından toplanmıştır. Bu nedenle motivasyon puanlarındaki farklılığın araç kaynaklı olmadığı düşünülmektedir.

Bilgisayımsal ürünlerin inşasını konu edinen birçok çalışmada konu alanı öğrenmelerinden ziyade öğrenci katılımı ve motivasyonu ön plana çıkmaktadır (Wilkerson-Jerde, 2014). Saez-Lopez ve Sevillano-Garcia (2017) sanat eğitiminde BD entegrasyonu üzerine gerçekleştirdiği çalışmada robotik teknolojiler ve programlama etkinliklerinin bilgisayımsal yetkinliklerin yanında öğrenci ilgisini, motivasyonunu ve katılımını da arttırdığını belirlemiştir. Araştırmacıların ders öğretmenleriyle gerçekleştirdikleri bireysel ve odak grubu görüşmeleri, robotik programlama destekli öğretimsel uygulamaların motivasyon ve eğlenebilirlik konusundaki avantajları üzerine fikir birliğini göz önüne sermiştir. Öte yandan teknoloji destekli müzik öğretimi, kontrol koşullarına kıyasla etkili öğrenme, temel kavramları belirleme, aktif katılım ve görevlerin öğrenmeye katkısı konusundaki öğrenci anlayışlarında anlamlı bir farklılık yaratmamıştır.

BD aynı zamanda diğer düşünme türleriyle de örtüşen çok sayıda bilişsel işlem setini kapsamaktadır. Mevcut araştırmada bilgisayımsal problem çözme sürecinde algoritmik, matematiksel, mantıksal, olasılıklı ve öz-düzenleyici düşünmenin ön plana çıktığı belirlenmiştir. Ayrıca programlama etkinliklerinin ayrıntılara dikkat etme, karmaşık durumlarla başa çıkma ve büyük konu üzerinde çalışma yoluyla soyutlama becerilerinin gelişimini desteklediği görülmüştür. Alanyazında algoritmik düşünmenin soyutlama ile birlikte en sık adreslenen BD bileşeni olduğu görülmektedir (Kalelioğlu vd., 2016). Shute, Sun ve Asbell-Clarke (2017) BD'nin matematiksel, mühendisliksel, tasarımsal ve sistemsel düşünmeyle benzer yönlerinin bulunduğunu raporlamıştır. Wing (2006) BD'nin matematiksel ve

mühendisliksel düşünme üzerine kurulu olduğunu belirtmiştir. Alanyazında programlama etkinliklerinin öğrencilerin algoritmik düşünme, mantıksal düşünme ve üstbilişsel düşünme olmak üzere üst düzey düşünme becerileri üzerindeki olumlu çıktılarını doğrulayan çok sayıda çalışma bulunmaktadır (García-Peñalvo & Mendes, 2018; Grover & Pea, 2013). Bununla birlikte BD araç ya da etkinlik türlerinin BD bileşenleriyle uğraşı sağlama noktasındaki etkinliklerinin farklılık gösterebileceği unutulmamalıdır (Grover & Pea, 2013).

Öğrencileri belirli bir konu alanıyla alakalı araştırmaya teşvik etmenin önemli bir yolu onlara kendilerine has ürünler inşa etme ve bu yolla daha fazla soruşturma yapma fırsatları sunmaktır. Yapım süreci öğrenenlerin geliştirdikleri ürünler ile konu arasında yeni ve farklı bağlantılar kurmasına, böylece öğrenme alanına ilişkin daha derinlemesine bir anlayış geliştirmelerine ilişkin önemli potansiyeller barındırmaktadır. Bilgisayımsal ürünlerin inşası çoklu fikirleri birleştirici bir süreçle bir araya getirme, anlayışlarını yeni yollarla organize etme ve komutların beklenenden farklı bir şey üretmesi durumunda mevcut anlayışlardaki hatalar üzerinde düşünme için öğrencileri teşvik etmektedir (Wilkerson-Jerde, 2014). Bu aynı zamanda öğrenmenin altındaki yapıları keşfetmelerine ve neden-sonuç ilişkileri kurmalarına yardımcı olmaktadır.

Fiziksel dünya yeni başlayan programcılar için daha sezgisel ve anlamlıdır (Papert, 1980). Bir programın sonuçlarının görselleştirilmesinin en büyük yararı, öğrenenlerin programladıkları nesnelerinin çalışmasını takip etmesi, tüm program elemanları arasındaki ilişkileri eşzamanlı olarak gözlemlemesi, program sonuçlarını bir bütün olarak ele alması ve böylece programlama mantığını daha iyi geliştirme fırsatı yakalamasıdır. Bu aynı zamanda programı değiştirmesi ya da yeniden inşa etmesi sürecinde önemli anlayışlar yakalamasını sağlamaktadır (Berland & Wilensky, 2015). Eğitsel robotlar ekran dışında mevcut olduğundan öğrenciler diğer öğrencilerle işbirliği yapma ve ortaya çıkan ürünler üzerinde tartışma fırsatı bulmuştur. Öğrenenler başlangıçta yazdıkları programları gözlem ve keşiflere dayalı bir şekilde iteratif adımlarla geliştirmiştir. Ayrıca somut robotikler sayesinde diğer takımların çalışmalarından haberdar olmuş, bu durum kendi gelişimlerini değerlendirmek için normlar oluşturmalarını sağlamıştır. Deney grubundaki öğrencilere Lego Mindstorms NXT 2.0 eğitsel robot kiti ile birlikte Lego yapı bloklarını tanıtan ve robot yapımıyla ilgili temel düzeydeki psikomotor becerileri

destekleyen bir bilgilendirme kitapçığı sağlanmıştır. İlk hafta öğrenenlerin bu kitapçıktan yararlanarak Lego parçalarını birleştirme ve eğitsel bir robot inşa etmesi istenmiştir. Uygulama süresinin kısaltılmış olması nedeniyle durum çalışması-2'de eğitsel robotlar temel olarak inşa edilmiş bir formda öğrencilere sunulmuştur. Öğrenciler ilerleyen haftalarda robotların donanımsal tasarımını görevlere uygun olarak gerçekleştirmiştir. Eğitsel robotlar çalışmanın başlangıç aşamasında tasarım ve montaj amaçlı kullanılmıştır. Daha sonra öğrencilerden programlama becerilerini ilerletmek için robot hareketlerini işlevsel olarak manipüle etmeleri istenmiştir. Aslında fiziksel dünyayla iletişim kurmak için bir araç olan robotlar bilgi edinimi ve diğer bilişsel becerilerin gelişimi için alternatif bir öğrenme çevresi olarak düşünülmüştür. Lego robotik setleri oldukça manipülatif bir öğrenme aracıdır. İnsanlar çevreleriyle etkileşime girdiğinde onu aktif bir şekilde manipüle eder. Bu aslında öğrenme için de önemli bir temel oluşturur. Birbiriyle uyumlu küçük yapı bloklarından oluşan Lego eğitsel robot seti öğrenenlerin etkinlik ve yönlendirmesine oldukça açıktır. Lego robotikleri sayesinde deney grubundaki öğrenciler yaptıkları manipülasyonların sonuçlarını gözlemleme ve değişimlerin anlamlılığına ilişkin çıkarımda bulunma fırsatı yakalamıştır. Öğrenenler yüksek düzeyde soyutlama becerisi gerektiren programlama pratikleriyle meşgul olurken programlara ilişkin fiziksel dönütler elde etmiştir. Bu soyut-somut geçişlerin öğrencilerin bilişsel yüklerinin indirgenmesine önemli katkılar sağladığı düşünülmektedir.

Önemine rağmen insanların programlamayı nasıl öğrendiğine ilişkin nispeten çok az şey bilinmektedir. Berland vd. (2013) basit bir görsel programlama dili olan IPRO mobil uygulaması aracılığıyla öğrenenlerin ortak bir çevrimiçi alan içerisinde sanal futbol oyuncularını (botlar) blok temelli olarak programlamasını sağlayıp kolay erişilebilir ve ilgi çekici bir yolla öğrenenlerin BD süreçlerini deneyimlemelerini sağlamaya çalışmıştır. Tamamı kadın olan 53 lise öğrencisiyle gerçekleştirilen uygulamada öğrencilerin programlama etkinliklerini keşfetme (Explore), yapıp bozma (Tinker) ve düzeltme (Refine) aşamalarından oluşan bir çerçeve etrafında (EXTIRE) karakterize eden araştırmacılar özellikle yapıp-bozmanın programlama sürecindeki önemine vurgu yapmıştır. Program durumunun değişimini takiben hesaplanan analitikler (ilkel hareket komutları, mantık ifadeleri ve sensör komutları, farklı komut sayısı, toplam komut uzunluğu, program niteliği (kazanma yüzdesi) vb.) yoluyla program niteliklerinin analiz edilmesiyle ortaya çıkan kümeler (aktif, dengeli,

kompakt, mantıksal, minimal, sınama amaçlı) sayesinde öğrenenlerin gelişimi izlenmiştir. Keşfetme sürecinde niteliği daha düşük programlar üretilirken, yapıp bozma ve düzeltme aşamalarında program niteliğinin geliştiği görülmüştür. Bu noktadan hareketle yapıp bozma sürecinin program niteliğini geliştirmenin bir yolu olduğu ileri sürülmüştür.

Acemi programcıların programlama süreçleri alanyazında uzun zamandır çalışılmaktadır. Bu çalışmaların büyük bölümü öğrenenlerin birbirinden bağımsız masaüstü bilgisayarda standart bir programlama dili aracılığıyla gerçekleştirdikleri programlama etkinliklerini konu edinmektedir (Berland vd., 2013). Bu çalışmalardan elde edilen bulguların ortak özelliği acemilerin bilgisayarların nasıl çalıştığına dair yetersiz bir zihinsel modele sahip olduğudur (Robins vd., 2003; Rogalski & Samurçay, 1990). Daha uzman programcıların aksine problem çözme sürecinde aşağıdan yukarı stratejileri kullanan acemilerin programlama süreci genellikle verimsiz kodlamalarla sonuçlanmaktadır. Bu programcılar bir problemi çözmek için atılması gereken adımları açıklayabildiklerinde bile sahip oldukları bilgi ve stratejileri uygun durumlara uygulamakta zorlanmaktadır. Ayrıca çok sayıda acemi programcı sözdizimsel zorluklar nedeniyle programlamaya karşı olumsuz tutum geliştirmektedir. Son yıllarda acemi programcıların karşılaştıkları sözdizimsel zorlukları azaltmak için program yazımını daha sezgisel hale getiren blok temelli programlama araçları tasarlanmıştır. Bu araçlar özellikle basit sözdizimsel nedenlerden ötürü çalışmayacak olan programlarının yeni başlayanları yıldırma riskini minimize etmektedir.

Mevcut çalışmada öğrencilere etkili ve verimli stratejiler geliştirmelerini kolaylaştırmak adına Polya'nın rehber sorularıyla çerçevelendirilen bir problem çözme formu sağlanmıştır. Problem çözme formlarıyla gerçekleştirilen kâğıt-kalem temelli etkinliklerde programlamanın semantik yönünün ön plana çıkarılması amaçlanmıştır. Alanyazında kâğıt kalem temelli programlama etkinliklerinin de programlama araçları kadar etkili sonuçlar verdiği görülmektedir. Kim vd. (2013) bilgisayar bilimleriyle ilgili olmayan katılımcılarla yürüttükleri deneysel nitelikli çalışmalarında kalem-kâğıtla programlama stratejisinin öğrenenlerin mantıksal düşünme becerilerini Logo programlama çevresi kadar geliştirdiğini, üstelik mantıksal düşünmenin bir kaç alt ölçeğinde daha yüksek puanların elde edilmesini sağladığını raporlamıştır. Ayrıca, Logo programlama ile karşılaştırıldığında kalem-

kâğıtla programlama öğrenenlerin BD kavramlarını anlamasına ve bilgisayar bilimlerini öğrenmeye karşı ilgilerinin artmasına anlamlı şekilde daha fazla katkı sağlamıştır. Araştırmacılar mantıksal düşünme becerilerinin gelişiminde kalem-kâğıt ve Logo programlama grupları arasında anlamlı bir farklılık olmadığını bulmuştur.

Mevcut çalışmada öğrencilerin problem çözme süreçlerinde kullandıkları yaklaşımların farklılaştığı görülmüştür. Karşılaştırma grubundaki öğrenciler daha ziyade aşağıdan-yukarı stratejiler kullanırken, deney grubundaki öğrenciler yukarıdan aşağıya stratejileri kullanmıştır. Her iki gruptaki öğrenciler de programlamanın sözdizimsel kuralları karşısında zorluklar yaşamıştır. Öğrencilerin bilgisayar programlamaya giriş derslerindeki deneyimleri, programlamaya karşı ilgilerini ve bilgisayar bilimlerine ilişkin kariyer tercihlerini etkilemektedir. Bu nedenle, programlamayı öğrenmeye yönelik etkinlikler tasarlarken, öğrenme katılımlarını arttırmak ve bilişsel yüklerini azaltmak için öğrencilere öğrenme desteği sağlamak önemlidir (Yang, Hwang, Yang & Hwang, 2015).

Araştırmacılar, bilgisayar programları geliştirmek gibi karmaşık görevlerle öğrencileri meşgul ederken hem bilişsel hem de duyuşsal hususların dikkate alınması gerektiğini belirtmişlerdir (Jonassen, 2000). Mevcut çalışmada öğrencilerin sıklıkla olumsuz duygularla (başarısızlık hissi, öğrenme kaygısı, pes etme eğilimi vb.) başa çıkmak zorunda kaldıkları ve duyuşsal desteğe ihtiyaç duydukları görülmüştür. Hatta bir öğrenci ders bırakmanın ötesinde okul değiştirmek istemiştir. Programlama dersleri yüksek ders bırakma oranına sahip derslerden birisidir. Bu nedenle öğrenme sürecinde ihtiyaç duyulan bilişsel ve duyuşsal desteklerin belirlenmesi ve sağlanması son derece önemlidir. Başta ders günlükleri olmak üzere problem çözme formlarının ve sesli düşünme etkinliklerinin öğrencilerin destek ihtiyaçlarına ilişkin önemli bilgiler sağladığı görülmüştür. Birçok araştırmada BD için gerçekleştirilen uygulamaların etkisi anketler, öğrencilerin programlama becerilerindeki kaydettiği ilerlemeler veya programlamaya veya çeşitli düşünme becerilerine ilişkin testler kullanılarak değerlendirilmektedir. Bu araçların yanında yansıtıcı düşünme etkinliklerinin ve bilişsel görev analizlerinin işe koşulması BD sürecine ilişkin farklı perspektiflerin yakalanmasına katkı sağlayabilir. Mevcut çalışmada kullanılan ders günlüklerinin öğrencilerin program yazma aşamasında yaşadıkları sorunları ana hatlarıyla adreslediği, bununla birlikte sesli düşünme

sürecinin, problem çözme formlarının ve odak grup görüşmelerinin ise bunların daha detaylı ve daha derinlemesine irdelenmesine olanak tanıdığı görülmüştür.

Çalışmada öğretmen desteğinin yanında akran desteğinin de oldukça önemli olduğu görülmüştür. Çalışma boyunca her iki test gurubunda da akran çalışmalarına izin verilmiştir. Yapılan çalışmalar işbirliğine dayalı problem temelli programlama öğretimi stratejisinin öğrencilerin bilişsel yüklerini düşürürken, aynı zamanda programlama becerilerini ve grup öğrenme öz yeterliklerini de önemli ölçüde arttırdığını göstermektedir (Wang & Hwang, 2017). Çalışmada akran işbirliğine olanak sağlanmasının amacı öğrencileri birbirlerinden öğrenebilecekleri ve akranlarından yorum veya yardım alabilecekleri bir sürece dâhil etmektir. Böylece test gruplarındaki öğrenciler sadece kendi problemleriyle başa çıkmalarının yanında akranlarının çalışmalarını gözden geçirme, akranlarının yaşadığı problemlerle uğraşma ve kendi problem çözme süreçleriyle ilgili olarak akranlarından dönüt alma fırsatı bulmuştur. Çalışmada programlama etkinliklerini sadece ekran üzerinden gerçekleştiren karşılaştırma grubunda öğrenci işbirliğinin sınırlandığı görülmüştür. Öte yandan eğitsel robotlar, doğası gereği işbirliği örüntülerine daha açıktır. Bu durum deney grubundaki öğrencilerin öğrenme toplulukları oluşturmasına, zorlu görevler sırasında birbirlerini karşılıklı olarak desteklemelerine ve özellikle robotların çalışması esnasında sosyal öğrenme fırsatları yakalamasına katkı sağlamıştır.

Çalışmada kalem-kâğıt temelli ölçme araçlarının BD becerisinin değerlendirilmesi için tek başına yeterli olmadığı görülmüştür. Öğrenciler bildiklerini kâğıda dökmekte zorlandıklarını sıklıkla belirtmiştir. BD'nin öğrenme sürecinde tamamlamaları gereken eksiklikler konusunda öğrenenlere hem dönüt vermeyi hem de onların hatalarından öğrenmelerini teşvik eden doğası aslında süreç odaklı değerlendirme yaklaşımlarıyla örtüşmektedir. Bu nedenle öğrenme çıktılarının değerlendirilmesinde öğrenme sürecini temele alan proje ve performans görevleri, görev analizleri, gözlem formları, öğrenci günlükleri vb. gibi süreç odaklı ölçmelerin daha uygun olduğu düşünülmektedir.

Çalışmanın mesleki ve teknik eğitim veren bir lisede gerçekleştirilmesi nedeniyle çalışmaya katılan kadın katılımcıların sayısı erkeklere göre daha az sayıda olmuştur. Bu durum öğrencilerin BD süreçlerinin cinsiyete göre incelenmesinde bir sınırlılık yaratmıştır. Aslında alanyazındaki birçok çalışma BD becerilerinin kazanımının cinsiyet açısından anlamlı bir farklılığa yol açmadığını

göstermektedir (Scott vd., 2015; Zhong vd., 2016). Bununla birlikte kadınların teknoloji ve mühendislik alanlarında nispeten az temsil edildiği sıkça vurgulanmaktadır. Kültürel kalıp yargılar gibi sosyal etkiler ve erkeklere göre erken yaşlardaki teknoloji deneyimlerinin daha az olması, kadınların bilgisayar bilimleri ve mühendislik alanlarına ilgilerinin ve bu alanlardaki öz yeterlik algılarının daha düşük olmasına yol açabilmektedir. Bu durum kadınların ilgili alanlardaki kariyer tercihlerini de olumsuz etkilemektedir. Master, Cheryan, Moscatelli ve Meltzoff (2017) erken yaştaki kız öğrencilerin (6 yaş) matematik ve fen derslerine kıyasla teknoloji ve mühendislik alanına karşı önyargılarının aynı yaştaki erkek öğrencilere kıyasla daha fazla olduğunu; bununla birlikte robot programlama ile ilgili deneyimlerin söz konusu alanlara karşı motivasyon ve öz yeterlik algılarını önemli ölçüde yükselterek cinsiyete bağlı farklılıkları giderdiğini raporlamıştır.

Programlama ve BD için kullanılan araç ve ortamların sunduğu olanaklar aynı zamanda öğrencilerin farklı alanlardaki kapasitelerinin geliştirilmesine de katkı sağlamaktadır. Daily ve Eugene (2013) Scratch blok programlama ortamından yararlanarak farklı gruplarla çalışmak için kritik öneme sahip duygusal öz-farkındalık ve empati özelliklerini STEM alanı için gerekli becerilerden sayılan BD becerileriyle geliştirmeye çalışmıştır. Scratch programlama ortamını kullanarak gerçekleştirilen 6 haftalık dijital öyküleme uygulaması sonunda öğrenenlerin açıkça tanımlayamasalar bile proje ürünlerinde şartlı ifadeler, sıralı mantık, döngü ve programlama nesneleriyle etkileşim gibi bilgisayımsal kavramları başarılı bir şekilde kullandıkları görülmüştür.

BD'ye ilişkin anlayışımızı geliştirmek için kayda değer çabalar sarf edilmesine rağmen, özellikle bu düşüncenin okullara entegre edilmesi konusunda halen belirsizlikler bulunmaktadır. Bu belirsizlikler arasında müfredat tasarımı, öğrenci başarısının değerlendirilmesi, öğretmen eğitimleri ve erişim engellerinin aşılması yer almaktadır (Grover & Pea, 2013). Bu alanlarda gelişme sağlamak için BD'nin iyi belirlenmiş ve ölçülebilir kavram ve/veya pratik kümesine ayrılması gerekmektedir. Bununla birlikte BD'nin eğitim çevrelerine entegrasyonu için çerçeveler ve kılavuzlar oluşturmaya yönelik çok sınırlı sayıda araştırma bulunmaktadır. Bu durum son yıllarda giderek artan öğretimsel uygulamaların pedagojik ve pratik çıktılarının değerlendirilmesinde de zorluklar yaratmaktadır. Zira iyi bir öğretim modeli değerlendirmeden ayrı düşünülemez. Uygun bir değerlendirilmeye tabi

tutulmaksızın BD becerisine yönelik herhangi bir öğretimsel pratiğin başarılı olduğuna dair beklentiler varsayımdan öteye geçmeyecektir.

Bu çalışma eğitsel robot ve nesne yönelimli programlama etkinlikleri bağlamında öğrencilerin BD becerilerinin gelişimini incelemiştir. Çalışmada karma araştırma yöntemi benimsenmiştir. Karma araştırma desenlerinde araştırmacı nicel ve nitel verilerin uyuşmamasıyla karşı karşıya kalabilmektedir. Her ne kadar çelişkili durumlar konuyla ilgili yeni görüşler geliştirilmesine katkı sağlasa da bu farklılıkların anlaşılması ya da bir nedene bağlanması zor olabilmektedir. Olası çelişkili durumların bir kez daha test edilmesi, sonuçların tutarlığının teyit edilmesini ya da doğrulanmasını, böylelikle daha eksiksiz ve net bir tablonun ortaya konulmasını sağlayabilir. Alanyazında BD'ye yönelik araştırmaların büyük bölümü tek bir kesitsel çalışmaya dayalı olarak gerçekleştirilmiştir. Bu çalışmada ise birbirinden ayrık iki deneysel nitelikli çalışma yürütülmüştür. Zamansal anlamda iki durum çalışması arasında yaklaşık 1 yıl bulunmaktadır. Farklı durumlardan elde edilen veriler araştırma bulgularının geçerliği ve tutarlılığı yönünden önemli çıkarımlar sağlamıştır.

Öneriler

Bu bölümde gelecek araştırmalara ve öğretimsel pratiklere yol göstermesi için araştırma bulgularına dayalı olarak sunulan öneriler yer almaktadır.

Gelecek araştırmalara yönelik öneriler. Soyutlama becerisinin yapı olarak bir bütün halinde geliştirilmesi yerine soyutlama türlerine göre gerçekleştirilecek alana-özgü öğretimsel müdahalelere ve problem çözme süreçlerine odaklanılması önerilmektedir. Farklı soyutlama türleri için farklı öğretim tasarımı süreçlerinin işe koşulması daha etkili sonuçlar verebilir.

Alanyazındaki çalışmaların büyük çoğunluğu bilişim derslerine yerleşik olup, bilgisayımsal becerilerin bütünleştirileceği alanlara yetersiz vurgu yapmaktadır. Bu çalışmalarda BD kavramları ve pratikleri üzerinde durulurken bu becerilerin farklı alanlarda nasıl işe koşulabileceği pratiği yeterince adreslenmemektedir. Bilgisayımsal okuryazarlığa sahip öğrenenleri yetiştirirken sadece BD ve programlama kavramlarına odaklanmak herkes için BD ilkesine aykırı düşmektedir. Alanyazında BD'nin farklı disiplin alanlarındaki pratiklerle birleştirilmesine ihtiyaç bulunmaktadır.

Mevcut çalışma için seçilen katılımcılar düşük akademik performans gösteren popülasyonu temsil etmektedir. Farklı öğretim kademesinde öğrenim gören ve/veya nispeten daha yüksek akademik başarı düzeyine sahip katılımcı grubuyla yapılacak çalışmalar BD ve eğitsel robot programlama konusundaki anlayışlara katkı sağlayabilir. Her ne kadar bazı çalışmalar bilgisayımsal çabaların özellikle lise seviyesindeki K12 sınıflarına yerleştirilmesine odaklandığını belirtse de (Mouza vd., 2017) gerçekleştirilen kesitsel alanyazın analizi durumun böyle olmadığını ortaya koymaktadır. BD ile ilgili çalışmaların daha ziyade lisans, ilkokul ve ortaokul bağlamına yerleşik olduğu belirlenmiştir. Bu nedenle lise kademesindeki öğrencilere yönelik çalışmalara halen ihtiyaç bulunmaktadır.

BD öğretiminin sadece yüz yüze eğitim ortamlarında değil, aynı zamanda uzaktan eğitim bağlamında incelenmesi araştırmacılara farklı perspektifler kazandırabilir. Ek olarak, öğrenmenin informal boyutunun giderek önem kazandığı günümüz toplumunda BD'yi sadece formal öğrenme bağlamlarına sınırlamak makul bir yaklaşım olarak gözükmemektedir. BD becerisinin gelişimi için tasarlanan çok sayıda web ve mobil uygulama araçları bulunmaktadır. BD'nin informal öğrenme ortamlarına uygulandığı araştırmalar alanyazına önemli katkılar sağlayabilir.

Mevcut araştırma farklı öğrenci gruplarıyla gerçekleştirilen deneysel nitelikteki iki kesitsel incelemeye dayalıdır. Durum çalışmalarının sayısının arttırılması ya da aynı katılımcılardan farklı zaman serilerinde elde edilecek boylamsal veriler, etkisi incelenen değişkenlere ilişkin farklı perspektifler ortaya koyabilir.

Öğretimsel pratiklere yönelik öneriler. Öğrencilerin sadece test sonuçlarına bakarak bilgisayımsal yeterlikleri kazanma durumlarına karar vermek tek başına yeterli değildir. Kalem-kâğıt temelli başarı testlerinin yanında öğrencilerin programlama etkinliklerindeki görev performansları ya da bu etkinlikler sonunda ortaya çıkan BD ürünleri de değerlendirilebilir. Ek olarak, yansıtıcı düşünme süreçlerini teşvik eden öz-raporlama araçları öğrencilerin öğrenme sürecindeki örtük anlayışlarının ortaya çıkarılmasını sağlayabilir. Böylece öğrenenlerin hem üst-bilişsel bir farkındalık içerisinde öz-yönetim becerilerini sergilemesi sağlanabilir hem de öğrenme sürecinde ihtiyaç duydukları bilişsel ve duyuşsal destek unsurları ortaya çıkarabilir.

Problem çözme ve BD etkinliklerinin en önemli unsurlarından birisi zaman yönetimidir. Mevcut çalışmada öğrencilerin özellikle dersin uygulama kısmındaki gerçekleştirdikleri problem çözme etkinliklerinde zaman baskısı yaşadığı görülmüştür. Bu anlamda BD etkinliklerinde süreç yönetimini kolaylaştırmak adına öğrencilerin iş planı ve zaman çizelgesi araçlarından yararlanması teşvik edilebilir.

Öğrencilerin dersin kuramsal oturumunda yapılan ders sunumlarından sıkıldıkları görülmüştür. Ders sunumlarının çoklu ortam kaynaklarıyla zenginleştirilmesi ve kuramsal oturumların içerisine kısa uygulama oturumlarının dâhil edilmesi öğrencilerin aktifleştirilmesine katkı sağlayabilir. Öte yandan, kod yazma sürecindeki katı prosedürel kuralların öğrenci kaygısını arttırdığı gözlemlenmiştir. Programlamaya giriş dersleri için yapılacak öğretimsel düzenlemelerde hem nesneye yönelik hem de robot programlama grupları için sırasıyla Scratch ve Lego Mindstorms gibi blok tabanlı programlama araçlarından yararlanılabilir.

Fiziksel robotik setleri için tedarik ve saklama olanağının kısıtlı olduğu eğitim ortamlarında bazı robotik programlama araçlarının (Robot C sanal robot, Robot Virtual Worlds) sunduğu 3B ve 2B sanal robot simülasyonları kullanılabilir. Sanal robot uygulamaları fiziksel robotlarda zaman zaman karşılaşılan arıza ve kaynak yetersizliği (piller, sensörler) sorunlarını ortadan kaldırmaktadır. Ayrıca simülasyon araçları robotik etkinliklerinin yüz yüze eğitimin ötesinde uzaktan eğitime de kolaylıkla entegre edilmesine olanak sağlayabilir.

Mevcut çalışmada öğrencilerin ders dışında öğrenme etkinlikleri için yeterince zaman ayırmadıkları belirlenmiştir. Öğrencilerin programlamaya ilişkin ders dışı ilgilerinin arttırılması için onlara okul dışında da öğrenme fırsatlarının yaratılması önemlidir. Özellikle web ya da mobil uygulamalar aracılığıyla gerçekleştirilebilecek programlama görevleri öğrencilerin ders dışında da programlama etkinlikleriyle meşgul olmalarını sağlayabilir.

Çalışmada gerçekleştirilen BD ve problem çözme etkinliklerinin öğrencilerin otantik yaşam problemlerine yaklaşımları ve problem çözme becerisi algıları üzerinde yeterince etkili olmadığı görülmüştür. Bu nedenle öğrencilerin BD becerilerinin geliştirilmesi için düzenlenecek öğrenme etkinliklerinde kullanılacak problemlerin gündelik hayatla daha ilişkili olacak şekilde seçilmesinin öğrenenlerin

bu becerilerini otantik yaşam problemlerine aktarmalarını kolaylaştırabileceği düşünülmektedir.

Kaynaklar

Agalianos, A., Noss, R., & Whitty, G. (2001). Logo in mainstream schools: The struggle over the soul of an educational innovation. *British Journal of Sociology of Education, 22*(4), 479-500.

Aho, A. V. (2012). Computation and computational thinking. *The Computer Journal, 55*(7), 832-835. doi:10.1093/comjnl/bxs074

Alimisis, D. (2009). Robotic technologies as vehicles of new ways of thinking about constructivist teaching and learning: The TERECoP project. *IEEE Robotics & Automation Magazine, 16*(3), 21-23. doi:10.1109/MRA.2009.933619

Ames, C., & Archer, J. (1988). Achievement goals in the classroom: Students' learning strategies and motivation processes. *Journal of Educational Psychology, 80*(3), 260-267.

Anderman, E. M., & Wolters, C. A. (2006). Goals, values, and affect: Influences on student motivation. In P. A. Alexander & P. H. Winne (Eds.), *Handbook of educational psychology.* (pp. 369-389). Mahwah, NJ, US: Lawrence Erlbaum Associates Publishers.

Anderson, J. R. (2009). *Cognitive psychology and its implications* (7th ed.). NY, USA: Worth Publishers.

Andre, T. (1986). Problem solving and education. In G. D. Phye & T. Andre (Eds.), *Cognitive classroom learning: Understanding, thinking, and problem solving* (pp. 169-204). Orlando, USA: Academic Press.

Angeli, C., Voogt, J., Fluck, A., Webb, M., Cox, M., Malyn-Smith, J., & Zagani, J. (2016). A K-6 computational thinking curriculum framework: Implication for teacher knowledge. *Educational Technology & Society, 19*(3), 47-57.

Balanskat, A., & Engelhardt, K. (2014). Computing our future: Computer programming and coding-priorities, school curricula and initiatives across Europe. Brussels, Belgium: European Schoolnet. Retrieved from http://www.eun.org/resources/detail?publicationID=481

Bandura, A. (1986). *Social foundations of thought and action: A social cognitive theory*. Englewood Cliffs, NJ, US: Prentice-Hall.

Barab, S., Thomas, M., Dodge, T., Carteaux, R., & Tuzun, H. (2005). Making learning fun: Quest Atlantis, a game without guns. *Educational Technology Research and Development, 53*(1), 86-107. doi:10.1007/bf02504859

Barr, D., Harrison, J., & Conery, L. (2011). Computational thinking: A digital age skill for everyone. *Learning & Leading with Technology, 38*(6), 20-23.

Barr, V., & Stephenson, C. (2011). Bringing computational thinking to K-12: what is Involved and what is the role of the computer science education community? *ACM Inroads, 2*(1), 48-54. doi:10.1145/1929887.1929905

Barrows, H. S., & Tamblyn, R. M. (1980). *Problem-Based Learning: An Approach to Medical Education.* NY, USA: Springer.

Basawapatna, A. (2016). Alexander meets Michotte: A simulation tool based on pattern programming and phenomenology. *Journal of Educational Technology & Society, 19*(1), 277-291.

Bassey, M. (1999). *Case study research in educational settings*. Buckingham, UK: Open University Press.

Baş, T., & Akturan, U. (2008). *Nitel araştırma yöntemleri: Nvivo 7.0 ile nitel veri analizi* (1st ed.). Ankara, Türkiye: Seçkin Yayıncılık.

Bauer, A., Butler, E., & Popović, Z. (2015). *Approaches for teaching computational thinking strategies in an educational game: A position paper*. Paper presented at the 2015 IEEE Blocks and Beyond Workshop (Blocks and Beyond), Atlanta, Georgia, USA.

Bayrak, F., & Usluel, Y. K. (2011). The effect of blogging on reflective thinking skill. *H. U. Journal of Education, 40*(2011), 93-104.

Bdeir, A. (2009, 16-18 Feb.). *Electronics as material: littleBits.* Paper presented at the 3rd International Conference on Tangible and Embedded Interaction, Cambridge, United Kingdom.

Bell, T., Alexander, J., Freeman, I., & Grimley, M. (2009). Computer science unplugged: School students doing real computing without computers. *The New Zealand Journal of Applied Computing and Information Technology, 13*(1), 20-29.

Ben-Ari, M. (2001). Constructivism in computer science education. *Journal of Computers in Mathematics and Science Teaching, 20*(1), 45-73.

Bergen, D. (2001). Technology in the classroom: Learning in the robotic world: Active or reactive? *Childhood Education, 77*(4), 249-250.

Berland, M. (2016). Making, tinkering, and computational literacy. In K. Peppler, E. R. Halverson, & Y. B. Kafai (Eds.), *Makeology: Makers as learners* (1st ed.). New York: Routledge.

Berland, M., Martin, T., Benton, T., Smith, C. P., & Davis, D. (2013). Using learning analytics to understand the learning pathways of novice programmers. *Journal of the Learning Sciences, 22*(4), 564-599. doi:10.1080/10508406.2013.836655

Berland, M., & Wilensky, U. (2015). Comparing virtual and physical robotics environments for supporting complex systems and computational thinking. *Journal of Science Education and Technology, 24*(5), 628-647. doi:10.1007/s10956-015-9552-x

Bers, M. U., Flannery, L., Kazakoff, E. R., & Sullivan, A. (2014). Computational thinking and tinkering: Exploration of an early childhood robotics curriculum. *Computers & Education, 72*, 145-157. doi:10.1016/j.compedu.2013.10.02

Blumenfeld, P. C. (1992). Classroom learning and motivation: Clarifying and expanding goal theory. *Journal of Educational Psychology, 84*(3), 272-281.

Bork, A. (1981). *Leaming with computers*. Bedford, MA: Digital Press.

Bransford, J. D., & Stein, B. S. (1984). *The ideal problem solver: A guide for improving thinking, learning, and creativity* (1st ed.). NY, USA: W. H. Freeman & Company.

Brennan, K., & Resnick, M. (2012). *New frameworks for studying and assessing the development of computational thinking.* Paper presented at the 2012 Annual Meeting of the American Educational Research Association, Vancouver, Canada.

Brock, J. D., Bruce, R. F., & Reiser, S. L. (2009). Using Arduino for introductory programming courses. *Journal of Computing Sciences in Colleges, 25*(2), 129-130.

Brockbank, A., & McGill, I. (2007). *Facilitating reflective learning in higher education* (2nd ed.). New York: Open University Press.

Bruner, J. S. (1960). *The process of education*. Cambridge, MA: Harvard University Press.

Bryman, A. (2006). Integrating quantitative and qualitative research: how is it done? *Qualitative Research, 6*(1), 97-113. doi:10.1177/1468794106058877

Buechley, L., & Eisenberg, M. (2008). The LilyPad Arduino: Toward wearable engineering for everyone. *IEEE Pervasive Computing, 7*(2), 12-15. doi:10.1109/MPRV.2008.38

Bundy, A. (2007). Computational thinking is pervasive. *Journal of Scientific and Practical Computing, 1*(2), 67-69.

Burke, Q., & Kafai, Y. B. (2010). Programming & storytelling: opportunities for learning about coding & composition. In *Proceedings of the 9th International Conference on Interaction Design and Children* (pp. 348-351). New York: ACM.

Burleson, W. S., Harlow, D. B., Nilsen, K. J., Perlin, K., Freed, N., Jensen, C. N., . . . Muldner, K. (2018). Active learning environments with robotic tangibles: children's physical and virtual spatial programming experiences. *IEEE Transactions on Learning Technologies, 11*(1), 96-106. doi:10.1109/tlt.2017.2724031

Burton, J. K., Moore, D. M., & Magliaro, S. G. (2004). Behaviorism and instructional technology. In D. Jonassen (Ed.), *Handbook of Research on Educational Communications and Technology* (2nd ed., pp. 3-36). Mahwah, NJ: Lawrence Erlbaum.

Carbonaro, M., Szafron, D., Cutumisu, M., & Schaeffer, J. (2010). Computer-game construction: A gender-neutral attractor to computing science. *Computers & Education, 55*(3), 1098-1111.

Carpenter, F. (1974). *The skinner primer: Behind freedom and dignity*. NY, USA: The Free Press.

Carter, L. (2006). Why students with an apparent aptitude for computer science don't choose to major in computer science. *SIGCSE Bulletin, 38*(1), 27-31. doi:10.1145/1124706.1121352

Chamot, A. U., Dale, M., O'Malley, J. M., & Spanos, G. A. (1992). Learning and problem solving strategies of ESL students. *Bilingual Research Journal, 16*(3-4), 1-28.

Chang, C. K. (2014). Effects of using Alice and Scratch in an introductory programming course for corrective instruction. *Journal of Educational Computing Research, 51*(2), 185-204. doi:10.2190/EC.51.2.c

Chen, G. H., Shen, J., Barth-Cohen, L., Jiang, S. Y., Huang, X. T., & Eltoukhy, M. (2017). Assessing elementary students' computational thinking in everyday reasoning and robotics programming. *Computers & Education, 109*, 162-175. doi:10.1016/j.compedu.2017.03.001

Chi, M. T. H., & Glaser, R. (1985). Problem-solving ability. In R. J. Sternberg (Ed.), *Human abilities: An information-processing approach* (pp. 227–250). NY, USA: Freeman.

Chih-Wei, C., Jih-Hsien, L., Po-Yao, C., Chin-Yeh, W., & Gwo-Dong, C. (2010). Exploring the possibility of using humanoid robots as instructional tools for teaching a second language in primary school. *Journal of Educational Technology & Society, 13*(2), 13-24.

Cohen, J. (1990). Things I have learned (so far). *American psychologist, 45*(12), 1304-1312. Retrieved from http://dx.doi.org/10.1037/0003-066X.45.12.1304

Cohen, J. (1994). The earth is round (p<. 05). *American psychologist, 49*(12), 997-1003.

Colburn, T., & Shute, G. (2007). Abstraction in computer science. *Minds and Machines, 17*(2), 169-184. doi:10.1007/s11023-007-9061-7

Cooper, S., Dann, W., & Pausch, R. (2000). Alice: a 3-D tool for introductory programming concepts. *Journal of Computing Sciences in Colleges*, 15(5):107-116.

Creswell, J. W., & Clark, V. L. P. (2007). *Designing and conducting mixed methods research* (2nd ed.). Thousand Oaks, California: Sage Publication.

Csikszentmihalyi, M. (1975). *Beyond boredom and anxiety.* San Francisco, CA: Jossey-Bass.

Csikszentmihalyi, M. (1990). *Flow: The psychology of optimal experience*. NY, USA: Harper & Row.

Csikszentmihalyi, M. (1993). *The evolving self: A psychology for the third millennium*. NY, USA: HarperCollins Publishers.

Dagiene, V., & Stupuriene, G. (2016). Bebras-A sustainable community building model for the concept based learning of informatics and computational thinking. *Informatics in Education, 15*(1), 25-44.

Daily, S. B., & Eugene, W. (2013). Preparing the future STEM workforce for diverse environments. *Urban Education, 48*(5), 682-704. doi:10.1177/0042085913490554

Demir, E., Saatçioğlu, Ö., & İmrol, F. (2016). Uluslararası dergilerde yayımlanan eğitim araştırmalarının normallik varsayımları açısından incelenmesi. *Current Research in Education, 2*(3), 130-148.

Demir, Ö., & Seferoğlu, S. S. (2017). Yeni kavramlar, farklı kullanımlar: Bilgi-işlemsel düşünmeyle ilgili bir değerlendirme. In H. F. Odabaşı, B. Akkoyunlu, & A. İşman (Eds.), *Eğitim teknolojileri okumaları 2017* (pp. 801-830). Ankara, Türkiye: Turkish Online Journal of Educational Technology.

Denning, P. J. (2009). The profession of IT beyond computational thinking. *Communications of the ACM, 52*(6), 28-30.

Denning, P. J., & McGettrick, A. (2005). Recentering computer science. *Communications of the ACM, 48*(11), 15-19. doi:10.1145/1096000.1096018

diSessa, A. A. (2001). *Changing minds: Computers, learning, and literacy* (1st ed.). Cambridge, MA: MIT Press.

Ebel, R. L. (1972). *Essentials of educational measurement* (1st ed.). Oxford, England: Prentice-Hall.

Edelson, D. C. (2001). Learning-for-use: A framework for the design of technology-supported inquiry activities. *Journal of Research in Science Teaching, 38*(3), 355-385.

Edelson, D. C., Gordin, D. N., & Pea, R. D. (1999). Addressing the challenges of inquiry-based learning through technology and curriculum design. *Journal of the Learning Sciences, 8*(3-4), 391-450.

Falloon, G. (2016). An analysis of young students' thinking when completing basic coding tasks using Scratch Jnr. on the iPad. *Journal of Computer Assisted Learning, 32*(6), 576-593. doi:10.1111/jcal.12155

Fang, A. D., Chen, G. L., Cai, Z. R., Cui, L., & Harn, L. (2017). Research on blending learning flipped class model in colleges and universities based on computational thinking - "database principles" for example. *Eurasia Journal of Mathematics Science and Technology Education, 13*(8), 5747-5755. doi:10.12973/eurasia.2017.01024a

Felleisen, M., Findler, R. B., Flatt, M., & Krishnamurthi, S. (2004). The TeachScheme! project: Computing and programming for every student. *Computer Science Education, 14*(1), 55-77. doi:10.1076/csed.14.1.55.23499

Feurzeig, W., Papert, S. A., & Lawler, B. (2011). Programming-languages as a conceptual framework for teaching mathematics. *Interactive Learning Environments, 19*(5), 487-501. doi:10.1080/10494820903520040

Fields, D. A., Searle, K. A., Kafai, Y. B., & Min, H. S. (2012). *Debuggems to assess student learning in e-textiles*. Paper presented at the 43rd ACM technical symposium on Computer Science Education, Raleigh, North Carolina, USA.

Fleiss, J. L. (1981). The measurement of interrater agreement, statistical methods for rates and proportions (2nd ed.). NY, USA: Wiley-Interscience.

Flórez, B. F., Casallas, R., Hernández, M., Reyes, A., Restrepo, S., & Danies, G. (2017). Changing a generation's way of thinking: Teaching computational thinking through programming. *Review of Educational Research, 87*(4), 834-860. doi:10.3102/0034654317710096

Flowers, T. R., & Gossett, K. A. (2002). Teaching problem solving, computing, and information technology with robots. *Journal of Computing Sciences in Colleges, 17*(6), 45-55.

Fowler, A., Fristce, T., & MacLauren, M. (2012). Kodu game lab: A programming environment. *The Computer Games Journal, 1*(1), 17-28.

Fraenkel, J. R., & Wallen, N. E. (2006). *How to design and evaluate research in education* (6th ed.). NY, USA: McGraw-Hill.

Fraser, N. (2015, Oct. 22). *Ten things we've learned from Blockly.* Paper presented at the 2015 IEEE Blocks and Beyond Workshop (Blocks and Beyond), Atlanta, GA, USA.

Frederiksen, N. (1983). Implications of cognitive theory for instruction in problem solving. *ETS Research Report Series, 1983*(1), 363-407.

Furber, S. (2012). Shut down or restart? The way forward for computing in UK schools. *London, UK: The Royal Society*. Retrieved from https://royalsociety.org/-/media/education/computing-in-schools/2012-01-12-computing-in-schools.pdf

García-Peñalvo, F. J., & Mendes, A. J. (2018). Exploring the computational thinking effects in pre-university education. *Computers in Human Behavior, 80*, 407-411. doi:https://doi.org/10.1016/j.chb.2017.12.005

Garcia, P. G. F., & De la Rosa, F. (2016). RoBlock–web app for programming learning. *International Journal of Emerging Technologies in Learning, 11*(12), 45-53. doi:https://doi.org/10.3991/ijet.v11i12.6004

Garneli, V., & Chorianopoulos, K. (2018). Programming video games and simulations in science education: exploring computational thinking through code analysis. *Interactive Learning Environments, 26*(3), 386-401. doi:10.1080/10494820.2017.1337036

Gick, M. L., & Holyoak, K. J. (1980). Analogical problem solving. *Cognitive Psychology, 12*(3), 306-355. doi:https://doi.org/10.1016/0010-0285(80)90013-4

Gick, M. L., & Holyoak, K. J. (1983). Schema induction and analogical transfer. *Cognitive Psychology, 15*(1), 1-38. Retrieved from https://doi.org/10.1016/0010-0285(83)90002-6

Glaser, B., & Strauss, A. L. (1967). Discovery of grounded theory: Strategies for qualitative research. Chicago: Aldine.

Gorman, H., & Bourne, L. E. (1983). Learning to think by learning LOGO: Rule learning in third-grade computer programmers. *Bulletin of the Psychonomic Society, 21*(3), 165-167.

Greeno, J. G., Collins, A. M., & Resnick, L. B. (1996). Cognition and learning. In D. C. Berliner & R. C. Calfee (Eds.), *Handbook of educational psychology* (pp. 15-46). London, England: Prentice Hall International.

Greitzer, F. L., Kuchar, O. A., & Huston, K. (2007). Cognitive science implications for enhancing training effectiveness in a serious gaming context. *Journal on Educational Resources in Computing, 7*(3), 1-16. doi:10.1145/1281320.1281322

Gretter, S., & Yadav, A. (2016). Computational thinking and media & information literacy: An integrated approach to teaching twenty-first century skills. *TechTrends, 60*(5), 510-516. doi:10.1007/s11528-016-0098-4

Gross, P., & Kelleher, C. (2010). Non-programmers identifying functionality in unfamiliar code: strategies and barriers. *Journal of Visual Languages & Computing, 21*(5), 263-276.

Grover, S., & Pea, R. (2013). Computational thinking in K–12: A review of the state of the field. *Educational Researcher, 42*(1), 38-43. doi:10.3102/0013189x12463051

Guzdial, M. (2008). Education: Paving the way for computational thinking. *Communications of the ACM, 51*(8), 25-27. doi:10.1145/1378704.1378713

Guzdial, M., & Forte, A. (2005). Design process for a non-majors computing course. *ACM SIGCSE Bulletin, 37*(1), 361-365.

Gülbahar, Y. (Ed.) (2018). *Bilgi İşlemsel Düşünmeden Programlamaya* (2. baskı). Ankara: Pegem Akademi.

Hair, J. F., Anderson, R. E., Tatham, R. L., & Black, W. C. (1995). *Multivariate data analysis: With readings* (4th ed.). Englewood Cliffs, NJ: Prentice Hall International Editions.

Hartley, J., & McWilliam, K. (2009). Computational power meets human contact. In J. Hartley & K. McWilliam (Eds.), *Story circle: Digital storytelling around the world* (pp. 3-15). Oxford, UK: Wiley-Blackwell.

Hazzan, O. (2002, June 24-26). *Reducing abstraction level when learning computability theory concepts.* Paper presented at the 7th annual conference on Innovation and Technology in Computer Science Education (ITICSE), Aarhus, Denmark.

Hazzan, O., & Kramer, J. (2007). Abstraction in computer science & software engineering: A pedagogical perspective. *Frontier Journal, 4*(1), 6-14.

Heppner, P. P. (1988). *The problem solving inventory (PSI): Research manual*. Palo Alto, CA: Consulting Psychologists Press.

Heppner, P. P., & Petersen, C. H. (1982). The development and implications of a personal problem-solving inventory. *Journal of Counseling Psychology, 29*(1), 66-75. doi:10.1037/0022-0167.29.1.66

Hill, J. H., Houle, B. J., Merritt, S. M., & Stix, A. (2008). Applying abstraction to master complexity: The comparison of abstraction ability in computer science majors with students in other disciplines. *In Proceedings of the 2nd International Workshop on The Role of Abstraction in Software Engineering* (pp. 15-21). Leipzig, Germany: ACM.

Hmelo-Silver, C. E. (2004). Problem-based learning: What and how do students learn? *Educational Psychology Review, 16*(3), 235-266. doi:10.1023/B:EDPR.0000034022.16470.f3

Hmelo-Silver, C. E., Duncan, R. G., & Chinn, C. A. (2007). Scaffolding and achievement in problem-based and inquiry learning: A response to Kirschner, Sweller, and Clark (2006). *Educational Psychologist, 42*(2), 99-107. doi:10.1080/00461520701263368

Hoganson, K. (2010). Teaching programming concepts with GameMaker. *Journal of Computing Sciences in Colleges, 26*(2), 181-188.

Horwitz, P. (1995). Linking models to data: Hypermodels for science education. *The High School Journal, 79*(2), 148-156.

Isaksen, S. G., & Gaulin, J. P. (2005). A reexamination of brainstorming research: Implications for research and practice. *Gifted Child Quarterly, 49*(4), 315-329. doi:10.1177/001698620504900405

Israel, M., Pearson, J. N., Tapia, T., Wherfel, Q. M., & Reese, G. (2015). Supporting all learners in school-wide computational thinking: A cross-case qualitative analysis. *Computers & Education, 82*, 263-279. doi:10.1016/j.compedu.2014.11.022

Jacobson, M. J., & Wilensky, U. (2006). Complex systems in education: Scientific and educational importance and implications for the learning sciences. *Journal of the Learning Sciences, 15*(1), 11-34. doi:10.1207/s15327809jls1501_4

Jaipal-Jamani, K., & Angeli, C. (2017). Effect of robotics on elementary preservice teachers' self-efficacy, science learning, and computational thinking. *Journal of Science Education and Technology, 26*(2), 175-192. doi:10.1007/s10956-016-9663-z

Jona, K., & Vondracek, M. (2013). A remote radioactivity experiment. *The Physics Teacher, 51*(1), 25-27.

Jonassen, D. H. (1995). Computers as cognitive tools: Learningwith technology, notfrom technology. *Journal of Computing in Higher Education, 6*(2), 40-73.

Jonassen, D. H. (2000). Toward a design theory of problem solving. *Educational Technology Research and Development, 48*(4), 63-85. doi:10.1007/bf02300500

Jun, S., Han, S., Kim, H., & Lee, W. (2014). Assessing the computational literacy of elementary students on a national level in Korea. *Educational Assessment Evaluation and Accountability, 26*(4), 319-332. doi:10.1007/s11092-013-9185-7

Kahn, K. (1996). ToonTalk™—An animated programming environment for children. *Journal of Visual Languages & Computing, 7*(2), 197-217. Retrieved from https://doi.org/10.1006/jvlc.1996.0011

Kalelioğlu, F. (2015). A new way of teaching programming skills to K-12 students: Code.org. Computers in Human Behavior, 52, 200-210. Retrieved from https://doi.org/10.1016/j.chb.2015.05.047

Kalelioğlu, F., & Gülbahar, Y. (2014). The effects of teaching programming via Scratch on problem solving skills: A discussion from learners' perspective. *Informatics in Education, 13*(1), 33-50.

Kalelioğlu, F., Gülbahar, Y., Akçay, S., & Doğan, D. (2014, Sep. 22-25). *Curriculum integration ideas for improving the computational thinking skills of learners through programming via Scratch*. Paper presented at the 7th International Conference on Informatics in Schools: Situation, Evoluation and Perspectives (ISSEP), İstanbul, Turkey.

Kalelioğlu, F., Gülbahar, Y., & Kukul, V. (2016). A framework for computational thinking based on a systematic research review. *Baltic Journal of Modern Computing, 4*(3), 583-596.

Kelleher, C. (2006). Motivating programming: Using storytelling to make computer programming attractive to middle school girls (Doctoral dissertation). Carnegie-Mellon University, Pittsburgh PA (CMU-CS-06-171).

Keller, J. M. (1983). Motivational design of instruction. In C. M. Reigeluth (Ed.), *Instructional-design theories and models: An overview of their current status* (Vol. 1, pp. 383-434). Hillsdale, NJ: Erlbaum.

Keller, J. M. (1987a). *IMMS: Instructional materials motivation survey.* Tallahassee, FL: Florida State University.

Keller, J. M. (1987b). Strategies for stimulating the motivation to learn. *Performance and Instruction, 26*(8), 1-7. Retrieved from https://doi.org/10.1002/pfi.4160260802

Keller, J. M. (2010). Motivational design for learning and performance: The ARCS model approach. New York, NY: Springer US.

Keskin, H. Ü. (2010). Kovaryans analizi (ANCOVA). In Ş. Kalaycı (Ed.), *SPSS uygulamalı çok değişkenli istatistik teknikleri* (5. baskı). Ankara: Asil Yayın Dağıtım.

Kılıç, D., & Sağlam, N. (2009). Öğrencilerin mantıksal düşünme yeteneklerinin bazı değişkenler açısından incelenmesi. *Ege Eğitim Dergisi, 2*(10), 23-38.

Kim, B., Kim, T., & Kim, J. (2013). Paper-and-pencil programming strategy toward computational thinking for non-majors: Design your solution. *Journal of Educational Computing Research, 49*(4), 437-459. doi:10.2190/EC.49.4.b

Kirk, J., & Miller, M. L. (1986). *Reliability and validity in qualitative research*. Beverly Hills, CA: Sage Publications.

Kirk, R. E. (1996). Practical significance: A concept whose time has come. *Educational and Psychological Measurement, 56*(5), 746-759.

Kirk, R. E. (2001). Promoting good statistical practices: Some suggestions. *Educational and Psychological Measurement, 61*(2), 213-218.

Kirschner, P. A., Sweller, J., & Clark, R. E. (2006). Why minimal guidance during instruction does not work: An analysis of the failure of constructivist, discovery, problem-based, Experiential, and Inquiry-Based Teaching. *Educational Psychologist, 41*(2), 75-86. doi:10.1207/s15326985ep4102_1

Korkmaz, H. (2002). *Fen eğitiminde proje tabanlı öğrenmenin yaratıcı düşünme, problem çözme ve akademik risk alma düzeylerine etkisi* (Yayımlanmamış doktora tezi), Hacettepe Üniversitesi, Sosyal Bilimler Enstitüsü, Ankara.

Korkmaz, Ö., Çakır, R., & Özden, M. Y. (2016). Bilgisayarca düşünme beceri düzeyleri ölçeğinin (BDBD) ortaokul düzeyine uyarlanması. *Gazi Eğitim Bilimleri Dergisi, 1*(2), 143-162.

Korkmaz, Ö., Çakır, R., Özden, M. Y., Oluk, A., & Sarıoğlu, S. (2015). Bireylerin bilgisayarca düşünme becerilerinin farklı değişkenler açısından incelenmesi. *Ondokuz Mayıs Üniversitesi Eğitim Fakültesi Dergisi, 34*(2), 68-87.

Koschmann, T. D., Feltovich, P. J., Myers, A. C., & Barrows, H. S. (1992). Implications of CSCL for problem-based learning. *ACM SIGCUE Outlook, 21*(3), 32-35.doi: doi: 10.1145/130893.130902

Köksal, A. (1981). *Bilişim terimleri sözlüğü*. Ankara, Türkiye: Türk Dil Kurumu Yayınları.

Köksal, A. (2010). *Adı bilgisayar olsun Aydın Köksal yazılarından bir seçki* İstanbul, Türkiye: Cumhuriyet Kitapları.

Kölling, M. (2010). The greenfoot programming environment. *ACM Transactions on Computing Education, 10*(4), 1-21. doi:10.1145/1868358.1868361

Krajcik, J. S., & Shin, N. (2014). Project-based learning. In R. K. Sawyer (Ed.), *The Cambridge handbook of the learning sciences* (2 ed., pp. 275-297). Cambridge: Cambridge University Press.

Kuhn, D., Black, J., Keselman, A., & Kaplan, D. (2000). The development of cognitive skills to support inquiry learning. *Cognition and Instruction, 18*(4), 495-523. doi:10.1207/S1532690XCI1804_3

Kutu, H., & Sözbilir, M. (2011). Öğretim materyalleri motivasyon anketinin Türkçeye uyarlanması: Güvenirlik ve geçerlik çalışması. *Necatibey Eğitim Fakültesi Elektronik Fen ve Matematik Eğitimi Dergisi, 5*(1), 292-312.

Latham, G. P., & Locke, E. A. (1979). Goal setting—A motivational technique that works. *Organizational Dynamics, 8*(2), 68-80. doi:https://doi.org/10.1016/0090-2616(79)90032-9

Leech, N. L., & Onwuegbuzie, A. J. (2002). *A call for greater use of nonparametric statistics*. Paper presented at the annual meeting of the Mid-South Educational Research Association, Chattanooga, TN.

Leonard, J., Buss, A., Gamboa, R., Mitchell, M., Fashola, O. S., Hubert, T., & Almughyirah, S. (2016). Using robotics and game design to enhance children's self-efficacy, STEM attitudes, and computational thinking skills. *Journal of Science Education and Technology, 25*(6), 860-876. doi:10.1007/s10956-016-9628-2

Lincoln, Y., & Guba, E. (1985). *Naturalistic inquiry*. Newbury Park, CA: Sage.

Linn, M. C., Clark, D., & Slotta, J. D. (2003). WISE design for knowledge integration. *Science education, 87*(4), 517-538.

Litts, B. K., Kafai, Y. B., Lui, D. A., Walker, J. T., & Widman, S. A. (2017). Stitching codeable circuits: High school students' learning about circuitry and coding with electronic textiles. *Journal of Science Education and Technology, 26*(5), 494-507. doi:10.1007/s10956-017-9694-0

Locke, E. A., & Latham, G. P. (1990). *A theory of goal setting & task performance*. Englewood Cliffs: Prentice Hall.

Locke, E. A., & Latham, G. P. (2002). Building a practically useful theory of goal setting and task motivation: A 35-year odyssey. *American psychologist, 57*(9), 705-717. doi:10.1037/0003-066X.57.9.705

Ludi, S. (2012). Educational robotics and broadening participation in STEM for underrepresented student groups. In B. S. Barker, G. Nugent, N. Grandgenett, & V. I. Adamchuk (Eds.), *Robots in K-12 education: A new technology for learning* (pp. 343-361). Hersey: IGI Global.

Lye, S. Y., & Koh, J. H. L. (2014). Review on teaching and learning of computational thinking through programming: What is next for K-12? *Computers in Human Behavior, 41*(0), 51-61. doi:http://dx.doi.org/10.1016/j.chb.2014.09.012

Majherova, J., & Kralik, V. (2017). Innovative methods in teaching programming for future informatics teachers. *European Journal of Contemporary Education, 6*(3), 390-400. doi:10.13187/ejced.2017.3.390

Maslow, A. H. (1970). *Motivation and personality* (2nd ed.). New York: Harper & Row.

Master, A., Cheryan, S., Moscatelli, A., & Meltzoff, A. N. (2017). Programming experience promotes higher STEM motivation among first-grade girls. *Journal of Experimental Child Psychology, 160*, 92-106. doi:https://doi.org/10.1016/j.jecp.2017.03.013

Matlock-Hetzel, S. (1997, 23-25 January). *Basic concepts in item and test analysis.* Paper presented at the Annual Meeting of the Southwest Educational Research Association, Austin, TX.

Mauch, E. (2001). Using technological innovation to improve the problem-solving skills of middle school students: Educators' experiences with the LEGO mindstorms robotic invention system. *The Clearing House, 74*(4), 211-213.

Mayer, R. E., Dyck, J. L., & Vilberg, W. (1986). Learning to program and learning to think: what's the connection? *Communications of the ACM, 29*(7), 605-610. doi:10.1145/6138.6142

McNerney, T. S. (2004). From turtles to tangible programming bricks: explorations in physical language design. *Personal and Ubiquitous Computing, 8*(5), 326-337.

Merriam, S. B. (1998). Qualitative research and case study applications in education: revised and expanded from case study research in education (1st ed.). San Francisco, CA: Jossey-Bass Publishers.

Merrick, K. E. (2010). An empirical evaluation of puzzle-based learning as an interest approach for teaching introductory computer science. *IEEE Transactions on Education, 53*(4), 677-680.

Mertens, D. M. (2003). Mixed methods and the politics of human research: The transformative-emancipatory perspective. In A. Tashakkori & C. Teddlie (Eds.), *Handbook of mixed methods in social and behavioral research* (2nd ed., pp. 135-164). Thousand Oaks, CA: Sage Publications.

Millî Eğitim Bakanlığı. (2017). *Millî Eğitim İstatistikleri, Örgün Eğitim 2016/'17*. https://sgb.meb.gov.tr/meb_iys_dosyalar/2017_09/08151328_meb_istatistikleri_orgun_egitim_2016_2017.pdf adresinden erişilmiştir.

Mouza, C., Yang, H., Pan, Y. C., Ozden, S. Y., & Pollock, L. (2017). Resetting educational technology coursework for pre-service teachers: A computational thinking approach to the development of technological pedagogical content knowledge (TPACK). *Australasian Journal of Educational Technology, 33*(3), 61-76. doi:10.14742/ajet.3521

Newell, A., & Simon, H. A. (1972). *Human problem solving*. Englewood Cliffs, NJ: Prentice-Hall.

Ngan, S. C., & Law, K. M. Y. (2015). Exploratory network analysis of learning motivation factors in e-learning facilitated computer programming courses. *Asia-Pacific Education Researcher, 24*(4), 705-717. doi:10.1007/s40299-014-0223-0

Nicholls, J. G. (1984). Achievement motivation: Conceptions of ability, subjective experience, task choice, and performance. *Psychological Review, 91*(3), 328-346. doi:10.1037/0033-295X.91.3.328

Norton, S. J., McRobbie, C. J., & Ginns, I. S. (2007). Problem solving in a middle school robotics design classroom. *Research in Science Education, 37*(3), 261-277. doi:10.1007/s11165-006-9025-6

Nugent, G., Barker, B., Welch, G., Grandgenett, N., Wu, C., & Nelson, C. (2015). A model of factors contributing to STEM learning and career orientation. *International Journal of Science Education, 37*(7), 1067-1088. doi:10.1080/09500693.2015.1017863

Numanoğlu, M., & Keser, H. (2017). Programlama öğretiminde robot kullanımı-Mbot örneği. *Bartın Üniversitesi Eğitim Fakültesi Dergisi, 6*(2), 497-515.

Özçınar, H. (2017). Hesaplamali düşünme araştırmalarının bibliyometrik analizi. *Eğitim Teknolojisi Kuram ve Uygulama, 7*(2), 149-171.

Özçınar, H., & Öztürk, E. (2018). Hesaplamalı düşünmenin öğretimine ilişkin özyeterlik algısı ölçeği: Geçerlik ve güvenirlik çalışması. *Pamukkale Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*(30), 173-195. doi:10.5505/pausbed.2018.82574

Özsoy, S., & Özsoy, G. (2013). Effect size reporting in educational research. *Elementary Education Online, 12*(2), 334-346.

Özüorçun, N. C., & Bicen, H. (2017). Does the inclusion of robots affect engineering students' achievement in computer programming courses? *Eurasia Journal of Mathematics Science and Technology Education, 13*(8), 4779-4787. doi:10.12973/eurasia.2017.00964a

Paliokas, I., Arapidis, C., & Mpimpitsos, M. (2011, May 4-6). *Playlogo 3d: A 3d interactive video game for early programming education: Let logo be a game.* Paper presented at the 3rd International Conference on Games and Virtual Worlds for Serious Applications (VS-GAMES), Athens, Greece.

Papert, S. (1980). Mindstorms: Children, computers, and powerful ideas. NY, USA: Basic Books.

Papert, S. (1993). The children's machine: Rethinking school in the age of the computer. NY, USA: Basic Books.

Papert, S. (1996). An exploration in the space of mathematics educations. *International Journal of Computers for Mathematical Learning, 1*(1), 95-123.

Park, J. (2015). Effect of robotics enhanced inquiry based learning in elementary science education in South Korea. *Journal of Computers in Mathematics and Science Teaching, 34*(1), 71-95.

Parr, C. S., Jones, T., & Songer, N. B. (2002). CyberTracker in BioKIDS: Customization of a PDA-based scientific data collection application for inquiry learning. In P. Bell, R. Stevens, & T. Satwicz (Eds.), *Keeping learning complex: The proceedings of the fifth international conference of the learning sciences.* (pp. 574-575). Mahwah, NJ: Erlbaum.

Patterson, D. A. (2005). Restoring the popularity of computer science. Communications of the ACM*, 48*(9), 25-28. doi:10.1145/1081992.1082011

Patton, M. Q. (1980). *Qualitative evaluation methods*. Beverly Hills, CA: Sage Publications.

Pea, R. D., & Kurland, D. M. (1984). On the cognitive effects of learning computer programming. *New Ideas in Psychology, 2*(2), 137-168. Retrieved from https://doi.org/10.1016/0732-118X(84)90018-7

Pellas, N. (2014). Exploring interrelationships among high school students' engagement factors in introductory programming courses via a 3D multi-user serious game created in Open Sim. *Journal of Universal Computer Science, 20*(12), 1608-1628.

Pellas, N., & Peroutseas, E. (2016). Gaming in Second Life via Scratch4SL: Engaging high school students in programming courses. *Journal of Educational Computing Research, 54*(1), 108-143. doi:10.1177/0735633115612785

Perkins, D. N., Hancock, C., Hobbs, R., Martin, F., & Simmons, R. (1986). Conditions of learning in novice programmers. *Journal of Educational Computing Research, 2*(1), 37-55. doi:10.2190/gujt-jcbj-q6qu-q9pl

Piaget, J. (1954). *The construction of reality in the child*. NY, USA: Basic Books.

Polya, G. (1957). *How to solve it?* (2nd ed.). Princeton, NJ: Princeton University Press.

Polya, G. (1973). *How to solve it: A new aspect of mathematical method* (2nd ed.). Princeton, NJ: Princeton University Press.

Porter, L. W., & Lawler, E. E. (1968). *Managerial attitudes and performance*. Homewood, IL: Dorsey Press.

Psycharis, S., & Kallia, M. (2017). The effects of computer programming on high school students' reasoning skills and mathematical self-efficacy and problem solving. *Instructional Science, 45*(5), 583-602. doi:10.1007/s11251-017-9421-5

Puntambekar, S., & Kolodner, J. L. (2005). Toward implementing distributed scaffolding: Helping students learn science from design. *Journal of Research in Science Teaching, 42*(2), 185-217.

Quade, D. (1967). Rank analysis of covariance. *Journal of the American Statistical Association, 62*(320), 1187-1200. doi:10.1080/01621459.1967.10500925

Qualls, J. A., & Sherrell, L. B. (2010). Why computational thinking should be integrated into the curriculum. *Journal of Computing Sciences in Colleges, 25*(5), 66-71.

Quintana, C., Reiser, B. J., Davis, E. A., Krajcik, J., Fretz, E., Duncan, R. G., . . . Soloway, E. (2004). A scaffolding design framework for software to support science inquiry. *Journal of the Learning Sciences, 13*(3), 337-386. doi:10.1207/s15327809jls1303_4

Reigeluth, C. M., & Frick, T. W. (1999). Formative research: A methodology for creating and improving design theories. In C. M. Reigeluth (Ed.), *Instructional design theories and models* (Vol. 2, pp. 633-652). Mahwah, NJ: Lawrence-Erlbaum Associates.

Reiser, B. J. (2004). Scaffolding complex learning: The mechanisms of structuring and problematizing student work. *The Journal of the Learning sciences, 13*(3), 273-304.

Repenning, A., Webb, D. C., Brand, C., Gluck, F., Grover, R., Miller, S., . . . Muyang, S. (2014). Beyond Minecraft: Facilitating computational thinking through modeling and programming in 3D. *Computer Graphics and Applications, IEEE, 34*(3), 68-71. doi:10.1109/MCG.2014.46

Resnick, L. B. (1985). Cognition and instruction: Recent theories of human competence Psychology and learning. In B. L. Hammonds (Ed.)*, Psychology and learning: The master lecture series* (Vol.4, pp. 123-186). Washington, DC, US: American Psychological Association.

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., . . . Silverman, B. (2009). Scratch: programming for all. *Communications of the ACM, 52*(11), 60-67.

Robertson, J., & Howells, C. (2008). Computer game design: Opportunities for successful learning. *Computers & Education, 50*(2), 559-578.

Robins, A., Rountree, J., & Rountree, N. (2003). Learning and teaching programming: A review and discussion. *Computer Science Education, 13*(2), 137-172.

Rogalski, J., & Samurçay, R. (1990). Acquisition of programming knowledge and skills. In J. M. Hoc, T. R. G. Green, R. Samurçay, & D. J. Gilmore (Eds.), *Psychology of programming* (pp. 157-174). London: Academic Press.

Rogers, C., & Portsmore, M. (2004). Bringing engineering to elementary school. *Journal of STEM Education, 5*(3), 17-28.

Rogers, C. R. (1963). Actualizing tendency in relation to "Motives" and to consciousness. In M. R. Jones (Ed.), *Nebraska symposium on motivation.* (pp. 1-24). Oxford, England: U. Nebraska Press.

Roscoe, J. F., Fearn, S., & Posey, E. (2014). *Teaching computational thinking by playing games and building robots*. Paper presented at the 2014 International Conference on Interactive Technologies and Games, Nottingham, UK.

Rouse, R. (2005). *Game design theory and practice* (2nd ed.). Plano, TX: Jones & Bartlett Learning.

Rusk, N., Resnick, M., Berg, R., & Pezalla-Granlund, M. (2008). New pathways into robotics: Strategies for broadening participation. *Journal of Science Education and Technology, 17*(1), 59-69. doi:10.1007/s10956-007-9082-2

Rutherford, A. (2001). *Introducing ANOVA and ANCOVA: a GLM approach* (1st ed.). Thousand Oaks, CA: Sage Publications.

Saez-Lopez, J. M., & Sevillano-Garcia, M. L. (2017). Sensors, programming and devices in art education sessions. One case in the context of primary education. *Cultura Y Educacion, 29*(2), 350-384. doi:10.1080/11356405.2017.1305075

Savaşır, I., & Şahin, N. H. (1997). *Bilişsel-davranışçı terapilerde değerlendirme: Sık kullanılan ölçekler*. Ankara: Türk Psikologlar Derneği Yayınları.

Saygıner, Ş. (2017). *Blok tabanlı görsel ve metin tabanlı programlama öğretimlerinin erişi, mantıksal düşünme ve motivasyona etkileri*. Bilgisayar ve Öğretim Teknolojileri Eğitimi (Yayımlanmamış yüksek lisans tezi). Hacettepe Üniversitesi. Ankara.

Schunk, D. H. (2008). Attributions as motivators of self-regulated learning. In D. H. Schunk & B. J. Zimmerman (Eds.), *Motivation and self-regulated learning theory, research, and applications* (pp. 245–266). NY, USA: Taylor & Francis.

Schunk, D. H. (2012). *Learning theories: An educational perspective* (M. Şahin, Trans. 6th ed.). Boston, MA: Pearson.

Scott, M. J., Counsell, S., Lauria, S., Swift, S., Tucker, A., Shepperd, M., & Ghinea, G. (2015). Enhancing practice and achievement in introductory programming with a robot olympics. *IEEE Transactions on Education, 58*(4), 249-254.

Shute, V. J., Sun, C., & Asbell-Clarke, J. (2017). Demystifying computational thinking. *Educational Research Review, 22*, 142-158. Retrieved from https://doi.org/10.1016/j.edurev.2017.09.003

Simon, H. A. (1973). The structure of ill structured problems. *Artificial intelligence, 4*(3-4), 181-201.

Sipitakiat, A., Blikstein, P., & Cavallo, D. P. (2004, June 22-26). *GoGo board: augmenting programmable bricks for economically challenged audiences.* Paper presented at the 6th international conference on Learning sciences, Santa Monica, CA.

Sivan, E. (1986). Motivation in social constructivist theory. *Educational Psychologist, 21*(3), 209-233. doi:10.1207/s15326985ep2103_4

Slotta, J. D. (2004). The web-based Inquiry science environment (WISE): Scaffolding knowledge integration in the science classroom. In M. C. Linn, E. A. Davis, & P. Bell (Eds.), *Internet environments for science education* (pp. 203-232). Mahwah, NJ: Lawrence Erlbaum Associates Inc.

Small, R. V. (1999). Motivation in instructional design. In R. M. Branch & M. A. Fitzgerald (Eds.), *Educational media and technology yearbook 1999* (pp. 89-92). Englewood, CO: Libraries Unlimited.

Smith, D. C., Cypher, A., & Tesler, L. (2000). Programming by example: novice programming comes of age. *Communications of the ACM, 43*(3), 75-81.

Snodgrass, M. R., Israel, M., & Reese, G. C. (2016). Instructional supports for students with disabilities in K-5 computing: Findings from a cross-case analysis. *Computers & Education, 100*, 1-17. doi:10.1016/j.compedu.2016.04.011

Soloway, E. (1993). Should we teach students to program? *Communications of the ACM, 36*(10), 21-25.

Squire, K. (2003). Video games in education. International Journal of Intelligent Games & Simulation, 2(1), 49-62.

Stair, R., & Reynolds, G. (2013). *Principles of information systems* (6th ed.). Boston, MA: Course Technology – ITP.

Stolee, K. T., & Fristoe, T. (2011, Mar. 9-12). *Expressing computer science concepts through Kodu game lab.* Paper presented at the 42nd ACM Technical Symposium on Computer Science Education, Dallas, TX.

Sullivan, A. A., Bers, M. U., & Mihm, C. (2017). Imagining, playing, and coding with KIBO: Using robotics to foster computational thinking in young Children. In S. C. Kong, J. Sheldon, & K. Y. L (Eds.), *Conference Proceedings of International Conference on Computational Thinking Education 2017* (pp. 110-115). Hong Kong: The Education University of Hong Kong.

Sullivan, F. R. (2008). Robotics and science literacy: Thinking skills, science process skills and systems understanding. *Journal of Research in Science Teaching, 45*(3), 373-394.

Sullivan, F. R., & Heffernan, J. (2016). Robotic construction kits as computational manipulatives for learning in the STEM disciplines. *Journal of Research on Technology in Education, 48*(2), 105-128.

Sweetman, D., Badiee, M., & Creswell, J. W. (2010). Use of the transformative framework in mixed methods studies. *Qualitative Inquiry, 16*(6), 441-454. doi:10.1177/1077800410364610

Şahin, N., Şahin, N. H., & Heppner, P. P. (1993). Psychometric properties of the problem solving inventory in a group of Turkish university students. *Cognitive Therapy and Research, 17*(4), 379-396. doi:10.1007/bf01177661

Şahiner, A., & Kert, S. B. (2016). Komputasyonel düşünme kavramı ile ilgili 2006-2015 yılları arasındaki çalışmaların incelenmesi. *Avrupa Bilim ve Teknoloji Dergisi, 5*(9), 38-43.

Şenaras, A. E., & Sezen, H. K. (2017). Sistem Düşüncesi. *Journal of Life Economics, 4*(1), 39-58.

Taub, R., Armoni, M., & Ben-Ari, M. (2012). CS unplugged and middle-school students' views, attitudes, and intentions regarding CS. *ACM Transactions on Computing Education, 12*(2), 1-29. doi: 10.1145/2160547.2160551

Thorndike, E. L. (1913). *The psychology of learning*. NY, USA: Teachers College Press.

Tisue, S., & Wilensky, U. (2004, May 16-21). *Netlogo: A simple environment for modeling complexity.* Paper presented at the International Conference on Complex Systems (ICCS 2004), Boston, MA.

Tolman, C. W. (1999). Society versus context in individual development: Does theory make a difference? In Y. Engeström, R. Miettinen, & R.-L. Punamaki (Eds.), *Perspectives on activity theory* (pp. 70–86). Cambridge, UK: Cambridge University Press.

Treffinger, D. J. (1995). Creative problem solving: Overview and educational implications. Educational Psychology Review, 7(3), 301-312.

Treffinger, D. J., & Isaksen, S. G. (2005). Creative problem solving: The history, development, and implications for gifted education and talent development. Gifted Child Quarterly, 49(4), 342-353. doi:10.1177/001698620504900407

Turbak, F., & Berg, R. (2002). Robotic design studio: Exploring the big ideas of engineering in a liberal arts environment. *Journal of Science Education and Technology, 11*(3), 237-253. doi:10.1023/a:1016376818781

Turner, J. C. (2001). Using context to enrich and challenge our understanding of motivational theory. In S. Volet & S. Jarvela (Eds.), *Motivation in learning contexts: Theoretical advances and methodological implications* (pp. 85-104). Elmsford, NY, US: Emerald Group Publishing.

Tüzün, H. (2004). *Motivating learners in educational computer games* (Unpublished doctoral dissertation). Indiana University, Bloomington, USA.

US National Research Council. (2010). *Report of a workshop on the scope and nature of computational thinking*. Washington DC: National Academies Press.

US National Research Council. (2011). Report of a workshop on the pedagogical aspects of computational thinking: National Academies Press.

Uslu, N. A., Mumcu, F., & Eğin, F. (2018). Görsel programlama etkinliklerinin ortaokul öğrencilerinin bilgi-işlemsel düşünme becerilerine etkisi. *Ege Eğitim Teknolojileri Dergisi, 2*(1), 19-31.

Vacha-Haase, T. (2001). Statistical significance should not be considered one of life's guarantees: Effect sizes are needed. *Educational and Psychological Measurement, 61*(2), 219-224.

Voogt, J., Fisser, P., Good, J., Mishra, P., & Yadav, A. (2015). Computational thinking in compulsory education: Towards an agenda for research and practice. *Education and Information Technologies, 20*(4), 715-728. doi:10.1007/s10639-015-9412-6

Vroom, V. H. (1964). *Work and motivation*. NY: Wiley.

Vygotsky, L. S. (1978). *Mind in society*. Cambridge, MA: Harvard University Press.

Wallas, G. (1926). *The art of thought* (1st ed.). London, UK: Jonathan Cape.

Wang, X. M., & Hwang, G. J. (2017). A problem posing-based practicing strategy for facilitating students' computer programming skills in the team-based learning mode. *Educational Technology Research and Development, 65*(6), 1655-1671. doi:10.1007/s11423-017-9551-0

Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining computational thinking for mathematics and science

classrooms. *Journal of Science Education and Technology, 25*(1), 127-147. doi:10.1007/s10956-015-9581-5

Weintrop, D., & Wilensky, U. (2012, Aug. 21-25). *RoboBuilder: A program-to-play constructionist video game.* Paper presented at the Constructionism 2012: Theory, Practice and Impact, Athens, Greece.

Wiersma, W., & Jurs, S. G. (1990). *Educational measurement and testing* (2nd ed.). Boston, MA: Allyn and Bacon.

Wilkerson-Jerde, M. H. (2014). Construction, categorization, and consensus: student generated computational artifacts as a context for disciplinary reflection. *Educational Technology Research and Development, 62*(1), 99-121. doi:10.1007/s11423-013-9327-0

Wing, J. M. (2006). Computational thinking. *Communications of the ACM, 49*(3), 33-35. doi:10.1145/1118178.1118215

Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 366*(1881), 3717-3725. doi:10.1098/rsta.2008.0118

Wing, J. M. (2011). Computational thinking-what and why? Retrieved from https://www.cs.cmu.edu/link/research-notebook-computational-thinking-what-and-why

Wlodkowski, R. J. (1978). *Motivation and teaching: A practical guide*. Washington, DC: National Education Association.

Wyeth, P. (2008). How young children learn to program with sensor, action, and logic blocks. *Journal of the Learning Sciences, 17*(4), 517-550. doi:10.1080/10508400802395069

Yang, T.-C., Hwang, G.-J., Yang, S. J. H., & Hwang, G.-H. (2015). A two-tier test-based approach to improving students' computer-programming skills in a web-based learning environment. *Educational Technology & Society, 18*(1), 198–210.

Yıldırım, A., & Şimşek, H. (2008). *Sosyal bilimlerde nitel araştırma yöntemleri* (7. baskı). Ankara: Seçkin Yayıncılık.

Yıldız, M., Çiftçi, E., & Karal, H. (2017). Bilişimsel düşünme ve programlama. In H. F. Odabaşı, B. Akkoyunlu, & A. İşman (Eds.), *Eğitim teknolojileri okumaları 2017* (pp. 75-86). Ankara, Türkiye: Turkish Online Journal of Educational Technology.

Zendler, A., & Klaudt, D. (2012). Central computer science concepts to research-based teacher training in computer science: An experimental study. *Journal of Educational Computing Research, 46*(2), 153-172.

Zhang, J. X., Liu, L., de Pablos, P. O., & She, J. (2014). The auxiliary role of information technology in teaching: Enhancing programming course using Alice. *International Journal of Engineering Education, 30*(3), 560-565.

Zhong, B. C., Wang, Q. Y., Chen, J., & Li, Y. (2016). An exploration of three-dimensional integrated assessment for computational thinking. *Journal of Educational Computing Research, 53*(4), 562-590.

Zuckerman, O., & Resnick, M. (2003, Apr. 5-10). *A physical interface for system dynamics simulation.* Paper presented at the Conference on Human Factors in Computing (CHI 2003) New Horizons, Fort Lauderdale, FL.

EK-A: Problem Çözme Etkinlik Örnekleri

PROGRAMLAMA TEMELLERİ DERSİ
EĞİTSEL ROBOT PROGRAMLAMA GRUBU

10. HAFTA LAB. UYGULAMALARI

Aşağıdaki etkinlikleri POLYA'nın problem çözme süreci aşamalarını dikkate alarak, algoritma ve akış diyagramlarını oluşturarak, RobotC yazılımını kullanarak yazınız.

1. Bir belediye, caddelerden birinde gürültü kirliliği konusunda bir çalışma yapmak istemektedir. Bu konuyla ilgili belediye personeli Ali Bey bu caddede 20 defa ölçüm yaptıktan sonra bir rapor hazırlayacaktır. Ali bey raporunda, ortalama gürültü değerini müdürüne sunmak durumundadır. Bu raporu hazırlaması için Ali beye yardım edecek programı yazınız.

2. Sınırda mayın arama için görevlendirilen "Robot-er" 60 sn boyunca hareket etmekte ve bu hareketi boyunca her 1 sn'de bir ışık ölçümü yapmaktadır. Işık şiddeti ölçümlerini bir diziye kaydeden bu robot, bir mayın tespit ettiğinde (ışık şiddeti 30'un altına düştüğünde) bir uyarı sinyali (SoundBeepBeep) çıkarmakta ve mayınları saymaktadır. Robot-er'in bu şekilde mayın taraması yapmasını sağlayan programı yazınız ve hareket sonunda toplam mayın sayısını ekranda 5 sn boyunca gösteren programı yazınız.

   NOT: Robotun 60 sn boyunca hareket etmesini sağlayıp, mayın olarak algılaması için hareket rotasına siyah noktalar koyunuz.

3. Arda bir zarı 6 kez sallamıştır (random fonksiyonunu kullanınız). Bu zar değerlerini bir diziye alıp, en yüksek değeri NXT ekranında 5 sn gösteren programı yazınız.

Yukarıda yer alan problemleri çözdükten sonra eğitsel robotları istediğiniz şekilde programlayabilirsiniz.

PROGRAMLAMA TEMELLERİ DERSİ

NESNE YÖNELİMLİ GÖRSEL PROGRAMLAMA GRUBU

10. HAFTA LAB. UYGULAMALARI

Aşağıdaki problemleri POLYA'nın problem çözme süreci aşamalarını dikkate alarak, algoritma ve akış diyagramlarını oluşturarak ve C# yazılımını kullanarak çözünüz.

1. Rakamlar isimli dizi içerisine bir döngü yardımıyla 0-9 arası rakamları tersten yükleyip, bunları mesaj penceresinde görüntüleyiniz.

2. 5 adet rastgele sayı üretip bunu bir dizi içerisine alınız. Sonra bu dizideki değerleri döngü kullanarak listBox'a ekleyiniz.

3. 0-49 arasında rastgele 6 sayı üreterek bunları bir dizi içerisine alıp, bunları daha sonra listBox'a küçükten büyüğe doğru listeleyiniz.

Yukarıda yer alan problemlerin çözümünü gerçekleştirdikten sonra istediğiniz programlama etkinliklerini yapabilirsiniz.

EK-B: Programlama Başarı Testi Soru Örnekleri

BÖLÜM-1: Temel Programlama Kavramları

3. "**Algoritma**" kavramını açıklayınız.

……

……

……

……

……..……

5. "**Şartlı ya da koşullu dallanma**" kavramını açıklayınız.

……

……

……

……

……..……

BÖLÜM-2: Programlama Becerisi

1.BAŞLA

2.sayac=0

3.sayac<10 İSE TEKRARLA { sayac=sayac+1; 3'e GİT }

4. BİTİR

10. Yukarıdaki algoritma ifadesi için aşağıdakilerden hangisi yanlıştır?

a) TEKRARLA bir döngü işlemidir.

b) sayac=0 Döngünün başlangıç değerini gösterir.

c) sayac=sayac+1 sayacın her defasında kendisinin 2 katından 1 fazla olacak şekilde artacağını gösterir.

d) sayac<10 Döngünün koşulunu gösterir.

e) TEKRARLA içindeki işlemler 10 kez tekrar edilir.

18. Aşağıda verilen ifadelerden hangisi sonsuz bir döngüye girer?

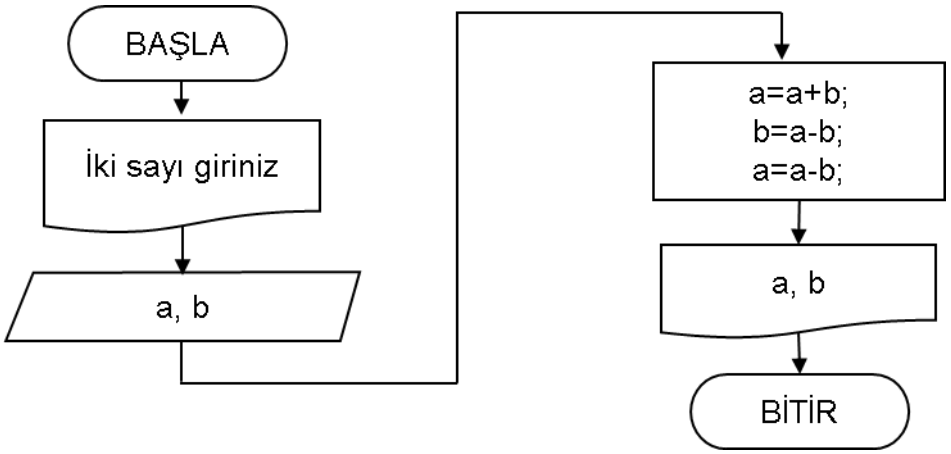
a) ADIM1: sayac=0;
ADIM2: sayac<100 İSE TEKRARLA{sayac=sayac+1; ADIM2'ye GİT)

b) ADIM1: sayac=0;
ADIM2: sayac<100 İSE TEKRARLA{sayac=sayac-1; ADIM2'ye GİT)

c) ADIM1: sayac=100;
ADIM2: sayac>0 İSE TEKRARLA{sayac=sayac-1; ADIM2'ye GİT)

d) ADIM1: sayac=0;
ADIM2: sayac<=100 İSE TEKRARLA{sayac=sayac+5; ADIM2'ye GİT)

e) ADIM1: sayac=10;
ADIM2: sayac<0 İSE TEKRARLA{sayac=sayac-5; ADIM2'ye GİT)



19. Yukarıda akış şeması verilen programa ilişkin olarak aşağıdakilerden hangisi doğrudur?

a) Girilen iki sayının sırasını değiştirip ekrana yazar.

b) Girilen iki sayının önce büyük olanını sonra diğerini yazar.

c) Girilen iki sayının önce küçük olanını sonra diğerini yazar.

d) Girilen iki sayının eşit olup olmadıklarını kontrol eder.

e) Girilen iki sayıdan birincisi küçük ise, ikinciyi ekrana yazar.

24. Aşağıda verilen program çalıştırıldığında ekranda ne yazar?

```
a = 10;
b = 20;
a = a + b * 2 + 5;
YAZ a;
```

a) 35 b) 45 c) 55 d) 65 e) 150

EK-C: Odak Grup Görüşme Soruları

Bilgilendirme Metni

Sevgili öğrenciler,

Sizlerle C# programlama yazılımının Programlama Temelleri dersinde kullanılması ile ilgili bir grup görüşmesi gerçekleştirilecektir. Yaklaşık 20 dakika sürecek bu görüşmede, soruları içtenlikle yanıtlamanız yürüttüğümüz araştırma çalışmasına önemli katkı sağlayacaktır. Bu görüşme ders notlarınızı kesinlikle etkilemeyecektir. Onay vermeniz halinde görüşme, ses veya görüntü kayıt cihazı kullanılarak kaydedilecektir. Görüşme kayıtları analizi esnasında katılımcılara kod isimleri verilecektir. Yani görüşmeye verdiğiniz yanıtlar kişisel bilgilerinizle eşleştirilmeyecektir.

Ayırdığınız zaman ve katkılar için şimdiden teşekkür ederiz.

Doç. Dr. Hakan TÜZÜN (Danışman) Bilim Uzm. Murat ÇINAR

Yarı-Yapılandırılmış Görüşme Soruları

1. Programlama Temelleri dersini C# aracılığıyla işlemenizin hoşunuza giden yönleri nelerdi?
2. Programlama Temelleri dersini C# aracılığıyla işlemenizin hoşunuza gitmeyen yönleri nelerdi?
3. Programlama görevlerini, bireysel ya da takım halinde gerçekleştirmeniz öğrenme sürecinizi ve görev başarınızı nasıl etkiledi?
4. Programlama görevlerini gerçekleştirmeniz esnasında yaşadığınız sorunlar ve bu sorunlarla baş çıkmak için izlediğiniz stratejiler nelerdi?
5. Derste gerçekleştirdiğiniz programlama etkinliklerinin programlama becerinizin haricinde hangi becerilerinizi geliştirdiğini düşünüyorsunuz?
6. Bilgisayar programlarken izlediğiniz yöntemleri diğer derslerinizde nasıl kullanabilirsiniz?
7. C# yazılımı programlama ile ilgili kavramları (değişken kullanımı, aritmetiksel ve mantıksal işlemler, koşul ifadeleri, döngüler, fonksiyonlar, prosedürler, diziler gibi) anlamanızda size hangi açılardan yardımcı oldu?
8. Programlama Temelleri dersini C# programlama aracı aracılığıyla işlemenizin ders başarınızı nasıl etkilediğini düşünüyorsunuz?

Bilgilendirme Metni

Sevgili öğrenciler,

Sizlerle eğitsel robotların Programlama Temelleri dersinde kullanılması ile ilgili bir grup görüşmesi gerçekleştirilecektir. Yaklaşık 20 dakika sürecek bu görüşmede, soruları içtenlikle yanıtlamanız yürüttüğümüz araştırma çalışmasına önemli katkı sağlayacaktır. Bu görüşme ders notlarınızı kesinlikle etkilemeyecektir. Görüşme kayıtları analizi esnasında katılımcılara kod isimleri verilecektir. Yani görüşmeye verdiğiniz yanıtlar kişisel bilgilerinizle eşleştirilmeyecektir. Onay vermeniz halinde görüşme ses veya görüntü kayıt cihazı kullanılarak kaydedilecektir.

Ayırdığınız zaman ve katkılar için şimdiden teşekkür ederiz.

Doç. Dr. Hakan TÜZÜN (Danışman) Bilim Uzm. Murat ÇINAR

Yarı-Yapılandırılmış Görüşme Soruları

1. Programlama Temelleri dersini Eğitsel Robotlar aracılığıyla işlemenizin hoşunuza giden yönleri nelerdi?
2. Programlama Temelleri dersini Eğitsel Robotlar aracılığıyla işlemenizin hoşunuza gitmeyen yönleri nelerdi?
3. Robot programlama görevlerini grup çalışmasıyla gerçekleştirmeniz öğrenme sürecinizi ve görev başarınızı nasıl etkiledi?
4. Eğitsel robot programlama görevlerini gerçekleştirmeniz esnasında yaşadığınız sorunlar ve bu sorunlarla başa çıkmak için izlediğiniz stratejiler nelerdi?
5. Eğitsel robot etkinliklerinin programlama becerinizin haricinde hangi becerilerinizi geliştirdiğini düşünüyorsunuz?
6. Eğitsel robot programlarken izlediğiniz yöntemleri diğer derslerinizde nasıl kullanabilirsiniz?
7. Eğitsel robot etkinlikleri programlama ile ilgili kavramları (değişken kullanımı, aritmetiksel ve mantıksal işlemler, koşul ifadeleri, döngüler, fonksiyonlar, prosedürler, diziler gibi) anlamanızda size hangi açılardan yardımcı oldu?
8. Programlama Temelleri dersini eğitsel robotlar aracılığıyla işlemenizin ders başarınızı nasıl etkilediğini düşünüyorsunuz?

EK-Ç: Ders Günlüğü Örnek Formu

Sevgili Öğrenciler,

Bu ders günlüğü bugün Programlama Temelleri dersinde gerçekleştirdiğiniz programlama etkinlikleri hakkındaki duygu ve düşüncelerinizi almak için hazırlanmıştır. Ders günlükleri için aşağıda rehber sorular başlığında yazan sorulardan yararlanabilirsiniz. Günlük yazarken kâğıdın ön ve arka yüzünü kullanabilir, bu kâğıda farklı bir kâğıt iliştirebilirsiniz. Sizden elde edeceğimiz veriler yalnızca araştırma amacıyla kullanılacak ve başkalarıyla paylaşılmayacaktır. Görüşlerinizi içtenlikle dile getirmenizi rica ediyoruz. Katkılarınız için şimdiden teşekkür ederiz.

Doç. Dr. Hakan TÜZÜN (Danışman) Bilim Uzm. Murat ÇINAR

Rehber Sorular:

1- Bu hafta Programlama Temelleri dersinde neler öğrendin?
2- Bugün derste öğrendiklerin ve yaşadıklarınla ile ilgili duygu ve düşüncelerin nelerdir?
3- Atölye uygulamasında verilen programlama görevlerini gerçekleştirirken yolunda giden şeyler nelerdi?
4- Atölye uygulamasında verilen programlama görevlerini gerçekleştirirken yolunda gitmeyen şeyler nelerdi?

Sevgili Günlük,

……
……
……
……
……
……
……
……
……
……
……
……
……
……
……
……
……
……

EK-D: Nitel Veri Analizi Sonucunda Ortaya Çıkan Serbest Kodlar ve Kodlama Sıklıkları

Sıra No	Kodlar	Kodlanma Sıklığı	Sıra No	Kodlar	Kodlanma Sıklığı
1	Acele etme	10	**57**	Buton	5
2	Açıklama satırı	2	**58**	Büyük konu üzerinde çalışmak	4
3	Akademik beklentiler	39	**59**	Büyük-küçük harf	2
4	Akılda kalma	26	**60**	C#	53
5	Akıllı (etkileşimli) tahta	1	**61**	Can sıkıntısı	19
6	Akış deneyimi	29	**62**	Cazibe	27
7	Akış şeması	20	**63**	Çabaların boşa çıkması	9
8	Akran desteği	65	**64**	Çalışma kâğıtları	6
9	Akran düşüncesini dikkate alma	3	**65**	Çalışma masası sayısı	2
10	Akran rehberliği	6	**66**	Çekinme (Dışa kapalı olma)	2
11	Akrandan öğrenme	9	**67**	Değer vermek	3
12	Alan ve çevre hesaplama	10	**68**	Değerlendirme (öğrenci)	6
13	Algılama kapasitesi (becerisi)	7	**69**	Değişken hatası	4
14	Algoritmayı izlemek (uygulamak)	7	**70**	Değişkenler	35
15	Algoritmik düşünme	104	**71**	Değişkenlere değer atama	5
16	Alt programlar (fonksiyon ve prosedürler)	27	**72**	Deneme yanılma	14
17	Amaç (hedef) belirleme	12	**73**	Deneyim edinme	12
18	Analiz etme becerisi	3	**74**	Deneyimlerin transfer edilmesi	22
19	Analojilerden yararlanma	1	**75**	Denklem	8
20	Anlaşılırlık	25	**76**	Ders başarısı	20
21	Anlaşılması güç olmak	6	**77**	Ders günlüğü	5
22	Anlaşılmasını kolaylaştırmak	9	**78**	Ders saati	4
23	Araç kutusu	3	**79**	Ders saatinin artması	2
24	Araştırma yapmak	16	**80**	Derse hazırlık	1
25	Aritmetiksel işlemler	41	**81**	Derse karşı ilgi gösterme	10
26	Arkadaşının eksiğini tamamlamak	3	**82**	Dersi bırakma (okul değiştirme)	5
27	Asabiyet	18	**83**	Dersi dinleme	34
28	Aşağıdan-yukarı (peyderpey) tasarım	4	**84**	Dersin işlenişi (yapısı)	20
29	Aşinalık	11	**85**	Dersin teorik kısmı	6
30	Aynı açıdan bakmak	1	**86**	Dersin teorik ve uygulaması arasındaki denge	3
31	Ayrıcalık hissi	1	**87**	Dersin teorik ve uygulaması aynı oturumda olması	5
32	Ayrık tamamlanan işlerin birleştirilmesi	1	**88**	Dersin uygulama kısmı	14
33	Ayrıntılara dikkat etme	12	**89**	Devamsızlık	14
34	Basitlik	14	**90**	Diğer bilişim dersleri	3
35	Başarısızlık hissi (korkusu)	18	**91**	Diğer derslerin programlamaya etkisi	4
36	Başarma hissi	20	**92**	Diğer takımlardan yardım alma	4
37	Başarma inancı	13	**93**	Diğer takımları gözlemleme	10
38	Başka (yeni) konuya geçme	31	**94**	Dikkat becerisi	22
39	Başkalarından takdir görmek	4	**95**	Dikkatin dağılması	9
40	Beyin fırtınası	1	**96**	Dikkatini toplama (odaklanma)	26
41	Bildiğini kanıtlamak	2	**97**	Diziler	35
42	Bildiklerini uygulama	15	**98**	Doğrudan çözüme yönelme	11
43	Bilgileri tazelemek	1	**99**	Dokunma sensörü komutları	1
44	Bilgilerin bir yerde buluşması	16	**100**	Döngüler	88
45	Bilgisayar başında ortak çalışma	1	**101**	Dönme hareketi	26
46	Bilgisayar okuryazarlığı	2	**102**	Eğitsel robot	71
47	Bilgisayar sayısı	1	**103**	Eğlenerek öğrenme	4
48	Bilgisayara karşı tutum	1	**104**	Ekran komutları	26
49	Bilgisayarda problem çözme	43	**105**	El becerisi	4
50	Bilgisayarın hata vermesi (bozulması)	3	**106**	Empati kurma (empatik düşünme)	1
51	Bilgisayarın yönlendirmesi	1	**107**	Ergonomik sıkıntılar	5
52	Bilişsel yükün fazla olması	1	**108**	Eşli programlama	25
53	Bireysel çalışma	49	**109**	Etkinlikler	7
54	Bireysel çalışma becerisi	3	**110**	Evde bilgisayarın olma(ma)sı	6
55	Boşluk bırakma	1	**111**	Faktöriyel kavramı	7
56	BT kullanımı becerisi	5	**112**	Farklı çözüm yolları	14

Sıra No	Kodlar	Kodlanma Sıklığı	Sıra No	Kodlar	Kodlanma Sıklığı
113	Farklı fikirlerin karmaşıklığa yol açması	1	**172**	İngilizce dersi	14
114	Farklı fikirlerle başa çıkma	6	**173**	İpuçları	5
115	Fikir alış-verişi	12	**174**	İş ortamı	3
116	Fikir çatışması (uyumsuzluğu)	4	**175**	İş ortamında istenen yeterlilikler	1
117	Fikir mülkiyeti	2	**176**	İşaret hatası	2
118	Fikir üretme	5	**177**	İşbirlikli çalışma	51
119	Fikirlerini rahatça ifade etme	3	**178**	İşbirlikli programlama	3
120	Fizik	3	**179**	İşlem basamaklarını belirlemek	3
121	Fiziksel koşullar	9	**180**	İşlem önceliği	10
122	Flowchart	2	**181**	İşlem yapma	13
123	Form tasarımı	8	**182**	Kâğıt üzerinde çözme	10
124	Gelişmiş (ayrıntılı) olma	3	**183**	Karekök alma	2
125	Gerçek hayata transfer etme	20	**184**	Karmaşıklık hissi	22
126	Geride kalmama	10	**185**	Katılım (uğraşma)	28
127	Görev paylaşımı	16	**186**	Kendi fikirlerini uygulama	2
128	Görev performansı	156	**187**	Kendi programlarını yazmak	5
129	Görev puanı	1	**188**	Kendini takdir etmek	2
130	Göreve ait işlerin ayrık yürütülmesi	4	**189**	Kılavuzdan yararlanmak	10
131	Görevin bir kişinin öncülüğünde ilerlemesi	4	**190**	Kimya	1
132	Görevler	209	**191**	Kişisel sıkıntılar (rahatsızlıklar)	30
133	Görevleri aynı kişilerin yapması	2	**192**	Klavye kullanma becerisi	11
134	Görevleri başaramama	225	**193**	Klavye tuşlarının işlevini öğrenme	5
135	Görevlerin ardışık (diğer takım üyelerinden bağımsız) yürümesi	4	**194**	Kod (sözdizimi) hataları	53
136	Görevlerin yetişmemesi	14	**195**	Kod editörü	2
137	Görsellik	5	**196**	Kod tamamlama özelliği	4
138	Görünüş	1	**197**	Kod yazmak	223
139	Gözünde büyütme (önyargı)	14	**198**	Kod yoğunluğunu azaltmak	6
140	Hâkimiyet	3	**199**	Koda dökememek	26
141	Hareket uzunluğunu hesaplama	17	**200**	Kodlamanın sonuçlarını görme	23
142	Harf hatası (yazım yanlışı)	31	**201**	Kodlara aşina olmak	14
143	Hata bulma	75	**202**	Kodları canlandırma	1
144	Hata bulma becerisi	11	**203**	Kodları ezberlemek	25
145	Hata düzeltme (debugging)	117	**204**	Kodları tekrar yazmak	16
146	Hata düzeltme araçları	1	**205**	Kodları unutmak	2
147	Hata mesajlarını anlamak	25	**206**	Kodların görsel olması	8
148	Hata penceresi	20	**207**	Kodların sorunsuz çalışması	31
149	Hata penceresindeki uyarılar	19	**208**	Kodların uzun olması	5
150	Hata üzerinde düşünme	16	**209**	Komutlar (kodlar)	43
151	Hata yapma	170	**210**	Konu (ders) tekrarı	23
152	Hata yapma korkusu	4	**211**	Konuyu yeniden gözden geçirme	2
153	Hatalardan öğrenme	6	**212**	Koordinat sistemi	2
154	Hataları giderme çabasının olmaması	3	**213**	Kullanım kolaylığı	75
155	Hataları görememek	11	**214**	Küçüktür büyüktür işareti	2
156	Hataların artması	10	**215**	Lego	14
157	Hataların azalması	10	**216**	Liste kutusu	8
158	Hataların giderilememesi	8	**217**	Mantık hatası	18
159	Hataların otomatik giderilmesi	2	**218**	Mantık yürütme (akıl yürütme)	41
160	Hatalarını görmek	36	**219**	Mantıksal düşünme	13
161	Hatalarla başa çıkma becerisi	8	**220**	Matematik	80
162	Hataya açıklık	2	**221**	Matematiksel düşünme	9
163	Hatayı anında fark etmek	4	**222**	Materyal (video, film vb.)	9
164	Hayal gücünü kullanma	1	**223**	Memnuniyet (hoşnutluk)	48
165	Hazır fonksiyonlar	29	**224**	Merak uyandırma	11
166	Heyecanlanmak	16	**225**	Mesaj penceresi	5
167	Hızlı çalışma	1	**226**	Mesleki beklentiler	15
168	Hızlı düşünme ve karar alma	7	**227**	Mesleki kariyer	9
169	Işık komutları	8	**228**	Metin kutusu	17
170	İlerleme (gelişim)	13	**229**	Mod alma	3
171	İlişki kurma (neden-sonuç ilişkisi)	3	**230**	Mola vermek	6

Sıra No	Kodlar	Kodlanma Sıklığı	Sıra No	Kodlar	Kodlanma Sıklığı
231	Motivasyon	9	**284**	Problemi anlamak	136
232	Motor gücünü ayarlama	11	**285**	Problemi tekrar tekrar okuma	1
233	Mouse (fare) kullanımı	1	**286**	Problemin karmaşık olması	11
234	Müşteri beklentileri	2	**287**	Problemin net (açık) olmaması	10
235	Noktalama hataları	24	**288**	Problemin uzun olması	4
236	Noktalama işaretlerinin yerleri	4	**289**	Programdan kodları anlamak (yukarıdan-aşağıya yaklaşım)	1
237	Not almak (deftere yazmak)	52	**290**	Programın hataları göstermesi	2
238	Notlara bakma	49	**291**	Programın uzun olması (kod yoğunluğunun fazla olması)	6
239	Nöbetçi olma	2	**292**	Programlama etkinlikleri	90
240	NXT yazılımı	36	**293**	Programlama kavramları	5
241	Olasılıklı düşünme	4	**294**	Programlamanın diğer derslere etkisi	12
242	Operatörler	42	**295**	Programlamaya karşı sevgi	12
243	Ortak amaç	1	**296**	Proje çalışması	4
244	Oturma düzeni	5	**297**	Rakam hatası (değer girişi hatası)	1
245	Ödev	1	**298**	Rekabet (meydan okuma)	14
246	Öğrenci özerkliği (kendisinin yapması)	23	**299**	Resmi tatil (ara verme)	1
247	Öğrenci sayısı	7	**300**	Robot C	77
248	Öğrenci yansımaları	1	**301**	Robot inşa işinin tamamlanamaması	3
249	Öğrencinin aktif olması	2	**302**	Robot montajında hata yapmak	6
250	Öğrencinin desteğe ihtiyaç duyması	11	**303**	Robot parçalarını aramak	2
251	Öğrenileni geri getirme	1	**304**	Robot parçalarını karıştırmak	3
252	Öğrenme (konuyu anlamak)	180	**305**	Robot parçalarını nereye takacağını bilememek	1
253	Öğrenme desteği	4	**306**	Robot parçalarının çok küçük olması	3
254	Öğrenmeye çabalamak	33	**307**	Robot parçalarının farklı renkte olması	3
255	Öğretmen desteği	164	**308**	Robot parçalarının işlevini anlamak	3
256	Öğretmen odaklı olma	12	**309**	Robot parçalarının kaybolması	2
257	Öğretmenin konuyu hızlı anlatması	3	**310**	Robot parçalarının sayısının fazla olması	2
258	Önbilgi eksikliği (ön yeterlilik)	14	**311**	Robot sayısı	1
259	Önceki konuların (kavramların) yeni görevleri olumsuz etkilemesi	3	**312**	Robot tasarımı (yapımı)	34
260	Önceki öğrenmelerle bağlantı kurma	11	**313**	Robot üzerinde denemek	28
261	Önceki örneklerden yararlanma	16	**314**	Robot üzerinden programlamak	2
262	Öz-beklentilerin karşılanmaması	3	**315**	Robota bakış açısının değişmesi	1
263	Öz-düzenleyici düşünme	28	**316**	Robota program yüklemek	17
264	Özel karakterler	2	**317**	Robotla iletişim kurma	2
265	Özgüven	40	**318**	Robotu çalıştırma	12
266	Parantez hatası	20	**319**	Robotu hatasız inşa etmek	2
267	Parantezin aşağıya kayması	1	**320**	Robotu kurma	6
268	Parçaları birleştirme	11	**321**	Robotu programlamak	54
269	Parçaları incelemek	2	**322**	Robotu yeni baştan tasarlamak	2
270	Pes etmek	4	**323**	Robotun arıza vermesi	2
271	Planlama	11	**324**	Robotun hareket etmesi	32
272	Polya'nın problem çözme adımları	103	**325**	Robotun hareketlerini gözlemleme	15
273	Problem çözme adımlarını atlamak	1	**326**	Robotun hareketlerini kontrol etmek	56
274	Problem çözme aşamaları	122	**327**	Robotu test etmek	1
275	Problem çözme becerisi	12	**328**	Sabitler	2
276	Problem çözme sürecini tamamlayamamak	3	**329**	Sağlarlık	2
277	Problem çözme sürecinin sistematikleştirilmesi	24	**330**	Sanal robot	2
278	Problem çözme süresini azaltmak	5	**331**	Sayaç	4
279	Problem çözmek için harcanan çabanın artması	25	**332**	Sayı doğrusu (negatif-pozitif olma)	3
280	Problem karşısında yapılacak ilk iş	8	**333**	Sayı sistemleri	1
281	Problem türü	4	**334**	Sayısal	1
282	Probleme takılıp kalmak	28	**335**	Sayısal beceriler	3
283	Problemi analiz etmek (girdi-çıktı)	15	**336**	Sensörler	65

Sıra No	Kodlar	Kodlanma Sıklığı	Sıra No	Kodlar	Kodlanma Sıklığı
337	Ses komutları	17	**394**	Takım üyelerinin akademik başarılarının birbirine yakın olması	1
338	Sıkıcı olma	62	**395**	Takım üyelerinin aynı puanı alması	4
339	Sınav heyecanı (kaygısı)	2	**396**	Takım üyelerinin farklı alanlarda yeteneklere sahip olması	2
340	Sınavda yapamama	9	**397**	Takım üyelerinin ilgi göstermemesi	12
341	Sınıf mantığı	4	**398**	Takım üyelerinin sayısı	9
342	Sonuç (çözüm) üretme	43	**399**	Takım yönetimi	3
343	Sonuç odaklı değerlendirme	2	**400**	Takıma katkı sağlama	16
344	Sonuçları görme ve geriye bakma	21	**401**	Takıma katkının yeterli olmaması	38
345	Sorulara yoğunlaşma	16	**402**	Takıma yol gösterme	2
346	Soruların giderek zorlaşması	16	**403**	Tartışma (takım içi tartışma)	43
347	Sorumluluktan kaçma	3	**404**	Tasarım yapmak	8
348	Sorunlara daha geniş perspektiften bakmak	2	**405**	Tek başına programlama	26
349	Sorunlarla başa çıkma becerisi	13	**406**	Teknik aksaklık	7
350	Soyutu somut yapma	5	**407**	Teknik destek	3
351	Söküp birleştirmek	17	**408**	Teknik sorunlarla başa çıkma	1
352	Sözdizimi	42	**409**	Teknolojiye ayak uydurma	1
353	Sözel	1	**410**	Tekrar hata yapma	5
354	Sözel grubu dersler	2	**411**	Temizlik	1
355	Strateji belirleme (geliştirme)	38	**412**	Test parkuru	1
356	Stratejiyi uygulama	17	**413**	Tip dönüşümleri	8
357	Stres olma	6	**414**	Toplum	1
358	Sürdürme isteği (bağlanma)	35	**415**	Tuğla (Brick)	2
359	Süreç değerlendirme	3	**416**	Tutum geliştirme	16
360	Sürükle bırak (blok temelli olma)	2	**417**	Türkçe karakter	2
361	Süslü parantez hatası	25	**418**	Uygulama (ürün) geliştirme	11
362	Şart (koşul) ifadeleri	77	**419**	Üslü işlemler	19
363	Tabure	3	**420**	Veri tipi (türü)	46
364	Tahtadakini uygulama	28	**421**	Yabancı dilde olmak	42
365	Takım arkadaşına destek olma	27	**422**	Yanlışlıkla silme	2
366	Takım arkadaşına kızma	2	**423**	Yapa-boza öğrenme	13
367	Takım arkadaşından beklentiler	15	**424**	Yaptıklarından başkalarının faydalanması	1
368	Takım arkadaşını değiştirme	1	**425**	Yarar algısı	37
369	Takım arkadaşını dikkate alma	8	**426**	Yaşama dönüklük	50
370	Takım arkadaşını uyarma	3	**427**	Yazılım	3
371	Takım arkadaşının daha az bilmesi	9	**428**	Yeni bir şey ortaya çıkarma	7
372	Takım arkadaşının daha iyi bilmesi	12	**429**	Yenilik etkisi	7
373	Takım arkadaşının eksiğini tamamlama	5	**430**	Yeteneklerinin farkına varma	1
374	Takım arkadaşının hatasını düzeltme	12	**431**	Yeterli tekrarın olmaması (Konuları tekrar etmeme)	50
375	Takım arkadaşlarının çaba göstermesi	5	**432**	Yetersizlik hissi	6
376	Takım çalışması	176	**433**	Yol gösterme	1
377	Takım değiştirme	7	**434**	Yön belirleme	2
378	Takım halinde hareket etme	4	**435**	Yüzde kavramı	1
379	Takım içi aylaklık	11	**436**	Zaman kaybı	25
380	Takım içi baskı	1	**437**	Zaman sıkıntısı (zamanın yetmemesi)	42
381	Takım içi çatışma	5	**438**	Zaman tasarrufu	25
382	Takım içi iletişim sorunu	4	**439**	Zamanı etkili kullanma	7
383	Takım içi rekabet	1	**440**	Zamanlama komutları	17
384	Takım içi sorunlar	27	**441**	Zayıf kalma	1
385	Takım içi sorunlarla başa çıkma	4	**442**	Zevkli (eğlenceli) olmak	172
386	Takım içi yardımlaşma	20	**443**	Zihinde canlandırma	6
387	Takım içinde sürekli aynı işi yapma	1	**444**	Zihinsel çaba	12
388	Takım kararı	5	**445**	Zihinsel yönlendirme (manipülasyon)	2
389	Takım kararlarının bireysel kararları kısıtlaması	1	**446**	Zorluk düzeyi	87
390	Takım oluşturma	13	**447**	Zorun kolay olması (basitleştirmek)	27
391	Takım sayısı	2	**448**	3B baskı	1
392	Takım uyumu	38	**449**	3B tasarım	1
393	Takım üyelerine yük binmesi	24	**450**	3B üretim	1

EK-E: Nitel Veri Analizi Sonucunda Ortaya Çıkan Temalar (🏛), Kategoriler (☂) ve Kodlar (🔑)

[Düşünme Biçimleri ve Bilişsel Unsurlar] 🏛
- [Düşünme Biçimleri] ☂
  - Algoritmik Düşünme 🔑
    - Analiz Etme becerisi 🔑
    - İşlem Basamaklarını Belirlemek 🔑
    - Algoritmayı İzlemek (Uygulamak) 🔑
  - Matematiksel Düşünme 🔑
  - Mantıksal Düşünme 🔑
    - Mantık Yürütme (Akıl Yürütme) 🔑
      - İlişki Kurma (Neden-Sonuç İlişkisi) 🔑
  - Olasılıklı Düşünme 🔑
  - Öz-düzenleyici Düşünme 🔑
  - [Soyutlama] ☂
    - Ayrıntılara Dikkat Etme 🔑
    - Karmaşıklık Hissi 🔑
    - Büyük Konu Üzerinde Çalışmak 🔑
    - Anlaşılması Güç Olmak 🔑
    - Zorun Kolay Olması (Basitleştirmek) 🔑
    - Soyutu Somut Yapma 🔑
    - Analojilerden Yararlanma 🔑
- [Bilişsel Süreçler] ☂
  - Algılama Becerisi 🔑
    - Bilişsel Yükün Fazla Olması 🔑
  - Zihinsel (Bilişsel) Çaba 🔑
  - Dikkat Becerisi 🔑
    - Dikkatini Toplama (Odaklanma) 🔑
    - Dikkatin Dağılması 🔑
  - Hayal Gücünü Kullanma 🔑
  - Zihinde Canlandırma 🔑
  - Zihinsel Yönlendirme (Manipülasyon) 🔑

[Dersin Genel Çerçevesi ve Pedagojik Unsurlar] 🏛
- [Öğrenme Süreçleri] ☂
  - Akılda Kalma (Kalıcılık) 🔑
    - Öğrenileni Geri Getirme 🔑
    - Yeterli Tekrarın Olmaması 🔑
  - Derse Hazırlık 🔑
  - Önceki Konuların (Kavramların) Yeni Görevleri Olumsuz Etkilemesi 🔑
    - Önceki Öğrenmelerle Bağlantı Kurma 🔑
  - Not Alma (Deftere Yazmak) 🔑
  - Ders Günlüğü (Yansıma) 🔑
  - Öğrenme (Konuyu Anlamak) 🔑
    - Öğrenme Desteği 🔑
      - Materyal (Video, Film) 🔑
    - Öğrenmeye Çabalamak 🔑
    - Eğlenerek Öğrenme 🔑
    - Yapa-boza Öğrenme 🔑
      - Aşağıdan Yukarı (Peyderpey) Tasarım 🔑
      - Söküp Birleştirmek 🔑
      - Parçaları İncelemek 🔑
      - Parçaları Birleştirme 🔑
  - Programdan Kodları Anlamak (Yukarıdan Aşağıya Yaklaşım) 🔑

- Öğretmen Odaklı Olma
- [Öğretimsel Çerçeve]
  - Ders Başarısı
  - Dersi Bırakma
  - Derse Karşı İlgi Gösterme
  - Dersi Dinleme
  - Dersin İşlenişi (Yapısı)
    - Dersin Teorik Kısmı
    - Dersin Teorik ve Uygulaması Arasındaki Denge
    - Dersin Teorik ve Uygulaması Aynı Oturumda Olması
    - Dersin Uygulama Kısmı
    - Mola Vermek
    - Etkinlikler
    - Ödev
    - Çalışma Kâğıtları
    - Ders Saati
      - Ders Saatinin Artması
  - Konu (Ders) Tekrarı
    - Bilgileri Tazelemek
  - Başka Konuya Geçme
    - Öğretmenin Konuyu Hızlı Anlatması
  - Fiziksel Koşullar
    - Bilgisayar Sayısı
    - Çalışma Masası Sayısı
    - Oturma Düzeni
    - Öğrenci Sayısı
    - Temizlik
  - Değerlendirme (Öğrencinin Değerlendirilmesi)
    - Sınavda Yapamama
    - Sınav Heyecanı (Kaygısı)
    - Sonuç Odaklı Değerlendirme
    - Süreç Değerlendirme
      - Proje Çalışması

[Problem Çözme ve Bilgisayımsal Düşünme Etkinlikleri]

- [Problem Çözme]
  - Polya'nın Problem Çözme Adımları
    - Problem Çözme Sürecinin Sistematikleştirilmesi
    - Anlaşılmasını Kolaylaştırmak
    - Problem Çözme Adımlarını Atlamak
      - Doğrudan Çözüme Yönelme
  - Problem Çözme Aşamaları
    - Problemi Anlamak
      - Problemin Net Olmaması
      - Problemin Uzun Olması
      - Problemin Karmaşık Olması
      - Problemi Analiz Etmek (Girdi-Çıktı)
      - Problem Karşısında Yapılacak İlk İş
      - Sorulara Yoğunlaşma
        - Problemi Tekrar Tekrar Okumak
    - Strateji Belirleme
      - Planlama
      - Araştırma Yapmak
      - Notlara Bakma
        - Önceki Örneklerden Yararlanma
        - Konuyu Yeniden Gözden Geçirme

- Farklı Çözüm Yolları
  - Aynı Açıdan Bakmak
  - Fikir Üretme
    - Beyin Fırtınası
  - Sorulara Daha Geniş Perspektiften Bakmak
- Stratejiyi Uygulama
  - Kâğıt Üzerinde Çözme
  - Koda Dökememek
- Sonuçları Görme ve Geriye Bakma
  - Deneme-yanılma
  - Deneyim Edinme
  - Sonuç Üretme
- Yaşama Dönüklük
  - Sorunlarla Başa Çıkma Becerisi
  - Deneyimlerin Transfer Edilmesi
    - Gerçek Hayata Transfer Etme
- Bilgisayarda Problem Çözme
  - Problem Çözme Süresini Azaltmak
  - Bildiklerini Uygulama
- Zorluk Düzeyi
  - Problem Çözmek İçin Harcanan Çabanın Artması
    - Çabanın Boşa Çıkması
  - Probleme Takılıp Kalmak
  - Soruların Giderek Zorlaşması
- Problem Çözme Becerisi
  - Hızlı Düşünme ve Karar Verme
- Problem Çözme Sürecini Tamamlayamamak
  - İpuçları
- Problem Türü
  - Sayısal
  - Sözel

Programlama Kavramları

- Akış Şeması
- Aritmetiksel İşlemler
- Sayı Sistemleri
- Değişkenler
  - Tip Dönüşümleri
  - Veri Tipi
  - Değişkenlere Değer Atama
- Sabitler
- Döngüler
  - Sayaç
- Operatörler
- Şart İfadeleri
- Hazır Fonksiyonlar
- Diziler
  - Sınıf Mantığı
- Alt Programlar (Fonksiyon ve Prosedürler)

Görevler

- Uygulama (Ürün) Geliştirme
- Görev Puanı
- Görevleri Başaramama
  - Tahtadakini Uygulama
- Görevlerin Yetişmemesi
- Görev Performansı

Programlama Etkinlikleri

- Programın Uzun Olması (Kod Yoğunluğunun Fazla Olması)
  - Kod Yoğunluğunu Azaltmak (Programı Kısaltmak)
- Kod Yazmak
  - Komutlar (Kodlar)
  - Sözdizimi
  - Açıklama (Yorum) Satırı
  - Kod Editörü
    - Kod Tamamlama Özelliği
  - Mouse (Fare) Kullanımı
- Bildiğini Kanıtlamak
- Kodlamanın Sonuçlarını Görme (Kodların Derlenmesi)
- Kodlara Aşina Olmak
- Kodların Sorunsuz Çalışması
- Kodları Unutmak
  - Kodları Ezberlemek
- [Programlama Tipi]
  - Tek Başına Programlama
  - Eşli Programlama
  - İşbirlikli Programlama

[Hatalar]

- Hata Üzerinde Düşünme
- Hataları Giderme Çabasının Olmaması
- Hataları Görememek
- Hatalarla Başa Çıkma Becerisi
- Hataya Açıklık
- Programın Hataları Göstermesi
  - Hata Penceresi
    - Hata Penceresindeki Uyarılar
    - Hatalarını Görmek
    - Hatayı Anında Fark Etmek
    - Hata Mesajlarını Anlamak
- Hata Bulma
  - Hata Bulma Becerisi
- Hata Düzeltme (Debugging)
  - Tekrar Hata Yapma
  - Hata Düzeltme Araçları
  - Kodları Tekrar Yazmak
- Hata Yapma
  - Kod (Sözdizimi) Hataları
    - Harf Hatası (Yazım Yanlışı)
      - Türkçe Karakter
      - Büyük/Küçük Harf
    - Parantez Hatası
      - Parantezin Aşağıya Kayması
      - Süslü Parantez Hatası
    - Boşluk Bırakma
    - Değişken Hatası
    - Noktalama Hataları
      - Noktalama İşaretlerinin Yerleri
    - Rakam Hatası (Değer Girişi Hatası)
    - İşaret Hatası
      - Küçüktür Büyüktür İşareti
  - Mantık Hatası
- Hata Yapma Korkusu
- Hatalardan Öğrenme
- Hataların Artması

- Yanlışlıkla Silme
- Kodların Uzun Olması
- Hataların Azalması
  - Hataların Otomatik Giderilmesi
- Hataların Giderilememesi

[Zaman]

- Zaman Almak (Kaybı)
- Zaman Sıkıntısı (Zaman Baskısı)
  - Acele Etme
- Zaman Tasarrufu (Kazanma)
- Zamanı Etkili Kullanma

[Araçlar]

- Eğitsel Robot
  - El Becerisi
  - Robot C
    - Hızlı Çalışma
    - Gelişmiş (Ayrıntılı) Olma
  - NXT Yazılımı
    - Kodların Görsel Olması
    - Sağlarlık
    - Sürükle-Bırak (Blok-temelli)
    - Zayıf Kalma
  - Lego
  - Ayrıcalık Hissi
  - Robot Tasarımı (Yapımı)
    - Kılavuzdan Yararlanmak
    - Robot Parçalarını Aramak
      - Robot Parçalarının Farklı Renkte Olması
      - Robot Parçalarının Kaybolması
        - Robot Parçalarının Çok Küçük Olması
    - Robotu Hatasız İnşa Etmek
    - Robotu İnşa İşinin Tamamlanamaması
      - Robot Montajında Hata Yapmak
        - Robotu Yeni Baştan Tasarlamak
        - Robot Parçalarının Nereye Takılacağını Bilememek
      - Robot Parçalarını Karıştırmak
        - Robot Parçalarının Sayısının Fazla Olması
    - Robot Parçalarının İşlevini Anlamak
    - Robotu Test Etmek
  - Robota Bakış Açısının Değişmesi
  - Robotu Çalıştırma
    - Test Parkuru
  - Robotu Kurma
  - Robotun Arıza Vermesi
  - Robotun Hareket Etmesi
    - Robotun Hareketlerini Gözlemleme
  - Robotla İletişim Kurma
  - Robotu Programlamak
    - Robota Program Yüklemek
    - Robot Üzerinden Programlamak
    - Robotun Hareketlerini Kontrol Etmek
      - Zamanlama Komutları
      - Dönme Hareketi
      - Hareket Uzunluğunu Hesaplama
      - Yön Belirleme

- Motor Gücünü Ayarlama
- Ses Komutları
- Ekran Komutları
- Dokunma Sensörü Komutları
- Işık Komutları
- Robot Üzerinde Denemek
  - Kodları Canlandırma (Somutlaştırma)
- Robot Sayısı
- Sanal Robot
- Sensörler
- Tuğla (Brick)
- C#
  - Form Tasarımı
  - Mesaj Penceresi
  - Araç Kutusu
  - Metin Kutusu
  - Liste Kutusu
  - Buton
- Flowchart
- Yazılım
- Akıllı (Etkileşimli) Tahta

Öğrenci Çalışmaları

- İşbirlikli Çalışma
  - Bilgisayar Başında Ortak Çalışma
  - Ortak Amaç
  - Tartışma (Takım)
  - Fikir Alış-verişi
    - Fikirlerini Rahatça İfade Etme
      - Akran Düşüncesini Dikkate Alma
    - Bilgilerin Bir Yerde Buluşması
    - Fikir Çatışması (Uyumsuzluğu)
      - Farklı Fikirlerle Başa Çıkma
      - Farklı Fikirlerin Karmaşıklığa Yol Açması
      - Fikir Mülkiyeti
  - Takım Çalışması
    - Takım Değiştirme
      - Takım Arkadaşını Değiştirme
    - Takım İçi Sorunlar
      - Takım İçi İletişim Sorunu
      - Takım Üyelerine Yük Binmesi
        - Sorumluluktan Kaçma
        - Takım İçi Baskı
        - Görevleri Aynı Kişilerin Yapması
      - Takım İçi Sorunlarla Başa Çıkma
        - Takım Arkadaşını Uyarma
      - Takım İçinde Sürekli Aynı İşi Yapma
      - Takıma Katkının Yeterli Olmaması
        - Takım İçinde Aylaklık
        - Takım Üyelerinin İlgi Göstermemesi
        - Takım Üyelerinin Aynı Puanı Alması
      - Takım Kararlarının Bireysel Kararları Kısıtlaması
      - Takım İçi Çatışma
        - Takım Arkadaşına Kızma
    - Takım Oluşturma
      - Takım Üyelerinin Farklı Alanlarda Yeteneklere Sahip Olması

- Takım Üyelerinin Akademik Başarılarının Birbirine Yakın Olması
- Takım Uyumu
  - Takım İçi Yardımlaşma
    - Takım Arkadaşına Destek Olma
    - Takım Arkadaşının Daha Az Bilmesi
    - Takım Arkadaşının Daha İyi Bilmesi
    - Takım Arkadaşının Eksiğini Tamamlama
    - Takım Arkadaşının Hatasını Düzeltme
  - Takım Arkadaşından Beklentiler
    - Takım Arkadaşlarının Çaba Göstermesi
    - Takıma Katkı Sağlama
    - Takıma Yol Gösterme
  - Takım Üyelerinin Sayısı
  - Takım İçi Rekabet
  - Takım Yönetimi
    - Takım Kararları
      - Takım Halinde Hareket Etme
    - Takım Arkadaşını Dikkate Alma
  - [Diğer Takımlar]
    - Diğer Takımları Gözlemleme
    - Diğer Takımlardan Yardım Alma
  - Takım Sayısı
  - Görev Paylaşımı
    - Göreve Ait İşlerin Ayrık Yürümesi
      - Ayrık Tamamlanan İşlerin Birleştirilmesi
    - Görevlerin Ardışık Yürümesi
    - Görevin Bir Kişinin Öncülüğünde Yürümesi
- Bireysel Çalışma
  - Bireysel Çalışma Becerisi
  - Çekinme (Dışa Kapalı Olma)
  - Evde Bilgisayar Olması
  - Öğrenci Özerkliği
    - Öğrencinin Aktif Olması

[Destekler ve Engeller]

- [Destekler]
  - Akran Desteği
    - Akrandan Öğrenme
    - Akran Rehberliği
    - Arkadaşının Eksiğini Tamamlamak
    - Yol Gösterme
  - Bilgisayarın Yönlendirmesi
  - Öğretmen Desteği
  - Öğrencinin Desteğe İhtiyaç Duyması
  - Teknik Destek
    - Teknik Sorunlarla Başa Çıkma Yöntemi
- [Engeller]
  - Yabancı Dilde Olmak
  - Önbilgi Eksikliği (Ön Yeterlilikler)
  - Bilgisayar Okuryazarlığı
  - Devamsızlık
  - Resmi Tatil (Ara Verme)
  - Öğrencinin Nöbetçi Olması
  - Klavye Kullanma Becerisi
    - Aşinalık
    - Klavye Tuşlarının İşlevini Öğrenme

Özel Karakterler
Teknik Aksaklık
Bilgisayarın Hata Vermesi (Bozulması)
Ergonomik Sıkıntılar
Tabure
Kişisel Sıkıntılar (Rahatsızlıklar)

[Duyuşsal Boyut]
Başarısızlık
Yetersizlik Hissi
Öz-beklentilerin Karşılanamaması
Pes Etmek
Asabiyet
Sıkıcı Olma
Can Sıkıntısı
Gözünde Büyütme
Stres Olma
Başarma Hissi
Özgüven
Başarma İnancı
Yeteneklerinin Farkına Varma
Başkalarından Takdir Görmek
Kendini Takdir Etmek
Motivasyon
Memnuniyet
Yeni Bir Şeyler Ortaya Çıkarma
İlerleme (Gelişim)
Değer Vermek
Yarar Algısı
Cazibe
Yenilik Etkisi
Görünüş
Kullanım Kolaylığı
Basitlik
Anlaşılırlık
Görsellik
Rekabet (Meydan Okuma)
Geride Kalmama
Amaç Belirleme
Merak
Sürdürme İsteği (Bağlanma)
Zevkli (Eğlenceli) Olmak
Heyecanlanmak
Katılım (Uğraşma)
Kendi Fikirlerini Uygulama
Tasarım Yapmak
Kendi Programlarını Yazmak
Hâkimiyet
Akışa Kapılmak
Tutum Geliştirme
Bilgisayara Karşı Tutum
Programlamaya Karşı Sevgi
Empati Kurma (Empatik Düşünme)

[Beklentiler ve Diğer Çıktılar]
[Beklentiler]

- İş Ortamı
  - İş Ortamında İstenen Yeterlilikler
- Mesleki Beklentiler
  - Mesleki Kariyer
  - Teknolojiye Ayak Uydurma
  - BT Kullanımı Becerisi
- Müşteri Beklentileri
  - 3B Tasarım
  - 3B Baskı
  - 3B Üretim
- Toplum
  - Yaptıklarından Başkalarının Faydalanması
- Akademik Beklentiler

[Diğer Çıktılar]

- Diğer Derslerin Programlamaya Etkisi
- Programlamanın Diğer Derslere Etkisi
  - Matematik
    - Denklem
    - İşlem Yapma
    - İşlem Önceliği
    - Üslü İşlemler
    - Karekök Alma
    - Mod Alma
    - Yüzde Kavramı
    - Koordinat Sistemi
    - Alan ve Çevre Hesaplama
    - Sayı Doğrusu (Pozitif-negatif Olma)
    - Faktöriyel Kavramı
  - Fizik
  - Kimya
  - İngilizce Dersi
  - Diğer Bilişim Dersleri
  - Sayısal Beceriler
  - Sözel Grubu Dersler
- Öğrenci Yansımaları

EK-F: Etik Komisyonu Onay Bildirimi

T.C.
HACETTEPE ÜNİVERSİTESİ
Rektörlük

Sayı : 35853172/ 433 - 1945

20 Haziran 2016

EĞİTİM BİLİMLERİ ENSTİTÜ MÜDÜRLÜĞÜNE

İlgi: 06.06.2016 tarih ve 1385 sayılı yazınız.

Enstitünüz Bilgisayar ve Öğretim Teknolojileri Eğitimi Anabilim Dalı doktora programı öğrencilerinden **Murat ÇINAR**'ın **Doç. Dr. Hakan TÜZÜN** danışmanlığında yürüttüğü **"Bilgisayımsal Düşünmenin Bileşenlerinin Belirlenmesi ve K12 Düzeyindeki Bilişim Derslerinde Öğrenenlerin Bilgisayımsal Becerilerinin Geliştirilmesi"** başlıklı tez çalışması, Üniversitemiz Senatosu Etik Komisyonunun **14 Haziran 2016** tarihinde yapmış olduğu toplantıda incelenmiş olup, etik açıdan uygun bulunmuştur.

Bilgilerinizi ve gereğini rica ederim.

Prof. Dr. Rahime M. NOHUTCU
Rektör a.
Rektör Yardımcısı

Hacettepe Üniversitesi Rektörlük 06100 Sıhhiye-Ankara
Telefon: 0 (312) 305 3001 - 3002 • Faks: 0 (312) 311 9992
E-posta: yazimd@hacettepe.edu.tr • www.hacettepe.edu.tr

Ayrıntılı Bilgi için:
Yazı İşleri Müdürlüğü
0 (312) 305 1008

EK-G: Etik Beyanı

Hacettepe Üniversitesi Eğitim Bilimleri Enstitüsü, tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmasında,

- tez içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- görsel, işitsel ve yazılı bütün bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
- başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
- atıfta bulunduğum eserlerin bütününü kaynak olarak gösterdiğimi,
- kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- bu tezin herhangi bir bölümünü bu üniversitede veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

beyan ederim.

17/05/2019



Murat ÇINAR

EK-H: Doktora Tez Çalışması Orijinallik Raporu

26/07/2019

HACETTEPE ÜNİVERSİTESİ

Eğitim Bilimleri Enstitüsü

Bilgisayar ve Öğretim Teknolojileri Eğitimi Ana Bilim Dalı Başkanlığına,

Tez Başlığı: Nesneye Yönelik ve Robot Programlamanın Öğrenci Başarısına, Soyutlamaya, Problem Çözmeye ve Motivasyona Etkilerinin İncelenmesi

Yukarıda başlığı verilen tez çalışmamın tamamı (kapak sayfası, özetler, ana bölümler, kaynakça) aşağıdaki filtreler kullanılarak **Turnitin** adlı intihal programı aracılığı ile kontrol edilmiştir. Kontrol sonucunda aşağıdaki veriler elde edilmiştir:

Rapor Tarihi	Sayfa Sayısı	Karakter Sayısı	Savunma Tarihi	Benzerlik Oranı	Gönderim Numarası
26/07/2019	281	470030	17/05/2019.	%6	1155124626

Uygulanan filtreler:

1. Kaynaklar hariç
2. Alıntılar dâhil
3. 5 kelimeden daha az örtüşme içeren metin kısımları hariç

Hacettepe Üniversitesi Eğitim Bilimleri Enstitüsü Tez Çalışması Orijinallik Raporu Alınması ve Kullanılması Uygulama Esasları'nı inceledim ve çalışmamın herhangi bir intihal içermediğini; aksinin tespit edileceği muhtemel durumda doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi ve yukarıda vermiş olduğum bilgilerin doğru olduğunu beyan eder, gereğini saygılarımla arz ederim.

Ad Soyadı:	Murat ÇINAR
Öğrenci No.:	N12246055
Ana Bilim Dalı:	Bilgisayar ve Öğretim Teknolojileri Eğitimi
Programı:	Bilgisayar ve Öğretim Teknolojileri Eğitimi
Statüsü:	☐ Y.Lisans ☒ Doktora ☐ Bütünleşik Dr.



DANIŞMAN ONAYI



UYGUNDUR.

Prof. Dr. Hakan Tüzün

EK-I: Dissertation Originality Report

26/07/2019

HACETTEPE UNIVERSITY

Graduate School Of Educational Sciences

To The Department Of Computer Education and Instructional Technology

Thesis Title: An Examination Of The Effects Of Object-Oriented and Robotics Programming on Student Achievement, Abstraction, Problem Solving and Motivation

The whole thesis that includes the *title page, introduction, main chapters, conclusions and bibliography section* is checked by using **Turnitin** plagiarism detection software take into the consideration requested filtering options. According to the originality report obtained data are as below.

Time Submitted	Page Count	Character Count	Date of Thesis Defense	Similarity Index	Submission ID
26/07/2019	281	470030	17/05/2019	%6	1155124626

Filtering options applied:

1. Bibliography excluded
2. Quotes included
3. Match size up to 5 words excluded

I declare that I have carefully read Hacettepe University Graduate School of Educational Sciences Guidelines for Obtaining and Using Thesis Originality Reports; that according to the maximum similarity index values specified in the Guidelines, my thesis does not include any form of plagiarism; that in any future detection of possible infringement of the regulations I accept all legal responsibility; and that all the information I have provided is correct to the best of my knowledge.

I respectfully submit this for approval.

Name Lastname:	Murat ÇINAR
Student No.:	N12246055
Department:	Computer Education and Instructional Technology
Program:	Computer Education and Instructional Technology
Status:	☐ Masters ☒ Ph.D. ☐ Integrated Ph.D.

ADVISOR APPROVAL

APPROVED
Prof. Dr. Hakan TÜZÜN

EK-İ: Yayımlama ve Fikrî Mülkiyet Hakları Beyanı

Enstitü tarafından onaylanan lisansüstü tezimin/raporumun tamamını veya herhangi bir kısmını, basılı (kâğıt) ve elektronik formatta arşivleme ve aşağıda verilen koşullarla kullanıma açma iznini Hacettepe Üniversitesine verdiğimi bildiririm. Bu izinle Üniversiteye verilen kullanım hakları dışındaki tüm fikri mülkiyet haklarım bende kalacak, tezimin tamamının ya da bir bölümünün gelecekteki çalışmalarda (makale, kitap, lisans ve patent vb.) kullanım haklan bana ait olacaktır.

Tezin kendi orijinal çalışmam olduğunu, başkalarının haklarını ihlal etmediğimi ve tezimin tek yetkili sahibi olduğumu beyan ve taahhüt ederim. Tezimde yer alan telif hakkı bulunan ve sahiplerinden yazılı izin alınarak kullanılması zorunlu metinlerin yazılı izin alınarak kullandığımı ve istenildiğinde suretlerini Üniversiteye teslim etmeyi taahhüt ederim.

Yükseköğretim Kurulu tarafından yayınlanan "**Lisansüstü Tezlerin Elektronik Ortamda Toplanması, Düzenlenmesi ve Erişime Açılmasına ilişkin Yönerge**" kapsamında tezim aşağıda belirtilen koşullar haricince YÖK Ulusal Tez Merkezi / H.Ü. Kütüphaneleri Açık Erişim Sisteminde erişime açılır.

- Enstitü/ Fakülte yönetim kurulu kararı ile tezimin erişime açılması mezuniyet tarihinden itibaren 2 yıl ertelenmiştir. (1)
- Enstitü/Fakülte yönetim kurulunun gerekçeli kararı ile tezimin erişime açılması mezuniyet tarihimden itibaren … ay ertelenmiştir. (2)
- Tezimle ilgili gizlilik kararı verilmiştir. (3)

17/05/2019

Murat ÇINAR

"Lisansüstü Tezlerin Elektronik Ortamda Toplanması, Düzenlenmesi ve Erişime Açılmasına İlişkin Yönerge"

(1) Madde 6. 1. Lisansüstü tezle ilgili patent başvurusu yapılması veya patent alma sürecinin devam etmesi durumunda, tez danışmanının önerisi ve enstitü anabilim dalının uygun görüşü üzerine enstitü veya fakülte yönetim kurulu iki yıl süre ile tezin erişime açılmasının ertelenmesine karar verebilir.

(2) Madde 6. 2. Yeni teknik, materyal ve metotların kullanıldığı, henüz makaleye dönüşmemiş veya patent gibi yöntemlerle korunmamış ve internetten paylaşılması durumunda 3. şahıslara veya kurumlara haksız kazanç; imkânı oluşturabilecek bilgi ve bulguları içeren tezler hakkında tez danışmanın önerisi ve enstitü anabilim dalının uygun görüşü üzerine enstitü veya fakülte yönetim kurulunun gerekçeli kararı ile altı ayı aşmamak üzere tezin erişime açılması engellenebilir.

(3) Madde 7. 1. Ulusal çıkarları veya güvenliği ilgilendiren, emniyet, istihbarat, savunma ve güvenlik, sağlık vb. konulara ilişkin lisansüstü tezlerle ilgili gizlilik kararı, tezin yapıldığı kurum tarafından verilir. Kurum ve kuruluşlarla yapılan işbirliği protokolü çerçevesinde hazırlanan lisansüstü tezlere ilişkin gizlilik kararı ise, ilgili kurum ve kuruluşun önerisi ile enstitü veya fakültenin uygun görüşü Üzerine üniversite yönetim kurulu tarafından verilir. Gizlilik kararı verilen tezler Yükseköğretim Kuruluna bildirilir.*
Madde 7.2. Gizlilik kararı verilen tezler gizlilik süresince enstitü veya fakülte tarafından gizlilik kuralları çerçevesinde muhafaza edilir, gizlilik kararının kaldırılması halinde Tez Otomasyon Sistemine yüklenir.

** Tez danışmanının önerisi ve enstitü anabilim dalının uygun görüşü üzerine enstitü veya fakülte yönetim kurulu tarafından karar verilir.*