

T.C.
MARMARA ÜNİVERSİTESİ
EĞİTİM BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ EĞİTİMİ ANABİLİM DALI
BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ ÖĞRETMENLİĞİ BİLİM
DALI

PROGRAMLAMA YÖNTEMLERİNİN
ORTAOKUL ÖĞRENCİLERİNİN BİLGİ İŞLEMSEL DÜŞÜNME
BECERİSİNE YÖNELİK ÖZ YETERLİK ALGISINA VE
PROGRAMLAMA BAŞARISINA ETKİSİ

Oğuzhan ÖZEL
(Yüksek Lisans Tezi)

İstanbul – 2019

T.C.
MARMARA ÜNİVERSİTESİ
EĞİTİM BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ EĞİTİMİ ANABİLİM DALI
BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ ÖĞRETMENLİĞİ BİLİM
DALI

PROGRAMLAMA YÖNTEMLERİNİN
ORTAOKUL ÖĞRENCİLERİNİN BİLGİ İŞLEMSEL DÜŞÜNME
BECERİSİNE YÖNELİK ÖZ YETERLİK ALGISINA VE
PROGRAMLAMA BAŞARISINA ETKİSİ

Oğuzhan ÖZEL
(Yüksek Lisans Tezi)

Danışman
Dr. Öğr. Üyesi Ahmet Feyzi SATICI

İstanbul – 2019



**Tüm kullanım hakları
M.Ü. Eğitim Bilimleri Enstitüsü'ne aittir.**

© 2019

ETİK İLKELERE UYGUNLUK BEYANI

Marmara Üniversitesi Eğitim Bilimleri Enstitüsü Müdürlüğü Lisansüstü Tez Yazım Kılavuzuna uygun olarak hazırladığım çalışmamda;

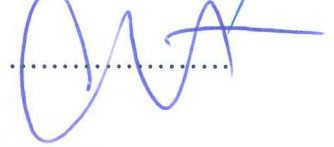
- Sunduğum bilgileri, dokümanları ve verileri akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Çalışmamda yararlandığım eserlerin tamamına atıfta bulunarak kaynak gösterdiğimi,
- Elde ettiğim verilerde ve sonuçlarda herhangi bir değişiklik yapmadığımı bildirir, aksi bir durumda aleyhimde doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

Yazarın Adı Soyadı : Oğuzhan ÖZEL

İmza :

ONAY

Oğuzhan ÖZEL tarafından hazırlanan “Programlama Yöntemlerinin Ortaokul Öğrencilerinin Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algısına ve Programlama Başarısına Etkisi” konulu bu çalışma 21 Kasım 2019 tarihinde yapılan savunma sınavı sonucunda jüri tarafından başarılı bulunmuş ve yüksek lisans tezi olarak kabul edilmiştir.

	Adı Soyadı	İmza
TEZ DANIŞMANI	Dr. Öğretim Üyesi Ahmet Feyzi SATICI	
JÜRİ ÜYESİ	Dr. Öğretim Üyesi Erol SÜZÜK	
JÜRİ ÜYESİ	Doç. Dr. Serhat Bahadır KERT	

ÖZGEÇMİŞ

2001 – Meliha ve Doğan Akad İlköğretim Okulu

2005 – İzmir Atatürk Anadolu Lisesi

2009 – Çanakkale Onsekiz Mart Üniversitesi – Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümü (Lisans Eğitimi)

2009 – MEB – Mustafa Vasıf Karslıgil İlköğretim Okulu – Bilişim Teknolojileri Öğretmenliği

2010 – 2013 - MEB – Saniye Sezgin Elmas İlköğretim Okulu – Bilişim Teknolojileri Öğretmenliği

2013 – 2015 - MEB – TOKİ Seyitnizam Şehit Semih Balaban İmam Hatip Ortaokulu – Bilişim Teknolojileri Öğretmenliği

2015 – Devam ediyor - MEB – Vedide Baha Pars Ortaokulu – Bilişim Teknolojileri Öğretmenliği

İletişim Bilgileri

Görev Yaptığı Kurum : MEB - Vedide Baha Pars Ortaokulu (İstanbul / Fatih)

E-posta : oguzhanozel35@hotmail.com

ÖNSÖZ

2009 yılında başlamış olduğum yüksek lisans eğitimine verdiğim uzun bir aradan sonra, tez aşamasında bilgisini, deneyimini ve değerli zamanını esirgemeyip her aşamada büyük destek veren değerli tez danışmanım Dr. Öğr. Üyesi Ahmet Feyzi SATICI'ya teşekkür ederim.

Öğretmenlik hayatım boyunca onlarla yaşadığım tüm güzel anılar ve tezime konu olan araştırma süreçlerine gönüllü ve istekli katılımları için öğrencilerime teşekkür ederim.

Bu günlere gelebilmem için gecesini gündüzüne katan, hayatın tüm zorluklarına karşı bize siper olan çok kıymetli annem Zekiye ÖZEL'e, desteği ve varlığı ile hayatımda çok önemli yeri olan dedem Hulki ÖZEL'e, yüksek lisans ders sürecinde ve hayatımın her anında yanımda olan ablaların en güzeli Neslihan ÖZ'e çok teşekkür ederim.

Son sözü varlıklarıyla hayatıma anlam katan eşim ve kızıma ayırmak istiyorum. Bu süreçte onlarla geçireceğim vakitten almasına rağmen sabır gösteren ve beni hep destekleyen mutluluk kaynaklarım sevgili eşim Nihan ÖZEL'e ve güzeller güzeli kızım Özge ÖZEL'e çok ama çok teşekkür ederim. Siz olmasanız bir yerde pes edebilirdim. İyi ki varsınız. İyi ki benim yanımdasınız.

Oğuzhan ÖZEL
İstanbul, 2019

ÖZET

Çalışmanın amacı, farklı programlama yöntemleri ile verilen eğitimin, bilgi işlemsel düşünme becerisine yönelik öz yeterlik algısına ve programlama başarısına etkisini araştırmaktır. Öğrenciler farklı programlama yöntemleri olarak robotik programlama, blok tabanlı programlama ve metin tabanlı programlama ile eğitim almışlardır. Öz yeterlik ve başarı açısından farkları ortaya çıkarmak adına ön test-son test kontrol gruplu araştırma modeline uygun yarı deneysel bir çalışma yapılmıştır. Çalışma grubu İstanbul'da bulunan bir devlet okulundaki 202 ortaokul 5. sınıf öğrencisinden (10 – 12 yaş arası) oluşmuştur. Çalışma grubu 90 erkek (% 44.5) ve 112 kız (% 55.5) öğrenci ile oluşmuştur. Bu öğrenciler öğretim yöntemlerine göre üç farklı grup oluşturmuşlardır. Eğitim öncesinde öğrencilere bilgi işlemsel düşünmeye yönelik öz yeterlik algılarını ölçmek için geçerlik ve güvenirlik çalışması yapılmış olan “Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algısı Ölçeği” uygulanmıştır. Eğitim süresi (sekiz hafta) boyunca 3 ayrı grupta farklı programlama yöntemleri ile eğitimlere devam edilmiş ardından tekrar “Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algısı Ölçeği” uygulanmıştır. Eğitimin sonunda ayrıca tüm öğrencilere bilgi işlemsel düşünme ve programlama ile ilgili kazanımlardan oluşan bir programlama bilgisi sınavı uygulanmıştır. Çalışma sonucunda robotik programlama eğitimi alan gruptaki öğrencilerin, blok veya metin tabanlı programlama yapan gruplardaki öğrencilere göre bilgi işlemsel düşünme becerisine yönelik öz yeterlik algısı seviyelerinin ve programlama başarılarının daha yüksek olduğu görülmüştür, fakat istatistiksel olarak anlamlı bir fark sadece robotik programlama grubu ile blok tabanlı programlama grubu arasında ortaya çıkmıştır. Blok tabanlı programlama grubu ile metin tabanlı programlama grupları arasında istatistiksel olarak anlamlı bir fark oluşmamıştır. Cinsiyete göre yapılan analizlerde ise hiçbir değişkene göre anlamlı bir farklılık görülmemiştir.

Anahtar kelimeler: Bilgi İşlemsel Düşünme, Programlama Eğitimi, Robotik

ABSTRACT

The aim of this study is to investigate effects of different programming methods on perception of self-efficacy for elementary students' computational thinking skills and programming achievement. Students are trained in robotics programming, block-based programming and text-based programming as different programming methods. In order to reveal the differences in terms of self-efficacy and success, a quasi-experimental study was conducted in accordance with the pre-test and post-test control group research model. The study group consisted of 202 middle school 5th grade students (10-12 years old) in a public school in Istanbul. The study group consisted of 90 boys (44.5%) and 112 girls (55.5%). These students formed three different groups according to their teaching methods. In order to measure students' perceptions of self-efficacy towards computational thinking skills, "Self-Efficacy Perception Scale for Computational Thinking Skills" was applied to the students before the training. During the training period (eight weeks), training was continued in 3 different groups with different programming methods and then "Self-Efficacy Perception Scale for Computational Thinking Skills" was applied again. At the end of the training, all students were given a programming knowledge exam consisting of computational thinking and gains related to programming. As a result of the study, it was seen that the students in the robotic programming group had higher levels of self-efficacy perception and programming achievement than the students in the block or text-based programming groups, but there was a statistically significant difference between the robotic programming group and the block-based programming group only. There was no statistically significant difference between the block-based programming group and the text-based programming groups. No significant difference was found in any analysis according to gender.

Key words: Computational Thinking, Programming Education, Robotics

İÇİNDEKİLER

ETİK İLKELERE UYGUNLUK BEYANI	i
ONAY	ii
ÖZGEÇMİŞ	iii
ÖNSÖZ	iv
ÖZET	v
ABSTRACT.....	vi
İÇİNDEKİLER	vii
TABLOLAR LİSTESİ	xi
ŞEKİLLER TABLOSU	xiv
KISALTMA VE SEMBOLLER.....	xv
BÖLÜM I: GİRİŞ	1
1.1. Problem	1
1.2 Araştırmanın Amacı.....	5
1.3. Araştırmanın Önemi	7
1.4. Sınırlılıklar	7
1.5. Tanımlar	8
BÖLÜM II: ALAN YAZIN / İLGİLİ ARAŞTIRMALAR.....	9
2.1. Bilgi İşlemsel Düşünme (BİD)	9
2.2. Bilgi İşlemsel Düşünme(BİD) ile İlgili Müfredat Bilgileri, Etkinlik Örneği	13
2.2.1. Müfredatımızda Bilgi İşlemsel Düşünme Becerisi ve Programlama	13
2.2.2. Bilge Kunduz (Uluslararası Enformatik ve Bilgi İşlemsel Düşünme Etkinliği)	15
2.3. Programlama ve Programlama Öğretimi	17
2.3.1. Blok Tabanlı Programlama	18
2.3.1.1. Blok Tabanlı Programlama Nedir?	18

2.3.1.2. Blok Tabanlı Programlama Öğretim Araçları	18
2.3.1.3. Blok Tabanlı Programlama ve BİD	18
2.3.2. Metin Tabanlı Programlama	22
2.3.2.1. Metin Tabanlı Programlama Nedir?	22
2.3.2.2. Metin Tabanlı Programlama Öğretim Araçları	23
2.3.2.3. Metin Tabanlı Programlama ve BİD	23
2.3.3. Robotik Programlama	24
2.3.3.1. Robot nedir?	24
2.3.3.3. Robotik Programlama ve BİD	26
2.3.4. Bilgisayarsız Uygulamalar	28
BÖLÜM III: YÖNTEM	30
3.1. Araştırmanın Modeli	30
3.2. Çalışma Grubu	31
3.3. Deney Süreci	32
3.4. Veri Toplama Araçları	34
3.4.1. Ortaokul Öğrencileri için Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algısı Ölçeği (BİDBÖA)	34
3.4.2. Programlama Bilgisi Sınavı	34
3.5. Verilerin Toplanması	35
3.6. Verilerin Çözümlemesi	35
BÖLÜM IV: BULGULAR	36
4.1. Varsayımların İncelenmesi	36
4.1.1. Deney Gruplarına Göre Varsayımların İncelenmesi	36
4.1.2. Cinsiyete Göre Varsayımların İncelenmesi	37
4.2. Grup Denkliklerinin İncelenmesi	38
4.2.1. Deney Gruplarına Göre Denklik İncelenmesi	38
4.2.2. Cinsiyete Göre Denklik İncelenmesi	39
4.3. Öğrencilerin Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algıları Ön Test ve Son Test Puanları Arasında Yapılacak Analizlere Ait Bulgular	40
4.3.1. Öğrencilerin Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algıları Ön Test ve Son Test Puanları Arasında Blok Tabanlı Programlama Öğretim Yöntemini Gören Öğrenciler İçin Fark Var mıdır? Sorusuna Ait Bulgular	40
4.3.2. Öğrencilerin Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algıları Ön Test ve Son Test Puanları Arasında Metin Tabanlı Programlama Öğretim Yöntemini Gören Öğrenciler İçin Fark Var mıdır? Sorusuna Ait Bulgular	41
4.3.3. Öğrencilerin Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algıları Ön Test ve Son Test Puanları Arasında Robotik Programlama Öğretim Yöntemini Gören Öğrenciler İçin Fark Var mıdır? Sorusuna Ait Bulgular	42

4.4. Öğrencilerin Katıldıkları Programlama Öğretim Yöntemine Göre Yapılacak Analizlere Ait Bulgular.....	44
4.4.1. Öğrencilerin Katıldıkları Programlama Öğretim Yöntemine Göre Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algıları Son Test Puanları Arasında Fark Var Mıdır? Araştırma Sorusuna Ait Bulgular	44
4.4.2. Öğrencilerin Katıldıkları Programlama Öğretim Yöntemine Göre Algoritma Tasarlama Yeterliği Alt Faktörü Puanları Arasında Fark Var Mıdır? Araştırma Sorusuna Ait Bulgular	46
4.4.3. Öğrencilerin Katıldıkları Programlama Öğretim Yöntemine Göre Problem Çözme Yeterliği Alt Faktörü Puanları Arasında Fark Var Mıdır? Araştırma Sorusuna Ait Bulgular	47
4.4.4. Öğrencilerin Katıldıkları Programlama Öğretim Yöntemine Göre Veri İşleme Yeterliği Alt Faktörü Puanları Arasında Fark Var Mıdır? Araştırma Sorusuna Ait Bulgular	48
4.4.5. Öğrencilerin Katıldıkları Programlama Öğretim Yöntemine Göre Temel Programlama Yeterliği Alt Faktörü Puanları Arasında Fark Var Mıdır? Araştırma Sorusuna Ait Bulgular	49
4.4.6. Öğrencilerin Katıldıkları Programlama Öğretim Yöntemine Göre Özgüven Yeterliği Alt Faktörü Puanları Arasında Fark Var Mıdır? Araştırma Sorusuna Ait Bulgular	51
4.4.7. Öğrencilerin Katıldıkları Programlama Öğretim Yöntemine Göre Programlama Başarıları Arasında Fark Var Mıdır? Araştırma Sorusuna Ait Bulgular	52
4.5. Öğrencilerin Cinsiyetlerine Göre Yapılacak Analizlere Ait Bulgular	54
4.5.1. Öğrencilerin Cinsiyetlerine Göre Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algıları Son Test Puanları Arasında Fark Var Mıdır? Araştırma Sorusuna Ait Bulgular	54
4.5.2. Öğrencilerin Cinsiyetlerine Göre Algoritma Tasarlama Yeterliği Alt Faktörü Puanları Arasında Fark Var Mıdır? Araştırma Sorusuna Ait Bulgular	55
4.5.3. Öğrencilerin Cinsiyetlerine Göre Problem Çözme Yeterliği Alt Faktörü Puanları Arasında Fark Var Mıdır? Araştırma Sorusuna Ait Bulgular	56
4.5.4. Öğrencilerin Cinsiyetlerine Göre Veri İşleme Yeterliği Alt Faktörü Puanları Arasında Fark Var Mıdır? Araştırma Sorusuna Ait Bulgular	57
4.5.5. Öğrencilerin Cinsiyetlerine Göre Temel Programlama Yeterliği Alt Faktörü Puanları Arasında Fark Var Mıdır? Araştırma Sorusuna Ait Bulgular	58
4.5.6. Öğrencilerin Cinsiyetlerine Göre Özgüven Yeterliği Alt Faktörü Puanları Arasında Fark Var Mıdır? Araştırma Sorusuna Ait Bulgular	59
4.5.7. Öğrencilerin Cinsiyetlerine Göre Programlama Başarı Puanları Arasında Fark Var Mıdır? Araştırma Sorusuna Ait Bulgular	60
BÖLÜM V: SONUÇ	62
5.1. Sonuçlar ve Tartışma	62
5.2. Öneriler	68
BÖLÜM VI: KAYNAKÇA	70
BÖLÜM VII: EKLER	86
Ek 1: Araştırma İzni	86
Ek 2: Blok Tabanlı Programlama Grubu Öğretim Aracı ve Eğitim Süreci	87
Ek 3: Metin Tabanlı Programlama Grubu Öğretim Aracı ve Eğitim Süreci	89

Ek 4: Robotik Programlama Grubu Öğretim Aracı ve Eğitim Süreci..... 91

Ek 5: Programlama Bilgisi Sınav Soru Örnekleri – Cevap Anahtarı – Belirtke Tablosu 94



TABLÖLER LİSTESİ

Tablo 1 : MEB – İlkokul 1, 2, 3 ve 4. Sınıflarda Uygulanan Bilişim Teknolojileri ve Yazılım Dersi Öğretim Programındaki Programlama Bilgisi ve BİD Becerilerine Yönelik Örnek Kazanımlar.....	14
Tablo 2 : MEB – Ortaokul 5. 6. 7. ve 8. Sınıflarda Uygulanan Bilişim Teknolojileri ve Yazılım Dersi Öğretim Programlarındaki Programlama Bilgisi ve BİD Becerilerine Yönelik Örnek Kazanımlar.....	14
Tablo 3 : MEB – Liselerde Uygulanan Bilgisayar Bilimi Dersi Öğretim Programındaki Programlama Bilgisi ve BİD Becerilerine Yönelik Örnek Kazanımlar	15
Tablo 4 : Araştırma Deseni.....	30
Tablo 5 : Deney Süreci	33
Tablo 6: Normallik Testi (Test Puanlarının Deney Gruplarına Göre Dağılımı).....	37
Tablo 7: Normallik Testi (Test Puanlarının Cinsiyete Göre Dağılımı)	38
Tablo 8: BİDBÖA Ön Test Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları.....	38
Tablo 9: BİDBÖA Ön Test Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili Kruskal Wallis H Testi Sonuçları.....	39
Tablo 10: BİDBÖA Ön Test Puanlarının Cinsiyete Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları	39
Tablo 11: BİDBÖA Ön Test Puanlarının Cinsiyete Göre Farklılaşma Durumu ile İlgili Kruskal Wallis H Testi Sonuçları	40
Tablo 12: Blok Tabanlı Programlama Eğitimi Alan Öğrencilerin BİDBÖA Ön Test ve Son Test Puanları ile İlgili Betimsel İstatistik Sonuçları.....	40
Tablo 13: Blok Tabanlı Programlama Eğitimi Alan Öğrencilerin BİDBÖA Ön Test ve Son Test Puanları için Wilcoxon İşaretleli Sıralar Testi Sonuçları	41
Tablo 14: Metin Tabanlı Programlama Eğitimi Alan Öğrencilerin BİDBÖA Ön Test ve Son Test Puanları ile İlgili Betimsel İstatistik Sonuçları.....	41
Tablo 15: Metin Tabanlı Programlama Eğitimi Alan Öğrencilerin BİDBÖA Ön Test ve Son Test Puanları için Wilcoxon İşaretleli Sıralar Testi Sonuçları	42
Tablo 16: Robotik Programlama Eğitimi Alan Öğrencilerin BİDBÖA Ön Test ve Son Test Puanları ile İlgili Betimsel İstatistik Sonuçları.....	43
Tablo 17: Robotik Programlama Eğitimi Alan Öğrencilerin BİDBÖA Ön Test ve Son Test Puanları için Wilcoxon İşaretleli Sıralar Testi Sonuçları	43
Tablo 18: BİDBÖA Son Test Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları	44
Tablo 19: BİDBÖA Son Test Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili Kruskal Wallis H Testi Sonuçları	45
Tablo 20: BİDBÖA Son Test Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili İkili Karşılaştırma Sonuçları.....	45
Tablo 21: Algoritma Tasarlama Yeterliği Alt Faktör Son Test Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları	46

Tablo 22: Algoritma Tasarlama Yeterliđi Alt Faktör Son Test Puanlarının Gruplara Göre Karşılaştırılması, Kruskal-Wallis Test Sonuçları	47
Tablo 23: Problem Çözme Yeterliđi Alt Faktör Son Test Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları	47
Tablo 24: Problem Çözme Yeterliđi Alt Faktör Son Test Puanlarının Gruplara Göre Karşılaştırılması, Kruskal-Wallis Test Sonuçları	48
Tablo 25: Veri İşleme Yeterliđi Alt Faktör Son Test Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları	48
Tablo 26: Veri İşleme Yeterliđi Alt Faktör Son Test Puanlarının Gruplara Göre Karşılaştırılması, Kruskal-Wallis Test Sonuçları	49
Tablo 27: Temel Programlama Yeterliđi Alt Faktör Son Test Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları	50
Tablo 28: Temel Programlama Yeterliđi Alt Faktör Son Test Puanlarının Gruplara Göre Karşılaştırılması, Kruskal-Wallis Test Sonuçları	50
Tablo 29: Özgüven Yeterliđi Alt Faktör Son Test Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları	51
Tablo 30: Özgüven Yeterliđi Alt Faktör Son Test Puanlarının Gruplara Göre Karşılaştırılması, Kruskal-Wallis Test Sonuçları	51
Tablo 31: Özgüven Yeterlik Algısı Son Test Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili İkili Karşılaştırma Sonuçları	52
Tablo 32: Programlama Başarı Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları	53
Tablo 33: Programlama Başarı Puanlarının Gruplara Göre Karşılaştırılması, Kruskal-Wallis Test Sonuçları	53
Tablo 34: Programlama Başarı Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili İkili Karşılaştırma Sonuçları	54
Tablo 35: BİDBÖA Son Test Puanlarının Cinsiyete Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları	54
Tablo 36: BİDBÖA Son Test Puanlarının Cinsiyete Göre Cinsiyete Göre Kruskal Wallis H Testi Sonuçları	55
Tablo 37: Algoritma Tasarlama Yeterliđi Alt Faktörü Son Test Puanlarının Cinsiyete Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları	55
Tablo 38: Algoritma Tasarlama Yeterliđi Alt Faktörü Son Test Puanlarının Cinsiyete Göre Cinsiyete Göre Kruskal Wallis H Testi Sonuçları	56
Tablo 39: Problem Çözme Yeterliđi Alt Faktörü Son Test Puanlarının Cinsiyete Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları	56
Tablo 40: Problem Çözme Yeterliđi Alt Faktörü Son Test Puanlarının Cinsiyete Göre Cinsiyete Göre Kruskal Wallis H Testi Sonuçları	57
Tablo 41: Veri İşleme Yeterliđi Alt Faktörü Son Test Puanlarının Cinsiyete Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları	57
Tablo 42: Veri İşleme Yeterliđi Alt Faktörü Son Test Puanlarının Cinsiyete Göre Cinsiyete Göre Kruskal Wallis H Testi Sonuçları	58
Tablo 43: Temel Programlama Yeterliđi Alt Faktörü Son Test Puanlarının Cinsiyete Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları	58

Tablo 44: Temel Programlama Yeterliđi Alt Faktörü Son Test Puanlarının Cinsiyete Göre Cinsiyete Göre Kruskal Wallis H Testi Sonuçları.....	59
Tablo 45: Özgüven Yeterliđi Alt Faktörü Son Test Puanlarının Cinsiyete Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları.....	59
Tablo 46: Özgüven Yeterliđi Alt Faktörü Son Test Puanlarının Cinsiyete Göre Cinsiyete Göre Kruskal Wallis H Testi Sonuçları.....	60
Tablo 47: Programlama Başarı Puanlarının Cinsiyete Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları.....	60
Tablo 48: Programlama Başarı Puanlarının Cinsiyete Göre Cinsiyete Göre Kruskal Wallis H Testi Sonuçları	61
Tablo 49 : Blok Tabanlı Programlama Eğitime Ait Günlük Ders Planı Örneđi (5.hafta)	88
Tablo 50 : Metin Tabanlı Programlama Eğitime Ait Günlük Ders Planı Örneđi (5.hafta)	90
Tablo 51 : Robotik Programlama Eğitime Ait Günlük Ders Planı Örneđi (5.hafta).....	93
Tablo 52 : Programlama Bilgisi Sınavı Soru Örnekleri - Cevapları ve Kazanımları	94

ŞEKİLLER TABLOSU

Şekil 1: BİD ile ilgili Beceri Boyutları Grafiği.....	2
Şekil 2: Bilge Kunduz Etkinliği Örnek Sorusu.....	16
Şekil 3 : Araştırmada Kullanılan Ortaokul Öğrencileri için Bilgi İşlemsel Düşünme Becerilerine Yönelik Öz Yeterlik Algısı Ölçeği'nin Kullanımı için Ölçeği Hazırlayan Araştırmacılardan Gelen İzin E-postası	86
Şekil 4 : Ortaokul Öğrencileri için Bilgi İşlemsel Düşünme Becerilerine Yönelik Öz Yeterlik Algısı Ölçeği'nin Kullanımı Hakkında İstanbul İl Milli Eğitim Müdürlüğünden Alınan İzin Yazısı	86
Şekil 5 : Code.org Öğrenci Takip Ekranı	87
Şekil 6 : Code.org Kurs 2 Ders 13 / 4. Programlama Görevi ve Çözümü.....	88
Şekil 7 : Small Basic Programı Arayüzü ve Örnek Kod	89
Şekil 8 : Metin Tabanlı Programlama Eğitimi Örnek Programlama Görevi Çözümü	90
Şekil 9 : Robotis Dream Seviye 2'den Örnek Robotlar	91
Şekil 10 : RoboPlus Task Programı Arayüzü ve Örnek Kod	92
Şekil 11 : Robotik Programlama Grubu - Örnek Robot ve Koşul Yapısı Örneği	93

KISALTMA VE SEMBOLLER

BİD: Bilgi İşlemsel Düşünme

BİDBÖA: Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algısı Ölçeği

B3: Bilgisayarsız Bilgisayar Bilimi

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

IEEE: The Institute of Electrical and Electronics Engineers (Elektrik ve Elektronik Mühendisleri Enstitüsü)

TDK: Türk Dil Kurumu

BÖLÜM I: GİRİŞ

1.1. Problem

Dünya üzerinde bilinen her zaman dilimi farklı olaylar ile hatırlanmaktadır. Yaşanmış olaylar, yaşamış insanlar, buluşlar vb. çağların isimlerini, başlangıçlarını ve bitişlerini etkilemiştir. Yaşamakta olduğumuz zaman dilimi ise muhakkak bazı olaylar, kişiler ve teknolojik gelişmeler ile anılacaktır. Günümüz dünyasını etkileyen bilgi, teknoloji, ihtiyaçlar, beceriler, meslekler gibi kavramlarının ortak paydasında ise “hız ve değişim” bulunmaktadır. Hızlı değişen bilgi, hızlı değişen teknoloji, hızlı değişen ihtiyaçlar, hızlı değişen beceriler, hızlı değişen meslekler vb. hızlı değişimler günümüzün hatırlanacak en önemli unsurları gibi gözükmemektedir.

Günümüzdeki hız ve değişime ayak uydurabilen bireylerin nasıl yetiştirileceği konusunda ise uzmanlar farklı görüşler ortaya koymaktadır. Teknoloji okuryazarlığı ve problem çözme yeteneği bu hız ve değişime ayak uydurmak için olmazsa olmaz yeteneklerdendir. Teknoloji okuryazarlığı ve problem çözme becerisini kazanmayı farklı bir bakış açısı ile ele alan Papert (1980) bilgi işlemsel düşünme (BİD) becerisini alanyazına kazandırmıştır. Bu çalışmalardan yaklaşık 26 yıl sonra Wing (2006) BİD kavramından bahsetmiş:

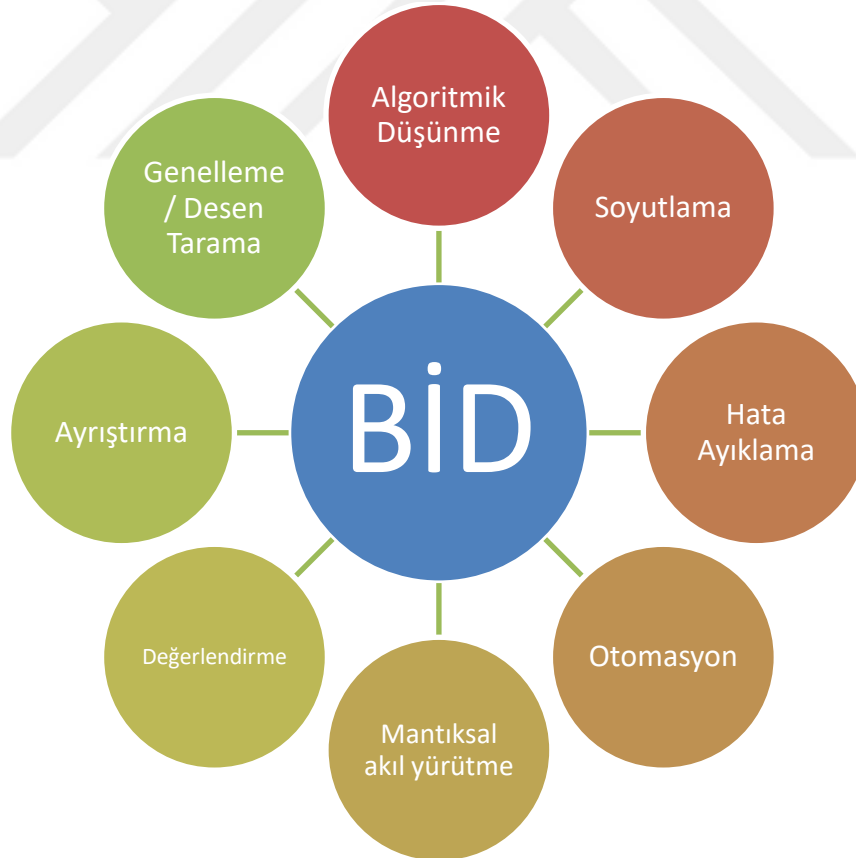
“Bilgi işlemsel düşünme, zor görülen bir problemi, çözülebilecek şekle dönüştürmek üzere tekrar formüle etmektir..... Yinelemeli olarak düşünme sürecidir..... Büyük karmaşık bir sistemle karşılaşıldığında ya da böyle bir sistemi tasarlarlarken soyutlama ve ayırtırmayı kullanmaktır..... Hasar önleme ve hata düzeltme yoluyla, en kötü durum senaryolardan korunma, bu senaryoları engelleme ya da telafi etme üzerine düşünme sürecidir..... Sonuca ulaşmak için sezgisel akıl yürütmeyi kullanmaktır.” (akt. Kert, 2017, ss.1-22).

Cuny, Snyder ve Wing (2010) tarafından “BİD, bir bilgi işlemci (bilgisayar veya insan) tarafından problemleri ve çözümlerini formüle etmek için gerçekleştirilen düşünme sürecidir” diye tanımlanmıştır (akt. Wing 2010). Özet olarak BİD, çeşitli özellikler ve yaklaşımlar içeren bir problem çözme sürecidir (Keşfet Projesi, 2019).

Bilgisayar bilimcinin birçok süreçte kullandığı BİD süreçlerinin günlük hayatta kullanılabileceği düşünülmektedir. Wing (2006), sadece bilgisayar bilimcilerin değil

herkesin sahip olması gereken bir beceri olarak tanımlamıştır. Bu tanımlamasından yaklaşık 10 yıl sonra Wing (2017) öngörüsünü şöyle ifade etmiştir; BİD, 21. yüzyılın ortalarında herkes tarafından kullanılan okuma, yazma ve aritmetik gibi temel bir beceri olacaktır. Bu öngörüü destekleyecek şekilde BİD becerilerini kullanan insanların günlük hayattaki problemlere daha rahat çözüm bulabileceğine dair araştırmalar yapılmaktadır (Angeli vd., 2016; Brennan ve Resnick, 2012; Mikropoulos ve Bellou, 2013; Ruesink, 2018). Son yıllarda günlük hayatta kullanılabilecek bir beceri olarak tanımlanan ve eğitim müfredatlarında yerini alan BİD, tek bir tanımla ifade edilemeyecek kadar kapsamlı bir beceridir. Araştırmacılar BİD’i daha iyi ifade edip anlatabilmek için, BİD’i oluşturan alt beceri boyutlarından bahsetmişlerdir.

Yeni (2018), çalışmasında BİD becerisi ile ilgili alt beceri boyutlarını grafiğe dökmüştür. Bu grafik ile BİD kavramını oluşturan alt becerileri net bir şekilde ifade etmiştir (Şekil 1).



Şekil 1: BİD ile ilgili Beceri Boyutları Grafiği

BİD'i oluşturan tüm becerilerin çok önemli olduğu, süreç içinde bu alt becerilerin tamamını kullanarak "bilgi işlemsel düşünür" (computational thinker) olunabileceği görülmektedir. Uluslararası Eğitim Teknolojileri Topluluğu (International Society for Technology in Education - ISTE) 21. yüzyılda öğrencilerde olması gereken 7 özellikten birini "bilgi işlemsel düşünür olmak" olarak göstermiştir (ISTE, 2016). Bu özelliğe sahip bireyler de bu yönde yapılandırılmış bir eğitim sürecinden yetiyecektir. BİD'in kazanımlara entegre edildiği eğitim süreçleri için de çalışmalar yapılmaktadır.

Bu çalışmalardan birinde Weinberg (2013) BİD becerisinin geliştirilmesi için yapılacak bilgisayar bilimi uygulamalarına örnekler vermiştir. Bu uygulamalar şu şekildedir:

- Bilgisayarsız uygulamalar
- Blok tabanlı uygulamalar
- Disiplinler arası uygulamalar
- Robot programlama

Bilgisayarsız uygulamalarda, bilgisayar ve teknoloji desteği olmadan da etkinlikler oluşturulup BİD becerisi geliştirilebilir. Ülkemizde ve dünyada bu tarz örneklerin sayıları günümüzde gitgide artmaktadır. Dünya genelinde uygulanan örnekler; CSunplugged.org, CS4FN, Code.org unplugged, Bebras ve Türkiye'de uygulanan örnekler; Keşf@ Projesi, Bilge Kunduz ve Tospaa.org olarak gösterilebilir.

Blok tabanlı programlama ortamları, programlama eğitimine yeni başlayanların görseller, grafikler ve nesneler ile programlama yapmalarını sağlayan platformlardır. Daha çok küçük yaş gruplarına hitap etmektedir. Sürükle bırak mantığı ile çalışmaktadır. Code.org, Scratch, AppInventor, Alice, Blockly vb. birçok ortam bulunmaktadır. BİD becerisini geliştirmede özellikle küçük yaş gruplarında çok etkilidir.

Metin tabanlı programlama daha geniş kapsamlı, karmaşık zihinsel süreçleri içerip birçok beceriyi geliştirmektedir. C, C++, Java, Python, Small Basic vb. metin tabanlı programlama dilleri ile BİD'e ait alt beceri boyutlarının geliştirilebileceği düşünülmektedir. Örneğin, Lee vd. (2011) blok ve metin tabanlı programlama ortamlarının BİD becerilerini desteklemedeki olumlu etkilerinden bahsetmiştir.

Disiplinler arası uygulamalar, BİD becerilerinin sadece bilgisayar bilimlerinde değil diğer disiplinlerde de kullanılabileceğini, hatta bu kullanımlar arasında bağ oluşturarak öğrenmenin kalıcılığını ve disiplinler arası aktarımını artırmaktadır. Ülkemizde yürütülen “Keşfet Projesi” kapsamında oluşturulan internet sitesinde disiplinler arası uygulamaların yararına şöyle vurgu yapılmıştır:

“BİD, bilgisayar uygulamalarının geliştirilmesi açısından çok önemlidir, ancak sosyal bilimler, matematik ve fen bilimleri de dahil olmak üzere tüm disiplinlerde problem çözme desteklemek için kullanılabilir. Müfredatları gereği BİD’i öğrenen öğrenciler, akademik alanlar arasındaki ilişkileri görebilir, sınıf içindeki ve sınıf dışındaki yaşam arasında ilişki kurabilir.” (Keşfet Projesi, 2019).

Robotik uygulamalar, günümüzde kullanım sıklığı gittikçe artan robotik programlama ile öğrencilere yazdıkları kodu somut olarak test etme (Yeni, 2018), işbirlikli öğrenme (Huang ve Wu, 2012), problem çözme becerilerini geliştirme (Atmatzidou, Demetriadis ve Nika, 2018), ince motor ve mühendislik becerilerini geliştirme (Egushi, 2016) ve motivasyonunu artırma (Saad, Shuff, Loewen ve Burton, 2012) imkanı sunar. Ülkemizde ve dünyada Robotis Dream, Lego Wedo, Lego Mindstorm, Mbot, Vex Robotic vb. setler ile okul öncesi, ilkökul, ortaokul ve lise düzeyinde robotik eğitimi verilmektedir. BİD ile ilgili becerilerin robotik ile geliştirilebileceğinden bahseden çalışmalar vardır (Berland, Martin, Benton, Petrick Smith ve Davis, 2013; Berland ve Wilensky, 2015; Bers, Flannery, Kazakoff ve Sullivan, 2014; Chaudhary, Agrawal, Sureka ve Sureka, 2016). Araştırmacılar son zamanlarda eğitsel robotiğin özellikle ayırttırma, soyutlama, algoritma oluşturma ve hata ayıklama gibi beceriler yoluyla BİD becerisini geliştireceğini aktarmaktadırlar (Eguchi, 2014a; Eguchi, 2014b; Eguchi, 2014c; Ioannaou ve Makridou, 2018).

Aynı zamanda, bazı çalışmalar ise robotiğin öğrencilerde BİD becerileri geliştirmede yeterli etkiyi göstermediğini ifade etmiştir (Benitti, 2012). Araştırmacılar, eğitimde robotik kullanımı ile ilgili alanyazının çoğunun, bireysel inisiyatiflerle olumlu sonuçlar elde eden öğretmenlerin raporlarına dayandığını vurgulayıp sonuçların genellenmesine temkinli yaklaşmışlardır (Caci, Cardaci ve Lund, 2003; Petre ve Price, 2004; Williams, Ma, Prejean, Lai, Ford, 2007).

Kert’e (2017) göre 1980’li yıllarda Papert’ın erken yaşta programlama eğitimi ve BİD kavramından bahsettiği ilk çalışmaları ile başlayan ve disiplinler arası ilişkilendirmenin

olmaması, vizyon eksikliği gibi nedenlerle o yıllarda vazgeçilen bilgisayar bilimleri eğitiminin toplumsal yaşamın bir parçası haline gelmesi fikri günümüzde yeniden canlanmaktadır. Bu fikrin yeniden canlanması ile bilgisayar bilimi eğitimi sadece bir teknoloji eğitimi olmaktan çıkıp, tüm toplumu etkileyecek bir biçime dönüşmüştür. Bu dönüşüm bireylerin 21.yüzyılda edinmesi gereken becerilerin de değişmesine sebebiyet vermiştir. ISTE (2016) belirlediği standartlar ile öğrencilerin edinmesi gereken bazı becerileri listelemiştir.

- Yetkin Öğrenen – (Empowered Learner)
- Dijital Vatandaş – (Digital Citizen)
- Bilgiyi Düzenleyen – (Knowledge Constructor)
- Yenilikçi Tasarımcı – (Innovative Designer)
- Bilgi İşlemsel Düşünür – (Computational Thinker)
- Yaratıcı İletişimci - Creative Communicator
- Global İşbirlikçi - Global Collaborator

Listede görüldüğü gibi bilgi işlemsel düşünme becerisi ISTE'nin (2016) günümüz dünyasında öğrencilerde bulunması gereken en önemli becerilerden biri sayılmıştır.

Bu araştırmada günümüz bireylerinin edinmesi gereken becerilerden olan BİD becerisini merkeze alarak, bilgisayar bilimleri alanında günümüzde yakalanan bu ivme ile olumlu veya olumsuz sonuçlar bulan araştırmaların ışığında “Programlama Yöntemlerinin ve Cinsiyetin Ortaokul Öğrencilerinin Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algısına ve Programlama Başarısına Etkisi” araştırılacaktır.

1.2 Araştırmanın Amacı

Bu araştırmanın amacı, farklı programlama yöntemleri ile verilen eğitimin ve cinsiyetin, bilgi işlemsel düşünme becerisine yönelik öz yeterlik algısına ve programlama başarısına etkisini araştırmaktır. 2018 – 2019 Eğitim Öğretim yılında sekiz haftalık sürede gerçekleştirilmiştir.

Programlama yöntemlerinin ve cinsiyetin ortaokul öğrencilerinin bilgi işlemsel düşünme becerilerine yönelik öz yeterlik algısına ve programlama başarısına etkisi var mıdır?

Araştırmanın Alt Problemleri

1. Öğrencilerin bilgi işlemsel düşünme becerisine yönelik öz yeterlik algıları ön test ve son test puanları arasında
 - a. Blok tabanlı programlama öğretim yöntemini gören öğrenciler için fark var mıdır?
 - b. Metin tabanlı programlama öğretim yöntemini gören öğrenciler için fark var mıdır?
 - c. Robotik tabanlı programlama öğretim yöntemini gören öğrenciler için fark var mıdır?
2. Öğrencilerin katıldıkları programlama öğretim yöntemine göre:
 - a. Bilgi işlemsel düşünme becerisine yönelik öz yeterlik algıları puanları arasında fark var mıdır?
 - b. Algoritma tasarlama yeterliği alt faktörü puanları arasında fark var mıdır?
 - c. Problem çözme yeterliği alt faktörü puanları arasında fark var mıdır?
 - d. Veri işleme yeterliği alt faktörü puanları arasında fark var mıdır?
 - e. Temel programlama yeterliği alt faktörü puanları arasında fark var mıdır?
 - f. Özgüven yeterliği alt faktörü puanları arasında fark var mıdır?
 - g. Programlama başarıları arasında fark var mıdır?
3. Cinsiyete göre
 - a. Bilgi işlemsel düşünme becerisine yönelik öz yeterlik algıları puanları farklılaşmakta mıdır?
 - b. Algoritma tasarlama yeterliği alt faktörü puanları farklılaşmakta mıdır?
 - c. Problem çözme yeterliği alt faktörü puanları farklılaşmakta mıdır?
 - d. Veri işleme yeterliği alt faktörü puanları farklılaşmakta mıdır?
 - e. Temel programlama yeterliği alt faktörü puanları farklılaşmakta mıdır?
 - f. Özgüven yeterliği alt faktörü puanları farklılaşmakta mıdır?
 - g. Programlama başarıları farklılaşmakta mıdır?

1.3. Araştırmanın Önemi

Dünyada ve ülkemizde farklı uygulamalar ile karşımıza çıkan robotik programlama, blok tabanlı programlama ve metin tabanlı programlama eğitimlerinin bilgi işlemsel düşünme becerisine yönelik öz yeterlik algısını ve programlama başarısını ne kadar etkilediğini öğrenmek açısından çalışmanın sonuçları önemlidir. Bu çalışma;

- Ülkemizde programlama eğitimine erken yaşlarda başlanılan örneklerin arttığı günümüzde alandaki eğitimcilere katkı sağlayacaktır.
- BİD ile ilgili kazanımları ders içi aktivitelerine katmak isteyen eğitimcileri cesaretlendirecektir.
- Okullarında kısıtlı imkanları dahi olsa farklı programlama yöntemleri ile öğrencilerinin 21. Yüzyıla uygun bir şekilde “bilgi işlemsel düşünür” olarak yetiştirmeleri konusunda eğitimcilere rehberlik edecektir.
- Programlamanın elektronik cihazlar üzerinde somut olarak uygulandığı ve öğrencilerin bilgisayar bilimine karşı ilgisini arttıran “robotik” alanının önemini eğitimcilere ve velilere gösterecektir.

Özetle araştırma sonucu alandaki diğer çalışmalar ile değerlendirilip Milli Eğitim Bakanlığı’nın müfredat oluşturucu kurullarına fikir verecektir. Okullarda bu müfredatları uygulayacak eğitimcilere yol gösterici olacaktır.

1.4. Sınırlılıklar

Uygulama süreci 8 hafta ile sınırlıdır.

Blok tabanlı programlama, Code.org eğitim sitesi ile sınırlıdır.

Robotik programlama, eğitimde kullanılan Robotis Dream eğitim setleri ile sınırlıdır.

Metin tabanlı programlama, Small Basic yazılımı ile sınırlıdır.

Robotik programlama ve metin tabanlı programlama eğitimine katılan öğrenciler gönüllük esaslı ile eğitime dahil olmuşlardır. Fakat blok tabanlı programlama eğitimine katılan öğrenciler Bilişim Teknolojileri ve Yazılım Dersi kapsamında eğitim almışlardır. Herhangi bir gönüllülük yoktur.

Programlama bilgisi sınavı, araştırmacının da içinde bulunduğu alanında uzman 3 öğretmen tarafından hazırlanmıştır. Sınavın ölçme ve değerlendirme işlemleri de yine aynı öğretmenler tarafından yapılmıştır.

1.5. Tanımlar

Bilgi İşlemsel Düşünme (BİD): Bir bilgi işlemci (bilgisayar veya insan) tarafından problemleri ve çözümlerini formüle etmek için gerçekleştirilen düşünme sürecidir (Cuny, Snyder ve Wing, 2010; akt., Wing, 2010).

Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algısı Ölçeği (BİDBÖA): Bu çalışmada kullanılan ve Gülbahar, Kert ve Kalelioğlu (2018) tarafından hazırlanmış olan ölçektir.

Bilgisayarsız Bilgisayar Bilimi (B3): Bilgisayar bilimine ait kavramları herhangi bir elektronik cihaz kullanmadan öğretme yaklaşımıdır (Kalelioğlu, 2017). Kalelioğlu (2017) “computer science unplugged” olarak bahsi geçen kavramı “bilgisayarsız bilgisayar bilimi” ve kısaltmasını da “B3” olarak aktarmıştır.

BÖLÜM II: ALAN YAZIN / İLGİLİ ARAŞTIRMALAR

2.1. Bilgi İşlemsel Düşünme (BİD)

Bilgi işlemsel düşünme, yabancı kaynaklarda computational thinking olarak adlandırılmakla beraber ülkemizdeki farklı kaynaklarda farklı çevirileri bulunmaktadır. “Bilgisayarca Düşünme” (Korkmaz, Çakır, Özden, Oluk ve Sarıoğlu, 2015; Özden, 2015), “Kompütasyonel Düşünme” (Aldağ ve Tekdal, 2015; Şahiner ve Kert, 2016), “Bilgi-İşlemsel Düşünme” (Barut, Tuğtekin, ve Kuzu, 2016; Demir ve Seferoğlu, 2017; Gülbahar, Kert ve Kalelioğlu, 2018; MEB, 2018a), “Bilgisayımsal Düşünme” (Doğan, Çınar, Bilgiç, ve Tüzün, 2015), “Hesaplamalı Düşünme” (Yecan, Özçınar ve Tanyeri, 2017), ve “Bilişimsel Düşünme” (Sayın ve Seferoğlu, 2016) gibi farklı çevirileri bulunmaktadır. Bu çalışmada ise “Bilgi İşlemsel Düşünme (BİD)” Türkçe karşılığı kullanılmıştır. Bu tercih yapılırken, Wing’in (2006) “bir kişi bu terimin (computational thinking) bir bilgisayar gibi düşünmek anlamına geldiğini düşünebilir ve bu yaygın bir hatadır, çünkü bilgisayar cansızdır ve düşünmez, insan zeki ve yaratıcıdır; bilgisayar akıllı ve rasyoneldir.” açıklaması dikkate alınmıştır. Bu düşünme biçiminin alt basamaklarını da kapsadığı düşünüldüğünden bu çalışmada bilgi işlemsel düşünme (BİD) ifadesi kullanılmıştır.

Bilgi işlemsel düşünme kavramı ilk olarak Papert (1980) ve Papert ve Harel’in (1991) çalışmasında karşımıza çıkmaktadır. Fakat bu çalışmalarda kavrama ait bir açıklama bulunmamaktadır. Papert’in (1996) başka bir çalışmasında yine BİD kavramını kullanmış ama detay bilgi vermemiştir. BİD kavramı Papert’in çalışmasından yaklaşık 20 yıl sonra Wing’in (2006) çalışmasında, bilgisayar biliminin temel kavramlarını kullanarak problem çözme, sistemleri tasarlamayı ve insan davranışını anlamayı içeren bir düşünme becerisi olarak tanımlanmıştır. Wing’in bu kavramı tekrardan gündeme getirmesi ve teknolojik gelişmeler göz önüne alındığında BİD kavramının öneminin ve popüleritesinin artmasına sebep olmuştur. Alana oluşan bu ilgi konu ile ilgili araştırmaların artmasını sağlamıştır (Denning, 2017; Kukul, 2018). Son yıllarda sayıları giderek artan alana ait araştırmalarda BİD kavramının farklı yönleriyle tanımlandığı görülmektedir. BİD kavramının isminde olduğu gibi tanımında da birçok farklı görüş bulunmaktadır.

Wing'e (2006) göre BİD, bilgisayar bilimi için temel olan kavramları kullanarak; problem çözmeye, sistemleri tasarlama ve insan davranışını anlama yöntemi veya yaklaşımı olarak tanımlanabilir. Uluslararası Eğitim Teknolojileri Topluluğu (International Society for Technology in Education - ISTE) BİD'i, öğrencilerin teknoloji kullanımının çözüm geliştirme ve çözümleri test etme gücünden yararlanarak problemleri anlaması ve çözecek stratejiler geliştirmesi olarak tanımlamaktadır (ISTE, 2016). BİD kavramını, teknolojiyi kullanmanın problem çözümündeki gücünden ve bilgisayar biliminin temelinden yola çıkarak tanımlayan (Wing, 2006; ISTE, 2016) çalışmaları destekleyen bir tanım da Carnegie Mellon Üniversitesi Bilgi İşlemsel Düşünme Merkezi tarafından yapılmıştır. Bu tanımda ISTE'ye benzer şekilde bilgisayar bilimlerini kullanmayı BİD'in merkezine almıştır. Carnegie Mellon Üniversitesi Bilgi İşlemsel Düşünme Merkezi'ne göre BİD;

- bilgisayar biliminin temel kavramlarını kullanarak; problemleri çözmenin, sistemleri tasarlamamanın ve insan davranışını anlamamanın bir yoludur,
- problemleri daha etkili bir şekilde anlamak ve çözmek için farklı soyutlama düzeylerini oluşturmak ve kullanmaktır,
- matematiksel kavramları uygulama yeteneğiyle ve algoritmik düşünerek daha verimli, adil ve güvenli çözümler geliştirmektir (Center for Computational Thinking Carnegie Mellon, 2019).

Öğrencilerde BİD becerisinin geliştirilmesi üzerine her yıl çevrimiçi düzenlenen uluslararası bir etkinlik olan Bebras (Bilge Kunduz) BİD'i, yazılım mühendislerinin programlar ve uygulamalar yazmak için kullandıkları bir dizi problem çözmeye beceri ve tekniklerini tüm bireylerin birçok farklı alanda kullanabilmesi olarak tanımlamaktadır (Bebras.org, 2019). Her yıl dünya genelinde milyonlarca öğrenciyi BİD ile tanıştıran bebras.org'da belirtildiği gibi BİD becerisinin sadece bilgisayar bilimcilerde değil tüm bireylerde olması gerektiğinden bahseden birçok çalışma bulunmaktadır (Altaher ve Ferchichi, 2018; Lye ve Koh, 2014; Wing, 2006). Ülkeler de eğitim müfredatlarına BİD becerisini ekleyerek, tüm bireylere bu beceriyi kazandırmayı hedeflemişlerdir (Hubwieser vd., 2015).

Sadece bilgisayar bilimcilerin değil tüm bireylerin BİD becerisini edinmesi gerektiğinde hem fikir olan alan araştırmacıları, Kalelioğlu, Gülbahar ve Kukul'un (2016) belirttiği

gibi BİD'in tanımı üzerinde ortak noktaya varamamışlardır. Bilgisayar bilimi ve BİD'in nispeten yeni alanlar olması sebebiyle tanımlarda fikir birliğinin olmaması doğal karşılanmaktadır (Çetin ve Uçar, 2017). Bazı araştırmacılar bu fikri destekleyerek, BİD'in kesin bir tanımının gerekli olmadığını ifade etmişlerdir (Guzdial, 2011; Hu, 2011). Tam bu noktada Denning (2017) bilgi işlemsel düşünmeyi -zihinsel bir beceri olarak- nelerin oluşturduğunu öğrenilmesinin çok daha önemli olduğunu belirterek BİD becerisini oluşturan alt boyutların önemine dikkat çekmiştir. Bu boyutlara verilecek önem ile gençlerin bilime ilgisinin artabileceğini ve bilimi ilerletmek için insanoğlunun yeteneğinin artacağını belirtmiştir.

BİD becerisinin alt boyutları ile ilgili araştırmalar incelendiğinde ortak noktaların çoğaldığı görülmektedir. Bu alanda uzun yıllardır çalışan ISTE BİD'in soyutlaştırma, veri toplama, veri analizi, algoritmik düşünme, sorunları bileşenlerine ayırma, problem çözme, otomasyona dönüştürme gibi alt faktörlerden oluştuğu belirtilmiştir (ISTE, 2016). BİD'i daha az ama benzer alt boyutlara ayıran başka bir çalışmada Csizmadia, Curzon, Dorling, Humphreys, Ng, Selby, ve Woolard (2015) BİD'in; algoritmik düşünme, analiz etme, genelleme, soyutlama, değerlendirme alt becerilerden oluştuğunu belirtmiştir.

Dünya genelinde milyonlarca öğrenciye çevrimiçi olarak enformatik ve BİD etkinliği sunan Bebras.org (bilge kunduz) BİD'i; karmaşık görevleri basit bileşenlere ayırma, algoritma tasarımı, örüntü tanıma, örüntü genelleştirme ve soyutlamanın dahil olduğu problem çözme becerileri olarak nitelendirmektedir (Bebras.org, 2019). Problem çözme becerisi boyutundan bahseden bir başka çalışmada Selby ve Woolard (2013) BİD'i; soyutlama yapabilme, parçalarına ayırabilme, algoritmik düşünme, değerlendirme yapabilme, genellemeler yapabilme yeteneklerini yansıtan bilişsel bir düşünme becerisi olarak tanımlamıştır. Barr, Harrison ve Conery (2011) BİD'e ait alt boyutların birçok ders içeriğinde öğrencilerin karşısına çıktığını aktarmıştır. Dersler dahil hayatın birçok yerinde öğrencilerin karşılaştığı -zengin alt boyut becerilerine sahip olan- BİD'i Altaher ve Ferchichi (2018), neredeyse tüm disiplinlere ve çalışma alanlarına uygulanan oldukça önemli ve değerli bir düşünce şekli olarak tanımlayarak BİD ve alt boyut becerilerinin önemine işaret etmiştir.

Alan arařtırmacıları ve eęitimcileri tarafından kabul görmüş bazı BİD alt boyutlarına ait kısa açıklamalar ařaęıda verilmiřtir.

Veri Toplama ve Analizi: Veri, problemin veya sorunun bir parçası olan bilgidir, bilgilerin nasıl organize edildięini ve nasıl eriřildięini gösterir (Perkovic, Settle, Hwang, ve Jones, 2010). Problemin çözümüne giden yoldaki ilk adım problemi anlamaktır. Problemi anlayabilmek adına iře yarayacak tüm verilerin tespiti ve toplanması önemlidir. Toplanan bu veriler analiz edilerek problem çözümündeki dięer ařamalarda da kullanılır.

Algoritmik Düşünme: Tüm adımların açık bir şekilde tanımlanmasıyla çözüme ulaşmanın bir yoludur. Bazı problemler bir defalıktır; çözülürler, çözümler uygulanır ve bir daha karřımıza çıkmazlar. Algoritmik düşünme, benzer problemlerin tekrar tekrar karřımıza çıkıp çözülmesi gerektiğinde kullanılacak bir düşünme biçimidir. Her seferinde yeniden düşünmeyi gerektirmeyecek şekilde her zaman iře yarayan bir çözüme ulařtırır (Csizmadia vd., 2015).

Deęerlendirme: Bir algoritmanın hazırlanma amacına uygun olması için, süreç ve sonucun incelenmesidir (Csizmadia vd., 2015).

Otomasyona Dönüřtürme: Otomasyon, tekrarlanması gereken iřlemlerin bilgisayar veya makinelere yaptırılmasıdır. BİD alt boyutu olarak otomasyona dönüřtürme ise algoritmik düşünme çerçevesinde çözümlerin otomatikleřmesinden bahsedilmektedir (Çetin ve Uçar, 2017). Tek bir örneęi nasıl çözeceęimizi öğrendikten sonra, çözümümüzü çok çeřitli örneklere uygulamak için otomasyon kavramını kullanırız (Hromkovič, Kohn, Komm, ve Serafini, 2016).

Problemleri Bileřenlerine Ayırma: Problemi veya durumu oluřturan bileřenlerin daha küçük parçalara bölmektir. Parçalar daha sonra ayrı ayrı anlamaya çalışılır, çözülür, geliřtirilir ve deęerlendirilir. Bu karmařık problemlerin çözülmesini, yeni durumların daha iyi anlaşılmasını ve büyük sistemlerin tasarlanmasını kolaylařtırır (Csizmadia vd., 2015).

Genelleme: Önceki sorunlara dayanarak yeni sorunları hızla çözmenin ve önceki deneyimleri temel almanın bir yoludur. Kalıpların, benzerliklerin ve baęlantıların tanımlanması ve bu özelliklerin kullanılması ile iliřkilidir (Csizmadia vd., 2015).

Soyutlama: Bir ürünü gereksiz detaylarından arındırarak daha anlaşılır hale getirme sürecidir (Yeni, 2018). Wing (2010) bilgi işlemsel düşünmede en önemli ve yüksek seviyedeki düşünce sürecinin soyutlama süreci olduğunu vurgulamıştır. Wing (2006) bu beceri ile hedeflenen noktanın; bir bilgisayar bilimcisi gibi soyutlama becerisini kullanarak çok boyutlu düşünme becerisini ortaya çıkarmak olduğunu ifade etmiştir. Grover ve Pea (2013) ise soyutlama sürecinin, BİD ile diğer düşünme becerileri arasındaki en önemli fark olduğunu belirtmiştir.

2.2. Bilgi İşlemsel Düşünme(BİD) ile İlgili Müfredat Bilgileri, Etkinlik Örneği

2.2.1. Müfredatımızda Bilgi İşlemsel Düşünme Becerisi ve Programlama

Bilgi işlemsel düşünme becerisini eğitim müfredatlarına entegre eden ülke sayısı her geçen gün artmaktadır. Bocconi, Chiocciariello, Dettori, Ferrari ve Engelhardt'ın (2016) araştırmasında görüldüğü gibi Türkiye BİD becerilerine yönelik kazanımları müfredatlarına entegre etmiş ülkelerdendir. Türkiye’de zorunlu eğitim kapsamına giren 1-12. Sınıflar arasındaki tüm kademelerde BİD becerisine yer verildiği görülmektedir.

Türkiye’de ilkokul 1, 2, 3 ve 4. sınıflar için 2018 yılında hazırlanan Bilişim Teknolojileri Ve Yazılım Dersi Öğretim Programında BİD becerilerine yönelik ifadeler bulunmaktadır. Öğretim programı özel amaçları başlığı altına “Öğrencilerin, problem çözme ve bilgi işlemsel düşünme becerileri kazanmaları amaçlanmaktadır.” ifadesi bulunmaktadır. Öğrenciye kazandırılması hedeflenen beceriler başlığı altında ise “Bilgi İşlemsel Düşünme” becerisi bulunmaktadır. Bunun yanı sıra öğrencilere kazandırılmak istenen birçok kazanım bulunmaktadır. Problem Çözme ve Programlama Teması altındaki 41 kazanım cümlesi BİD becerisini öğrencilere kazandırmaya yöneliktir (MEB, 2018a). Ülkemizdeki ilkokullarda uygulanan bu program sınıf öğretmenleri tarafından serbest etkinlik saatlerinde -öğretmen tercihine göre- uygulanabilmektedir. Etkinlikler sınıf ortamında ve bilgisayarsız bilgisayar bilimi (B3) şeklinde yapılmaktadır. İlkokul 1, 2, 3 ve 4. sınıflarda uygulanan bilişim teknolojileri ve yazılım dersi öğretim programındaki programlama bilgisi ve BİD becerilerine yönelik kazanımlardan örnekler Tablo 1’de verilmiştir.

Tablo 1 : MEB – İlkokul 1, 2, 3 ve 4. Sınıflarda Uygulanan Bilişim Teknolojileri ve Yazılım Dersi Öğretim Programındaki Programlama Bilgisi ve BİD Becerilerine Yönelik Örnek Kazanımlar

Kazanım Numarası	Kazanım İfadesi
BT.5.D2.5.	Bir problemi alt problemlere bölerek gösterir.
BT.5.D2.8.	Bir problemin çözümü için algoritma oluşturma.
BT.5.D3.2.	Bir problemi çözebilmek için gerekli verileri ifade eder.
BT.5.D4.8.	Karmaşık işlemler için algoritma tasarlar.
BT.5.D4.9.	Hazırladığı algoritmada hataları ayıklar.

Türkiye’de ortaokul 5. ve 6. sınıflarda zorunlu ders kapsamında okutulan Bilişim Teknolojileri ve Yazılım dersi öğretim programında da BİD becerilerine yönelik ifadeler bulunmaktadır. Öğretim programı özel amaçları başlığı altına “Öğrencilerin, Problem çözme ve bilgi-işlemsel düşünme becerileri edinmeleri ve geliştirmeleri amaçlanmaktadır.” ifadesi bulunmaktadır. Öğretim programının uygulanması aşamasında dikkat edilmesi gereken hususlarda “her öğrenciye bilgi-işlemsel düşünme becerisi kazanma fırsatı sağlanmalıdır” diye belirtmiştir. Bunun yanı sıra öğrencilere kazandırılmak istenen tüm kazanımların %50’si BİD ile ilgilidir. Problem Çözme ve Programlama Teması altındaki 50 kazanım cümlesi BİD becerisini öğrencilere kazandırmaya yöneliktir (MEB, 2018b). Ortaokul 7. ve 8. sınıflarda ise seçmeli olarak okutulmaya devam eden derste 15 kazanım cümlesi BİD becerilerini öğrencilere kazandırmaya yöneliktir (MEB, 2018c). Ortaokul 5, 6, 7 ve 8. sınıflarda uygulanan bilişim teknolojileri ve yazılım dersi öğretim programlarındaki programlama bilgisi ve BİD becerilerine yönelik kazanımlardan örnekler Tablo 2’de verilmiştir.

Tablo 2 : MEB – Ortaokul 5. 6. 7. ve 8. Sınıflarda Uygulanan Bilişim Teknolojileri ve Yazılım Dersi Öğretim Programlarındaki Programlama Bilgisi ve BİD Becerilerine Yönelik Örnek Kazanımlar

Kazanım Numarası	Kazanım İfadesi
BT.5.5.1.6.	Problemi çözmek için gerekli değişken, sabit ve işlemleri açıklar.
BT.6.5.1.9.	Problemin çözümünü benzer problemler için geneller.
BT.6.5.2.14.	Farklı programlama yapılarını kullanarak karmaşık problemlere çözüm üretir.
SBT.7.3.2.4.	Verilen bir sözdizimini test eder ve hataları ayıklar.
SBT.8.3.2.4.	Kullanıcı girdilerini alarak en uygun veri yapısında saklar.

Türkiye’de liselerde 2 kur şeklinde okutulan Bilgisayar Bilimi dersi öğretim programında da BİD becerilerine yönelik ifadeler bulunmaktadır. Öğretim programı özel amaçları başlığı altına “Öğrencilerin, problem çözme ve bilgi-işlemsel düşünme becerileri edinmeleri ve geliştirmeleri amaçlanmaktadır.” ifadesi bulunmaktadır. Öğrenciye kazandırılması hedeflenen özel beceriler başlığı altında ise “Bilgi İşlemsel Düşünme” becerisi bulunmaktadır. Bu bölümde BİD alt boyutları ile ele alınmıştır. Bunun yanı sıra öğretim programında bulunan kazanımların yaklaşık olarak yarısı BİD ile ilgilidir (MEB, 2018d). Liselerde 2 kur şeklinde uygulanan bilgisayar bilimi dersi öğretim programındaki programlama bilgisi ve BİD becerilerine yönelik örnek kazanımlar Tablo 3’te verilmiştir.

Tablo 3 : MEB – Liselerde Uygulanan Bilgisayar Bilimi Dersi Öğretim Programındaki Programlama Bilgisi ve BİD Becerilerine Yönelik Örnek Kazanımlar

Kazanım Numarası	Kazanım İfadesi
1.2.1.4.	Bir problemin çözüm adımlarının doğru bir şekilde belirlenmesinin ve sıralanmasının önemini ifade eder.
1.2.2.4.	Veri türlerini problemin çözümünde kullanır.
1.2.3.2.	Verilen problemi alt problemlerine böler.
1.2.3.4.	Algoritmayı analiz ederek sonucunu yordar.
1.2.6.6.	Problem çözme süreçlerinde karar yapılarını kullanarak akış şeması oluşturur.

2.2.2. Bilge Kunduz (Uluslararası Enformatik ve Bilgi İşlemsel Düşünme Etkinliği)

Bilge Kunduz, bilgisayar bilimini ve bilgi işlemsel düşünmeyi her yaştan öğrenciye öğretmek amacı ile oluşturulmuş, bu konudaki farkındalıkları artırırken eğlendirmeyi de önemseyen uluslararası bir etkinliktir. Bu etkinlik kapsamında, birçok ülke aynı dönemde öğrencilerin enformatik ile ilgili yetenekleri test eden çevrimiçi bir etkinlik düzenlenmektedir.

“Etkinlikteki, kısa sorulara Bilge Kunduz görevleri denmektedir. Bu görevler, enformatik konusunda hiçbir ön bilgisi olmayan kişiler tarafından yanıtlanabilir. Bu görevleri çözebilmek için öğrenciler bildiklerini gözden geçirmeli, hesaplama yapmalı, karar vermeli, neden-sonuç ilişkisi kurmalı, analitik düşünme ve problem çözme gibi üst düzey düşünme becerilerini kullanmalıdır. Bilge Kunduz etkinliği, sadece genç öğrencilere değil, tüm insanlara ulaşmayı hedeflemektedir. Herkesin her türlü teknolojik cihaza sahip olduğu günümüzde, insanları enformatik konusunda bilinçlendirmek son derece önem teşkil

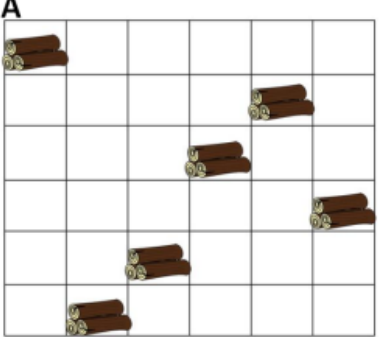
etmektedir. Etkili teknoloji kullanımının yanı sıra bilimsel olarak insanların teknoloji deneyimlerini arttırmak bilişim konusunda çalışanların hedefi olmalıdır. 2018 yılında ülkemizde 67.897 öğrenci bilge kunduz etkinliğine çevrimiçi olarak katılmıştır.” (Bilge Kunduz, 2019).

Ülkemizde “Bilge Kunduz” olarak adlandırılan etkinliğin uluslararası adı “Bebras” olarak bilinmektedir.

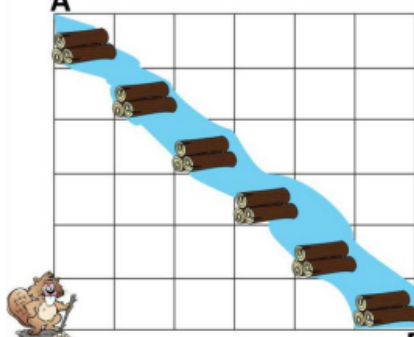
Bebras etkinliği, her yaştan okul öğrenciler arasında bilgisayar bilimini ve BİD’i geliştirmeyi amaçlayan uluslararası bir girişimdir. Bebras, öğrencilerin karmaşık görevleri basit bileşenlere ayırma, algoritma tasarımı, örüntü tanıma, örüntü genelleştirme ve soyutlama dahil olmak üzere problem çözme becerilerini geliştirmelerini ve bilişim kavramlarını öğrenmelerini teşvik eder (Bebras.org, 2019). 2018 yılında dünya genelinde 50 ülkeden 2.5 milyona yakın öğrenci bu etkinliğe çevrimiçi olarak katılmıştır. Şekil 2’de örnek bir bilge kunduz sorusu görülmektedir.

Baraj İnşaatı

Mühendis kunduz, evini kışın olası sellerden korumak için bir baraj yapmayı düşünmektedir. Şekil-1’de gösterilen ağaç yığınlarını kullanarak Şekil-2’deki gibi bir baraj inşa etmek istemektedir. Ağaç yığının dikey olarak taşınması için 1 saat, yatay olarak taşınması için 2 saat süre gerekmektedir.



Şekil-1



Şekil-2

Soru

Mühendis kunduzun barajı inşa edebilmesi için gereken süre en az kaç saattir?

A) 16
B) 11
C) 14
D) 12

Şekil 2: Bilge Kunduz Etkinliği Örnek Sorusu

2.3. Programlama ve Programlama Öğretimi

Programlama, bir bilgisayarın gerçekleştirmesi için talimatlar yazmak, daha detaylı bir anlatımla; bilgisayarlar tarafından çalıştırılmak üzere, özel bir dil kullanarak algoritma oluşturma ve onu sözdizimi haline getirmektir.

Bilgisayar programlama genellikle “kodlama” olarak nitelendirilmekte olduğu görülmektedir ancak çok daha geniş bir beceri kümesini ifade etmektedir. Bu beceriler; yazılım geliştirme, analiz, tasarım, kodlama, test etme ve hata ayıklama gibi ögelere ayrılabilir (Bell, 2018). Bu düşünceye katkı sunan Lye ve Koh (2014) programlamanın, kodlamanın ötesinde bir şey olduğunu belirterek, soyutlama, ayrıştırma, problem çözme içeren bilgi işlemsel düşünme becerisini kullandıran karmaşık bir süreç olduğunu belirtmiştir. Bell (2018) programlamanın bazı insan merkezli nitelikleri ve disiplinleri de içerdiği için basitçe “kodlama” olarak nitelendirilememesi gerektiğini eklemiştir. Programlamanın gerektirdiği insan merkezli nitelik ve disiplinleri ise şöyle açıklamıştır: yazılımın geliştirileceği bir durumun analizi insanlarla çok iyi iletişim kurmayı gerektirir; tasarım, yaratıcılığı içerir; test etme titizlik gerektirir ve hata ayıklama sebat gerektirir.

Şendurur (2017) programlama etkinliklerinin öğrencilere bazı zihinsel becerileri kazandırmak için yüksek potansiyele sahip olduğunu belirtmiştir. Buna ilave olarak programlama etkinliklerinin tek kazanım çıktısının “programlama bilgisi” olmaması gerektiğini, yüksek potansiyeli kullanmak için farklı kazanımlar elde etmek için yeni yöntemlere ihtiyaç olduğunu belirtmiştir. Ayrıca bilgisayar programlama veya kodlama eğitimi öğrencilerdeki BİD becerisini geliştirmek amaçlı da kullanıldığı aktarılmıştır (Calao, Moreno-Leon, Correa ve Robles, 2015). Bu görüşü destekleyip alana katkı sağlamak adına, bu çalışmada programlama etkinliklerinin programlama bilgisine olan etkisinin yanı sıra BİD ve alt boyutlarına ait kazanımların edinilmesine etkisine bakılmıştır. Bu etkilere bakılırken farklı programlama dilleri ve yöntemleri kullanılmıştır. Çalışmada kullanılacak programlama dili ve yöntemi; ihtiyaçlara, programlayanın yaşına, ön bilgisine ve programlama amacına göre belirlenmiştir. Bu çalışmada kullanılacak programlama dillerine ve yöntemlerine yönelik açıklamalar aşağıda verilmiştir.

2.3.1. Blok Tabanlı Programlama

2.3.1.1. Blok Tabanlı Programlama Nedir?

Blok tabanlı programlama, küçük yaştaki öğrencilerin geleneksel programlama dillerinin karmaşık kod yapılarını (sözdizimi ve noktalama işaretleri) öğrenmelerine gerek kalmadan, programlama blokları ile uygulamalar yazabilmelerini sağlamaktadır (Resnick vd., 2009). Öğrenciler, sözdizimi (syntax) yerine görsel ağırlıklı kod bloklarını sürükle-bırak mantığı ile bir araya getirerek programlama deneyimi yaşamaktadır.

2.3.1.2. Blok Tabanlı Programlama Öğretim Araçları

Günümüzde programlama eğitiminde birçok blok tabanlı programlama aracı kullanılmaktadır. Bunlar: Code.org, Scratch, Scratch Jr., Alice, App Inventor, Google Blockly, Kodu Game vb. araçlardır. Bu çalışmada ise blok tabanlı programlama aracı olarak code.org tercih edilmiştir.

Code.org internet sitesi 2013 yılından beri her yaştan öğrenciye bilgisayar bilimleri ve programlama mantığını öğretmek için oluşturmuş bir platformdur. Blok tabanlı programlama mantığı ile çalışmaktadır. Kâr amacı bulunmayan platform vizyonlarını şöyle ifade etmektedir: “Her okuldaki her öğrencinin tıpkı biyoloji, kimya veya cebir gibi bilgisayar bilimleri öğrenme fırsatını sağlamak”. Code.org, Amazon, Facebook, Google, Infosys Vakfı, Microsoft vb. firmalar tarafından desteklenmektedir (Code.org, 2019).

2.3.1.3. Blok Tabanlı Programlama ve BİD

Blok tabanlı programlama araçlarının dünyadaki ve ülkemizdeki örnekleri ele alındığında genellikle 6-12 yaş seviyesi arasında kullanılmaktadır. Bu araçların sadece öğrencilere programlama yapmayı öğretmesi dışında farklı amaçları da bulunmalıdır. Aksi takdirde Harel’in (2016) “pop computing” olarak adlandırdığı ve günümüzde dünya genelinde çok büyüyen blok tabanlı programlamanın yüzeysel olduğu eleştirisi doğrulanmış olacaktır. Popülerliğin getirdiği yüzeysellik yolundan çıkış reçetesini Scratch’in (blok tabanlı programlama aracı) geliştiricisi Resnick (2013) “kodlamayı öğrenme değil öğrenmek için kodlama” ifadesi ile vermektir. Bu ifadesine açıklama olarak; öğrencilerin kodlama yaparken; matematik, bilgi işlemsel fikirler, problem

çözme yöntemler, projeler tasarlamak, fikir alış verişinde bulunmak gibi birçok beceriyi öğrendiklerini belirtmektedir. Bu becerilerin sadece bilgisayar bilimcilere değil, yaş, ilgi ve meslek ayırt edilmeksizin herkese lazım olduğunu ifade etmektedir (Resnick, 2013). 21. yüzyıl için çok önemli olan bu becerileri ders kazanımlarına ekleyip uygulayacak eğitimcilere de bazı görevler düşmektedir. BİD ile ilgili alanyazında oluşturulmuş çerçevelerden faydalanan bir eğitimci blok tabanlı programlama araçlarını sadece öğrencilere program yazdırmak için kullanılan bir araç olarak görmeyecektir. Bu etkinlikler vasıtasıyla BİD ile ilgili kavram ve becerileri kazandırma amacını da ön planda tutacaktır (Çetin ve Uçar, 2017). Yükseltürk ve Altıok (2017) blok tabanlı programlama araçlarının sahip oldukları özellikler sebebiyle, problem çözme ve BİD becerilerini geliştirmek için uygun ortamları sağladığını belirtmiştir.

Feldhausen, Weese ve Bean (2018) çalışmalarında iki ayrı gruba farklı programlama yöntemleri ile eğitim vermişlerdir. Farklı programlama yöntemlerinin öğrencilerdeki BİD ve alt boyut becerilerine yönelik öz yeterlik algılarındaki etkisini gözlemlemişlerdir. 5. ve 6. sınıf öğrencilerden oluşan gruba bilgisayarsız etkinlikler ve blok tabanlı programlama (Scratch) ile eğitim vermişlerdir. 7, 8 ve 9. sınıf öğrencilerinden oluşan bir diğer gruba ise fiziksel programlama platformu Arduino üzerinde scratch (blok tabanlı programlama) ve Arduino'nun kendisine ait yazılımı ile metin tabanlı programlama eğitimi verilmiştir. Eğitim sonunda her iki gruptaki öğrencilerin BİD ve alt boyut becerilerine yönelik öz yeterlik algı seviyelerinde yükselme gözlemlenmiştir. Bu çalışma bilgisayarsız etkinliklerin, blok tabanlı programlamanın ve metin tabanlı programlamanın BİD ve alt boyut becerilerine yönelik öz yeterlik algısına olumlu etkisi olduğunu göstermektedir.

Başka bir çalışmada Sakamoto, Takano, Washizaki ve Fukazawa (2013) Japonya'da üniversiteye giden kız öğrencilerle uyguladığı blok tabanlı programlama eğitiminin kız öğrencilerin BİD becerisine ve programlama bilgisine olumlu etkiye sahip olduğunu belirtmiştir. Türkiye'de 5.sınıf 31 öğrencinin katıldığı araştırmada blok tabanlı programlama aracı Scratch eğitimi sonunda programlama becerileri ile bilgi işlemsel düşünme becerileri arasında yüksek düzeyde anlamlı bir ilişki olduğu görülmüştür. Cinsiyete göre ne programlama bilgisi değerlendirmesinde ne de BİD ölçeğinde anlamlı farklılık görülmemiştir (Oluk & Korkmaz, 2016). Blok tabanlı programlama eğitimi ile BİD arasındaki ilişkinin incelendiği başka bir çalışmada Hsu (2014) 46 ilkokul

öğrencisi ile blok tabanlı programlama yazılımı (scratch) ile oyun tasarlamışlardır. Bu çalışmanın sonuçlarına göre, erkek gruplarla kız grupları arasında bilgi işlemsel kavramların toplam puanında anlamlı bir fark görülmemiştir. Kod bloklarını kullanma konusunda da cinsiyetler arasında fark görülmemiştir. Cinsiyetler arasında anlamlı olarak görülen tek fark sayaçlı döngü yapılarının kızlar tarafından daha iyi anlaşıldığıdır. Cinsiyete göre büyük farklılıklar görülmeyen bu çalışmaları destekleyen ama erken yaşlarda programlama eğitime başlanılmaması durumunda bilgisayar bilimlerindeki başarı ve BİD becerileri seviye farklarının erkek öğrenciler lehine artacağını belirtmiştir. Çalışmanın detayları şu şekildedir; İtalya’da yapılan bir araştırma iki yıllık bir projenin sürecini ve sonucunu analiz etmektedir. Code.org internet sitesi ile desteklenmiş eğitim materyali ile uygulanan eğitimin bilgisayar bilimleri konusundaki farkındalığa etkisi incelenmiştir. Proje ile ilgili nicel ve nitel verilerin detaylı bir analizi sunulmuş ve iyileştirme alanları belirlenmiştir. Araştırma sonuçları bilgisayar bilimleri kabulünün, öğrenciler küçükken cinsiyete göre daha yüksek bir bağımsızlığa sahip olduğunu ima ederek, öğrencileri erken yaşta bilişim ile tanışmasının önemini desteklediğini belirtmektedir (Corradini, Lodi ve Nardelli, 2017). Etkinliğe katılan öğrencilerde yaş ilerledikçe cinsiyetler arasındaki farkın erkekler lehine arttığı görülmüştür. Bu sebeple bilgisayar bilimlerine -özellikle birçok düşünme becerisini destekleyen- programlama eğitime erken yaşlarda başlamanın önemine dikkat çekilmiştir.

BİD’e ait alt boyutlarından olan algoritmik düşünme becerisini öne çıkaran bir çalışmada Wong ve Jiang (2018), programlama öğretiminin, 5.sınıf öğrencilerinin algoritmik düşünme ve hata ayıklama becerileri üzerindeki etkisini araştırmıştır. Bilgisayarsız etkinlikler ve blok tabanlı programlama faaliyetlerinin bir arada kullanıldığı karma bir yaklaşımla, algoritmik düşünme ve hata ayıklama becerilerinin geliştiği gözlemlenmiştir. İtalya’da 81 ilkokul dördüncü sınıf öğrencisi ile yapılan başka bir çalışmada “Kodu Game” isimli blok tabanlı programlama uygulaması kullanılmıştır. Çalışmanın sonunda BİD becerilerinde ve bilgisayar programlamanın algılanması konusunda gelişmeler gözlemlenmiştir. Aynı çalışmada kız öğrencilerin erkek öğrencilere göre programlamada daha olumlu sonuçlar aldığını belirtmiştir (Chiazzese, Fulantelli, Pipitone, ve Taibi, 2018). Bu çalışmalar blok tabanlı programlama eğitiminin öğrencilerin BİD becerilerini geliştirebileceğini göstermektedir.

İşbirlikli bir şekilde blok tabanlı programlamanın BİD'e olan etkisine bakılan bir araştırmada, Denner, Werner, Campe ve Ortiz (2014) blok tabanlı programlama aracı (Alice) kullanarak yürüttükleri çalışmaya 320 ortaokul öğrencisi katılmıştır. Bireysel veya ikili programlama yapan öğrencilerin BİD ve programlama puanları karşılaştırılmıştır. Çalışma sonuçlarına göre ikili programlama yapan öğrencilerin bireysel programlama yapan öğrencilere göre daha yüksek puan aldıkları belirlenmiştir. İkili programlama yapmanın, bireysel olarak programlama yapmaya göre, BİD becerisi ve özellikle de az deneyimli öğrenciler arasında programlama bilgisi oluşturmak için avantajlı olduğu görülmüştür.

Blok tabanlı programlama araçlarının -özellikle programlamaya yeni başlayanlar için- bilgisayar bilimleri başarısı, ilgi, tutum, motivasyon vb. değişkenler üzerinde olumlu etkilerini gösteren çalışmalar vardır. 10-14 yaş arası öğrencilerle önce B3 uygulamaları ile ardından blok tabanlı programlama ile çalışmalar yaptırılmış, eğitsel ve motivasyon açısından olumlu sonuçlar gözlemlenmiştir (Gardeli ve Vosinakis, 2017). Malan ve Leitner'in (2007) bu olumlu etkiyi destekleyen ve blok tabanlı programlama araçlarının öğrenmeyi kolaylaştırdığını, başarı ve motivasyonu arttırdığını gözlemledikleri bir çalışmaları mevcuttur. Blok tabanlı programlama araçlarının programlamaya yeni başlayanlara olan en önemli faydası ise; java, python gibi metin tabanlı programlama dillerine geçişte kullanılabilecek bir araç olmasıdır. Bilgisayar bilimlerine ilk defa giriş yapan öğrencileri motive eden ve olası sözdizim hatası olmadan programlamaya giriş yapmalarını sağlar (Dorling ve White, 2015; Malan ve Leitner, 2007). Bahsi geçen araştırmalarda görüldüğü gibi her zaman küçük yaşlarda değil üniversite düzeyinde de blok tabanlı programlama ile ilgili araştırmalar yapılmaktadır. Tayvan'da 158 üniversite birinci sınıf öğrencisi ile yapılan araştırmada blok tabanlı programlama eğitiminin BİD becerisini ve alt becerilerini geliştirdiğini göstermektedir (Lin, Chao, Lin & Tzeng, 2018)

Blok tabanlı programlama araçlarının çeşitli beceriler üzerinde anlamlı bir etki oluşturmadığını belirten çalışmalar da bulunmaktadır. Türkiye'de 55 ortaokul öğrencisi ile yapılan çalışmada scratch (blok tabanlı programlama) eğitiminin BİD beceri seviyesine etkisi araştırılmıştır. Çalışmanın sonunda öğrencilerin BİD becerilerinde istatistiksel olarak anlamlı bir farklılık oluşmamıştır. Cinsiyete göre de BİD becerilerinde bir farklılık görülmemiştir (Uslu, Mumcu ve Eğin, 2018). Ülkemizde

yapılan bir diğerk çalışmada, 5.sınıfta okuyan 49 öğrenci bulunmaktadır. Blok tabanlı programlamanın (scratch), ortaokul öğrencilerinin problem çözme becerilerinde önemli bir farklılık yaratmadığı görölmüştür (Gölbahar ve Kalelioğlu, 2014). İrlanda’da pilot uygulaması yapılan BİD ve programlama odaklı bilgisayar bilimi projesinde bilgisayarsız etkinlikler ve blok tabanlı programlama uygulaması scratch ile 10 haftalık bir eğitim verilmiştir. Eğitime katılan 22 öğrenciye eğitim başında ve sonunda Bebras (Uluslararası enformatik ve BİD etkinliğı) sorularından oluşan test yapılmıştır. Test puanlarına ait analiz sonuçlarında sontest puanlarının öntest puanlarına göre düşük olduğu görölmüştür. Araştırmacılar bu olumsuz sonuca varılmasını; öğrenci devamsızlığının fazlalığına, pilot uygulamanın kısa zamanda yoğun olarak yapılmasına ve öntest ve sontest zorluklarının farklı olma ihtimaline bağlamışlardır (Lockwood ve Mooney, 2017). Adleberg (2013) ise 9-12 yaş arası 98 öğrenci ile yaptığı çalışmada, blok tabanlı programlama (scratch) çalışması sonunda, programlamaya karşı tutumları, içsel ve dışsal motivasyonları arasında cinsiyete bağlı olarak bir fark bulunmamıştır.

2.3.2. Metin Tabanlı Programlama

2.3.2.1. Metin Tabanlı Programlama Nedir?

Metin tabanlı programlama, programlama işleminin sözdizimi ile yapılmasına denir. Sözdizimi, bir programlama diline ait dilbilgisi kuralları anlamına gelir ve Türkçe, İngilizce veya başka dilde olan dilbilgisi kurallarına benzer. Yine de bu benzerlik içinde büyük bir farkı barındırır. Bu fark, bilgisayarların bu dilbilgisini veya sözdizimini nasıl yapılandırıldığı konusunda gerçekten titiz olmasıdır. Bilgisayarlar, yalnızca bilgisayarın beklediğı biçimde yazdığınızda ne yazdığınızı anlayan esnek olmayan makinelerdir. Yani metin tabanlı programlama dilleri ile hazırlanan program eğer sözdizimsel olarak doğruysa çalışır; eğer sözdizimi hatası varsa işlem başarısız olur ve bir hata mesajı ile programlama yapan kişiyi uyarır. Bu durum programlamaya yeni başlayanlar için bir zorluk olarak görölmektedir. Özellikle erken yaşlarda programlama eğitimi ile ilgili çalışmalarda bu zorluklardan bahsedilmektedir (Mladenović, Boljat ve Žanko, 2018; Kelleher, C., & Pausch, R., 2005).

2.3.2.2. Metin Tabanlı Programlama Öğretim Araçları

C, C++, Java, Python, PHP, Javascript, Kotlin, Basic vb. birçok programlama dili bulunmaktadır. Bu araştırmada metin tabanlı programlama eğitimi alan gruptaki öğrencilerin ilk programlama deneyimi olması sebebiyle başlangıç seviyesine uygun bir dil seçilmeye çalışılmıştır. Small Basic'in arayüzü, özellikleri ve hitap ettiği yaş seviyesi göz önüne alındığında araştırma amacına en uygun metin tabanlı programlama ortamı sağlayacağına karar verilmiştir.

Small Basic, programlamayı yeni başlayanlar için erişilebilir ve kolay hale getirmeye odaklanan bir projedir. Üç ayrı parçadan oluşur:

- Dil
- Programlama Ortamı
- Kütüphaneler (Small Basic, 2019)

Small Basic, BASIC isimli gelişmiş programlama dilinin temel özelliklerini içeren bir versiyonudur. Blok tabanlı kodlamadan metin tabanlı kodlamaya geçişi sağlamak için öğrencilere özel olarak oluşturulan tek programlama dilidir (Small Basic, 2019).

2.3.2.3. Metin Tabanlı Programlama ve BİD

Programlama eğitiminin önem kazandığı günümüzde, bu eğitim küçük yaşlarda genellikle blok tabanlı ve görsel ağırlıklı olarak verilmektedir. Öğrenciler için bu seviyenin yeterli olacağı düşünülmektedir. Ancak yapılan çalışmalar ilkokul ve ortaokul öğrencilerinin metin tabanlı programlama dilini de öğrenebilmelerinin mümkün olduğunu göstermiştir (Dorling ve White, 2015; Park ve Yoo, 2018). Yine de metin tabanlı programlama eğitime başlamadan önce blok tabanlı programlama ile temel becerileri kazandırmanın faydalı olacağı aktarılmıştır (Tabet, Gedawy, Alshikhabobakr ve Razak, 2016).

Feldhausen vd. (2018) yaptıkları çalışmada 7, 8 ve 9. sınıfa giden bir grup öğrenciye önce scratch yazılımı ile blok tabanlı programlama ardından fiziksel programlama platformu Arduino ve yazılımı üzerinde metin tabanlı programlama yaptırmışlardır. Eğitime katılan öğrencilerin BİD becerilerine yönelik öz yeterlik algısında olumlu etki gözlemlenmiştir. Başka bir çalışmada üniversitede okutulan “bilgi işlemsel düşünmeye

giriş” isimli dersten ve bu derste basit metin tabanlı programlama (python), bilgi işlemsel düşünme ve bilgisayar bilimi isimli üç üniteden bahsedilmiştir. Dersin öğrenciler üzerine olumlu etki oluşturduğu ve bu amaçla açılan bir ders için Python’un (metin tabanlı programlama) mükemmel bir başlangıç dili olduğunu belirtmiştir (Hambruch, Hoffmann, Korb, Haugan & Hosking, 2009). Şili’de lise öğrencileri ile yapılan çalışmada C++ (metin tabanlı programlama dili) ile verilen eğitimlerin öğrencilerin BİD becerilerini geliştirdiği aktarılmaktadır. Bu olumlu sonuçla beraber ilk defa programlama eğitimi alan öğrencilerin C++ ile programlama yaparken bazı zorluklarla karşılaşabileceğini ifade etmiştir (González, López, ve Castro, 2018).

Literatür taramasında metin tabanlı programlama ve BİD ile ilgili çalışmaların diğer programlama yöntemlerine göre daha az olduğu görülmektedir.

2.3.3. Robotik Programlama

2.3.3.1. Robot ve Robotik Nedir?

Robot” kelimesi ilk defa Çekoslovak yazar Karel Capek tarafından 1921 tarihli Rossum’s Universal Robots (R.U.R. Rossum’un Evrensel Robotları) isimli tiyatro eserinde kullanılmıştır. Bu kelimenin tiyatro oyunundaki anlamı “zorla çalıştırılan işçi”dir (Capek, 2013). Bu tiyatro eserinde kullanımının ardında önce slav dillerine ardından İngilizceye geçmiştir.

Robot kavramı ilk ortaya çıkmasından bu yana farklı tanımlamalar yapılmıştır. Günümüzde robotik alanındaki gelişmeler sonucu varılan noktada yapılan bazı tanımlamalar şu şekildedir.

“Robot, sahip olduğu bilgisayarın programlanmasıyla karmaşık bir dizi eylemi otomatik olarak gerçekleştirebilen bir makinedir.” (Oxford Sözlük, 2019). TDK (2019) tarafından robot, “Belirli bir işi yerine getirmek için manyetizma ile kendisine çeşitli işler yaptırılabilen otomatik araç” olarak tanımlanmıştır. İnsanlığın yararına gelişen teknoloji konusunda söz sahibi önemli kuruluşlardan olan The Institute of Electrical and Electronics Engineers (IEEE) robot kelimesini, “ortamını algılayabilen, karar vermek için hesaplamalar yapabilen ve gerçek dünyada eylemler gerçekleştirebilen otonom bir makinedir.” şeklinde tanımlamıştır (IEEE, 2019).

Robot kavramı kelime olarak son yüz yılı kapsar gibi gözükse de robot sayılabilecek otonom makinelerin tarihi yaklaşık 800 yıl önceye kadar gitmektedir. 1200'lü yıllarda Anadolu'da yaşamış bir mühendis olan El Cezeri'nin "Kitab'ül Hiyel" isimli kitabında izah ettiği otonom makineler ve robotlar bulunmaktadır. El Cezeri'nin kitabında tüm detaylarıyla izah ettiği düzenekleri sadece hayal etmediği imal edip çalıştırdığı da anlaşılmaktadır (Çalışkan, 2019). Bu yönleri ile El Cezeri robotiğin kurucusu olarak bilinmektedir. El Cezeri'den yaklaşık 250 yıl sonra Leonardo Da Vinci tarafından da otonom makineler ve robotlar tasarlandığı bilinmektedir (Leonardo Robot Society, 2019).

Robotik kelimesi ise yazar İsaac Asimov tarafından yazılan "Ben Robot" kitabının içinde bulunan "Yalancı" isimli kısa hikayesinde ilk olarak kullanılmıştır (Asimov, 2018). Bu kavram "Robotların tasarımı, yapımı, işletimi ve uygulamasıyla ilgilenen teknoloji dalı" olarak açıklanmaktadır (Oxford Sözlük, 2019).

2.3.3.2. Eğitsel Robotlar

Üçgül (2017) robotların günümüzde endüstri, medikal, uzay, eğitim, eğlence ve yarışmalar gibi birçok alanda kullanıldığını belirtmiştir. Bahsedilen tüm kullanım alanlarında olduğu gibi eğitim alanında da robotların gelişimi devam etmektedir. Bu hızlı gelişim süreci eğitsel robotların eğitimdeki yeri ile ilgili araştırmaların yapılmasını sağlamaktadır.

Robotların eğitsel amaçlarla kullanım fikri Papert'a aittir. Papert (1980) ve Papert (1996) çalışmalarında eğitimde ilk defa robotlardan faydalanmıştır. Eğitsel robot Logo ve Papert'ın alanyazına kazandırmış olduğu inşacılık kuramı ilk olarak bu çalışmalarda detaylarıyla açıklanmıştır.

Günümüzde yerli yabancı birçok firma eğitsel robotlar üretmektedir. Genel olarak bu robotların amaçları kullanıcıların programlama ve mekanik bilgisini geliştirmektir. Bunu yaparken öğrenciler arasında işbirliğini ve iletişimi, problem çözme becerilerini, eleştirel düşünme becerilerini ve yaratıcılığı artırabilen bir öğrenme ortamı oluşturabilir (Eguchi, 2015).

Günümüzde tüm dünyada tercih edilen bazı eğitsel robotlar şöyledir:

- Robotis firmasına ait; Play, Dream, Smart, Stem

- Lego firmasına ait ; Wedo 2.0, Spike, Mindstorms EV3
- MakeBlock firmasına ait; Mbot, Neuron, Codey Rocky

Araştırmadaki robotik programlama eğitimi alan deney grubu Robotis firmasına ait Robotis Dream eğitsel robotik seti ile çalışmıştır. Bu sebeple eğitimde kullanılan eğitsel robotik seti ile ilgili bilgiler verilmiştir.

Robotis 1999 yılında Güney Kore’de kurulmuştur. Kendi internet sitelerinde bulunan açıklamalarında isimlerinin “Robot nedir?” sorusuna basit bir cevaptan türetildiğini ve o zamandan beri birçok alan için robotlar ürettiklerini belirtmektedirler. Çocukların hayallerini gerçekleştirmeleri için de eğitsel robot setleri ürettiklerini eklemişlerdir (Robotis, 2019).

Robotis Dream, çocukları ile defa robotik ile tanıştırmak için hazırlanan ve çeşitli seviyeleri olan bir eğitsel robot setidir. Robotis Dream Seviye 1, elektriksel güç, yerçekimi, kuvvet, 2 ayaklı ve 4 ayaklı robotlarda hareket gibi temel fizik, elektronik ve robotik prensiplerini giriş seviyesinde veren bir robot kitidir. Robotis Dream Seviye 2, bir robotun nasıl hareket ettiğinin, sensörlerin kullanımı, hız ve kuvvet, yürüyüş ve sürüş gibi robotiğin temellerini öğretir (Robotis Dream, 2019).

2.3.3.3. Robotik Programlama ve BİD

Programlamanın somut elektronik cihazlar üzerinde uygulandığı bir alan olan robotik, son yıllarda BİD’e olan etkisi ile ilgili araştırmalara da konu olmuştur (Çetin ve Uçar, 2017). Bu çalışmalarda BİD ile ilgili becerilerin robotik ile geliştirilebileceğinden bahsedilmektedir (Berland vd., 2013; Berland ve Wilensky, 2015; Bers vd., 2014; Chaudhary vd., 2016). Benitti (2012) bazı çalışmaların sonucunda robotiğin öğrencilerde BİD becerileri geliştirmede yeterli etkiyi göstermediğini ifade etmiştir. Robotik ve BİD ile ilgili araştırmalar okul öncesi öğrencilerin seviyesinden başlayıp üniversiteye kadar çıkmaktadır.

Eğitimde en erken yaş olan okul öncesi seviyesindeki öğrencilerin BİD becerilerinin robotik ile geliştirilebileceğini gösteren bulgular olduğunu iletmışlerdir (Bers, 2008; Bers vd., 2014). Okul öncesi öğrenciler ile eğitici robotik aktiviteleri düzenleyen bir başka araştırmada; 131 okul öncesi öğrencisi ile gerçekleştirilen robotik aktivitelerinin BİD’i ve programlama becerilerini geliştirmesindeki etkileri gözlemlenmiştir. Araştırma

deney ve kontrol gruplarını kullanarak ön test ve son test ölçümleriyle yarı deneysel olarak tasarlanmıştır. BİD becerisini üç alt boyuta ayırmıştır: diziler (algoritmalar), talimat-eylem ilişkisi ve hata ayıklama. Robotik aktivitelere katılan çocukların BİD yeterliliğinin üç alt boyutunda da kontrol grubuna göre daha büyük bir ilerleme sağladığı görülmüştür. Deney ve kontrol grubu üyeleri arasında bulunan önemli farklılıklar, deney grubundaki analiz edilen değişkenlerin her biri için daha fazla öğrenmenin varlığını göstermektedir (Muñoz-Repiso ve Caballero-González, 2019).

Başka bir araştırma, Avustralya’da ilkokul öğretmenlerinin eğitsel robotları ve kodlamayı eğitime nasıl entegre ettiklerini ve bunun öğrencilerin BİD becerileri üzerindeki algılanan etkilerini inceleyen bir çalışmasıdır. Ayrıca robotik ve kodlama eğitimi vermenin öğretmenlerin BİD’i öğretmek için bilgilerini geliştirme ve kendine güvenlerine katkısı incelenmiştir. Çalışmada, dört ayrı ilkokuldaki öğretmenler öğrencilere (1-6 yaş) Lego Wedo 2.0 robotik kitleri ile eğitim verilmiştir. Araştırma sonunda anketlerden, günlük veri girişleri ve yarı yapılandırılmış görüşmelerden toplanan veriler analiz edilmiştir. Sonuçlar, eğitsel robotların derste kullanımının öğrencilerin BİD becerileri üzerindeki algılanan olumlu etkiden bahsetmektedir. Robot kitlerini ve etkinliklerini araştırmanın ve kullanmanın, öğretmenlerin öğrencilere BİD’i tanıtmak için kendi güvenlerini ve bilgilerini geliştirmelerinde yardımcı olduğunu göstermektedir (Chalmers, 2018).

Robotik ile ilgili okul öncesi ve ilkokuldaki çalışmalar gibi ortaokul düzeyindeki öğrencilerle yapılan çalışmalar da gün geçtikçe artmaktadır. Hindistan’da yapılan bir çalışmada Lego Mindstorms EV3 robotik eğitim setini kullanarak 9 öğrenciye 10 haftalık bir eğitim verilmiştir. Robotik eğitiminin bilgi işlemsel düşünme, programlama, problem çözme, takım çalışması ve proje yönetimi becerilerine etkisi incelenmiştir. Eğitim sonunda 9 öğrenciden 6’sının bilgi işlemsel düşünme ve programlama becerilerinin düzeylerinde önemli gelişmeler olduğu görülmüştür (Chaudhary vd., 2016). Robotik aktivitelerin etkisinin cinsiyete göre farklılaşıp farklılaşmadıkları ile ilgili sonuçlar bulan çalışmalar da mevcuttur. Türkiye’de 6. ve 7. sınıfa giden 55 öğrenci ile yapılan çalışmada, robotik aktiviteler ile programlama eğitim alan öğrencilerin BİD becerilerinin, programlama öz yeterlik algısının ve problem çözme amaçlı yansıtıcı düşünme becerilerine ilişkin beceri düzeyleri belirlenmiş ve deneyimleri incelenmiştir. Araştırma sonucunda kız öğrencilerin, BİD becerilerinin,

programlama öz yeterlik algısının ve problem çözme amaçlı yansıtıcı düşünme becerilerinin erkek öğrencilere göre daha yüksek olduğu gözlenmiştir (Durak, Yılmaz ve Yılmaz, 2019).

Üniversite düzeyindeki bir araştırmada 21 öğretmen adayının fen bilgisi dersinde uygulanan robotik faaliyetinin öz yeterliklerine, fen kavramlarını anlamalarına ve BİD beceri düzeylerine etkisi incelenmiştir. Bulgular, robotik faaliyetinin robotiğe olan ilgiyi arttırmak, robotik öğretmek için öz-yeterliği artırmak, fen kavramlarını anlamak ve BİD becerilerinin düzeyini arttırmak için etkili bir öğretim stratejisi olduğunu ortaya koymaktadır (Jaipal-Jamani ve Angeli, 2017).

Bu olumlu sonuçların yanında robotiğin hala gelişimini sürdüren bir alan olduğunu ve yeni araştırmalara ihtiyaç olduğunu belirten çalışmalar bulunmaktadır. Araştırmacılar, eğitimde robotik kullanımı ile ilgili alanyazının çoğunun, bireysel inisiyatiflerle olumlu sonuçlar elde eden öğretmenlerin raporlarına dayandığını vurgulayıp sonuçların genellenmesine temkinli yaklaşmışlardır (Caci, Cardaci ve Lund, 2003; Petre ve Price, 2004; Williams, Ma, Prejean, Lai, Ford, 2007). Robotiğin eğitimde kullanımı ile ilgili olumlu sonuçların kalıcılığını artırmak ve genellenebilir olabilmesi için Lye, Wong ve Chiou (2013) eğitsel robotiğin öğrencinin sadece bir kere proje hazırladığı ve yarıştığı bir etkinlik olmaması gerektiğini, ilkokuldan üniversite öncesine yayılacak bir süreç olması gerektiğini ifade etmiştir.

2.3.4. Bilgisayarsız Uygulamalar

Alanyazında Kalelioğlu (2017) tarafından bilgisayarsız bilgisayar bilimi (B3) olarak bahsedilen kavram, birçok bilgisayar bilimi etkinliğinin bilgisayar kullanılmadan da yapılabileceğini ifade etmektedir. Özellikle programlama eğitime başlamadan önce; programlama ile ilgili temel kavramları, algoritma oluşturmaya, hata ayıklamayı ve problem çözme öğretmek için B3 uygulamalarından faydalanılmaktadır. Bu uygulamalara CSunplugged.org, CS4FN, Code.org unplugged, Keşf@ Projesi, Bilge Kunduz (bebras) ve Tospaa.org örnek gösterilebilir. Bu etkinlikler temel bilgisayar becerilerini kazandırma, programlama öncesi kavramsal bilgileri verme ve BİD becerisi kazandırma amaçları için yapılmaktadır. Teknolojik alt yapı eksikliği sorunu yaşayan öğrenciler için de nispeten bir fırsat eşitliği sunmaktadır (Kalelioğlu, 2017). Bu amaçlarla yapılan ve olumlu etkilere rastlanan çalışmalar mevcuttur (Bell, Alexander,

Freeman ve Grimley, 2009; Brackmann, Román-González, Robles, Moreno-León, Casali ve Barone, 2017; Curzon ve McOwan, 2008; Curzon, McOwan, Plant ve Meager, 2014; Faber, Wierdsma, Doornbos, Van Der Ven ve De Vette, 2017; Gardelli ve Vosinakis, 2017; Looi, How, Longkai, Seow, ve Liu, 2018; Sharaf, Ahmed, Adel, Abdennadher ve Berkling, 2019; Thies ve Vahrenhold, 2013). Bahsedilen araştırmalara rağmen B3'ün yeni bir alan olması sebebiyle alanyazında az çalışma bulunmaktadır. Bu sebeple eğitimde kullanımında ve değerlendirilmesinde de yeni çalışmalara açık bir alandır. Jun (2018) B3 uygulamalarının son yıllarda popüler hale geldiğini lakin B3 uygulamalarında kullanılmak üzere değerlendirme standartı bulunmadığını belirterek bu alandaki eksikliği vurgulamıştır. B3 uygulamaları değerlendirme standardı oluşturmaya yönelik çalışmasını paylaşmıştır.

BÖLÜM III: YÖNTEM

3.1. Araştırmanın Modeli

Programlama yöntemlerinin ve cinsiyetin ortaokul öğrencilerinin bilgi işlemsel düşünme becerisine yönelik öz yeterlik algısına ve programlama başarısına etkisini belirlemeyi amaçladığından, araştırma ön test-son test kontrol gruplu araştırma modeline uygun yarı deneysel bir çalışmadır.

Eğitim öncesinde “Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algısı Ölçeği (BİDBÖA)” öğrencilere uygulanmıştır. Eğitim süresi (sekiz hafta) boyunca eğitimlere devam edilmiş ardından tekrar aynı ölçek uygulanmıştır. Ayrıca BİD ve programlama ile ilgili kazanımlardan oluşan bir programlama bilgisi sınavı uygulanmıştır. Tablo 4’de araştırma deseni özet bir şekilde açıklanmıştır.

Tablo 4 : Araştırma Deseni

	ÖN TEST	DENEY	SON TEST
BLOK TABANLI PROGRAMLAMA GRUBU	<u>TÜM GRUPLARA</u>	Algoritma Akış diyagramı Blok tabanlı programlama (Code.org)	<u>TÜM GRUPLARA</u> Ortaokul Öğrencileri İçin Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algısı Ölçeği (BİDBÖA)
METİN TABANLI PROGRAMLAMA GRUBU	Ortaokul Öğrencileri İçin Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algısı Ölçeği (BİDBÖA)	Algoritma Akış diyagramı Metin tabanlı programlama (Small Basic)	ve Programlama Bilgisi Sınavı
ROBOTİK PROGRAMLAMA GRUBU		Algoritma Akış diyagramı Robotik programlama (Robotis Dream / Roboplus Task)	

3.2. Çalışma Grubu

Çalışma grubu İstanbul’da bulunan bir devlet ortaokulundaki öğrencilerden seçilmiştir. Bu araştırmanın çalışma grubunu 202 ortaokul 5. Sınıf öğrencisi (10 – 12 yaş arası) oluşturmaktadır. Araştırma sürecinde programlama eğitimi alan öğrencilerin 90’ı erkek (% 44.5), 112 tanesi ise kız (% 55.5) öğrencidir. Araştırma süreci başında, daha önce programlama eğitimi almış olan beş öğrenci araştırma dışında bırakılmıştır.

Deney grubunu robotik eğitimi alan 32 öğrenciden oluşturmaktadır. Bu grup -daha önce hiç programlama eğitimi almamış- 5. Sınıf öğrencileri arasından robotik eğitimini almak isteyen öğrencilerden oluşmuştur. Eğitim boyunca tasarladıkları robotları yarı metin (C++) yarı blok tabanlı programlama yaptıran bir uygulama olan Roboplus Task yazılımı ile tasarladıkları robotlarını programlamışlardır.

Birinci kontrol grubu ise Bilişim Teknolojileri ve Yazılım dersi alan -daha önce hiç programlama eğitimi almamış- 5.sınıfta okuyan ve Bilişim Teknolojileri ve Yazılım dersi alan 137 öğrenciden oluşmaktadır. Eğitim boyunca blok tabanlı programlama platformu olan Code.org’u kullanarak programlama yapmışlardır.

İkinci kontrol grubu ise Bilişim Teknolojileri ve Yazılım dersinden Destekleme ve Yetiştirme Kursuna gelmek isteyen -daha önce hiç programlama eğitimi almamış- 5. sınıfa giden 33 öğrenciden oluşmaktadır. Eğitim boyunca metin tabanlı programlama aracını (Small Basic) kullanarak programlama yapmışlardır.

3.3. Deney Süreci

Deney sürecinde öğrenciler üç ayrı grupta farklı programlama yöntemleri ve öğretim araçlarıyla eğitim almışlardır. Sadece eğitimin ilk iki haftasında algoritma ve akış diyagramı konularını bilgisayarlı / bilgisayarsız etkinlikler ile tüm gruplara aynı şekilde verilmiştir.

İkinci haftadan itibaren her grup Tablo 5’te gösterilen şekilde eğitime devam etmişlerdir. Code.org öğretim aracını kullanarak blok tabanlı programlama eğitimi alan grubun öğretim aracı ve eğitim süreci ile ilgili detaylı bilgi “Ek 2: Blok Tabanlı Programlama Grubu Öğretim Aracı ve Eğitim Süreci” başlıklı ek bölümde verilmiştir.

Small Basic öğretim aracını kullanarak metin tabanlı programlama eğitimi alan grubun öğretim aracı ve eğitim süreci ile ilgili detaylı bilgi “Ek 3: Metin Tabanlı Programlama Grubu Öğretim Aracı ve Eğitim Süreci” başlıklı ek bölümde verilmiştir.

Robotik Dream ve Roboplus Task öğretim araçlarını kullanarak robotik programlama eğitimi alan grubun öğretim aracı ve eğitim süreci ile ilgili detaylı bilgi “Ek 4: Robotik Programlama Grubu Öğretim Aracı ve Eğitim Süreci” başlıklı ek bölümde verilmiştir.

Araştırmacı, yukarıda detaylandırılan deney süreçlerinde gruplara eğitim vermiştir. Tüm gruplara verilen eğitimlerde, araştırmacı aynı zamanda öğretmen rolünü üstlenmiştir.

Tablo 5 : Deney Süreci

<i>Hafta</i>	<i>Blok Tabanlı Programlama Uygulama Süreci</i>	<i>Metin Tabanlı Programlama Uygulama Süreci</i>	<i>Robotik Programlama Uygulama Süreci</i>
1	Konu: Algoritma nedir? Algoritma örnekleri Algoritma oluşturma Algoritmada hata ayıklama Uygulama: Bilgisayarlı / Bilgisayarsız etkinlik	Konu: Algoritma nedir? Algoritma örnekleri Algoritma oluşturma Algoritmada hata ayıklama Uygulama: Bilgisayarlı / Bilgisayarsız etkinlik	Konu: Algoritma nedir? Algoritma örnekleri Algoritma oluşturma Algoritmada hata ayıklama Uygulama: Bilgisayarlı / Bilgisayarsız etkinlik
2	Konu: Akış diyagramı nedir? Akış diyagramı oluşturma Programlama nedir? Blok tabanlı programlama nedir? Uygulama: Code.org tanıtım	Konu: Akış diyagramı nedir? Akış diyagramı oluşturma Programlama nedir? Metin tabanlı programlama nedir? Uygulama: Small basic tanıtım	Konu: Akış diyagramı nedir? Akış diyagramı oluşturma Programlama nedir? Robotik programlama nedir? Uygulama: Robotis Dream tanıtım
3	Konu: Veri, değişken ve döngü kavramları Uygulama: Code.org – Kurs 4 Ders 6-7 Code.org – Kurs 2 Ders 6-7	Konu: Veri, değişken ve döngü kavramları Uygulama: Small Basic ile konuya ait programlama	Konu: Veri, değişken ve döngü kavramları Uygulama: Robotis Dream ile robot tasarımı ve konuya ait programlama
4	Konu: Döngüler Uygulama: Code.org – Kurs 2 Ders 8 Code.org – Kurs 3 Ders 11- 12-13	Konu: Döngüler Uygulama: Small Basic ile konuya ait programlama	Konu: Döngüler ve Koşul Yapıları Uygulama: Robotis Dream ile robot tasarımı ve konuya ait programlama
5	Konu: Koşul Yapıları Uygulama: Code.org – Kurs 2 Ders 13 Code.org – Kurs 3 Ders 7-8	Konu: Koşul Yapıları Uygulama: Small Basic ile konuya ait programlama	Konu: Koşul Yapıları Uygulama: Robotis Dream ile robot tasarımı ve konuya ait programlama
6	Konu: Hata ayıklama ve koşul yapıları Uygulama: Code.org – Kurs 2 Ders 10-11 Code.org – Kurs 3 Ders 14	Konu: Hata ayıklama ve koşul yapıları Uygulama: Small Basic ile konuya ait programlama	Konu: Hata ayıklama ve koşul yapıları Uygulama: Robotis Dream ile robot tasarımı ve konuya ait programlama
7	Konu: Fonksiyonlar Uygulama: Code.org – Kurs 3 Ders 5 Code.org – Kurs 4 Ders 12	Konu: Fonksiyonlar Uygulama: Small Basic ile konuya ait programlama	Konu: Fonksiyonlar Uygulama: Robotis Dream ile robot tasarımı ve konuya ait programlama
8	Uygulama: Eğitim süreci boyunca öğrenilen programlama bilgilerini içeren genel tekrar – Code.org	Uygulama: Eğitim süreci boyunca öğrenilen programlama bilgilerini içeren genel tekrar – Small Basic	Uygulama: Eğitim süreci boyunca öğrenilen programlama bilgilerini içeren genel tekrar – Robotis Dream

3.4. Veri Toplama Araçları

3.4.1. Ortaokul Öğrencileri için Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algısı Ölçeği (BİDBÖA)

Veri toplama aracı olarak geçerlik ve güvenirlik çalışması yapılmış olan “Ortaokul Öğrencileri için Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algısı Ölçeği (BİDBÖA)” kullanılmıştır. Gerekli izinler ölçeği geliştiren araştırmacılardan alınmıştır.

Ölçek formu 5 alt faktöre ayrılmış toplam 36 maddeden oluşmaktadır. Ölçeği geliştiren araştırmacılar tarafından oluşturulan 5 alt faktör şu şekildedir:

- Algoritma tasarlama yeterliği (9 madde)
- Problem çözme yeterliği (11 madde)
- Veri işleme yeterliği (7 madde)
- Temel programlama yeterliği (6 madde)
- Özgüven yeterliği (6 madde)

Ölçek ortaokul öğrencilerinin rahat cevap vermesi için 3'lü Likert yapıda desenlenmiştir. Bu yapı “Evet”, “Kısmen” ve “Hayır” cevaplarından oluşmaktadır.

Ölçeğin geliştirilme aşamasında, “faktörlerin düzeltilmiş madde-toplam puan korelasyon değerlerinin 0.632 ile 0.386 arasında olduğu, Cronbach Alfa katsayılarının ise 0.762 ile 0.930 arasında değiştiği, t-testi sonuçları, üst %27 ile alt %27 grupların madde ortalamaları arasındaki tüm farkların anlamlı olduğu” ölçeği geliştiren araştırmacılar tarafından aktarılmıştır (Gülbahar vd., 2018). Ölçekten en az 36 ile en fazla 108 arasında bir puan alınabilmektedir. Bu puanlar öğrencilerin BİD'e yönelik öz yeterlik algı seviyeleri açısından bize ipuçları vermektedir.

Ölçeğin uygulanması ile elde edilen BİD becerisine yönelik öz yeterlik algı puanının düşük olması, öğrencinin öz yeterlik algı düzeyinin yüksek olduğu sonucunu vermektedir. Aynı şekilde BİD becerisine yönelik öz yeterlik algı puanının yüksek olması, öğrencinin öz yeterlik algı düzeyinin düşük olduğu sonucunu vermektedir.

3.4.2. Programlama Bilgisi Sınavı

İkinci bir veri toplama aracı olarak, BİD ve programlama ile ilgili kazanımlardan oluşan programlama bilgisi sınavı kullanılmıştır. Bu programlama bilgisi sınavından aldıkları

puanlar, verilerin çözümlenmesinde programlama başarı puanı hanesinde değerlendirilmiştir. Programlama bilgisi sınavı, belirtke tablosu ve cevap anahtarı üç bilişim teknolojileri uzmanı tarafından hazırlanmıştır. Ek 5'te bulunan Programlama Bilgisi Sınavı Soru Örnekleri - Cevapları ve Kazanımları tablosunda özet bilgi verilmiştir. Programlama bilgisi sınav sorularının hazırlanması ve değerlendirmesi alan uzmanı 3 öğretmen tarafından yapılmıştır.

3.5. Verilerin Toplanması

Eğitim öncesinde ve sonrasında uygulanan “Ortaokul Öğrencileri için Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algısı Ölçeği” verileri toplanarak SPSS programına aktarılmıştır. Ayrıca uygulama sürecinin sonunda uygulanan -BİD ve programlama ile ilgili kazanımlardan oluşan- programlama bilgisi sınavı puanları da toplanarak SPSS programına aktarılmıştır.

3.6. Verilerin Çözümlenmesi

Verilerin çözümlenmesi için SPSS programı (versiyon 22) ile veri analizi yapılmıştır. Veri analizi yapılırken betimsel istatistik, normallik testi, Kruskal-Wallis H, Wilcoxon işaretli sıralar testi ve ikili karşılaştırma testi kullanılmıştır.

BÖLÜM IV: BULGULAR

Araştırmanın bu bölümünde “ortaokul öğrencileri için bilgi işlemsel düşünme becerisine yönelik öz yeterlik algısı ölçeği (BİDBÖA)” ve “programlama bilgisi sınavı” ile elde edilen veriler bulunmaktadır. Bu verilerin analiz edilmesiyle elde edilen bulgulara yer verilmektedir.

Veri analizleri ile elde edilen bulguların doğru yorumlanabilmesi açısından BİDBÖA ölçeğinden elde edilen BİD’e yönelik öz yeterlik algı puanının düşük olmasının, öğrencinin BİD öz yeterlik algı düzeyinin yüksek olduğu ile ilgili ipucu vereceği unutulmamalıdır. Aynı şekilde BİD’e yönelik öz yeterlik algı puanının yüksek olmasının, öğrencinin BİD öz yeterlik algı düzeyinin düşük olduğu sonucunu vermektedir.

4.1. Varsayımların İncelenmesi

Araştırmadaki analiz işlemlerine geçmeden önce kullanılacak analizlerin varsayımlarının incelenmesi gerekmektedir. Bu varsayımların içinde en önemlilerinden birisi verilerin normal dağılıma sahip olup olmadığına bakmaktır. Verilerin normal dağılım göstermesi ve diğer varsayımları karşılaması parametrik testleri, normal dağılım göstermemesi durumunda ise parametrik olmayan testleri kullanmamız gerektiğini bize gösterecektir (Büyüköztürk, 2009).

4.1.1. Deney Gruplarına Göre Varsayımların İncelenmesi

Deney öncesi ve sonrası öğrencilere uygulanan “BİDBÖA” ön test, son test puanları ve “programlama bilgisi sınavı” puanlarının deney gruplarına göre dağılımı normallik testine tabi tutulmuştur. Bu teste ait sonuçlar Tablo 6’da görülmektedir.

Tablo 6: *Normallik Testi (Test Puanlarının Deney Gruplarına Göre Dağılımı)*

	Deney Grubu	n	Kolmogorov-Smirnov ^a			Shapiro-Wilk		
			Test Puanı	Sd	p	Test Puanı	Sd	p
Öz Yeterlik	Blok	137	,077	137	,045	,969	137	,003
Ön Test	Robotik	32	,142	32	,099	,947	32	,118
Puanı	Metin	33	,087	33	,200*	,963	33	,313
Öz Yeterlik	Blok	137	,110	137	,000	,937	137	,000
Son Test	Robotik	32	,198	32	,003	,763	32	,000
Puanı	Metin	33	,125	33	,200*	,892	33	,003
Programla	Blok	137	,218	137	,000	,841	137	,000
ma Başarı	Robotik	32	,375	32	,000	,708	32	,000
Puanı	Metin	33	,263	33	,000	,742	33	,000

*. Bu, gerçek öneme sahip bir alt sınırdır.

a. Lilliefors Significance Correction

Normalliği incelerken kullanılacak test, grupların büyüklüğüne göre seçilmektedir. Grup büyüklüğünün 50’den büyük olması durumunda Kolmogorov-Smirnov testi, 50’den küçük olma durumunda Shapiro-Wilk testi kullanılmaktadır (Büyüköztürk, 2009). Tablo 6’da hem 50’den küçük hem de 50’den büyük gruplar bulunduğu için her grup için ilgili grup büyüklüğüne göre doğru analiz verileri baz alınmıştır. Tablo 6’dan elde edilen veriler ışığında öğrencilerin “BİDBÖA” ön test, son test puanları ve “programlama bilgisi sınavı” puanları -metin tabanlı programlama ve robotik programlama eğitimi alan grupların ön test puanları hariç- deney gruplarına göre normal bir dağılım göstermemektedir. Verilerin büyük çoğunluğu normal dağılım göstermediği için, ön test, son test ve sınav puanlarının deney gruplarına göre analiz edilmesinde non-parametrik testlerden faydalanılmıştır.

4.1.2. Cinsiyete Göre Varsayımların İncelenmesi

Deney öncesi ve sonrası öğrencilere uygulanan “BİDBÖA” ön test, son test puanları ve “programlama bilgisi sınavı” puanlarının cinsiyete göre dağılımı normallik testine tabi tutulmuştur. Bu teste ait sonuçlar Tablo 7’de görülmektedir.

Tablo 7: *Normallik Testi (Test Puanlarının Cinsiyete Göre Dağılımı)*

	Cinsiyet	n	Kolmogorov-Smirnov ^a			Shapiro-Wilk		
			Test Puanı	Sd	p	Test Puanı	Sd	p
Öz Yeterlik Ön Test Puanı	Erkek	90	,095	90	,046	,959	90	,007
	Kız	112	,093	112	,018	,961	112	,003
Öz Yeterlik Son Test Puanı	Erkek	90	,144	90	,000	,898	90	,000
	Kız	112	,108	112	,003	,931	112	,000
Programlama Başarı Puanı	Erkek	90	,254	90	,000	,809	90	,000
	Kız	112	,232	112	,000	,808	112	,000

a. Lilliefors Significance Correction

Analiz yapacağımız grup büyüklüklerinin 50'den yüksek olması sebebiyle Kolmogorov-Smirnov testine ait analizler incelenmiştir (n=90 ve n=112). Tablo 7'den elde edilen veriler incelendiğinde öğrencilerin "BİDBÖA" ön test, son test puanları ve "programlama bilgisi sınavı" puanları cinsiyete göre normal bir dağılım göstermemektedir. Bu sebeple ön test, son test ve sınav puanlarının cinsiyete göre analiz edilmesinde nonparametrik testlerden faydalanılmıştır.

4.2. Grup Denkliklerinin İncelenmesi

Araştırma, deney grupları ve cinsiyet ile ilgili analiz içerdiği için, analizlere geçmeden önce deney grupları ve cinsiyetler arasında başlangıçta denklik olup olmadığını kontrol etmemiz gerekmektedir.

4.2.1. Deney Gruplarına Göre Denklik İncelenmesi

Deney gruplarının ön test puanlarına ait betimsel istatistikler Tablo 8'de sunulmuştur.

Tablo 8: *BİDBÖA Ön Test Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları*

Puan	Gruplar	n	Ortalama	Ortanca	Varyans	Ss	Min	Maks
Öz Yeterlik	Blok	137	77.76	79.00	121.00	11.00	52	99
Ön Test	Robotik	32	72.81	73.50	199.77	14.13	52	104
Puanı	Metin	33	74.87	76.00	161.67	12.71	53	97

Tablo 8’deki BİD becerilerine yönelik öz yeterlik algısı ön test puanlarının betimsel istatistikleri incelendiğinde robotik programlama eğitimi alacak grubun ortalama ön test puanlarının diğer iki gruba göre daha az yani öz yeterlik düzeyinin daha yüksek olduğu görülmüştür. Betimsel istatistiğin ardından grupların ön test puanları Kruskal Wallis H testi ile karşılaştırılmıştır.

Tablo 9: BİDBÖA Ön Test Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili Kruskal Wallis H Testi Sonuçları

Puan	Gruplar	n	Sıra Ort.	sd	x ²	p
Öz Yeterlik Ön Test Puanı	Blok	137	107.71	2	5.44	.066
	Robotik	32	82.55			
	Metin	33	94.11			

Öz Yeterlik Algısı Ön Test Puanları Analiz Tablosundaki değerler incelendiğinde anlamlı bir fark bulunmamıştır (p=.066). Bu sonuca göre deney gruplarının ön test puanlarının benzer olduğu görülmektedir.

4.2.2. Cinsiyete Göre Denklik İncelenmesi

Cinsiyete göre ön test puanlarına ait betimsel istatistikler Tablo 10’da sunulmuştur.

Tablo 10: BİDBÖA Ön Test Puanlarının Cinsiyete Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları

Puan	Gruplar	n	Ortalama	Ortanca	Varyans	Ss	Min	Maks
Öz Yeterlik	Erkek	90	74.77	74.00	151.90	12.32	55	104
Ön Test Puanı	Kız	112	77.90	79.00	131.15	11.45	52	101

Tablo 10’daki BİD becerilerine yönelik öz yeterlik algısı ön test puanlarının betimsel istatistikleri incelendiğinde eğitim alacak erkek öğrencilerin ortalama ön test puanlarının kız öğrencilere göre daha az yani öz yeterlik düzeyinin daha yüksek olduğu görülmüştür. Betimsel istatistiğin ardından gruplar arasında BİDBÖA ön test puanlarının denk olup olmadığını test etmek için Kruskal Wallis H testi uygulanmıştır.

Tablo 11: BİDBÖA Ön Test Puanlarının Cinsiyete Göre Farklılaşma Durumu ile İlgili Kruskal Wallis H Testi Sonuçları

Puan	Gruplar	n	Sıra Ort.	sd	x ²	p
Öz Yeterlik	Erkek	90	92.89	2	3.52	.061
Puanı	Kız	112	108.42			

Öz Yeterlik Algısı Ön Test Puanları Analiz Tablosundaki değerler incelendiğinde cinsiyete göre gruplar arasında bu sonuca göre anlamlı bir fark bulunmamıştır (p=.061). Bu sonuca göre cinsiyete göre ön test puanlarının denk olduğu söylenebilir.

4.3. Öğrencilerin Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algıları Ön Test ve Son Test Puanları Arasında Yapılacak Analizlere Ait Bulgular

4.3.1. Öğrencilerin Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algıları Ön Test ve Son Test Puanları Arasında Blok Tabanlı Programlama Öğretim Yöntemini Gören Öğrenciler İçin Fark Var mıdır? Sorusuna Ait Bulgular

Bu başlık altında blok tabanlı programlama eğitimini alan öğrencilerin BİDBÖA ön test ve son test puanları arasında fark olup olmadığı sınanacaktır. Analizlerden önce Tablo 12’de ön test ve son test puanlarına yönelik betimsel istatistikler verilmiştir.

Tablo 12: Blok Tabanlı Programlama Eğitimi Alan Öğrencilerin BİDBÖA Ön Test ve Son Test Puanları ile İlgili Betimsel İstatistik Sonuçları

Deney	Puan	Ortalama	Ortanca	Varyans	Ss	Min	Maks
Blok Tabanlı	Ön Test	77.76	79.00	121.04	11.00	52	99
Programlama	Son Test	49.81	49.00	122.83	11.08	36	87

Büyüköztürk’e (2009) göre deneklerin puanları normal dağılım göstermediğinde ve aynı denekler üzerinde iki farklı zamanda yapılan ölçümlere ait puanların arasında fark olup olmadığına bakılırken Wilcoxon İşaretli Sıralar Testi uygulanmalıdır. Tablo 13’te blok tabanlı programlama eğitimi alan öğrencilerin BİDBÖA ön test ve son test puanları arasında fark olup olmadığı analiz edilmiştir.

Tablo 13: Blok Tabanlı Programlama Eğitimi Alan Öğrencilerin BİDBÖA Ön Test ve Son Test Puanları için Wilcoxon İşaretili Sıralar Testi Sonuçları

Son Test – Ön Test	n	Sıra Ortalaması	Sıra Toplamı	z	p
Negatif Sıra	132	70.29	9278.00	10.03*	.000
Pozitif Sıra	4	9.50	38.00		
Eşit	1				

*Negatif sıralar temeline dayalı

Tablo 13'teki analiz sonuçlarına göre, blok tabanlı programlama eğitimi alan öğrencilerin BİDBÖA ön test ve son test puanları arasında anlamlı bir fark olduğunu göstermektedir ($z=10.03$, $p=.000$). Betimsel istatistikler incelendiğinde BİDBÖA puan ortalamalarının eğitim sonrasında 77.76'dan 49.81'e düştüğü görülmektedir. Ortalamalarda görülen bu değişiklik blok tabanlı programlama eğitimi alan öğrencilerin BİD becerisine yönelik öz yeterlik algı seviyelerinde istatistiksel olarak anlamlı bir yükselme olduğunu gösterir. Bu sonuçlara göre, blok tabanlı programlama eğitiminin çocukların BİD becerisine yönelik öz yeterlik algısını geliştirmede önemli bir etkisinin olduğu söylenebilir.

4.3.2. Öğrencilerin Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algıları Ön Test ve Son Test Puanları Arasında Metin Tabanlı Programlama Öğretim Yöntemini Gören Öğrenciler İçin Fark Var mıdır? Sorusuna Ait Bulgular

Bu başlık altında metin tabanlı programlama eğitimini alan öğrencilerin BİDBÖA ön test ve son test puanları arasında fark olup olmadığı sınanacaktır. Analizlerden önce Tablo 14'te ön test ve son test puanlarına yönelik betimsel istatistikler verilmiştir.

Tablo 14: Metin Tabanlı Programlama Eğitimi Alan Öğrencilerin BİDBÖA Ön Test ve Son Test Puanları ile İlgili Betimsel İstatistik Sonuçları

Deney	Puan	Ortalama	Ortanca	Varyans	Ss	Min	Maks
Metin Tabanlı	Ön Test	74.87	76.00	161.67	12.71	53	97
Programlama	Son Test	48.57	47.00	120.02	10.95	36	70

Tablo 15’te metin tabanlı programlama eğitimi alan öğrencilerin BİDBÖA ön test ve son test puanları arasında fark olup olmadığı Wilcoxon İşaretli Sıralar Testi ile analiz edilmiştir.

Tablo 15: *Metin Tabanlı Programlama Eğitimi Alan Öğrencilerin BİDBÖA Ön Test ve Son Test Puanları için Wilcoxon İşaretli Sıralar Testi Sonuçları*

Son Test – Ön Test	n	Sıra Ortalaması	Sıra Toplamı	z	p
Negatif Sıra	32	17.41	557.00	4.94*	.000
Pozitif Sıra	1	4.00	4.00		
Eşit	0				

*Negatif sıralar temeline dayalı

Tablo 15’teki analiz sonuçlarına göre, metin tabanlı programlama eğitimi alan öğrencilerin BİDBÖA ön test ve son test puanları arasında anlamlı bir fark olduğunu göstermektedir ($z=4.94$, $p=.000$). Betimsel istatistikler incelendiğinde BİDBÖA puan ortalamalarının eğitim sonrasında 74.87’den 48.57’ye düştüğü görülmektedir. Ortalamalarda görülen bu değişiklik metin tabanlı programlama eğitimi alan öğrencilerin BİD becerisine yönelik öz yeterlik algı seviyelerinde istatistiksel olarak anlamlı bir yükselme olduğunu gösterir. Bu sonuçlara göre, metin tabanlı programlama eğitiminin çocukların BİD becerisine yönelik öz yeterlik algılarını geliştirmede önemli bir etkisinin olduğu söylenebilir.

4.3.3. Öğrencilerin Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algıları Ön Test ve Son Test Puanları Arasında Robotik Programlama Öğretim Yöntemini Gören Öğrenciler İçin Fark Var mıdır? Sorusuna Ait Bulgular

Bu başlık altında robotik programlama eğitimini alan öğrencilerin BİDBÖA ön test ve son test puanları arasında fark olup olmadığı sınanacaktır. Analizlerden önce Tablo 16’da ön test ve son test puanlarına yönelik betimsel istatistikler verilmiştir.

Tablo 16: Robotik Programlama Eğitimi Alan Öğrencilerin BİDBÖA Ön Test ve Son Test Puanları ile İlgili Betimsel İstatistik Sonuçları

Deney	Puan	Ortalama	Ortanca	Varyans	Ss	Min	Maks
Robotik	Ön Test	72.81	73.50	199.70	14.13	52	104
Programlama	Son Test	44.68	42.00	105.06	10.24	36	86

Tablo 17’de robotik programlama eğitimi alan öğrencilerin BİDBÖA ön test ve son test puanları arasında fark olup olmadığı Wilcoxon İşaretli Sıralar Testi ile analiz edilmiştir.

Tablo 17: Robotik Programlama Eğitimi Alan Öğrencilerin BİDBÖA Ön Test ve Son Test Puanları için Wilcoxon İşaretli Sıralar Testi Sonuçları

Son Test – Ön Test	n	Sıra Ortalaması	Sıra Toplamı	z	p
Negatif Sıra	31	17.00	527.00	4.91*	.000
Pozitif Sıra	1	1.00	1.00		
Eşit	0				

*Negatif sıralar temeline dayalı

Tablo 17’deki analiz sonuçlarına göre, robotik programlama eğitimi alan öğrencilerin BİDBÖA ön test ve son test puanları arasında anlamlı bir fark olduğunu göstermektedir ($z=4.91$, $p=.000$). Betimsel istatistikler incelendiğinde BİDBÖA puan ortalamalarının eğitim sonrasında 72.81’den 44.68’e düştüğü görülmektedir. Ortalamalarda görülen bu değişiklik robotik programlama eğitimi alan öğrencilerin BİD becerisine yönelik öz yeterlik algı seviyelerinde istatistiksel olarak anlamlı bir yükselme olduğunu gösterir. Bu sonuçlara göre, robotik programlama eğitiminin çocukların BİD becerisine yönelik öz yeterlik algısını geliştirmede önemli bir etkisinin olduğu söylenebilir.

4.4. Öğrencilerin Katıldıkları Programlama Öğretim Yöntemine Göre Yapılacak Analizlere Ait Bulgular

4.4.1. Öğrencilerin Katıldıkları Programlama Öğretim Yöntemine Göre Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algıları Son Test Puanları Arasında Fark Var mıdır? Araştırma Sorusuna Ait Bulgular

Bu başlık altında BİDBÖA son test puanlarının deney gruplarına göre değişimi ile ilgili betimsel istatistik sonuçları verilmiştir. Ardından programlama eğitimini farklı öğretim yöntemlerine göre alan deney gruplarının uygulamalar sonrasındaki BİDBÖA son test puanları arasında fark olup olmadığını gözlemlemek için Kruskal-Wallis H Testi uygulanmıştır.

Tablo 18: BİDBÖA Son Test Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları

Puan	Gruplar	n	Ortalama	Ortanca	Varyans	Ss	Min	Maks
Öz Yeterlik	Blok	137	49.81	49.00	122.83	11.08	36	87
Son Test	Robotik	32	44.68	42.00	105.06	10.24	36	86
Puanı	Metin	33	48.57	47.00	120.00	10.95	36	70

Betimsel istatistik verilerine göre grupların aritmetik ortalamaları dikkate alındığında, uygulama sonrasında en düşük BİDBÖA son test puanına ve en yüksek BİD öz yeterlik algı düzeyine robotik programlama grubundaki öğrencilerin ($X=44.68$, $ss=10.24$) sahip olduğu görülmüştür. Metin tabanlı programlama grubundaki öğrencilerin ($X=48.57$, $ss=10.95$), Blok tabanlı programlama grubundaki öğrencilere ($X=49.81$, $ss=11.08$) göre daha düşük BİD öz yeterlik algı puanına yani daha yüksek BİD öz yeterlik algı düzeyine sahip olduğu bir diğer analiz sonucudur.

Programlama eğitimini farklı öğretim yöntemlerine göre alan deney gruplarının uygulamalar sonrasındaki BİDBÖA son test puanlarının Kruskal-Wallis Testi sonuçları Tablo 19’da verilmiştir.

Tablo 19: BİDBÖA Son Test Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili Kruskal Wallis H Testi Sonuçları

Puan	Gruplar	n	Sıra Ort.	sd	χ^2	p
Öz Yeterlik Son Test Puanı	Blok	137	107.68	2	7.39	.025
	Robotik	32	76.56			
	Metin	33	100.03			

Analiz sonuçları, deneysel çalışmaya katılan öğrencilerin BİDBÖA son testinden aldığı puanların, aldıkları öğretim yöntemine göre anlamlı bir şekilde farklılaştığını göstermektedir (χ^2 (sd=2, n=202) =7.39, p=.025). Bu bulgu, kullanılan üç yöntemin öğrencilerin BİD becerilerine yönelik öz yeterlik algılarını artırmada farklı etkilere sahip olduğunu göstermiştir.

Tablo 19’da görülen bu anlamlı farkın hangi gruplar arasında gerçekleştiğini ortaya koymak adına, Kruskal Wallis H testinin post hoc yöntemlerinden ikili karşılaştırmalar yapılmıştır.

Tablo 20: BİDBÖA Son Test Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili İkili Karşılaştırma Sonuçları

Grup 1	Grup 2	Standart Test İstatistiği	Standart Hata	p	Ayarlanmış p
Robotik	Blok	2.71	11.46	.007	.020
Robotik	Metin	-1.62	14.48	.105	.315
Metin	Blok	.676	11.31	.499	1.000

İkili karşılaştırma sonuçları robotik programlama eğitimi alan grup ile blok tabanlı programlama eğitimi alan grup arasında anlamlı bir fark olduğunu ortaya koymuştur (p=.020). Robotik programlama eğitimi alan grubun BİD öz yeterlik düzeylerinin, blok tabanlı programlama eğitimi alan grubun BİD öz yeterlik düzeylerine göre anlamlı olarak daha yüksek olduğu görülmüştür.

4.4.2. Öğrencilerin Katıldıkları Programlama Öğretim Yöntemine Göre Algoritma Tasarlama Yeterliği Alt Faktörü Puanları Arasında Fark Var mıdır? Araştırma Sorusuna Ait Bulgular

Bu başlık altında algoritma tasarlama yeterliği alt faktör son test puanlarının deney gruplarına göre değişimi ile ilgili betimsel istatistik sonuçları verilmiştir. Ardından programlama eğitimini farklı öğretim yöntemlerine göre alan deney gruplarının uygulamalar sonrasındaki algoritma tasarlama yeterliği alt faktör puanları arasında fark olup olmadığını gözlemlemek için Kruskal-Wallis H Testi uygulanmıştır.

Tablo 21: Algoritma Tasarlama Yeterliği Alt Faktör Son Test Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları

Puan	Gruplar	n	Ortalama	Ortanca	Varyans	Ss	Min	Maks
Algoritma	Blok	137	11.72	11.00	9.83	3.13	9	21
Tasarlama	Robotik	32	10.50	9.00	3.93	1.98	9	16
Yeterliği Alt	Metin	33	11.54	10.00	8.00	2.82	9	18
Faktörü Son Test								
Puanı								

Grupların aritmetik ortalamaları dikkate alındığında, uygulama sonrasında en düşük algoritma tasarlama yeterliği alt faktör puanına ve en yüksek algoritma tasarlama yeterlik düzeyine robotik programlama grubundaki öğrencilerin sahip olduğu görülmüştür. Metin tabanlı programlama grubundaki öğrencilerin, blok tabanlı programlama grubundaki öğrencilere göre daha düşük algoritma tasarlama yeterliği alt faktör puanına yani daha yüksek algoritma tasarlama yeterlik düzeyine sahip olduğu bir diğer analiz sonucudur.

Programlama eğitimini farklı öğretim yöntemlerine göre alan deney gruplarının uygulamalar sonrasındaki algoritma tasarlama yeterliği alt faktör puanlarının Kruskal-Wallis Testi sonuçları Tablo 22’de verilmiştir.

Tablo 22: *Algoritma Tasarlama Yeterliği Alt Faktör Son Test Puanlarının Gruplara Göre Karşılaştırılması, Kruskal-Wallis Test Sonuçları*

Puan	Gruplar	n	Sıra Ort.	sd	χ^2	p
Algoritma Tasarlama	Blok	137	106.12	2	5.10	.078
Yeterliği Alt Faktörü Son	Robotik	32	80.89			
Test Puanı	Metin	33	102.29			

Analiz sonuçları, deneysel çalışmaya katılan öğrencilerin algoritma tasarlama yeterliği alt faktör puanlarının, aldıkları öğretim yöntemine göre anlamlı bir şekilde farklılaşmadığını göstermektedir, χ^2 (sd=2, n=202) =5.10, p=.078.

4.4.3. Öğrencilerin Katıldıkları Programlama Öğretim Yöntemine Göre Problem Çözme Yeterliği Alt Faktörü Puanları Arasında Fark Var mıdır? Araştırma Sorusuna Ait Bulgular

Bu başlık altında problem çözme yeterliği alt faktör son test puanlarının deney gruplarına göre değişimi ile ilgili betimsel istatistik sonuçları verilmiştir. Ardından programlama eğitimini farklı öğretim yöntemlerine göre alan deney gruplarının uygulamalar sonrasındaki problem çözme yeterliği alt faktör puanları arasında fark olup olmadığını gözlemlemek için Kruskal-Wallis H Testi uygulanmıştır.

Tablo 23: *Problem Çözme Yeterliği Alt Faktör Son Test Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları*

Puan	Gruplar	n	Ortalama	Ortanca	Varyans	Ss	Min	Maks
Problem Çözme	Blok	137	13.30	13.00	11.18	3.34	10	23
Yeterliği Alt	Robotik	32	12.50	11.50	13.35	3.65	10	28
Faktörü Son	Metin	33	12.72	11.00	12.39	3.52	10	22
Test Puanı								

Grupların aritmetik ortalamaları dikkate alındığında, uygulama sonrasında en düşük problem çözme yeterliği alt faktör puanına ve en yüksek problem çözme yeterlik düzeyine robotik programlama grubundaki öğrencilerin sahip olduğu görülmüştür. Metin tabanlı programlama grubundaki öğrencilerin, blok tabanlı programlama

grubundaki öğrencilere göre daha düşük problem çözme yeterliği alt faktör puanına yani daha yüksek problem çözme yeterlik düzeyine sahip olduğu bir diğer analiz sonucudur.

Programlama eğitimini farklı öğretim yöntemlerine göre alan deney gruplarının uygulamalar sonrasındaki problem çözme yeterliği alt faktör puanlarının Kruskal-Wallis Testi sonuçları Tablo 24’te verilmiştir.

Tablo 24: *Problem Çözme Yeterliği Alt Faktör Son Test Puanlarının Gruplara Göre Karşılaştırılması, Kruskal-Wallis Test Sonuçları*

Puan	Gruplar	n	Sıra Ort.	sd	χ^2	p
Problem Çözme Yeterliği Alt Faktörü Son Test Puanı	Blok	137	107.42	2	5.13	.077
	Robotik	32	83.61			
	Metin	33	94.26			

Analiz sonuçları, deneysel çalışmaya katılan öğrencilerin problem çözme yeterliği alt faktör puanlarının, aldıkları öğretim yöntemine göre anlamlı bir şekilde farklılaşmadığını göstermektedir, χ^2 (sd=2, n=202) =5.13, p=.077.

4.4.4. Öğrencilerin Katıldıkları Programlama Öğretim Yöntemine Göre Veri İşleme Yeterliği Alt Faktörü Puanları Arasında Fark Var mıdır? Araştırma Sorusuna Ait Bulgular

Bu başlık altında veri işleme yeterliği alt faktör son test puanlarının deney gruplarına göre değişimi ile ilgili betimsel istatistik sonuçları verilmiştir. Ardından programlama eğitimini farklı öğretim yöntemlerine göre alan deney gruplarının uygulamalar sonrasındaki veri işleme yeterliği alt faktör puanları arasında fark olup olmadığını gözlemlemek için Kruskal-Wallis H Testi uygulanmıştır.

Tablo 25: *Veri İşleme Yeterliği Alt Faktör Son Test Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları*

Puan	Gruplar	n	Ortalama	Ortanca	Varyans	Ss	Min	Maks
Veri İşleme	Blok	137	9.63	9.00	6.14	2.47	7	17
Yeterliği Alt	Robotik	32	8.90	7.00	7.44	2.72	7	16
Faktörü Son	Metin	33	9.51	9.00	7.38	2.71	7	16
Test Puanı								

Grupların aritmetik ortalamaları dikkate alındığında, uygulama sonrasında en düşük veri işleme yeterliği alt faktör puanına ve en yüksek veri işleme yeterlik düzeyine robotik programlama grubundaki öğrencilerin sahip olduğu görülmüştür. Metin tabanlı programlama grubundaki öğrencilerin, blok tabanlı programlama grubundaki öğrencilere göre daha düşük veri işleme yeterliği alt faktör puanına yani daha yüksek veri işleme yeterlik düzeyine sahip olduğu bir diğer analiz sonucudur.

Programlama eğitimini farklı öğretim yöntemlerine göre alan deney gruplarının uygulamalar sonrasındaki veri işleme yeterliği alt faktör puanlarının Kruskal-Wallis Testi sonuçları Tablo 26’da verilmiştir.

Tablo 26: *Veri İşleme Yeterliği Alt Faktör Son Test Puanlarının Gruplara Göre Karşılaştırılması, Kruskal-Wallis Test Sonuçları*

Puan	Gruplar	n	Sıra Ort.	sd	χ^2	p
Veri İşleme Yeterliği Alt Faktörü Son Test Puanı	Blok	137	106.23	2	4.57	.103
	Robotik	32	82.28			
	Metin	33	100.48			

Analiz sonuçları, deneysel çalışmaya katılan öğrencilerin veri işleme yeterliği alt faktör puanlarının, aldıkları öğretim yöntemine göre anlamlı bir şekilde farklılaşmadığını göstermektedir, χ^2 (sd=2, n=202) =4.54, p=.103.

4.4.5. Öğrencilerin Katıldıkları Programlama Öğretim Yöntemine Göre Temel Programlama Yeterliği Alt Faktörü Puanları Arasında Fark Var mıdır? Araştırma Sorusuna Ait Bulgular

Bu başlık altında temel programlama yeterliği alt faktör son test puanlarının deney gruplarına göre değişimi ile ilgili betimsel istatistik sonuçları verilmiştir. Ardından programlama eğitimini farklı öğretim yöntemlerine göre alan deney gruplarının uygulamalar sonrasındaki temel programlama yeterliği alt faktör puanları arasında fark olup olmadığını gözlemlemek için Kruskal-Wallis H Testi uygulanmıştır.

Tablo 27: Temel Programlama Yeterliği Alt Faktör Son Test Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları

Puan	Gruplar	n	Ortalama	Ortanca	Varyans	Ss	Min	Maks
Temel	Blok	137	8.19	8.00	8.15	2.85	5	15
Programlama	Robotik	32	7.12	7.00	4.69	2.16	5	14
Yeterliği Alt	Metin	33	8.24	8.00	8.68	2.94	5	15
Faktörü Son Test								
Puanı								

Grupların aritmetik ortalamaları dikkate alındığında, uygulama sonrasında en düşük temel programlama yeterliği alt faktör puanına ve en yüksek temel programlama yeterlik düzeyine robotik programlama grubundaki öğrencilerin sahip olduğu görülmüştür. Blok tabanlı programlama grubundaki öğrencilerin, metin tabanlı programlama grubundaki öğrencilere göre daha düşük temel programlama yeterliği alt faktör puanına yani daha yüksek temel programlama yeterlik düzeyine sahip olduğu bir diğer analiz sonucudur.

Programlama eğitimini farklı öğretim yöntemlerine göre alan deney gruplarının uygulamalar sonrasındaki temel programlama yeterliği alt faktör puanlarının Kruskal-Wallis Testi sonuçları Tablo 28’de verilmiştir.

Tablo 28: Temel Programlama Yeterliği Alt Faktör Son Test Puanlarının Gruplara Göre Karşılaştırılması, Kruskal-Wallis Test Sonuçları

Puan	Gruplar	n	Sıra Ort.	sd	χ^2	p
Temel Programlama	Blok	137	104.89	2	3.70	.157
Yeterliği Alt Faktörü Son	Robotik	32	83.45			
Test Puanı	Metin	33	104.94			

Analiz sonuçları, deneysel çalışmaya katılan öğrencilerin temel programlama yeterliği alt faktör puanlarının, aldıkları öğretim yöntemine göre anlamlı bir şekilde farklılaşmadığını göstermektedir, χ^2 (sd=2, n=202) =3.70, p=.157.

4.4.6. Öğrencilerin Katıldıkları Programlama Öğretim Yöntemine Göre Özgüven Yeterliği Alt Faktörü Puanları Arasında Fark Var mıdır? Araştırma Sorusuna Ait Bulgular

Bu başlık altında özgüven yeterliği alt faktör son test puanlarının deney gruplarına göre değişimi ile ilgili betimsel istatistik sonuçları verilmiştir. Ardından programlama eğitimini farklı öğretim yöntemlerine göre alan deney gruplarının uygulamalar sonrasındaki özgüven yeterliği alt faktör puanları arasında fark olup olmadığını gözlemlemek için Kruskal-Wallis H Testi uygulanmıştır.

Tablo 29: Özgüven Yeterliği Alt Faktör Son Test Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları

Puan	Gruplar	n	Ortalama	Ortanca	Varyans	Ss	Min	Maks
Özgüven	Blok	137	6.94	6.00	5.82	2.41	5	15
Yeterliği Alt	Robotik	32	5.90	5.00	2.86	1.69	5	12
Faktörü Son	Metin	33	6.54	6.00	3.94	1.98	5	11
Test Puanı								

Grupların aritmetik ortalamaları dikkate alındığında, uygulama sonrasında en düşük özgüven yeterliği alt faktör puanına ve en yüksek özgüven yeterlik düzeyine robotik programlama grubundaki öğrencilerin sahip olduğu görülmüştür. Metin tabanlı programlama grubundaki öğrencilerin, blok tabanlı programlama grubundaki öğrencilere göre daha düşük özgüven yeterliği alt faktör puanına yani daha yüksek özgüven yeterlik düzeyine sahip olduğu bir diğer analiz sonucudur.

Programlama eğitimini farklı öğretim yöntemlerine göre alan deney gruplarının uygulamalar sonrasındaki özgüven yeterliği alt faktör puanlarının Kruskal-Wallis Testi sonuçları Tablo 30’da verilmiştir.

Tablo 30: Özgüven Yeterliği Alt Faktör Son Test Puanlarının Gruplara Göre Karşılaştırılması, Kruskal-Wallis Test Sonuçları

Puan	Gruplar	n	Sıra Ort.	sd	x ²	p
Özgüven Yeterliği Alt Faktörü Son Test Puanı	Blok	137	108.03	2	8.20	.017
	Robotik	32	76.92			
	Metin	33	98.23			

Analiz sonuçları, deneysel çalışmaya katılan öğrencilerin özgüven yeterliği alt faktör puanlarının, aldıkları öğretim yöntemine göre anlamlı bir şekilde farklılaştığını göstermektedir, χ^2 (sd=2, n=202) =8.20, p=.017. Bu bulgu, kullanılan üç yöntemin öğrencilerin özgüven yeterliği algılarını artırmada farklı etkilere sahip olduğunu göstermiştir.

Tablo 30’da görülen bu anlamlı farkın hangi gruplar arasında gerçekleştiğini ortaya koymak adına, Kruskal Wallis H testinin post hoc yöntemlerinden ikili karşılaştırmalar yapılmıştır.

Tablo 31: Özgüven Yeterlik Algısı Son Test Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili İkili Karşılaştırma Sonuçları

Grup 1	Grup 2	Standart Test İstatistiği	Standart Hata	p	Ayarlanmış p
Robotik	Blok	2.84	10.95	.005	.014
Robotik	Metin	-1.54	13.83	.124	.371
Metin	Blok	.906	10.81	.365	1.000

İkili karşılaştırma sonuçları robotik programlama eğitimi alan grup ile blok tabanlı programlama eğitimi alan grup arasında anlamlı bir fark olduğunu ortaya koymuştur (p=.014). Robotik programlama eğitimi alan grubun özgüven yeterlik düzeylerinin, blok tabanlı programlama eğitimi alan grubun özgüven yeterlik düzeylerine göre anlamlı olarak daha yüksek olduğu görülmüştür.

4.4.7. Öğrencilerin Katıldıkları Programlama Öğretim Yöntemine Göre Programlama Başarıları Arasında Fark Var mıdır? Araştırma Sorusuna Ait Bulgular

Bu başlık altında programlama başarı puanlarının deney gruplarına göre değişimi ile ilgili betimsel istatistik sonuçları verilmiştir. Ardından programlama eğitimini farklı öğretim yöntemlerine göre alan deney gruplarının uygulamalar sonrasındaki programlama başarı puanları arasında fark olup olmadığını gözlemlemek için Kruskal-Wallis H Testi uygulanmıştır.

Tablo 32: Programlama Başarı Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları

Puan	Gruplar	n	Ortalama	Ortanca	Varyans	Ss	Min	Maks
Programlama	Blok	137	91.49	95.00	84.69	9.20	65	100
Başarı Puanı	Robotik	32	95.93	100.00	36.19	6.01	80	100
	Metin	33	92.42	95.00	111.12	10.54	65	100

Grupların aritmetik ortalamaları dikkate alındığında, uygulama sonrasında en yüksek programlama başarı puanına robotik programlama grubundaki öğrencilerin sahip olduğu görülmüştür. Metin tabanlı programlama grubundaki öğrencilerin, blok tabanlı programlama grubundaki öğrencilere göre daha yüksek programlama başarı puanına sahip olduğu bir diğer analiz sonucudur.

Programlama eğitimini farklı öğretim yöntemlerine göre alan deney gruplarının uygulamalar sonrasındaki programlama başarı puanlarının Kruskal-Wallis Testi sonuçları Tablo 33'te verilmiştir.

Tablo 33: Programlama Başarı Puanlarının Gruplara Göre Karşılaştırılması, Kruskal-Wallis Test Sonuçları

Puan	Gruplar	n	Sıra Ort.	sd	χ^2	p
Programlama Başarı Puanı	Blok	137	95.42	2	6.73	.035
	Robotik	32	123.31			
	Metin	33	105.59			

Analiz sonuçları, deneysel çalışmaya katılan öğrencilerin programlama başarı puanlarının, aldıkları öğretim yöntemine göre anlamlı bir şekilde farklılaştığını göstermektedir, χ^2 (sd=2, n=202) =6.73, p=.035. Bu bulgu, kullanılan üç yöntemin öğrencilerin programlama başarı puanlarını artırmada farklı etkilere sahip olduğunu göstermiştir.

Tablo 33'te görülen bu anlamlı farkın hangi gruplar arasında gerçekleştiğini ortaya koymak adına, Kruskal Wallis H testinin post hoc yöntemlerinden ikili karşılaştırmalar yapılmıştır. Sonuçları Tablo 34'te verilmiştir.

Tablo 34: Programlama Başarı Puanlarının Deney Gruplarına Göre Farklılaşma Durumu ile İlgili İkili Karşılaştırma Sonuçları

Grup 1	Grup 2	Standart Test İstatistiği	Standart Hata	p	Ayarlanmış p
Robotik	Blok	2.55	10.92	.011	.032
Robotik	Metin	-1.28	13.80	.199	.598
Metin	Blok	.943	10.79	.346	1.000

İkili karşılaştırma sonuçları robotik programlama eğitimi alan grup ile blok tabanlı programlama eğitimi alan grup arasında anlamlı bir fark olduğunu ortaya koymuştur ($p=.032$). Robotik programlama eğitimi alan grubun programlama başarı düzeylerinin, blok tabanlı programlama eğitimi alan grubun programlama başarı düzeylerine göre anlamlı olarak daha yüksek olduğu görülmüştür.

4.5. Öğrencilerin Cinsiyetlerine Göre Yapılacak Analizlere Ait Bulgular

4.5.1. Öğrencilerin Cinsiyetlerine Göre Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algıları Son Test Puanları Farklılaşmakta Mıdır? Araştırma Sorusuna Ait Bulgular

Bu başlık altında BİDBÖA son test puanlarının cinsiyete göre değişimi ile ilgili betimsel istatistik sonuçları verilmiştir. Ardından BİDBÖA son test puanlarının cinsiyete göre farklı olup olmadığını gözlemlemek için Kruskal-Wallis H Testi uygulanmıştır.

Tablo 35: BİDBÖA Son Test Puanlarının Cinsiyete Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları

Puan	Gruplar	n	Ortalama	Ortanca	Varyans	Ss	Min	Maks
Öz Yeterlik	Erkek	90	48.87	47.50	142.96	11.95	36	87
Son Test Puanı	Kız	112	48.73	47.00	105.94	10.29	36	79

Grupların uygulama sonrasında BİDBÖA son test puanları dikkate alındığında, puanlar birbirine çok yakın olmakla beraber kız öğrencilerin erkek öğrencilere göre daha düşük öz yeterlik algı puanına yani daha yüksek öz güven yeterlik düzeyine sahip olduğu görülmektedir.

Uygulamalar sonrasındaki bilgi BİDBÖA son test puanlarının cinsiyete göre analizini içeren Kruskal-Wallis Testi sonuçları Tablo 36’da verilmiştir.

Tablo 36: BİDBÖA Son Test Puanlarının Cinsiyete Göre Cinsiyete Göre Kruskal Wallis H Testi Sonuçları

Puan	Gruplar	n	Sıra Ort.	sd	χ^2	p
Öz Yeterlik	Erkek	90	99.90	1	.12	.727
Son Test Puanı	Kız	112	102.79			

Analiz sonuçları, deneysel çalışmaya katılan öğrencilerin BİD becerileri öz yeterlik algısı ölçeğinden aldığı puanların, cinsiyete göre anlamlı bir şekilde farklılaşmadığını göstermektedir, χ^2 (sd=1, n=202) =.12, p=.727.

4.5.2. Öğrencilerin Cinsiyetlerine Göre Algoritma Tasarlama Yeterliği Alt Faktörü Puanları Farklılaşmakta Mıdır? Araştırma Sorusuna Ait Bulgular

Bu başlık altında algoritma tasarlama yeterliği alt faktörü son test puanlarının cinsiyete göre değişimi ile ilgili betimsel istatistik sonuçları verilmiştir. Ardından algoritma tasarlama yeterliği alt faktörü son test puanlarının cinsiyete göre farklı olup olmadığını gözlemlemek için Kruskal-Wallis H Testi uygulanmıştır.

Tablo 37: Algoritma Tasarlama Yeterliği Alt Faktörü Son Test Puanlarının Cinsiyete Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları

Puan	Gruplar	n	Ortalama	Ortanca	Varyans	Ss	Min	Maks
Algoritma	Erkek	90	11.18	10.00	8.53	2.92	9	19
Tasarlama	Kız	112	11.75	11.00	8.82	2.96	9	21
Yeterliği Alt Faktörü Son Test Puanı								

Grupların uygulama sonrasında algoritma tasarlama yeterliği alt faktörü puanları dikkate alındığında, erkek öğrencilerin kız öğrencilere göre daha düşük algoritma tasarlama yeterliği puanına yani daha yüksek algoritma tasarlama yeterliği düzeyine sahip olduğu görülmektedir.

Uygulamalar sonrasındaki algoritma tasarlama yeterliği alt faktörü puanlarının cinsiyete göre analizini içeren Kruskal-Wallis Testi sonuçları Tablo 38’de verilmiştir.

Tablo 38: *Algoritma Tasarlama Yeterliği Alt Faktörü Son Test Puanlarının Cinsiyete Göre Cinsiyete Göre Kruskal Wallis H Testi Sonuçları*

Puan	Gruplar	n	Sıra Ort.	sd	χ^2	p
Algoritma Tasarlama	Erkek	90	95.59	1	1.74	.186
Yeterliği Son Test	Kız	112	106.25			
Puanı						

Analiz sonuçları, deneysel çalışmaya katılan öğrencilerin algoritma tasarlama yeterliği alt faktörü puanlarının, cinsiyete göre anlamlı bir şekilde farklılaşmadığını göstermektedir, χ^2 (sd=1, n=202) =1.74, p=.186.

4.5.3. Öğrencilerin Cinsiyetlerine Göre Problem Çözme Yeterliği Alt Faktörü Puanları Farklılaşmakta Mıdır? Araştırma Sorusuna Ait Bulgular

Bu başlık altında problem çözme yeterliği alt faktörü son test puanlarının cinsiyete göre değişimi ile ilgili betimsel istatistik sonuçları verilmiştir. Ardından problem çözme yeterliği alt faktörü son test puanlarının cinsiyete göre farklı olup olmadığını gözlemlemek için Kruskal-Wallis H Testi uygulanmıştır.

Tablo 39: *Problem Çözme Yeterliği Alt Faktörü Son Test Puanlarının Cinsiyete Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları*

Puan	Gruplar	n	Ortalama	Ortanca	Varyans	Ss	Min	Maks
Problem Çözme	Erkek	90	13.46	12.00	14.04	3.74	10	28
Yeterliği Alt	Kız	112	12.70	12.00	9.77	3.12	10	23
Faktörü Son Test								
Puanı								

Grupların uygulama sonrasında problem çözme yeterliği alt faktörü puanları dikkate alındığında, kız öğrencilerin erkek öğrencilere göre daha düşük problem çözme yeterliği puanına yani daha yüksek problem çözme yeterliği düzeyine sahip olduğu görülmektedir.

Uygulamalar sonrasındaki problem çözme yeterliği alt faktörü puanlarının cinsiyete göre analizini içeren Kruskal-Wallis Testi sonuçları Tablo 40'ta verilmiştir.

Tablo 40: *Problem Çözme Yeterliği Alt Faktörü Son Test Puanlarının Cinsiyete Göre Cinsiyete Göre Kruskal Wallis H Testi Sonuçları*

Puan	Gruplar	n	Sıra Ort.	sd	χ^2	p
Problem Çözme Yeterliği Son Test Puanı	Erkek	90	107.39	1	1.72	.189
	Kız	112	96.76			

Analiz sonuçları, deneysel çalışmaya katılan öğrencilerin problem çözme yeterliği alt faktörü puanlarının, cinsiyete göre anlamlı bir şekilde farklılaşmadığını göstermektedir, χ^2 (sd=1, n=202) =1.72, p=.189.

4.5.4. Öğrencilerin Cinsiyetlerine Göre Veri İşleme Yeterliği Alt Faktörü Puanları Farklılaşmakta Mıdır? Araştırma Sorusuna Ait Bulgular

Bu başlık altında veri işleme yeterliği alt faktörü son test puanlarının cinsiyete göre değişimi ile ilgili betimsel istatistik sonuçları verilmiştir. Ardından veri işleme yeterliği alt faktörü son test puanlarının cinsiyete göre farklı olup olmadığını gözlemlemek için Kruskal-Wallis H Testi uygulanmıştır.

Tablo 41: *Veri İşleme Yeterliği Alt Faktörü Son Test Puanlarının Cinsiyete Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları*

Puan	Gruplar	n	Ortalama	Ortanca	Varyans	Ss	Min	Maks
Veri İşleme Yeterliği Alt Faktörü Son Test Puanı	Erkek	90	9.32	9.00	6.42	2.53	7	17
	Kız	112	9.64	9.00	6.66	2.58	7	17

Grupların uygulama sonrasında veri işleme yeterliği alt faktörü puanları dikkate alındığında, erkek öğrencilerin kız öğrencilere göre daha düşük veri işleme yeterliği puanına yani daha yüksek veri işleme yeterliği düzeyine sahip olduğu görülmektedir.

Uygulamalar sonrasındaki veri işleme yeterliği alt faktörü puanlarının cinsiyete göre analizini içeren Kruskal-Wallis Testi sonuçları Tablo 42’de verilmiştir.

Tablo 42: *Veri İşleme Yeterliği Alt Faktörü Son Test Puanlarının Cinsiyete Göre Cinsiyete Göre Kruskal Wallis H Testi Sonuçları*

Puan	Gruplar	n	Sıra Ort.	sd	χ^2	p
Veri İşleme Yeterliği Son Test Puanı	Erkek	90	96.81	1	1.08	.297
	Kız	112	105.27			

Analiz sonuçları, deneysel çalışmaya katılan öğrencilerin veri işleme yeterliği alt faktörü puanlarının, cinsiyete göre anlamlı bir şekilde farklılaşmadığını göstermektedir, χ^2 (sd=1, n=202) =1.08, p=.297.

4.5.5. Öğrencilerin Cinsiyetlerine Göre Temel Programlama Yeterliği Alt Faktörü Puanları Farklılaşmakta Mıdır? Araştırma Sorusuna Ait Bulgular

Bu başlık altında temel programlama yeterliği alt faktörü son test puanlarının cinsiyete göre değişimi ile ilgili betimsel istatistik sonuçları verilmiştir. Ardından temel programlama yeterliği alt faktörü son test puanlarının cinsiyete göre farklı olup olmadığını gözlemlemek için Kruskal-Wallis H Testi uygulanmıştır.

Tablo 43: *Temel Programlama Yeterliği Alt Faktörü Son Test Puanlarının Cinsiyete Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları*

Puan	Gruplar	n	Ortalama	Ortanca	Varyans	Ss	Min	Maks
Temel Programlama Yeterliği Alt Faktörü Son Test Puanı	Erkek	90	7.91	7.00	8.32	2.88	5	15
	Kız	112	8.13	7.00	7.39	2.71	5	15

Grupların uygulama sonrasında temel programlama yeterliği alt faktörü puanları dikkate alındığında, erkek öğrencilerin kız öğrencilere göre daha düşük temel programlama yeterliği puanına yani daha yüksek temel programlama yeterliği düzeyine sahip olduğu görülmektedir.

Uygulamalar sonrasındaki temel programlama yeterliği alt faktörü puanlarının cinsiyete göre analizini içeren Kruskal-Wallis Testi sonuçları Tablo 44’te verilmiştir.

Tablo 44: *Temel Programlama Yeterliği Alt Faktörü Son Test Puanlarının Cinsiyete Göre Cinsiyete Göre Kruskal Wallis H Testi Sonuçları*

Puan	Gruplar	n	Sıra Ort.	sd	χ^2	p
Temel Programlama	Erkek	90	97.46	1	.79	.373
Yeterliği Son Test	Kız	112	104.75			
Puanı						

Analiz sonuçları, deneysel çalışmaya katılan öğrencilerin temel programlama yeterliği alt faktörü puanlarının, cinsiyete göre anlamlı bir şekilde farklılaşmadığını göstermektedir, χ^2 (sd=1, n=202) =.79, p=.373.

4.5.6. Öğrencilerin Cinsiyetlerine Göre Özgüven Yeterliği Alt Faktörü Puanları Farklılaşmakta Mıdır? Araştırma Sorusuna Ait Bulgular

Bu başlık altında özgüven yeterliği alt faktörü son test puanlarının cinsiyete göre değişimi ile ilgili betimsel istatistik sonuçları verilmiştir. Ardından özgüven yeterliği alt faktörü son test puanlarının cinsiyete göre farklı olup olmadığını gözlemlemek için Kruskal-Wallis H Testi uygulanmıştır.

Tablo 45: *Özgüven Yeterliği Alt Faktörü Son Test Puanlarının Cinsiyete Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları*

Puan	Gruplar	n	Ortalama	Ortanca	Varyans	Ss	Min	Maks
Özgüven	Erkek	90	6.98	6.00	7.02	2.64	5	15
Yeterliği Alt	Kız	112	6.50	6.00	3.60	1.89	5	15
Faktörü Son								
Test Puanı								

Grupların uygulama sonrasında özgüven yeterliği alt faktörü puanları dikkate alındığında, kız öğrencilerin erkek öğrencilere göre daha düşük özgüven yeterliği puanına yani daha yüksek özgüven yeterliği düzeyine sahip olduğu görülmektedir.

Uygulamalar sonrasındaki özgüven yeterliği alt faktörü puanlarının cinsiyete göre analizini içeren Kruskal-Wallis Testi sonuçları Tablo 46’da verilmiştir.

Tablo 46: Özgüven Yeterliği Alt Faktörü Son Test Puanlarının Cinsiyete Göre Cinsiyete Göre Kruskal Wallis H Testi Sonuçları

Puan	Gruplar	n	Sıra Ort.	sd	χ^2	p
Temel Programlama	Erkek	90	103.30	1	.16	.681
Yeterliği Son Test	Kız	112	100.05			
Puanı						

Analiz sonuçları, deneysel çalışmaya katılan öğrencilerin özgüven yeterliği alt faktörü puanlarının, cinsiyete göre anlamlı bir şekilde farklılaşmadığını göstermektedir, χ^2 (sd=1, n=202) = .16, p=.681.

4.5.7. Öğrencilerin Cinsiyetlerine Göre Programlama Başarı Puanları Farklılaşmakta Mıdır? Araştırma Sorusuna Ait Bulgular

Bu başlık altında programlama başarı puanlarının cinsiyete göre değişimi ile ilgili betimsel istatistik sonuçları verilmiştir. Ardından programlama başarı puanlarının cinsiyete göre farklı olup olmadığını gözlemlemek için Kruskal-Wallis H Testi uygulanmıştır.

Tablo 47: Programlama Başarı Puanlarının Cinsiyete Göre Farklılaşma Durumu ile İlgili Betimsel İstatistik Sonuçları

Puan	Gruplar	n	Ortalama	Ortanca	Varyans	Ss	Min	Maks
Programlama	Erkek	90	92.44	95.00	81.59	9.03	65	100
Başarı Puanı	Kız	112	92.27	95.00	85.08	9.22	65	100

Grupların uygulama sonrasında programlama başarı puanları dikkate alındığında, erkek öğrencilerin kız öğrencilere göre daha yüksek programlama başarı puanına sahip olduğu görülmektedir.

Uygulamalar sonrasındaki programlama başarı puanlarının cinsiyete göre analizini içeren Kruskal-Wallis Testi sonuçları Tablo 48’de verilmiştir.

Tablo 48: Programlama Başarı Puanlarının Cinsiyete Göre Cinsiyete Göre Kruskal Wallis H Testi Sonuçları

Puan	Gruplar	n	Sıra Ort.	sd	χ^2	p
Programlama	Erkek	90	102.26	1	.30	.863
Başarı Puanı	Kız	112	100.89			

Analiz sonuçları, deneysel çalışmaya katılan öğrencilerin özgüven yeterliği alt faktörü puanlarının, cinsiyete göre anlamlı bir şekilde farklılaşmadığını göstermektedir, χ^2 (sd=1, n=202) =.30, p=.863.

BÖLÜM V: SONUÇ

5.1. Sonuçlar ve Tartışma

Bu bölümde, araştırma sorularına ait analizler ile ulaşılan sonuçlar paylaşılmıştır. Bu sonuçların alanyazındaki çalışmalarla kıyaslanarak tartışılmıştır.

“Öğrencilerin bilgi işlemsel düşünme becerisine yönelik öz yeterlik algıları ön test ve son test puanları arasında blok tabanlı programlama öğretim yöntemini gören öğrenciler için fark var mıdır?” (1.a.) araştırma sorusu ile ilgili analizlerin sonucuna göre, blok tabanlı programlama öğretim yöntemini gören öğrencilerin BİD becerisine yönelik öz yeterlik algıları ön test ve son test puanları arasında anlamlı bir farklılık gözlemlenmiştir. Buna göre, blok tabanlı programlama eğitiminin öğrencilerin BİD becerisi öz yeterlik algısına olumlu etkisi olduğu düşünülebilir. BİD ile ilgili alanyazın taraması yapıldığında araştırmacılar tarafından yoğun olarak merak edilen konulardan birinin de blok tabanlı programlama eğitiminin BİD’e olan etkisi olduğunu görmekteyiz. Bu çalışmada ortaya çıkan blok tabanlı programlama eğitiminin BİD becerisi öz yeterlik algısına olumlu etkisi farklı çalışmalarda da gözlemlenmiştir. Bu sonucu destekleyen sonuçlara sahip olan Feldhausen vd.’ne (2018) ait çalışmada benzer bir programlama eğitimi uygulanıp, BİD becerilerine yönelik öz yeterlik algısında olumlu etki gözlemlenmiştir. Alanyazında blok tabanlı programlama eğitiminin BİD becerisine olumlu etkisinin gözlemlendiği başka çalışmalar da bulunmaktadır (Chiazzese vd., 2018; Denner vd., 2014; Lin vd., 2018; Oluk ve Korkmaz, 2016; Sakamoto vd., 2013). Olumlu sonuçların yanı sıra blok tabanlı programlama eğitiminin BİD üzerinde anlamlı bir etki bırakmadığı çalışmalar da bulunmaktadır (Gülbahar ve Kalelioğlu, 2014; Lockwood ve Mooney, 2017; Uslu vd., 2018). Bu çalışmada görülen olumlu etkinin diğer çalışmalarda görülmemesinin birkaç sebebi olabilir. Kıyasladığımız çalışmaların da benzer yaş gruplarıyla uygulama yaptığı görüldüğünden, öğretim aracının farklılığının, eğitime ayrılan süre farklılığının ve fiziksel imkan farklılığının bu farklı sonuçların alınmasında etkili olabileceği söylenebilir.

“Öğrencilerin bilgi işlemsel düşünme becerisine yönelik öz yeterlik algıları ön test ve son test puanları arasında metin tabanlı programlama öğretim yöntemini gören

öğrenciler için fark var mıdır?” (1.b.) araştırma sorusu ile ilgili analizlerin sonucuna göre, metin tabanlı programlama öğretim yöntemini gören öğrencilerin BİD becerisine yönelik öz yeterlik algıları ön test ve son test puanları arasında anlamlı bir farklılık gözlemlenmiştir. Buna göre, metin tabanlı programlama eğitiminin öğrencilerin bilgi işlemsel düşünme becerisi öz yeterlik algısına olumlu etkisi olduğu düşünülebilir. Feldhausen vd. (2018) yaptıkları çalışmada bir grup öğrenciye önce scratch yazılımı ile blok tabanlı programlama ardından fiziksel programlama platformu Arduino ve yazılımı üzerinde metin tabanlı programlama yaptırmışlardır. Eğitime katılan öğrencilerin BİD becerilerine yönelik öz yeterlik algısında olumlu etki gözlemlenmiştir. Bu yönüyle metin tabanlı programlama grubundaki olumlu etki ve Feldhausen vd.’nin (2018) yaptığı çalışma sonuçları örtüşmektedir. Bununla birlikte alanyazında metin tabanlı programlama eğitiminin BİD’e olan etkisini araştıran çalışmaların azlığı -genellikle blok tabanlı ve robotik üzerine araştırmalar bulunmakta- sebebiyle Hambruch’un (2009) üniversitede okutulan “bilgi işlemsel düşünmeye giriş” isimli dersin öğrencilerin bilgisayar bilimlerine olan ilgisini arttırdığı sonucuyla bu çalışmanın sonucunun nispeten örtüştüğü söylenebilir. Erken yaşlarda metin tabanlı programlama ve BİD ile ilgili çalışmaların azlığının sözdizimi zorluklarından kaynaklandığı düşünüldüğünde, Dorling ve White (2015) ve Park ve Yoo (2018) gibi çalışmalarında bahsedilen ilkökul ve ortaokul öğrencilerinin metin tabanlı programlama konusunda başarılı olabilir yorumu bu çalışmadaki sonuçlarla uyum göstermektedir.

İlk iki araştırma sorusu ile ilgili analiz sonuçlarındaki blok veya metin tabanlı programlamanın BİD becerisi yönelik öz yeterlik algısı üzerindeki olumlu etkisi beraber yorumlandığında, Kwon ve Kim’in (2018) Kore’de 205 üniversite öğrencisi ile yaptıkları çalışma ile benzer sonuçlar alındığı görülmektedir. Çalışmada programlama eğitiminde; bilgisayarsız etkinlikler ve blok tabanlı programlama etkinlikleri ile birlikte metin tabanlı programlama dili olan python gösterilmiştir. BİD becerilerinin entegre edildiği programlama eğitimi sonunda, öğrencilerde BİD ve programlama ile ilgili öğrenme memnuniyetinde, öz yeterlikte ve gelecekte bu alanda meslek seçimine karşı oluşan olumlu değişikliğin bu çalışma sonuçlarını desteklediği görülmektedir.

“Öğrencilerin bilgi işlemsel düşünme becerisine yönelik öz yeterlik algıları ön test ve son test puanları arasında robotik programlama öğretim yöntemini gören öğrenciler için fark var mıdır?” (1.c.) araştırma sorusu ile ilgili analizlerin sonucuna göre, robotik

programlama öğretim yöntemini gören öğrencilerin BİD becerisine yönelik öz yeterlik algıları ön test ve son test puanları arasında anlamlı bir farklılık gözlemlenmiştir. Buna göre, robotik programlama eğitiminin öğrencilerin BİD becerisi öz yeterlik algısına olumlu etkisi olduğu düşünülebilir. Bu çalışmadaki gibi Robotis firmasına ait eğitsel robotik setleri ve Roboplus Task programı ile yapılan bir çalışmada da benzer sonuçlar bulunmuştur. Vieira, Penmetcha, Magana ve Matson'un (2016) 44 üniversite öğrencisi ile yaptıkları çalışma robotik programlamanın BİD ve ilgili becerileri geliştirmede yararlı olduğunu göstermektedir. Farklı öğrenci seviyelerinde yapılan çalışmalar da robotik programlama eğitiminin BİD ve ilgili beceriler üzerinde olumlu etkileri olduğunu aktarmaktadırlar (Berland vd., 2013; Berland ve Wilensky, 2015; Bers, 2008; Bers vd., 2014; Chaudhary vd., 2016). Robotik programlama faaliyetleri sonunda çıkan somut ürünün ve öğrencilerin bu somut ürünü programlayıp sonucunu eş zamanlı görmeleri çalışmalardan çıkan olumlu sonuçların sebeplerini açıklamaktadır. Robotların, öğrenciler için ilgi çekici olması da bu olumlu sonuçlarda pay sahibi olmuş olabilir.

“Öğrencilerin katıldıkları programlama öğretim yöntemine göre bilgi işlemsel düşünme becerisine yönelik öz yeterlik algıları son test puanları arasında fark var mıdır?” (2.a.) araştırma sorusu ile ilgili analizlerin sonucuna göre, öğrencilerin katıldıkları programlama öğretim yöntemine göre BİD becerisine yönelik öz yeterlik algı puanları arasında anlamlı bir farklılık gözlemlenmiştir. Buna göre, robotik programlama eğitimi alan öğrencilerin BİD becerisi öz yeterlik algı seviyelerinin, blok tabanlı programlama eğitimi alan öğrencilerden anlamlı olarak yüksek olduğu düşünülebilir. Robotik programlama eğitimi alan öğrencilerin, metin tabanlı programlama eğitimi alan öğrencilere göre yüksek öz yeterlik algısına sahip oldukları görülmesine rağmen istatistiksel olarak anlamlı bir fark gözlemlenmemiştir. Aynı şekilde metin tabanlı programlama eğitimi alan öğrencilerin, blok tabanlı programlama eğitimi alan öğrencilere göre yüksek öz yeterlik algısına sahip oldukları görülmesine rağmen istatistiksel olarak anlamlı bir fark gözlemlenmemiştir. Türkiye’de 6. Sınıf öğrencileri ile yapılan çalışmada arduino ile robotik çalışan öğrencilerin, blok tabanlı programlama yapan öğrencilere göre daha yüksek BİD beceri seviyesine sahip olduğu görülmüştür (Karaahmetoğlu ve Korkmaz, 2019). Benzer sonuçlara ulaşılan başka bir çalışmada ise 9.sınıfa giden öğrencilerden oluşan blok tabanlı ve robotik programlama yapan grup ile

sadece blok tabanlı programlama yapan grubun BİD becerileri özyeterlik düzeyleri karşılaştırılmıştır. Blok ve robotik programlama yapan grubun BİD becerileri özyeterlik düzeylerinin sadece blok tabanlı programlama yapan grubun düzeyine göre anlamlı olarak daha yüksek olduğu görülmüştür (Saritepeci ve Durak, 2017). Bu sonuçlar da göstermektedir ki programlama çıktılarının robot vb. fiziki bir nesne üzerinde somut olarak görülmesi öğrencilerin BİD becerilerini diğer yöntemlere göre daha fazla geliştirmektedir. Bu yönleriyle araştırmadan çıkan sonuçlar farklı araştırmaların sonuçları ile benzer sonuçlar göstermektedir. Bu sonuçların yanı sıra farklı programlama yöntemlerinin uygulandığı grupların BİD becerisi başarı puanları ve BİD becerisi özyeterlik algı seviyelerinin anlamlı fark göstermediği çalışmalar da bulunmaktadır. Yolcu'ya (2018) ait çalışmada blok tabanlı programlama yapan grup ile robotik programlama yapan grubun BİD becerisi ve özyeterlik algı seviyelerinin istatistiksel olarak anlamlı bir farklılık göstermediği belirtilmiştir. Djambong ve Freiman (2016) yaptıkları çalışmada blok tabanlı programlama yapan grup ile robotik programlama yapan grup arasında BİD beceri seviyeleri açısından anlamlı bir fark gözlemlenmemiştir. Bu çalışmada görülen anlamlı farkın bazı çalışmalarda görülmemesini farklı eğitsel robotik set kullanımı, farklı blok tabanlı programlama aracı seçimi ve öğrenci sınıf ve yaş seviyesi farklılığı gibi faktörlerden kaynaklandığı söylenebilir.

Öğrencilerin katıldıkları programlama öğretim yöntemine göre,

- algoritma tasarlama yeterliği alt faktör puanları (2.b.),
- problem çözme yeterliği alt faktör puanları (2.c.),
- veri işleme yeterliği alt faktör puanları (2.d.),
- temel programlama yeterliği alt faktör puanları (2.e.) arasında anlamlı bir farklılık gözlemlenmemiştir.

Feldhausen vd. (2018) yaptıkları çalışmada farklı gruplara bilgisayarsız etkinlikler, blok taban programlama veya fiziksel programlama platformu ile hem blok hem de metin tabanlı programlama yaptırmışlardır. BİD'in algoritmik düşünme, soyutlama, veri, problemleri ayrıştırma gibi alt boyutlarına yönelik öğrenci öz yeterlik algı seviyelerini kontrol etmişlerdir. Programlama eğitimlerinin her birinin öz yeterlik algı seviyelerini yükselttiğini ama aralarında anlamlı bir fark oluşturmadığını ifade etmiştir.

Karaahmetoğlu ve Korkmaz’a (2019) ait çalışmada robotik programlama eğitimi alan grup ile blok tabanlı programlama eğitimi alan grup arasında BİD becerisi ve problem çözme alt faktöründe robotik lehine anlamlı fark görülmüştür, fakat diğer alt faktörlerde anlamlı fark görülmemiştir. Bu çalışmada alınan sonuçlar ile bahsi geçen çalışma kıyaslandığında problem çözme alt faktöründe oluşan anlamlı fark dışında diğer alt faktörlere ait sonuçlar benzerlik göstermektedir. Sarıtepeci ve Durak’ın (2017) çalışmasından elde edilen sonuçlar incelendiğinde robotik programlama eğitimi alan grubun BİD becerisini oluşturan tüm alt faktör puanlarının diğer gruplara göre anlamlı olarak daha yüksek olduğu görülmektedir. Bu yönleriyle bu çalışmada gözlemlenmeyen sonuçlar incelenen çalışmada anlamlı farklılık olarak karşımıza çıkmaktadır. İki çalışmada arasında oluşan bu farkın katılımcılar arasındaki yaş ve sınıf farkından (5.sınıf öğrencileri - 9.sınıf öğrencileri) ve kullanılan değerlendirme araçları farkından kaynaklandığı düşünülmektedir.

“Öğrencilerin katıldıkları programlama öğretim yöntemine göre özgüven yeterliği alt faktör puanları arasında fark var mıdır?” (2.f.) araştırma sorusu ile ilgili analizlerin sonucuna göre, öğrencilerin katıldıkları programlama öğretim yöntemine göre özgüven yeterliği alt faktör puanları arasında anlamlı bir farklılık gözlemlenmiştir. Buna göre, robotik programlama eğitimi alan öğrencilerin özgüven yeterlik algı seviyelerinin, blok tabanlı programlama eğitimi alan öğrencilerden anlamlı olarak yüksek olduğu anlaşılmıştır. Gruplar arası diğer karşılaştırmalarda ise anlamlı farklılıklar görülmemiştir. Bu anlamlı farkın robotik faaliyetlerin öğrencinin motivasyonunu ve ilgisi artıran yapısından kaynaklı olduğu düşünülebilir. Ayrıca öğrencinin somut olarak bir ürün oluşturmamasının o konuya dair özgüvenini daha da artırabileceği öngörülebilmektedir.

“Öğrencilerin katıldıkları programlama öğretim yöntemine göre programlama bilgisi başarıları arasında fark var mıdır?” (2.g.) ile ilgili analizlerin sonucuna göre, öğrencilerin katıldıkları programlama öğretim yöntemine göre programlama başarıları arasında anlamlı bir farklılık gözlemlenmiştir. Buna göre, robotik programlama eğitimi alan öğrencilerin programlama başarı seviyelerinin, blok tabanlı programlama eğitimi alan öğrencilerden anlamlı olarak yüksek olduğu anlaşılmıştır. Gruplar arası diğer karşılaştırmalarda ise anlamlı farklılıklar görülmemiştir. Atmatzidou, Demetriadis ve Nika (2018) öğrencilerin robotlarla etkileşime girmesinin temel programlama

kavramlarını anlamalarında etkili olduğunu belirtmiştir. Yolcu'ya (2018) ait çalışmada robotik programlama eğitimi alan grubun, blok tabanlı programlama eğitimi alan gruba göre programlama alanı akademik başarısının daha yüksek olduğundan bahsedilmektedir. Farklı çalışmalardan alınan bu bilgiler ve sonuçlar ile bu çalışmada ortaya çıkan sonuçlar robotik programlama eğitimi ile yazdıkları kodun somut bir ürünlerdeki karşılığını görmelerinin, öğrencilerin programlama bilgisini diğer yöntemlere göre daha fazla yükselttiğini göstermektedir.

Öğrencilerin cinsiyetlerine göre,

- bilgi işlemsel düşünme becerisine yönelik öz yeterlik algı puanları (3.a.)
- algoritma tasarlama yeterliği alt faktör puanları (3.b.)
- problem çözme yeterliği alt faktör puanları (3.c.)
- veri işleme yeterliği alt faktör puanları (3.d.)
- temel programlama yeterliği alt faktör puanları (3.e.)
- özgüven yeterliği alt faktör puanları (3.f.)
- programlama başarı puanları (3.g.) arasında anlamlı bir farklılık gözlemlenmemiştir.

Bu sonuçları destekleyen çalışmalar bulunmaktadır. Kore'de 86 ilkokul öğrencisi ile yapılan çalışmada programlama eğitimi alan öğrencilerin cinsiyetlerine göre BİD becerileri açısından anlamlı bir fark olmadığı görülmüştür (Lee, Jung, & Park, 2017). Başka bir çalışmada Yağcı (2018) 445 lise öğrencisinin BİD becerilerinin cinsiyete göre anlamlı olarak farklılık göstermediğini görmüştür. Vieira, Penmetcha, Magana ve Matson'un (2016) robotik programlamanın BİD'e olan etkisinin cinsiyete göre anlamlı olarak farklılaşmadığı sonucu aktarılmıştır. Diğer çalışmalar da BİD becerisinde cinsiyete göre farklılık görmemişlerdir (Atmatzidou ve Demetriadis, 2016; Hsu, 2014; Oluk ve Korkmaz, 2016; Uslu vd., 2018). Corradini vd. (2017) ise diğer çalışmalarda olduğu gibi erkek yaşlarda BİD becerisinin cinsiyete göre farklılaşmadığını fakat yaş ilerledikçe aradaki farkın erkek öğrencilerin lehine arttığını gözlemlemiştir. Bu sonuç programlama bilgisi eğitiminin ve BİD becerisinin erken yaşlarda başlanması gerektiğini göstermektedir. Uluslararası enformatik ve BİD etkinliğinin 2018 yılı Türkiye uygulama raporunda ortaokul düzeyinde kız öğrencilerin daha iyi puan aldığı fakat lise düzeyinde erkek öğrencilerin puanlarının kız öğrencileri geçtiği görülmektedir

(Bilge Kunduz, 2018). Bu sonuçlar erken yaşta bilgisayar bilimleri, programlama ve BİD konusunda cinsiyet arası farkın az olduğunu fakat ileri yaşlarda erkekler lehine farkın açılabilceğini göstermektedir. Bilgisayar bilimleri eğitimlerine erken yaşlarda başlamanın cinsiyet ayırt etmeksizin bu alandaki başarıyı ve ilgiyi artırabileceği görülmektedir.

5.2. Öneriler

Bu alanda çalışma yapacak eğitimcilere ve araştırmacılara öneriler şu şekildedir.

Eğitimcilere Öneriler

Bu çalışma alanda çalışan eğitimcilere fiziksel imkanlara bağlı olmadan farklı yöntemlerle bilgisayar bilimi eğitimi verilebileceğini örnek uygulamalar ve alanyazın taramasıyla göstermektedir. Bilgisayarsız bilgisayar bilimi (B3) etkinlikleri buna en güzel ve başarılı örneklerdir. Bununla birlikte fiziksel imkanlarının artmasının öğrencinin ilgi ve motivasyonunu artırdığı da bir gerçektir. Somut öğrenme çıktıları olan robotik eğitiminin hem bu çalışmada hem de alanyazındaki diğer çalışmalardaki olumlu etkisi eğitimciler tarafından dikkate alınmalıdır.

Araştırmacılara Öneriler

Robotik eğitiminin, programlama ve BİD becerisi başta olmak üzere farklı boyutlarda öğrencilere katkısı araştırılmaya açık bir alandır. Bu alan -özellikle ülkemizde- öğrenci yararına çalışma ve araştırmalara ihtiyaç duymaktadır. Araştırmacılar yapacakları çalışmalar ile bu alandaki boşluğu doldurabilirler.

Programlama eğitimi ve BİD ile ilgili çalışma yapacak araştırmacılara programlama eğitiminde seçilecek programlama dili ve aracının önemi göz önüne alındığında, farklı programlama dilleri ve araçları ile benzer çalışmalar yapılarak sonuçların karşılaştırılabileceği önerilmektedir.

Bilgisayarsız bilgisayar bilimi (B3) etkinliklerinin BİD becerilerini kazandırmadaki olumlu etkisini göz önüne alındığında B3 etkinliklerinin sürecin tamamına yayıldığı programlama eğitimleri sonunda öğrencilerin BİD beceri seviyeleri çeşitli değerlendirme araçları ile ölçülebilir.

BİD becerilerine ait değerlendirme araçlarının istenilen çeşitlilikte olmadığı farklı çalışmalarda bahsedilmektedir. Buradan yola çıkarak BİD becerilerini ölçebilecek geçerli ve güvenilir değerlendirme araçları geliştirme konusunda çalışmalar yapılabilir.

BİD ile ilgili alanyazına bakıldığında, k12 seviyesinde BİD ile ilgili çok araştırma bulunmasına rağmen, BİD'in yüksek öğretimdeki potansiyeli üzerine yapılan araştırmalara daha az ilgi duyulmuştur. BİD ile ilgili bu araştırmalardaki görülen eksiklikten yola çıkılarak, BİD'in farklı boyutlarını ele alan çalışmalar yüksek öğrenim gören öğrenciler ile de yapılabilir.

Metin tabanlı programlamanın -özellikle erken yaşta- BİD'e etkisine bakan çalışmaların azlığı alanyazında görülmektedir. Bu eksikliğe istinaden yeni çalışmalar yapılabilir.

Var olan çalışmalarda erken yaşlarda BİD ve programlama eğitimi alan kız ve erkek arasında seviye olarak fark yokken, hatta kız öğrencilerin önde olduğu çalışmalar bulunmaktayken, yaş ilerledikçe erkeklerin lehine dönen bir süreçten bahsedilmektedir. Erken yaşta bu eğitimlere başlayıp devam eden grup ile ileri yaşlarda böyle bir eğitime başlayan gruplar arasındaki fark gözlemlenebilir. Tabi ki bu çalışma uzun soluklu bir süreç gerektirecektir.

BÖLÜM VI: KAYNAKÇA

- Adleberg, B.M. (2013). *Scratch programming and remix culture: Gender differences in interaction and motivation for pre-adolescents* (Yayınlanmamış yüksek lisans tezi). Georgetown University, Washington, D.C.
- Aldağ, H., & Tekdal, M. (2015). Bilgisayar kullanımı ve programlama öğretiminde cinsiyet farklılıkları. Ağrıdağ, G. (Ed), *1. Uluslararası Çukurova Kadın Çalışmaları Kongresi* (ss. 236–243). Adana.
- Altaher, M., & Ferchichi, A. (2018). AlgoThink: An Algorithmic Computational Thinking Approach. *2018 JCCO Joint International Conference on ICT in Education and Training, International Conference on Computing in Arabic, and International Conference on Geocomputing (JCCO: TICET-ICCA-GECO)*, Tunisia / Hammamet, Tunisia, 1-7.
- Angeli, C., Voogt, J., Fluck, A., Webb, M., Cox, M., Malyn-Smith, J., & Zagami, J. (2016). A k-6 computational thinking curriculum framework: implications for teacher knowledge. *Educational Technology & Society*, 19(3), 47–57.
- Asimov, I. (2018). *I, Robot*. London: HarperCollins.
- Atmatzidou, S., & Demetriadis, S. (2016). Advancing students' computational thinking skills through educational robotics: A study on age and gender relevant differences. *Robotics and Autonomous Systems*, 75, 661-670.
- Atmatzidou, S., Demetriadis, S., & Nika, P. (2018). How Does the Degree of Guidance Support Students' Metacognitive and Problem Solving Skills in Educational Robotics?. *Journal of Science Education and Technology*, 27(1), 70-85.
- Barr, D., Harrison, J., & Conery, L. (2011). Computational thinking: A digital age skill for everyone. *Learning & Leading with Technology*, 38(6), 20-23.

- Barut, E., Tuğtekin, U., & Kuzu, A. (2016). Robot uygulamalar ile bilgi işlemsel düşünme becerilerine bakış. *3rd International Conference on New Trend in Education (ICNTE 2016)*.
- Bebras.org. (2019). *What does computational thinking involve?*. 13.02.2019 tarihinde <https://www.bebas.org> adresinden erişilmiştir.
- Bell, T., Alexander, J., Freeman, I., & Grimley, M. (2009). Computer science unplugged: School students doing real computing without computers. *Journal of Applied Computing and Information Technology*, 13(1), 20–29.
- Bell, T. (2018). Computer science in k-12 education: the big picture. *Olympiads in Informatics*, 12, 3–11.
- Benitti, F. B. V. (2012). Exploring the educational potential of robotics in schools: A systematic review. *Computers & Education*, 58(3), 978–988.
- Berland, M., Martin, T., Benton, T., Petrick Smith, C., & Davis, D. (2013). Using learning analytics to understand the learning pathways of novice programmers. *Journal of the Learning Sciences*, 22(4), 564–599.
- 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.
- Bers, M. U. (2008). *Blocks, robots and computers: Learning about technology in early childhood*. New York: Teacher's College Press.
- 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.
- Bilge Kunduz. (2018). *2018 Yılı Uygulama Sonuçları*. 19.03.2019 tarihinde http://www.bilgekunduz.org/wp-content/uploads/2019/04/bilgekunduz_2018_v4.pdf adresinden erişilmiştir.

- Bilge Kunduz. (2019). *Etkinlik hakkında*. 13.02.2019 tarihinde <https://www.bilgekunduz.org> adresinden erişilmiştir.
- Bocconi, S., Chiocciariello, A., Dettori, G., Ferrari, A., & Engelhardt, K. (2016). *Developing computational thinking in compulsory education – Implications for policy and practice*; EUR EN; doi:10.2791/792158
- Brackmann, C. P., Román-González, M., Robles, G., Moreno-León, J., Casali, A., & Barone, D. (2017). Development of computational thinking skills through unplugged activities in primary school. *In Proceedings of the 12th Workshop on Primary and Secondary Computing Education*. ACM, pp. 65–72.
- Brennan K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. *In Proceedings of the 2012 annual meeting of the American Educational Research Association, Vancouver, Canada*, 1-25.
- Büyüköztürk, Ş. (2009). *Sosyal bilimler için veri analizi el kitabı*. Ankara: Pegem Akademi.
- Caci, B., Cardaci, M., & Lund, H.H. (2003). Assessing educational robotics by the “Robot edutainment questionnaire”. *Technical report, The Maersk Mc-Kinney Moller Institute for Production Technology*, University of Southern Denmark.
- Calao, L. A., Moreno-Leon, J., Correa, H. E., & Robles, G. (2015). Developing Mathematical Thinking with Scratch. *Design for Teaching and Learning in a Networked World*, 9307, 17-27.
- Capek, K. (2013). *R.U.R. Rossum'un Evrensel Robotları*. (P. Öztürk, Çeviri). Ankara: Elips Kitap.
- Center for Computational Thinking Carnegie Mellon (2019). *What is Computational Thinking?*. 29.03.2019 tarihinde <https://www.cs.cmu.edu/~CompThink/> adresinden erişilmiştir.

- Chalmers, C. (2018). Robotics and computational thinking in primary school. *International Journal of Child-Computer Interaction*, 17, 93-100.
- Chaudhary, V., Agrawal, V., Sureka, P., & Sureka, A. (2016). An experience report on teaching programming and computational thinking to elementary level children using lego robotics education kit. In *2016 IEEE Eighth International Conference on Technology for Education (T4E)* (pp. 38-41). IEEE.
- Chiazzese, G., Fulantelli, G., Pipitone, V., & Taibi, D. (2018). Engaging Primary School Children in Computational Thinking: Designing and Developing Videogames. *Education in the Knowledge Society (EKS)*, 19(2), 63-81.
- Code.org. (2019). *About us*. 12.01.2019 tarihinde <https://code.org/international/about> adresinden erişilmiştir.
- Corradini, I., Lodi, M., & Nardelli, E. (2017). Computational thinking in italian schools: quantitative data and teachers' sentiment analysis after two years of "Programma il Futuro" project. *Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education*, 224-229.
- Csizmadia, A., Curzon, P., Dorling, M., Humphreys, S., Ng, T., Selby, C., & Woolard, J. (2015) *Computational thinking: A guide for teachers*. 12.03.2019 tarihinde <https://www.computingschool.org.uk/computationalthinking> adresinden erişilmiştir.
- Curzon, P. & McOwan P.W. (2008). Engaging with computer science through magic shows. *ACM SIGCSE Bulletin*, 40(3), 179-183.
- Curzon, P., McOwan, P. W., Plant, N., & Meagher, L. R. (2014). Introducing teachers to computational thinking using unplugged storytelling. In *Proceedings of the 9th workshop in primary and secondary computing education* (pp. 89-92). ACM.
- Czerkawski, B. C., & Lyman, E. W. (2015). Exploring issues about computational thinking in higher education. *TechTrends*, 59(2), 57-65.

- Çalışkan, D. (2019). *Cezeri'nin olağanüstü makineleri*. İstanbul: Babil Kitap.
- Çetin, İ., & Uçar, Z. T. (2017). Bilgi işlemsel düşünme tanımı ve kapsamı. Gülbahar, Y. (Ed), *Bilgi İşlemsel Düşünmeden Programlamaya*(s.41-78), Ankara: Pegem Akademi.
- Demir, Ö., & Seferoğlu, S. S. (2017). Yeni kavramlar, farklı kullanımlar: bilgi-işlemsel düşünmeyle ilgili bir değerlendirme. H. F. Odabaşı, B. Akkoyunlu ve A. İşman (Ed). *Eğitim Teknolojileri Okumaları 2017*, 41, 801–83. TOJET ve Sakarya Üniversitesi, Adapazarı.
- Denner, J., Werner, L., Campe, S., & Ortiz, E. (2014). Pair programming: Under what conditions is it advantageous for middle school students?. *Journal of Research on Technology in Education*, 46(3), 277-296.
- Denning, P. J. (2017) Computational Thinking in Science. *American Scientist*, 150, 13-17.
- Djambong, T., & Freiman, V. (2016). Task-Based assessment of students' computational thinking skills developed through visual programming or tangible coding environments. *13th International Conference on Cognition and Exploratory Learning in Digital Age (CELDA 2016)*, 41- 51.
- Doğan, D., Çınar, M., Bilgiç, H. G., & Tüzün, H. (2015) Sarmal eğitsel oyun tasarımı modeline göre dijital oyun geliştirme süreci: <e-adventure> örneği. *Uluslararası Oyun ve Oyuncaklar Kongresi* (ss. 442-452). Erzurum, Türkiye.
- Dorling, M., & White, D. (2015). Scratch: A way to logo and python. *In Proceedings of the 46th ACM Technical Symposium on Computer Science Education* (pp. 191–196). New York, USA.
- Durak, H. Y., Yilmaz, F. G. K., & Yilmaz, R. (2019). Computational thinking, programming self-efficacy, problem solving and experiences in the programming process conducted with robotic activities. *Contemporary Educational Technology*, 10(2), 173-197.

- Eguchi, A. (2014a). Educational robotics for promoting 21st century skills. *Journal of Automation Mobile Robotics and Intelligent Systems*, 8(1), 5–11.
- Eguchi, A. (2014b). Learning Experience Through RoboCupJunior: Promoting STEM Education and 21st Century Skills with Robotics Competition. In M. Searson & M. Ochoa (Eds.), *Proceedings of SITE 2014--Society for Information Technology & Teacher Education International Conference* (pp. 87-93). United States: AACE.
- Eguchi, A. (2014c). Why Robotics in Education? - Robotics as a Learning Tool for Educational Revolution. In M. Searson & M. Ochoa (Eds.), *Proceedings of SITE 2014--Society for Information Technology & Teacher Education International Conference* (pp. 94-95). United States: AACE.
- Eguchi, A. (2015). Educational Robotics as a Learning Tool for Promoting Rich Environments for Active Learning (REALs). In J. Keengwe (Ed.), *Handbook of Research on Educational Technology Integration and Active Learning* (pp. 19-47). Hershey, PA: IGI Global.
- Eguchi, A. (2016). RoboCupJunior for promoting STEM education, 21st century skills, and technological advancement through robotics competition. *Robotics & Autonomous Systems*, 75, 692–699.
- Faber, H. H., Wierdsma, M. D., Doornbos, R. P., van der Ven, J. S., & de Vette, K. (2017). Teaching computational thinking to primary school students via unplugged programming lessons. *Journal of the European Teacher Education Network*, 12, 13-24.
- Feldhausen, R., Weese, J. L. & Bean, N. H. (2018). Increasing Student Self-Efficacy in Computational Thinking via STEM Outreach Programs. *SIGCSE'18*, Baltimore, ACM.
- Gardeli, A., & Vosinakis, S. (2017). Creating the computer player: an engaging and collaborative approach to introduce computational thinking by combining

‘unplugged’ activities with visual programming. *Italian Journal of Educational Technology*, 25(2), 36-50.

González, F., López, C., & Castro, C. (2018). Development of Computational Thinking in High School Students: A Case Study in Chile. *37th International Conference of the Chilean Computer Science Society (SCCC)*, Santiago, Chile, 2018, pp. 1-8.

Grover, S., & Pea, R. (2013). Computational thinking in K–12: A review of the state of the field. *Educational Researcher*, 42(1), 38-43.

Guzdial, M. (2011). Re: A Definition of Computational Thinking from Jeannette Wing. 12.05.2019 tarihinde <http://computinged.wordpress.com/2011/03/22/a-definition-of-computational-thinking-from-jeanette-wing/> adresinden alınmıştır.

Gülbahar, Y., & Kalelioğlu F. (2014). The effects of teaching programming via scratch on problem solving skills: A discussion from learners' perspective. *Informatics in Education*, 13(1), 33-50.

Gülbahar, Y., Kert, S. B., & Kalelioğlu F. (2018). Ortaokul öğrencileri için bilgi işlemsel düşünme becerisine yönelik öz yeterlik algısı ölçeği: geçerlik ve güvenirlik çalışması. *Türk Bilgisayar ve Matematik Eğitimi Dergisi*. Advance online publication. doi:10.16949/turkbilmat.385097

Hambruch, S., Hoffmann, C., Korb, J. T., Haugan, M., & Hosking, A. L. (2009). A multidisciplinary approach towards computational thinking for science majors. *ACM SIGCSE Bulletin*, 41(1), 183-187.

Harel, I. (2016). *American schools are teaching our kids how to code all wrong*. 12.06.2019 tarihinde <https://qz.com/691614/american-schools-are-teaching-our-kids-how-to-code-all-wrong/> adresinden erişilmiştir.

- Hromkovič, J., Kohn, T., Komm, D., & Serafini, G. (2016). Examples of algorithmic thinking in programming education. *Olympiads in Informatics*, 10(1-2), 111-124.
- Hsu, H.J. (2014). Gender Differences in Scratch Game Design. *Proceedings of the 2014 International Conference on Information, Business and Education Technology (ICIBET)*.
- Hu, C. (2011). Computational Thinking: What It Might Mean and What We Might Do About It. *Proceedings of the 16th Annual Joint Conference on Innovation and Technology in Computer Science Education*, 223-227. New York: ACM. doi:10.1145/1999747.1999811.
- Huang, W.Y., & Wu, S.Y. (2012). A case study of collaboration with multirobots and its effect on children's interaction. *Interactive Learning Environments*, 22(4), 1-15.
- Hubwieser, P., Giannakos, M. N., Berges, M., Brinda, T., Diethelm, I., Magenheimer, J., ... Jasute, E. (2015). A global snapshot of computer science education in K-12 schools, *Proc. ITiCSE Working Group Reports*.
- IEEE (2019). *What is a robot?*. 06.05.2019 tarihinde <https://robots.ieee.org/learn/> adresinden erişilmiştir.
- Ioannou, A., & Makridou, E.(2018). Exploring the potentials of educational robotics in the development of computational thinking: A summary of current research and practical proposal for future work. *Education and Information Technologies*, 23(6), 2531-2544.
- ISTE (2016). *ISTE Standards for Students*. 06.05.2019 tarihinde <https://www.iste.org/standards/for-students> adresinden erişilmiştir.
- 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*, 26,175–192.

- Jun, W. (2018). A study on development of evaluation standards for unplugged activity. *International Conference on Information and Communication Technology Convergence (ICTC)*, Jeju, 2018, pp. 279-281.
- Kafai, Y.B. (2016). From computational thinking to computational participation in k-12 education. *Communications of the ACM*, 59(8), 26-27.
- Kalelioğlu, F. (2017). Bilgisayarsız bilgisayar bilimi (B3) öğretimi. Gülbahar, Y. (Ed). *Bilgi İşlemsel Düşünmeden Programlamaya*, (8.bölüm, ss.183-206). Ankara: Pegem Akademi.
- Kalelioglu, 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.
- Karaahmetoğlu, K., & Korkmaz, Ö. (2019). The effect of project-based arduino educational robot applications on students' computational thinking skills and their perception of Basic Stem skill levels. *Participatory Educational Research*, 6 (2), 1-14.
- Kelleher, C., & Pausch, R. (2005). Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers. *ACM Computing Surveys (CSUR)*, 37(2), 83-137.
- Kert. S. B. (2017). Bilgisayar Bilimi Eğitime Giriş, Gülbahar, Y. (Ed). *Bilgi İşlemsel Düşünmeden Programlamaya*, (1.bölüm, s.1-22). Ankara: Pegem Akademi.
- Keşfet Projesi (2019). *Eğitimciler İçin Bilgi İşlemsel Düşünme*. 16.01.2019 tarihinde <https://acikders.kesfetprojesi.org/#/unit/1/8> adresinden erişilmiştir.
- 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.

- Kukul, V. (2018). *Programlama öğretiminde farklı yapılandırılan süreçlerin öğrencilerin bilgi işlemsel düşünme becerilerine, özyeterliliklerine ve programlama başarılarına etkisi* (Doktora tezi). Gazi Üniversitesi, Ankara.
- Kwon, J. & Kim, J. (2018). A Study on the Design and Effect of Computational Thinking and Software Education. *KSII Transactions on Internet and Information Systems*, 12(8), 4057-4071.
- Lee, I., Martin, F., Denner, J., Coulter, B., Allan, W., Erickson, J., ... Werner, L. (2011). Computational thinking for youth in practice. *ACM Inroads*, 2(1), 32-37.
- Lee, J., Jung, Y., & Park, H. (2017). Gender Differences in Computational Thinking, Creativity, and Academic Interest on Elementary SW Education. *Korean Association of Information Education*, 21(4), 381–391.
- Leonardo Robot Society (2019). *The robots*. 07.02.2019 tarihinde <https://www.leonardorobotsociety.org/the-robots.html> adresinden alınmıştır.
- Lin, C. C., Chao, P. Y., Lin, E. T., & Tzeng, H. L. (2018). Exploring the role of visual programming activities in computational thinking. In *2018 1st International Cognitive Cities Conference (IC3)* (pp. 135-138). IEEE.
- Lockwood, J., & Mooney, A. (2017). A pilot study investigating the introduction of a computer-science course at second level focusing on computational thinking. *The Irish Journal of Education*, 42, 108-127.
- Looi, C. K., How, M. L., Longkai, W., Seow, P., & Liu, L. (2018). Analysis of linkages between an unplugged activity and the development of computational thinking. *Computer Science Education*, 28(3), 255-279.
- Lye, N. C., Wong, K. W., & Chiou, A. (2013). Framework for educational robotics: a multiphase approach to enhance user learning in a competitive arena. *Interactive Learning Environments*, 21(2), 142-155.

- 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, 51-61.
- Malan, D. J., & Leitner, H. H. (2007). Scratch for budding computer scientists. *In ACM Sigcse Bulletin*, 39(1), 223- 227.
- Milli Eğitim Bakanlığı [MEB] (2018a). Bilişim teknolojileri ve yazılım dersi öğretim programı (İlkokul 1,2,3 ve 4.sınıflar). Ankara.
- Milli Eğitim Bakanlığı [MEB] (2018b). Bilişim teknolojileri ve yazılım dersi öğretim programı (Ortaokul 5. ve 6.sınıflar). Ankara.
- Milli Eğitim Bakanlığı [MEB] (2018c). Bilişim teknolojileri ve yazılım dersi öğretim programı (Ortaokul ve İmam Hatip Ortaokulu 7. ve 8.sınıflar). Ankara.
- Milli Eğitim Bakanlığı [MEB] (2018d). Bilgisayar bilimi dersi öğretim programı (Kur 1- 2). Ankara.
- Mikropoulos, T. A., & Bellou, I. (2013). Educational robotics as mindtools. *Themes in Science and Technology Education*, 6(1), 5-14.
- Mladenović, M., Boljat, I., & Žanko, Ž. (2018). Comparing loops misconceptions in block-based and text-based programming languages at the K-12 level. *Education and Information Technologies*, 23(4), 1483-1500.
- Muñoz-Repiso, A. G. V., & Caballero-González, Y. A. (2019). Robotics to develop computational thinking in early childhood education. *Comunicar: Media Education Research Journal*, 27(59), 63-72.
- Oluk, A., & Korkmaz, Ö. (2016). Comparing students' scratch skills with their computational thinking skills in terms of different variables. *International Journal of Modern Education and Computer Science(IJMECS)*, 8(11), 1-7.
- Oxford Sözlük. (2019). *Oxford dictionaries*. 07.02.2019 tarihinde <https://languages.oup.com/> sayfasından erişilmiştir.

- Özden, M. Y. (2015). *Computational Thinking*. 29.12.2018 tarihinde <http://myozden.blogspot.com/2015/06/computational-thinking-bilgisayarca.html> sayfasından erişilmiştir.
- Papert, S. (1980). *Mindstorm: Children, computers, and powerful ideas*. New York: Basic Books.
- Papert, S., & Harel, I. E. (1991). *Constructionism*. New York : Ablex Publishing.
- Papert, S. (1996). An exploration in the space of mathematics educations. *International Journal of Computers for Mathematical Learning*, 1(1), 95-123.
- Park, D., & Yoo, I. (2018). Python Instructional Model using robot for elementary school students. *Journal of The Korean Association of Information Education*, 22(3), 357–366.
- Perkovic, L., Settle, A., Hwang, S., & Jones, J. (2010). A framework for computational thinking across the curriculum. *In Proceedings of the 2010 Conference on Innovation and Technology in Computer Science Education*, (pp. 123-127). Bilkent, Ankara.
- Petre, M., & Price, B. (2004). Using robotics to motivate ‘back door’ learning. *Education and Information Technologies*, 9 (2), 147-158.
- Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., ... Kafai, Y. (2009). Scratch: programming for all. *Communications of the ACM*, 52(11), 60-67.
- Resnick, M. (2013). *Learn to Code, Code to Learn*. 05.05.2019 tarihinde <https://www.edsurge.com/news/2013-05-08-learn-to-code-code-to-learn> adresinden erişilmiştir.
- Robotis (2019). *About robotis*. 07.02.2019 tarihinde http://en.robotis.com/model/page.php?co_id=introduce adresinden erişilmiştir.

- Robotis Dream (2019). *Robotis Dream II*. 07.02.2019 tarihinde <http://www.robotis.us/dream/> adresinden erişilmiştir.
- Ruesink, M. (2018). *8 surprising ways computer science benefits society*. 07.02.2019 tarihinde <https://www.rasmussen.edu/degrees/technology/blog/ways-computer-science-benefits-society/> adresinden alındı.
- Saad, A., Shuff, T., Loewen, G., & Burton, K. (2012). Supporting undergraduate computer science education using educational robots. *Proceedings of the 50th Annual Southeast Regional Conference*, 343-344.
- Sakamoto, K., Takano, K., Washizaki, H., & Fukazawa, Y. (2013). Learning system for computational thinking using appealing user interface with icon-based programming language on smartphones. In *Proceedings of the 21st International Conference on Computers in Education, ICCE*.
- Saritepeci, M., & Durak, H. Y. (2017). Analyzing the Effect of Block and Robotic Coding Activities on Computational Thinking in Programming Education. Koleva, I., & Duman, G. (Ed.). *Educational Research and Practice*, (49. Bölüm, ss. 464-473). Sofya: St. Kiment Ohridski University Press.
- Sayın, Z., ve Seferoğlu, S. S. (2016). Yeni bir 21. yüzyıl becerisi olarak kodlama eğitimi ve kodlamanın eğitim politikalarına etkisi. *Akademik Bilişim Konferansı*, Aydın.
- Schweikardt, E., & Gross, M. D. (2006). roBlocks: A robotic construction kit for mathematics and science education. In *Proceedings of the 8th international conference on Multimodal interfaces* (pp. 72–75). ACM.
- Selby, C., & Woollard, J. (2013) Computational Thinking: The developing definitions. In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education, SIGCSE 2014*. ACM.
- Sharaf, N., Ahmed, G., Adel, A., Abdennadher S. & Berkling, K. (2019). Koding4kinder: teaching computational thinking to pupils using a combination

- of programming and electronics platforms. *IEEE Global Engineering Education Conference (EDUCON)*, 634-638.
- Sklar, E., Eguchi, A., & Johnson, J. (2003). RoboCupJunior: Learning with educational robotics. *AI Magazine*, 24(2), 43-46.
- Small Basic. (2019). *Welcome to Small Basic!*. 16.03.2019 tarihinde <https://smallbasic-publicwebsite.azurewebsites.net/> adresinden erişilmiştir.
- 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.
- Şendurur, P. (2017). Bilişsel araçlar ve bilgi işlemsel düşünme. Gülbahar, Y. (Ed). *Bilgi İşlemsel Düşünmeden Programlamaya*, (4.bölüm, ss.79-99). Ankara: Pegem Akademi.
- Tabet, N., Gedawy, H., Alshikhabobakr, H., & Razak, S. (2016). From alice to python. Introducing text-based programming in middle schools, *Annual Conference on Innovation and Technology in Computer Science Education ITiCSE*, pp. 124-129.
- Thies, R., & Vahrenhold, J. (2013). On plugging “unplugged” into CS classes. *Proceeding of the 44th ACM Technical Symposium on Computer Science Education - SIGCSE '13* (pp. 365–370). New York: ACM Press.
- Türk Dil Kurumu [TDK]. (2019). Türk dil kurumu sözlükleri. 18.06.2019 tarihinde <http://sozluk.gov.tr/> adresinden erişilmiştir.
- Uslu, N. A., Mumcu, F., & Eğin, F. (2018). Görsel Programlama Etkinliklerinin Ortaokul Öğrencilerinin Bilgi-İşlemsel Düşünme Becerilerine Etkisi. *Ege Eğitim Teknolojileri Dergisi*, 2(1), 19- 31.

- Üçgöl, M. (2017). Eğitsel robotlar ve bilgi işlemsel düşünme. Gülbahar, Y. (Ed). *Bilgi İşlemsel Düşünmeden Programlamaya*, (12.bölüm, ss.295-317). Ankara: Pegem Akademi.
- Vieira, C., Penmetcha, M., Magana, A., & Matson, E. (2016). Computational Thinking as a Practice of Representation: A Proposed Learning and Assessment Framework. *Journal of Computational Science Education*, 7(1), 21-30.
- Weinberg, A. E. (2013). *Computational thinking: An investigation of the existing scholarship and research*. (Yayınlanmamış doktora tezi). Colorado State University, Colorado.
- Wing, J. M. (2006). Computational thinking. *Communicational of the ACM*, 49(3), 33-35.
- Wing, J. M. (2010). *Computational thinking: what and why? Center for computational thinking*. 18.12.2019 tarihinde <http://www.cs.cmu.edu/~CompThink/resources/TheLinkWing.pdf> adresinden erişilmiştir.
- Wing, J.M. (2017). Computational thinking's influence on research and education for all. *Italian Journal of Educational Technology*, 25(2), 7-14.
- Williams, D., Ma, Y., Prejean, L., Lai, G., & Ford, M. (2007). Acquisition of physics content knowledge and scientific inquiry skills in a robotics summer camp. *Journal of Research on Technology in Education*, 40 (2), 201-216.
- Wong, G. K. W., & Jiang, S. (2018). Computational thinking education for children: algorithmic thinking and debugging. *IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE)*, Wollongong, NSW, 2018, pp. 328-334.
- Yağcı, M. (2018). A study on computational thinking and high school students' computational thinking skill levels. *International Online Journal of Educational Sciences*, 10(2), 81-96.

- Yecan, E., Özçınar, H., & Tanyeri, T. (2017). Bilişim teknolojileri öğretmenlerinin görsel programlama öğretimi deneyimleri. *İlköğretim Online*, 16(1), 377-393.
- Yeni, S. (2018). Bilişim teknolojileri ve bilgi-işlemsel düşünme. Kert, S.B. (Ed). *Bilişim Teknolojileri*, (1.bölüm, s. 1-19). Ankara: Nobel Yayınları.
- Yolcu, V. (2018). *Programlama eğitiminde robotik kullanımının akademik başarı, bilgi-işlemsel düşünme becerisi ve öğrenme transferine etkisi* (Yayınlanmamış yüksek lisans tezi). Süleyman Demirel Üniversitesi, Isparta.
- Yükseltürk, E. & Altıok, S. (2017). Blok tabanlı programlama. Gülbahar, Y. (Ed). *Bilgi İşlemsel Düşünmeden Programlamaya*, (10.bölüm, ss.241-266). Ankara: Pegem Akademi.

BÖLÜM VII: EKLER

Ek 1: Araştırma İzni

Re: "Ortaokul Öğrencileri İçin Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlilik Algısı Ölçeği" İzin Talebi

20.12.2018 Per 12:44 tarihinde yanıtladınız

SB serhat bahad <sbkert@gmail.com>
20.12.2018 Per 11:33
Siz

Oğuzhan Merhabalar,

Ölçeği kullanmanız uygundur, iyi çalışmalar dilerim.

Şekil 3 : Araştırmada Kullanılan Ortaokul Öğrencileri için Bilgi İşlemsel Düşünme Becerilerine Yönelik Öz Yeterlilik Algısı Ölçeği'nin Kullanımı için Ölçeği Hazırlayan Araştırmacılar Gelen İzin E-postası

T.C.
İSTANBUL VALİLİĞİ
İl Millî Eğitim Müdürlüğü

Sayı : 59090411-20-E.10522077
Konu : Anket ve Araştırma İzin Talebi

28/05/2019

VALİLİK MAKAMINA

İlgi: a) 17.05.2019 tarihli ve 9780928 Gelen Evrak No'lu dilekçe.
b) MEB. Yen. ve Eğ. Tk. Gn. Md. 22.08.2017 tarih ve 12607291/ 2017/25 No'lu Gen.
c) Millî Eğitim Müdürlüğü Araştırma ve Anket Komisyonunun 28.05.2019 tarihli tutanağı.

Marmara Üniversitesi Eğitim Bilimleri Enstitüsü yüksek lisans öğrencisi Oğuzhan ÖZEL'in "Robotik Eğitiminin Ortaokul Öğrencilerinin Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlilik Algısına Etkisi" konulu tezi kapsamında, ilimiz Fatih ilçesinde bulunan ortaokullarda, bilgi işlemsel düşünme becerisine yönelik öz yeterlilik algısı ölçeğini uygulama ismi hakkındaki ilgi (a) dilekçe ve ekleri Müdürlüğümüzce incelenmiştir.

Araştırmacının söz konusu talebi; bilimsel amaç dışında kullanılmaması, uygulama sırasında bir örneği müdürlüğümüzde muhafaza edilen mühürlü ve imzalı veri toplama araçlarının kurumlarımıza araştırmacı tarafından ulaştırılarak uygulanması, katılımcıların gönüllülük esasına göre seçilmesi, araştırma sonuç raporunun müdürlüğümüzden izin alınmadan kamuoyuyla paylaşılmaması koşuluyla, okul idarelerinin denetim, gözetim ve sorumluluğunda, eğitim-öğretimi aksatmayacak şekilde ilgi (b) Bakanlık emri esasları dâhilinde uygulanması, sonuçtan Müdürlüğümüze rapor halinde (CD formatında) bilgi verilmesi kaydıyla Müdürlüğümüzce uygun görülmektedir.

Makamlarınızca da uygun görülmesi halinde olurlarınıza arz ederim.

Levent YAZICI
İl Millî Eğitim Müdürü

Ek:
1- Genelge.
2- Komisyon Tutanağı.

OLUR
28/05/2019

Ahmet Hamdi USTA
Vali a.
Vali Yardımcısı

İl Millî Eğitim Müdürlüğü Binbirdirek M. İsmail Ökten Cad.
No:1 Faki Adıyaman Bulvarı Sultanhani Fatih/İstanbul
E-Posta: ilym@meb.gov.tr

A. BALTA YHK
Tel: (0 212) 455 04 00-239

Her türlü gizlilik ihtiyacı için de kullanılabilir: <https://evrak.meb.gov.tr/adresimiz> D030-4058-30e0-9584-886f /sade ile ilgili olabilir.

Şekil 4 : Ortaokul Öğrencileri için Bilgi İşlemsel Düşünme Becerilerine Yönelik Öz Yeterlilik Algısı Ölçeği'nin Kullanımı Hakkında İstanbul İl Millî Eğitim Müdürlüğünden Alınan İzin Yazısı

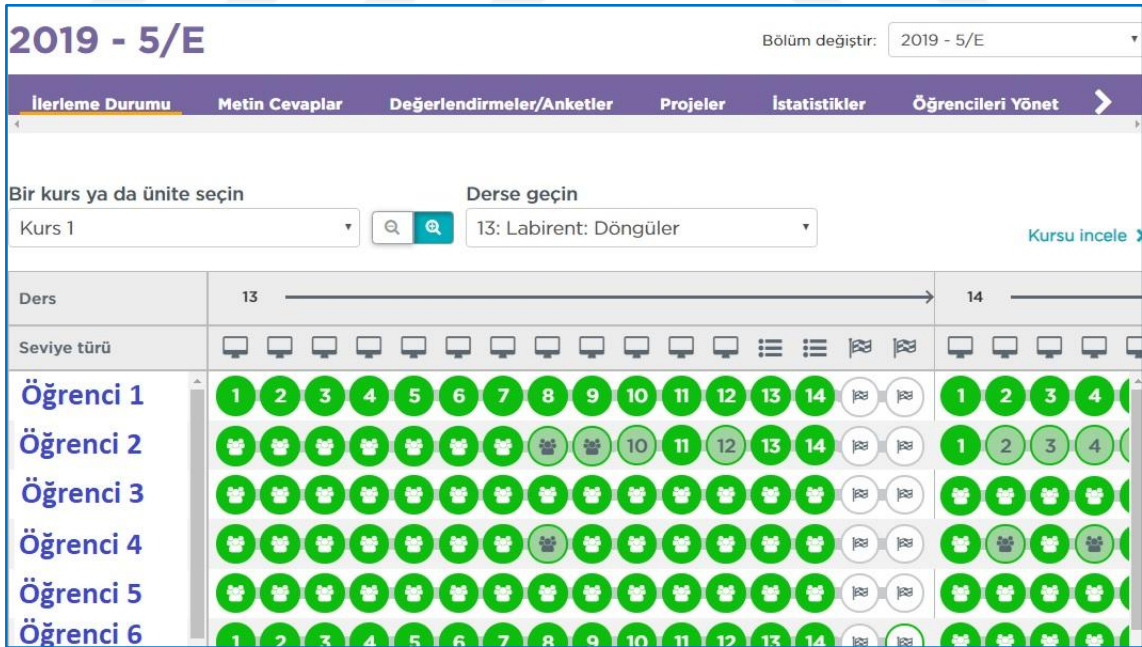
Ek 2: Blok Tabanlı Programlama Grubu Öğretim Aracı ve Eğitim Süreci

Araştırmada blok tabanlı programlama eğitimi verilen gruptaki öğrencilere öğretim aracı olarak Code.org internet sitesi seçilmiştir.

Code.org nedir?

Code.org internet sitesi 2013 yılından beri her yaştan öğrenciye bilgisayar bilimlerini ve programlama mantığını öğretmek için oluşturmuş bir platformdur. Blok tabanlı programlama mantığı ile çalışmaktadır. Kâr amacı bulunmayan platform vizyonlarını şöyle ifade etmektedir: “Her okuldaki her öğrencinin tıpkı biyoloji, kimya veya cebir gibi bilgisayar bilimlerini öğrenme fırsatını sağlamak”. Code.org, Amazon, Facebook, Google, Infosys Vakfı, Microsoft vb. firmalar tarafından desteklenmektedir (Code.org, 2019).

Eğitim süreci öncesinde öğrencilerin takibinin daha iyi yapılabilmesi adına tüm öğrencilere üyelik oluşturulmuştur. Tüm eğitim sürecinde öğretmen kontrol sayfasından öğrencilerin hangi aşamada olduğu takip edilmiştir.



Şekil 5 : Code.org Öğrenci Takip Ekranı

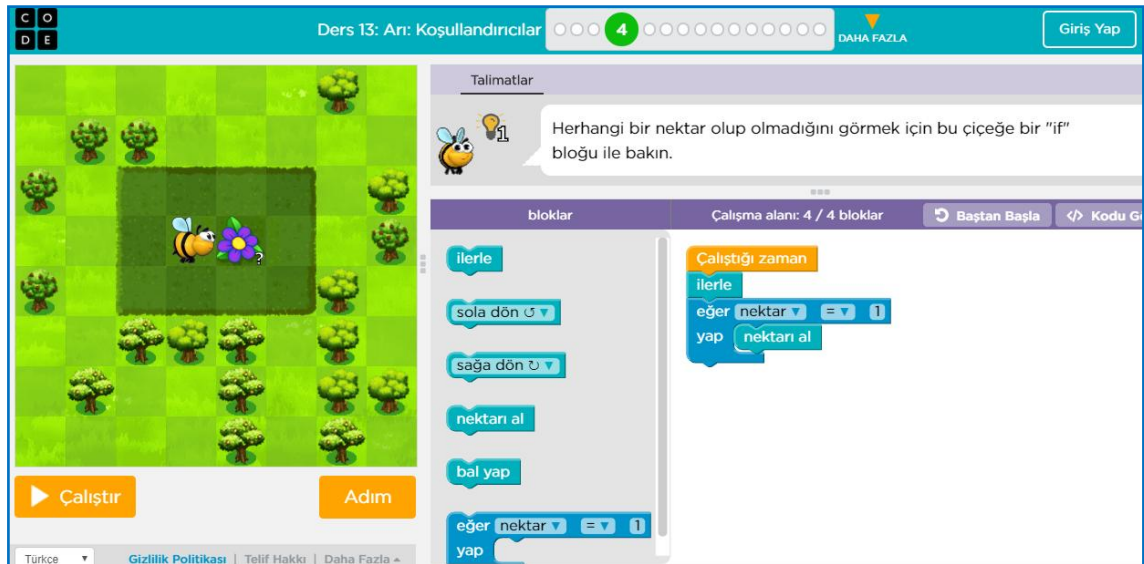
Bu takipler sonunda eğitime katılan tüm öğrencilerin hedeflenen tüm örneklerdeki programlama görevlerini yerine getirdiği görülmüştür.

Blok tabanlı programlama eğitimine ait örnek günlük ders planı tablo 49’da verilmiştir.

Tablo 49 : Blok Tabanlı Programlama Eğitimine Ait Günlük Ders Planı Örneği (5.hafta)

Kazanımlar	Konular	Açıklama	Yöntem - Teknik
- Karar yapısını ve işlevlerini açıklar. - Karar yapısını içeren programlar oluşturur.	Karar Yapısı – Eğer / Değilse (if-else)	Karar yapısı ve işlevleri açıklanır. Karar yapısına ait günlük hayattan örnekler verilir. Öğrencilerin bu örnekleri çoğaltmaları istenir. Ardından code.org sitesindeki karar yapısıyla alakalı olan bölümlerden örnek birkaç programlama görevinin çözümü gösterilir. Sonraki aşamada tüm öğrencilerin bilgisayarlarında code.org sitesindeki hesaplarına girmeleri istenir. Code.org sitesi içeriğindeki “Kurs 2 Ders 13” ve “Kurs 3 Ders 7-8” bölümlerindeki örneklerin öğrenciler tarafından programlanması istenir. Öğrenci takip ekranından bulundukları aşamalar takip edilir.	Gösterip yaptırma, soru cevap, Tartışma

Tablo 49’da günlük ders planı verilen karar yapılarının anlatıldığı haftadan örnek programlama görevi ve çözümü şekil 5’de verilmiştir.



Şekil 6 : Code.org Kurs 2 Ders 13 / 4. Programlama Görevi ve Çözümü

Ek 3: Metin Tabanlı Programlama Grubu Öğretim Aracı ve Eğitim Süreci

Araştırmada metin tabanlı programlama eğitimi verilen gruptaki öğrencilere öğretim aracı olarak Small Basic programı seçilmiştir.

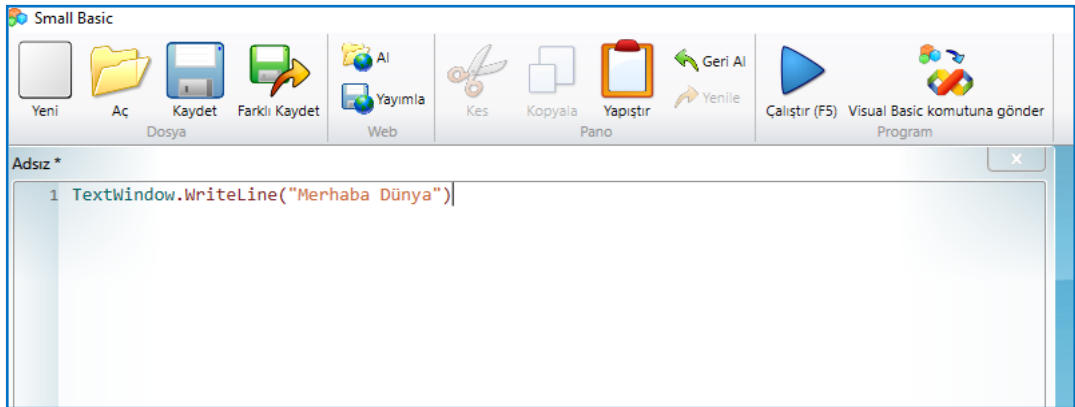
Bu araştırmada metin tabanlı programlama eğitimi alan gruptaki öğrencilerin ilk programlama deneyimi olması sebebiyle başlangıç seviyesine uygun bir dil seçilmeye çalışılmıştır. Small Basic'in arayüzü, özellikleri ve hitap ettiği yaş seviyesi göz önüne alındığında araştırma amacına en uygun metin tabanlı programlama ortamı sağlayacağına karar verilmiştir.

Small Basic Nedir?

Small Basic, programlamayı yeni başlayanlar için erişilebilir ve kolay hale getirmeye odaklanan bir projedir. Üç ayrı parçadan oluşur:

- Dil
- Programlama Ortamı
- Kütüphaneler (Small Basic, 2019)

Small Basic, BASIC isimli gelişmiş programlama dilinin temel özelliklerini içeren bir versiyonudur. Blok tabanlı kodlamadan metin tabanlı kodlamaya geçişi sağlamak için öğrencilere özel olarak oluşturulan tek programlama dilidir (Small Basic, 2019).



Şekil 7 : Small Basic Programı Arayüzü ve Örnek Kod

Small Basic, metin tabanlı programlama dillerinin temel öğelerini öğreterek, öğrencilere Java ve C # gibi daha karmaşık programlama diller ile başa çıkma becerileri ve güveni sağlar. Geliştirme sürecindeki denemelerimizde, 10 ila 16 yaş arasındaki çocuklarla

başarı elde ettik. 7 ila 107 yaşları arasında, Small Basic kodlamayı öğrenmenin en kolay yollarından biridir (Small Basic, 2019). Kandemir (2017) çalışmasında erken yaşlarda programlama becerisi ve BİD becerisi kazandırmak amacıyla Small Basic programlama dilinin kullanılabileceğini belirtmiştir. Metin tabanlı programlama eğitimine ait örnek günlük ders planı tablo 50’de verilmiştir.

Tablo 50 : *Metin Tabanlı Programlama Eğitimine Ait Günlük Ders Planı Örneği (5.hafta)*

Kazanımlar	Konular	Açıklama	Yöntem - Teknik
- Koşul yapısını ve işlevlerini açıklar. - Koşul yapısını içeren programlar oluşturur.	Koşul Yapısı – Eğer / Değilse (if-else)	Koşul yapısı ve işlevleri açıklanır. Koşul yapısına ait günlük hayattan örnekler verilir. Öğrencilerin bu örnekleri çoğaltmaları istenir. Ardından Small Basic programında karar yapısıyla ilgili birkaç tane örnek program yazılır ve öğrencilere gösterilir. Sonraki aşamada tüm öğrencilerin bilgisayarlarında Small Basic programını açmaları istenir. Öğrencilerin Small Basic ile programlama yaparak önceden hazırlanmış koşul yapısını içeren problemleri sözdizimine dökerek çözmeleri istenir. Öğrencilerin programlama süreçleri takip edilir.	Gösterip yaptırma, soru cevap, Tartışma

Tablo 50’de günlük ders planı verilen karar yapılarının anlatıldığı haftadan örnek programlama görevi şöyledir: “Belediyelerde ulaşım araçlarında genellikle 65 yaş üzeri kişilerin ücretsiz ulaşım hakkı bulunmaktadır. Buradan yola çıkarak kullanıcıdan yaşını girmesini isteyen ardından doğru koşul yapısı oluşturularak kişinin ücretli veya ücretsiz ulaşımdan faydalanacağını ekrana yazdıran programı yazınız.”

```

TextWindow.WriteLine("Lütfen yaşınızı giriniz")
yas = TextWindow.Read()
If yas > 65 Then
    TextWindow.WriteLine("Ücretsiz Geçiş")
EndIf
If yas <= 65 Then
    TextWindow.WriteLine("Ücretli Geçiş")
EndIf

```

Şekil 8 : *Metin Tabanlı Programlama Eğitimi Örnek Programlama Görevi Çözümü*

Ek 4: Robotik Programlama Grubu Öğretim Aracı ve Eğitim Süreci

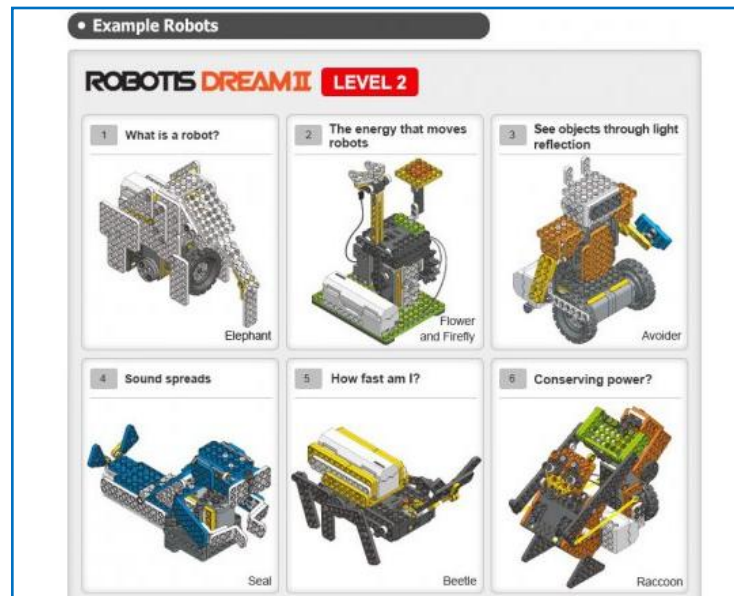
Araştırmadaki robotik programlama eğitimi gruptaki öğrencilere öğretim aracı olarak Robotis Dream eğitsel robotik seti ve Roboplus Task programı seçilmiştir.

Robotis Nedir?

Robotis 1999 yılında Güney Kore’de kurulmuştur. Kendi internet sitelerinde bulunan açıklamalarında isimlerinin “Robot nedir?” sorusuna basit bir cevaptan türetildiğini ve o zamandan beri birçok alan için robotlar ürettiklerini belirtmektedirler. Çocukların hayallerini gerçekleştirmeleri için de eğitsel robot setleri ürettiklerini eklemişlerdir (Robotis, 2019).

Robotis Dream Nedir?

Robotis Dream, çocukları ile defa robotik ile tanıştırmak için hazırlanan ve çeşitli seviyeleri olan bir eğitsel robot setidir (Robotis Dream, 2019). Robotis Dream Seviye 1, elektriksel güç, yerçekimi, kuvvet, 2 ayaklı ve 4 ayaklı robotlarda hareket gibi temel fizik, elektronik ve robotik prensiplerini giriş seviyesinde veren bir robot kitidir. Robotis Dream Seviye 2, bir robotun nasıl hareket ettiğinin, sensörlerin kullanımı, hız ve kuvvet, yürüyüş ve sürüş gibi robotiğin temellerini öğretir (Robotis Dream, 2019).

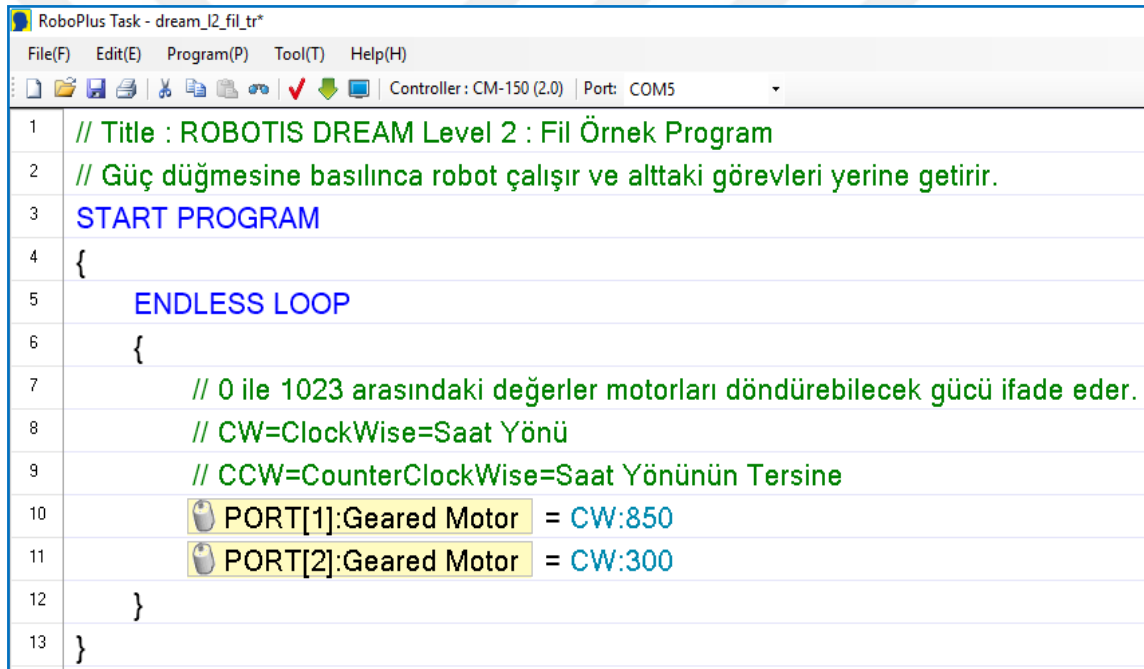


Şekil 9 : Robotis Dream Seviye 2’den Örnek Robotlar

Araştırma boyunca robotik programlama eğitimi alan gruba Robotis Dream Seviye 2'den altı ayrı robot mekanik olarak yaptırılmıştır. Mekanik olan tamamlanan robotlara ise C++ tabanlı programlama imkanı sunan RoboPlus Task programı ile öğrenciler tarafından kod yazılmıştır.

Robotis RoboPlus Task Programı Nedir?

RoboPlus Task, robotlara C++ programlama diline göre kod yazıp, bu kodları robotlara aktarmaya yarayan bir yazılımdır. Yazılımın bazı bölümlerinde metin tabanlı programlama ile ilerlerken bazı bölümlerinde hazırlanmış kod bloklarından faydalanılmaktadır.



```
1 // Title : ROBOTIS DREAM Level 2 : Fil Örnek Program
2 // Güç düğmesine basılınca robot çalışır ve alttaki görevleri yerine getirir.
3 START PROGRAM
4 {
5     ENDLESS LOOP
6     {
7         // 0 ile 1023 arasındaki değerler motorları döndürebilecek gücü ifade eder.
8         // CW=ClockWise=Saat Yönü
9         // CCW=CounterClockWise=Saat Yönünün Tersine
10        PORT[1]:Geared Motor = CW:850
11        PORT[2]:Geared Motor = CW:300
12    }
13 }
```

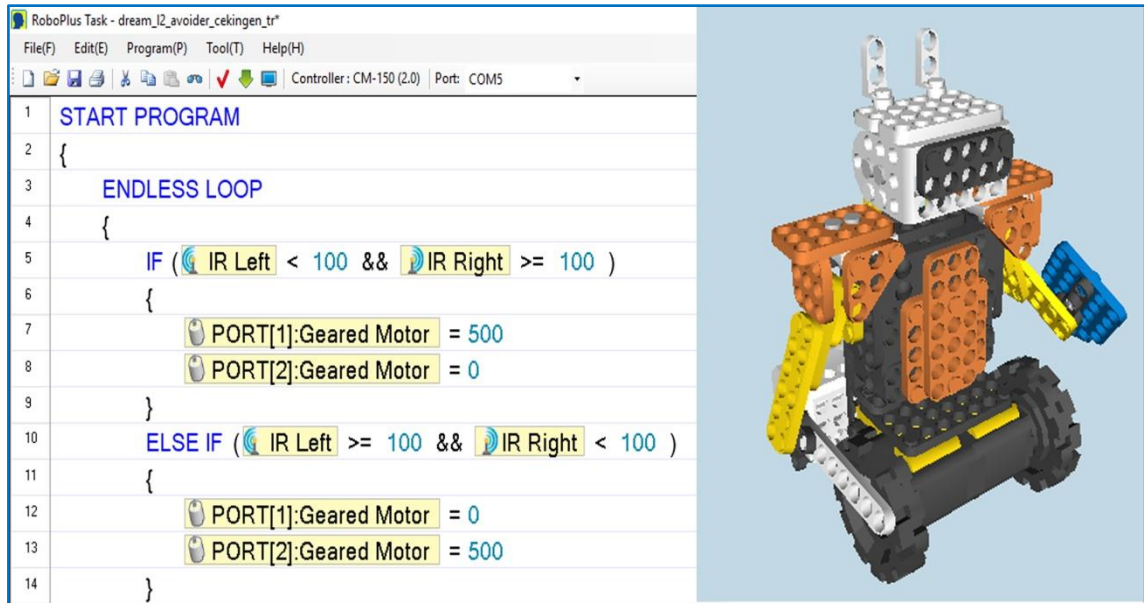
Şekil 10 : RoboPlus Task Programı Arayüzü ve Örnek Kod

Detaylı bilgileri verilen eğitsel robotik seti ve programlama aracı ile verilen robotik programlama eğitimine ait örnek günlük ders planı tablo 51'de verilmiştir.

Tablo 51 : Robotik Programlama Eğitimine Ait Günlük Ders Planı Örneği (5.hafta)

Kazanımlar	Konular	Açıklama	Yöntem - Teknik
- Koşul yapısını ve işlevlerini açıklar. - Koşul yapısını içeren programlar oluşturur.	Koşul Yapısı – Eğer / Değilse (if-else)	Koşul yapısı ve işlevleri açıklanır. Koşul yapısına ait günlük hayattan örnekler verilir. Öğrencilerin bu örnekleri çoğaltmaları istenir. Ardından koşul yapısıyla çalışabilecek robotlar ile ilgili fikir alışverişinde bulunulur. Sonraki aşamada tüm öğrencilerin eğitsel robotik setleriyle Seviye 2'ye ait robotlardan “Çekingen Robot'u” mekanik olarak hazırlamaları istenir. Mekanik hazırlık tamamlandıktan sonra öğrencilerden çekingen robota RoboPlus Task yazılımı ile programlama yaparak önünde herhangi bir nesne gördüğü zaman ondan kaçmasını sağlayacak hale getirmeleri istenir. Öğrencilerin programlama süreçleri takip edilir.	Gösterip yaptırma, soru cevap, Tartışma

Tablo 51’de günlük planı verilen koşul yapısı konusuna ait robot ve program örneği Şekil 11’de verilmiştir.



Şekil 11 : Robotik Programlama Grubu - Örnek Robot ve Koşul Yapısı Örneği

Ek 5: Programlama Bilgisi Sınav Soru Örnekleri* – Cevap Anahtarı – Belirtke Tablosu

Tablo 52 : Programlama Bilgisi Sınavı Soru Örnekleri - Cevapları ve Kazanımları																																
Soru	Cevap	İlgili Kazanım ve Değerlendirme Rubriği																														
1. “Belirli bir problemi çözmek ve belli bir sonuca ulaşmak için çizilen yola denir.” Yukarıda tanımlı verilen bilgisayar, matematik bilimlerinde sıkça kullanılan ve kodlamanın temelini oluşturan kavram hangisidir? a) Algoritma b) Binary (ikili sayı sistemi) c) Veri d) İşletim	“Belirli bir problemi çözmek ve belli bir sonuca ulaşmak için çizilen yola ALGORİTMA denir.”	Kazanım: Algoritma kavramını açıklar. Ölçme ve Değerlendirme: Soruya verilen doğru cevap +4 puan olarak hesaplanmıştır.																														
2. Aşağıdaki tabloda bulunan bilgilerden yola çıkarak veri tipi alanındaki boşlukları doldurunuz. <table border="1"> <thead> <tr> <th>VERİ</th><th>DEĞER</th><th>VERİ TİPİ</th></tr> </thead> <tbody> <tr> <td>İsim</td><td>Ahmet</td><td></td></tr> <tr> <td>Soyisminin ilk harfi</td><td>T</td><td></td></tr> <tr> <td>Sınıfı</td><td>5</td><td></td></tr> <tr> <td>Bilişim Dersini sever mi? (Evet / Hayır)</td><td>Evet</td><td></td></tr> </tbody> </table>	VERİ	DEĞER	VERİ TİPİ	İsim	Ahmet		Soyisminin ilk harfi	T		Sınıfı	5		Bilişim Dersini sever mi? (Evet / Hayır)	Evet		<table border="1"> <thead> <tr> <th>VERİ</th><th>DEĞER</th><th>VERİ TİPİ</th></tr> </thead> <tbody> <tr> <td>İsim</td><td>Ahmet</td><td>Karakter Dizisi Veri Tipi</td></tr> <tr> <td>Soyisminin ilk harfi</td><td>T</td><td>Karakter Veri Tipi</td></tr> <tr> <td>Sınıfı</td><td>5</td><td>Sayısal Veri Tipi</td></tr> <tr> <td>Bilişim Dersini sever mi? (Evet / Hayır)</td><td>Evet</td><td>Mantıksal Veri Tipi</td></tr> </tbody> </table>	VERİ	DEĞER	VERİ TİPİ	İsim	Ahmet	Karakter Dizisi Veri Tipi	Soyisminin ilk harfi	T	Karakter Veri Tipi	Sınıfı	5	Sayısal Veri Tipi	Bilişim Dersini sever mi? (Evet / Hayır)	Evet	Mantıksal Veri Tipi	Kazanım: Verileri toplayarak türlerine göre sınıflandırır. Ölçme ve Değerlendirme: Doğru bilinen her veri tipi +2 puandır. Sorunun tamamı 8 puan üzerinden değerlendirilmeye alınmıştır.
VERİ	DEĞER	VERİ TİPİ																														
İsim	Ahmet																															
Soyisminin ilk harfi	T																															
Sınıfı	5																															
Bilişim Dersini sever mi? (Evet / Hayır)	Evet																															
VERİ	DEĞER	VERİ TİPİ																														
İsim	Ahmet	Karakter Dizisi Veri Tipi																														
Soyisminin ilk harfi	T	Karakter Veri Tipi																														
Sınıfı	5	Sayısal Veri Tipi																														
Bilişim Dersini sever mi? (Evet / Hayır)	Evet	Mantıksal Veri Tipi																														
3. Aşağıdaki akış diyagramındaki soru işaretleri yerine ne yazılmalıdır ki havanın sıcaklığına göre kıyafet seçimi yapan algoritma hatasız çalışsın.		Kazanım: Bir algoritma için akış şeması çizer. Ölçme ve Değerlendirme: Soruda verilen boşluklara yazılan her doğru cevap +2 puandır. Tüm soru 6 puan üzerinden değerlendirilmiştir. (Farklı derece miktarı ve soru ile oluşturulan koşul yapıları da akış diyagramının hatasız olmasını sağlıyorsa, cevap doğru kabul edilmiştir.)																														

*Telif hakları sebebiyle sınav sorularının tamamı yayınlanmamıştır.