

BAŞKENT ÜNİVERSİTESİ
EĞİTİM BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ EĞİTİMİ
ANABİLİM DALI
BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ EĞİTİMİ TEZLİ
YÜKSEK LİSANS PROGRAMI

FİZİKSEL PROGRAMLAMA ARACI DESTEKLİ PROGRAMLAMA
ÖĞRETİMİNİN ORTAOKUL ÖĞRENCİLERİNİN BİLGİ İŞLEMSEL
DÜŞÜNME BECERİLERİNE VE ÖZYETERLİK ALGILARINA
ETKİSİ

HAZIRLAYAN
EZGİ ARZU YURDAKÖK

YÜKSEK LİSANS TEZİ

ANKARA – 2022

BAŞKENT ÜNİVERSİTESİ
EĞİTİM BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ EĞİTİMİ
ANABİLİM DALI
BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ EĞİTİMİ TEZLİ
YÜKSEK LİSANS PROGRAMI

FİZİKSEL PROGRAMLAMA ARACI DESTEKLİ PROGRAMLAMA
ÖĞRETİMİNİN ORTAOKUL ÖĞRENCİLERİNİN BİLGİ İŞLEMSEL
DÜŞÜNME BECERİLERİNE VE ÖZYETERLİK ALGILARINA
ETKİSİ

HAZIRLAYAN
EZGİ ARZU YURDAKÖK

YÜKSEK LİSANS TEZİ

TEZ DANIŞMANI
Doç. Dr. Filiz KALELİOĞLU

ANKARA – 2022

BAŞKENT ÜNİVERSİTESİ
EĞİTİM BİLİMLERİ ENSTİTÜSÜ

Bilgisayar ve Öğretim Teknolojileri Eğitimi Anabilim Dalı Bilgisayar ve Öğretim Teknolojileri Eğitimi Tezli Yüksek Lisans çerçevesinde Ezgi Arzu YURDAKÖK tarafından hazırlanan bu çalışma, aşağıdaki jüri tarafından Yüksek Lisans Tezi olarak kabul edilmiştir.

Tez Savunma Tarihi: 10 / 08 / 2022

Tez Adı: Fiziksel Programlama Aracı Destekli Programlama Öğretiminin Ortaokul Öğrencilerinin Bilgi İşlemsel Düşünme Becerilerine ve Özyeterlik Algılarına Etkisi

Tez Jüri Üyeleri (Unvanı, Adı- Soyadı, Kurumu)

İmza

Doç. Dr. Filiz KALELİOĞLU/ Başkent Üniversitesi

.....

Doç. Dr. H. Tuğba ÖZTÜRK/ Ankara Üniversitesi

.....

Dr. Öğr. Üyesi Halil ERSOY/ Başkent Üniversitesi

.....

ONAY

Yukarıdaki imzaların, adı geçen öğretim üyelerine ait olduğunu onaylıyorum.

Prof. Dr. Servet ÖZDEMİR

Eğitim Bilimleri Enstitüsü

Müdürü

Tarih: ... / ... /

BAŞKENT ÜNİVERSİTESİ
EĞİTİM BİLİMLERİ ENSTİTÜSÜ
YÜKSEK LİSANS TEZİ ORJİNALLİK RAPORU

Tarih: 31 /08 /2022

Öğrencinin Adı, Soyadı: Ezgi Arzu YURDAKÖK

Öğrencinin Numarası: 21820118

Anabilim Dalı: Bilgisayar ve Öğretim Teknolojileri Eğitimi Anabilim Dalı

Programı: Bilgisayar ve Öğretim Teknolojileri Eğitimi Tezli Yüksek Lisans Programı

Danışmanın Unvanı/Adı, Soyadı: Doç. Dr. Filiz Kalelioğlu

Tez Başlığı: Fiziksel Programlama Aracı Destekli Programlama Öğretiminin Ortaokul Öğrencilerinin Bilgi İşlemsel Düşünme Becerilerine ve Özyeterlik Algılarına Etkisi

Yukarıda başlığı belirtilen Yüksek Lisan tez çalışmamın; Giriş, Ana Bölümler ve Sonuç Bölümünden oluşan, toplam 84 sayfalık kısmına ilişkin, 31/ 08 / 2022 tarihinde tez danışmanım tarafından turnitin adlı intihal tespit programından aşağıda belirtilen filtrelemeler uygulanarak alınmış olan orijinallik raporuna göre, tezimin benzerlik oranı %14'dür. Uygulanan filtrelemeler:

1. Kaynakça hariç
2. Alıntılar hariç
3. Beş (5) kelimeden daha az örtüşme içeren metin kısımları hariç

“Başkent Üniversitesi Enstitüleri Tez Çalışması Orijinallik Raporu Alınması ve Kullanılması Usul ve Esaslarını” inceledim ve bu uygulama esaslarında belirtilen azami benzerlik oranlarına tez çalışmamın herhangi bir intihal içermediğini; aksinin tespit edileceği muhtemel durumda doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi ve yukarıda vermiş olduğum bilgilerin doğru olduğunu beyan ederim.

Öğrenci İmzası:.....

ONAY

Tarih: 31 / 08 / 2022

Doç. Dr. Filiz KALELIOĞLU

TEŞEKKÜR

Bilgisayar ve Öğretim Teknolojileri Öğretmenliğinden mezun olduğum değerli okuluma yüksek lisans yapmak amacıyla tekrar başlamakta mutluyum. Yüksek lisansa başlamamda destek olan ve her zaman yeni öğrenme yolları açarak mezun olduğum dönemden bu yana kendimi gerçekleştirmemi sağlayan, alanımıza katkıları yadsınamaz çok değerli hocalarım Prof. Dr. Yasemin Gülbahar ve Doç. Dr. Filiz Kalelioğlu'na her zaman destek oldukları ve inandıkları için teşekkür ederim. Sadece tezimde değil, lisans yıllarımdan beri beni destekleyen öğrenme yolculuğumda bana eşlik eden, her zaman yeni kapılar aralayarak kendimi geliştirmemi sağlayan; tez aşamasında bilgisini, tecrübesini ve değerli zamanını benimle paylaşan, her konuda benden desteğini esirgemeyen değerli tez danışmanım Sayın Doç. Dr. Filiz Kalelioğlu'na sonsuz teşekkürlerimi borç bilirim. Öğretmenlik hayatıma başladığımdan beri “İyi ki öğretmenim!” demekten hep gurur duydum. Öğretme yolculuğumda bana her yıl başka anılar katan, onlarla öğrenip geliştiğim ve tezime konu olan araştırma sürecimde katılım gösteren öğrencilerime teşekkür ederim.

Bu çalışmayı yapabilmemde micro:bit programlama araçlarını bana temin ettikleri için Micro:bit Educational Foundation teşekkür ederim. Ayrıca, tezimi geliştirmem de yapıcı öneri ve eleştirileriyle katkı sunan değerli hocalarım Doç. Dr. H. Tuğba Öztürk ve Dr. Öğr. Üyesi Halil Ersoy'a teşekkürlerimi sunarım.

Bu günlere gelebilmemde desteğini hiç esirgemeyen “Bilgi güçtür!” diyerek beni okumaya ve çalışmaya teşvik eden, hayatın tüm zorluklarına karşı bana güç veren çok değerli canım annem Sabiha Güneş'e, benim ilk öğretmenim, hep sen yaparsın diyen, desteği ve varlığı ile hayatımda çok önemli yeri olan sevgili canım babam Mustafa Güneş'e, hayatımın her anında yanımda olan ve bana destek olan canım kardeşim Tuğçe Güneş'e çok teşekkür ederim, iyi ki varsınız ve iyi ki benim ailemsiniz. Son sözü varlıklarıyla hayatıma anlam katan değerli canım eşim ve oğluma ayırmak istiyorum. Bu süreçte onlarla geçireceğim vakitten alarak onları ihmal etmenin vicdan azabına karşın bana destek oldular ve anlayışla beni beklediler. Sevgili eşim Volkan Yurdakök tüm sabrın ve desteğin için çok ama çok teşekkür ederim. Hayat kaynağım canım oğlum Ata Kemal Yurdakök, her şey seninle güzel meleşim. Siz olmasanız bir yerde pes edebilirdim ama varlığınız bana hep güç verdi. İyi ki benimlesiniz.

Ezgi Arzu YURDAKÖK- Ankara, 2022

ÖZET

Ezgi Arzu YURDAKÖK

**Fiziksel Programlama Aracı Destekli
Programlama Öğretiminin Ortaokul Öğrencilerinin
Bilgi İşlemsel Düşünme Becerilerine ve Özyeterlik Algılarına Etkisi**

**Başkent Üniversitesi
Eğitim Bilimleri Enstitüsü
Bilgisayar ve Öğretim Teknolojileri Eğitimi Anabilim Dalı
Tezli Yüksek Lisans Programı**

Ağustos 2022

Bu çalışmanın amacı, fiziksel programlama aracı destekli metin tabanlı programlama öğretiminin ortaokul öğrencilerinin bilgi işlemsel düşünme becerilerine ve özyeterlik algılarına etkisini incelemektir. Karma araştırma yöntemi türlerinden sıralı açıklayıcı desende yürütülen çalışmaya, Ankara ilinde özel bir ortaokulda okuyan 85 ortaokul öğrencisi katılmıştır. Çalışma kapsamında 6 hafta süresince fiziksel programlama aracı olan micro:bit destekli Python programlama dili öğretimi yapılmıştır. Veri toplama aracı olarak Bilgi İşlemsel Düşünme Becerilerine Yönelik Öz Yeterlik Algısı Ölçeği (BİDBÖA), 2016-2020 tarihleri arasından seçilmiş Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme Etkinliği Soruları, programlama aracı, kavram ve süreçlerine yönelik kazanım testi ile yarı yapılandırılmış görüşme soruları kullanılmıştır. Ön test ve son testten elde edilen bulgulara göre, öğrencilerin bilgi işlemsel düşünme becerilerine yönelik öz yeterlik algılarında ve bilgi işlemsel düşünme becerilerinde anlamlı ve pozitif yönde bir farklılık bulunmuştur. Yapılan öğretim sonucunda öğrencilerin programlama kavram ve süreçlerine yönelik kazanımlara ulaştığı ve dersteki performanslarının yüksek olduğu görülmüştür. Programlama öğretiminde öğrenciler, fiziksel programlama aracı destekli Python programlama dili öğrenirken uygun söz dizimi yazma, verilen bir söz dizimini test etme ve hata ayıklama becerilerinin yanı sıra; bu süreçte programlama kavramlarından

değişken, koşullu ifade, döngü ve fonksiyonu kullanabildikleri gözlemlenmiştir. Uygulama sonunda öğrencilerle yapılan odak grup görüşmesinde; öğrenciler fiziksel programlama aracı destekli metin tabanlı programlama dili ile programlama yaparken motive oldukları ve daha çok programlama etkinliği yapmak istediklerini belirtmişlerdir. Ayrıca fiziksel programlama aracı ile programlama kavramlarını daha kolay anladıklarını ve programlama yaparken hatalarını daha rahat bulabildiklerini ifade etmişlerdir. Öğrencilerin çoğunluğu daha önce program yazarken döngü ve liste kullanımında sorun yaşarken fiziksel programlama aracı ile metin tabanlı programlama öğrenirken zorlanmadıklarını ifade etmişlerdir. Tüm bu bulgular ışığında, fiziksel programlama aracı destekli programlama öğretiminin öğrencilerin programlama kavram ve süreçlerine ilişkin öğrenmelerini olumlu yönde etkilediği ortaya çıkmıştır.

Anahtar Kelimeler: Fiziksel programlama, programlama öğretimi, metin tabanlı programlama, bilgi işlemsel düşünme, bilgi işlemsel düşünmeye yönelik özyeterlik algısı.

ABSTRACT

Ezgi Arzu YURDAKÖK

The Effect of Physical Programming Tool Supported Programming Teaching on Secondary School Students' Computational Thinking Skills and Self-Efficacy Perceptions

Başkent University

Institute of Educational Sciences

Computer Education and Instructional Technology Major Science

Master's Thesis Program

August 2022

This study aims to examine the effect of teaching text-based programming with the physical programming tool on secondary school students' computational thinking skills and self-efficacy perceptions. The study was carried out in a sequential explanatory design in the mixed research method type. The study group consisted of 85 secondary school students studying in a private secondary school in Ankara. Within the scope of the study, the micro: bit which is a physical programming tool was used to teach Python programming for 6 weeks. As data collection tools, the Self-Efficacy Perception Scale for Computational Thinking Skills, Bebras: International Challenge on Informatics and Computational Thinking Tasks, tests focused on programming tools, concepts and processes, and semi-structured interview questions were used. According to the findings obtained from the pre-test and post-test, a significant and positive difference was found in the computational thinking skills and the self-efficacy perceptions towards computational thinking skills of the students. As a result of the programming teaching, it was seen that the students learned the programming concepts and processes. While learning Python programming with the physical programming tool, students not only gained the skills to write appropriate syntax, test and debug but also learn programming concepts such as variable, conditional expression, loop and function. In the focus group interview held with

the students, the students stated that they were motivated while learning Python programming with the physical programming tool and they wanted to do more programming activities. They also stated that they understand programming concepts more easily with the physical programming tool and that they can find errors more easily while programming. The majority of the students stated that they had problems using loops and lists while writing programs, but they did not have any difficulties while learning to program with the physical programming tool. As a result, it has been revealed that teaching programming with the physical programming tool has a positive effect on students' learning of programming concepts and processes.

Key Words: Physical computing, programming education, text-based programming language, computational thinking skills, self-efficacy perceptions towards computational thinking skills.

ÖNSÖZ

Alanda yer alan programlama öğretimlerini incelediğimde ve verdiğim derslerde karşılaştığım öğrenci soruları, sorunları ve dönütlerini göz önünde bulundurarak tez konumu seçmeye çalıştım. Değerli danışmanım sayesinde tanışmış olduğum fiziksel programlama aracı micro:bit’ in ülkemizde bilinmediğini ve kullanımının sınırlı olduğunu fark ettim. Araştırmalarım sonucunda programlama öğretimi kavramları için oldukça çok ve farklı kullanım alanı olduğunu keşfettim. Bu doğrultuda programlama çalışmak istediğim tez çalışmamda bu fiziksel programlama aracını da kullanmak istedim. 21. yüzyıl becerilerinin alanyazında sıklıkla çalışıldığını buna karşın özellikle programlama öğretimi alanında bilgi işlemsel düşünme becerilerini geliştirici etkinliklerin ülkemizde öğretim programlarında yeterince yer bulmadığını fark ettim. Bu sebeple araştırmamı bu yönde şekillendirdim. Bunların sonucunda tez konusu olarak “Fiziksel Programlama Aracı Destekli Programlama Öğretiminin Ortaokul Öğrencilerinin Bilgi İşlemsel Düşünme Becerilerine ve Özyeterlik Algılarına Etkisi” olarak karar verdim. Bu süreçte hem alanyazın taraması yaparak hem de akademisyenlerden, alan öğretmenlerinden görüş ve fikirlerini alarak çalışmamı şekillendirdim. Tez aşamasında, maalesef küresel çapta dünyayı sarsan covid-19 pandemisi sebebiyle uygulama yapmakta sıkıntı yaşadım. Okullar açıldığında uygulama aşamasına geçebildim. Çalışmamı hem eğitimcilere hem de alana katkı sağlaması amacıyla gerçekleştirdim.

İÇİNDEKİLER

	Sayfa
TEŞEKKÜR	İ
ÖZET	İİ
ABSTRACT	İV
ÖNSÖZ	VI
TABLolar LİSTESİ	X
ŞEKİLLER LİSTESİ.....	Xİİ
SİMGELER VE KISALTMALAR LİSTESİ.....	Xİİİ
I. BÖLÜM	1
GİRİŞ.....	1
1.1. PROBLEM DURUMU	1
1.2. ARAŞTIRMANIN AMACI.....	6
1.3. ARAŞTIRMANIN ÖNEMİ.....	7
1.4.SAYILTIAR	9
1.5.SINIRLILIKLAR	9
1.6.TANIMLAR	10
II. BÖLÜM	11
KURAMSAL ÇERÇEVE	11
2.1. PROGRAMLAMA ÖĞRETİMİ VE BİLGİ İŞLEMSEL DÜŞÜNME	12
2.1.1. Programlama öğretiminde yöntem ve yaklaşımlar	17
2.1.2. Programlama öğretiminde metin tabanlı programlama dilleri	21
2.1.3.Fiziksel programlama araçları	23
YÖNTEM	30
3.1. ARAŞTIRMA MODELİ	30
3.2. ÇALIŞMA GRUBU	32
3.3. VERİLERİN TOPLANMASI	32
3.3.1. Bilgi işlemsel düşünme becerilerine yönelik öz yeterlik algısı ölçeği (BİDBÖA)	34

3.3.2. Bilge kunduz uluslararası enformatik ve bilgi işlemsel düşünme etkinliği görevleri.....	35
3.3.3. Ders süreci gözlem formu.....	39
3.3.4. Ders sonu değerlendirme soruları.....	41
3.3.5. Programlama aracı, kavram ve süreçlerine yönelik kazanım testi	41
3.3.6. Odak grup görüşmesi.....	44
3.4. VERİLERİN ANALİZİ.....	45
3.5. ARAŞTIRMA SÜRECİ.....	47
3.6. ARAŞTIRMACININ ROLÜ	50
BULGULAR VE TARTIŞMA.....	52
4.1.BULGULAR	52
4.1.1. Uygulama öncesi ve sonrası öğrencilerin bilgi işlemsel düşünme becerilerine yönelik öz yeterlik algılarına ilişkin bulgular;.....	52
4.1.2. Uygulama öncesi ve sonrası öğrencilerin bilgi işlemsel düşünme becerilerine ilişkin bulgular;.....	56
4.1.3. Uygulama sürecinde ve sonrası öğrencilerin fiziksel programlama aracı destekli metin tabanlı programlama kavram ve süreçlerine yönelik öğrenme düzeylerine ilişkin bulgular;.....	59
4.1.4. Fiziksel programlama aracı destekli metin tabanlı programlama öğretimine yönelik öğrencileri görüşlerine ilişkin bulgular;	64
4.2. TARTIŞMA.....	69
SONUÇ VE ÖNERİLER	77
5.1. SONUÇ	77
5.2. ÖNERİLER	82
KAYNAKLAR.....	84
EKLER	114
EK 1: BİLGE KUNDUZ ULUSLARARASI ENFORMATİK VE BİLGİ İŞLEMSEL DÜŞÜNME GÖREVLERİ	
EK 2: PROGRAMLAMA ARACI VE KAVRAMLARINI ÖĞRENİM DÜZEYİ ÖLÇME TESTİ	
EK 3: HAFTALIK DEĞERLENDİRME TABLOLAR VE SONUÇLAR	
EK 4: DERS ANLATIM – ÇALIŞMA GÖRÜNTÜLERİ	
EK 5: DERS GÖZLEM FORMU	

EK 6 : ETİK KOMİSYONU ONAY BİLDİRİMİ

EK 7: YÜKSEK LİSANS TEZ ÇALIŞMASI ORJİNALLİK RAPORU



TABLolar LİSTESİ

Sayfa

Tablo 2.1. Problem çözme süreci olarak bilgi işlemsel düşünme (kalelioğlu ve ark., 2016).	16
Tablo 3.1. Cinsiyet ve sınıflarına göre öğrencilerin frekans ve yüzde dağılımı.....	32
Tablo 3.2. BİDBÖA güvenilirlik katsayıları.	34
Tablo 3.3. Bilge kunduz görevleri uzman değerlendirmesi.....	36
Tablo 3.4. Yıllara göre seçilen görevlerin isimleri, zorluk düzeyleri ve kazanımları.	37
Tablo 3.5. Seçilen görevlerin enformatik anlamda ölçtüğü kavramların görevler bazında listesi.....	38
Tablo 3.6. Ders süreci gözlem formu kriterleri	40
Tablo 3.7. Programlama aracı, kavram ve süreçlerine yönelik kazanım testi sorularının kazanım frekansı.....	42
Tablo 3.8. Programlama aracı ve kavramlarını öğrenim düzeyi ölçme testi sorularının programlama yapıları ve bid becerileri açısından değerlendirilmesi.....	43
Tablo 3.9. Öğretim programı tablosu.	49
Tablo 4.1. BİDBÖA ölçeği ve öğrenciye uygulanan ön testin cronbach's alpha katsayısı.	53
Tablo 4.2. BİDBÖA ölçeğinin ve alt faktörlerine ait ön test sonuç tablosu.....	54
Tablo 4.3. BİDBÖA ölçeğinin ve alt faktörlerine ait son test sonuç tablosu.	55
Tablo 4.4. BİDBÖA ölçeğine ait ön test ve son test karşılaştırma sonuçları.	55
Tablo 4.5. Bilge kunduz uluslararası enformatik ve bilgi işlemsel düşünme görevleri ön testinin istatistik sonuçları.	57
Tablo 4.6. Bilge kunduz uluslararası enformatik ve bilgi işlemsel düşünme görevleri son testinin istatistik sonuçları.	58
Tablo 4.7. Bilge kunduz uluslararası enformatik ve bilgi işlemsel düşünme testi ön test ve son test tek örneklem t-testi sonuçları.	58
Tablo 4.8. Haftalık değerlendirme analizi	60

Tablo 4.9. Haftalık deęerlendirme ve kazanım testi analiz sonuçları.	63
Tablo 4.10. Öğrencilerin fiziksel programlama aracı ile metin tabanlı programlama öğrenmelerine ilişkin görüşleri.	64
Tablo 4.11. Öğrencilerin fiziksel programlama aracı ile metin tabanlı programlama öğrenmenin iyi yönlerine ilişkin görüşleri	65
Tablo 4.12. Öğrencilerin fiziksel programlama aracı ile metin tabanlı programlama öğrenme sürecine yönelik önerilere ilişkin görüşleri	67
Tablo 4.13. Öğrencilerin fiziksel programlama aracı ile metin tabanlı programlama öğrenme ile ilgili dięer düşüncelerine ilişkin görüşleri	68



ŞEKİLLER LİSTESİ

	Sayfa
Şekil 1.1. 21. Yüzyıl becerilerinin üç kategoride gruplanmaktadır.....	2
Şekil 2.1. Micro: bit micro denetleyici kart ve micro: bit python editörü ekran görüntüsü.	26
Şekil 3.1. Sıralı açıklayıcı desenin veri ve analiz şeması (Toraman, 2021).....	31
Şekil 3.2. Araştırmanın uygulama süreci.....	33



SİMGELER VE KISALTMALAR LİSTESİ

Kısaltma	Açıklama
MEB	: Millî Eğitim Bakanlığı
TDK	: Türk Dil Kurumu
BİD	: Bilgi İşlemsel Düşünme
BİDBÖA	: Bilgi İşlemsel Düşünme Becerilerine Yönelik Öz Yeterlik Algısı Ölçeği
WEF	: World Economic Forum – Dünya Ekonomik Forumu.
ISTE	: International Society for Technology in Education- Uluslararası Eğitim Teknolojileri Topluluğu
CSTA	: Computer Science Teachers Association, Bilgisayar Bilimi Öğretmenleri Derneği

I. BÖLÜM

GİRİŞ

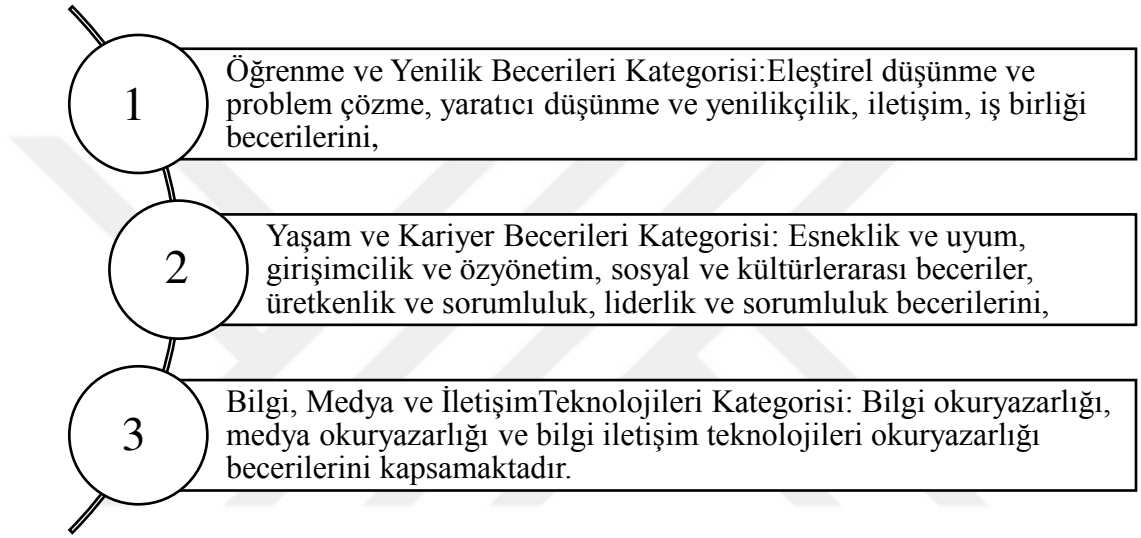
1.1. Problem Durumu

İnsanlık tarihine bakıldığında, insan var olduğu zamandan bugüne kadar öncelikle kendini, sonra da çevresini sorgulamaya başlamıştır. Çünkü merak insanın doğasında vardır. İnsan bu merakı sebebiyle tarih boyunca en hızlı gelişebilme gücünü gösteren canlı olmuştur. Her yüz yılda bir çağ atlamış ve yeni keşiflerle, icatlarla hem kendini hem de çevresini dönüştürüp geliştirebilmiştir. Ersöz ve Özmen'in (2020) bahsettiği gibi özellikle sanayi devriminin sonunda insanoğlu köklü bir değişimin ve dönüşümün içerisine girmiştir. Bu dönüşümü hızlı bir şekilde yapabilmesini sağlayan ise teknolojidir. Bu teknolojik gelişmeler ile bilgi toplumunun geldiği son nokta olan Endüstri 4.0 çok hızlı bir şekilde dünya gelişiminde yerini almış ve teknoloji gittikçe gelişerek hayatımızın bir parçası olmuştur. Bunun bir sonucu olarak da toplumlar daha hızlı bilgiye ulaşan, kullanan, düşünen ve üreten insan gücüne ihtiyaç duymaktadır. Bu gelişmeler neticesinde bireylerden beklenen, ihtiyacı olan bilgiye ulaşmaları ve onu etkili kullanma becerisine sahip olmalarıdır. Böyle bireyleri yetiştirmek için okullara ve öğretmenlere çok iş düşmektedir. Onlardan beklenen bilgiye etkin erişerek aynı zamanda teknolojiyi doğru kullanabilen bireyler yetiştirmeleridir (Karataş, Güneş ve Coşkun, 2012).

Akkoyunlu ve Kurbanoglu (2003), eğitim sürecine bakıldığında bilgi ve iletişim teknolojilerinin öğrenme ve öğretme sürecinde bütünleştirilmesinin zorunlu hale geldiğini belirtmişlerdir. Araştırmalar gösteriyor ki 1980'den sonra, bilişim teknolojilerinde görülen hızlı gelişmeler, eğitimi doğrudan etkilemiş ve değişimi de daha hızlı tetiklemiştir. Kültekin (2006) çalışmasında günümüzde eğitim uzmanlarının, bireyleri bilgi toplumuna hazırlarken aynı zamanda gelecek yıllardaki mesleklerin bir üyesi yapma görevini de üstlenmiş olduklarını ifade etmektedirler. Eğitim alanında yaşanan bu gelişmeler, öğretim sürecini ve değerlendirme yaklaşımlarını oldukça değiştirmektedir. Seferoğlu (2015), eğitimde yaşanan gelişmeler neticesinde eğitimin merkezinde yer alan öğrencilerin öğrenme alışkanlıklarının da değiştiğini belirtmektedir. Bu nedenle yeni bilgi ve becerilerin kazandırılmasının yanı sıra bu kazanımların nasıl öğrencilere aktarılacağı da eğitimin bir sorunu olarak ortaya çıktığını ifade etmişlerdir (Seferoğlu, 2015; Ersöz ve Özmen, 2020). Öğrencilerin ilgi ve yetenekleriyle birlikte aynı zamanda toplumsal

değişimlerin de meslek tercihlerinde etkili olduğu ifade edilmektedir (Gülen ve Dönmez, 2021). Konu hakkında ilgili kurum ve kuruluşların belirttiği hem de alanyazında yer almış pek çok 21. yüzyıl becerisi tanımı ile karşılaşılmaktadır. İlgili alanyazında en çok kabul gören 21. Yüzyıl Becerileri Ortaklığı (Battelle for Kids (P21), 2021) tarafından hazırlanmış olanıdır ve 21. yüzyıl becerileri üç kategoride gruplanmaktadır. Bu kategoriler Şekil 1.1.'de özetlenmiştir.

Şekil 1.1. 21. Yüzyıl becerilerinin üç kategoride gruplanmaktadır.



Şekil 1.1.' de görüldüğü gibi 21. yüzyıl becerileri temel konuları içermekle birlikte; yaşam ve kariyer becerileri ile bilgi, öğrenme ve yenilik becerileri, medya ve teknoloji becerileri olarak üç ana beceri alanından oluşmaktadır (Partnership for 21st Century Learning, 2021; Yalçın, 2018; Kylonen, 2012).

Eğitim bilimciler, bireylerin 21. yüzyılda sahip olması gereken becerileri açıklayarak ifade etmekte ve çağdaş eğitime yön vermektedirler (Günüç, Odabaşı ve Abdullah, 2013; Çiftci, Yayla ve Sağlam, 2021). Geleceğin bireylerinde olması beklenen bu beceriler; eleştirel düşünebilme, problem çözme becerisine sahip olabilme, iş birliği, liderlik yapabilme, uyum sağlayabilme, merak ve hayal gücü, girişimcilik, inisiyatif alabilme, bilgiye ulaşabilme, etkili şekilde sözlü ve yazılı iletişim kurabilme ve analiz edebilme şeklinde sıralanmaktadır (Dağhan, Kibar, Çetin, Telli ve Akkoyunlu, 2017; Gelen, 2017; Erten, 2020). Bahsedilen bu becerileri geliştirebilmek adına eğitimciler, öğrencilere gerçek yaşam problemlerini çözebilecekleri, iş birliği içerisinde ürün geliştirebilecekleri, değişen

süreçler ve teknolojilere uyum sağlayabilecekleri fırsatlar sunmalıdır. Geleceğin mesleklerine ve 21. yüzyıl becerilerine öğrencileri hazırlayabilmek, eğitim sistemlerinin en önemli hedeflerinden biri olduğu ifade edilmiştir (Cengiz, 2019; Karal, Şılbır ve Yıldız, 2017).

21. yüzyıl becerilerinin kapsamının belirlenmesi, sınıflandırılması, öğretilmesi ve değerlendirmelerinin nasıl olacağı konusunda birçok kurum ve kuruluş çalışmalar yapmaktadır. Uluslararası alanyazın incelendiğinde, 21. yüzyıl becerileri hakkında çalışmaları ile öne çıkan bazı kuruluşlar şu şekilde sıralanabilir; Asya-Pasifik Ekonomik İşbirliği Forumu (APEC), 21. Yüzyıl Becerileri Ortaklığı (Partnership For 21st Century Learning), Kuzey Merkez Bölge Eğitim Laboratuvarı (North Central Regional Educational Laboratory), Ekonomik Kalkınma ve İşbirliği Örgütü (Organization for Economic Co-operation and Development), Uluslararası Eğitimde Teknoloji Topluluğu (International Society for Technology in Education), 21. Yüzyıl Becerilerinin Öğretimi ve Değerlendirilmesi (Assessment and Teaching of 21st Century Skills), Avrupa Komisyonu (European Commission). 21. yüzyıl becerisini destekleyen bu platformlar; öğrencilerin ve toplumun, sosyal ve ekonomik ihtiyaçlarına cevap verebilmek adına hem okullarda hem de eğitimde reformlara ihtiyaç duyulduğunu belirtmektedirler (Ananiadou ve Claro, 2009). OECD ülkeleri 1997 yılında “Yetkinliklerin Tanımı ve Seçimi: Teorik ve Kavramsal Temeller” (DeSeCo, 2022) projesini başlatarak bilgi, beceri ve yetkinliklerin bireyler ve toplumlar için çok önemli olduğunu ifade etmişlerdir. Aynı projede iyi eğitimli, bilgili, yüksek nitelikli bir vatandaşın, bugünün ve geleceğin zorluklarıyla yüzleşmede önemli bir rol oynayacağı öngörülmektedir. Ayrıca hayat boyu öğrenme için önemli olan bireysel yetkinlikleri geliştirmeyi objektif olarak değerlendirebilmek adına bir çerçeve oluşturmaya çalışmışlardır (Çiftci, Yayla ve Sağlam, 2021).

“Mesleklerin Geleceği 2020” (WEF, 2020) Dünya Ekonomi Forumu tarafından açıklanan raporda, yeni nesil teknolojileri geliştirebilme, uygulayabilme ve kullanabilme ile ilgili mesleklerin geleceğin meslekleri olarak beklediklerini ifade ettiler. İçinde bulunduğumuz teknoloji çağında hazırlanan bu raporlar ve analizler gösteriyor ki, şu anda var olmayan ama önümüzdeki birkaç yıl içerisinde meslek kolu olacak yeni alanlar ortaya çıkacak gibi görünmektedir (WEF, 2020; NRC, 2012). Örneğin; Algoritma teknisyenliği, endüstriyel veri bilimciliği, bulut hesaplama uzmanlığı, dron mühendisliği, etkileşimli arayüz tasarımcılığı, robot koordinatörlüğü, veri güvenliği uzmanlığı, 3d yazıcı mühendisliği,

it/iot çözüm mimarlığı, giyilebilir teknoloji tasarımcılığı ve iş terapistleği gibi yeni meslekler de ortaya çıkabilecektir (Ecevit ve Kaptan, 2019).

Günümüzde öğrencilerden beklenen sadece tüketen değil aynı zamanda üreten bireyler olmalarıdır (Kalelioğlu, 2015). Bu sebeple öğrencilere, 21. yüzyıl becerileri içerisinde yer alan ve günümüzde oldukça önemli teknolojik okuryazarlık becerisini de kapsayan programlama becerisi kazandırılmalıdır (Lau ve Yuen, 2011). 2013 yılında TEDX konuşmasında Resnick, gelecekte iş dünyasında bilgisayar programcısı ve bilgisayar bilimcisine ihtiyaç olacağını ifade etmiştir. Geleceğin meslek sahipleri olacak günümüz öğrencileri programlama becerisinin yanında bilgi ve teknoloji okuryazarı olmaları, problem çözmeleri, işbirlikçi çalışmaları, eleştirel düşünceleri, analiz ve sentez yapabilmeleri gibi niteliklere sahip olmaları amaçlanmaktadır. Bu becerileri kazandırmaya yönelik öğretim programlarının da gözden geçirilmesi gerekmektedir (Sırakaya, 2019). İfade edilen bu becerilere ek olarak bilgi işlemsel düşünme becerisi de sahip olunması gereken önemli bir beceri olarak ortaya çıkmaktadır (Wing, 2010). Alanyazında bilgi işlemsel düşünme becerisinin pek çok tanımı olduğu gibi pek çok farklı alt becerileri de tanımlanmıştır. Wing'e (2008) göre bilgi işlemsel düşünme bir çeşit analitik düşünmeyi kapsarken; Kalelioğlu, Gülbahar ve Kukul'a (2016) göre soyutlama, algoritmik düşünme ve problem çözmedir. ISTE (2015) ise eleştirel düşünme, yaratıcılık, problem çözme, algoritmik düşünme ve iş birliği bileşenlerinden oluştuğunu belirtmiştir. Çetin (2016), bilgi işlemsel düşünmenin bir bileşeni olarak problem çözme becerisinin üst düzey beceriler arasında olduğunu belirtmiştir. Bilgi işlemsel düşünmenin gelecek yıllarda eğitimin ayrılmaz bir parçası olacağını savunan Wing (2008), öğrencilere erken yaşlarda öğretilmesinin önemli olduğunu ifade etmiştir. Alanyazında bilgi işlemsel düşünmenin öğrencilere nasıl öğretileceği, öğrencilerin düşünme düzeyleri attıkça kavramların nasıl verileceği ve kavramları öğretirken kullanılacak olan araçların öğretime ve öğrenciye nasıl aktarılacağı konusunda sorulara cevap aranmaya çalışılmaktadır (Sayın ve Seferoğlu, 2016). Bu alandaki çalışmalara bakıldığında, bilgi işlemsel düşünmeyi öğretmeyi amaçlayan Türkiye'de ve dünyada pek çok kurum/kuruluş vardır. Bunlar hem web siteleri aracılığıyla hem de uluslararası düzenledikleri yarışma ve aktiviteler ile bu alanı tanıtmaya, öğretmen ve öğrencilere kazandırmayı hedeflemektedirler. Örneğin; Google Education ile ISTE'nin birlikte hazırlamış olduğu eğitimciler için hazırlanan ve her yıl kurs düzenlenmekte, bilgisayar biliminden bilgi işlemsel düşünmenin farklı olduğunu anlatmak, bu alanda farkındalık oluşturmak, disiplinler arası nasıl öğretilebileceğini öğretmek ve

öğretim programına nasıl entegre edilebileceğini anlatmak amacı ile hazırlanmıştır (Google Education, 2022). Türkiye’de hazırlanmış olan Keşf@ Projesi de öğretmenlerin bilgi işlemsel düşünme konusunda kendilerini geliştirebilmeleri amacı ile oluşturulmuş çevrimiçi eğitim platformudur (Keşf@ Bilinçli İnternet Hareketi-Öğretmen Portalı, 2022). 2013 yılından günümüze “Kod Saati”- Code.org adıyla bilinen tüm dünyada teknoloji, siyasi, spor ve müzik alanlarında ünlü isimlerinde destek verdiği, programlamanın öneminin anlatıldığı çevrimiçi bir platformdur. Her yıl aralık ayında yarışmalar düzenleyen Code.org, okullarda bilgisayar bilimine olan ilgiyi arttırmayı hedefleyen ve bilgi işlemsel düşünmeyi öğrencilere kazandırmaya çalışan bir sivil toplum kuruluşudur (Code.org, 2022). Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme Etkinliği (Bebras) uluslararası bir etkinliktir. İlk kez 2004 yılında Litvanya’da düzenlenmeye başlanan Bebras etkinliği son 7 yıldır ülkemizde Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme Etkinliği adıyla yapılmaktadır. Her yıl eğitim-öğretim yılının ilk döneminde ilkokul, ortaokul ve lise kademelerinde öğrencilere bilgi işlemsel düşünme becerilerine yönelik sorular sorulmaktadır. Etkinlik ile Ülkemizde öğrenci, öğretmen ve velilere bilgi işlemsel düşünme becerisi farkındalığı kazandırılmak hedeflenmektedir.

Yapılan çalışmalar incelendiğinde bilgi işlemsel düşünmenin matematiksel, tasarımsal ve algoritmik düşünme gibi düşünme becerilerini kapsarken bazı noktalarda da kesiştiği ifade edilmektedir (Lee ve ark., 2011). Bu kadar gerekli görülen ve geniş bir alanı etkileyen bilgi işlemsel düşünme becerisinin, uluslararası standartlarda yer alması da kaçınılmazdır (Üzümcü ve Bay, 2018). Endüstri 4.0 ile ortaya çıkan bilişim ile ilgili yapılan çalışmaların önemi ve sayısı arttıkça bu becerilerin kazanımı daha da önem kazanmıştır (Sayın ve Seferoğlu, 2016). Programlama öğretimi ile kazandırılması hedeflenen bu beceriler öğretim programlarının da güncellenmesi gerekliliğini ortaya çıkarmıştır. Ülkemizde ortaokul kademesinde bilişim teknolojileri ve yazılım derslerinde programlama öğretimi verilmektedir. Öğrenciler programlama dili öğrenme becerisini edinmekte ve süreç içerisinde öğrendiklerini performansa dönüştürmekte oldukça güçlük çekmektedirler (Özdinç ve Altun, 2014). Yapılan araştırmalarda öğrencilerin programlamayı zor algıladıklarını, programlama derslerini algılanması zor bir ders gibi düşündüklerini, birçok işlemin ve kavramın soyut kaldığını belirtmişlerdir. Bunlara ek olarak öğrendikleri bilgileri hayata geçirmekte ve somutlaştırmak da zorluk yaşadıkları ifade edilmektedir (Aşkar ve Davenport, 2009). Bu çalışmada kullanılmış olan fiziksel

programlama aracı sayesinde öğrencilerin programlama öğretimi gibi soyut süreçte yazacakları programların, somut sonucunu görerek daha kolay programlama kavramlarını öğrenebilecekleri ve buna bağlı olarak derse karşı motivasyonlarının daha fazla artacağı düşünülmektedir. Bu konuda çalışmalar yapan araştırmacılar fiziksel programlama aracı desteğiyle metin tabanlı programlama öğretimi verdikleri çalışmalarında bilgi işlemsel düşünme becerilerine yönelik öz yeterlik algılarında ve programlama kavramlarını öğrenme düzeylerinde artış gözlemlemişlerdir (Faber ve ark., 2017; Feldhausen, Weese ve Bean, 2018; Hambruch ve ark., 2009; González, López ve Castro, 2018).

Byrne ve Lyons (2001); programlama öğretiminde kullanılan geleneksel programlama öğretiminden kaynaklı öğrenen kayıplarına sebep olduğunu ifade etmektedirler. Alanyazın incelendiğinde, teknolojinin gelişmesinin sonucu olarak programlamanın her alanda kullanımının yaygınlaşmasına karşın, programlama öğretiminde farklı yöntem ve tekniklerin kullanılmadığı ayrıca bu konuda yapılan çalışmaların sayısının çok sınırlı olduğu görülmektedir (Kalelioğlu ve Gülbahar, 2014; Yurdakök ve Kalelioğlu, 2020). Bunun yanı sıra, programlama öğretimi araştırmalarında fiziksel programlama aracı ile programlama öğretilirken bilgi işlemsel düşünme becerisine odaklanılmadığı görülmektedir (Eryılmaz ve Deniz, 2019). Bu çerçevede alana katkı sağlayacağı düşüncesiyle programlama öğretiminde, fiziksel programlama aracı kullanılarak öğretilmesinin bilgi işlemsel düşünme becerisi üzerine etkisinin incelenmesi gerekli görülmüştür.

1.2. Araştırmanın Amacı

Bu araştırmanın amacı, ortaokul öğrencilerinin metin tabanlı programlama öğrenimi sürecinde fiziksel programlama aracı kullanarak programlamayı öğrenmelerinin bilgi işlemsel düşünme becerilerine etkisini ve programlamaya yönelik deneyimlerine ilişkin görüşlerini incelemektir. Bu bağlamda aşağıdaki araştırma sorularına yanıt aranmıştır:

Fiziksel programlama aracı destekli metin tabanlı programlama öğretiminin,

1. Öğrencilerin uygulama öncesi ve sonrası bilgi işlemsel düşünme becerilerine yönelik öz yeterlik algıları arasında anlamlı bir fark var mıdır?
2. Öğrencilerin uygulama öncesi ve sonrası bilgi işlemsel düşünme becerileri arasında anlamlı bir fark var mıdır?

3. Fiziksel programlama aracı destekli metin tabanlı programlama öğretimi sonrasında öğrencilerin programlama kavram ve süreçlerine yönelik öğrenme düzeyleri nasıldır?
4. Öğrencilerin fiziksel programlama aracı destekli metin tabanlı programlama öğretimine yönelik görüşleri nasıldır?

1.3. Araştırmanın Önemi

Bilişim Teknolojileri ve Yazılım dersinin 2012 yılında yayınlanan öğretim programı ile 5., 6. sınıflarda zorunlu olarak; 7. ve 8. sınıflarda seçmeli olarak okutulmaya başlanmıştır (MEB-TTKB, 2018). Ortaokul düzeyinde verilen Bilişim Teknolojileri ve Yazılım Dersinin öğretim programına bakıldığında, dersin amaçları içerisinde bireylerin algoritma tasarlama, problem çözme ve bilgi işlemsel düşünme becerileri gibi birçok kavramın kazandırılması hedeflendiği görülmektedir. Ders kapsamındaki yeterliklerine bakıldığında ise “bilişim teknolojileri okuryazarlığı, problem çözme, bilişim teknolojilerini kullanarak iletişim kurma, programlama, bilgi paylaşma, araştırma yapma, özgün ürün geliştirme, kendini ifade etme, bilgiyi yapılandırma ve işbirlikçi çalışma” olarak belirlenmiştir. Özellikle dersin amaç ve kazanımlarının yanı sıra okul dışı öğrenme ortamlarında da bu becerileri geliştirmeye ve zenginleştirmeye yönelik atölyeler, kurslar, etütler ve yarışmalar gibi çalışmalar yapılmaktadır. Bu kurs ve yarışmalara her geçen yıl ilginin arttığı görülmektedir.

Bunlara ek olarak, Türkiye’de 2018-2022 yılları arasında yayınlanmış yüksek lisans ve doktora tezlerinde en çok çalışılan konunun “21. yüzyıl becerileri” olduğu görülmüştür (Yurdakök ve Kalelioğlu, 2020). Cansoy (2018) araştırmasında sadece 21. yüzyıl becerilerinin eğitim programlarını etkilemekle kalmadığını bireylere kazandırılması beklenen bilgi, beceri ve yeterliklerdeki zorunlu değişikliklerin de olduğunu ifade etmiştir. OECD tarafından “The Future of Education and Skills Education 2030” ismiyle 2018 yılında hazırlanan raporda çalışmanın yapıldığı dönemde öğrencilerin belirtilen yılda yetişkin olacaklarını, birkaç mesleğin kaybolacağını ve bir takım yeni mesleklerin ortaya çıkacağı ifade edilmektedir (OECD, 2022). Buna göre geleceğin iş gücünü oluşturacak günümüz öğrencilerinin geleneksel yöntemlerin dışında, öğrenciyi merkeze alarak kendilerinden beklenen geleceğe yönelik bilgi ve beceriye sahip, temel kodlama ve bilgi

işlemsel düşünme becerilerini geliştirecek öğretim süreçleri tasarlanmasının önemi artmıştır (Gander ve ark., 2013).

Yapılan çalışmalara bakıldığında programlama öğretiminin ve programlamanın en çok çalışılan başlık olduğu görülmektedir (Eryılmaz ve Deniz, 2019). Bu çalışmada soyut işlemler döneminin başında olan öğrenciler, metin tabanlı programlama dilinde problem çözümünü soyut olarak ifade edecekler ve kullanılacak fiziksel programlama aracı ile kodların somut olarak çıktısını görebileceklerdir (Numanoğlu ve Keser, 2017; Yıldız, 2019; Ramazanoğlu, Demir, 2015; Sinap, 2017). Buna bağlı olarak öğretim süreci daha verimli, programlama kavramlarının öğrenilmesi daha kolay ve anlaşılır olabilmektedir. Bu çalışma ile öğrencilerin bilgi işlemsel düşünme becerilerinde ve metin tabanlı programlama python dili öğreniminde oluşan gelişim, bu becerinin ve programlama kavramlarının öğretilmesinde kullanılan fiziksel programlama aracının etkililiği ve uygulanacak programlama öğretiminin etkililiği araştırılmak istenmektedir. Alanyazını incelendiğinde programlama öğretiminde bilgi işlemsel düşünme ve metin tabanlı programlama ile ilgili çalışmaların daha az olduğu görülmektedir (Özel, 2019). Bu nedenle araştırma sonuçlarının bu alandaki yeni araştırmalara ve uygulamalara katkı sağlayacağı düşünülmektedir. Ayrıca, öğrencilerin fiziksel programlama aracı destekli metin tabanlı programlama öğrenmeye ilişkin düşünceleri ve önerileri ile ileride bu konuda yapılacak öğretim tasarım süreçlerine yönelik katkı sağlayacağı düşünülmektedir.

Çalışmada kullanılan plan ve yöntemler öğretmenlerin öğretim programlarına fiziksel programlama öğretimini nasıl entegre edebileceklerine yönelik ipuçları sunması açısından çalışmanın alana ışık tutacağı düşünülmektedir. Ülkemizdeki alanyazın incelendiğinde, programlama öğretimi çalışmalarında programlama araçları ile ilgili fiziksel programlama aracı olan micro:bit' in python programlama dilinin öğretiminde kullanılmadığı ve bu bağlamda bilgi işlemsel düşünme becerisinin değişken olarak ele alınmadığı görülmektedir (Yurdakök ve Kalelioğlu, 2020). Bu nedenle, yapılacak olan bu çalışma ile fiziksel programlama aracı destekli python dili ile metin tabanlı programlama öğretiminin öğrencilerin bilgi işlemsel düşünme becerilerine ve programlamaya yönelik düşüncelerine olan etkisinin araştırılması önem arz etmektedir.

1.4.Sayıtlar

“Fiziksel Programlama Aracı Destekli Programlama Öğretiminin Ortaokul Öğrencilerinin Bilgi İşlemsel Düşünme Becerilerine ve Özyeterlik Algılarına Etkisi” konulu araştırmaya ilişkin sayıtlar şu şekilde sıralanabilir;

1. Araştırmaya katılan öğrencilerin metin tabanlı programlama bilgisi bakımından birbirine yakın oldukları,
2. Araştırmanın veri toplama sürecinde ön test, son test ve odak grup görüşmesinde sorulan sorulara içtenlikle yanıt verdikleri kabul edilmiştir.

1.5.Sınırlılıklar

“Fiziksel Programlama Aracı Destekli Programlama Öğretiminin Ortaokul Öğrencilerinin Bilgi İşlemsel Düşünme Becerilerine ve Özyeterlik Algılarına Etkisi” konulu araştırmaya ilişkin sınırlılıklar şöyle sıralanabilir;

1. Araştırma, 7. ve 8. sınıf ortaokul öğrencileri ile sınırlıdır.
2. Araştırma uygulama süresi ile sınırlıdır (6 hafta ders anlatım süreci, 2 hafta veri toplama süreci olmak üzere toplamda 8 hafta ile sınırlıdır).
3. Araştırmada kullanılan fiziksel programlama aracı ile sınırlıdır.
4. Araştırma metin tabanlı programlama kapsamında Python dili ile sınırlıdır.
5. Bilgi işlemsel düşünme becerisine yönelik öz yeterlik algısının değerlendirilmesi kullanılan ölçek ile sınırlıdır.
6. Bilgi işlemsel düşünme becerilerin ölçülmesi 2016-2020 yıllarına ait Bilge Kunduz görevleri ile sınırlıdır.
7. Araştırma sonrası programlama ve aracın kullanımına yönelik değerlendirme amaçlı kullanılan çoktan seçmeli sorular ile sınırlıdır.
8. Araştırma sonrası öğrencilerin fiziksel programlama aracı ile programlama öğrenimi hakkındaki görüşleri ile sınırlıdır.

1.6.Tanımlar

Kodlama: Belirli bir programlama dili yazarak belli bir amaca ulaşmak için geliştirilen algoritmalar.

Algoritma: Bir problemin çözümü için adım adım yazılmasında izlenecek yolun mantıksal sıralanmasıdır.

Programlama Dili: Bilgisayar dili olarak isimlendirilen özel birtakım semboller ile kelimelerin bir araya gelerek oluşturdukları komutlar bütünü.

Micro:bit : BBC (British Broadcasting Corporation) micro:bit, yazılım ve donanımın birlikte nasıl çalıştığını tanıtan cep boyutunda bir bilgisayardır. LED ışıklı ekranı, butonları, sensörleri ve program yapabilen ve fiziksel olarak etkileşime girebilecek birçok giriş/çıkış özelliğine sahiptir (Micro:bit, 2022).

Python: Python, nesne yönelimli, yorumlamalı, modüler ve etkileşimli yüksek seviyeli bir programlama dilidir (Özel, 2019).

Bilgi İşlemsel Düşünme (BİD): Bilgisayarlar aracılığıyla problemleri formülleştirerek elde olan veriyi analiz etme, düzenleme, çözüm üretme, var olan kaynakları verimli bir şekilde analiz etme, ulaşılan çözümleri transfer ederek başka problemlerde kullanabilmek üzere geliştirilen düşünme süreci ya da problem çözme süreci olarak ifade edilmektedir (Yağcı, 2018; Wing, 2010).

II. BÖLÜM

KURAMSAL ÇERÇEVE

Öğrenme sürecinin araştırma ve keşfetme yolları ile oluştuğu ifade edilmektedir (Kürkçüoğlu, 2009). Sönmez (2000), bilişsel gelişimi bebeklikten yetişkinliğe kadar kişinin içinde bulunduğu sosyal ortamı anlama ve düşünme yollarının daha karmaşık ve etkin hale gelme süreci olduğunu ifade etmiştir. Bu çalışmada, öğretimin yapılacağı hedef kitle Piaget'in bilişsel gelişim evrelerinden soyut işlemler dönemine denk gelmektedir. Bu yaşa kadar ki süreçte bireyler okul döneminde elde ettikleri kazanımlarının üstüne yeni kazanımlar ekleyerek yeni deneyimler elde etmeye başlamaktadır. Bu dönemdeki öğrencilerin akıl yürütme becerisine sahip olduğundan (Tatlı, 2013) bağdaştırma, bitleştirme, sıralama ve parça-bütün ilişkilerini kurma gibi becerileri yapabileceği bilinmektedir. Böylece bireyler daha üst biliş düzeyinde dengeye sahip olabilmektedir (Vrugt ve Oort, 2008). Soyut işlemler döneminde olan bir çocuk, semboller düzeyinden bir seviye sonrası olan düşünce düzeyine ulaşabildiği söylenmektedir. Dolayısıyla düşünce düzeyine ulaştığında ise bir sorunu çözebilmek için gerekli olan farklı hipotezler geliştirebileceği gibi aynı zamanda bu geliştirdiği hipotezleri ayrı ayrı deneyerek çözümlenebilir. Çünkü soyut işlemler döneminde olan bir çocuk daha farklı ve ayrıntılı düşünebilir, sorunlara yaklaşımı daha düzenlidir ve akıl yürütebilme becerisi gelişmiştir (Cüceloğlu, 2000).

21. yüzyıl becerilerini öğreteceğimiz geleceğin meslek kollarında çalışacak soyut işlem dönemindeki öğrencilere programlama öğretimi verilirken pek çok soru eğitimcilerin aklına gelmektedir. Programlama ve bilgi işlemsel düşünme kavramları ışığında alanyazın incelendiğinde, eğitimci, bilgisayar bilimcisi ve Logo programlama dilinin yaratıcılarından biri olan Seymour Aubrey Papert dikkat çekmektedir. Piaget ile çalışmış olan Papert yapılandırmacılığı bir üst seviyeye çıkararak inşacılık (Şimşek, 2004; Üçgül, 2013) adıyla kullanılan öğrenme yaklaşımını geliştirmiştir. Papert'e göre Piaget'in teorilerinde bilişsel yaklaşımlar öne çıkarken duyuşsal ve devinişsel süreçler göz ardı edilmektedir (Ackermann, 2022; Papert ve Harel, 2022). Bu yaklaşımla Papert, en iyi öğrenme şeklinin gerçek dünyada anlaşılır ve somut bir ürün ortaya koyarak inşacılık yaklaşımının gerçekleşeceğini ifade etmiştir (Stager, 2007). Ayrıca Papert göre inşacılık yaklaşımı; öğrencilerin, deneyerek, yaparak ve yaşayarak yani keşif yoluyla, farklı fikir ve bilgilerle bağlantı kurarak bunları işbirlikçi şekilde eğlenerek ürüne dönüştürmesi olarak

açıklanabilir (Papert, 1993). Her iki yaklaşım incelendiğinde ayrıştığı noktalar olsa da benzer bazı ilkeler de bulunmaktadır. Bunlar öğrencilerin eğitim sürecinde aktif olması, önceki öğrenmelerin yeni öğrenmeler için işe koşulması ve öğrenci için farklı, yenilikçi, dikkat çekici etkinlikler içeren ortam hazırlanması şeklinde ifade edilmiştir (Akpınar, 1999). Papert'in inşacılık teorisine göre bilgi doğrudan doğruya öğrenciye aktarılamaz. Bunun yerine öğrencinin sürece aktif katılımı sağlandığında kalıcı bir öğrenme mümkündür (Baki, 2000). Bu sebeple öğrencinin soyut olguları aktif olarak inşa edebilmesi için Papert tarafından Logo programlama dili geliştirilmiştir. Logo sayesinde öğrencilerin soyut olguları öğrenmek için soyut işlemler dönemine gelmesinin beklenmesinin gerekmediği ifade edilmiştir. Öğrenciye uygun ortam ve araç ile farklı uygulamalarla desteklenerek soyut kavramlar öğretilabilir düşüncesi paylaşılmıştır (Akpınar, 1999). Alanyazın incelendiğinde, bir eğitimcinin soyutlama becerisini bilgisayar bilimlerini öğretebilmek için gerekli temel bir beceri olduğunu belirtmişlerdir (Bennedsen, Caspersen ve Kölling, 2008; Özçınar, 2018). Kramer'e (2007) göre bir bilgisayar programcısının diğer programcılar ile karşılaştırıldığında ne kadar iyi olduğu soyutlama becerisine sahip olup olmadığı ile ilgilidir. Soyutlama becerisi için bir programlamanın, Piaget 'nin soyut düşünme düzeyine ulaşmış olması gerektiğini belirtmiştir (Özçınar, 2018). Araştırma kapsamında tüm bu alanyazını değerlendirildiğinde bu çalışmanın kuramsal çerçevesi Papert'in inşacılık yaklaşımı üzerine tasarlanmıştır.

2.1. Programlama Öğretimi ve Bilgi İşlemsel Düşünme

Programlama, bir problemin çözümü amacıyla bilgisayara ne yapması gerektiğini sıralı talimatlardan oluşan bir dizi sözcüklerden ya da bloklardan oluşmaktadır (Karalar, 2019). Programlama yapabilmek için çözüm aşamaları adım adım oluşturulmalıdır (Hürnalı, 2021). Bu adımların öncelik sırası vardır. Bunlar sırasıyla; problemin tanımlanması, problemin parçalarına ayrılması, her bir parçanın çözümüne yönelik olarak algoritmaların geliştirilmesi, algoritmaların bilgisayarın anlayabileceği programlama dillerinden bir koda dönüştürmesi, kodların derlenmesi ve test edilmesi, hataların ayıklanması ve düzeltilmesi şeklindedir. Bu sıralı aşamaların sonucunda oluşturulan programlama, üst düzey düşünme becerileri gerektiren, problem çözme, eleştirel düşünme ve ürün oluşturma gibi becerilerin süreçte kullanılması gerekliliği sonucunda, öğrenilmesi zor ve karmaşık bir süreç olarak görülmektedir (Gomes ve Mendes 2007; Günbatır ve Karalar, 2018; Durak ve Sarıtepeci, 2018; Koorsse, Chilliers ve Calitz, 2015; Genç ve

Karakuş, 2011). Tüm bunlar gösteriyor ki programlama becerisi önemli bir konu olarak ortaya çıkmaktadır. Gelişen teknolojiyle beraber, programlama becerisi ve bu becerilerin öğrenilmesinin öneminin daha da çok arttığı belirtilmektedir (Akpınar ve Altun, 2014). Programlama süreci incelendiğinde programlama yapılırken, öğrencilerin bilişsel beceri ve üst düzey düşünme becerisini kullanması gerekmektedir (Kesici ve Kocabaş, 2007; Law, Lee ve Yu, 2010). Programlama öğretimi geleneksel yöntemler ile anlatıldığında öğrencilerin ilgisini çekmemektedir (Türk, 2019; Saygıner ve Tüzün, 2017; Alkar, 2017; Yükseltürk ve Altıok, 2015). Yapılan çalışmalara göre öğrencilerin bir kısmı programlamayı zor bulmakta; buna bağlı olarak uzmanlık gerektirdiğini düşünmekte ve sadece ileri düzeyde programlama okuryazarlığına sahip kişilerin programlama yapabileceklerini düşünmektedirler (Başer, 2013; Lahtinen, Ala-Mutka ve Järvinen, 2005).

Programlama becerisinin bilişsel gelişime olan katkıları sebebi ile araştırmacılar programlama öğrenimi sayesinde bu becerinin gelişeceğini aynı zamanda 21. yüzyıl becerilerine de doğrudan katkı sağlayacağını ifade etmişlerdir ve 21. yüzyıl becerileri içerisine programlamayı da dahil etmişlerdir (Wing, 2010; Sayın ve Seferoğlu, 2016). Programlama öğretimi sürecinde fiziksel programlama araçları kullanılarak öğretimin yapılması hakkında alanyazın incelendiğinde, öğrencilerin problem çözme ve analitik düşünme becerileri, ürüne dönük projeler geliştirme, karmaşık problemlere çözüm üretme becerisi kazanma, yaparak-yaşayarak öğrenme deneyimleri yaşamaları gibi birçok becerinin gelişmesine yardımcı olacağı ifade edilmektedir (Çakıroğlu, Sarı ve Akkan, 2011). Programlama öğrenme sürecinde karşılaşılan pek çok sorunun döngü, değişken, fonksiyon, dizi vb. programlama diline özgü kavramların soyut olması ve karmaşıklığından kaynaklandığını gösteren çalışmalardan bahsedilmektedir (Ozoran, Çağıltay ve Topallı, 2012). Özellikle programlamaya yeni başlayan öğrenciler için karmaşık mantıksal yapıda komutlar yazarken bir de sözdizimi (syntax) hatalarıyla uğraşmak zorlamaktadır (Choi, 2013). DuBoulay (1986), yaptığı çalışmasında programlama öğretiminde yaşanan zorlukları 4 başlıkta açıklamıştır. Bunlardan birincisi programlamanın neden yapılacağını ve problemin ne olduğunu anlama, programlama dilindeki kodların dizilimini anlamlandırma, programlama yapılırken kullanılan yapıları anlamlandırma ve programı yazabilmek için gerekli olan ileri düzey bilgi ve becerilere sahip olma şeklinde açıklamıştır. Öğrenciler bu zorlukların programlama becerilerinin öğrenilmesine engel olduğunu düşünmektedir. Buna bağlı olarak motivasyonlarının düştüğünü ve zamanla

programlamaya olan ilgilerinin azaldığını, sonuçta çaba sarf etmeyi bıraktıkları düşünülmektedir (Ozoran, Çağıltay ve Topallı, 2012; Ersoy, Madran ve Gülbahar, 2011).

Yapılan çalışmalar göstermektedir ki programlama becerisi öğretiminde algoritmik düşünme ve matematiksel düşünme becerileri birbirini geliştirmekte ve birbirlerini tamamlamaktadır (Lee ve ark., 2011). Algoritmik düşünme yeni bir kavram olmasa da özellikle bilgisayar konuları ile ilgili kodlama ve bilgi işlemsel düşünme konuları günümüzde yaygınlaştıkça daha da öne çıkmaktadır. Bilgisayar öğretiminin temelini oluşturan algoritma öğretimi çeşitli boyutlar açısından bilgi işlemsel düşünmenin kapsamına girmektedir. Bu nedenle bilgi işlemsel düşünmenin bilgisayar biliminden daha geniş bir alanı kapsadığını ve birden fazla düşünme becerisine katkı sağlayacağı düşünülmektedir (Üzümcü ve Bay, 2018).

Alan yazını incelendiğinde bilgi işlemsel düşünme kavramının ilk olarak Papert tarafından kullanıldığı görülmektedir. Kavramın kullanımı konusunda Wing (2010) öne çıksa da Papert (1996) geometrik problemlerin çözümü sırasında bilgisayarın nasıl kullanılabileceğini ifade ederken BİD'den yararlanmıştır. (Gülbahar, Kert ve Kalelioğlu, 2018; Gülbahar, 2017). Bilgi işlemsel düşünme (computational thinking) kavramında geçen “computational” kelimesinin Türkçe alanyazında pek çok farklı şekilde; “hesaplamalı”, “bilişimsel”, “bilgi işlemsel” ve “bilgisayarca” (Özçınar ve Öztürk, 2018; Korkmaz, Çakır, Özden, Oluk ve Sarioğlu, 2015; Yıldız, Çiftçi ve Karal, 2017) gibi kullanıldığı gözlemlenmektedir. Son yıllarda yapılan çalışmalarda ve resmî kurumların kullanımlarında ise “bilgi işlemsel” olarak alanyazında kullanıldığı görülmektedir (Yüksek Öğrenim Kurulu, 2018; Talim ve Terbiye Kurulu, 2018; Kalelioğlu ve Gülbahar, 2015). “Computational thinking” kavramında da yine bazı farklılıklar alanyazında görülebilmektedir; “bilişimsel düşünme”, “bilgisayarca düşünme”, “hesaplı düşünme”, “kompütasyonel düşünme” ve “bilgi işlemsel düşünme” olarak karşılık bulmaktadır (Şahiner ve Kert, 2016; Yıldız, Çiftçi ve Karal, 2017; Özçınar ve Öztürk, 2018; Aldağ ve Tekdal, 2015; Korkmaz, Çakır ve Özden, 2015). Bu çalışmada ise “bilgi işlemsel düşünme” (Kalelioğlu ve Gülbahar, 2015) kavramı kullanılması uygun bulunmuştur. Bunun sebebi 2010 yılında Wing ve arkadaşlarının bilgi işlemsel düşünme kavramını tekrar tanımlamış olmalarıdır. Bu tanıma göre, bilgi işlemsel düşünmenin programlama mantığı sürecini daha anlaşılır tanımlamış olmalarıdır. Wing ve arkadaşarı (2010) tanımlarında; bir bilgi işleme birimi tarafından çözümlerin etkin şekilde yerine getirilecek

formda sunulması amacıyla formülleştirilen problem ve çözümleri içeren bir düşünme sürecidir şeklinde ifade etmişlerdir. Bu tanım bilgi işlemsel düşünme ile programlamanın iç içe bir süreç olarak bu çalışmada yer alan öğrenciler ile yapılmış olan eğitimin amacına da netleştirdiği düşünülmektedir.

ISTE (Uluslararası Eğitim Teknolojileri Topluluğu) (2016) belirlediği standartlar ile öğrencilerin edinmesi gereken bazı becerileri listelemiştir. Bilgi işlemsel düşünen, yetkin öğrenen, bilgiyi düzenleyen, dijital vatandaş, yaratıcı iletişim kurabilen, global iş birliği yapabilen, yenilikçi tasarımlar oluşturabilen şeklinde beceriler sıralanmaktadır. Beceriler incelendiğinde bilgi işlemsel düşünme becerisi ISTE'nin (2016) günümüz dünyasında öğrencilerde bulunması gereken en önemli becerilerden biri sayılmıştır. ISTE ve CSTA (Bilgisayar Bilimi Öğretmenleri Derneği) tarafından hazırlanan ortak bir çalışmada ise bilgi işlemsel düşünme bir problem çözme süreci olarak ifade edilmiş ve belli özelliklere sahip olması gerektiği belirtilmiştir.

- Algoritmik düşünme çerçevesinde çözümleri otomatikleştirme,
- Problemleri bilgisayar veya başka araçlar yardımı ile çözebilmek için formülleştirme,
- Bulunan çözümü farklı problemlere transfer etme ve genelleştirme,
- Kaynakları etkin ve etkili bir şekilde kullanarak en uygun çözümü/leri tanımlama, çözümleme ve uygulama,
- Modeller, simülasyonlar aracılığı ile verileri sunma,
- Mantıklı bir şekilde verileri düzenleme ve çözümleme (ISTE ve CSTA, 2011).

Yaptıkları çalışmada, problem çözme süreçlerine odaklanan bilgi işlemsel düşünme kavramının hangi alt boyutlara temelde sahip olduğunu belirlemeye çalışmışlardır (Kalelioğlu, Gülbahar ve Kukul, 2016). Her bir problemin çözme süreci adımı ile ilişkili bilgi işlemsel düşünme becerisi Tablo 2.1'de gösterilmektedir.

Tablo 2.1 Problem çözme süreci olarak bilgi işlemsel düşünme (Kalelioğlu ve ark., 2016).

Problem çözme süreci	Bilgi işlemsel düşünme süreci
Problemi Tanımlama	Soyutlama, ayrıştırma
Veri toplama, sunma ve görselleştirme	Veri toplama, veri analizi, örüntü tanıma, kavramsallaştırma, veri sunma
Çözümü planlama, seçme ve genelleme	Matematiksel sorgulama, algoritma ve süreçleri oluşturma, eş zamanlı çalışma
Çözümü uygulama	Otomasyon, modelleme ve simülasyon
Çözümü değerlendirme ve iyileştirme için geliştirme	Test etme, hata ayıklama, genelleme

Alanyazın incelendiğinde pek çok bilgi işlemsel düşünme tanımını olması yanında aynı zamanda bilgisayar programlama becerisini açıklamada bilgi işlemsel düşünme ilkelerinin kullanıldığı görülmektedir. Sysło ve Kwiatkowska (2013) bilgi işlemsel düşünmeyi; bilgisayar programlama ilkelerini merkeze alan bir dizi düşünme becerisi olarak ifade etmektedirler. Yapmış oldukları alan yazını çalışmalarında bilgi işlemsel düşünme ilkelerini incelemişler ve bilgi işlemsel düşünmenin problem çözme, örüntü tanıma, algoritma, soyutlama, problem ayrıştırma ve değerlendirmeyi kapsadığını ifade etmişlerdir (Sadik, Leftwich ve Nadiruzzaman, 2017). Bilgi işlemsel düşünmenin alt boyutlarına bakıldığında; algoritma tasarımı, soyutlama, ayrıştırma, otomasyon, veri çözümleme, veri toplamaeş zamanlı çalışma, örüntü genelleştirme, veri sunma, örüntü tanıma ve modelleme olarak sıralanmaktadır (Kalelioğlu ve Gülbahar, 2015). Programlama öğretiminde, bahsedilen bu kavram ve süreçler bilgi işlemsel düşünmeyi geliştirir ve destekler (ISTE ve CSTA, 2011).

Gülbahar, Kert ve Kalelioğlu'nun da (2018) belirttiği gibi yeni bir kavram olan bilgi işlemsel düşünme becerisinin kazandırılması için pek çok ülke, problem çözmeyle beraber programlama yöntemiyle öğretilmesi konusunda çalışmaya başlamıştır. Öğretilmesi boyutunda pek çok çalışma olsa da özellikle Türkçe alanyazında değerlendirme amacıyla geliştirilmiş yeterli araç yoktur (Demir ve Seferoğlu, 2017). Bilgi işlemsel düşünme becerisi bir problem çözme süreci olarak ifade edilmiştir (Kalelioğlu ve ark., 2016). Problem çözme sürecinde veriler kullanarak algoritma tasarımları ile programlama yapmak hedeflenir. Türk Dil Kurumu tarafından program, "bilgisayara bir işlemi yaptırmak için yazılan komutlar dizisi"dir şeklinde ifade edilmektedir (TDK, 2021). Alanyazınına

bakıldığında ise programlamayı kullanıcıların belirli bir problemin çözümüne yönelik gerekli komutların programlama dilleri kullanılarak kod satırları şeklinde yazılması, derlenmesi ve çalıştırılması olarak ifade edilmektedir (Arabacıoğlu, Bülbül ve Filiz, 2007; Kesici ve Kocabaş, 2001). Bilgi işlemsel düşünme kavramını özetleyecek olursak, çeşitli özellikleri ve yaklaşımları içerisinde barındıran bir problem çözme sürecidir. Bilgi işlemsel düşünmenin alt boyutlarından Yeni (2018); algoritma düşünme, otomasyon, ayrıştırma, mantıksal akıl yürütme, hata ayıklama, değerlendirme, genelleme/desen tarama ve soyutlama şeklinde bahsetmiştir. Bahsedilen bu kavram ve süreçler aslında programlama öğretiminde bilgi işlemsel düşünmeyi geliştirdiği ve desteklediği alanda yapılan çalışmalarda gözlemlenmektedir (Ma, Liu, Li, Fan ve Liang, 2019; ISTE ve CSTA, 2011).

Yapılan çalışmalar göstermektedir ki bilgi işlemsel düşünme becerisini geliştirmede programlama öğretiminin önemi büyüktür. Buna örnek olarak uluslararası düzeyde İngiltere Bilişim Dersi Öğretim Programı ve ders kitabını vermek mümkündür. Bu kitabın kapsamı ve yöntemi tamamen bilgi işlemsel düşünme becerileri çerçevesinde oluşturulmuştur (Department of Education, 2013). Bu sayede programlama becerisi ile eş zamanlı olarak bilgi işlemsel düşünme becerisinin de geliştirilmesi amaçlanmıştır. Bu çalışmada da bu nedenlerden dolayı ortaokul öğrencilerinin programlama öğretimi ile bilgi işlemsel düşünme becerilerini geliştirilmesi beklenmektedir.

2.1.1. Programlama öğretiminde yöntem ve yaklaşımlar

Yapılan çalışmalar göstermektedir ki son yıllarda makale ve tezlerde en çok çalışılan konular programlama ve öğrenme yöntemleridir (Eryılmaz ve Deniz, 2019; Yurdakök ve Kalelioğlu, 2020). Dünya’da olduğu gibi ülkemizde de programlama öğretimine gösterilen önem artmaktadır ve eğitimciler tarafından programlama öğretiminin nasıl yapılacağına ilişkin araştırmalar yapılmaktadır. Bilgisayar bilimi öğretimi pedagojik olarak incelendiğinde yapılan çalışmalarda inşacılık yaklaşımı programlama öğretiminde ön plandadır. İnşacılık yaklaşımını öne süren Papert, Piaget’ in yapılandırmacı kuramından etkilenmiştir ve inşacılık kavramını ortaya çıkarmıştır. Papert bu kavramı; öğrencinin merkezde olduğu ve deneyim haline getirebilmek adına “yaparak-yaşayarak” öğrenmesi şeklinde ifade etmiştir. İnşacılık yaklaşımı ile öğretilen programlama sürecine bakıldığında çok zengin, çok yönlü ancak sonuçları bakımından derin bir öğrenme ortaya çıkmaktadır (Ackermann, 2022; Papert ve Harel, 2022). Geleneksel yaklaşımlar ile

yenilikçi yaklaşımlar karşılaştırıldığında yenilikçi yaklaşımlar daha az çaba sarf edilerek daha az sorunla programlama öğretiminin verilmesinin avantaj olduğunu ifade etmişlerdir (Resinovic, 2015; Byrne ve Lyons, 2001; Uzun ve Uz, 2018).

Tüm bunların sonucunda eğitimde küçük yaştan itibaren programlamayı öğretmek amacıyla programlama dilleri, ortamları, platformları ve araçları ortaya çıkmıştır denilebilir (Numanoğlu ve Keser, 2017). Programlama öğretimi için pek çok farklı öğretim programı, dili ve aracı mevcuttur. Bunun nedeni farklı yaş grupları ve eğitim seviyeleri için çeşitli araçların kullanılması gerekliliğinin ortaya çıkarmasıdır. Bunun hakkında bir araştırma yapan Weinberg (2013), bilgi işlemsel düşünme becerisinin etkin öğrenilebilmesi için farklı yöntemler kullanılabileceğini ifade etmiştir. Bu yöntemler bilgisayarsız uygulamalar denen CS Unplugged yöntemi, blok tabanlı ve metin tabanlı programlama yöntemi, robot programlama yöntemi ve disiplinler arası öğretim yöntemleri ile hem bilgi işlemsel düşünme becerisinin kazandırılabilceğini hem de programlama öğretilabileceğini ifade etmiştir.

Bilgisayarsız Etkinlikler (CS Unplugged) ya da Kalelioğlu'nun (2020) ifade ettiği şekliyle B³ (Bilgisayarsız Bilgisayar Bilimi) olarak alan yazınında kullanılan bu yöntemde etkinlikler; bilgisayar kullanmadan bilgisayar bilimi öğretimini çevreden rahatlıkla bulunabilecek kâğıt, kalem gibi araç gereçler yardımı ile programlama mantığını vererek algoritma eğitimi yapmak anlamına gelmektedir (Bell, Alexander, Freeman ve Grimley, 2009; CSUnplugged, 2022; Aydoğdu, 2019; Kukul ve Karataş, 2016). Bilgisayar biliminin yapı taşı olan algoritma çok önemlidir (Schneider ve Gersting, 2018). Son yıllarda özellikle küçük yaş programlama eğitiminde sıklıkla öne çıkan bu yöntem hem uluslararası hem de ülkemizde kullanılmaktadır. Bunlara örnekler şu şekilde sıralanabilir; CSunplugged.org, CS4FN, Code.org unplugged, Bebras, CS@Minds, STEM Learning Computer Science Unplugged, Computer Science in a Box: Unplug Your Curriculum (NCWIT), Google for Education CS First, STAR Net Libraries STEM Clearinghouse – Computational Thinking Collection, Kodable, Keşf@ Projesi, Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme Etkinliği ve Tospaa.org. Yaptıkları çalışmada öğrencilerin bilgisayar bilimine ilgilerini arttırmak amacıyla Alice programlama ile etkinler yapmışlardır ve öğrencilerin derse karşı ilgilerinin olumlu olduğu sonucunda ulaşmışlardır (Mano, Allan ve Cooley, 2010). Yapmış oldukları B³ etkinlikleri sonucunda öğrencilerin bilgisayar bilimine olan ilgilerinin ve bilişsel becerilerinde artış gözlemlediklerini ifade etmişlerdir (Lambert

ve Guiffre, 2009). B³ programlama öğretim yönteminin, bilgisayar bilimine ve programlamaya karşı öğrencilerin ilgi ve motivasyonlarını arttırdığı söylenmektedir (Jiang ve Wong, 2018; Hermans ve Aivaoglou, 2017).

Blok Tabanlı Programlama, bilgisayarsız etkinliklere kıyasla okuma yazma bilen ancak yine yeni programlamaya başlayan öğrencilere yönelik bir yöntemdir. Daha renkli görseller, grafikler ve nesneler ile yap-boza benzeyen, sürükle bırak mantığı kullanılarak programlama öğretiminin bilgisayar kullanılarak öğretilmesini sağlayan bir yöntem olduğu söylenmektedir (Özel, 2019; Resnick, 2013). Blok tabanlı programlama ile yapılan çalışmalarda; öğrenciler tarafından görsel ara yüzünün renkli olmasından dolayı daha eğlenceli ve dikkat çekici bulunduğunu, ayrıca anlaşılabilirliğini daha yüksek bulduklarından tekrar kullanmayı tercih ettiklerini, problem çözme ve temel programlama becerilerini kazandırmada etkili olduğu sonucuna varılmıştır (Ersoy ve Aydın, 2015; Bishop- Clark, Courte ve Howard; 2007; Pakman, 2018; Aytekin, Çakır, Yücel ve Kulaöz, 2018). Blok tabanlı programlamalara örnekler şu şekilde sıralanabilir; Code.org, Scratch, M-block, PictoBlox, Kodu Game Lab, Makecode, AppInventor, Alice, Blockly Games-Eba. Blok tabanlı programlama sürükle bırak mantığıyla çalışmaktadır ve öğrenciler tarafından rahatlıkla kullanılabilir (Totan, 2021). Bu da öğrencilerin programlamaya olan tutumlarını olumlu etkilediğini ifade etmişlerdir (Çalışkan, 2019).

Metin Tabanlı Programlama, kelimelerin ve komutların kullanılması ile programlama dilinin söz dizim kurallarına uygun şekilde oluşturularak yapılan kodlama yöntemidir. Bir programlama diline ait dilbilgisi kuralları söz dizimi terimi ile ifade edilir. Bunu bir dilin dil bilgisi kurallarına benzeterek; farklarının da olduğunu ifade ederek bilgisayarın daha katı olan kurallarla bunu işlediği ifade edilmiştir (Özel, 2019). Blok tabanlı programlamaya göre daha geniş bir içeriğe sahip aynı zamanda karmaşık zihinsel süreçleri içerdiğinden birçok beceriyi geliştirdiği düşünülmektedir (Kandemir, 2018a). Karmaşık zihinsel süreçler içermesine karşın ortaokul seviyesindeki öğrencilerin metin tabanlı programlama dili Python öğrenebildiklerini ifade etmişlerdir (Park ve Yoo, 2018). Metin tabanlı programlamaya örnekler şu şekilde verilebilir; Small Basic, Python, C, C++, C#, Java, Php, Visual Basic, Deplhi, Perl. Programlama öğretimi yapılmadan önce, yöntem olarak hangi programlama dilinin seçileceği hedef kitleye uygun olup olmadığına göre değerlendirilmelidir. Bunlardan herhangi birini görmezden gelmek yapılacak olan kodlama işinin verimliliğini azaltacak ve öğretim sırasında birtakım sorunlara neden olacaktır.

Robot Programlama, son yıllarda özellikle bilgi işlemsel düşünme becerisini de geliştirdiği yapılan çalışmalarda görülmüş ve bir programlama yöntemi olarak uygulanmaktadır (Çetin ve Uçar, 2017; Berland ve Wilensky, 2015; Muñoz-Repiso ve Caballero-González, 2019; Durak, Yılmaz ve Yılmaz, 2019). Günümüzde öğrencilerin uygulama yaparak yazmış oldukları kod parçalarını somut olarak çalışabilirliğini gördükleri robot setlerine eğitsel robotlar ya da fiziksel programlama araçları isimleri verilmektedir (Üçgül, 2017; Weinberg, 2013). Robotların eğitsel amaçla kullanılması ilk defa Papert'in eğitimde robot kullanmasıyla başlamıştır denebilir (Papert, 1993; Çetinkaya, 2019; Uzun ve Uz, 2018). İnşacılık kuramını kullandığı Logo programı ile eğitsel robot programlamayı alanyazınına kazandırmıştır. Günümüzde pek çok oyun ve programlama odaklı robot kitleri üreten firmalar mevcuttur. Bu eğitsel fiziksel programlama araçları ya da diğer adıyla robot programlama araçları ile öğrenciler arasında iş birliği ve iletişimi sağlama, problem çözme, eleştirel düşünme becerilerini ve yaratıcılığı geliştiren bir öğrenme ortamının oluşturulabileceği ifade edilmiştir (Eguchi, 2015; Çetinkaya, 2019; Uzun ve Uz, 2018). Disiplinler arası olarak hazırlanmış öğretim programının öğrenciler açısından pek çok faydası vardır (Ramazanoğlu ve Toytok, 2018; Kurt, 2019); problem çözme becerisini geliştirir (Furner ve Kumar, 2007). Fiziksel programlama araçları öğrencilerin bilgisayar dersinde öğrenmiş oldukları programlama bilgileri ile hem uygulama yapma hem öğrendiklerini tekrar etme, farklı derslerde etkinlikler yaparak farklı bakış açıları geliştirebilmelerine ve eski öğrendikleri ile yeni öğrendiklerini pekiştirebilme imkânı bulabilecekleri ortamlar sunar (Freudenthal, Ogrey, Roy ve Siegel, 2010; Yıldırım ve Altun, 2015; Cortina, Dann, Frieze, Cinillo, Tananis ve Trahan, 2012). Bunların sonucunda öğrencilerin öğrendiklerini farklı ortam ve durumlara transferini kolaylaştırarak öğrencilerin öğrendiklerini kalıcı hale getirme fırsatı sağlar. Bu noktada bilgi işlemsel düşünme becerisi; bilgisayar, matematik ve fen dersleri başta olmak üzere disiplinler arası yaklaşımların kullanıldığı aynı zamanda proje tabanlı öğretim süreçlerini de destekleyen bir beceridir denilebilir (Gülbahar, Kalelioğlu, Doğan ve Karataş, 2020). Bilgi işlemsel düşünme sadece bilgisayar bilimi eğitiminde değil farklı disiplinlerde de kullanıldığında öğrenciler açısından başarı ve anlamlı olduğu görülmektedir. Disiplinler arası hazırlanan kazanımlar bilgisayar kullanımı ile öğretildiğin de öğrencilerin bağ kurarak öğrenmenin kalıcılığını arttırdığı görülmüştür (Zengin, 2016; Maden, 2021).

Programlama öğretimi ile ilgili yapılan araştırmalar, üst düzey bir beceri olması sebebiyle içerisinde pek çok başka kavram barındırmaktadır ve bu yüzden başka

kavramlarla ilişkilendirilmekte, araştırılmaktadır. Alanyazın incelendiğinde programlama çalışılırken en çok bilgi ve beceri başlıkları ile araştırılmıştır (Yurdakök ve Kalelioğlu, 2020). Araştırmalarda bilgi başlığında programlama ile ilgili tanımlar, algoritma süreci, programlama dili yapısı, hataları ve çözümlenmesi yer almaktadır. Beceri başlığı içerisinde, bilgi aşamasında edinilen veriler ışığında bir işlemi amaca uygun olarak hayata geçirme, sonuçlandırma yeteneğidir denmektedir (TDK, 2021; Caspersen, 2007). Araştırmacılar programlama becerisini bu iki başlık dışında pek çok kavramla da incelemişlerdir. Bu kavramlar; 21. yüzyıl becerileri, bilgi işlemsel düşünme, problem çözme becerisi, eleştirel düşünme, matematik başarısı, öz yeterlik algısı ve yaratıcılık. Programlama öğretiminde fiziksel araç kullanımının yaratıcı düşünceyi ve problem çözme becerisini geliştirirken öğrenmeyi eğlenceli hale getirerek öğrencilerin programlamaya yönelik olumlu tutum geliştirdiklerini ve motivasyonlarını artırdığını yapmış oldukları çalışmalarda ifade etmişlerdir (Karim ve ark., 2015; Resinovic, 2015; Yadagiri, Krishnamoorthy ve Kapila, 2015).

Araştırmalar sonucunda bilgi işlemsel düşünmeyi daha etkili verebilmek adına metin tabanlı programlama ve fiziksel programlama yaklaşımlarına dayalı bir öğretim programı tasarlanmış ve uygulanmıştır. Araştırma kapsamında fiziksel programlama aracı desteğiyle metin tabanlı programlama öğretiminde bilgi, beceri, bilgi işlemsel düşünme ve öz yeterlik algısı üzerinde durulmaktadır.

2.1.2. Programlama öğretiminde metin tabanlı programlama dilleri

Bilgisayarların, istenilen bir işlemi yapabilmelerini mümkün kılan yazılımlara ihtiyaçları vardır. Bu yazılımlar ise ancak programcılar tarafından belirlenmiş bir takım sembol ve kelimelerden oluşan komutları yazmayı sağlayan programlama dili ile geliştirilebilmektedir (Ersoy, Madran ve Gülbahar, 2011). Fortran ve Cobol gibi ilk bilgisayar programları ile başlayan ve günümüze kadar olan süreçte pek çok programlama dili ortaya çıkmış ve bazıları günümüzde artık kullanılmamaktadır. Her bir programlama dilinin kendine özgü yazım kuralları ve kalıpları vardır. Yazılan bu programlar bir işin nasıl yapılacağını bilinmediği durumlarda işin yapılması için gerekli donanımlara sahip olan bilgisayarlara, adım adım o işin nasıl yapılacağını anlatan kod parçalarıdır (Çağıltay ve Fal, 2014). Programlama yapılmadan önce sanki bir matematik problemi çözer gibi

problemi iyi anlamak yani analiz etmek önemlidir. Sonrasında süreci adım adım ayrıntılı bir şekilde tasarlamak yani algoritmasını oluşturmak gerekir.

Tüm dünyada ve ülkemizde ortaokul-lise düzeyinde müfredatlar da yer alan programlama ünitesi içerisinde temel düzey programlama kazanımları yer almaktadır. Programlamaya blok tabanlı programlama ile başlayan öğrenciler sonrasında metin tabanlı programlama ile karşılaşmaktadırlar. Yapılan çalışmalarda sürükle bırak yönteminden tamamen kendilerinin yazmaya başladığı sürece geçişte öğrencilerin zorlandıkları görülmektedir (Genç ve Karakuş, 2011; Ozoran, Çağıltay ve Toplalı, 2012). Alanyazında öğrencilerin yaşadıkları zorluklara karşın programlama eğitiminin özellikle de metin tabanlı programlama eğitiminin mantıksal düşünme becerilerinde oldukça katkı sağladığını gösteren çalışmalarda mevcuttur. (Mains, 1997; Clements ve Gullo, 1984; Papert, 1993; Swain, 2013; Sebetci ve Aksu, 2014).

Öğrenciler üzerinde blok tabanlı ve metin tabanlı programlama öğretiminin etkisini ölçmek istedikleri çalışmalarında; programlama öğretimi dersi almış öğrencilerin blok tabanlı programlama ilgisini çekerken, metin tabanlı programlama öğretimi alan öğrencilerin sürecine bakıldığında profesyonel bir programcının sürecine benzer bir süreç geçirerek tecrübe kazanmış olmalarından dolayı daha etkili bir süreç geçirmiş olduklarını belirlemişlerdir (Weintrop ve Wilensky, 2017). Türkiye’de, Avrupa ve Amerika’da olduğu gibi 1980’lerin başlarında ilk ve orta öğretimde bilgisayar kullanımına başlanmıştır (Akpınar ve Altun, 2014). Millî Eğitim Bakanlığı’nın 2018 yılında yayımlanan Bilgisayar Bilimi dersi öğretim programında ilk defa metin tabanlı programlama öğretimini almasıyla okullarda öğrencilere metin tabanlı programlama eğitimi verilmeye başlanmıştır. Metin tabanlı programlama dillerinin sayısı günümüzde teknoloji ve internetin gelişmesiyle beraber artmıştır. Ortaokul ve liselerde en çok kullanılan diller; Small Basic, Python, Java, Ruby, HTML, C, C++, C#, Php, SQL vb. gibi (Baştuğ, 2019; Alp, 2019; Özel, 2019). Bu araştırmada ortaokul öğrencilerine Python metin tabanlı programlama dili öğretilmiştir. Python programlama dili Hollanda’da 1980 yıllarda Guido Van Rossum tarafından geliştirilmiştir. Python programlama dili birçok programlama dilinden farklı olarak derlemeye gerek kalmadan çalıştırılabilir (Özel, 2019). Kolay öğrenilebilir olması, kolayca okunabilmesi sebebiyle okullarda küçük yaşlarda çok rahat öğretilbilmektedir (Aytekin, Çakır, Yücel ve Kulaöz, 2018; Baştuğ, 2019).

Python programlama dili birçok alanda kullanılmaktadır. Yaygın olarak kullanıldığı alanlar; masaüstü ara yüz uygulamaları, web uygulamaları, ağ programlama, bilimsel, oyun geliştirme, sayısal, yapay zekâ, akademik alanlarda, sistem yönetimi, veri analizi ve veri işleme, makine öğrenmesi, güvenlik ve veri tabanıdır. Python programlama diliyle yazılmış şirketler; Instagram, Netflix, Google, Dropbox, Spotify, Pinterest, Reddit'tir (Çelik, 2019). Metin tabanlı programlama öğretiminde, programlama dili seçimi çok önemlidir. Zor veya karmaşık sözdizimi (syntax) yapıları ve kurallarına dikkat edilmesi gerekir. Seçilen programlama dili öğrenci grubunun bilişsel gelişim sürecine uygun olduğu takdirde öğrenme kazanımlarına daha kolay ulaşılabilir (Kandemir, 2018b). Öğretmen ve öğrencilerin tercih etmelerinin bir sebebi de birçok proje için elverişli olmalarından dolayı hem ilk ve orta öğretimde hem de yükseköğretim programlarında metin tabanlı programlama dilleri kullanılmaktadır (Şendurur, 2018). Ortaokul 7., 8. ve 9. sınıfa giden öğrencilerle yapılan çalışmada, fiziksel programlama aracı kullanarak metin tabanlı programlama yaptırmışlardır ve sonucunda bilgi işlemsel düşünme becerilerine yönelik öz yeterlik algılarında olumlu etki gözlemlendiği belirtilmiştir (Feldhausen ve ark., 2018).

Birçok araştırmanın sonucuna bağlı olarak araştırmacılar bilgi işlemsel düşünmeye ait alt becerilerin gelişmesinde metin tabanlı programlamanın etkin olduğunu düşünmektedirler (Lee ve ark., 2011). Bu nedenden dolayı fiziksel programlama aracı micro:bit ile “Python ” metin tabanlı programlama dili kullanılmasına karar verilmiş ve çalışmada hazırlanmış olan programlama öğretiminde kullanılan etkinlikler, materyaller ve ölçme değerlendirme araçları buna göre seçilerek hazırlanmıştır.

2.1.3.Fiziksel programlama araçları

Alan yazınında fiziksel programlama ile ilgili farklı isimlerle karşılaşılmaktadır. Bazı kaynaklarda “Robot Tabanlı Programlama” ya da “Robot Programlama” olarak kullanılmaktadır. Fiziksel programlama araçlarının ilk örneği inşacılık yaklaşımında kullanılmaya başlanmıştır (Harel ve Papert, 1990). Fiziksel programlama araçları ile öğrenciler programlama yaparken sanal, iki boyutlu ekranlardan çıkarak etkileşimde daha fazla bulundukları, dokunabildikleri ve öğrenme sürecinde hareket edebildikleri, çalışmalarının sonucunu somut fiziksel ortamlara taşıyabilmektedirler (Üçgül, 2017; Richard, 2008; Banzi ve Shiloh, 2015;). Fiziksel programlama araçları günümüzde

teknolojinin de gelişmesi ile eğitim amaçlı üretilmiş robotik setleri ve mikroişlemcileri içine alan ucuzdan pahalıya çok geniş bir pazar alanına sahip öğretimi destekleyici bir unsurdur. (Green, Wagner ve Green, 2018). Ancak, bu eğitsel robotik setlerin kullanımı için gerekli olan karmaşık alan bilgisine, tasarım becerisine (Rubio ve ark., 2014; Junior ve ark., 2013), öğretimin yapılacağı alanın ayarlanması ve fiziksel programlama aracını temin etmeye bağlı sorunlar her öğretmen ve öğrencinin bu araçlara ulaşımına ulaşımını zorlaştırmakta olduğunu ifade etmişlerdir (Mozo ve ark., 2017; Peixoto ve ark., 2018).

Alanyazında bilgi işlemsel düşünme becerilerinin geliştirilmesi sürecinde metin tabanlı programlama dilleri ile kullanılan fiziksel programlama araçları kullanımı görülmektedir (Nouri, Zhang, Mannila ve Noren, 2019). Hubwieser ve arkadaşları (2015) tarafından 12 ülkede 14 eyalette; Almanya-Bavyera, Amerika-SGD, Almanya-NRW, Hindistan, Amerika-Gürcistan, Kore, Fransa, İsveç, İngiltere, Yeni Zelanda, Finlandiya, Amerika-İsrail, Rusya, İtalya; yapılmış olan K-12 seviyesinde bilgisayar bilimleri eğitimi incelenmiştir. Yapılan çalışma kapsamında ortaöğretim düzeyindeki kurumlarda 14 vaka çalışmasının sonuçlarına göre bilgisayar eğitiminde amaçlanan hedef, içerik ve yetkinlikler ele alınmıştır. Bunlara bağlı olarak öğretilen programlama dilleri ve araçları ile değerlendirme boyutları incelenmiştir. Ayrıca çalışmada ülkeler bazında tercih edilen programlama dilleri ve araçları da değerlendirilmiştir. Yapılan çalışmada görülmüştür ki ülkeler kendi dillerinde eğitsel programlama ortamları olan Scratch, Alice, LOGO, AgentCube, Kodu, Blockly, Robot Karol, Micro Worlds, Game Maker ve Squeak Etoys kullandıkları gibi profesyonel olarak kullandıkları programlama dilleri olan Python, Java, JavaScript, C++, C#, AppInventor, HTML, Basic, Visual Basic, Cobol, Php, Pascal, Fortran gibi dilleri de kullandıkları gözlemlenmiştir. Başka bir araştırmada programlama öğretiminde kullanılacak fiziksel programlama araçlarının öğrenci odaklı, düşük maliyetli, ek bir yazılım yüklenmesine ihtiyaç duymadan, öğrenme odaklı, eğitim amaçlı, problem çözmeye dayalı ve motive edici olabileceği belirtilmektedir.

Alanyazın incelendiğinde elde edilen sonuçlara göre; programlama öğretiminde metin / blok tabanlı programlama ortam ve araçları ile eğitim amaçlı fiziksel programlama araçları okullarda gün geçtikçe daha yaygın kullanım alanı bulmaya başladığı söylenebilir (Yurdakök ve Kalelioğlu, 2020). Kullanılan ortam ve araçların öğrencilerin programlama becerisini geliştirdiğini ve öğrenme sürecini daha aktif hale getirerek öğrenme sürecini daha etkili hale dönüştürdüğü, öğrencilerin derse katılımlarını arttırdığı buna bağlı olarak

da başarılarını arttırdığı, yaratıcı düşünme ve problem çözme becerilerini geliştirdiği ifade edilmektedir (Numanoğlu ve Keser, 2017). Przybylla ve Romeike (2014) çalışmalarında, fiziksel programlamanın öğrenciler üzerinde ve bilgisayar dersi kapsamında önemli katkıları olabileceğini ifade etmişlerdir. Öğrencilerin fiziksel programlama araçları kullandıkları öğretim sürecinde dikkatlerini daha uzun süre toplayarak çalışma heveslerinin arttığını ve fiziksel programlama aracına karşı ilgi duydukları söylenebilir (Barba ve Chanecellor, 2015; Sentance ve Schwiderski-Grosche, 2012; Türk ve Kuzu, 2019). Ancak, yapılan çalışmalarda sınırlı sayıda fiziksel programlama aracı ve platformu incelendiği görülmüştür.

Bu çalışmada programlama öğretiminde kullanılan micro:bit fiziksel programlama aracının ülkemizde çalışılmadığı, yabancı kaynaklı alanyazında ise çok az bilgiye ulaşılmıştır. Micro:bit programlama aracının eğitsel süreçlere getirdiği katkı düzeyinin, öğretilen hedef kitleye olan katkısının ve hangi programlama ile nihai amaçlara ne derece ulaşılabildiği gibi çalışmaların oldukça az olduğu görülmektedir. Örneğin; programlama başarısını, programlama becerisini, bilgi-işlemsel düşünme becerisini ne ölçüde arttırdığı, öğrencileri derse karşı ne ölçüde motive ettiğini veya derse katılımı ne ölçüde arttırdığına dair az çalışma bulunmaktadır. Bu nedenlerden dolayı bu çalışmanın amacı fiziksel programlama aracı destekli metin tabanlı programlama öğretimi ile ortaokul öğrencileri üzerinde bilgi işlemsel düşünme becerileri ve öz yeterlik algılarını artırmaya yönelik kullanılabilirliğini belirlemektir.

2.1.4. Fiziksel programlama aracının özellikleri

Programlama öğretimi hakkında alanyazında, yerli ve yabancı öğretim programları incelendiğinde özellikle son yıllarda farklı sınıf düzeylerinde farklı öğretim şekilleri ve yaklaşımları kullanıldığı görülmektedir. Bu programlama öğretimi genel pedagojik yaklaşımlara örnek olarak; bilgisayarsız etkinlikler, blok ve metin tabanlı programlama platformları, fiziksel programlama araçları ve oyuncakları verilebilmektedir (Falkner ve Vivian, 2015). Fiziksel programlama araçları kendi içinde farklı isimlere ayrılmaktadır. Micro:bit fiziksel programlama araçları içerisinde eğitimsel mikro denetleyicilere bir örnek teşkil etmektedir. İngiltere'de BBC'nin (British Broadcasting Corporation) 1980'lerde üretmiş olduğu BBC Micro bilgisayar projesinin günümüzün teknolojisine ve ihtiyaçlarına göre yeniden tasarlanarak üretilmiş hali micro:bit'tir ve isminin buradan geldiği söylenmektedir (Rogers ve ark., 2017). 2016 yılında 7. sınıfa giden tüm öğrencilere (12-13

yaş arası yaklaşık bir milyon) BBC'nin desteği ile geliştirilmiş mikro: bit mikro denetleyici kartı programlama yapmaları amacıyla ücretsiz olarak dağıtılmıştır (İzgöl, 2017; Andrews, 2016). Bunun sebebi öğrencilerin kredi kartı boyutundaki bir programlama aracı ile kolaylıkla bilgisayara bağlayarak program yazmalarını sağlamaktır (Austin ve ark., 2020). Aracın en büyük özellikleri kolay bir şekilde bilgisayara bağlayabilmek, hızlı ve basit bir başlatma yapılabilmesi sebebiyle hem öğrenci hem de öğretmenler açısından öğrenimi- öğretimi kolay ve motive edici bir fiziksel programlama aracı olmasıdır (Schmidt, 2016; Tyren ve ark., 2018).

Fiziksel programlama aracı micro:bit üzerinde ön bölümde sırasıyla yer alan bileşenler; düğmeler, LED ekran ve ışık sensörü, pinler-GPIO, pim-3 volt güç, pim-topraklama, dokunmatik logo, mikrofon LED'i bulunmaktadır (Şekil 2.1). Bu çalışmada ilk micro: bit tasarımı ile programlama öğretimi yürütülmüştür.

Şekil 2.1. Micro: bit micro denetleyici kart ve micro: bit Python editörü ekran görüntüsü.



Micro:bit üzerinde yer alan arka bölüm bileşenleri; radyo ve bluetooth anteni, işlemci ve sıcaklık sensörü, pusula, ivmeölçer, pimler, mikro USB soketi, sıfırlama düğmesi, pil yuvası ve USB arabirim yongası bulunmaktadır. Micro:bit kartı ile microbit.org/code/ adresi kullanılarak internete bağlanabilen herhangi bir bilgisayar ile tarayıcı üzerinden kolaylıkla giriş yapılarak kod yazmaya başlanabilmektedir. Bir program yazıldıktan sonra, çevrimiçi olarak derlenebilir ve daha sonra web sitesinden indirilebilir. Gerçek donanıma dağıtmak için, BBC micro:bit USB bağlantı noktasına bağlanır. USB bellek gibi davranan micro:bit'e program yüklenir ve derlenen program sadece sürücüye kopyalanır. BBC micro:bit' in sıfırlanmasından sonra, saklanan son dosya otomatik olarak yürütülür. Kablosuz olarak da micro:bit kullanılabilir (Ball ve ark., 2016) . Bluetooth sayesinde bir kablo bağlantısına ihtiyaç duymadan micro:bit'e yazılan programlar yönlendirilmektedir. Micro:bit programlarının uzantısı “.hex” şeklindedir.

Web sitesi üzerinden farklı programlama dillerini (MakeCode, JavaScript ve Python) destekleyen düzenleyiciler mevcuttur. Web sitesi çeşitli öğreticiler ve örnekler sunmaktadır. Micro:bit ile kodlama yaparken sadece bilgisayar değil aynı zamanda hem android hem de IOS kullanan tablet ve telefonlar için de uygulamalar mevcuttur. Bunların dışında BBC micro:bit'e bağlı olmayan editörlerde vardır. Bunlar; Edu Blocks (Python), Mu (Python), AppInventor, Tinkercad, Arduino IDE (C++), Art:bit, Bitty Software App, CodeMao, Kittenblock, Kodu Game, MicroBlocks, Mind+, mBlock 5, TI Graphing Calculators (Python) (Micro:bit, 2022). Bu çalışmada kullanılan micro:bit Python düzenleyici (<http://python.microbit.org>) İngilizce dilindedir. Düzenleyicinin ana menüsü, düzenleyici ve micro:bit aygıtıyla etkileşim ve kolay kullanımı için çeşitli düğmeler içerir. Metin düzenleyici, programın anlaşılabilirliği için farklı bölümler farklı renklerden oluşmaktadır. Örneğin, Python anahtar sözcükleri gridir. Daha parlak renkli kelimeler, oluşturduğunuz programın parçalarıdır. Kahverengi kelimeler asla değişmeyen sabit değerlerdir ve mor kelimeler görüntülenecek karakter dizilerini temsil eder. Tüm satırlar, geçerli satır vurgulanarak numaralandırılmıştır (Şekil 2.1).

Yapılan çalışmalar fiziksel programlama araçlarının derslerde kullanılmasıyla öğrenme süreçlerinin iyileştiği ve öğrencilerin teknoloji ve tasarım yöntemleri hakkında daha fazla bilgi edindiğini ortaya koymuştur (Radu, Cole, Dabacan, Harris ve Sexton, 2011; Jin, Haynie ve Kearns, 2016; Cápay, Kvaššayová, Bellayová, Mansell ve Petrik, 2022; British Council, 2022). Bekker ve arkadaşları (2015) yılında yapmış oldukları çalışmada fiziksel programlama araçlarının öğrencilerin problem çözme becerilerini, yaratıcılıklarını ve eleştirel düşünme becerilerini geliştirdiği ve çocukların STEM konularına ilgilerini artırırken bu alanlarda mesleki ilgilerinin de arttığını belirtmişlerdir. BBC Micro:bit, öğrencilerin programlama ve fiziksel programlama aracının kullanımı hakkında karmaşık ön bilgiye ihtiyaç duymadan, programlamanın etkili bir şekilde öğrenilmesini sağlamak için oldukça basit bir şekilde tasarlanmıştır. Ekstra donanımlar eklemeye gerek kalmadan yerleşik sensörler aracılığıyla pek çok programlama yapılabilmektedir (Bernad, Sic, Repnik ve Osrajnik, 2021). BBC Micro:bit aracı ile yapılabilecekler öğrencilerin yaratıcılığına ve programlama becerilerine bağlı olduğunu; buna ek olarak öğrencilerin aracın üzerinde bulunan bileşenlere ne kadar hâkim olurlarsa o kadar çok şey yaratabileceklerini ifade etmektedirler (Kalelioğlu ve Sentance, 2020). Micro:bit ile basit ara yüz ve kullanım kolaylığı ile algoritma, döngü, rastgelelik, mantık,

değişkenler ve hata ayıklama gibi programlama dili kavramlarının öğretilmesi kolay bir şekilde gerçekleştiği söylenmektedir (Tyren, Carlborg, Health ve Erkişon, 2018).

Micro: bit ile programlama öğretiminin nihai sonucunun, Scratch gibi blok tabanlı programlama ortamından Python metin tabanlı programlama gibi daha yüksek seviyeli dil programlamayı fiziksel programlama ortamlarına transferini sağlayarak kolaylaştırmak ve kodlamayı öğrenmede eğitim sürecini yenilemektir (Videnovik, Zdravevski, Lameski ve Trajkovik, 2018). Çalışmalarında öğrencilerin her yerde basit bir şekilde bilgi işlem uygulamaları oluşturmaya olanak tanıdığını ifade etmişlerdir (Rogers ve ark., 2017). Videnovik ve arkadaşları (2018) yaptıkları çalışmada cihazlara ve sensörlere bağlanabilme imkânının kolay olması, öğrencilerin çalışmalarının sonuçlarını görebilmeleri ve dokunabilmeleri nedeniyle programlama öğrenimindeki zorlukları azalttığını hatta kodlama öğrenmeye ilgilerinin arttığını belirtmişlerdir. Lancaster Üniversitesi Eğitim Araştırmaları Departmanı tarafından, Birleşik Krallık'ta 40 ortaokulda (11-14 yaş) yapılan araştırmada yapılan analiz sonucunda, micro:bit' in en sık ortaokul düzeyinde kullanıldığı ortaya çıkmıştır. Ayrıca micro:bit fiziksel programlama aracı ile en çok grup çalışması yapıldığı, çalışılan disiplin olarak da bilgisayar, teknoloji ve fen derslerinde proje çalışmasında kullanıldığı araştırma sonuçlarında ortaya çıkmıştır (Knowles, Finney, Beck ve Devine, 2018). İngiltere'de yaptıkları çalışma ile BBC micro:bit'in kullanılabilirlik, yaratıcılık, cihazın somutluğu ve öğrencilerin programlama öğrenmeleri açısından deneyimlerinde olumlu veriler elde etmişlerdir (Sentance, Waite, Hodges, MacLeod ve Yeomans, 2017). Kuzey İrlanda'da öğrencilerin BBC micro:bit aracını kullanmaya yönelik algılarının araştırıldığı çalışmada; %64'ü kullanımının kolay/çok kolay olduğunu, %90'ı problem çözmekte faydalı olduğunu ve %90'ı eğlenceli/çok eğlenceli olduğunu ifade etmişlerdir (Gibson ve Bradley, 2017).

Tüm bu araştırmalar sonucunda, ortaokul öğrencilerine programlama öğretiminde fiziksel programlama aracının önemi büyüktür. Alanyazınına göre fiziksel programlama araçlarından biri olan İngiltere'de geliştirilmiş BBC Micro:bit' in araç olarak kullanılabilirliği, programlama öğretimi açısından teşvik edici ve olumlu sonuçlarından dolayı bu çalışmada fiziksel programlama aracı olarak BBC micro:bit kullanılmasına karar verilmiştir. Araştırmalar sonucunda ortaokul öğrencilerinin programlama öğretimi ile bilgi işlemsel düşünme becerilerini geliştirilmesi adına metin tabanlı Python programlama dili, programlama öğretiminin daha aktif ve etkili olması adına fiziksel programlama

araçlarından micro:bit ile öğretim programı tasarlanmış ve uygulanmıştır. Araştırma kapsamında fiziksel programlama aracı desteğiyle metin tabanlı programlama öğretiminde bilgi, beceri, bilgi işlemsel düşünme ve öz yeterlik algısı üzerinde durulmaktadır. Bu nedenden dolayı fiziksel programlama aracı micro:bit ile “Python” metin tabanlı programlama dili kullanılmasına karar verilmiş ve çalışmada hazırlanmış olan programlama öğretiminde kullanılan etkinlikler, materyaller ve ölçme değerlendirme araçları buna göre seçilerek hazırlanmıştır.



BÖLÜM III

YÖNTEM

3.1. Araştırma Modeli

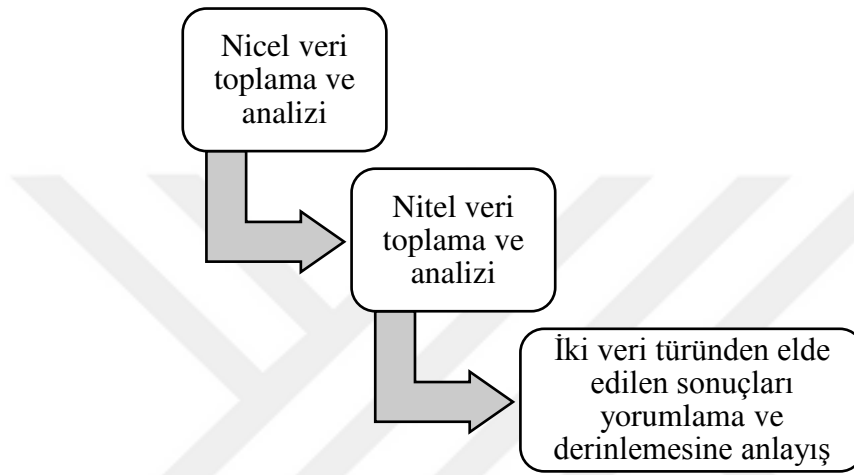
Bu çalışmada hem nicel hem de nitel araştırma verilerinin kullanılması sebebiyle karma yöntem kullanılmıştır. Karma yöntem araştırması, araştırmacıların tek bir çalışmada nitel ve nicel yaklaşımları, yöntemleri kullanarak veri topladığı, bulguları analiz ettiği ve en sonda çıkarımlarda bulunduğu bir yöntem olarak ifade edilmektedir (Tashakkori ve Creswell, 2007; Creswell ve Plano Clark, 2014; Leech ve Onwuegbuzie, 2009; Alkan, Şimşek ve Armağan Erbil, 2019). Karma yöntem araştırmalarının temel özelliklerini Creswell şöyle sıralamıştır (2021); nicel ve nitel verileri toplayıp analiz ederek araştırma sorularını cevaplandırmak; nicel ve nitel yöntemlerin ayrı ayrı güçlü yanlarının kullanılması; bazı durumlarda araştırmayı bir kuram veya felsefe içinde ele alma; nitel ve nicel verileri özel bir karma yöntem deseni içerisinde kullanarak verileri birleştirerek bütünleştirmek. Karma yöntemle hazırlanacak bir araştırmanın aşamaları Johnson ve Onwuegbuzie (2004) tarafından aşağıdaki şekilde sıralanmıştır:

1. Araştırma problemine karar vermek,
2. Araştırmanın karma desene uygun olup olmadığına karar vermek,
3. Karma yöntem veya karma model araştırma desenini seçmek,
4. Verileri toplamak,
5. Verileri analiz etmek,
6. Verileri yorumlamak,
7. Sonuçları ve sonuç raporunu yazmaktır.

Bu temel sınıflamaya göre karma desen yönteminin türleri; açıklayıcı, dönüşen, gömülü, eşzamanlı, açıklayıcı ve çok aşamalı şeklinde ifade edilmektedir (Creswell, 2015). Bu çalışmada karma araştırma yönteminin türlerinden sıralı açıklayıcı desen kullanılmıştır. Creswell ve Plano-Clark (2014) sıralı açıklayıcı deseni şu şekilde açıklamışlardır; desenin iki aşamalı bir süreç olduğunu, öncelikle nicel verilerin toplandığını ve analiz edildiğini ifade etmişlerdir. Sonrasında ikinci aşama olarak nicel verileri açıklamak ve ayrıntılı olarak incelemek için nitel verilerin toplanarak analiz edilmektedir. Bu desen, araştırmacının nicel çalışmasında beklenmedik ya da şaşırtıcı sonuçlar ile özellikle ortaya çıkan önemli detayları destekleyici nitel verilerle açıklamak istediği durumlarda kullanmanın çok

avantajlı olduğunu söylemektedirler (Creswell, 2021; Creswell ve Plano-Clark, 2014). Sıralı açıklayıcı desende temel olan çalışmanın nicel bulgularıdır. Nitel bulgular ise ikincil rol oynar ve nicel veriler nitel verileri anlamlandırarak desteklemek için kullanılır (Büyüköztürk,2007). Sıralı açıklayıcı desenin veri ve analiz şeması Şekil 3.1’de verilmektedir.

Şekil 3.1. Sıralı açıklayıcı desenin veri ve analiz şeması (Toraman, 2021).



Bu çalışmanın nicel araştırma kısmında tek grup ön test – son test araştırma deseni kullanılmıştır. Bu desenin kullanılma amacı; belirli bir gruba öncelikle ön test araştırma ölçümü yapılır, sonrasında çalışmanın kapsamına uygun süreç uygulanır ve ardından son test uygulanmasıdır (Creswell, 2009). Ön test içerisinde; Bilgi İşlemsel Düşünme Becerilerine Yönelik Öz Yeterlik Algısı Ölçeği (BİDBÖA) ve araştırmacı tarafından 2016-2020 tarihleri arasından seçilmiş Bilge Kunduz-Uluslararası Enformatik ve Bilgi İşlemsel Düşünme Etkinliği görevleri bulunmaktadır. Öğretim süreci sırasında araştırmacı tarafından hazırlanmış her ders sonu değerlendirme soruları ve ders süreci gözlem formu bulunmaktadır. Süreç sonunda ise son test olarak; Bilgi İşlemsel Düşünme Becerilerine Yönelik Öz Yeterlik Algısı Ölçeği (BİDBÖA), araştırmacı tarafından 2016-2020 tarihleri arasından seçilmiş Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme Etkinliği görevleri, araştırmacı tarafından hazırlanmış 25 soruluk çoktan seçmeli programlama aracı, kavram ve süreçlerine yönelik kazanım testi ve odak grup görüşmesi bulunmaktadır. Çalışmanın yönteminin başarısına ilişkin öğrencilerin kazanım testleri sonuçları değerlendirildikten sonra uygulamaya ilişkin en çok beğenilen yönlerin öğrencilerle görüşmeler yapılarak ortaya konmaya çalışılmıştır.

3.2. Çalışma Grubu

Araştırmanın çalışma grubunu, Ankara ilinde özel bir okulun 7. ve 8. sınıflarında okuyan ve seçmeli bilişim teknolojileri ve yazılım dersi alan toplam 85 öğrenci oluşturmaktadır. Çalışmaya katılan öğrenciler 12-13 yaşlarındadır. Tablo 3.1’de cinsiyet ve sınıflarına göre öğrencilerin 46’sını (% 54.11’ni) 7. sınıf öğrencileri, 39’nu (% 45.88’ni) 8. sınıf öğrencileri oluşturmaktadır. Buna ek olarak öğrencilerin 13’nü (% 15.30’nu) kız öğrenciler, 72’sini (% 84.70’ni) erkek öğrenciler oluşturmaktadır. Cinsiyete bağlı dağılımın, seçmeli ders olması sebebi ile erkek öğrencilerin daha çok tercih etmelerinden kaynaklı olarak sayıları kız öğrencilere oranla fazladır. Bu sebeple araştırma kapsamında cinsiyete bağlı bir değerlendirme yapılmamıştır. Öğrenciler, 5. sınıfta metin tabanlı programlama olarak small basic programlama dili öğrenmiş; sonrasında da herhangi bir programlama dili öğrenmemiştir.

Tablo 3.1. Cinsiyet ve sınıflarına göre öğrencilerin frekans ve yüzde dağılımı.

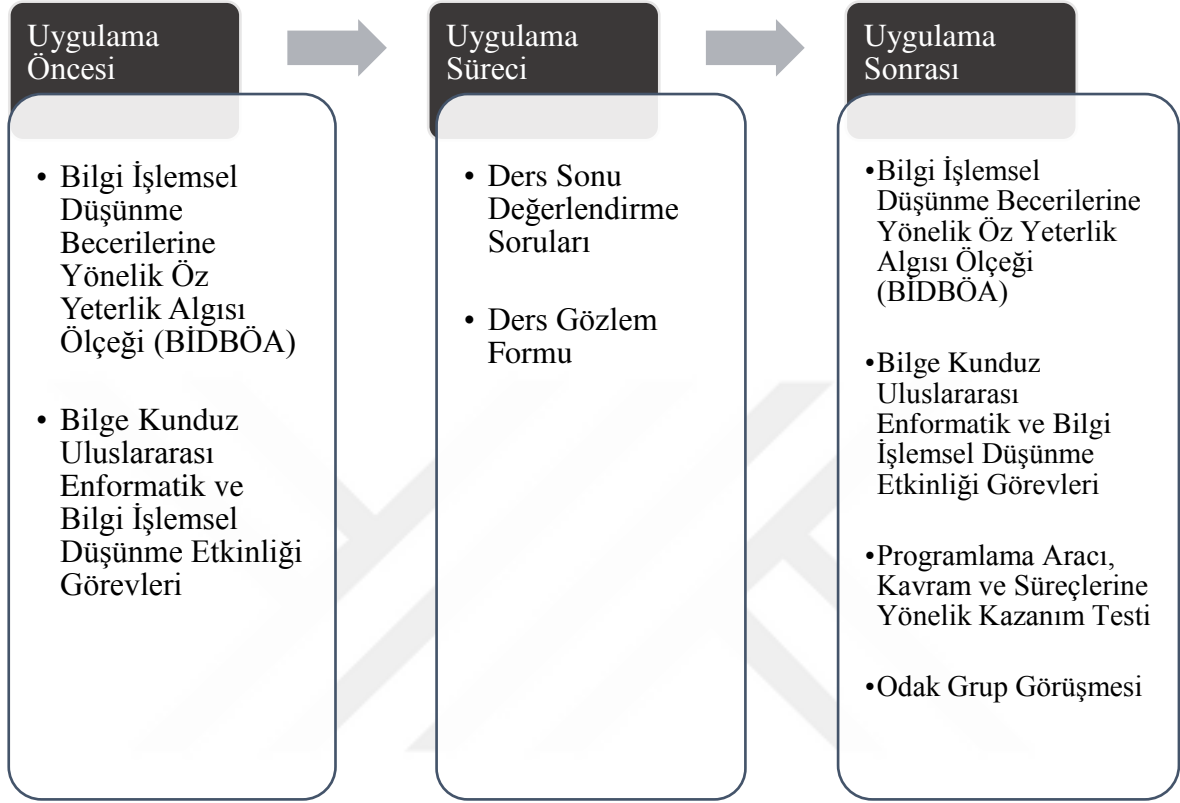
	Kız öğrenci sayısı (n)	Kız öğrenci sayısı (%)	Erkek öğrenci sayısı (n)	Erkek öğrenci sayısı (%)	f	%
7. Sınıf	9	19.57	37	80.43	46	54.11
8. Sınıf	4	10.25	35	89.75	39	45.88
Toplam	13	15.30	72	84.70	85	100

3.3. Verilerin Toplanması

Araştırma sorularına yanıt vermek için üç aşamada toplanan veriler ele alınmış ve değerlendirilmiştir. Alanyazında bilgi işlemsel düşünmenin nasıl ölçüleceği hakkında tam bir tutarlı görüş bulunmamaktadır (Gonzalez, 2015; Demir ve Seferoğlu, 2017). Ancak buna karşın çeşitli ölçme yaklaşımları mevcuttur. Bunlardan bazıları; ölçek, çoktan seçmeli test, performans değerlendirme, dereceli puanlama anahtarlarına göre değerlendirme, ürün değerlendirme, portfolyo değerlendirme ve görüşme vb. Alan yazınına bakıldığında bilgi işlemsel düşünme konusunda çalışma yapacak araştırmacılara, çalışmaların olumlu yönde başarıya ulaşabilmesi adına önerilerde bulunmuş araştırmacılar dikkat çekmektedir. Bu araştırmacılar, yapılacak çalışma sürecini mümkün olduğunca uzun planlanması gerektiğini, ölçme-değerlendirme için sadece sonucun ya da sürecin değil ikisinin beraber değerlendirmeye katılması gerektiğini, bunu yaparken de nitel ve nicel verilerin beraber toplanarak değerlendirilmesi ve mümkün olduğunca birden fazla değerlendirme yapılması gerektiğini tavsiye etmişlerdir (Buss ve Gamboa, 2017; Atmatzidou ve Demetriadis, 2016).

Uygulama öncesi, süreci ve sonrası veri toplama aşamaları aşağıda sırasıyla yer almaktadır (Şekil 3.2).

Şekil 3.2. Araştırmanın uygulama süreci.



1. Uygulama Öncesi: Uygulama öncesi öğrencilere “Bilgi İşlemsel Düşünme Becerilerine Yönelik Öz Yeterlik Algısı Ölçeği” ve “Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme Etkinliği Görevleri” ön test olarak uygulanmıştır.

2. Uygulama Sürecinde: Uygulama sürecinde kazanımlar göz önünde bulundurularak, araştırmacı tarafından hazırlanmış, ilgili haftanın kazanımlarını içeren “Ders Sonu Kendimi Değerlendiriyorum” sorularını öğrenciler cevaplamışlardır. Ayrıca araştırmacı tarafından öğrencilerin ders sırasında bilişsel ve psikomotor becerilerindeki gelişimlerini ölçmek ve ders takibini yapabilmek için gözlem formu hazırlanmış ve her ders sonunda doldurulmuştur.

3. Uygulama Sonrası: Uygulamanın sonunda çalışma öncesinde yapılmış olan ön testler aynı şekilde son test olarak da uygulanmıştır. “Bilgi İşlemsel Düşünme Becerilerine Yönelik Öz Yeterlik Algısı Ölçeği” ve “Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme Etkinliği Görevleri” ne ek olarak, araştırmacı tarafından hazırlanmış

programlama öğretiminde fiziksel programlama aracı ve metin tabanlı programlama dili ile ilgili araştırma sürecinde öğrencilere kazandırılması hedeflenen kazanımları etkililiği hakkında 25 soruluk çoktan seçmeli kazanım testi ve çalışma hakkında odak grup görüşmesi yapılmıştır.

3.3.1. Bilgi işlemsel düşünme becerilerine yönelik öz yeterlik algısı ölçeği (BİDBÖA)

Bilgi İşlemsel Düşünme Becerilerine Yönelik Öz Yeterlik Algısı Ölçeği (BİDBÖA) 2018 yılında Gülbahar, Kert ve Kalelioğlu tarafından geliştirilmiştir. Gülbahar, Kert ve Kalelioğlu (2018), çalışmalarında bilgi işlemsel düşünme becerilerine ilişkin kişisel algılarını ortaya çıkararak kendilerini tanımları amacıyla bu ölçeği geliştirdiklerini ifade etmişlerdir. Ölçek geliştirilirken yapılan araştırmalarda bu becerinin beş alt boyuttan oluştuğunu belirlemişlerdir. Ölçek formu, 5 alt faktöre ayrılmış, toplam 36 maddeden oluşmaktadır. Bu beş tane alt boyut sırasıyla şöyle ifade edilmiştir; problem çözme yeterliği, algoritma tasarlama yeterliği, temel programlama yeterliği, veri işleme yeterliği ve özgüven yeterliği. Bilgi işlemsel düşünme becerisi soyut bir beceridir ve tek bir ölçme aracının sonucuna bağlı olarak bir karara varmanın doğru sonuçlar vermeyeceği ifade edilmiştir. Bundan dolayı farklı değerlendirme araçlarının bir arada kullanılarak desteklenmesi gerekliliğini ifade etmişlerdir (Gülbahar, Kert ve Kalelioğlu, 2018). Bu nedenle araştırmada öz yeterlik ölçeği ile Bilge Kunduz görevlerine de yer verilmiştir. Ölçeğin kullanımına ilişkin gerekli izinler alınmıştır. Ölçek, ortaokul düzeyinde, öğrenciler tarafından daha rahat karar verilmesine destek olmak amacıyla, 3'lü likert yapıdadır.

Bilgi İşlemsel Düşünme Becerilerine Yönelik Öz Değerlendirme Yeterlik Algısı ölçeğinin toplam güvenilirlik katsayısı Cronbach Alpha analizinde 0.943 olarak ortaya çıkmıştır. Alt faktörlere ilişkin güvenilirlik katsayısı Tablo.3.2'de sunulmuştur.

Tablo 3.2. BİDBÖA güvenilirlik katsayıları.

Alt faktör	Madde sayısı	Güvenirlik katsayısı
Algoritma tasarlama yeterliliği	9	0.930
Problem çözme yeterliliği	10	0.880
Veri işleme yeterliliği	7	0.856
Temel programlama yeterliği	5	0.838
Özgüven yeterliği	5	0.762
Tüm ölçek	36	0.943

3.3.2. Bilge kunduz uluslararası enformatik ve bilgi işlemsel düşünme etkinliği görevleri

Bilge Kunduz (Bebras), uluslararası düzeyde bir etkinlik olup öğrencilerin hem bilgi işlemsel düşünme becerilerini geliştirirken hem de her yaş seviyesindeki öğrenciye bilgisayar bilimini öğretmeyi hedeflemektedir. Türkiye 5. ve 6. sınıflar ile uygulamaya başlanan etkinliğe artık ilköğretim kademesinde 3., 4. sınıflar, 5., 6. sınıflar ve 7., 8. sınıflar ile ortaöğretim kademesinde 9., 10. sınıflar ve 11.ve 12. sınıflar olmak üzere 5 farklı seviyede katılmaktadır (Bilge Kunduz, 2022). Bilge Kunduz etkinliğinde yer alan kısa sorular “Bilge Kunduz Görevleri” ismiyle kullanılmaktadır. Bilge Kunduz görevlerini çözebilmek için öğrencilerin enformatik konusunda ön bilgiye sahip olması gerekmemektedir. Bu görevleri çözebilmeleri için öğrencilerin bildiklerini gözden geçirmesi, hesaplama yapması, neden-sonuç ilişkisi kurması, karar vermesi, analitik düşünme ve problem çözme gibi üst düzey düşünme becerilerini kullanması beklenmektedir (Gülbahar, Kalelioğlu, Doğan ve Karataş, 2020).

Çalışmanın öğretim süreci sırasında, programlama, programlama aracı kullanımı, algoritma analizi ve hata ayıklama gibi kazanımları içeren Bilge Kunduz görevleri, araştırmacı tarafından 2016- 2020 yılları 7. ve 8. sınıf için sınıflandırılmış Türkçe sorular arasından incelenerek seçilmiştir. Bilge Kunduz görevlerinin içerik ve kavramsal anlamda uzman görüşü alınarak değerlendirilmesi yapılmış ve nihai sorulara karar verilmiştir. Bilge kunduz görevleri uzman değerlendirmeleri hakkında ve bilge kunduz görevleri sınıflandırmaları Tablo 3.3’te yer almaktadır.

Tablo 3.3. Bilge kunduz görevleri uzman değerlendirmesi.

Soru no	Uzman çoğunluk	Puan	$\bar{X}$	Uzman sınıflandırması	Bilge kunduz sınıflandırması
1	5	10	1.4	Kolay	Kolay
2	4	10	1.4	Kolay	Zor
3	4	18	2.4	Zor	Zor
4	4	11	1.7	Orta	Kolay
5	3	14	2	Zor	Zor
6	4	12	1.7	Orta	Orta
7	3+3	15	2.1	Orta + Zor	Zor
8	7	7	1	Kolay	Kolay
9	5	16	2.2	Orta	Zor
10	4	15	2.1	Orta	Orta
11	6	19	2.7	Zor	Zor
12	5	10	1.4	Kolay	Zor
13	3+3	16	2.2	Orta + Zor	Kolay
14	4	17	2.4	Orta	Orta
15	5	9	1.2	Kolay	Orta

Tablo 3.3.'de görüldüğü gibi Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme görevlerinin seçiminde uzman değerlendirmesi sonucunda görevlerin düzeyleri belirlenmiştir. Buna göre düzeyler Bilge Kunduz düzeylerine uygun olarak dağılımı yapılmıştır. Bilge Kunduz görevlerinin dağılımında 5 kolay düzeyde görev, 5 orta düzeyde görev ve 5 zor düzeyde görev yer almıştır. Puanlama şekli kolay sorular için doğru cevapta 6 puan, orta düzey soruların cevaplamasında doğru cevap 9 puan ve zor soruların puanlanması ise doğru cevap için 12 puan olarak hesaplanmıştır. Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme görevlerinden alınabilecek en yüksek puan 135'tir. Hem ön test hem de yapılan son testte öğrencilerin hiçbirinin yanıtsız soru bırakmadığına dikkat edilmiştir. Her iki testte de öğrencilere 35'er dakika verilmiştir. Seçilen görevlerin isimleri, yıllara göre dağılımı, zorluk düzeyleri ve kazanımları Tablo 3.4.'de yer almaktadır.

Tablo 3.4. Yıllara göre seçilen görevlerin isimleri, zorluk düzeyleri ve kazanımları.

Soru yılı	Soru no	Sorunun adı	Sorunun zorluk düzeyi	Sorunun kavramları / işlemsel düşünme ile ilişkisi	enformatik / bilgi ile
2020	1. Soru	Robot Kol	Kolay	Zamanlama, Programlama, Hata ayıklama	
	2. Soru	Yanıp Sönen Işıklar	Kolay	Programlama, Sayısal Elektronik, Hata ayıklama, Arduino	
	3. Soru	Kunduzlar Kangurulara Karşı	Zor	Algoritma	
2019	4. Soru	Sihirbaz Melek	Orta	Kodlama, Veri transferi	
	5. Soru	Ayakkabı Boyutu	Zor	Algoritma, Mantıksal akıl yürütme	
	6. Soru	Dijital Sayı	Orta	Kodlama	
	7. Soru	Kereste Fabrikası	Zor	Algoritma, Akış, Programlama	
2018	8. Soru	Bir Uzaylının Mutasyonu	Kolay	Programlama, Fonksiyon, Değişkenler	
	9. Soru	Tıbbi Laboratuvar	Orta	Algoritma	
	10. Soru	Robo-Bahçıvan	Orta	Algoritma, Döngü	
2017	11. Soru	Karma Fonksiyon	Zor	Algoritma, Dizi	
	12. Soru	Labirentten Kaçış	Kolay	Programlama, Döngü, Algoritma	
2016	13. Soru	Arabalar	Zor	Algoritma, koşullar	
	14. Soru	Pul Koleksiyonu	Orta	Hata ayıklama	
	15. Soru	Çılgın Robotlar	Kolay	Algoritma, Programlama	

Çalışmada ön test ve son test olarak kullanılan Bilge Kunduz görevleri araştırmacı tarafından seçilmiş 15 soru olarak öğrencilere uygulanmıştır. Bilge Kunduz görevlerinin enformatik alanında ölçmeyi hedeflediği kavramlar listesi Tablo 3.5.'de yer almaktadır.

Tablo 3.5. Seçilen görevlerin enformatik anlamda ölçtüğü kavramların görevler bazında listesi

Sorunun Kavramları / Bilgi İşlemsel Düşünme ile İlişkisi	2020			2019			2018			2017			2016		
	1. S	2. S	3. S	4. S	5. S	6. S	7. S	8. S	9. S	10. .S	11. .S	12. .S	13. .S	14. .S	15. .S
Zamanlama	✓														
Hata ayıklama	✓	✓												✓	
Programlama	✓	✓		✓		✓		✓				✓			✓
Arduino		✓													
Sayısal elektronik		✓													
Algoritma			✓		✓		✓		✓	✓	✓	✓	✓		✓
Veri transferi				✓											
Mantıksal akıl yürütme					✓										
Akış							✓								
Fonksiyonlar											✓				
Değişkenler								✓							
Döngü										✓		✓			
Dizi											✓				
Koşullar														✓	
Veri yapıları											✓				

Tablo 3.5.'de görüldüğü gibi seçilen görevlerin ölçtüğü kavramlar algoritma ve programlamaya ait kazanımlara uygun sorulardır ve bazı sorular birden fazla farklı kavramsal düzeyi ölçmeye yöneliktir. Uygulama öncesi (ön test) ve sonrası (son test) kullanılacak olan bu görevler, algoritma ve programlama kavramlarının bilgi düzeyinde farklılıkları tespit etmeye yönelik seçilmiştir. Bilge Kunduz görevleri seçimi yapılırken 21. yüzyıl becerileri kapsamında öğrencilerin okuduklarını anlamaları ve analiz edebilmeleri gibi yeterlikleri ölçebilecek sorular içermesi hedeflenmiştir. Ayrıca öğrencilerin programlama mantığını kavrama yeterliklerini, bir problemi analiz ederek ve hata ayıklayabilecek düzeyde bilgiye sahip olduklarını ortaya çıkaracak görevler seçilmeye çalışılmıştır.

Bilge Kunduz görevlerinden seçilen 15 görevin 9 tanesinde algoritma oluşturma kavramsal düzeyi bulunmaktadır. Bu etkinlikte temel kazandırılması istenilen davranış, algoritma yazma becerisi ölçülmek istendiğinden, görevlerin çoğunda bu kazanım yer almıştır. Diğer kazanımlar incelendiğinde; programlamada zaman, hata ayıklama, programlama, arduino, sayısal elektronik, veri transferi, mantıksal akıl yürütme, akış,

fonksiyonlar, deęiřkenler, dng kullanımı, dizi, kořullar ve veri yapıları olduęu grlmektedir. Arařtırmada kullanılan Bilge Kunduz Uluslararası Enformatik ve Bilgi İřlemsel Dřnme Etkinlięi grevlerinin ierięini oluřturan kazanımlar aynı zamanda fiziksel programlama aracı ve metin tabanlı programlama dili Python ęretiminde belirlenen kazanımlar ile benzerlik gstermektedir. Grevlerin zorluk dzeylerinin sınıflandırılmasında 7 uzmanın grřne bařvurulmuřtur. Bunun iin ikisi eęitim teknolojileri alanında uzman ve beři programlama eęitimi konusunda arařtırma yapan akademisyenden soruların zorluk dzeylerini deęerlendirmeleri istenmiřtir. Soruların zorluk dzeylerinin sınıflandırılmasında, uzmanların yanıtlarının ortalaması alınarak zorluk dzeyleri belirlenmiřtir (bkz Tablo 3.3).

3.3.3. Ders sreci gzlem formu

Arařtırmacı tarafından hazırlanmıř olan ve her ders sresince ęrencilerin derse katılımının, ders kapsamındaki kazanımlara ulařma dzeyi ve ne kadarını uygulayabildiklerinin llmesi saęlanmıřtır. Arařtırmacı her ders sresince ęrencilerin anlatılanları ne dzeyde anladıklarını, etkinlikleri ne dzeyde yapabildiklerini gzlemleyerek not etmiřtir. Ders sreci gzlem formu rneęi Tablo 3.6’da sunulmuřtur.

Tablo 3.6. Ders süreci gözlem formu kriterleri

Sıra no	Değerlendirme kriteri
1	Micro:bit aracının parçalarını bilebilme.
2	Micro:bit aracını doğru şekilde bilgisayara bağlayabilme.
3	MicroPython editörünü açabilme.
4	MicroPython editöründe program yazabilme.
5	MicroPython editöründe çalışma adını değiştirebilme.
6	Python programlama diliyle kod yazabilme.
7	MicroPython editöründe yazdığı kod parçasını micro:bit aracına yükleyebilme.
8	MicroPython editöründe yazdığı kod parçasını micro:bit aracında çalıştırabilme.
9	MicroPython editöründe yazdığı kod parçasında bir hatayla karşılaştığında çözebilme.
10	Karşılaştığı bir problemin daha önce karşılaştığı problemlere benzeyip benzemediğini ayırt edebilme. (Transfer yapabilme)
11	Micro:bit aracında karşılaştığı bir sorunu çözebilme.
12	Verilen etkinliği tek başına yapabilme.
13	Verilen bir konu hakkında (proje) program yazarak programlama aracında çalıştırabilme.
14	Kendi başına yaratıcı bir program oluşturma ve sunma.

Alanyazında yapılan araştırmalar göstermektedir ki, sınıf ortamında öğretmenin yapmış olduğu gözlemler, çoğu zaman değerlendirme amacıyla yapılan testler ya da ölçeklerden daha sağlıklı bilgiler verdiği gibi geçerliliği de daha yüksektir denilebilir (Kayabaşı, 2007). Öğrencilere karmaşık ve zor gelen programlama etkinlikleri (Aşkar ve Davenport, 2009); öğrenci performansının düşmesine, yanlış öğrenmelere, kavramsal bilgi eksikliğine (Kalelioğlu, 2015) ya da öğrenme zorluğuna kadar bazı sorunlar oluşturabilmektedir. Bu nedenle çalışmada, programlama öğretimi sırasında metin tabanlı programlama dili Python ile fiziksel programlama aracını ilk defa öğrenen öğrencilerin zorlandığı konularda ders esnasında hızlı müdahalede bulunmak amacıyla ders gözlemine önem verilmiştir. Böylece yanlış öğrenmelerin önüne geçilmesi hedeflenmiştir.

3.3.4. Ders sonu deęerlendirme soruları

Öğrencilerin o günkü derste anlatılan konulardan ne kadarını öğrendiklerini, o ders içerisinde anlatıların hangilerinde eksik ya da geri kaldığını belirlemek amacıyla “Ders Sonu Deęerlendirme Soruları” öğrencilerle dersin son 15 dakikası paylaşılmıştır. Bu ölçme aracı öğrencilerin yanlış, eksik ya da öğrenilememiş kavramları hemen ders sonrasında ortaya çıkarmaktadır. Buna göre araştırmacı bir sonraki dersin başlangıcında öğrencilere eksik olan kısımları gözden geçirerek tekrar etme imkânı sağlamıştır.

Ders sonu “Kendimi Deęerlendiriyorum” soruları araştırmacı tarafından hazırlanmış ve branş öğretmenlerinden uzman görüşü alınarak düzenlenmiştir. Sorular her hafta o derste öğrenilen kazanımları kapsamakta ve öğrencilerin kendilerini deęerlendirmelerine imkân vermektedir. Her ders sonu öğrencinin dersten ayrılmadan önce cevaplaması istenen farklı türlerde hazırlanmış sorulardır. Deęerlendirme soruları, farklı soru türlerinde, ilgili dersin programlama kavramlarına uygun olarak hazırlanmış ve öğrencilerden çevrimiçi olarak ders sonunda yanıtlamaları istenmiştir. Ders sonu kendimi deęerlendiriyorum soruları öğrencilerin süreç sonunda uygulanacak programlama kavramlarını öğrenme düzeyini ölçecek sorulardan bağımsız hazırlanmıştır. Soruların soruş şekli veya içeriğinde benzeşecek türden veriler içermemektedir. Uzman görüşü alınırken bu noktaya da dikkat edilmiştir. Süreç sonunda uygulanacak sorulara örnek teşkil etmemesine ve hatırlatıcı olmamasına önem verilmiştir.

3.3.5. Programlama aracı, kavram ve süreçlerine yönelik kazanım testi

Araştırma sonunda öğrencilere uygulanan çalışma sonu programlama aracı, kavram ve süreçlerine yönelik kazanım testinin amacı, öğrencilere kazandırılmaya çalışılan fiziksel programlama aracı ve metin tabanlı programlama dili ile ilgili kazanımlara süreç sonunda öğrencilerin ne kadarının ne düzeyde ulaştığını ortaya çıkarmaktır. Bu sebeple araştırmacı tarafından hazırlanmış 25 soru ve dört şıktan oluşan çoktan seçmeli test öğrencilere uygulanmıştır. Öğrenciler her bir doğru cevap için dört puan, yanlış veya boş bırakılan cevaplar için sıfır puan almışlardır. Kazanım testi, 100 puan üzerinden hesaplanmış olup testteki soruların içerięi, MEB “Seçmeli 7. ve 8. sınıf Bilişim Teknolojileri ve Yazılım Dersi” müfredatında yer alan programlama konusundaki kazanımlar ile uyumludur. Ayrıca testte, metin tabanlı programlama kavramları ile fiziksel programlama aracının ara yüzü ve kullanımı hakkında sorular da yer almaktadır. Kazanım testi hazırlandıktan sonra branş

öğretmenlerinden içerik ve görünüş geçerliliği hakkında uzman görüşleri alınarak gerekli düzenlemeler yapılmıştır. Tablo 3.7.'de programlama aracı, kavram ve süreçlerine yönelik hazırlanan kazanım testinin kazanımlar açısından frekans dağılımı gösterilmektedir.

Tablo 3.7. Programlama aracı, kavram ve süreçlerine yönelik kazanım testi sorularının kazanım frekansı.

Kazanım	Frekans
SBT.7.3.2.1. Programlama aracının ara yüzünü ve özelliklerini tanır.	3
SBT.7.3.2.2. Belirli bir problemi çözmek üzere geliştirdiği algoritmayı hatasız bir programa dönüştürür.	2
BT.7.3.2.3. Verilen bir probleme uygun söz dizimi oluşturur.	2
SBT.7.3.2.4. Verilen bir söz dizimini test eder ve hataları ayıklar.	3
SBT.7.3.2.5. Problemin çözümüne yönelik değişkenleri kullanır.	3
SBT.7.3.2.6. Problemin çözümüne yönelik koşullu ifadeleri kullanır.	3
SBT.7.3.2.7. Problemin çözümüne yönelik döngüleri kullanır.	4
SBT.7.3.2.8. Problemin çözümüne yönelik fonksiyonları kullanır.	5
Toplam	25

Tablo 3.7.'de görüldüğü gibi programlama aracı, kavram ve süreçlerine yönelik kazanım testi kapsamında MEB kazanımları ile sorular eşleştirildiğinde 25 sorunun; 3'ü programlama aracının ara yüzünü ve özellikleri, 2'si algoritmanın hatasız olarak programa dönüştürülmesi, 2'si verilen bir probleme uygun söz dizimi oluşturma, 3'ü verilen bir söz dizimini test etme ve hataları ayıklama, 2'si problemin çözümüne yönelik değişkenleri kullanma, 3'ü problemin çözümüne yönelik koşullu ifadeleri kullanma, 4'ü problemin çözümüne yönelik döngüleri kullanma ve 5'i problemin çözümüne yönelik fonksiyonları kullanma kazanımına karşılık gelmiştir.

Tablo 3.8.'e göre bilgi işlemsel düşünme beceri testindeki sorularda yer alan bilgi işlemsel düşünme boyutları tabloda listelenmiştir. Programlama aracı, kavram ve süreçlerine yönelik kazanım testi sorularının içeriği bilgi işlemsel düşünme boyutlarına göre şu şekilde olduğu söylenebilir; sorularda bilgi işlemsel düşünmenin bilgi boyutu algoritmanın bileşenlerini oluşturan kavramlardan meydana gelmektedir.

Tablo 3.8. Programlama aracı ve kavramlarını öğrenim düzeyi ölçme testi sorularının programlama yapıları ve bid becerileri açısından değerlendirilmesi.

	Ara yüz tanıma	Bileşen özelliklerini tanıma	Problem analizi	Fonksiyon kullanımı	Parçalara ayırma	Soyutlama	Döngü kullanımı	Değerlendirme / Hata ayıklama	Listele kullanımı	Kütüphane kullanımı	Programlama dili	Modül kullanımı	Problem çözme	Algoritma	Koşul kullanımı	Değişken kullanımı
1. Soru	✓															
2. Soru		✓	✓	✓												
3. Soru					✓	✓	✓	✓								
4. Soru			✓	✓	✓	✓										
5. Soru								✓	✓							
6. Soru				✓						✓	✓					
7. Soru				✓							✓					
8. Soru				✓						✓		✓				
9. Soru				✓				✓		✓						
10. Soru			✓		✓		✓									
11. Soru	✓					✓		✓							✓	
12. Soru	✓	✓	✓			✓		✓								
13. Soru	✓		✓			✓		✓							✓	
14. Soru			✓				✓	✓							✓	✓
15. Soru				✓		✓									✓	✓
16. Soru						✓	✓									
17. Soru	✓		✓			✓	✓	✓								
18. Soru	✓	✓														
19. Soru	✓															
20. Soru	✓	✓														
21. Soru			✓	✓		✓		✓							✓	
22. Soru			✓	✓		✓	✓	✓								✓
23. Soru	✓							✓								
24. Soru								✓							✓	
25. Soru			✓	✓		✓		✓								

3.3.6. Odak grup görüşmesi

Odak grup görüşmesi bir grubun içerisinde 4 – 12 kişilik bir katılımcı grup oluşturularak bir moderatörün yönetiminde önceden dikkatlice planlanmış sorulara gerçek duygu ve düşüncelerini saklamadan cevap verilmesi, herkese eşit şekilde söz hakkı tanınarak oluşturulacak ortamda nitel veri toplama tekniği olarak ifade edilmektedir (Gülcan, 2021; Krueger ve Casey, 2000). Odak grup görüşmelerinin olumlu yanları; nitel verileri kullanılabilir hale getirir, katılımcıların görüşme sırasında etkileşim halinde olarak birbirlerini etkilemeleri (cesaretlendirmek, nerden başlayacağına yardımcı olmak, fikirlerini toplamak vb.), bir fikir ortaya çıktıktan sonra tüm katılımcılar tarafından bu fikir geliştirilebilir böylece konu ile ilgili detaylı bilgilere ulaşılabilir (Lewis, 1995; Kitzinger, 1995; Gibbs, 1997).

Araştırma sorularından “Öğrencilerin uygulanan fiziksel programlama aracı destekli metin tabanlı programlama eğitimine yönelik görüşleri nelerdir?” sorusuna yanıt verebilmek amacıyla öncelikle sorular hazırlanmıştır. Araştırmacı tarafından hazırlanan odak grup görüşme soruları, sonrasında alan uzmanları ve akademisyenlerden, uzman görüşleri alınarak son haline dönüştürülmüştür. Görüşme soruları toplam 4 sorudan oluşmaktadır. Soruların içeriği, fiziksel programlama aracını kullanarak metin tabanlı programlama dili Python öğrenme sürecinde öğrencilerin deneyimlerine yönelik görüşlerine yönelik hazırlanmıştır.

Odak grup görüşmesinde öğrencilere yöneltilen sorular aşağıda maddelendirilmiştir;

1. Fiziksel programlama aracı ile Python öğrenme konusunda ne düşünüyorsunuz? Python programlama öğrenirken fiziksel programlama aracının programlama kavramlarını / programlamayı öğrenirken size yardımcı olduğunu düşünüyor musunuz?
2. Fiziksel programlama aracı ile Python öğrenirken en çok sizi ne motive etti? Bu sürecin iyi yönleri nelerdir?
3. Fiziksel programlama aracı ile Python öğrenirken zorladığınız veya sorun yaşadığınız durumlar oldu mu? Varsa bu sürecin olumsuz yönleri nelerdir?
4. Fiziksel programlama aracı ile Python öğretim sürecinde nelerin değişmesini istersiniz? Daha iyi öğrenmek için neler önerirsiniz?

Odak grup görüşmesi için seçilen öğrenciler; araştırmanın uygulama sonrası öğrencilere uygulanan programlama aracı, kavram ve süreçlerine yönelik kazanım testinden aldıkları puanlara göre 2 kişi en yüksek, 2 kişi orta ve 2 kişi en düşük olarak tespit edilen her seviyeden 6 öğrenci seçilmiştir. Her seviyeden 6 kişi olarak seçilen (7. sınıflardan 6 öğrenci ve 8. sınıflardan 6 öğrenci olmak üzere) toplam 12 öğrenci ile odak grup görüşmesi yapılmıştır. Görüşme öncesi öğrencilerden izinleri alınarak görüşmenin sesli kaydı alınmıştır. Daha sonra nicel araştırmalara destek olması amacıyla kaydedilen ses kayıtları nitel veri çözümleme yöntemlerine göre değerlendirilmiş ve raporlaştırılmıştır.

3.4. Verilerin Analizi

Verilerin analizi için IBM SPSS Statistics” versiyonu 28.0.1.1 istatistik programı kullanılmıştır. Ön test ve son test olarak “Bilgi İşlemsel Düşünme Becerilerine Yönelik Öz Yeterlik Algısı Ölçeği” ve “Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme Görevleri” ile son test olarak programlama aracı, kavram ve süreçlerine yönelik kazanım testi uygulanmıştır. Araştırmanın evrenini oluşturan gruplarda ön test ve son test puanları bakımından bir farklılık olup olmadığına bakmak için tek örneklem t-testi (one sample t-test) yapılmıştır. Tek örneklem t-testi; bir ortalama hakkında bir öngörü ya da bir tahminde bulunduğumuz zaman bunun doğruluğunu test etmek için kullanılmaktadır.

Uygulama sürecinde öğrencilerin ders sonu değerlendirme sorularına verdikleri cevaplar her ders sonunda araştırmacı tarafından okunarak değerlendirilmiştir. Değerlendirme sonucuna göre bir sonraki hafta dersin plan ve etkinlikleri gözden geçirilmiştir. Haftalık ders değerlendirme soruları, ilgili haftanın son dersinin 15 dakikası ayrılarak öğrencilere cevaplatılmıştır. Her öğrencinin her soruya cevap vermesi sağlanmış ve cevaplar kontrol edilerek toplanmıştır. Haftalık sorular hazırlanırken öğrencilerin daha hızlı ve kolay cevaplayacağı düşünülerek ilk hafta 3'lü likert tipi hazırlanmıştır. Ancak öğrenciler cevaplama yaparken hem öğretmene sordukları sorulardan hem de kendi aralarındaki konuşmalarından bu tarz bir cevaplamaya alışkın olmadıkları dikkat çekmiştir. Çünkü öğrencilerin olumsuz işaretlemekten çekindikleri, olumlu bile düşünseler ortalama cevap vermenin daha iyi olacağına karar verdikleri gözlemlenmiştir. Öğrencilerin not alma kaygısı ile cevap vermemeleri için her değerlendirme başında verecekleri cevapların sonucunda not almayacakları hatırlatılmıştır. Her hafta öğrenciler gözlemlenerek sorulara tepkileri değerlendirilmiştir. İkinci hafta ve sonrasında farklı ölçme sorularına yer verilerek

likert tipi ölçme soru sayısı azaltılmış çoklu onay ve doğru/yanlış soru tipleriyle sorular sorulmuştur. Öğrencilerin çoğunlukla doğru ya da yanlış gibi net cevap verebilecekleri soruları cevaplamaya daha yatkın oldukları gözlemlenmiştir. Haftalık değerlendirmelerde likert tipi sorularda cevaplar katılıyorum 2 puan, ne katılıyorum ne katılmıyorum 1 puan ve katılmıyorum 0 puan; evet/hayırlı sorularda cevaplar evet 3 puan, hayır 0 puan; çoklu onay soruların cevaplarında işaretledikleri her şık için 3 puan şeklinde puanlar hesaplanmıştır. Ayrıca süreç içerisinde her ders öğrenciler için ders süreci gözlem formu doldurulmuştur. Bu form sayesinde her öğrencinin o sürece kadar ki ilerleme düzeyini görme imkânı bularak öğrencilerin seviyesine göre dersi sürdürme ya da gerekli görüldüğü durumlarda tekrar etkinlikleri yapılması sağlanmıştır. Bu sayede öğrencilerden kaynaklı (farklı öğrenme yöntemi, hızı veya dışsal motivasyon gibi) ders akışını bozacak durumlar engellenmeye çalışılmıştır.

Uygulama sonrasında yapılan 25 soruluk çoktan seçmeli programlama aracı, kavram ve süreçlerine yönelik kazanım testi için puan hesaplama yöntemi; her doğru cevap için 4 puan, yanlış ve boş bırakılan sorular için sıfır puan verilmiştir. Programlama aracı, kavram ve süreçlerine yönelik kazanım testinden alınabilecek en yüksek not 100 olarak hesaplanmıştır. Çünkü 7. ve 8. sınıflarda Seçmeli Bilişim Teknolojileri ve Yazılım dersi iki kredilik ders kapsamındadır. Bu nedenle dönemde iki sınav notu ve bir ders içi katılım notu verilmektedir. Bu bağlamda yapılan programlama aracı, kavram ve süreçlerine yönelik kazanım testi aynı zamanda öğrencilerin dersin yapıldığı dönemin bir sınav notu şeklinde değerlendirilmiştir. Ayrıca odak grup görüşmesinin yapılacağı grup seçilirken programlama aracı, kavram ve süreçlerine yönelik kazanım testi puanları göz önünde bulundurularak grup oluşturulmuştur.

Öğrencilerin uygulama sonrasında odak grup görüşmesi ile öğrenme süreci hakkında görüşleri alınmış ve buna göre nitel bir değerlendirme yapılmıştır. Odak grup görüşmelerinde, verilerin analizinde, içerik analizinin en uygun yöntem olduğu söylenmektedir (Kitzinger ve Farquhar, 1999). İçerik analizi yaparken araştırmacılara, elde ettikleri verileri niceliksel yöntemle saymak yerine niteliksel bir yaklaşımla daha zengin ve karma yöntemle değerlendirmelerini tavsiye etmektedirler (Onwuegbuzie ve arkadaşları, 2009). Odak grup görüşmesinin raporlaştırılmasında önemli olan nicel yani sayısal değerler değil katılımcıların ne söylediğidir (Creswell, 2021). Görüşme sonunda öğrencilerden toplanan veriler küçük parçalara ayrılmış ve bu parçalar sınıflandırılmıştır.

Bu sınıflandırmalar cevaplar arasında benzerlik içeren anahtar kelimelere göre karşılaştırılarak analiz edilmiştir (Gülcan, 2021). Odak grup görüşmesi analizinde genel olarak anahtar kelimelere göre belirlenen cevaplara yer verilmiştir. Tüm bu durumlar göz önüne alınarak odak grup görüşmesi analizi yapılarak bir değerlendirme ortaya çıkartılmıştır.

3.5. Araştırma Süreci

2021-2022 eğitim ve öğretim yılı bahar döneminde yapılan araştırma 6 haftalık öğretim sürecinde yapılmıştır. Araştırma öncesi araştırmacı tarafından çalışmış olduğu ortaokuldan bu araştırmayı yapabilmek adına izin alınmıştır. Ayrıca bu araştırma için Başkent Üniversitesi Eğitim Bilimleri Enstitüsü, Bilimsel Araştırma ve Yayın Etiği Kurulunun kararı ile alınan izinle yürütülmüştür. Ortaokul seçmeli bilişim teknolojileri ve yazılım dersi müfredatı kapsamında programlama öğretiminde metin tabanlı programlama dili ve fiziksel programlama aracı ile ilgili doğrudan bir kazanım bulunmamaktadır. Fiziksel programlama aracı ile ilgili öğrencilere kazandırılması hedeflenen kazanımlar ihtiyaç doğrultusunda belirlenmiştir ve bunlara yönelik kazanımlar oluşturulmuştur. Metin tabanlı programlama dili Python öğretimi micro:bit ile verildiğinden, süreç içerisinde programlama yapılırken öncelikle bilgisayarda program yazılması gerekmiştir. Öğrencilerin çalışmalarını yürütebilmeleri için her öğrenciye bir bilgisayar düşecek şekilde laboratuvar da dersler yürütülmüştür. Laboratuvar içerisinde öğrencilerin yazdıkları programları somut bir şekilde deneyimleyebilecekleri micro:bit'lerin öğrenci sayısına uygun olması sağlanmıştır. Öğrencilerin motivasyonunu sağlamak ve interaktif bir eğitim ortamı yaratmak adına fiziksel programlama aracı ile öğrencilerin daha özgür hareket edebilmeleri açısından micro:bit'e takılabilen pillerden zaman zaman kullanılmıştır. Ders esnasında kullanılmak üzere araştırmacı tarafından hazırlanmış slaytlar, posterler, çalışma yaprakları kullanılarak dersler yürütülmüştür.

Programlama öğretiminde ülkemizde yeni yeni bilinmeye başlanan micro:bit fiziksel programlama aracı ve bu aracın metin tabanlı Python programlama dili öğretiminde kullanılması farklı bir öğretim programı olarak ortaya çıktığı ifade edilebilir. Araştırma sürecinde konu kazanımları araştırmacı tarafından öğrencilere anlatılıp, uygulamaları yaptırılmıştır. Bu süreçte sözlü, metinsel ve görsel medya kullanılmıştır. Ders sürecinde kazanımlar verilirken sırasıyla sürecin işleyişi; konu anlatımı, gösterip yaptırma, soru-

cevap, uygulama ve ders sonu deęerlendirmesi řeklinde olmuřtur. Arařtırmanın bařında ve sonunda yani uygulama ncesi ve sonrasında “Bilgi İřlemsel Düşünme Becerilerine Ynelik z Deęerlendirme Yeterlik Algısı leęi”, “Bilgi İřlemsel Düşünme Etkinlik Soruları”na ek olarak son testte “Programlama Aracı, Kavram ve Srelerine Ynelik Kazanım Testi” uygulanmıř ve “Odak Grup Grüşmesi” yapılmıřtır. Uygulama srecinde ęrencilerin konuyu anlayıp anlamadıklarını ise lmek adına soru-cevap yntemi kullanılmıř ayrıca ders gzlem formuna arařtırmacı tarafından srekli not alınmıřtır. ęrencilerin her ders sonunda laboratuvarından ıkmadan “Ders sonu kendimi deęerlendiriyorum” sorularını yanıtlaması saęlanmıřtır.

Ders srecinde programlama amacıyla yapılan etkinlikler fiziksel programlama aracı, metin tabanlı programlama dili, bilgi iřlemsel düşünme beceri ve kavramlarını iermektedir. ęretim sreci planlanırken konular, Bloom’un Tam ęrenme Modelindeki kk adımlar ilkesi gereęince kk birimlere blnen kazanımlar ařamalı olarak verilmiřtir (Hyder ve Bhamani, 2016; Nayef, Yaacob ve İsmail, 2013). ęretim programında yer aldıęı gibi belli kazanımlar iřlendikten sonra, iřlenen kazanımları ieren programlama ev devi verilmiřtir. Arařtırmanın ęretim srecinde toplamda 7 dev ęrencilere tanımlanmıřtır. Bu sayede ęrenciler ęrendięi kodlama kavramlarını tekrar ederek pekiřtirme imkânı bulmuřtur. Evde kendi kendine algoritma tasarlayarak verilen bir problemi zmeye alıřan bir ęrenci bir probleme dayalı programı yazmayı deneyimlemiř olduęu düşünlmektedir. Bunun sonucunda evde tek bařına programlama yapma fırsatı bularak kendi kendine konuyu tekrar ettięi ve pekiřtirdięi sylenebilir. Bir sonraki dersin bařlangıcında, ęrencilere devi yapan ve yapmayanlar sorularak kimlerin yapıp/yapmadıęı not alınmıřtır. Evde yapılan devlerin, seilen bir ęrenci ile sınıf ortamında uygulamalı olarak tekrar edilmesi saęlanır. İlgili devin zm ve kontrol bu řekilde yapılmıřtır. Arařtırmacı tarafından hazırlanan derslerde sunulan bilgi ierikli sunular ve devler konusu alan uzmanları tarafından kontrol edilip ierik ve grnm geerlilięi saęlanmıřtır. ęretimin programı Tablo 3.9’da ayrıntılı olarak yer almaktadır.

Tablo 3.9. Öğretim programı tablosu.

	Öncelik Sırası ve Amaç	Değerlendirme
1. Ders	Micro:bit'i tanıma.	Gözlem formu
1. Ders	Micro:bit üzerindeki bölümleri tanıma. Ödev Kazanımları: Micro:bit aracının üzerindeki bölümlerin ismini ve ne işe yaradığını tekrar etme fırsatı verir.	Gözlem formu – Ödev: “Micro:bit Bileşenlerini Öğreniyorum” boşluk doldurma kâğıdı öğrencilere dağıtılır. - Kendimi Değerlendiriyorum Soruları
2. Ders	Metin tabanlı programlama dilinin kullanılacağı platform micropython editörünün ara yüzünü tanıma.	Gözlem formu
2. Ders	Metin tabanlı programlama micropython editöründe çalışma adını değiştirebilme.	Gözlem formu
2. Ders	Metin tabanlı programlama olan micropython editörünün araç çubuğunu tanıma.	Gözlem formu- Kendimi Değerlendiriyorum Soruları
3. Ders	Metin tabanlı programlama olan micropython editöründe “Hello Word” kod bloğunu yazdırma. “from microbit import” ve “display.scroll” kod bloklarını açıklama. Etkinlik; Metin yazdırma.	Gözlem Formu
3. Ders	Metin tabanlı programlama olan micropython editöründe “Images” kod bloğunu yazdırma. “from microbit import” ve “display.scroll” kod bloklarını açıklama. Ödev Kazanımı: Kütüphane kullanımını keşfeder. İsteddiği bir görseli kütüphaneden çağıran algoritmayı kurar.	Gözlem formu – Ödev: “Hadi kendi görselimizi oluşturalım” konulu çalışma öğrencilere verilir.
4. Ders	Metin tabanlı programlama olan micropython editöründe “DIY Images” (Do it yourself) kod bloğunu yazdırma. “from microbit import” ve “display.scroll” kod bloklarını açıklama.	Gözlem formu
4. Ders	Metin tabanlı programlama olan micropython editöründe “Animation” kod bloğunu yazdırma. “While” döngü bloğunu kullanabilme. “Sleep” komutunu kullanır. Ödev Kazanımı: Kodlayarak döngü kavramını kullanır.	Gözlem formu – Ödev: “Ördeğimizi hareket ettirelim” konulu çalışma öğrencilere verilir. - Kendimi Değerlendiriyorum Soruları

5.Ders	Metin tabanlı programlama olan micropython editöründe “Listeler” mantığını bilir. Ödev Kazanımı: Loops-döngü mantığını öğrendiğini ölçek için daha önce yaptığı statik bot etkinliğini animasyon haline getiren ödev öğrenciye tanımlanır.	Gözlem formu – Ödev: “Teknemiz Batıyor!” konulu çalışma öğrencilere verilir.
6.Ders	Metin tabanlı programlama olan micropython editöründe “Image.all_cocks ve image.all_arrows” kavramlarını bilir.	Gözlem formu– Ödev: “Göz Kırmıyorum!” konulu çalışma öğrencilere verilir. - Kendimi Değerlendiriyorum Soruları
7.Ders	Metin tabanlı programlama olan micropython editöründe programlama aracı üzerinde yer alan pinlerle- butonlarla çalışma yapmayı bilir.	Gözlem formu. - Kendimi Değerlendiriyorum Soruları
8.Ders	Microbit programlama aracı üzerinde rastgele (random) sayı üretme. Ödev Sorusu: ““Bu programın neden ilk zar örneğinde olduğu gibi cihazı sıfırlamak yerine A düğmesine basmamızı gerektirdiğini çözebilir misiniz?”	Gözlem formu – Ödev: “Zar Oyunu” hakkında bir soru öğrencilere tanımlanır.
9.Ders	Microbit programlama aracı üzerinde bulunan “accelerometer” ivme-ölçeri tanır.	Gözlem formu- Kendimi Değerlendiriyorum Soruları
10.Ders	Micro:bit programlama aracı üzerinde bulunan “accelerometer” ivme-ölçeri tanır. Hareket algılama etkinliği yapılır. Ödev Kazanımı: “taş-kâğıt-makas” oyununu kodlar.	Gözlem formu – Ödev: “Taş-Kâğıt-Makas” konulu çalışma öğrencilere verilir.
10.Ders	Micro:bit programlama aracı üzerinde bulunan “Yön Bulma” kullanılarak “Pusula” etkinliği yapılır.	Gözlem formu- Kendimi Değerlendiriyorum Soruları
11-12.Ders	Proje Çalışması; Uygulama basamağı olduğu için öğrenciden tüm öğrendiği programlama bilgileriyle program oluşturarak micro: bit aracı ile sunması istenir. Örnek Olarak; “Magic-8” ya da “Tetris” oyunu verilebilir.	Dereceli Puanlama Anahtarı- Kendimi Değerlendiriyorum Soruları

3.6. Araştırmacının Rolü

Araştırmacı görev yaptığı okulda ve seçmeli ders olarak 2021-2022 eğitim ve öğretim yılında seçmeli bilişim teknolojileri ve yazılım dersini seçen 7. ve 8. sınıf öğrenciler ile bu araştırma gerçekleştirilmiştir. 7. ve 8. Sınıflar ile 6 hafta boyunca tüm ders anlatımları gerçekleştirilmiştir. Uygulama öncesi 2 ders ön test ve uygulama sonrası 2

ders son test olmak üzere toplan 4 ders saatinde ders anlatım haricinde veri toplama süreci için arařtırmacı tarafından gerekleřtirilmiřtir. Arařtırma kapsamda dersin planlaması, ders materyallerinin ve ölçme-değerlendirme amacıyla kullanılan araçların hazırlanması, derslerin işlenmesi ve katılımcıların arařtırma sürecinde takibi arařtırmacı tarafından yapılmıřtır. İlgili veri analizlerinin raporlařtırılması arařtırmacı tarafından gerekleřtirilmiřtir.



BÖLÜM IV

BULGULAR VE TARTIŞMA

4.1.Bulgular

Araştırmada toplanan nicel ve nitel verilerin analizi ile elde edilen sonuçlar, bu bölümde araştırmada incelenen alt problemler doğrultusunda sunulmuştur. Yapılan analizlerde anlamlılık düzeyi olarak “ $p \leq 0.05$ ” kabul edilmiştir. Çalışmanın analiz sürecine verilerin özetlenmesiyle başlanmıştır.

Bu bölümde fiziksel programlama aracı destekli metin tabanlı programlama öğretiminin öğrencilerin uygulama öncesi ve sonrası bilgi işlemsel düşünme becerileri arasında anlamlı bir fark olup olmadığı belirlenmek istenmiştir. Bu amaçla ön test ve son test olmak üzere Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme sorularına ilişkin bulgulara yer verilmiştir. Uygulanmış olan bu testlere ilk olarak normallik testi uygulanmıştır. BİDBÖA ölçeği ve Bilge Kunduz sorularından elde edilen verilerin normal dağılım sergilediğini belirlemek için şu kriterler değerlendirilmiştir; ortalama, medyan ve mod değerlerinin birbirine yakın olup olmadığı, histogram dağılımının çan eğrisini oluşturup oluşturmadığı ve Skewness (çarpıklık) ve Kurtosis (basıklık) katsayılarının -1.5 ve +1.5 aralığında olup olmama durumlarına bakılmıştır (Büyüköztürk ve arkadaşları, 2018; Can, 2019). Araştırmadaki örneklem sayısının 30 ve üstü ($n \geq 30$) olduğu için normal dağılım uygunluğu, Kolmogorov Smirnov testi ile analiz edilmiştir.

4.1.1. Uygulama öncesi ve sonrası öğrencilerin bilgi işlemsel düşünme becerilerine yönelik öz yeterlik algılarına ilişkin bulgular;

Bu bölümde fiziksel programlama aracı destekli metin tabanlı programlama öğretiminin bilgi işlemsel düşünme becerilerine yönelik öğrencilerin öz yeterlik algısına etkisini belirlemek amacıyla ön test ve son test olarak Bilgi İşlemsel Düşünme Becerilerine Yönelik Öz Yeterlik Algısı Ölçeği (BİDBÖA) uygulamasına ilişkin bulgular sunulmaktadır. Analiz aşamasında ölçeğin toplam puanı ve ölçekte var olan beş faktörün puanları hesaplanmıştır. Beş faktör ölçekte sırasıyla; Algoritma Tasarlama Yeterliği, Problem Çözme Yeterliği, Veri İşleme Yeterliği, Temel Programlama Yeterliği ve

Özgüven Yeterliği başlıklarıyla yer almıştır. Tüm bu faktörler öğrenci verileri ile karşılaştırılmıştır. Öğrencilerin bilgi işlemsel düşünmeye ilişkin öz yeterlik algıları ölçeğe yer alan 36 maddeye verdikleri cevaplar puanlanmış ve öğrencilerin toplam öz yeterlik puanları hesaplanmıştır. Ölçekten almış oldukları puanlar bilgi işlemsel düşünme becerilerine yönelik öz yeterlik algıları hakkında bilgi vermektedir. Buna göre ölçekten öğrencinin almış olduğu puanın düşük olması durumunda öğrencinin bilgi işlemsel düşünme öz yeterlik algısının düzeyinin düşük olduğu; öğrencinin ölçekten yüksek puan alması durumunda ise öğrencinin bilgi işlemsel düşünme öz yeterlik algı düzeyinin yüksek olduğu sonucu çıkarılmaktadır.

Öğrencilerden toplanan veriler ile yapılmış olan güvenirlik testinde; iç güvenirlik katsayısı Cronbach Alfa değeri 0.89 olarak bulunmuştur (Tablo 4.1). BİDBÖA ölçeğinde faktörlere ait güvenirlik katsayılarının 0.76 ile 0.93 arasında değiştiği görülmektedir. Araştırma sonucunda ön test olarak BİDBÖA ölçeğine ait güvenirlik katsayıları 0.78 ile 0.92 arasında bulunmuştur. Güvenirlik katsayısının 0.70 ile 0.90 aralığında olması ölçeğin yüksek güvenirliğe sahip olduğunu göstermektedir (Büyüköztürk, 2007; Hinton ve arkadaşları, 2014). Bu nedenle, bütün ölçek yapısı ve alt faktörlerinin yeterli güvenirlik değerlerine sahip olduğu söylenebilir.

Tablo 4.1. BİDBÖA ölçeği ve öğrenciye uygulanan ön testin Cronbach's alpha katsayısı.

Alt faktörler	Araştırma cronbach's alpha katsayısı
Algoritma tasarlama yeterliği	.92
Problem çözme yeterliği	.85
Veri işleme yeterliği	.83
Temel programlama yeterliği	.81
Özgüven yeterliği	.78
Genel ölçek sonucu	.89

Yapılan analizler sonucunda Skewness (çarpıklık) katsayısının (0.73) ve Kurtosis (basıklık) katsayısının (0.28) olarak bulunmuştur. Basıklık ve çarpıklık değerleri analiz sonuçları için +1.5 ve -1.5 aralığını normal dağılım olduğunu ifade edilmiştir (Tabachnick ve Fidell, 2012). Buna göre araştırmanın çarpıklık ve basıklık değerlerinin normal dağılım gösterdiği sonucuna ulaşılmıştır. Ortalama, medyan ve mod değerlerinin birbirine yakın

olması ve histogram dağılımının çan eğrisini oluşturmuş olması da yine normal bir dağılım olduğunu göstermektedir. BİDBÖA ölçeğinin ön test verisinin ortalaması (58.84), mod değeri (54.68) ve ortanca değeri (56.79) olarak hesaplanmıştır. Değerlerin birbirine yakın olması testin normal dağılım olduğunu göstermektedir. Corder ve Foreman (2009) tarafından, araştırmadaki örneklem sayısının 30 ve üstü ($n \geq 30$) olduğu durumlarda Kolmogorov Smirnov testi ile ölçekteki dağılımının uygunluğu test edilmesi önerilmektedir. Bu araştırmada çalışma grubunu 85 öğrenci oluşturduğu için bu test seçilmiş ve analizler buna göre değerlendirilmiştir. Kolmogorov Smirnov testi sonucu, 0.071 olduğundan ön test verisi normal dağılım göstermektedir.

Çalışma öncesi ve sonrası Bilgi İşlemsel Düşünme Becerilerine Yönelik Öz Yeterlik Algısı Ölçeği (BİDBÖA) öğrencilere uygulanmıştır ve veriler analiz edilmiştir. Verilerin analizi aşamasında ölçekte var olan beş faktör (algoritma tasarlama yeterliği, problem çözme yeterliği, veri işleme yeterliği, temel programlama yeterliği ve özgüven yeterliği) açısından bulgular sunulmuştur.

BİDBÖA Ölçeğine ve Alt Faktörlerine Ait Ön Test İstatistiksel Analiz Bulguları;

Araştırmanın uygulama süreci öncesinde öğrencilerin BİDBÖA ölçeğine vermiş oldukları yanıtların analizi sonucunda elde edilen veriler Tablo 4.2’de sunulmuştur.

Tablo 4.2. BİDBÖA ölçeğinin ve alt faktörlerine ait ön test sonuç tablosu.

Alt faktörler	$\bar{X}$	S	Min.	Max.
Algoritma tasarlama yeterliği	12.18	4.38	9	27
Problem çözme yeterliği	14.92	4.59	10	30
Veri işleme yeterliği	11.13	3.01	7	21
Temel programlama yeterliği	11.14	3.89	5	15
Özgüven yeterliği	9.47	2.23	5	15
Genel ölçek sonucu	58.84	5.47	36	78

Buna göre, algoritma tasarlama yeterliği faktörü ortalama puanı 12.18 (sd= 4.38), problem çözme yeterliği faktörü ortalama puanı 14.92 (sd= 4.59), veri işleme yeterliği faktörü ortalama puanı 10.42 (sd= 3.07), temel programlama yeterliği faktörü ortalama puanı 11.13 (sd=3.01), özgüven yeterliği faktörü ortalama puanı 9.47 (sd=2.23) olarak bulunmuştur. Faktörlere ilişkin puanların genel olarak ortalamanın altında olduğu görülmektedir. Buna göre uygulama öncesi öğrencilerin ölçekten almış oldukları puanların

düşük olması nedeniyle öğrencilerin bilgi işlemsel düşünme becerisine yönelik öz yeterlik algı düzeylerinin düşük olduğu sonucunda ulaşılabılır.

BİDBÖA Ölçeğine ve Alt Faktörlerine Ait Son Test İstatistiksel Analiz Bulguları;

Araştırmanın uygulama süreci sonunda öğrencilerin BİDBÖA ölçeğine vermiş oldukları yanıtların analizi sonucunda elde edilen veriler Tablo 4.3’de sunulmuştur.

Tablo 4.3. BİDBÖA ölçeğinin ve alt faktörlerine ait son test sonuç tablosu.

Alt faktörler	$\bar{X}$	S	Min.	Max.
Algoritma tasarlama yeterliği	24.16	3.89	9	27
Problem çözme yeterliği	27.87	4.06	10	30
Veri işleme yeterliği	18.92	3.94	7	21
Temel programlama yeterliği	13.76	3.02	5	15
Özgüven yeterliği	13.54	2.91	5	15
Genel ölçek sonucu	96.89	14.58	86	108

Buna göre, algoritma tasarlama yeterliği faktörü ortalama puanı 24.16 (sd= 3.89), problem çözme yeterliği faktörü ortalama puanı 27.87 (sd= 4.06), veri işleme yeterliği faktörü ortalama puanı 18.92 (sd= 3.94), temel programlama yeterliği faktörü ortalama puanı 13.76 (sd= 3.02), özgüven yeterliği faktörü ortalama puanı 13.54 (sd= 2.91) olarak bulunmuştur. Faktörlere ilişkin puanların ön teste göre arttığı, genel ölçek ortalama puanının da arttığı görülmektedir. Bu noktada uygulama sonrası öğrencilerin ölçekten almış oldukları puanların artmış olması sebebiyle öğrencilerin bilgi işlemsel düşünme becerisine yönelik öz yeterlik algı düzeylerinin yükselmiş olduğu sonucuna ulaşılabılır.

BİDBÖA Ölçeğine Ait Ön Test ve Son Test Karşılaştırma Bulguları;

Araştırmanın uygulama öncesi ve sonrası öğrencilerin BİDBÖA ölçeğine vermiş oldukları yanıtların analizleri sonucunda elde edilen veriler Tablo 4.4’de sunulmuştur.

Tablo 4.4. BİDBÖA ölçeğine ait ön test ve son test karşılaştırma sonuçları.

	N	$\bar{X}$	Medyan	S	Sd	t	p	d	Min.	Max.
Ön Test	85	58.84	56.79	13.2					36	78
Son Test	85	96.89	94.03	10.03	14	29.17	.027	2.17	86	108

İstatistik sonuçları değerlendirildiğinde Tablo 4.4’de görüldüğü gibi BİDBÖA puan ortalamalarının eğitim öncesinde 58.84’ten 96.89’a yükseldiği görülmektedir. Öğrenciler ölçekten uygulama öncesi 36 ile 78 arasında puanlar alırken, uygulama sonrası 86 ile 108 arasında puanlar almışlardır. Medyanın son testte 94.03 ve aritmetik ortalamanın 96.89 olması çalışmanın normal dağılıma yakın olduğunu göstermektedir. Ayrıca öğrencilerin çalışma sonrasında bilgi işlemsel düşünme becerilerine yönelik öz yeterlik algılarının ortalamasının altında olması ve ölçek alt puanına yakın olması gelişimlerini göstermektedir. Ön test ve son test puanları arasında istatistiksel olarak ulaşılan $p=0.03$ anlamlılık düzeyinin $p<0.05$ küçük olduğu görülmüştür. Anlamlılık düzeyinin farklı olması sebebiyle ve ortalamalarda görülen farklılıklar fiziksel programlama destekli metin tabanlı programlama eğitimi alan öğrencilerin bilgi işlemsel düşünme becerisine yönelik öz yeterlik algı seviyelerinde istatistiksel olarak anlamlı bir artış olduğunu göstermektedir. Test sonucunda hesaplanmış olan etki büyüklüğü son test ile ön test arasındaki farkın yüksek 2.17 olduğunu göstermektedir. Etki büyüklüğü 0.2 ise az, 0.5 ise orta, 0.8 ise büyük ve 1 değeri üzerinde ise çok büyük düzeyde etkisi olduğu ifade edilir (Green ve Salkind, 2014; Tabachnick ve Fidell, 2012). Bu sonuçlara göre, uygulama sonrası öğrencilerin ölçekten almış oldukları puanların artmış olması sebebiyle öğrencilerin bilgi işlemsel düşünme becerisine yönelik öz yeterlik algı düzeylerinin artmış olduğu sonucuna ulaşılmıştır. Sonuç olarak, fiziksel programlama aracı destekli metin tabanlı programlama öğretiminin bilgi işlemsel düşünme becerilerine yönelik öğrencilerin öz yeterlik algısına etkisi vardır denilebilir.

4.1.2. Uygulama öncesi ve sonrası öğrencilerin bilgi işlemsel düşünme becerilerine ilişkin bulgular;

Uygulama öncesi ve sonrası öğrencilerin bilgi işlemsel düşünme becerilerine yönelik anlamlı bir fark olup olmadığını ölçmek adına geçmiş yıllarda Türkiye’de uygulanmış olan Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme görevlerinden oluşan test uygulanmıştır.

Uygulama Öncesi Öğrencilerin Ön Test Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme Görevi Başarılarına İlişkin Bulgular; Çalışmada öğrencilere ön test ve son test olarak uygulanan 15 soruluk Bilge Kunduz Uluslararası Enformatik ve Bilgi

İşlemsel Düşünme görevlerinde üç farklı düzeyde görev olacak şekilde seçilmeye çalışılmıştır.

Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme Görevlerine Ait Ön Test İstatistiksel Analiz Bulguları; Çalışmada öğrencilere ön test olarak uygulanan 15 soruluk Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme görevlerinin ortalaması analiz sonucunda 19.81, çarpıklık değeri 0.538, basıklık değeri 0.62 olarak bulunmuştur. Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme Testinden alınan puanların çarpıklık ve basıklık değerleri +1.5 ve -1.5 aralığında olduğundan ve çalışma grubunu oluşturan öğrencilerin sayısının 30'dan fazla olması normal dağılım olduğunu göstermektedir (Tabachnick ve Fidell, 2012; Demir, Saatçioğlu ve İmrol, 2016). Bilge kunduz uluslararası enformatik ve bilgi işlemsel düşünme ön testinin istatistiksel sonuçları Tablo 4.5'te verilmiştir.

Tablo 4.5. Bilge kunduz uluslararası enformatik ve bilgi işlemsel düşünme görevleri ön testinin istatistik sonuçları.

Ön test	N	$\bar{X}$	Medyan	S	Min.	Max.
Bilge kunduz görevleri	85	19.81	21.35	17.28	17	72

Tablo 4.5.'te yer alan değerlere göre; öğrencilerin ön test puanlarının ortalaması 19.81, ortanca değeri 21.35, standart sapması 17.28 olarak hesaplanmıştır. Ayrıca öğrenciler Bilge kunduz uluslararası enformatik ve bilgi işlemsel düşünme görevlerinden en düşük 17 alırken en yüksek 72 puan almışlardır.

Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme Görevlerine Ait Son Test İstatistiksel Analiz Bulguları; Çalışmada öğrencilere son test olarak uygulanan 15 soruluk Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme görevlerinin çarpıklık değeri 0.348, basıklık değeri - 0.41 olarak bulunmuştur. Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme Testinden alınan puanların çarpıklık ve basıklık değerleri +1.5 ve -1.5 aralığında olduğundan ve çalışma grubunu oluşturan öğrencilerin sayısının 30'dan fazla olması normal dağılım olduğunu göstermektedir (Tabachnick ve Fidell, 2012).

Tablo 4.6. Bilge kunduz uluslararası enformatik ve bilgi işlemsel düşünme görevleri son testinin istatistik sonuçları.

Son test	N	$\bar{X}$	Medyan	S	Min.	Max.
Bilge kunduz görevleri	85	94.30	91	14.27	87	128

Tablo 4.6.'da görüldüğü üzere, çalışma sonunda öğrencilere uygulanan son test puanlarının ortalaması 94.30, ortanca değeri 91, standart sapması 14.27 olarak hesaplanmıştır. Öğrencilerin Bilge kunduz uluslararası enformatik ve bilgi işlemsel düşünme görevleri son testinden minimum 87 puan ve maksimum 128 puan aldıkları hesaplanmıştır.

Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme Görevine Ait Ön Test ve Son Test İstatistiksel Analiz Bulguları; Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme görevinden alınan puanların ön test ve son testleri arasındaki ilişkiyi değerlendirebilmek için tek örneklem T-testi yapılmıştır. Öğrenciler Bilge kunduz uluslararası enformatik ve bilgi işlemsel düşünme görevlerinde ön testten 19.41 puan ortalama alırken son testten 94.30 puan aldıkları hesaplanmıştır. Ön test ve son test istatistik sonuçlarında $p=0.035$ anlamlılık düzeyinin $p<0.05$ küçük olduğu görülmüştür. Anlamlılık düzeyinin farklı olması sebebiyle ve ortalamalarda görülen farklılıklar fiziksel programlama destekli metin tabanlı programlama eğitimi alan öğrencilerin bilgi işlemsel düşünme becerilerinde istatistiksel olarak anlamlı bir artış olduğunu göstermektedir. Test sonucunda hesaplanmış olan etki büyüklüğü ön test ile son test arasındaki farkın yüksek 1.39 olduğunu göstermektedir. Etki büyüklüğü 0.2 ise az, 0.5 ise orta, 0.8 ise büyük ve 1 değeri üzerinde ise çok büyük düzeyde etkisi olduğu ifade edilir (Green ve Salkind, 2014; Tabachnick ve Fidell, 2012).

Tablo 4.7. Bilge kunduz uluslararası enformatik ve bilgi işlemsel düşünme testi ön test ve son test tek örneklem t-testi sonuçları.

	$\bar{X}$	S	sd	p	t	d
Bilge kunduz ön test	19.41	17.28				
Bilge kunduz son test	94.30	14.87	26	.035	-7.65	1.39

Tablo 4.7’de paylaşılan sonuçlara göre; bu sonuçlara göre Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme görevinden alınan ön test ve son test puanlarının arasındaki farkın anlamlı olduğu söylenebilir. Öğrencilerin yapılan çalışmadan önceki bilgi işlemsel düşünme becerileri ile çalışma sonrası becerileri arasında puan artışı görülmektedir. Buna göre çalışma kapsamında yapılan fiziksel programlama aracı destekli metin tabanlı programlama Python dili öğretiminin öğrencilerin bilgi işlemsel düşünme becerilerine yönelik anlamlı yönde pozitif bir etkisi olduğu söylenebilir.

4.1.3. Uygulama sürecinde ve sonrası öğrencilerin fiziksel programlama aracı destekli metin tabanlı programlama kavram ve süreçlerine yönelik öğrenme düzeylerine ilişkin bulgular;

Uygulama sürecinde ve sonrasında öğrencilerin programlama kavramlarını öğrenme düzeylerine yönelik anlamlı bir fark olup olmadığını ölçmek için araştırmacı tarafından hazırlanmış farklı ölçme araç ve yöntemleri kullanılmıştır. Uygulama sürecinde ders esnasında öğrencileri gözlemleyerek “Ders Gözlem Formu” doldurulmuştur. Haftanın son dersinin sonunda öğrencilere “Haftalık Değerlendirme” soruları uygulanmıştır. Uygulama süreci sonunda yine araştırmacı tarafından hazırlanmış olan programlama aracı, kavram ve süreçlerine yönelik kazanım testi uygulanmıştır. Bu ölçme araçları ile ilgili bulgulara bu bölümde yer verilmiştir.

Ders Gözlem Formuna İlişkin Bulgular; Gözlem formu hem fiziksel programlama aracı kullanımına ilişkin temel bilgileri ve yeterlikleri ölçmek hem de metin tabanlı programlama ile kod yazabilme, karşılaştığı sorunu çözebilme, hata ayıklama ve ürün ortaya koyabilme gibi kriterleri içermektedir. Uygulama sürecinde o derste ilgili kriterlerde zorlanan öğrenciler ile birebir ilgilenilmeye çalışılmıştır. Ders dışı süreçte zaman ayırarak ve ders sırasında yanlış öğrenme ya da öğrenememe durumları giderilmeye çalışılmıştır. Haftalık değerlendirme sorularına verilen cevaplarda olduğu gibi bir sonraki ders tekrarlar bu kriterler temel alınarak yapılmış ve verilen ödevler bunlarla ilişkilendirilmiştir.

Haftalık Değerlendirilmelere İlişkin Bulgular; Araştırmacı tarafından hazırlanmış haftalık değerlendirme soruları farklı soru türlerinden oluşmaktadır. Tablo 4.8’de görüldüğü gibi her haftanın analizi yapılmıştır.

Tablo 4.8. Haftalık Değerlendirme Analizi

Hafta	Öğrenci		S	Medyan	Min.	Max.
	cevaplama (%)	$\bar{X}$				
1	79.86	12	14.86	10	4	15
2	91.48	14	16.03	16	2	18
3	81.92	10	13.39	8	0	12
4	94.90	32	15.2	34	0	36
5	81.22	19	14.34	16	0	21
6	98.13	7	15.27	8	0	9

1. Haftanın Değerlendirilme Bulguları: Öğrencilerin ilk hafta ders sonrası kendilerini değerlendirme sonuçlarına bakıldığında; Python programlama dilinin ne olduğunu söyleyebilecek, fiziksel programlama aracının ne amaçla kullanıldığını, bileşenlerinin ne olduğu ne işe yaradıklarını ve fiziksel programlama aracının bilgisayara nasıl bağlandığını kendi başlarına yapabilecek yeterlikte hissettikleri söylenebilir. Değerlendirmede katılıyorum 3 puan, ne katılıyorum ne katılmıyorum 2 puan ve katılmıyorum 0 puan şeklinde değerlendirilmiştir. Öğrencilerin beş soruya vermiş oldukları katılıyorum cevaplarının ortalaması 79.86'dır. Öğrencilerin Python programlama dilinin tanımını yapabilirim diyenlerin % 87.5'i ve fiziksel programlama aracı ile bilgisayar bağlantısını tek başıma yapabilirim diyenlerin % 82.4'ü olumlu cevap vermişlerdir. Öğrencilerin $\frac{3}{4}$ 'ü yani yarısından fazlasının ilk haftadan itibaren fiziksel programlama aracına karşı olumlu bakış açısı geliştirdikleri hem aracı hem de programlamayı öğrenmeye başladıklarını düşündükleri söylenebilir. Fiziksel programlama aracı bileşenlerini bilme ve ne işe yaradıklarını açıklayabilme maddesini tüm katılımcıların $\frac{1}{4}$ 'ü ne katılıyorum/ ne katılmıyorum olarak işaretlemişlerdir. Bu sonuca göre bir sonraki hafta fiziksel programlama aracının bileşenlerinin tekrar edilmesi ders planına eklenmiştir.

2. Haftanın Değerlendirilme Bulguları: Bir önceki hafta değerlendirmesine göre birinci dersin ilk kısmı fiziksel programlama aracının bileşenlerini hatırlamak ve tekrar etmek üzerine planlanmış ve plan uygulanmıştır. Haftalık derslerin bitiminde öğrencilerin fiziksel programlama aracı ile bilgisayar bağlantısını tek başlarına doğru şekilde takabilme özgüveninin geliştiği ve Python editörünü tek başlarına açmayı öğrendikleri söylenebilir.

Fiziksel programlama aracı ile Python editöründe kodlama yaparak araç üzerindeki ekranda metin kaydırma, istenen bir görüntüyü yayınlama ve döngü kurarak tekrarlı programlar yapabilecek özgüven de oldukları ifade edilebilir. Öğrenciler tarafından dört soruya verilen cevaplar içerisinde olumlu cevapların yüzdeleri minimum % 83.3 ve maksimum % 94.1 aralığında değişmektedir.

3. Haftanın Değerlendirilme Bulguları: Bu hafta programlamada öğrenilen “Listeler” konusunda öğrencilerin olumlu davranış geliştirdikleri gözlemlenmiştir. Liste oluşturulurken programlama dilinde syntax yani yazım kurallarının büyük oranda doğru öğrenildiği söylenebilir. Fiziksel programlama aracı üzerinde yer alan input/output kavramlarını bilerek butonları aktif kullanabilme yeterliliğine sahip olduklarını düşündükleri söylenebilir. Fiziksel programlama aracı üzerinde yer alan pinler hakkında bilgi sahibi oldukları ve kullanabildikleri söylenebilir. İlk iki hafta boyunca sorular içerisinde yer alan 3’lü likert tipi sorulara öğrencilerin yaklaşımının olumsuz olması ve kafalarının karıştığını cevap vermekte zorlandıklarının ifade edilmesi üzerine sorular içerisinden likert tipi sorular çıkartılmıştır. Böylece öğrencilerin kendilerini değerlendirirken olumsuz düşüncelerinin önüne geçilmeye çalışılmıştır. Öğrenciler tarafından dört soruya verilen cevaplar içerisinde olumlu cevapların yüzdeleri minimum % 72.2 ve maksimum % 94.4 aralığında değişmektedir.

4. Haftanın Değerlendirilme Bulguları: Öğrencilerin fiziksel programlama aracı üzerinde yer alan butonların nasıl kullanılması gerektiğini tam olarak anladıkları ve kendilerini yeterli hissettikleri söylenebilir. Öğrencilerin ilgili hafta sonunda büyük çoğunluğunun döngü kullanmayı bildiğini ve mantıksal operatör kavramlarını öğrendiklerini düşündükleri söylenebilir. Ancak buna karşın random komutlarından “.randint” ve “.randrange” kavramlarında kendilerini yeterli hissetmedikleri de söylenebilir. Araştırmacı tarafından bu dikkate alınarak 5. hafta derse başlamadan bu iki kavram üzerinde tekrar yapılmıştır. Öğrencilerin öğrenme eksiklikleri giderilmiştir. Öğrenciler tarafından iki sorunun 12 alt sorusuna verilen cevaplar içerisinde olumlu cevapların yüzdeleri minimum % 38.9 ve maksimum % 100 aralığında değişmektedir.

5. Haftanın Değerlendirilme Bulguları: Öğrencilerin çoğunluğunun ivmeölçerin ne olduğunu ve kodlamada nasıl kullanıldığını konusunda kendilerini yeterli buldukları söylenebilir. Öğrencilerin vermiş oldukları cevaplar ve ders gözleminde dikkati çeken

ivme ölçer kavramını kodlama yaparken “accelerometer” olarak kullandıklarından dolayı Türkçe şekliyle sorulduğunda bazı öğrencilerin anlamamasıdır. Öğrenciler tarafından dört sorunun yedi alt sorusuna verilen cevaplar içerisinde olumlu cevapların yüzdeleri minimum % 72.2 ve maksimum % 100 aralığında değişmektedir.

6. Haftanın Değerlendirilme Bulguları: Fiziksel programlama aracı üzerinde yer alan LED’lerin ölçülerini, koordinatlarını ve parlaklıklarını nasıl kodlayacaklarını öğrendikleri ve kendilerine bu konuda güvendikleri söylenebilir. Öğrenciler tarafından üç soruya verilen cevaplar içerisinde olumlu cevapların yüzdeleri minimum % 94.4 ve maksimum % 100 aralığında değişmektedir.

Öğrenciler altı haftalık ders süreci boyunca her ders sonunda o ders ile ilgili soruları cevaplamaya istek duydukları ve sorulara içtenlikle cevap verdikleri gözlemlenmiştir. Öğrenciler ders süresince motivasyonlarının yüksek olduğunu, fiziksel programlama aracı ve Python diline karşı ilgi duyduklarını dile getirmişlerdir. Öğretim sürecinin aktif ve olumlu davranış değişikliği oluşturduğu söylenebilir.

Programlama Aracı, Kavram ve Süreçlerine Yönelik Kazanım Testine İlişkin Bulgular; Yapılan çalışma sonunda fiziksel programlama aracı destekli metin tabanlı programlama öğretiminin öğrencilerin programlama kavramlarını öğrenme düzeylerine ilişkin veri toplamak amacıyla araştırmacı tarafından hazırlanan 25 soruluk fiziksel programlama aracı ve metin tabanlı programlama dili Python hakkında çoktan seçmeli dört şıklı sorular sorulmuştur. Her soru dört puan olarak hesaplanmıştır. Öğrenciler yapılan sınavda hiçbir soruyu boş bırakmamıştır. Sınav sonuçları incelendiğinde öğrenci ortalaması 88 çıkmıştır. En düşük puan 68, en yüksek puan 100 olarak hesaplanmıştır. Standart sapma 13.72 olarak analiz edilmiştir. Ayrıca kazanım testi sonuçları şu şekildedir; 85 öğrencinin % 56.64’ü (46 öğrenci) 100 puan, % 18.82’si (15 öğrenci) 96 puan, % 10.58’i (9 öğrenci) 92 puan, % 2.35’i 88 puan, % 3.5’u (4 öğrenci) 84 puan, % 4.7’si 80 puan ve % 4.35’i (2 öğrenci) 68 puan almıştır. Programlama öğretimi sonucunda kazanım testinden öğrencilerin aldıkları puanların dağılımı Tablo 4.9’da özetlenmiştir.

Tablo 4.9. Haftalık değerlendirme ve kazanım testi analiz sonuçları.

	$\bar{X}$	Medyan	S	Min.	Max.
1. Hafta	12	10	14.86	4	15
2. Hafta	14	16	12.03	2	18
3. Hafta	10	7	13.39	4	12
4. Hafta	32	34	12.20	3	36
5. Hafta	19	16	14.34	3	21
6. Hafta	8	8	11.27	3	9
Kazanım Testi	88	88	13.72	68	100

Tablo 4.9’da haftalık değerlendirme ve kazanım testi analiz sonuçlarına yer verilmiştir. Buna göre haftalık değerlendirmelerde hiçbir hafta sorulardan toplamda sıfır puan alan öğrenci olmamıştır. Her hafta sınıfın büyük çoğunluğunun ortalamasının üstünde olduğu söylenebilir. Birinci hafta değerlendirildiğinde ortalaması 12, ortanca değeri 10, standart sapması 14.86 olarak hesaplanmıştır. İkinci hafta değerlendirildiğinde ortalaması 14, ortanca değeri 16, standart sapması 12.03 olarak hesaplanmıştır. Üçüncü hafta değerlendirildiğinde ortalaması 10, ortanca değeri 7, standart sapması 13.39 olarak hesaplanmıştır. Dördüncü hafta değerlendirildiğinde ortalaması 32, ortanca değeri 34, standart sapması 12.20 olarak hesaplanmıştır. Beşinci hafta değerlendirildiğinde ortalaması 19, ortanca değeri 16, standart sapması 14.34 olarak hesaplanmıştır. Altıncı hafta değerlendirildiğinde ortalaması 8, ortanca değeri 8, standart sapması 11.27 olarak hesaplanmıştır.

Programlama aracı, kavram ve süreçlerine yönelik kazanım testi analiz edildiğinde test puanının ortalaması 88, ortanca değeri 92, standart sapması 13.72 olarak hesaplanmıştır. Öğrencilerin testlerden almış oldukları frekans, ortalama ve ortanca puanlarına göre fiziksel programlama aracı destekli metin tabanlı programlama Python dili öğretiminin öğrencilerin programlama kavramlarını öğrenme düzeylerinde anlamlı ve pozitif yönde bir gelişim gösterdikleri çıkarımı yapılabilir.

4.1.4. Fiziksel programlama aracı destekli metin tabanlı programlama öğretimine yönelik öğrencileri görüşlerine ilişkin bulgular;

Araştırmanın sonunda verileri desteklemek amacıyla ve metin tabanlı programlama öğretiminin etkisine yönelik öğrenciler ile odak grup görüşmesi yapılmıştır. Odak grup görüşmesinde öğrenci grubuna yöneltilen sorular ve öğrencilerin verdikleri yanıtlar frekans değerleri ile belirtilmiştir.

1. Öğrencilerin fiziksel programlama aracı ile metin tabanlı programlama öğrenmelerine ilişkin görüşleri; Öğrenci görüşleri incelendiğinde, öğrencilerin programlama öğrenirken fiziksel programlama aracının öğrenme sürecini kolaylaştırdığı görüşünü benimsedikleri anlaşılmaktadır. Ayrıca öğrencilerin tamamı fiziksel programlama aracı kullanmanın programlama öğrenimini kolaylaştırdığını ifade etmişlerdir. Özellikle daha önce öğrenmiş oldukları metin tabanlı Small Basic ve blok tabanlı Scratch ile Minecraft’la karşılaştırdıklarında Python programlama dili zor olsa da döngü ve liste kavramlarını daha iyi anladıklarını ifade etmişlerdir. Öğrenciler başarıma duygusunu yaşadıklarını ifade ettiler. Buna bağlı olarak program yazmakta özgüvenlerinin geliştiği gözlemlenmiştir. Öğrenciler hem derse hem de kodlamaya karşı daha olumlu bir bakış açısı geliştirmiştir denebilir.

Tablo 4.10. Öğrencilerin fiziksel programlama aracı ile metin tabanlı programlama öğrenmelerine ilişkin görüşleri.

Kod	Kodlanma sıklığı (f)
Motive edici olması	12
Öğretici olması	11
Anlamayı kolaylaştırması	10
Döngü kavramını öğrenmeyi kolaylaştırması	10
Liste kavramını öğrenmeyi kolaylaştırması	7
Programlama kavramlarını öğrenmede zorlaştırması	1

Bazı öğrenci görüşleri;

Ö1: “Micro:bit kullanmayı çok sevdim. İlk defa öğrendiğim Python öğrenmemi kolaylaştırdı. Döngüler ne demek önceki yıllarda gördüğümüz Small Basic ve Minecraft’ta anlamamıştım ama artık anladım. Çok kolay bir konuymuş.”

Ö2: “Daha önce arduino kullanmıştım. Ama micro:bit daha zevkli geldi. Hem küçük hem de birçok özellik var içinde.”

Ö3: “Hem programı yazmak hem de micro:bit’e yükleyip denemek zaman kaybı oldu.”

Ö7: “Daha önceki sınıfta small basic öğrenmiştim. Çok sıkıcı ve zor gelmişti. Python öğreneceğimizi duyunca korkmuştum ama micro:bit sayesinde yazdığım kodları daha rahat anladım.”

Ö10: “Python kodlarını yazın gittiğim bir kursta yazmaya başlamıştım. Ancak listeleme ve döngülerde zorlanmış anlamamıştım. Bu derste micro:bit ile daha rahat anladım. Kendimi yazılımcı gibi hissediyorum yaptığım kodları micro:bit ile uyguladığımda.”

Ö11: “Kodlamadan pek anlamıyordum. Her zaman ne yazdığımı anlamak zor oluyordu. Python ile kodlama yapamam diyordum ancak micro:bit sayesinde adım adım ne yaptığımı anladım. Şimdi çok eğlenerek kodlama yapıyorum.”

2. Öğrencilerin fiziksel programlama aracı ile metin tabanlı programlama öğrenmenin iyi yönlerine ilişkin görüşleri; Öğrenci görüşleri incelendiğinde, öğrencilerin programlama öğrenirken fiziksel programlama aracı ile eğlendikleri ve süreç içerisinde hareket kolaylığı sağlaması nedeniyle farklı ve olumlu görüş benimsedikleri anlaşılmaktadır. Buna ek olarak öğrencilerin büyük çoğunluğu yazmış oldukları kodların işe yarar bir projeye dönüştüğünü görmekten mutlu olduklarını ifade etmişlerdir. Çünkü yazdıkları kodların somut olarak çalışır halini görmenin onlara zevk verdiğini söylemişlerdir.

Tablo 4.11. Öğrencilerin fiziksel programlama aracı ile metin tabanlı programlama öğrenmenin iyi yönlerine ilişkin görüşleri

Kod	Kodlanma sıklığı (f)
Yazılan programların somut sonuçlar ve anlık dönüt	10
Motive edici	9
Kullanım kolaylığı	9

Sınıf içerisinde hareket etmeyi sağlaması ve iş birlikçi çalışmayı sağlaması	7
Farklı sınıf ortamı sağlaması	6
Kararsızım	1

Bazı öğrenci görüşleri;

Ö4: “Çok eğlenceliydi. Yazdığım kodların çalıştığını anında gördüm. Bu ne yaptığımı daha iyi anlamamı sağladı.”

Ö6: “Python programlama çok zor bir dil daha önce çok öğrenememiştim. Ancak micro:bit ile yazdığım kodları çalıştırmak ve işe yarar projeler oluşturmak çok eğlenceliydi. Kendimi mutlu hissettim. Adım sayar yapmak, ph değeri ölçmek ya da ortam sıcaklığı ölçmek çok güzeldi.”

Ö7: “Derste yazdığım kodları arkadaşlarımla denemek ve hareket etmek zevkliydi.”

Ö10: “Arkadaşlarla yaptığımız oyunları elimize alarak birlikte oynamamız çok güzeldi.”

Ö12: “Daha önce fiziksel programlama aracı kullanmamıştım. Farklı bir ders oldu. Program yazarken sıkılmadım.”

3. Öğrencilerin fiziksel programlama aracı ile metin tabanlı programlama öğrenmenin zorlayıcı yönlerine ilişkin görüşleri; Öğrenci görüşleri üçüncü maddeye göre incelendiğinde, öğrencilerin programlama öğrenirken fiziksel programlama aracı ile negatif bir duygu geliştirmedikleri tam tersine onları motive ettikleri anlaşılmaktadır. İki öğrenci dışında geri kalan 10 öğrencinin görüşlerinde olumsuz bir durumla karşılaşmadıkları ve negatif bir ifade yer almamaktadır.

Bazı öğrenci görüşleri;

Ö3: “Bilgisayarda kod yazıp onu yüklemek ve oradan bakmak yorucu geldi. Micro:bit kullanmak yorucuydu.”

Ö4: “Ben zorlanmadım. Micro:bit ile eğlendim derslerde Python’ da kod yazmak belki biraz zor çünkü yeni öğreniyorum ama yazdığım programların sonucunu hemen görebiliyorum. Yanlış yazdığımda çalışmıyor ve hemen kodlarda sorunu bulmaya çalışıyorum.”

Ö5: “Micro:bit kullanmaktan zorlanmadım. Ama (accelerometer) bileşenini ve Python’ da ki listeleme konusunu bazı şeyleri anlamakta zorlandım!”

Ö12: “Micro:bit’i öğrenmek ve kullanmak zor olmadı. Tam tersine kredi kartı gibi bir aracın içinde bu kadar özellik olduğunu öğrenmek şaşırtıcıydı. Ben Python öğrenmenin kolay olduğunu düşünüyorum belki de micro:bit ile öğrendiğim için. ”

4. Öğrencilerin fiziksel programlama aracı ile metin tabanlı programlama öğrenme sürecine yönelik önerilere ilişkin görüşleri; Öğrenci görüşleri dördüncü maddeye göre incelendiğinde, öğrencilerin programlama öğrenirken fiziksel programlama aracına karşı olumlu görüşe sahip olanların daha çok etkinlik çözerek micro:bit ile daha fazla zaman geçirmek istedikleri anlaşılmaktadır. Ancak buna karşın bir öğrenci zamanı etkin kullanmakta sıkıntı yaşadığını ifade etmiştir.

Tablo 4.12. Öğrencilerin fiziksel programlama aracı ile metin tabanlı programlama öğrenme sürecine yönelik önerilere ilişkin görüşleri

Kod	Kodlanma sıklığı (f)
Daha çok proje geliştirmek	11
Daha fazla programlama becerisini geliştirici etkinlikler yapmak	10
Fiziksel programlama aracı ile daha çok zaman geçirmek	7
Daha çok etkinlik yapmak	6
Zaman yönetiminde zorlanma	1

Bazı öğrenci görüşleri;

Ö7: “Daha çok etkinlik yapmak isterdim. Kod çok fazla yazınca daha iyi anlıyorum mantığını. ”

Ö8: “Micro:bit kullanarak bir şey geliştirmek isterdim. Yaptığım projelerin çalışmasını görmeyi çok seviyorum. Daha çok çalışmak için bu beni kamçılıyor.”

Ö9: “Micro:bit ile çok eğlendim daha çok ders yapabilmek isterdim. Oyun gibiydi dersler.”

Ö11: “Bazen derslerde etkinlikleri yetiştirmekte zorlandım. Yazım hataları yaptığım oldu o zamanlar kodu nasıl çalıştıracam diye çalışmaktan Micro:bit’e yüklemeye vaktim zor yetiyordu. ”

5. Öğrencilerin fiziksel programlama aracı ile metin tabanlı programlama öğrenme ile ilgili diğer düşüncelerine ilişkin görüşleri; Öğrenci görüşleri beşinci maddeye göre incelendiğinde, öğrencilerin programlama öğrenirken fiziksel programlama aracını ilgi çekici buldukları ve bu dersi bir daha almak istediklerini ifade ettiler. İçlerinden bazıları evlerine kullanmış oldukları fiziksel programlama aracını alacaklarını ve çalışmalarına evde de devam edeceklerini belirtmiştir.

Tablo 4.13. Öğrencilerin fiziksel programlama aracı ile metin tabanlı programlama öğrenme ile ilgili diğer düşüncelerine ilişkin görüşleri

Kod	Kodlanma sıklığı (f)
Fiziksel programlama aracı ile metin tabanlı programlama öğrenmek ilgi çekici	12
Başka derslerde de programlama yaparken fiziksel programlama aracı kullanmak isterim	12
Yarışmaya katılmak isterim	5
Fiziksel programlama aracını farklı programlama platformlarında kullanmak isterim	9

Bazı öğrenci görüşleri;

Ö1: “Hem Python hem de micro:bit ile çalışmak ilgimi çok çekti. Çok sevdim.”

Ö3: “Okulda diğer programların yarışması yapılıyor seneye bununda olmasını isterim.”

Ö5: “Seçmeli ders seneye de seçeceğim yine micro:bit ile Python yapmak isterim.”

Ö4: “Döngüler, Accelorometer gibi kavramları öğrendiğim için mutluyum. Ayrıca butonları kullanmak çok eğlenceliydi. Çok güzel kendi kendime oyunlar yaptım. Aileme söyledim evde de micro:bit ile programlamaya devam edeceğim.”

Ö6: “Python programlama dilinin zor olduğunu duymuştum. Ama şaşırdım ben bu kadar kolay ve eğlenceli olacağını düşünmemiştim.”

Ö8: “Evde araştırdım micro:bit ile pek çok programlama kullanılıyormuş. Onları da öğreneceğim.”

Ö11: “Micro:bit ile yarışmalara katılabiliyormuşuz. Bende katılacağım.”

Ö12: “Kredi kartı gibi bir şeyin içinde bu kadar özellik olduğunu düşünmezdim.”

Sonuç olarak, odak grup görüşmesine katılan öğrencilerin hemen hemen hepsinin dersten keyif aldıkları, dersin devamını isteyerek olumlu düşünmeye başladıkları söylenebilir. Öğrencilerin ifadelerinden soyut olan programlama dili Python ile yazdıkları kodların somut ve çalışır hallerini görmekten mutlu oldukları sonucu çıkmıştır. Aynı ortamda birlikte çalıştıkları arkadaşlarıyla yazmış oldukları programları oynama fırsatı bulmaları kendilerine güvenlerini arttırmıştır.

Öğrenciler uygulama sürecinde fiziksel programlama aracı ile çalışırken derse ve programlaya karşı motive oldukları gözlemlenmiştir. Ayrıca süreç içerisinde hem ders içi etkinlikleri yaparken hem de proje üzerinde çalışırken arkadaşları ile çalışmaktan keyif aldıklarını belirttiler. Ayrıca deneme yanılma yoluyla program üzerinde çalışmanın programlama kavramlarını anlamalarında daha etkili olduğunu ifade ettiler. Alanyazınına bakıldığında Sullivan ve Heffernan (2016) öğrencilere etkinlikler sırasında deneme yanılma yapabilecekleri etkinlikler düzenlenmesi gerektiğini çünkü bu şekilde elde ettikleri bilgilerin denemelerin ötesine geçerek daha kapsamlı problemleri çözebilecek düzeye ulaşabileceklerini ifade etmişlerdir. Bu nedenle çalışma sırasında zaman zaman deneme yanılma yapmaları için öğrencilere fırsat verilmiştir. Bu deneme yanılmalar üzerinden çıkan hatalar üzerine sınıf ortamında fikir alışverişi ve beyin fırtınaları yapılmış olması öğrencilerin daha farklı programlar yazmalarına ve fiziksel programlama aracıyla farklı etkinlikler üzerine düşünmelerine katkı sağladığı düşünülmektedir.

4.2. Tartışma

Bu çalışmanın amacı, ortaokul seçmeli Bilişim Teknolojileri ve Yazılım dersi alan 7. ve 8. sınıf öğrencilerinin fiziksel programlama aracı destekli metin tabanlı programlama öğrenmelerinin çeşitli değişkenler açısından incelenerek etkisini araştırmaktır. Çalışmada fiziksel programlama araçlarından biri olan micro:bit ile metin tabanlı programlama Python dili öğretiminin öğrencilerin bilgi işlemsel düşünme becerilerine etkisini ölçmek için uygulama öncesi ve sonrası bilgi işlemsel düşünme becerilerine yönelik öz yeterlik algıları ölçülmüştür ve bilgi işlemsel düşünmeye yönelik test uygulanmıştır. Uygulama sürecinde ders sırasında öğrenciler gözlemlenerek öğretmen tarafından gözlem formu doldurulmuş ve ders sonu değerlendirme uygulanmıştır. Uygulama sonrasında ise, öğrencilere programlama kavramları ve fiziksel programlama aracı hakkında programlama

aracı, kavram ve süreçlerine yönelik kazanım testi uygulanmıştır ayrıca öğrencilerden seçilen grup ile uygulama hakkında görüşlerini almak amacıyla odak grup görüşmeleri yapılmıştır.

Araştırmanın birinci alt problemi kapsamında; öğrencilerin uygulama öncesinde bilgi işlemsel düşünme becerisi öz yeterlik algılarının uygulama sonrası bilgi işlemsel düşünme becerisi öz yeterlik algıları arasında istatistiksel olarak anlamlı bir fark tespit edilmiştir. Öğrencilerin yarısından fazlası uygulama öncesi ön testten minimum puana yakın puan alırken, son test ile öğrencilerin yarısından fazlasının puanlarının maksimum puana yaklaştığı görülmektedir. Bu sonuca göre uygulama sonrası öğrencilerin kendilerini algoritma tasarlama, problem çözme, veri işleme, temel programlama ve özgüven konusunda daha yeterli hissettikleri söylenebilir. Buna göre fiziksel programlama aracı destekli metin tabanlı programlama öğretimi alan öğrencilerin, bilgi işlemsel düşünme becerilerine yönelik öz yeterlik algılarını artırmıştır denilebilir.

Elde edilen çalışma sonuçları alanyazında yapılan çalışmalar ile benzerlik göstermektedir. Bilgi işlemsel düşünme ve ilgili becerilerin geliştirilmesi adına kullanılan robotik programlama araçlarının etkileri üzerine yapılan çalışmalarda olumlu sonuçlar aktarmaktadırlar (Karaahmetoğlu ve Korkmaz, 2019; Vieira, Penmetcha, Magana ve Matson, 2016); Berland ve Wilensky, 2015; Bers, 2008; Bers vd., 2014; Chaudhary vd., 2016). Özel (2019) robotik programlama öğretimi yaptığı çalışmasının sonucunda metin tabanlı programlama eğitiminin öğrencilerin bilgi işlemsel düşünme becerisi öz yeterlik algısına olumlu etkisi olduğunu belirtmiştir. Ramazanoğlu'nun (2021) çalışmasında robotik kodlama uygulamalarının ortaokul öğrencilerinin bilgisayara yönelik tutumlarını ve bilgi işlemsel düşünme becerilerine yönelik öz yeterlik algıları üzerindeki etkilerini incelemiş ve sonucunda kaygılarının azalmış olduğunu, bilgi işlemsel düşünme becerilerine yönelik öz yeterlik algılarının ise yükseldiğini belirtmiştir. Öztürk (2021) çalışmasının sonucunda otantik programlama etkinliklerinin yaratıcılık, algoritmik düşünme, eleştirel düşünme ve bilgi işlemsel düşünme genel düzeylerini anlamlı şekilde geliştirebildiğini ifade etmiştir. Kukul (2018), çalışmasında bilgi işlemsel düşünme becerilerinde öğrencilerinin tamamında ölçekten anlamlı bir farklılık olduğu tespit etmiş ve buna bağlı olarak programlama öğretiminin, bireylerin soyutlama ve ayırtırma öz yeterliklerini olumlu etkilediğini söylemektedir. Sırakaya (2019) ise çalışmasında farklı öğrenciler arasında bilgi işlemsel düşünme düzeyinin programlama öğretiminden sonra

programlama beceri düzeyine göre geliştiğini söylemiştir. Ayrıca Şahin ve Arslan (2017) çalışmalarında, 6 -12 yaş aralığında kodlama eğitimi almış öğrenciler ile ilgili literatür taramasında ulusal/uluslararası seviyede yapılmış olan farklı metot ve sonuçları incelenmiş ancak bu çalışmaların birçoğunun robot kitleri kullanımının çocukların fen derslerine olan katkılarının ve memnuniyetlerinin arttırdığı bu nedenle bu fiziksel araçların programlama öğretiminde kullanılmasının araştırılması gerektiğini ayrıca analizinin ise sınırlı düzeyde kaldığını belirtmiştir. Benzer bir programlama eğitimi uygulanıp, bilgi işlemsel düşünme becerilerine yönelik öz yeterlik algısında olumlu etki gözlemlediklerini ifade etmişlerdir (Feldhausen ve ark., 2018; Paf, 2019). Ancak bu çalışmalara karşın alanyazında karşılaşılan farklı sonuçlar da vardır. Örneğin, Yolcu (2018) deney ve kontrol gruplu gerçekleştirdiği çalışmasının sonucunda; her iki grupta bilgi işlemsel düşünme beceri ve algı ölçeklerinde puanların arttığını fakat uygulama sonrası son test puanlarını karşılaştırdığında iki grup arasında anlamlı pozitif yönde bir fark oluşmadığını ifade etmiştir. Benzer şekilde deneysel çalışma yapan Djambong ve Freiman (2016) yine grupların bilgi işlemsel düşünme beceri puanlarında artış olduğunu ancak anlamlı bir fark gözlenmediği sonucuna ulaşmışlardır. Bu sonuçlar göre programlama öğretilirken fiziksel programlama araçları kullanmak yazılan programın somut olarak çıktısını görebilmek öğrencilerin bilgi işlemsel düşünme becerilerini diğer yöntemlere göre daha fazla geliştirmektedir denilebilir. Kılıç (2022) çalışmasının sonucunda öğrencilerin bilgi işlemsel düşünme becerisi gelişimlerini robotik programlamanın önemli ölçüde etkilediği sonucuna ulaşmıştır. Sonuç olarak fiziksel programlama aracı destekli metin tabanlı programlama Python dili öğretimi alan öğrencilerin bilgi işlemsel düşünme becerisi öz yeterlik algılarının geliştiği söylenebilir.

Araştırmanın ikinci alt problemi kapsamında, öğrencilerin uygulama öncesi bilgi işlemsel düşünme becerileri ve uygulama sonrası bilgi işlemsel düşünme becerileri arasındaki fark, Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme görevleri ile değerlendirilmiştir. Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme görevlerinden oluşan ön test ve son test sonuçları arasında anlamlı bir fark ortaya çıkmıştır. Buna göre fiziksel programlama aracı ile metin tabanlı programlama Python dili öğretimi alan öğrencilerin bilgi işlemsel düşünme becerilerinin geliştiği söylenebilir.

Yapılan çalışma sonuçları alanyazında yapılan çalışmalar ile benzerlik göstermektedir. Örneğin; Sırakaya (2019) çalışmasında elde ettiği sonuçlardan biri

programlama öğretiminin bilgi işlemsel düşünmenin algoritma düşünme alt boyutu üzerinde etkisi olduğudur. Berikan (2018) ise çalışmasında, ortaokul öğrencilerinin bilgi işlemsel düşünme becerisini geliştirmek amacıyla çalışmasında öğrencilerin bilgi işlemsel düşünme becerisi için önemli görülen stratejilerin problem çözme sürecinde etkin biçimde kullandıkları sonucuna varmıştır. Kukul (2018) çalışmasının sonucunda kontrol grubundaki öğrencilerin, öz yeterlik düzeylerinde ve bilgi işlemsel düşünme görev testinden alınan puanlarında anlamlı bir artış olduğunu ifade etmiştir. Erdem (2018) ortaokul öğrencileri ile yapılan kodlama öğretiminde öğrencilerin bilgi işlemsel düşünme becerilerine etkisi araştırdığı çalışmasında; yüz yüze eğitim ve ters yüz sınıf modeli ile eğitimin, öğrencilerin bilgi işlemsel düşünme becerilerinde anlamlı olmayan bir farklılık oluşturduğu ve kodlama eğitiminin diğer becerilere ilişkin öz yeterliklere olumlu bir etkisinin olduğunu ifade etmektedir. Sarıtepeci (2017) ise çalışmasında katılımcıların orta ile üst düzeyde bilgi işlemsel düşünme becerisine sahip olduklarını ve problem çözme becerisinin bilgi işlemsel düşünme becerisinin önemli bir belirleyicisi olduğu sonucuna ulaşmıştır. 2016 yılında yaptıkları çalışmalarında Barut, Tuğtekin ve Kuzu, kodlama eğitiminde bilgi işlemsel düşünme becerilerinin önemini ortaya çıkarmak amacıyla bilgi işlemsel düşünme becerileri ile programlama becerilerini karşılaştırmışlardır. Sonuçta hedeflerin benzer olduğunu, kodlama eğitimi ile bilgi işlemsel düşünme becerilerinin kazandırılabilceğini ifade etmişlerdir. Özet olarak pek çok araştırmacı bilgi işlemsel düşünme yeterliliğinin geliştirilebilir bir beceri olduğunu ifade eden çalışmalar ortaya koymuşlardır.

Araştırmanın üçüncü alt problemi kapsamında, öğrencilerin fiziksel programlama aracı destekli metin tabanlı programlama öğretimi ile programlama kavramlarını öğrenmelerinin ne düzeyde olduğu incelenmek istenmiştir. Uygulama sonrasında öğrencilerin ders sonu değerlendirmesinde hem fiziksel programlama aracı kullanımını hem de metin tabanlı programlama Python dili kavramlarına yönelik beklenen ve istenen kavramsal öğrenmelerin gerçekleştiği görülmektedir. Uygulama sürecinde her derste öğretmen tarafından doldurulan gözlem formu, her hafta ders bitiminde öğrencilerin doldurmuş oldukları haftalık değerlendirme sonuçları, bazı haftalar öğrencilere tanımlanan ödevler, öğrencilere uygulanan kazanım testi değerlendirildiğinde öğrencilerin çoğunluğunun bunları başarı ile gerçekleştirdiği ve yüksek puanlar aldıkları gözlemlenmiştir. Ayrıca ödevlerini başarı ile gerçekleştirmeleri de programlama becerilerinin geliştiğinin göstergesi olarak düşünülebilir. Öğrenciler hem uygulama

sürecinde hem de sonrasında odak grup görüşmesinde fiziksel programlama aracı ile Python metin tabanlı programlama dili öğrenmekten keyif aldıklarını, buna bağlı olarak programlamaya karşı heveslerinin arttığını ifade etmişlerdir. Bu bulgu odak grup görüşmelerindeki nitel veriler ile de desteklenmektedir.

Elde edilen sonuçlar ile alanyazında yapılan çalışmalar kavramsal öğrenmelere ilişkin benzer ve farklı sonuçlar göstermektedir. Atmatzidou, Demetriadis ve Nika (2018) öğrencilerin robotlarla etkileşime girmesinin temel programlama kavramlarını anlamalarında etkili olduğunu belirtmiştir. Saygıner (2017) çalışmasında metin tabanlı programlama eğitimi alan öğrencilerin; ön test ve son test puan ortalamaları karşılaştırıldığında anlamlı bir artış yaşandığı sonucuna ulaşmıştır. Barut, Tuğtekin ve Kuzu (2016) çalışmalarında programlama öğretiminin kazandırdığı kavramlar sayesinde öğrencilerin bilgi işlemsel düşünme becerilerinin de geliştirilebileceğini ifade etmişlerdir. En fazla çalışmanın programlama öğretimi ve robotik kodlama alanlarında yoğunlaştığını ifade etmişlerdir (Yang, Liu ve Chen, 2020). Özer (2019) yapmış olduğu alanyazını taramasına göre robotik etkinliklerinin programlama öğretiminde kolaylıklar sağladığı, problem çözme, ilgi ve motivasyon üzerinde olumlu etki oluşturduğunu ifade etmiştir. Şahiner ve Kert (2016), araştırmalarında sonuç olarak; alt becerilerin bir bölümünün teknik olarak programlama becerilerini içerdiği ve değişkenler, diziler, döngüler, nesne temelli ve hata ayıklamayı bazı temel programlama kavramlarını içerdiğini vurgulamışlardır. Baltalı (2016) ise araştırma sonucunda programlama öğretiminde görsel araç kullanımının ilköğretim öğrencileri (ortaokul kademesi dahil) için etkili olduğunu belirtmiştir. Benzer bir çalışma yapan Gülmez' de (2009) aynı şekilde programlama öğretiminde görselleştirme araçlarını kullanmanın pozitif etkisinin olduğunu söylemiştir. Buna karşın Armoni, Meerbaum-Salant ve Ben-Ari (2015) ise programlama öğrenimine ortaokul düzeyinde blok tabanlı programlama ile başlayıp lise de metin tabanlı programlamaya geçişteki rolünü incelemişlerdir. Blok tabanlı programlamanın süreci kolaylaştırdığını ancak hiç blok tabanlı programlama eğitimi almayan öğrencilere göre başarı düzeyinde anlamlı bir farklılık bulunmadığı sonucuna ulaşılmıştır. Numanoglu ve Keser (2017) programlama öğretiminde robot kullanımı çalışmalarında Mbot aracını kullanarak döngüler, değişkenler, koşul yapıları, listeler, fonksiyonlar-prosedürler ve diziler kavramlarını öğretmeyi amaçlamışlardır. Çalışmalarının sonucunda fiziksel programlama aracı kullanarak programlama öğretiminin olumlu etkisi olduğunu bununda soyut kavramları basit bir şekilde aracın somutlaştırdığını ve öğrencinin yazmış olduğu programın sonuçlarını anlık

görebilmesine bağlamışlardır. Çalışmalarında Romero ve Dupont (2016), eğitsel robot kitlerinin eğitimde kullanılmasıyla eleştirel düşünme, birlikte çalışma, yaratıcılık, problem çözme bilgi işlemsel düşünme becerilerinin geliştiğini ifade etmektedirler. Akçay ve Çoklar (2016) çalışmalarında programlama becerisinin problem çözme, algoritmik düşünme, sorgulama becerisinin gelişimine katkısı olduğundan bahsetmektedirler. Alanyazında pek çok araştırma sonucunda olduğu gibi yaptıkları çalışmada programlama dili öğrenme problem çözme, yaratıcı düşünme, eleştirel düşünme, işbirlikçi öğrenme gibi düşünsel 21. yüzyıl becerilerin gelişimine olumlu katkı sağladığını saptamışlardır (Lau ve Yuen, 2011; Kaucic ve Asic, 2011; Kim, Chunk ve Yu, 2013). Buna rağmen Yolcu (2018) deney ve kontrol gruplu gerçekleştirdiği çalışmasının sonucunda; her iki grupta bilgi işlemsel düşünme beceri puanları artmış ancak gruplar arası anlamlı bir fark görmediğini; programlama öğretiminin ise bilgi işlemsel düşünme becerilerini arttırdığını ama robotik etkinliklerin gruplar arasında ne bilgi işlem düşünme ne de öz yeterlik ölçeği sonuçlarını etkilemediği sonucuna ulaşmıştır. Alanyazında genellikle erken yaşlarda ya da programlamaya yeni başlayanlar için blok tabanlı programlama çalışmaları ve önerileri gözlemlenmektedir. Özellikle bilgi işlemsel düşünme ve metin tabanlı programlama ile ilgili çalışmaların bu grup için azlığının sebebinin sözdiziminin zor olmasından kaynaklandığı düşünülmektedir. Bu çalışmadaki sonuçlarla Park ve Yoo (2018) ile Dorling ve White'in (2015) çalışmalarında ilkökul ve ortaokul düzeyindeki öğrencilerin metin tabanlı programlama kavramlarını öğrenebilir ve performanslarında başarılı olabilir şeklindeki ifadeler ile uyum göstermektedir. Keith, Sullivan ve Pham (2019) ise çalışmalarında öğrencilerin robotik etkinlikler ile çalışmalarının sonucunda bilgi işlemsel düşünme becerilerinin geliştiğini gözlemlemiştir. Bu sebeple bilgi işlemsel düşünme becerisi geliştirmede robotik etkinlikleri önemli bir araç olarak görmektedirler. Berland ve Wilensky (2015) robotik programlama öğrenme araçlarının öğrencilerin bilgi işlemsel düşünme becerilerinin gelişmesine katkısı olduğunu belirtmişlerdir.

Alanyazın incelendiğinde öğrencilerin fiziksel programlama aracı destekli metin tabanlı programlama öğretimi ile programlama kavramlarını öğrenme düzeylerine ilişkin veriler incelenmiştir. Öğrencilerin fiziksel programlama aracı destekli metin tabanlı programlama ile bilgi işlemsel düşünme becerilerinin geliştiği söylenebilir. Buna karşın son yıllarda çok çalışılan bir kavram olmasına karşın hala sınırlı sayıda çalışma olduğu gözlemlenmiştir. Araştırma sonuçları ve alanyazını bir bütün olarak değerlendirildiğinde fiziksel programlama aracı ile metin tabanlı programlama öğretimi alan öğrencilerin

programlama kavramlarını daha rahat anladıkları söylenebilir. Öğrenciler uygulama sürecinde fiziksel programlama ile çalışırken programlama kavramlarını daha iyi anladıkları ve keyif aldıkları gözlemlenmiştir. Ayrıca süreç içerisinde hem ders içi etkinlikleri yaparken hem de proje üzerinde çalışırken hep birlikte çalışmanın motivasyonlarını arttırdığı ayrıca deneme yanılma yoluyla program üzerinde çalışmanın programlama kavramlarını anlamalarında daha etkili olduğunu ifade ettiler. Alanyazınına bakıldığında Sullivan ve Heffernan (2016) öğrencilere etkinlikler sırasında deneme yanılma yapabilecekleri etkinlikler düzenlenmesi gerektiğini çünkü bu şekilde elde ettikleri bilgilerin denemelerin ötesine geçerek daha kapsamlı problemleri çözebilecek düzeye ulaşabileceklerini ifade etmişlerdir. Bu nedenle çalışma sırasında zaman zaman deneme yanılma yapmaları için öğrencilere fırsat verilmiştir. Bu deneme yanılmalar üzerinden çıkan hatalar üzerine sınıf ortamında ufak tartışmalar ve beyin fırtınaları yapılmış olması öğrencilerin daha farklı program tasarlamak ve farklı fiziksel programlama araçları etkinlikleri üzerine düşünmelerine katkı sağladığı düşünülmektedir. Bu bağlamda Khine (2017) robotik eğitim sırasında takım çalışmasının ve öğrencilerin öğretmenle olan iletişimlerinin önemli olduğunu çünkü öğrencilerin bu şekilde sorunları çözerek en iyi çözümü elde edebileceğini ifade etmiştir. Ayrıca Wing (2008) programlama öğretiminde programlama kavramlarının etkili bir şekilde öğretilmesinde kullanılan programlama araçlarının öğrencilerin programlama kavramlarını anlamalarının önüne geçmesinin önlenmesi üzerinde durmuştur. Bu anlamda süreç içerisinde kullanılan fiziksel programlama aracı micro:bit' in bir oyun aracı olmadığı öğrenciler ile paylaşılmıştır. Uygulama sürecinde ve sonunda toplanan veriler ışığında öğrencilerin programlama öğrenmesi fiziksel programlama aracı destekli metin tabanlı programlama öğretimi ile programlama kavramlarını öğrenme düzeylerinin arttığı söylenebilir. Bunun göstergesi olarak çalışmadaki kazanım testinden öğrencilerin büyük çoğunluğunun yüksek puan alarak donanımsal kavramlar, uygulama ara yüzü, Python programlama dili, veri, değişken, döngü, listeler ve koşul yapıları gibi programlama kavramlarını ve programlama becerilerini öğrendiği söylenebilir. Ayrıca öğrencilerin başarılı olmasındaki önemli bir faktörün de fiziksel programlama aracı kullanarak programlama öğretiminin yapılmış olmasıdır. Bu durumun sebebi ise, ortaokul öğrencilerinin fiziksel programlama aracı ile programlama öğrendikleri sırada uygulamaları yaparken öğrenmelerini destekliyor olmasıdır. Çünkü fiziksel programlama aracı ile Python gibi metin tabanlı bir programlama dilinin soyut programlama kavramlarını somutlaştırarak öğrenmeyi kolaylaştırmaktadır. Bu

sayede öğrencilerin derse ve programlama öğrenmeye karşı olumlu düşüncelerinin arttığı söylenebilir.

Araştırmanın dördüncü alt problemi kapsamında, öğrencilerin uygulanan fiziksel programlama aracı destekli metin tabanlı programlama öğretimine yönelik öğrenci görüşleri alınmıştır. Öğrenciler uygulama sonunda görüşmede; fiziksel programlama aracı ile metin tabanlı programlama öğretimi için tasarlanan ders konularından ve etkinliklerinden çok zevk aldıklarını, motive olduklarını, zor bir programlama dilini konsantre olarak daha kalıcı öğrendiklerini ifade etmişlerdir. Gupta (2004) çalışmasında, öğretmenlerin metin tabanlı programlama dili seçerken hedef kitle ve kullanılacak programlama aracının öğrenme odaklı olmasına dikkat edilmesi gerektiğini vurgulanmaktadır. Bu görüşü destekleyecek şekilde odak grup görüşmesine katılan öğrenciler; Python öğrenmekte zorlanmadıklarını, keyifli vakit geçirdiklerini belirtmişlerdir. Yapılan çalışma göstermektedir ki öğrencilerin fiziksel programlama araçları metin tabanlı programlama öğretimi sonunda programlamanın profesyonel alandaki çalışma imkanlarını araştırdıkları ve hatta programlama ile ilişkili meslek seçeceklerine ilişkin farkındalık kazanmışlardır. Görüşme bulgularında öğrencilerin programlama yapmayı eğlenceli bulduklarını, ayrıca motivasyonlarını olumlu etkilediğini ifade etmişlerdir. Sürecin sonunda derslerin bitmesini istemediğini ifade eden öğrenciler olmuştur. Python programlamayı fiziksel programlama aracı ile daha iyi anladıklarını belirtmişlerdir. Gerekçe olarak da süreç içerisinde ne yaptıklarını fiziksel programlama aracı sayesinde somut olarak görebildiklerini belirttiler. Süreci değerlendirirken “Kendimi artık daha akıllı görüyorum!”, “Kendimi programcı gibi hissediyorum!” diyerek ifade eden öğrenciler olmuştur. Şahin ve Arslan (2017) yaptıkları çalışmada programlama öğretimi hakkında benzer çalışmalar da bulunmuşlardır ve programlama öğretimi sürecinin öğrencilerin motivasyonları üzerinde olumlu etkiye sahip olduğunu belirtmişlerdir. Çalışma sırasında öğrencilerin yazdıkları kodların çalışmasını görmeleri yani bir problemi çözebildiklerini somut bir şekilde görmeleri kendilerine olan öz güvenlerini de arttırdığını belirtmişlerdir. Bunu destekleyen çalışmalar alanyazında mevcuttur (Largo, Peterson ve Chen, 2005). Araştırma sonuçları ve alanyazını bir bütün olarak değerlendirildiğinde fiziksel programlama aracı ile metin tabanlı programlama öğretimi alan öğrencilerin programlama öğretimine yönelik görüşlerinin olumlu yönde olduğu söylenebilir.

BÖLÜM V

SONUÇ ve ÖNERİLER

Bu bölümde ortaokul öğrencilerinin metin tabanlı programlama öğrenimi sürecinde fiziksel programlama aracı kullanarak programlamayı öğrenmelerinin bilgi işlemsel düşünme becerilerine ve programlamaya yönelik deneyimlerine ilişkin görüşlerine yönelik hazırlanan öğretim programına ilişkin sonuç ve önerilere yer verilecektir.

5.1. Sonuç

Seçmeli ders kapsamında ortaokul 7. ve 8. sınıf öğrencilerine fiziksel programlama aracı ile metin tabanlı programlama dili olan Python programlama öğretimi için öğretim programı hazırlanmış ve uygulanmıştır. Bu amaçla araştırmada sırasıyla; ortaokul bilişim teknolojileri ve yazılım dersini seçmiş olan öğrencilerin bilgi işlemsel düşünme becerisine yönelik öz yeterlik algı düzeyleri, bilgi işlemsel düşünme becerileri, öğrencilerin programlama kavramlarını öğrenmelerine ilişkin düzeyi belirleme ve programlama deneyimlerine yönelik görüşlerine yönelik sonuçlar sunulmuştur.

Çalışmada uygulama öncesi ve sonrası olmak üzere bilgi işlemsel düşünme becerisi öz yeterlik algı düzeylerini ölçebilmek amacıyla geliştirilmiş olan ölçek uygulanmıştır. Uygulamalar sonucunda ön test ve son testler değerlendirildiğinde, ortaokul öğrencilerinin almış oldukları fiziksel programlama aracı ile Python programlama öğretiminin sonunda bilgi işlemsel düşünme becerisi öz yeterlik algılarında önemli bir artış olduğu görülmüştür. Buna bağlı olarak, fiziksel programlama aracı kullanılarak öğretilen metin tabanlı programlama eğitiminin öğrencilerin bilgi işlemsel düşünme becerisine yönelik öz yeterlik algılarını geliştirmede önemli bir etkisinin olduğu söylenebilir. Ulaşılan bu sonuca göre ortaokul öğrencilerinin bilgi işlemsel düşünme becerilerine yönelik öz yeterlik algılarının problem çözme yeterlikleri, algoritma tasarlama yeterlikleri, temel programlama yeterlikleri, veri işleme yeterlikleri ve özgüven yeterliklerinin fiziksel programlama aracı ile Python programlama dili öğretilmesinin etkilediği söylenebilir. Bu durumun sebebi ise, ortaokul öğrencilerinin bu şekilde programlama öğrenimi alması sırasında uygulamaların öğrenmeyi kolaylaştırmış olması, soyut kavramları somutlaştırarak öğrenmeyi kolaylaştırması, öğrencilerin derse ve programlama öğrenmeye karşı isteklerinin artmasından kaynaklandığı söylenebilir. Yaptıkları çalışmalarda bireylerin eğitim

sürecindeki öğrenmiş oldukları bilgi ve kavramları uyguladıkları zamanda öz yeterlik becerilerinin ve algılarının da arttığını söylemişlerdir (Toytok ve Çetin, 2015; Çetinkaya 2019). Günbatar ve Karalar, 2018 yılındaki çalışmalarında fiziksel bir programlama aracı ile programlama öğretiminin öğrenciler üzerinde öz yeterlik algıları ile programlamaya yönelik tutumlarını inceleyerek önemli bir artış olduğu sonucuna varmışlardır. Hinton (2017) ise fiziksel programlama araçlarından robotik kitlerin STEM eğitiminde kullanılmasına yönelik yapmış olduğu doktora tezinde, öğrencilerin STEM eğitime ve derse yönelik ilgilerinin arttığını gözlemlemiştir.

Çalışmada uygulama öncesi ve sonrası olmak üzere bilgi işlemsel düşünme becerilerini ölçebilmek amacıyla Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme Görevleri öğrencilere uygulanmıştır. Uygulamalar sonucunda ön test ve son testler değerlendirilmiş ortaokul öğrencilerinin almış oldukları fiziksel programlama aracı micro:bit ile Python metin tabanlı programlama dili öğretiminin bilgi işlemsel düşünme becerisini önemli bir düzeyde arttırdığı sonucuna ulaşılmıştır. Buna bağlı olarak, fiziksel programlama aracı kullanılarak öğretilen metin tabanlı programlama eğitiminin öğrencilerin bilgi işlemsel düşünme becerilerini geliştirmede önemli bir etkisinin olduğu söylenebilir. Yapmış oldukları çalışmalarda, robotik kit kullanımının öğrencilerin problem çözme ve bilişsel düşünme becerilerini olumlu bir şekilde etkilediğini belirtmişlerdir (Sullivan ve Bers, 2017; Özdoğru 2013). Çalışmalarında programlamanın problem çözme dışında 21. yüzyıl becerileri üzerinde de olumlu yönde etkisinin olduğu ifade etmişlerdir (Coşar, 2013). Ayrıca programlama becerileri öğrencilerin eleştirel, analitik ve algoritmik düşünme becerilerini geliştirmektedir (Akçay ve Çoklar, 2016). Bu çalışma kapsamında öğrenciler uygulama süreci sonunda problem çözmeye yönelik beceriler edindiklerini hem odak grup görüşmesinde hem de ders sürecinde belirtmiş, ders gözlem formu yoluyla toplanan veriler öğrencilerin farklı şekillerde problem çözme süreçleri içerisine girdiklerini ve dersi anladıklarını göstermiştir. Uygulama sürecinde ve sonunda öğrencilere tanımlanan ödev ve projelere olan ilgi ve dönütler göstermektedir ki öğrenciler derse karşı ilgili, motive ve programlama öğretiminde hedeflenmiş olan programlama kavramlarını (döngüler, değişkenler, koşul yapıları, listeler, fonksiyonlar-prosedürler ve diziler vb.) anlamışlardır. Öğrencilerin ödev ve projelerde zorlandıkları tek nokta ellerinde böyle bir aracın olmamasıdır. Bu nedenle tasarlayarak yazmış oldukları programları evde deneyememişlerdir. Tüm bunların sonucunda çalışmanın hedefine ulaşıldığı söylenebilir.

Çalışmada uygulama sonrası fiziksel programlama aracı destekli metin tabanlı programlama öğretiminin, öğrencilerin programlama öğretiminde kullanılan fiziksel programlama aracını ve Python dili programlama kavramlarını öğrenmelerine ilişkin düzeylerinin belirlenmesi amacıyla çoktan seçmeli kazanım testi uygulanmıştır. Fiziksel programlama aracının donanımsal kavramlarına (micro:bit aracını bilgisayara bağlayabilme, düğme ve sensörlerini tanıma ve kullanmaya) ilişkin sorular ile programlama kavramlarına (döngüler, değişkenler, koşul yapıları, listeler, fonksiyonlar-prosedürler ve diziler) ilişkin sorular öğrencilere sorulmuştur. Sonuçlar değerlendirildiğinde öğrencilerin büyük çoğunluğunun başarılı olmasına bağlı olarak fiziksel programlama aracı ile Python metin tabanlı programlama dili öğretiminin etkili olduğu gözlemlenmiştir. Bu nedenle metin tabanlı programlama dili Python öğretiminin olumlu sonuç vermiştir denebilir. Feldhausen, Weese ve Bean (2018) benzer bir çalışma gerçekleştirmişlerdir ve yapmış oldukları çalışma ile bu çalışmanın sonuçları örtüşmektedir. Yapılan uygulama sürecinin hedeflerinden biri de metin tabanlı programlama dili Python, bilgi ve becerilerinin öğrencilere kazandırılmasıydı. Bu doğrultuda öğrencilerin programlama ile ilgili birçok temel kavramı tanımlayabilmeleri ve uygulayabilmeleri önemliydi. Bu anlamda uygulama sürecinde ders gözlem formu doldurularak öğrencilerin programlama dili ve fiziksel programlama aracını kullanma düzeylerine bakılmıştır. Programlama kavramları ve süreçleri, fiziksel programlama aracının donanımı ve kullanımına ilişkin etkinliklerin öğrencilerin soyut programlama kavramlarını öğrenerek bu kavramlar arasında ilişki kurmalarını desteklediği söylenebilir. Dolayısıyla üst düzey düşünme ve akademik başarıyı arttırdığını belirtmişlerdir (Çatlak, Tekdal ve Baz, 2015; Akçay, Karahan ve Türk, 2019). Uygulama öncesi öğrencilerin tedirgin oldukları gözlemlenmiştir. Ders sırasında sohbetlerde öğrencilerin bir kısmının programlamadan özellikle çekindikleri gözlemlenmiştir. Programlamanın zor ve karmaşık olduğunu, program yazmanın çok zor olduğunu düşündüklerini ifade ettiler. Böyle bir ön yargıları olduğu için yıllardır seçmeli ders kapsamında yürütülen bilişim teknolojileri ve yazılım dersine kız öğrencilerin erkek öğrencilere oranla daha az başvurduğu görülmektedir.

Uygulama sırasında programlama öğretiminde karşılaşılan bazı zorluklar fark edilmiştir. Bu zorluklar şöyle ifade edilebilir; öğrenciler problemi analiz etmekte (soyutlama becerisi), gerekli işlem adımlarını belirlemede (algoritmik düşünme) ve çalışmayan noktaları belirlemede (değerlendirme) zaman zaman zorlandıkları görülmüştür.

Bu noktada eğitimciler tarafından pek çok fikir ve çözüm yolu sunulmaktadır. Programlama öğreniminde kalıcı ve etkili bir öğrenmenin gerçekleşebilmesi için bol tekrar ve pratik yapılması önerilmektedir (Özmen ve Altun, 2014). Bol örnek çözmek aynı zamanda anlaşılması zor kavramları daha iyi pekiştirerek anlamaya, yanlış öğrenmeleri düzeltmeye olanak sağlamaktadır (Schunk, 2011). Gagne'ye (1985) göre somut işlemler döneminde olan ve soyut kavramları öğrenemeyen öğrenciler için belirli aralıklarla etkinlikler yapılması gerektiğini ifade edilmiştir. Bu sayede öğrenciler programlama kavramlarını uzun süreli belleklerine transfer edebilirler (Cevahir ve Özdemir, 2017). Derus ve Ali (2012) çalışma sonuçlarında, öğrencilerinin en çok program yazarken kodlardaki hataları bulmakta zorlandıklarını ifade etmişlerdir. Programı analiz etmekte, algoritma tasarlamakta ya da çalışmayan noktaları belirleme olsun programlamada nerede zorlanılırsa zorlanılsın en önemli nokta programlama kavramlarını bilmemektir. Bu aşamada öğrenilmemiş ya da eksiklik olan konulara yönelik uygulamaların sayısını arttırmak öğrenciler için yardımcı olacaktır.

Çalışmada uygulama sonrası fiziksel programlama aracı destekli metin tabanlı programlama öğretimine yönelik öğrencilerin görüşleri odak grup görüşmesi ile alınmıştır. Yapılan görüşmelerde öğrencilerin tamamı uygulama sürecinden memnun olduklarını ve bir daha bu şekilde bir programlama dersi almak istediklerini ifade ettiler. Clement ve Gullo, (1984), araştırmalarında programlama becerisinin problem çözme, algoritmik düşünme, sorgulama becerisinin gelişimine katkısı olduğunu ifade etmektedirler. Çalışmada en belirgin olan, fiziksel programlama aracını kullanmaya dair sürecin en başından beri öğrencilerin olumlu tepkisi gözlemlenmiştir. Öğrencilerin neredeyse tamamı fiziksel programlama aracı ile hem problem çözme hem de programlama açısından faydalı, kullanışlı ve motive edici bulduklarını ifade ettiler. Öğrencilerin bir kısmı fiziksel programlama aracı ile program yazma, problem çözme ve takım çalışması yapma gibi önemli becerileri geliştirmesine ve buna ek olarak birlikte çalışma fırsatı sağladığı için kullanmaktan sıkılmadıklarını ifade etmişlerdir. Çalışma da öğrenciler en çok yazdıkları kodları fiziksel programlama aracı üzerinde görmekten ve onu kullanarak arkadaşları ile gezip birbirlerine çalışmalarını göstererek çok keyifli süreç geçirdiklerini ifade etmişlerdir. Derste oyun oynuyor gibi hissettiklerini vurgulamışlardır. Bu da derse karşı ve programlama öğrenimine karşı motivasyonları arttırmıştır. Bazı etkilerin kendi oyunlarını programlıyormuş gibi projeler geliştirmelerini sağlamıştır. Bu da uygulama çok kısa bir süreç olsa da öğrencilerin programlama öğrenmeleri konusunda ilerleme kaydettiklerini

göstermektedir. Görüşme süresince ve dersler esnasında öğrenciler fiziksel programlama aracı kullanarak ve Python programlama öğrenmenin motivasyonlarını arttırdığını ifade ettiler. Yaptıkları çalışmalarında öğrenci görüşlerinde, Python programlama dilinin zaman zaman zor ve ağır bir dil olduğunu ifade ettiklerini belirtmişlerdir (Largo-Wight, Peterson ve Chen, 2005). Hatta öğrenciler ilk kez Python programlama dersi aldıkları için daha çok kavram öğretimi üzerinde durulduğundan bu durum onları biraz süreç içerisinde zorladığı gözlemlenmiştir. Farklı bakış açıları geliştirme imkânı sunabilecek örnek uygulama zamanı çok olamamıştır. Program yazmada sorun yaşayanlar ya da öğrenme sıkıntısı yaşayan öğrenciler gözlemlenmiştir. Akademik anlamda daha çok zorlanan öğrencileri süreçte uyarlayabilmek katabilmek için daha uzun süreli bir öğretim programı uygulanması gerekliliği fark edilmiştir.

Sonuç olarak yapılan literatür taramasında bilgi işlemsel düşünme ve metin tabanlı programlama öğretimi ile ilgili çalışmaların diğer programlama yöntemlerine göre çok az olduğu görülmektedir. Yapılan bu çalışma ile benzerlik gösteren araştırmacılar ortaokul düzeyindeki öğrencilerin metin tabanlı programlama dilini öğrenerek başarılı olabileceklerini ifade etmişlerdir (Dorling ve White 2015; Park ve Yoo, 2018; Feldhausen, Weese ve Bean, 2018). Bilgi işlemsel düşünme becerilerinin çalışıldığı üniversite öğrencileri ile programlama eğitimi sonunda, öğrencilerde bilgi işlemsel düşünme ve programlama ile ilgili öğrenme memnuniyetinde, öz yeterlikte ve gelecekte bu alanda meslek seçmeye karşı oluşan olumlu değişikliğin bu çalışma sonuçları ile benzerlik göstererek desteklediği görülmektedir (Kwon ve Kim, 2018). Programlamanın fiziksel ortamla verilerek bilgi işlemsel düşünme ve ilgili becerileri üzerine farklı seviyelerdeki öğrencilerle yapılmış ve olumlu etkisi olduğunu aktaran araştırmacılar bu çalışma sonuçlarını destekler niteliktedir (Berland ve Wilensky, 2015; Bers ve ark., 2014). Öğrenciler çalışmalarda aynı vurguyu yapmışlardır; fiziksel programlama araçları ile programlama gibi soyut bir durumu, somut ürüne dönüştürebilmek ve programlayıp sonucu aynı zamanda görebilmeleri çalışmalardan çıkan olumlu sonuçları açıklamaktadır. Tüm bu sonuçlar bağlamında gerçekleştirilen bu çalışma ile ortaokul öğrencilerinin fiziksel programlama aracı kullanarak programlamayı öğrenmelerinin bilgi işlemsel düşünme becerilerine ve programlamaya yönelik deneyimlerine ilişkin görüşlerini görmek adına bu çalışmanın kullanımının uygulanabilirlik yönünden alanyazına katkı sağlayacağı düşünülmektedir. Çalışma sonuçlarının özellikle ülke genelindeki eğitim ve öğretim kurumlarında görevli bilişim teknolojileri ve yazılım alanında çalışan öğretmenler, politika

belirleyiciler, bireysel çalışan eğitimciler ve araştırmacılar için rehber niteliğinde olacağı, ayrıca geleceğe dönük bu alanda yapılacak araştırmalar için ışık tutacağı öngörülmektedir.

5.2. Öneriler

Bu bölümde bu çalışmadan elde edilen bulgular ışığında araştırma yapacak eğitimcilere ve araştırmacılara önerilere yer verilmiştir.

Bu çalışma ile alanda çalışmalarını yürüten eğitimcilere ortaokul düzeyinde programlama öğretimi verirken öğrencilere sadece blok tabanlı programlama değil metin tabanlı programlamada verilebileceklerini göstermektir. Programlama öğretiminde öğrencileri blok tabanlı programlarla sınırlamadan onlara fırsat verildiğinde metin tabanlı programlamayı da öğrenebileceklerini uygulama ve örneklerle göstermektir. Öğrencilere zor ve karmaşık gelen metin tabanlı programlamayı fiziksel programlama aracı kullanarak somut çıktıları görmeleri sağlandığında öğrencilerin ilgi ve motivasyonlarının arttığı söylenebilir. Bu noktada öğrencilerin derse katılımının artmasıyla programlama becerilerinin artması doğru orantılı olarak gözlemlenmiştir. Dolayısıyla bilgi işlemsel düşünme becerileri ile bilgi işlemsel düşünmeye yönelik öz yeterlik algılarında da artış olmuştur. Tüm bunların ışığında öğrencilere bilişim teknolojileri ve yazılım dersinin kazanımlarını ve 21. yüzyıl becerilerini kazandırmak adına bu çalışmanın yol gösterici olacağı düşünülmektedir. Öğrencilere metin tabanlı programlama öğretiminde fiziksel programlama araçlarının kullanılmasının bilgi işlemsel düşünme becerilerinde, öz yeterlik algılarında ve programlama kavramlarını öğrenme düzeylerinde görülen anlamlı artış gözlemlenmiştir. Buna bağlı olarak hem bu çalışma hem de alanyazındaki diğer çalışmaların sonuçlarındaki olumlu etkisinin programlama öğretimi ve bilgi işlemsel düşünme becerileri ile ilgili olarak eğitimciler açısından örnek teşkil edeceği düşünülmektedir.

Fiziksel programlama araçlarının kullanımına yönelik öğretim programının doğru bir şekilde planlanması hem dersten verim alınması hem de öğrenci motivasyonu açısından daha fazla çalışmanın yapılması önerilmektedir. Alanyazında bu çalışmada kullanılan fiziksel programlama aracı kullanılarak Python programlama öğretimi çalışmalarının sayısının az olması sebebiyle, bu çalışmanın uygulama süreci ve sonuçları bakımından bundan sonra metin tabanlı programlama çalışacaklara öneri olarak; platform seçimi, araç üzerinde gerçekleştirilebilecek interaktif etkinlikler düşünülebilir. Bu nokta da daha fazla

platformun incelenmesi ve eğitsel anlamda fiziksel programlama araçlarının kullanılabilirliğinin ve eğitsel süreçlere katkısının ne düzeyde olacağının belirlenmesi programlama öğretiminin etkililiğini ortaya çıkarmakta daha çok katkı sağlayacağı düşünülmektedir. Bu bağlamda yeni yapılacak çalışmalarda araştırmacılara fiziksel programlama aracı kullanılarak farklı programlama dilleri ile çalışırken farklı değişkenlerin araştırılması önerilir. Bu anlamda uzun süreli bir programlama öğretim süreci de değerlendirilebilir. Örneğin daha küçük yaşlarda blok tabanlı ile başlayan sonrasında metin tabanlı fiziksel programlama aracı çalışması yapılarak her iki süreçte de etkililiği incelenebilir. Farklı fiziksel programlama araçları ile programlama öğretimi hazırlanarak birkaç aracın sınırlılıkları, olumlu ve olumsuz yönleri açısından incelenmesi önerilmektedir.



KAYNAKLAR

- Ackermann, E. (2022). Piaget's Constructivism, Papert's Constructionism: What's the difference?, Erişim adresi: [https://learning.media.mit.edu/content/publications/EA.Piaget%20 %20Papert.pdf](https://learning.media.mit.edu/content/publications/EA.Piaget%20%20Papert.pdf)
- Andrews, C. (2016). BBC micro:bit a little bit too late?(IT Education), in *Engineering and Technology*, vol.11, no.4, s. 30-33, Mayıs 2016, doi: 10.1049/et.2016.0400
- Akçay, A. O., Karahan, E. ve Türk, S. (2019). Bilgi işlemsel düşünme becerileri odaklı okul sonrası kodlama sürecinde ilköğrencilerinin deneyimlerinin incelenmesi. Eskişehir Osmangazi Üniversitesi, Türk Dünyası Uygulama ve Araştırma Merkezi Eğitim Dergisi, 4(2), s.38-50. Erişim adresi: <https://dergipark.org.tr/tr/pub/estudamegitim/issue/50016/626608>
- Akçay, A. ve Çoklar, A. N. (2016). Bilişsel becerilerin gelişimine yönelik bir öneri: programlama eğitimi. A. İşman, H. F. Odabaşı ve B. Akkoyunlu (Ed.), Eğitim teknolojileri okumaları 2016, s. 121- 140 içinde. Ankara: TOJET-Sakarya Üniversitesi.
- Akkoyunlu, B. ve Kurbanoglu, S. (2003). Öğretmen adaylarının bilgi okuryazarlığı ve bilgisayar öz-yeterlik algıları üzerine bir çalışma. *Hacettepe Üniversitesi Eğitim Fakültesi Dergisi*, 24(24).
- Akpınar, Y. (1999). *Bilgisayar destekli öğretim ve uygulamalar*. Ankara: Anı Yayıncılık.
- Akpınar, Y. ve Altun, A. (2014). Bilgi toplumu okullarında programlama eğitimi gereksinimi. *İlköğretim Online*, 13(1), dy:1-4, 2014. Erişim adresi: <http://ilkogretim-online.org.tr>
- Aldağ, H., ve Tekdal, M. (2015). Bilgisayar kullanımı ve programlama öğretiminde cinsiyet farklılıkları. Ağrıdağ, G. (Ed.), *1. Uluslararası Çukurova Kadın Çalışmaları Kongresi* (s. 236–243)

- Alkar, E. (2017). *Ortaokul 6. sınıf sosyal bilgiler programı ülkemiz ve dünya ünitesinin bilgisayar destekli materyallerle öğretiminin öğrencilerin akademik başarılarına etkisi*. Mehmet Akif Ersoy Üniversitesi (Yüksek lisans tezi).
- Alkan, V., Şimşek, S. ve Armağan Erbil, B. (2019). Karma yöntem: Öyküleyici alan yazın incelemesi. *Eğitimde Nitel Araştırmalar Dergisi – Journal of Qualitative Research in Education*, 7(2), 559-582.
- Alp, Y. (2019). *Blok tabanlı programlama öğretiminin ortaokul öğrencilerinin problem çözme becerisine ve bilgisayara yönelik tutumuna etkisi*. İnönü Üniversitesi, Eğitim Bilimleri Enstitüsü, Bilgisayar ve Öğretim Teknolojileri Eğitimi Anabilim Dalı (Yüksek Lisans Tezi).
- Ananiadou, K. ve Claro, M. (2009). "21st Century skills and competences for new millennium learners in OECD countries", *OECD Education Working Papers*, 41, OECD Publishing, Paris, <https://doi.org/10.1787/218525261154>
- Armoni, M., Meerbaum-Salant, O. ve Ben-Ari, M. (2015). From scratch to "real" programming. *Transactions on Computing Education*, 14(4), DOI: 10.1145/2677087.
- Aşkar, P. ve Davenport, D. (2009). An investigation of factors related to self-efficacy for java programming among engineering students, *The Turkish Online Journal of Educational Technology – TOJET January*. 8(1).
- Atmatzidou, S. ve Demetriadis, S. (2016). Advancing students' computational thinking skills through educational robotics: A study on age and gender relevant differences. *Robotics and Autonomous Systems*, 75, s.661-670.
- Atmatzidou, S., Demetriadis, S. ve 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), s.70-85.

- Austin, J., Baker, H., Ball, T., Devine, J., Finney, J., Halleux, P.D., Hodges, S., Moskal, M. ve Stockdale, G. (2020). Association for Computing Machinery. Volume 63, Issue 3, s. 62-69.
- Aydoğdu, E. (2019). *Bilgisayarsız Etkinlikler Sürecinde Öğrencilerin Algoritmik Düşünme Becerilerinin İncelenmesi. Trabzon Üniversitesi. Eğitim Bilimleri Enstitüsü, BÖTE Anabilim Dalı (Yüksek lisans tezi).*
- Aytekin, A., Çakır, F. S., Yücel, Y. B. ve Kulaöz, İ. (2018). Geleceğe yön veren kodlama bilimi ve kodlama öğrenmede kullanılabilecek bazı yöntemler. *Avrasya Sosyal ve Ekonomi Araştırmaları Dergisi*, 5(5), s.24–41.
- Baki, A. (2000). Bilgisayar donanımlı ortamda matematik öğrenme, *Hacettepe Üniversitesi Eğitim Fakültesi Dergisi* 19 : 1, s. 86-193
- Ball, T., Protzenko, J., Bishop, J., Moskal, M., Halleux, J., Braun, M., Hodges, S. ve Riley, C. (2016). Microsoft touch develop and the BBC micro:bit. *IEEE/ ACM 38th IEEE International Conference on Software Engineering Companion*. s. 637-640.
- Baltalı, S. (2016). *Programlama öğretiminde kullanılabilecek yazılımlara ilişkin öğretmen görüşleri. Uludağ Üniversitesi, Eğitim Bilimleri Enstitüsü, BÖTE Ana Bilim Dalı (Yüksek Lisans Tezi).*
- Banzi, M. ve Shiloh, M. (2015). Getting started with Arduino (3rd ed.). CA: *Maker Media*.
- Barba, E. ve Chancellor, S. (2015). Tangible media approaches to introductory computer science. In V. Dagienė, C. Schulte ve T. Jevsikova (Edt.), *Proceedings of 2015 ACM conference on innovation and Technology in Computer Science Education*. s. 207–212.
- Barut, E., Tugtekin, U. ve Kuzu, A. (2016). Programlama eğitiminin bilgi işlemsel düşünme becerileri bağlamında incelenmesi. *Fourth International Instructional Technologies & Teacher Education Symposium*. October 2016.

- Başer, M. (2013). Bilgisayar programlamaya karşı tutum ölçeği geliştirme çalışması. *The Journal of academic social science studies*. Volume 6 Issue 6, S.199-215. 2013.
- Baştuğ, İ. (2019). *Jquery tabanlı blok programlama öğretiminin programlamaya yönelik tutuma etkisinin değerlendirilmesi (Başiskele Örneği)*. Sakarya Üniversitesi Eğitim Bilimleri Enstitüsü, Sakarya (Yüksek Lisans Tezi).
- Battelle for Kids (P21). (2021). Erişim Adresi: <http://www.battelleforkids.org/networks/>
- Bekker, T., Bakker, S., Douma, I., Van Der Poel, J. ve Scheltenaar, K. (2015). Teaching children digital literacy through design-based learning with digital toolkits in schools. *International Journal of Child-Computer Interaction*, vol. 5, pp. 29-38, 2015.
- Bell, T.C., Alexander, J., Freeman, I. ve Grimley, M. (2009). Computer science unplugged: school students doing real computing without computers. *Journal of Applied Computing and Information Technology*, 13, s. 20-29.
- Bennedsen, J., Caspersen, M.E. ve Kölling, M. (2008). Reflections on the teaching of programming: methods and implementations. *Lecture Notes in Computer Science 4821*, ISBN: 978-3-540-77933-9. DOI:10.1007/978-3-540-77934-6.
- Berland, M. ve 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.
- Bernad, P., Šic, D., Repnik, R. ve Osrajnik, D., 2021. Development of measurement systems with the BBC Micro:bit. *2021 44th International Convention on Information, Communication and Electronic Technology (MIPRO)*, 2021, pp. 853-858, doi: 10.23919/MIPRO52101.2021.9596834.
- Bilge Kunduz (2022). Bilge kunduz- uluslararası enformatik ve bilgi işlemsel düşünme etkinliği. Erişim Adresi: <http://www.bilgekunduz.org/>

- Bishop-Clark, C., Courte, J. ve Howard, E. V. (2007). A quantitative and qualitative investigation of using Alice programming to improve confidence, enjoyment and achievement among non-majors. *Journal of Educational Computing Research*, 37(2) 193-207.
- British Council (2022). 21st Century Schools Programme. Eriřim Adresi: <https://www.britishcouncil.me/en/programmes/education/21st-century-schools/about>
- Buss, A. ve Gamboa, R. (2017). Teacher transformations in developing computational thinking: Gaming and robotics use in after-school settings. *In Emerging Research, Practice, and Policy on Computational Thinking* (s. 189-203). Springer, Cham.
- B y k zt rk, ř. (2007). *Sosyal bilimler i in veri analizi el kitabı* (8. baskı). Ankara: Pegem Yayıncılık.
- B y k zt rk, ř.,  akmak, E.K., Akg n,  .E., Karadeniz, ř. ve Demirel F. (2018). *Eđitimde bilimsel arařtırma y ntemleri*. Pegem Akademi. Ekim 2018, 25.Baskı.
- Byrne, P. ve Lyons, G. (2001). The effect of student attributes on success in programming. *In ACM SIGCSE Bulletin*. Vol. 33, No. 3, (pp. 49-52). ACM.
- Can, A. (2019). *SPSS ile bilimsel arařtırma s recinde nicel veri analizi*. Ankara: PEGEM Akademi.
- Cansoy, R. (2018). Uluslararası  er evelere g re 21. y zyıl becerileri ve eđitim sisteminde kazandırılması. *İnsan ve Toplum Bilimleri Arařtırmaları Dergisi*. doi: 15869/itobiad.494286.
- C pay, M., Kvařřayov , N., Bellayov , M., Mansell, M. ve Petr k, ř. (2022). Programmable hardware BBC Micro:bit as a tool for developing teacher competencyities, 2022 *IEEE Global Engineering Education Conference (EDUCON)* , 2022, s. 1496-1501, doi: 10.1109/EDUCON52537.2022.9766487.

- Caspersen, M. E. (2007). *Educating novices in the skills of programming (PhD)*, University of Aarhus Denmark.
- Cevahir, H. ve Özdemir, M. (2017). Teacher's opinions and suggestions for solutions towards challenges in programming teaching (Programlama öğretiminde karşılaşılan zorluklara yönelik öğretmen görüşleri ve çözüm önerileri). *11. Uluslararası Bilgisayar ve Öğretim Teknolojileri Eğitimi Sempozyumu*, Malatya.
- Cengiz, S.A. (2019). *Endüstri 4.0 sürecinin eğitim sistemine yansımaları; Türkiye örneği*. Yayınlanmış. Nevşehir Hacı Bektaş Veli Üniversitesi, Sosyal Bilimler Enstitüsü, Kamu Politikası ve İşlemciliği Ana Bilim Dalı (Yüksek Lisans Tezi).
- Choi, H. (2013). Learners' reflections on computer programming using Scratch: Korean primary pre-service teachers' perspective. In *10th International Conference for Media in Education 2012 (ICoME)*. s. 22-24.
- Clements, D. H. ve Gullo, D. F. (1984). Effects of computer programming on young children's cognition. *Journal of Educational Psychology*. 76(6). s. 1051-1058.
- Coşkunserçe, O. (2021). *Eğitimde Robot Programlama*. Pegem Akademi. S.3.
- Creswell, J.W. (2021). *Karma yöntem araştırmalarına giriş*. Çeviri: Mustafa Sözbilir. 3.Baskı. Pegem Akademi. doi: 10.14527/9786053184720
- Creswell, J. W. (2014). *Araştırma deseni: Nitel, nicel ve karma yöntem yaklaşımları* (Çev. Ed. S. B. Demir). Ankara: Eğiten Kitap.
- Creswell, J. W. (2015). *Nitel araştırma yöntemleri beş yaklaşıma göre nitel araştırma ve araştırma deseni* (Çev. Ed. M. Bütün ve S. B. Demir). Ankara: Siyasal Kitabevi.
- Creswell, J. W. ve Plano Clark, V. L. (2014). *Karma yöntem araştırmaları: Tasarımı ve yürütülmesi* (Çev. Ed. Y. Dede ve S. B. Demir). Ankara: Anı Yayıncılık.

CSTA (2016). CSTA K-12 Computer Science Standards Revised.

CSTA, (2011). Deborah Seehorn, Stephen Carey, Brian Fuschetto, Irene Lee, Daniel Moix, Dianne O'Grady-Cunniff, Barbara Boucher Owens, Chris Stephenson, and Anita Verno. 2011. *CSTA K--12 Computer science standards: revised 2011. Technical Report. Association for Computing Machinery*, New York, NY, USA. ISBN: 978-1-4503-0881-6, doi: <https://doi.org/10.1145/2593249>

CSUnplugged (2022). Eriřim adresi: <https://www.csunplugged.org/en/>

Code.org, 2022. Bilgisayar bilimini öğren. Eriřim adresi: <https://code.org/>

Cortina, T. J., Dann, W. P., Frieze, C., Ciminillo, C., Tananis, C. ve Trahan, K. (2012). Work in Progress: ACTIVATE: Advancing computing and technology interest and innovation through teacher education. Paper presented at the *Frontiers in Education Conference*, Seattle, WA.

Cořar, M. (2013). *Problem temelli öğrenme ortamında bilgisayar programlama çalışmalarının akademik başarı, eleřtirel düşünme eğilimi ve bilgisayara yönelik tutuma etkileri*. Gazi Üniversitesi, Eğitim Bilimleri Enstitüsü, Ankara (Doktora Tezi.).

Çağıltay, N. E. ve Fal, M. (2014). *Scratch ile programlama öğreniyorum*. Ankara: ODTÜ Yayıncılık.

Çalışkan, E. (2019). Ortaokul öğrencilerinin kodlama dersine ilişkin bilgi ve düşüncelerinin incelenmesi. *28.Uluslararası Eğitim Bilimleri Kongresi Kongre Kitapçığı s.104*. 25-28 Nisan 2019. Hacettepe Üniversitesi.

Çakıroğlu, Ü., Sarı, E. ve Akkan, Y. (2011). The view of the teachers about the contribution of teaching programming to the gifted students in the problem solving. *Paper presented at 5th International Computer & Instructional Technologies Symposium* (22-24 September), Fırat University, Elâzığ.

- Çatlak, Ş., Tekdal, M. ve Baz, F. Ç. (2015). Scratch yazılımını ile programlama öğretimini durumu: Bir döküman inceleme çalışması. *Journal of Instructional Technologies & Teacher Education*, 4(3), 13–25.
- Çelik, İ. (2019). *Programlama ve gelişimi. Programlama öğretimi yaklaşımları*. 1.Basım. Edt: Prof. Dr. Özgen Korkmaz. Nobel Yayın. S.6.
- Çetin, E. (2016). Okul öncesi çocukların problem çözme sürecinde teknoloji destekli şematik düzenleyicilerin kullanımına yönelik bir durum çalışması. Gazi Üniversitesi, Eğitim Bilimleri Enstitüsü, Ankara (Doktora tezi).
- Çetinkaya, H. N. (2019). Bilişim teknolojileri ve yazılım dersindeki etkinliklerin bilgi işlemsel düşünme ve bazı değişkenler açısından incelenmesi. *İnönü Üniversitesi, Eğitim Bilimleri Enstitüsü, Bilgisayar ve öğretim teknolojileri eğitimi ana bilim dalı (Yüksek lisans tezi.)*.
- Çiftci, S., Yayla, A. ve Sağlam, A. (2021). 21. yüzyıl becerileri bağlamında öğrenci, öğretmen ve eğitim ortamları. *Rumelide Dil ve Edebiyat Araştırmaları Dergisi*, 24, s.718-734.
- Dağhan, G., Kibar, P. N., Çetin, N. M., Telli, E. ve Akkoyunlu, B. (2017). Bilişim teknolojileri öğretmen adaylarının bakış açısından 21. Yüzyıl öğrenen ve öğretmen özellikleri. *Eğitim Teknolojisi Kuram ve Uygulama*, 7(2), s.215-235.
- Demir, E., Saatçioğlu, Ö. ve İmrol, F. (2016). Uluslararası dergilerde yayımlanan eğitim araştırmalarının normallik varsayımları açısından incelenmesi. *Current Research in Education*, 2(3), 130-148.
- Demir, F. (2015). Programlama öğretiminde eğitsel programlama dilinin farklı kullanımlarının programlama başarısı ve kaygısına etkisi. *Atatürk Üniversitesi, Eğitim bilimleri enstitüsü, doktora tezi*.
- Demir, Ö. ve Seferoğlu, S. S. (2017). Yeni kavramlar, farklı kullanımlar: Bilgi-işlemsel düşünmeyle ilgili bir değerlendirme. H. F. Odabaşı, B. Akkoyunlu ve A. İşman (Ed.).

Derus, S. ve Ali, A. Z. (2012). Difficulties in learning programming: views of students. *1st International Conference. Current Issues in Education, ICCIE 2012*.s.74-78. Yogyakarta, Indonesia: University of Pendidikan Sultan Idris.

Department of Education, (2013). The national curriculum in England Framework document. İngiltere. Erişim adresi: <https://www.gov.uk/government/organisations/department-for-education>

DeSeCo, 2022. Erişim adresi: <https://www.deseco.ch/>

Djambong, T., ve Freiman, V. (2016). Task-Based assessment of students' computational thinking skills developed through visual programming or tangible coding environments. International Association for Development of the Information Society.

Dorling, M., ve White, D. (2015). Scratch: A way to logo and python. In proceedings of the 46th ACM Technical Symposium on Computer Science Education, s. 191– 196. New York, USA.

DuBoulay, B. (1986). Some difficulties of learning to program. *Journal of Educational Computing Research*, I, 57-73.

Durak, H.Y. ve Sarıtepeci, M. (2018). Analysis of the relation between computational thinking skills and various variables with the structural equation model. *Computer & Education. Publisher: Elsevier*. Ocak 2018.

Ecevit, T. ve Kaptan, F. (2019). 21.Yüzyıl becerilerinin kazandırılmasına yönelik tasarlanan argümantasyon destekli araştırma sorgulamaya dayalı öğretim modelinin betimlenmesi. *Hacettepe Üniversite Eğitim Fakültesi Dergisi*, doi: 10.16986/HUJE.2019056328.

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, s. 19- 47. Hershey, PA: IGI Global.

Ersoy, H. ve Aydın, S. (2015). Ortaokul öğrencilerine programlama becerileri kazandırmada scratch' in etkililiği. *Akademik Bilişim Konferansı*.

Ersoy, H., Madran, R.O. ve Gülbahar, Y. (2011). Programlama dilleri öğretimine bir model önerisi: robot programlama. *Akademik Bilişim 2011 Konferansı*.

Ersöz, B. ve Özmen, M. (2020). Dijitalleşme ve bilişim teknolojilerinin çalışanlar üzerindeki etkileri. *AJIT-e: Bilişim Teknolojileri Online Dergisi*, 2020, Summer/Yaz – Cilt/Vol: 11 - Sayı/Num: 42, (s.170-179) içinde. doi: 10.5824/ajite.2020.03.007.x

Erten, P. (2020). Öğretmen adaylarının 21. Yüzyıl becerileri yeterlilik algıları ve bu becerilerin kazandırılmasına yönelik görüşleri. *Milli Eğitim Dergisi*, 49(227), s.33-64.

Eryılmaz, S. ve Deniz, G. (2019). Türkiye’de programlama eğitimi ile ilgili yapılan çalışmaların incelenmesi: Bir betimsel analiz çalışması. *Eğitimde Kuram ve Uygulama*, 15(4), 319-338. doi: 10.17244/eku.645387

Faber, H. H., Wierdsma, M. D., Doornbos, R. P., van der Ven, J. S., ve 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.

Falkner, K. ve Vivian, R. (2015). A review of computer science resources for learning and teaching with K-12 computing curricula: An Australian case study. *Computer Science Education*, 25(4), s.390–429. doi: 10.1080/08993408.2016.1140410

Feldhausen, R., Weese, J. L. ve Bean, N. H. (2018). Increasing student self-efficacy in computational thinking via stem outreach programs. *SIGCSE'18*, Baltimore, ACM.

- Freudenthal, E., Ogrey, A. N., Roy, M.K. ve Siegel, A. (2010). A computational introduction to STEM studies. Paper presented at the *IEEE EDUCON 2010 Conference*, Madrid, Spain.
- Furner, J. M. ve Kumar, D. D. (2007). The mathematics and science integration argument: A stand for teacher education. *Eurasia Journal of Mathematics, Science ve Technology Education*, 3(3), s.185- 189.
- Kwon, J. ve 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.
- Gagne, R. M.(1985). *The conditions of learning*. New York, Holt, Rinehart ve Winston.
- Gander, W., Petit, A., Berry, G., Demo, B., Vahrenhold, J., McGettrick, A. ve Meyer, B. (2013). *Informatics education: Europe cannot afford to miss the boat*. In *ACM Europe: Informatics education report*. s. 1–21.
- Gelen, İ. (2017). P21-Program ve öğretimde 21. yüzyıl beceri çerçeveleri (ABD Uygulamaları). *Disiplinler arası Eğitim Araştırmaları Dergisi*, 1(2), s.15-29.
- Genç, Z. ve Karakuş, S. (2011). Tasarımla öğrenme: Eğitsel bilgisayar oyunları tasarımında Scratch kullanımı. 5. *Internatiol Computer & Instructional Technologies Symposium, Fırat Üniversitesi, Elâzığ*, 22-24 Eylül, s.981-987.
- Gibbs, A. (1997). Focus Groups. *Social research update*, 19. <http://sru.soc.surrey.ac.uk/SRU19.html>. Erişim Tarihi: 10.05.2022.
- Gibson, S.D. ve Bradley, P. (2017). A study of Northern Ireland key stage 2 pupils' perceptions of using the BBC Micro:bit in STEM education.
- Green, T., Wagner, R. ve Green, J. (2018). A look at robots and programmable devices for the K-12 classroom. *TechTrends*, 62(4), 414–422. doi:10.1007/s11528-018-0297-2

- Green, S.B. ve Salkind, N.J. (2014). *Using SPSS for Windows and Macintosh: Analyzing and understanding data*, (Seventh Edition), Pearson, New Jersey
- González, F., López, C. ve 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, s. 1- 8.
- Gomes, A. J. ve Mendes, A.J. N. (2007). Learning to program difficulties and solutions. C. S. Furtado and M. d. G. Rasteiro (Ed.), *Proceedings of the International Conference on Engineering Education n – ICEE, Coimbra, Portugal*, 3-7 September, 411.
- Google Education (2022). Computational thinking for educators. Erişim Adresi: <https://edu.google.com/resources/programs/exploring-computational-thinking/>
- Gülbahar, Y. (2017). *Bilgi işlemsel düşünme tanımı ve kapsamı, bilgi işlemsel düşünmeden programlamaya* (s.41 -74) içinde. Pegem Akademi (2.Baskı).
- Gülcan, C. (2021). Nitel Bir Veri Toplama Aracı: Odak (focus) Grup tekniğinin uygulanışı ve geçerliliği üzerine bir çalışma. *Mersin Üniversitesi Sosyal Bilimler Enstitüsü E-Dergi*, cilt: 4, sayı:2, Haziran 2021.
- Gülen, S. ve Dönmez, İ. (2021). Sosyal bilimlerde geleceğin meslekleri. *Uluslararası Sosyal Hizmet Araştırmaları Dergisi*, 1(1), 13-21.
- Gülmez, I. (2009). *Programlama öğretiminde görselleştirme araçlarının kullanımının öğrenci başarı ve motivasyonuna etkisi*. Marmara Üniversitesi Eğitim Bilimleri Enstitüsü, İstanbul (Yüksek Lisans Tezi).
- Günbatar, M. S. ve Karalar, H. (2018). Gender differences in middle school students' attitudes and self-efficacy perceptions towards mBlock programming. *European Journal of Educational Research*, 7(4), 925-933.
- Günüç, S., Odabaşı, H. ve Abdullah, K. (2013). 21. yüzyıl öğrenci özelliklerinin öğretmen adayları tarafından tanımlanması: bir twitter uygulaması/the defining characteristics of

students of the 21st century by student teachers: a twitter activity. *Eğitimde Kuram ve Uygulama*, 9(4), 436-455.

Harel, I. ve Papert, S. (1990). *Software design as a learning environment*. Interactive Learning Environments, 1(1), s.1-32.

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

Hermans, F. ve Aivaloglou, E. (2017). To scratch or not to scratch?: A controlled experiment comparing plugged first and unplugged first programming lessons. In Proceedings of the 12th Workshop on Primary and Secondary Computing Education, 49-56, ACM.

Hinton, P.R., McMurray, I. ve Brownlow, C. (2014). *SPSS explained-second edition*. ISBN: 978-0-415-61601-0.

Hinton, T. B. (2017). *An exploratory study of a robotics educational platform on STEM career interests in middle school students*. The Ohio State University (Doktora tezi).

Hubwieser, P., Giannakos, MN, Berges, M., Brinda, T., Diethelm, I., Magenheim, J., Pal, Y., Jacková, J. ve Jasute, E. (2015). K-12 Okullarında bilgisayar bilimleri eğitime küresel bir bakış. *2015 ITICSE Çalışma Grubu Raporları Tutanakları*.

Hürnalı, E. (2021). Algoritma geliştirme, ünite 1. ders notları. Erişim Sayfası: https://acikders.ankara.edu.tr/pluginfile.php/58694/mod_resource/content/0/DersNotu_AlgoritmaKavramiVeProblemCozme.pdf

Hyder, S.I. ve Bhamani, S. (2016). Bloom's taxonomy (cognitive domain) in higher education settings: reflection brief. *Journal of Education and Educational Development*. 3. 288. 10.22555/joed.v3i2.1039

ISTE (2015). Computational thinking leadership toolkit (Birinci Basım). Erişim Adresi: <https://www.iste.org/explore/articleDetail?articleid=152&category=Solutions&article=Computationalthinking-for-all>

ISTE (2016). International society for technology in education, ISTE 2016 Standards. <https://iste.org/standards/iste-standards-for-students>

Jin, K. H., Haynie, K. ve Kearns, G. (2016). "Teaching elementary students programming in a physical computing classroom", SIGITE '16: *Proceedings of the 17th Annual Conference on Information Technology Education*, s. 85-90, Erişim adresi: <https://doi.org/10.1145/2978192.2978238>

Jiang, S. ve Wong, G. K. (2018). Are children more motivated with plugged or unplugged approach to computational thinking?. In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*, 1094-1094, ACM.

Johnson, R.B. ve Onwuegbuzie, A.J. (2004). Mixed methods research: a research paradigm whose time has come. *Educational Researcher*, 33, 14-26. <http://dx.doi.org/10.3102/0013189X033007014>.

Junior, L. A., Neto, O. T., Hernandez, M. F., Martins, P. S., Roger, L. L. ve Guerra, F. A. (2013). A low-cost and simple Arduino-based educational robotics kit. *Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Robotics and Control (JSRC)*.

Kaucic, B. ve Asic, T. (2011). Improving introductory programming with Scratch? 34. *MIPRO konferansında sunulan bildiri*, Opatija, Croatia.

Kayabaşı, Y. (2007). Öğretimin değerlendirilmesinde güncel sorunlar ve sistemde yapılması mümkün düzeltmeler. *Kastamonu Eğitim Dergisi*. Ekim 2007, cilt 15, no: 2, s.539.

Kesici, T. ve Kocabaş, Z. (2007). *Bilgisayar 2 Ders Kitabı* (2. Baskı). Ankara: Semih Ofset.

- Kert, S.B. ve Uğraş, T. (2009). Programlama eğitiminde sadelik ve eğlence: Scratch örneği, *The First International Congress of Educational Research*, Çanakkale, Turkey.
- Kalelioğlu F. (2020). Bilgisayarsız Bilgisayar Bilimi (B3) Öğretimi. Yasemin Gülbahar editör. *Bilgi İşlemsel Düşünmeden Programlamaya* (4. Baskı). Ankara: Pegem Akademi; 2020
- Kalelioğlu, F. (2015). Türkiye’de programlama öğretimi. Y. Gülbahar, H. Karal (Ed.), *Kuramdan Uygulamaya Programlama Öğretimi* (s. 38-65). Ankara: PEGEM Akademi.
- Kalelioğlu, F. ve Gülbahar, Y. (2014). The effects of teaching programming via Scratch on problem solving skills: a discussion from learners' perspective. *Informatics in Education*, 13(1), 33.
- Kalelioğlu, F. ve Sentance, S. (2020). Teaching with physical computing in school: the case of the micro:bit. *Educ Inf Technol* 25, 2577–2603. <https://doi.org/10.1007/s10639-019-10080-8>.
- Kalelioğlu, F. ve Yurdakök, EA. (2017). Çocuklara programlama öğretimi ve önemi. 26. *Uluslararası Eğitim Bilimleri Kongresi: Antalya, Türkiye; 20/04/2017- 23/04/2017*
- Kalelioğlu, F., Gülbahar, Y. ve Kukul, V. (2016). A framework for computational thinking based on a systematic research review. *Baltic Journal of Modern Computing*, 4(3), s.583-596.
- Kandemir, C.M. (2018a). Metin tabanlı programlama. Y. Gülbahar (Ed.), *Bilgi İşlemsel Düşünmeden Programlamaya*, s. 267-292. Ankara: PEGEM Akademi.
- Kandemir, C.M. (2018b). Metin tabanlı programlama. Y. Gülbahar, H. Karal (Ed.), *Kuramdan Uygulamaya Programlama Öğretimi*, s. 299-336. Ankara: PEGEM Akademi.

- Karal, H., Şılbr, M. ve Yıldız, M. (2017). Kuramdan uygulamaya STEM eğitimi. S. Çepni (Ed.), *STEM Eğitiminde Bilişimsel Düşünme ve Kodlamanın Rolü* (389-409). 13.Bölüm. Ankara: Pegem Akademi.
- Karalar, H. (2019). Ortaokul öğretmenlerinin fiziksel programlamaya yönelik algıları ve deneyimleri. *Gazi Eğitim Bilimleri Dergisi*, 5(Özel Sayı), 130-145.
- Karasar, N. (2009). *Bilimsel Araştırma Yöntemi*. Ankara: Nobel Yayınları.
- Karataş, S., Güneş, E.A. ve Coşkun BK. (2012). Bilişim teknolojileri dersi ve ilköğretimde önemi. *6.Uluslararası Bilgisayar ve Öğretim Teknolojileri Sempozyumu: Gaziantep, Türkiye; 04/10/2012- 06/10/2012*
- Karim, M. E., Lemaignan, S.ve Mondada, F. (2015). A review: Can robots reshape K-12 STEM education?. *The 2015 IEEE International Workshop on Advanced Robotics and its Social Impacts* içinde s. 1-8. Lyon.
- Katıtaş, S. (2019). Karma yöntem araştırmalarına bütüncül bir bakış. *Social Sciences Students Journal*. 5 (49). 62-50-6260. doi: <http://dx.doi.org/10.264497sssj.1858>
- Keith, P. K., Sullivan, F. R. ve Pham, D. (2019). Roles, collaboration, and the development of computational thinking in a robotics learning environment. *In Computational Thinking Education* (s. 223-245). Singapore: Springer
- Keşf@ Bilinçli İnternet Hareketi-Öğretmen Portalı (2022). Eğitimler için bilgi işlemsel düşünme. Erişim Adresi: <http://acikders.kesfetprojesi.org/#/>
- Khine, M. S. (2017). *Robotics in STEM education: Redesigning the learning experience*. Erişim Adresi: <https://www.jstem.org/jstem/index.php/JSTEM/article/view/2311>
- Kılıç, S. (2022). Robotik programlamanın ön lisans öğrencilerinin bilgi işlemsel düşünme becerisi gelişimine etkisi. *Afyon Kocatepe Üniversitesi Sosyal Bilimler Dergisi*, cilt: 24, sayı:2 Haziran 2022, s. 480-494.

- Kim, S., Chung, K. ve Yu, H. (2013). Enhancing digital fluency through a training program for creative problem solving using computer programming. *The Journal of Creative Behavior*, 47, s.171-199.
- Kitzinger, J. (1995). Qualitative research: introducing focus groups, *British Medical Journal*, 311, s.299–302.
- Kitzinger, J. ve Farquhar, C. (1999). The analytical potential of ‘sensitive moments’ in focus group discussions. In R. Barbour ve J. Kitzinger (Eds), *Developing Focus Group Research: Politics, Theory and Practice*, s. 156–172. London: SAGE.
- Knowles, B. H., Finney, J., Beck, S. ve Devine, J. A. (2018). What children’s imagined uses of the bbc micro:bit tells us about designing for their iot privacy, security and safety. In *Living in the Internet of Things: Cybersecurity of the IoT IET Press*. <https://doi.org/10.1049/cp.2018.0015>
- Korkmaz, Ö., Çakır, R. ve Özden, M. Y. (2015). Bilgisayarca düşünme beceri düzeyleri ölçeğinin (BDBD) ortaokul düzeyine uyarlanması. *Gazi Eğitim Bilimleri Dergisi*, 1(2), s.67-86.
- Korkmaz, Ö., Çakır, R., Özden, M. Y., Oluk, A. ve Sarioğlu, S. (2015). Bireylerin bilgisayarca düşünme becerilerinin farklı değişkenler açısından incelenmesi. *Ondokuz Mayıs Üniversitesi Eğitim Fakültesi Dergisi*, 34(2), s.68- 87.
- Koorsse, M., Cilliers, C., ve Calitz, A. (2015). Programming assistance tools to support the learning of IT programming in South African secondary schools. *Computers & Education*, 82, 1s.62-178.
- Kotluk, N. ve Kocakaya, S. (2015). 21. Yüzyıl becerilerinin gelişiminde dijital öyküler: ortaöğretim öğrencilerinin görüşlerinin incelenmesi. *Eğitim ve Öğretim Araştırmaları Dergisi*, Mayıs 2015, cilt 4, sayı 2, makale no 36, s.354-363.

- Kukul, V. ve Karataş, S. (2016). Bilgisayar kullanmadan bilgisayar bilimi eğitimi: Öğretmen adaylarının görüşleri. *10th International Computer and Instructional Technologies Symposium (ICITS)*, Rize.
- Kurt, M. (2019). *STEM uygulamalarının 6. sınıf öğrencilerinin akademik başarılarına, problem çözme becerilerine ve STEM'e karşı tutumlarına etkisi üzerine bir araştırma. Gazi Üniversitesi, Eğitim Bilimleri Enstitüsü, Ankara, (Yüksek Lisans Tezi).*
- Kültekin, S. (2006). *Bilgi toplumu ve eğitim programları. Balıkesir Üniversitesi Sosyal Bilimler Enstitüsü, Eğitim Bilimleri Ana Bilim Dalı, (Yüksek Lisans Tezi).*
- Kürkçüoğlu, B.Ü. (2009). İletişim Becerilerinin Gelişimi. E. S. Batu (ed). *06-Yaş Arası Down Sendromlu Çocuklar ve Gelişimleri*. Kök Yayıncılık. Ankara.
- Krueger, R.A. ve Casey, M.A. (2000). Focus groups: a practical guide for applied research. *California: SAGE*.
- Kylonen, P.C. (2012). Measurement of 21st century skills within the common core state standards. *Paper presented at the Invitational Research Symposium on Technology Enhanced Assessments*, May 7-8.
- Lahtinen, E., Ala-mutka, K. ve Järvinen, H.-M. (2005). A study of the difficulties of novice programmers. *ACM SIGCSE Bulletin (C 37, s 14– 18). ACM*.
- Lambert, L. ve Guiffre, H. (2009). Computer science outreach in an elementary school. *Journal of Computing Sciences in colleges*, 24(3), s.118-124.
- Largo-Wight, E., Peterson, P. M., ve Chen, W. W. (2005). Perceived problem solving, stress, and health among college students. *American journal of health behavior*, 29(4), s.360-370.
- Lau, W. W. ve Yuen, A. H. 2011. Modelling programming performance: Beyond the influence of learner characteristics. *Computers & Education*, 57(1), s.1202-1213.

- Law, K. M., Lee, V. C. ve Yu, Y. T. (2010). Learning motivation in e-learning facilitated computer programming courses. *Computers & Education*, 55(1), s.218-228.
- Lee, I., Martin, F., Denner, J., Coulter, B., Allan, W., Erickson, J., Malyn-Smith, J. ve Werner, L. (2011). Computational thinking for youth in practice. *Acm Inroads*, 2(1), s.32-37. DOI: 10.1145/1929887.1929902
- Leech, N. L. ve Onwuegbuzie, A. J. (2009). A typology of mixed methods research designs. *Quality & Quantity*, 43(2), s.265-275.
- Lewis, M. (1995). *Focus group interviews in qualitative research: A review of the literature*, <http://www.scu.edu.au/schools/gcm/ar/arr/arrow/rlewis.html>
- Lye, S.Y. ve 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, s.51-61.
- Ma, X. L., Liu, J. J., Li, S., Fan, C. Y. ve Liang, J. (2019). Research on the curriculum design of the computer public course oriented to the cultivation of computational thinking ability. *Creative Education*, 10, s.3270-3285.
- Mains, M. G. (1997). The effects of learning a programming language on logical thinking skills. *Unpublished Doctoral Dissertation*. University of Nevada.
- Maden, S. (2021). Disiplinlerarasılık bağlamında ilk ve ortaöğretim programlarında türkçe algısı. *Disiplinlerarası Eğitim Araştırmaları Dergisi*, 5 (9), s.51-72. Erişim adresi: <https://dergipark.org.tr/tr/pub/jier/issue/64321/976466>
- Mano, C., Allan, V. ve Cooley, D. (2010). Effective in-class activities for middle school outreach programs. *In Frontiers in Education Conference*, 2010 IEEE.
- MEB-TTKB, (2018), *Bilişim teknolojileri ve yazılım dersi programı*, Ankara: Talim Terbiye Kurulu Başkanlığı.

Micro:bit (2022, Ocak). Eriřim adresi: <https://support.microbit.org/support/home>

Micro:bit Bileřenleri (2022, Nisan). Eriřim adresi: <https://microbit.org/get-started/user-guide/overview/>

Mozo, J. R., Quintero, H. M. ve Ariza, H. M. (2017). Educational robotics : Algorithm logic learning comparison. *International Journal of Applied Engineering Research*, 12(24), s.15470– 15474.

Nayef, E.G., Yaacob, N.R.N. ve Ismail, H. N. (2013). Taxonomies of educational objective domain. *International Journal of Academic Research in Business and Socail Sciences*. September 2013, Vol.3, NO:9, ISSN: 2222-6990

Nouri, J., Zhang, L., Mannila, L. ve Norén, E. (2019). Development of computational thinking, digital competence and 21st century skills when learning programming in K-9. *Education Inquiry*, s.1–17. Eriřim adresi: <https://www.tandfonline.com/doi/epub/10.1080/20004508.2019.1627844?needAccess=true>

NRC- National Research Council. (2012). Education for life and work: Developing transferable knowledge and skills in the 21st century. *National Academies Press*.

Numanoğlu, M. ve Keser, H. (2017). Programlama öğretiminde robot kullanımı- Mbot örneđi. *Bartın Üniversitesi Eğitim Fakültesi Dergisi*, Bartın. Vol: 6, Iss.2. S. 497-515. Doi: 10.14686/buefad.306198.

OECD, Organisation for Economic Co-operation and Development.

(Ekonomik Kalkınma ve İşbirliği Örgütü) (2022) – *The Future of Education and Skills 2030*. Eriřim adresi: <https://www.oecd.org/education/2030-project/>

Ozoran, D., Çağıltay, N. E. ve Topallı, D. (2012). Using scratch in introduction to programming course for engineering students. *2nd International Engineering Education Conference (IEEC2012)*, 2, s.125-132.

- Özçınar, H. (2018). Yeni piagetçi kurama göre bilişim teknolojileri öğretmen adaylarının soyutlama düzeylerinin ve programlama davranışlarının belirlenmesi. *Ankara Üniversitesi Eğitim Bilimleri Fakültesi Dergisi*, 2018, Cilt: 51, s. 1, 1-26.
- Özçınar, H. ve Öztürk, E., (2018). Hesaplamalı düşünmenin öğretimine ilişkin özyeterlilik algısı ölçeği: geçerlilik ve güvenilirlik. *Pamukkale Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, Ocak 2018, Sayı 30.
- Özdiç, F. ve Altun, A. (2014). Bilişim teknolojileri öğretmeni adaylarının programlama sürecini etkileyen faktörler. *İlköğretim Online*, 13 (4), s.1531-1541,2014. doi:10.17051/io.2014.54872
- Özdoğan, E. (2013). *Fiziksel olaylar öğrenme alanı için lego program tabanlı fen ve teknoloji eğitiminin öğrencilerin akademik başarılarına, bilimsel süreç becerilerine ve fen ve teknoloji dersine yönelik tutumlarına etkisi*. Dokuz Eylül Üniversitesi, İzmir (Yüksek Lisans Tezi).
- Özel, H. (2019). *Python programlama dili*. Erişim tarihi: Mart, 2022. Erişim adresi: <https://halilozel1903.medium.com/python-programlama-dili-1f1d88ef1e1d>
- Özel, O. (2019). *Programlama yöntemlerinin ortaokul öğrencilerinin bilgi işlemsel düşünme becerisine yönelik öz yeterlik algısına ve programlama başarısına etkisi*. Marmara Üniversitesi, İstanbul (Yüksek Lisans Tezi). Erişim adresi: <http://dspace.marmara.edu.tr/handle/11424/54695>
- Özer, F. (2019). *Kodlama eğitiminde robot kullanımının ortaokul öğrencilerinin erişim, motivasyon ve problem çözme becerilerine etkisi*. Hacettepe Üniversitesi, Eğitim Bilimleri Üniversitesi, Bilgisayar ve Öğretim Teknolojileri Eğitimi Ana Bilim Dalı, Ankara, (Yüksek Lisans Tezi).
- Özmen, B. ve Altun, A. (2014). Undergraduate students' experiences in programming: difficulties and obstacles. *Turkish Online Journal of Qualitative Inquiry*. 5(3), s. 9-27.

- Öztürk, M. (2021). Otantik programlama etkinlikleri: Öğrencilerin bilgi işlemsel düşünme ve programlama öz-yeterlilik inançlarına etkisi. *İnönü Üniversitesi Eğitim Fakültesi Dergisi*. Cilt 22, Sayı 2, 2021 ss. 1611-1640 DOI:10.17679/inuefd.773764
- Paf, M. (2019). *Ortaokul öğrencilerinin bilişimsel düşünme becerileri ile yaratıcı problem çözme becerileri arasındaki ilişki*. Aydın Adnan Menderes Üniversitesi, Sosyal Bilimler Enstitüsü, Aydın, (Yüksek Lisans Tezi).
- Pakman, N. (2018). *8-10 Yaş grubu öğrencilerine uygulanan temel düzey kodlama, robotik, 3d tasarım ve oyun tasarımı eğitiminin problem çözme ve yansıtıcı düşünme becerilerine etkisi*, Bahçeşehir Üniversitesi, Eğitim Bilimleri Enstitüsü, İstanbul, (Yüksek Lisans Tezi).
- Papert, S. (1993). *Mindstorms: Children, computers, and powerful ideas*. New York. New York: Basic Books.
- Papert, S. ve Harel, I. (2022). *Situating constructionism*. MIT Media Lab. Erişim adresi: https://web.media.mit.edu/~calla/web_comunidad/ReadingEn/situating_constructionism.pdf
- Park, D. ve Yoo, I. (2018). Python instructional model using robot for elementary school students. *Journal of The Korean Association of Information Education*, 22(3), s.357–366.
- Partnership For 21st Century Learning (P21), (2020). Erişim Adresi: <http://www.battelleforkids.org/networks/p21>
- Peixoto, A., Castro, M., Blazquez, M., Martin, S., Sancristobal, E., Carro, G. ve Plaza, P. (2018). Robotics tips and tricks for inclusion and integration of students. *In 2018 IEEE Global Engineering Education Conference (EDUCON)*. s. 2037–2041. IEEE. doi:10.1109/EDUCON.2018.8363487
- Przybylla, M. ve Romeike, R. (2014). Overcoming issues with students' perceptions of informatics in everyday life and education with physical computing. In Y. Gülbahar, E.

Karataş, & M. Adnan (Eds.), *Local proceedings of the 7th international conference on informatics in schools: Situation, evolution and perspectives*. Ankara University Press.- ISSEP 201. s. 9–22.

Radu, M. E., Cole, C., Dabacan, M. A., Harris, J. ve Sexton, S. (2011). The impact of providing unlimited access to programmable boards in digital design education, *IEEE Transactions on Education*, vol. 54, no. 2, s. 174-183, 2011.

Ramazanoğlu, M. ve Toytok, E. H. (2018). Teacher candidates' anxieties regarding facebook usage in education. *Journal of Education and Training Studies*, 6(n11a), s.229-235.

Ramazanoğlu, M. (2021). Robotik kodlama uygulamalarının ortaokul öğrencilerinin bilgisayara yönelik tutumlarına ve bilgi işelsel düşünme becerisine yönelik öz yeterlilik algılarına etkisi. *Türkiye Sosyal Araştırmalar Dergisi*. Yıl: 25, Sayı:1- Nisan 2021. s. 163-174.

Resinovic, B. (2015). The use of Nao, a humanoid robot, in teaching computer programming. *The International Conference on Informatics in Schools: Situation, Evolution and Perspectives—ISSEP* içinde, s. 63-64. Ljupljana.

Resnick, M. (2013). Reading, writing, and programming: Mitch Resnick at TEDxBeaconStreet. *TEDx BeaconStreet, Ideas in Action*. Erişim adresi: https://www.youtube.com/watch?v=42_30Rgf6F0

Richard, G. T. (2008). Employing physical computing in education: How teachers and students utilized physical computing to develop embodied and tangible learning objects. *The International Journal of Technology, Knowledge, and Society*, 4(3), 93–102. doi:10.18848/1832- 3669/CGP/v04i03/55887

Rogers, Y., Shum, V., Marquardt, N., Lechelt, S., Johnson, R., Baker, H. ve Davies, M. (2017). From the BBC micro to micro:bit and beyond: a British innovation. *Association for Computing Machinery*. Volume 24, Issue 2, DOI: 10.1145/3055204

- Romero, M. ve Dupont, Y. (2016). Educational robotics: from procedural learning to co-creative project oriented challenges with lego wedo. *8th Conference on Education and New Learning Technology, Barselona*.
- Rubio, M. A., Romero-Zaliz, R., Mañoso, C. ve de Madrid, A. P. (2014). Enhancing an introductory programming course with physical computing modules. *In 2014 IEEE Frontiers in Education Conference (FIE) Proceeding*, s. 1–8. IEEE. doi:10.1109/FIE.2014.7044153
- Sadik, O., Leftwich, A.O. ve Nadiruzzaman, H. (2017). Computational thinking conceptions and misconceptions: progression of preservice teacher thinking during computer science lesson planning. *In Emerging Research, Practice, and Policy on Computational Thinking*, s. 221–238. Cham: Springer International Publishing.
- Salerio, M., Carmo, B., Rodrigues J.M.F. ve du Buf J.M.H. (2013). A Low-Cost Classroom- Oriented educational robotics system. *International Conference on Social Robotics. Lecture Notes in Computer Science*.
- Saygıner, Ş. ve Tüzün, H. (2017). Programlama eğitiminde yaşanan zorluklar ve çözüm önerileri [The difficulties in programming education and suggestions for solutions]. I. *Uluslararası Bilgisayar ve Öğretim Teknolojileri Sempozyumu [International Computer and Instructional Technologies Symposium]*, Mayıs 2017.
- Saygıner, Ş. (2017). Blok tabanlı görsel ve metin tabanlı programlama öğretimlerinin erişil, mantıksal düşünme ve motivasyona etkileri. Hacettepe Üniversitesi, Bilgisayar ve Öğretim Teknolojileri Eğitimi Anabilim Dalı. Ankara, *(Yüksek Lisans Tezi)*.
- Sayın, Z. ve Seferoğlu, S. S. (2016). *Yeni bir 21. yüzyıl becerisi olarak kodlama eğitimi ve kodlamanın eğitim politikalarına etkisi*. Akademik Bilişim, Aydın, 30 Ocak-2 Şubat.
- Schmidt, A. (2016). Increasing computer literacy with the BBC micro:bit. Innovations in Ubicomp Products. Erişim adresi:

https://www.researchgate.net/publication/299570341_Increasing_Computer_Literacy_with_the_BBC_microbit

Schneider, G. M., ve Gersting, J. (2018). *Invitation to computer science 8th Edition: Cengage.*

Schunk, D. H. (2011). *Eğitimsel bir bakışla öğrenme teorileri (Learning theories and educational perspective)*. Çev. Şahin, M. Ankara: Nobel Akademik Yayıncılık.

Seferoğlu, S. S. (2015). *Okullarda teknoloji kullanımı ve uygulamalar: gözlemler, sorunlar ve çözüm önerileri*. Artı Eğitim, 123, 90-91. Erişim adresi: <http://www.egitimtercihi.com/okulgazetesi/17207-okullarda-teknoloji-kullan-m-ve-uygulamalar.html>

Sentance, S., Waite, J., Hodges, S., MacLeod, E. ve Yeomans, LE (2017). Creating cool stuff - pupils' experience of the BBC micro:bit. *In Proceedings of the 48th ACM Technical Symposium on Computer Science Education: SIGCSE 2017*, <https://doi.org/10.1145/3017680.3017749>

Sentance, S. ve Schwiderski-Grosche, S. (2012). Challenge and creativity: using .NET Gadgeteer in schools. In M. Knobelsdorf ve R. Romeike (Eds.), *Proceedings of the 7th Workshop in Primary and Secondary Computing Education*. s. 90–100.

Sinap, V. (2017). Programlama eğitiminde probleme dayalı öğrenmeye yönelik Arduino etkinliklerinin kullanılması: Bir eylem araştırması. *Yüksek lisans tezi, Süleyman Demirel Üniversitesi, eğitim bilimleri enstitüsü.*

Sullivan, A. ve Bers, M. U. (2017). Dancing robots: integrating art, music, and robotics in Singapore's early childhood centers. *International Journal of Technology and Design Education*, s.1-22.

Sönmez, V. (2000). *Öğretmenlik mesleğine giriş*. Ankara Anı Yayıncılık.

- Stager, G. S. (2007). *An Investigation of Constructionism in the Maine Youth Center. (Ph.D.), The University of Melbourne, Melbourne.*
- Sysło, M. M. ve Kwiatkowska, A. B. (2013). Informatics for all high school students: A computational thinking approach. In I. Diethelm ve R. T. Mittermeir (Eds.), *Informatics in schools: situation, evolution, and perspectives* (Vol. 7780, pp. 43–56). Heidelberg: Springer
- Şahin, M. ve Arslan Namlı, N. (2017). Algoritma eğitiminin problem çözme becerisi üzerine etkisi. *Recep Tayyip Erdoğan Üniversitesi Sosyal Bilimler Dergisi*, 3(5), s.135-153.
- Şahiner, A. ve Kert, S. B. (2016). Komputasyonel düşünme kavramı ile ilgili 2006-2015 yılları arasındaki çalışmaların incelenmesi. *EJOSAT: European Journal of Science and Technology, Avrupa Bilim ve Teknoloji Dergisi*, 5(9)- s.38-43.
- Şendurur, P. (2018). Bilişsel araçlar ve bilgi işlemsel düşünme. Y. Gülbahar (Ed.), *Bilgi İşlemsel Düşünmeden Programlamaya*, s. 79-98. Ankara: PEGEM Akademi.
- Sırakaya, D.A. (2019). Programlama öğretiminin bilgi işlemsel düşünme becerisine etkisi. *Türkiye Sosyal Araştırmalar Dergisi*. Yıl:23, Sayı:2 Ağustos 2019, s. 575-590.
- Şimşek, N. (2004). *Yapılandırmacı öğrenme ve öğretime eleştirel bir yaklaşım. Eğitim Bilimleri ve Uygulama*, 3,(5), s.115-139.
- Tabachnick, B. G. ve Fidell, L. S. (2012). *Using multivariate statistics (6 b.)*. Boston, MA: Pearson.
- Tashakkori, A. ve Creswell, J. (2007). The new era of mixed methods. *Journal of Mixed Methods Research*, 1 (1), s. 3-8.

Tatlı, C. E. (2013). *Piaget'e göre görelî kavramların kazanılması: 27 yıl sonra yeniden incelenmesi. Gazi Üniversitesi, Eğitim Bilimleri Enstitüsü, Ankara, (Yüksek Lisans Tezi).*

TDK (2021). Erişim Adresi: <https://sozluk.gov.tr/>

Totan, H.N. (2021). *Blok tabanlı kodlama eğitiminin ortaokul öğrencilerinin bilgi işlemsel düşünme becerileri ve kodlama öğrenimine yönelik tutumlarına etkisi: blocky örneği. Necmettin Erbakan Üniversitesi, Eğitim Bilimleri Enstitüsü, Bilgisayar ve Öğretim Teknolojileri Eğitimi Ana Bilim Dalı, (Yüksek Lisans Tezi).*

Toraman, S. (2021). Karma yöntemler araştırması: Kısa tarihi, tanımı, bakış açıları ve temel kavramlar. *Nitel Sosyal Bilimler*, 3(1), s.1-29.

Toytok, E. H. ve Çetin, A. (2015). The effects of pedagogical training program on students' professional attitudes and self efficacy levels. *The Turkish Online Journal of Educational Technology*, s.607-611.

Türk, H.B. (2019). *Öğrenci yanıt sistemlerinden Kahoot!'un üniversite öğrencilerinin İngilizce kelime öğrenme başarıları ve tutumları üzerindeki etkisi. Düzce Üniversitesi, Sosyal bilimler entitüsü, Eğitim Bilimleri Ana Bilim Dalı, (Yüksek Lisans Tezi).*

Türk, M. ve Kuzu, A. (2019). Fiziksel programlama. *programlama öğretimi yaklaşımları*. 1.Basım. Edt: Prof.Dr. Özgen Korkmaz. Nobel Yayın, s..99-137.

Uzun, A.ve Uz, R. (2018). Gömülü sistemler ve robotik uygulamalar dersine ilişkin öğrenen özellikleri ve görüşleri: bir öğretim tasarımına doğru. *Bursa Uludağ Üniversitesi Eğitim Fakültesi Dergisi* 31 (2), 2018, s. 533-559.

Üçgül, M. (2013). History and educational potential of LEGO Mindstorms NXT. *Mersin Üniversitesi Eğitim Fakültesi Dergisi*. Cilt:9, Sayı: 2, s. 127-137.

- Üçgöl, M. (2017). Eğitsel robotlar ve bilgi işlemsel düşünme. *Y. Gülbahar (Yay. haz.). Bilgi İşlemsel Düşünmeden Programlamaya, s. 295-317. Ankara: Pegem Akademi.*
- Üzümcü, Ö. ve Bay, E. (2018). Eğitimde yeni 21. yüzyıl becerisi: bilgi işlemsel düşünme. *Uluslararası Türk Kültür Coğrafyasında Sosyal Bilimler Dergisi (TURKSOSBİLDER) Cilt 03, Sayı 02, 2018, s.1-16.*
- Videnovik, M., Zdravevski, E., Lameski, P. ve Trajkovik, V. (2018). "The BBC Micro:bit in the international classroom: learning experiences and first impressions", *2018 17th International Conference on Information Technology Based Higher Education and Training (ITHET) , 2018, s. 1-5, doi: 10.1109/ITHET.2018.8424786.*
- Vrugt, A. ve Oort, F. J. (2008). Metacognition, achievement goals, study strategies and academic achievement: Pathways to achievement. *Metacognition and Learning.*
- WEF, 2020. World Economic Forum – Dünya Ekonomik Forumu. Erişim adresi: <https://www.weforum.org/reports/the-future-of-jobs-report-2020/>
- Weinberg, A. E. (2013). Computational thinking: An investigation of the existing scholarship and research (Order No. 3565573). Available from ProQuest Dissertations & Theses Global. Erişim adresi: <https://www.proquest.com/dissertations-theses/computational-thinking-investigation-existing/docview/1413309206/se-2?accountid=165780>
- Weintrop, D. ve Wilensky, U. (2017). Comparing block-based and text-based programming in high school computer science classrooms. *ACM Transactions on Computing Education (TOCE), 18(1), s.1-25.*
- Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society, 366, s.3717-3725. doi:10.1098/rsta.2008.0118.*

- Wing, J. M. (2010). Computational thinking: what and why? Center for computational thinking. 18.03.2022 Erişim adresi: <http://www.cs.cmu.edu/~CompThink/resources/TheLinkWing.pdf>
- Yadagiri, R. G., Krishnamoorthy, S. P. ve Kapila, V. (2015). A blocks-based visual environment to teach robot-programming to K-12 students. *The 2015 ASEE Annual Conference & Exposition* içinde, s. 26.17.1-26.17.11. Seattle.
- Yalçın, S. (2018). 21.Yüzyıl becerileri ve bu becerilerin ölçülmesinde kullanılan araçlar ve yaklaşımlar. Ankara Üniversitesi, Eğitim Bilimleri Fakültesi Dergisi, 2018, cilt: 51, sayı:1, s.183-201.
- Yeni, S. (2018). Bilişim teknolojileri ve bilgi-işlemsel düşünme. *Kert, S.B. (Ed). Bilişim Teknolojileri, 1.bölüm, s. 1-19*. Ankara: Nobel Yayınları.
- Yıldırım, B. ve Altun, Y. (2015). STEM eğitim ve mühendislik uygulamalarının fen bilgisi laboratuvar dersindeki etkilerinin incelenmesi. *El-Cezeri*, 2 (2). <https://dergipark.org.tr/tr/pub/ecjse/issue/4899/67132>
- Yıldız, T. (2019). Programlama öğretiminde akran öğretimi yönteminin öğrencilerin derse karşı tutum, akademik başarı ve programlama özyeterliklerine etkisi. *Necmettin Erbakan Üniversitesi, doktora tezi*.
- Yıldız, M., Çiftçi, E. ve Karal, H. (2017). Eğitim teknolojileri okumaları 2017, s.75-86 Editors: Odabaşı H. F., Akkoyunlu B., İşman A. Chapter: 5. Publisher: *Sakarya Üniversitesi-TOJET*, Ankara.
- Yolcu, V. (2018). Programlama eğitiminde robotik kullanımının akademik başarı, bilgi-işlemsel düşünme becerisi ve öğrenme transferine etkisi. *Süleyman Demirel Üniversitesi Eğitim Bilimleri Enstitüsü, Isparta, (Yüksek Lisans Tezi)*.
- Yurdakök, E.A. ve Kalelioğlu, F. (2020). Türkiye'de programlama öğretimi ile ilgili araştırmalardaki eğilimler: yüksek lisans ve doktora tezleri üzerine bir çalışma. 3. *Uluslararası Uzaktan Öğrenme ve Yenilikçi Eğitim Teknolojileri Konferansı*: Ankara, Çevrimiçi; 10/12/2020- 11/12/2020.

Yükseltürk, E. ve Altıok, S. (2015). Bilişim teknolojileri öğretmen adaylarının bilgisayar programlama öğretimine yönelik görüşleri. *Amasya Üniversitesi Eğitim Fakültesi Dergisi*, 4(1), s.50-65.

Zengin, M. (2016). İlkokul, Ortaokul ve Lise Öğrenciler Disiplinlerarası Eğitim & Öğretiminde Robotik Sistemlerinin Kullanımına Yönelik Görüşleri. *Üstün Zekalılar Eğitimi Araştırmaları Dergisi*. 2016, Cilt. 4 Sayı 2, s.48-70.



EKLER

EK 1: Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme Görevleri

Bilgi İşlemsel Düşünme Testi

Sevgili Öğrenciler,

Aşağıda yer alan Bilge Kunduz Uluslararası Enformatik ve Bilgi İşlemsel Düşünme Etkinliğinde 2020-2016 yılları arasında sorulmuş sorulardan bazıları bulunmaktadır. Sizlerden soruları içtenlikle cevaplamanız ve gerekli yere işaretlemeniz beklenmektedir.

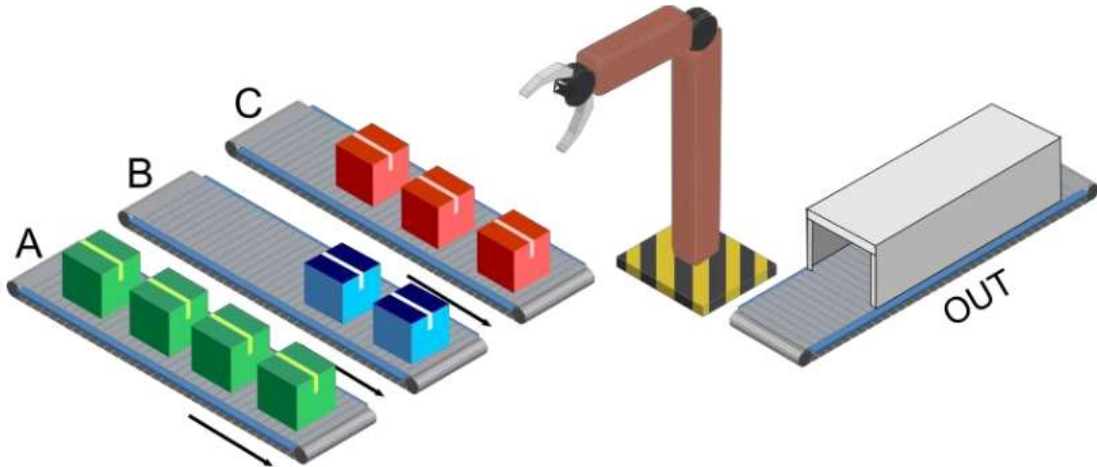
Başarılar dileriz.

1. Robot Kol

Bir robot kolu, nesneleri işleme taşıma bandına (OUT) taşımak için üç taşıma bandından (A, B ve C) nesneler alır. Robot kolu şu şekilde çalışır:

- önce A'dan bir nesne alır ve onu OUT'a taşır,
- sonra B'ye hareket eder, B'den bir nesne alır ve OUT'a taşır,
- son olarak, A ile tekrar başlamadan önce C ile aynı adımları gerçekleştirir (adım 1).

Taşıma bandına koyulacak bir nesne olmadığında robot kol bir nesne gelene kadar bekler, çünkü işlem ünitesinin ilerlemek için her bantta bir nesneye ihtiyacı vardır.



Soru

Yukarıdaki görselde gösterilen durumda, taşıma bantlarına (A, B ve C) yeni nesnelerin gelmeyeceği biliniyorsa, robot kol tarafından kaç nesne hareket ettirilecektir?

- A) 5 B) 6 C) 7 D) 8

2. Yanıp Sönen Işıklar

Programlanabilir bir elektronik kart aldınız ve onunla oynamaya başladınız. Bu kartta, üç LED (bir kırmızı, bir yeşil ve bir mavi) ve belirli bir ışık cihazı vardır, bunları bir programı açıp kapatarak kontrol edebilirsiniz (program başlamadan önce hepsi kapalıdır). İşte böyle küçük bir programa örnek:

REPEAT:

| turn_on (RED_LED);

| wait (1s);

| turn_off (RED_LED);

| wait (2s);

Bu program tarafından gerçekleştirilen eylemler aşağıdaki gibidir:

1. kırmızı LED'i açın,
2. 1 saniye boyunca hiçbir şey yapmadan bekleyin,
3. kırmızı LED'i kapatın,
4. 2 saniye boyunca hiçbir şey yapmadan bekleyin,
5. ve adım 1 ile tekrar başlayın.

Program çalıştırıldığında kırmızı LED bir saniye açık kalıp ardından iki saniye kapalı kalarak sonsuza kadar yanıp söner.

Soru

İnternette aşağıdaki programı buldunuz ve bunu denemek istiyorsunuz:

REPEAT:

| turn_on (BLUE_LED);

| wait (2s);

| turn_on (RED_LED);

| turn_on (GREEN_LED);

| wait (2s);

| turn_off (GREEN_LED);

| turn_off (BLUE_LED);

| wait (2s);

| turn_on (GREEN_LED);

| wait (2s);

| turn_off (RED_LED);

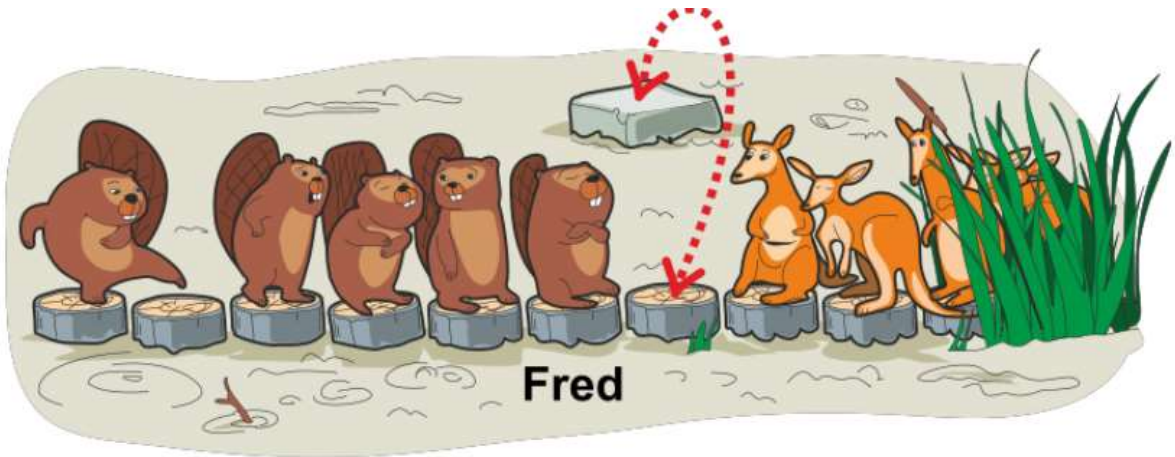
| turn_off (GREEN_LED)

Program başladıktan 13 saniye sonra kaç tane LED yanar?

- A) 0
- B) 1
- C) 2
- D) 3
- E) Rf

3. Kunduzlar Kangurulara Karşı

Beş kunduz kütük yolunu kullanarak bir bataklıktan geçerken zıt yöne giden bir grup kanguru ile karşılaşır. Kimse ıslanmak ya da kirlenmek istemiyor, böylece yolda kalıyorlar. Kangurular, belirli bir kütükten kütük yolunun yanındaki bir taşa atlayıp o kütüğe geri sıçramanın mümkün olduğunu keşfettiler. Bununla birlikte, bir kerede sadece bir kanguru taş üzerinde durabilir.



Kangurular ve kunduzlar, kangurularla ilk buluşan lider kunduz Fred hariç, geri dönmeyi umursamazlar. Fred sadece 10 kez geri adım atmak istiyor.

Soru

Fred'in davranışıyla, kaç kanguru geri çekilmeden onu geçebilir?

- A) Tam olarak 10 kanguru Fred'i geçebilir.
- B) Tam olarak 6 kanguru Fred'i geçebilir.
- C) Tam olarak 4 kanguru Fred'i geçebilir.
- D) 4'ten az sayıda kanguru Fred'i geçebilir.

4. Sihirbaz Melek

Sihirbaz Melek Sihirbaz Melek, üzerinde 49 adet jeton olan bir masanın önünde durmaktadır. Gözleri kapalı iken tek bir jetonu ters çevirmenize izin verir. Gözlerini açtığında hangi jetonun ters çevrildiğini hızlıca bulur. Bunu nasıl yaptığını size açıklar:

Her satır ve sütunda çift sayıda yıldız sembolü olan jeton vardır. Jetondan biri çevrildiğinde, o satır ve sütundaki yıldız sembolü olan jetonların sayısı tek olmaktadır. Böylece kesişim noktasındaki jetonun ters çevrildiği kolayca anlaşılr.



Soru

Şimdi sıra sizde! Aşağıdaki jetonlardan hangisinin ters çevrilmiş olduğunu bulunuz.

	A	B	C	D	E	F	G
1	★	★	★	★	★	★	★
2	★	★	★	★	★	★	★
3	★	★	★	★	★	★	★
4	★	★	★	★	★	★	★
5	★	★	★	★	★	★	★
6	★	★	★	★	★	★	★
7	★	★	★	★	★	★	★

A) B4

B) D6

C) E7

D) F2

5. Ayakkabı Boyutu

Ayakkabı Boyutu Kunduz Çınar, bir çift ayakkabı almak için ayakkabı dükkânına gitti. Resimde gösterildiği gibi ekranda birkaç ayakkabı gördü. Ayakkabılar, numara ve genişlik sırasına göre düzenlenmişti. Ayakkabılar, en küçük numara ve dar ayakkabılar sol altta, en büyük numara geniş ayakkabılar sağ üstte olacak biçimde düzenlenmişti. Tüm ayakkabıların farklı boy ve genişlikleri vardır. Unutkan bir kunduz olan Çınar, ayakkabı numarasını hatırlamıyordu. O yüzden doğru olanı bulana kadar ayakkabıları denemesi gerekecekti. Doğru bir uyum, doğru numara ve doğru genişlik demektir.



Genişlik	Dar ← ----- → Geniş					
Numara						
Büyük ↑						
•						
•						
•						
•						
•						
•						
↓ Küçük						

Ancak, Kunduz Çınar, 'n' denemede uygun ayakkabıyı bulmasını garanti eden bir yöntem kullanıyor.

Soru

‘n’ değeri en küçük kaç olabilir?

A) 2

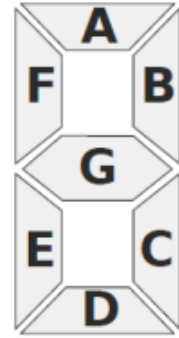
B) 3

C) 4

D) 5

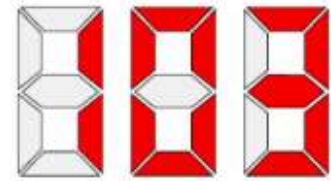
6. Dijital Sayı

Kunduz Kaya, ışık yayan diyotları (LED'ler) kullanarak sayıları görüntülemek istiyor. Her sayıyı temsil etmek için 7 LED parçası kullanabiliyor. Parçalar, yanda gösterildiği gibi A, B, C, D, E, F ve G olarak etiketlenmiştir.



Belirli bir LED parçasını açmak için, tabloda karşılık gelen hücreyi belirtmesi gerekir. Örneğin, aşağıdaki tabloda üç basamaklı sayı 103 gösterilecektir:

	A	B	C	D	E	F	G
↓							



Soru

Aşağıdaki tabloyu kullanırsak, ekranda hangi sayı görüntülenir?

[illegible]

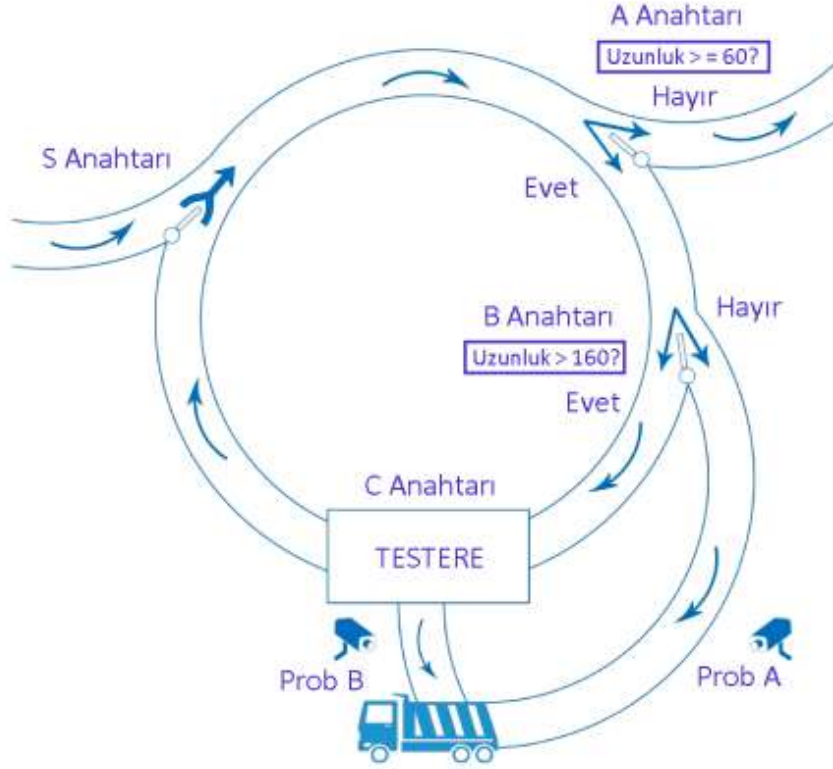
- A) 423958 B) 624785 C) 433968 D) 624958

7. Kereste Fabrikası

Kereste Fabrikası Bir kütük evi inşa etmek için doğru uzunlukta kütüklere ihtiyacınız vardır. Kütükler çeşitli uzunluklarda gelir ve aşağıdaki fabrika her bir tomruk kütlesini 60 ila 160 cm uzunluğunda bir kamyonu koyar. Kayıtlar fabrikanın soluna gelir. Fabrikanın farklı kısımları şekilde görülmektedir:

- S anahtarı otomatik olarak kütükleri iki akıştan birine öncelikli olarak yönlendirir.
- A anahtarı, uzunluğu ≥ 60 cm olan kütüklerin geçmesini ve kısa kütüklerin gönderilmesini sağlar.

- B anahtarı, uzunluğu > 160 cm olan kütüklerin geçmesine izin verir ve kalanları kamyonu gönderir.
- C anahtarı, kütüğü ikiye bölmek için testereyi çalıştırır. İlk 160 cm uzunluğunda olacak ve derhal kamyonu gönderecek. Kalan kütük diğer işlem için gönderilir.
- Prob A ve Prob B, kamyonu gönderilen kütük sayısını saymak için kullanılır.



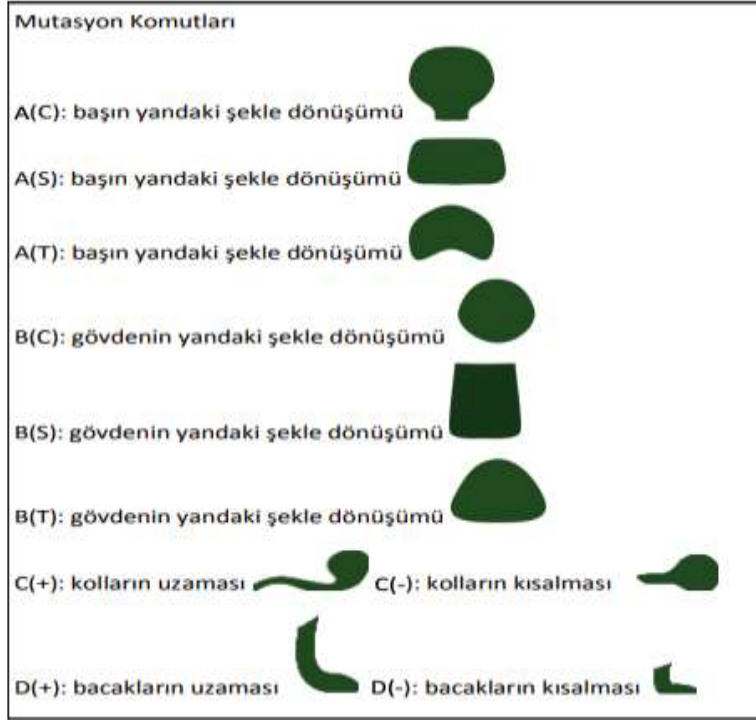
Soru

Fabrikaya üç farklı büyüklükte kütük (60 cm, 140 cm ve 360 cm) gönderilir. Tüm kütükler tamamen işlendiğinde, problemlerin saydığı değerler nelerdir?

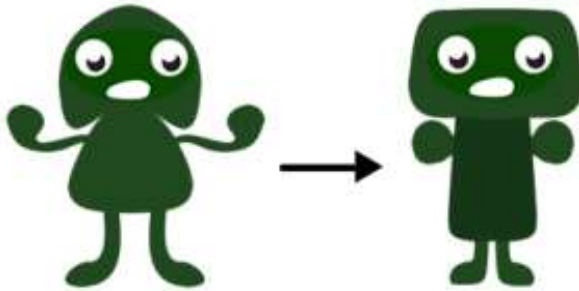
- A) Prob A: 1 kütük, Prob B: 3 kütük
- B) Prob A: 3 kütük, Prob B: 1 kütük
- C) Prob A: 2 kütük, Prob B: 2 kütük
- D) Prob A: 0 kütük, Prob B: 4 kütük

8. Bir Uzaylının Mutasyonu

Bir uzaylının bir başı, bir gövdesi, iki kolu ve iki bacağı bulunmaktadır. Uzaylı aşağıdaki mutasyon komutlarına göre dönüşüm geçirmektedir. Parçalar birden fazla dönüşüme uğrayabilmektedir.

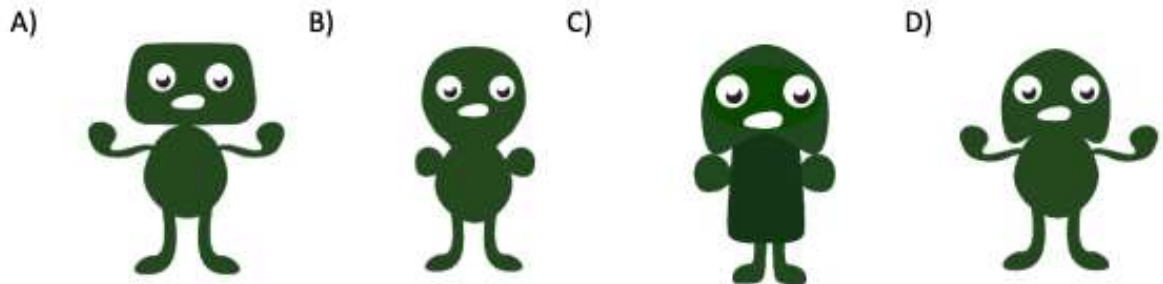


Örnek: A(S), B(S), C(-), D(-) kodları uygulandığında uzaylı aşağıdaki gibi bir dönüşüm yaşamıştır. Resimde uzaylının ilk ve ikinci görünüşleri gösterilmektedir:



Soru

Aşağıdaki komutlar sırasıyla ilk görünüme uygulandığında uzaylı nasıl bir görünüm kazanacaktır? A(T), C(+), B(T), C(+), A(C), C(-), B(C)



9. Tıbbi Laboratuvar

Tıbbi laboratuvardaki bir teşhis cihazı, hastalardan alınan örnekleri tekrar tekrar çalkalamalıdır. Cihaz, numaralı satırlarda yazılı olan bir bilgisayar programına göre çalışır. Cihaz, programı satır okur. Her zaman bir satır okur ve hemen çalıştırır. Satır "X numaralı satıra git" komutunu içeriyorsa, aygıt X satırına atlar ve okuyup çalıştırmaya devam eder. Program, bir A değerini saklayabilir, A'da saklanan sayıya 1 ekleyebilir ve A değerini bir başka sayıyla karşılaştırabilir.

Örnek bir algoritma aşağıda görülmektedir.

1. A olsun 0
2. A ekle 1
3. 6 numaralı satıra git
4. eğer A eşittir 60 ise 8 numaralı satıra git
5. A olsun 0
6. A ekle 1
7. 2 numaralı satıra git
8. A kere tekrarla örnekleri çalkala
9. son

Soru

Bu algoritmaya göre teşhis cihazı örnekleri kaç kez çalkalayacaktır?

- A) Örnekler asla çalkalanmayacaktır.
- B) Örnekler bir kez çalkalanır.
- C) Örnekler 60 kez çalkalanacaktır.
- D) Teşhis prosedürü, örneklerin çalkalanmasını durdurmaz.

10. Robo-Bahçıvan

Bilge Kunduz kendisine ağaç dikiminde yardım etmesi için bir robot satın almıştır. Robotun programlama dili aşağıdaki komutlardan oluşmaktadır:

Çalıştır: Robotu çalışır hale getirir.

İleri(X): Robotu X metre ileriye götürür.

Geri(X): Robotu X metre geriye götürür.

Sol(X): Robotu X derece sola döndürür.

Sağ(X): Robotu X derece sağa döndürür.

AgacDik: Bir ağaç diker.

Tekrar X {komutlar}: Süslü parantez içerisindeki komutları X defa tekrar ettirir.

Kapat: Robotu kapatır. Bahçenin çevresinde bulunan 16 farklı noktaya ağaç dikilecektir.

Bahçenin çevresinin her bir kenarı 8 metredir ve ardışık ağaçlar arasında 2 metre mesafe bulunmalıdır. Robot çalıştırıldığı zaman, başlangıç noktasından, yönü okun yönünde

olacak şekilde çalışacaktır. İlk durumda robot kapalıdır ve işlem bittiğinde robot kapatılmalıdır.

Ağaç dikimi gerçekleştikten sonra robotun yoluna çıkacak veya onu engelleyecek herhangi bir şey bulunmamaktadır.

Soru

Aşağıdaki programlardan hangisi, şekilde görüldüğü gibi, robotun bütün ağaçları dikmesini sağlar?

A) Çalıştır
Tekrar 4{
 Tekrar 4{AgacDik, İleri(2)},
 Sağ(90)}
Kapat

B) Başla
Tekrar 4{
 Tekrar 4{AgacDik, İleri(2)},
 Sol(90)}
Kapat

C) Çalıştır
Tekrar 4{
 Tekrar 4{İleri(1), AgacDik},
 Sağ(90)}
Kapat

D) Çalıştır
Tekrar 4{
 Tekrar 4{İleri(2), AgacDik},
 Sol(90)}
Kapat

11. Karma Fonksiyon

Bilgisayar bilimi dersinde, öğrenciler sayı listesi oluşturmak için fonksiyon yazmaktadır. Sayı listesi belirli bir sırada beş sayı tutmakta ve [a, b, c, d, e] şeklinde gösterilmektedir. Her fonksiyon bir sayı listesi ile çağrılır ve bunun sonucunda farklı sırada aynı beş sayıdan oluşan bir liste çıktısı geri döndürmektedir.

- Ayşe, ayse ([a, b, c, d, e]) adında bir fonksiyon yazar ve [e, b, c, d, a] değerlerini döndürür.
- Ahmet ise ahmet ([a, b, c, d, e]) adında bir fonksiyon yazar ve [e, d, c, b, a] değerlerini döndürür.

Öğretmen öğrencilerin fonksiyonlarını kontrol eder ve doğru çalıştığını görür. Öğretmen daha sonra karma ([a, b, c, d, e]) adında bir fonksiyon yazar ve bu fonksiyon ayse (ahmet (ayse ([a, b, c, d, e]))) değerini geri döndürür. Öğretmen öğrencilere karma fonksiyonun nasıl çalıştığını anlatır:

- Önce içteki fonksiyonu çalıştırın: ayse (ahmet (ayse ([a, b, c, d, e])))
- Şimdi iç fonksiyonu tekrar çalıştırın: ayse (ahmet ([e, b, c, d, a]))
- Şimdi ayse fonksiyonu çalıştırın: ayse ([a, d, c, b, e])
- Geri dönen liste : [e, d, c, b, a] şeklindedir der.

Soru

Öğretmen sınıfı test etmek için başka bir fonksiyon yazıyor. Test ([a, b, c, d, e]) şeklinde bir fonksiyon ahmet (karma (ayse ([a, b, c, d, e]))) değerini döndürdüğüne göre bu fonksiyonun değerleri aşağıdakilerden hangisidir?

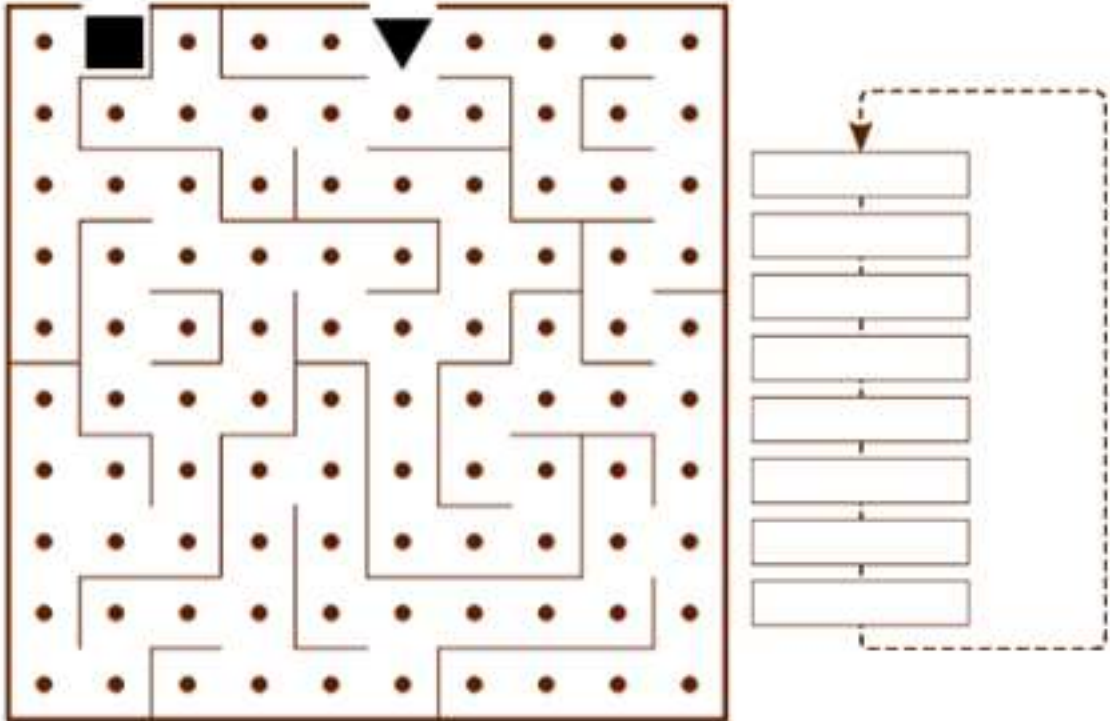
- A) [a, b, c, d, e]
- B) [e, b, c, d, a]
- C) [e, d, c, b, a]
- D) [a, d, c, b, e]

12. Labirentten Kaçış

Can labirentin üçgen ile gösterilen girişinden girip, kare ile gösterilen çıkışına ulaşmalıdır. Can bunu başarmak için aşağıda verilen komutları kullanacaktır.

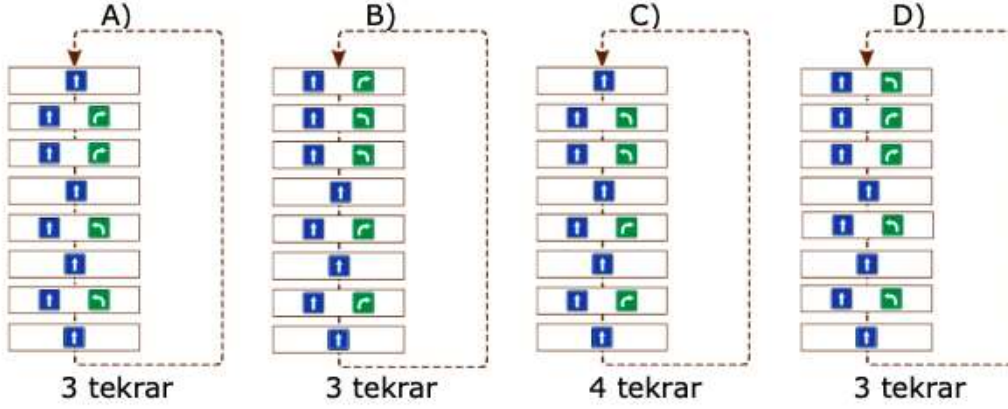
	Bir adım ileriye doğru adım at, sonra sola dön
	Bir adım ileriye doğru adım at, sonra sağa dön
	Bir adım ileriye doğru adım at

Can sadece sekiz komut dizisini aklında tutabilmekte ve bu diziyi birkaç kez tekrarladığında çıkışa ulaşabilmektedir.



Soru

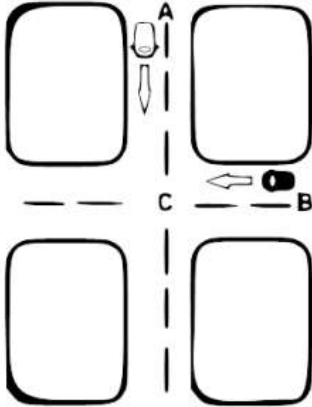
Can'ın çıkışa ulaşabilmesi için takip etmesi gereken komut dizisi ve bu diziyi kaç kez tekrarlayacağı hangi seçenekte doğru verilmiştir?



13. Arabalar

Bir kavşakta aşağıdaki kurallara göre araçlar hareket etmektedir.

- Beyaz arabalar için A noktasından C noktasına gitmek 3 dakika sürmektedir.
- Siyah arabalar için B noktasından C noktasına gitmek 2 dakika sürmektedir.
- Beyaz bir araba C noktasından geçtiğinde diğer bir beyaz araba A noktasından C noktasına gitmek için hareket etmektedir.
- Siyah bir araba C noktasından geçtiğinde diğer bir siyah araba B noktasından C noktasına gitmek için hareket etmektedir.
- Eğer araçlar kavşağa aynı anda varırlarsa önce siyah araba geçmektedir, beyaz araba ise 1 dakika beklemektedir.



Soru

Buna göre 12 dakikada C noktasından kaç beyaz ve kaç siyah araba geçer?

- A) 3 beyaz, 6 siyah
- B) 4 beyaz, 6 siyah
- C) 5 beyaz, 5 siyah
- D) 6 beyaz, 3 siyah

14. Pul Koleksiyonu

Bilge Kunduz gezdiği şehirlerden aldığı pullarla bir koleksiyon yapmaktadır. Bugün, koleksiyonunu yeniden inceleyerek bunları renk ve bölgelere göre yeniden sınıflandırmaya karar vermiştir.



Bilge Kunduz pulları sınıflandırırken yorulduğunu hissederek masanın üzerinde yaptığı sınıflandırmalara ilişkin notlar bırakarak uyumuştur. Bu sırada Bilge Kunduz'un yaramaz küçük kardeşi Bilgin Kunduz yapılan sınıflandırmaya bir pul eklemiştir. Pul eklendikten sonraki sonuç tablosu aşağıdaki gibidir.

	Kırmızı	Sarı	Yeşil	Mavi	Kahverengi	Toplam
Marmara	2	1	0	2	2	7
Doğu Anadolu	0	1	1	2	2	6
Karadeniz	1	2	3	0	1	6
İç Anadolu	0	1	2	1	0	4
Akdeniz	1	0	0	0	0	1
Güney Doğu Anadolu	0	2	1	1	0	4
Ege	0	1	0	0	2	3
Toplam	4	8	6	6	7	31

Soru

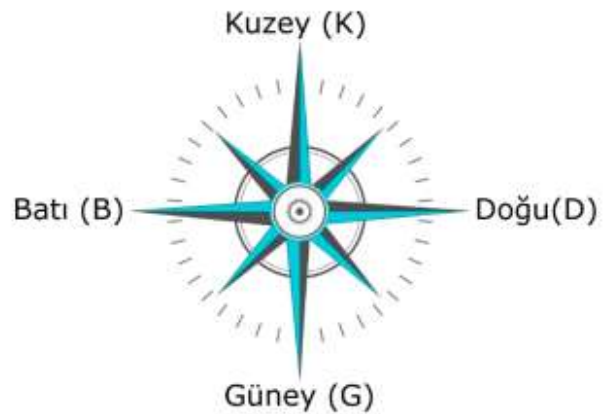
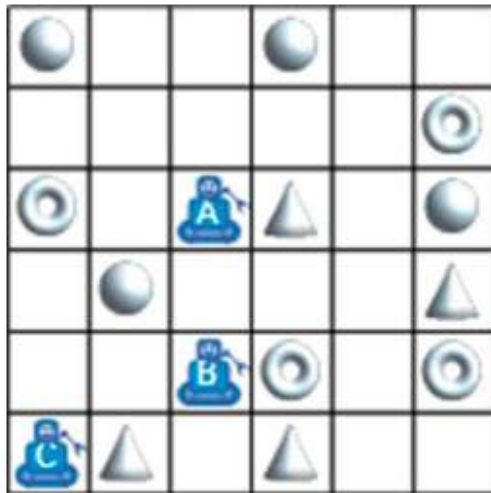
Bilgin Kunduz tarafından eklenen pula göre tablodaki hatalı bilgi nedir?

- A) Marmara bölgesindeki mavi renkli pul sayısı 1 olmalıdır.
- B) Güney Doğu Anadolu bölgesindeki sarı renkli pul sayısı 1 olmalıdır.
- C) Karadeniz bölgesindeki yeşil renkli pul sayısı 2 olmalıdır.
- D) Akdeniz bölgesindeki kırmızı renkli pul sayısı 0 olmalıdır.

15. Çılgın Robotlar

Bir depoda, A, B ve C robotları takım halinde çalışmaktadır. Bu robotlar depoda aynı yöne aynı anda hareket etmektedirler. Verilen yönleri takip eden robotlar depodaki cisimleri toplamaktadır.

Örneğin, verilen yönler K, K, G, G, D şeklinde olduğunda A robotu bir külah, B robotu yüzük ve C robotu da külah almaktadır.



Soru

A, B ve C robotlarının sırasıyla küre, yüzük ve külah alabilmesi için hangi yönlerin takip edilmesi gerekmektedir?

- A) K, D, D, D
- B) K, D, D, G, D
- C) K, K, G, D, K
- D) K, D, D, G, B



EK 2: Programlama Aracı ve Kavramlarını Öğrenim Düzeyi Ölçme Testi

1. Micro:bit elektronik kart içerisinde içerisindeki kodu yeniden başlatmaya yarayan bileşen hangisidir?

a) LED'ler b) B Butonu c) A Butonu d) Reset Butonu

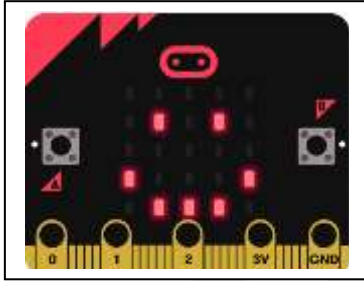
2. Micro:bit Python kullanılarak Micro:bit elektronik kartında çıktı olarak "Hello World" yazısını görmek istiyorsak aşağıdaki kod satırlarından hangisini yazmamız gerekir?

a) display("Hello World") b) display.scroll("Hello World") c) print("Hello World") d) not("Hello World")

```
while running_time() < 10000:  
    display.show(Image.ASLEEP)
```

3. Yandaki resimde verilen programda yer alan "10000" milisaniye kaç saniyeye eşittir?

a) 1 sn b) 100 sn c) 10 sn d) 1000 sn



4. Micro:bit Python kullanılarak Micro:bit elektronik kartının üzerinde yer alan resmin LED'ler üzerinde çıkması için aşağıdaki kod satırlarından hangisinin kullanılması gerekir?

a) display.show(Image.HAPPY)
b) display.import(Image.HAPPY)
c) display.print(Image.HAPPY)
d) display.while(Image.HAPPY)

5. Micro:bit Python kullanılarak micro:bit'te resimler adlı bir liste oluşturulacaktır. Aşağıdakilerden hangisi doğru bir liste tanımlamasıdır?

a) resimler = () b) resimler = [] c) resimler = { } d) resimler = < >

6. Micro:bit Python kullanılarak Micro:bit elektronik kartının kütüphanesinde bulunan tüm modüllere ulaşabilmek için aşağıdaki kod satırlarından hangisi kullanılır?

a) from microbit import* b) from microbit* c) import* d) microbit import*

7. Micro:bit Python kullanılarak Micro:bit elektronik kartının üzerindeki metni LED'ler üzerinde sürekli kaydırmak için aşağıdaki seçeneklerden hangisi kullanılır?

a) display () b) display.show () c) display.scroll () d) print ()

8. Micro:bit Python ile rastgele seçimler yapılması için hangi modül kütüphaneden çağırılmalıdır?

a) import b) import random c) import random: d) import random*

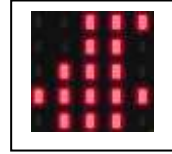
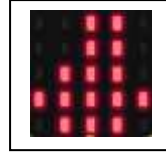
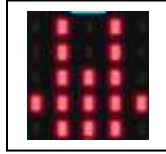
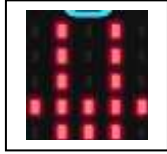
9. Micro:bit elektronik kartının üzerindeki A butonuna kaç kere basıldığı bilgisi aşağıdaki Micro:bit Python komutlarından hangisi kullanılarak öğrenilir?

- a) getpresses b) display.show c) display.scroll d) get_presses

10. Aşağıdakilerden hangisi ile Micro:bit Python’da sonsuz döngü başlatır?

- a) while True b) for c) while True: d) if

11. `display.show(Image("09090: 09090: 09090: 99999: 09990"))` kod satırı editörde çalıştırıldığında Micro:bit elektronik kartının LED’leri üzerinde aşağıdaki şekillerden hangisi ortaya çıkar?



a)
d)

b)

c)

12. Micro:bit elektronik kartı ivme ölçerken kullandığı eksenler aşağıdaki seçeneklerden hangisinde doğru sıralamayla verilmiştir?

- a) (q,w,x) b) (a,b,c) c) (y,x,q) d) (x,y,z)

```
1 from microbit import *
2
3 while True:
4     reading = accelerometer.get_x()
5     if reading > 20:
6         display.show("R")
7     elif reading < -20:
8         display.show("L")
9     else:
10        display.show("-")
```

- a) R b) L c) - d) LR

13. Yandaki programı inceleyiniz. Micro:bit elektronik kartı düz tutularak program çalıştırılırsa ekran çıktısı ne olur?

14. Resimdeki programda yer alan if, elif, else kodlarının görevi nedir?

- a) Koşul belirtir b) Döngü başlatır c) İvmeyi algılar d) Sonucu ekrana yazar

15. Micro:bit Python’da tanımlanmış bir listede elemanların rastgele seçilebilmesi için aşağıdaki seçeneklerden hangisi kullanılır?

- a) choice() b) random.choice() c) random() d) choice.random()

16. Micro:bit Python’da `display.scroll` komutunun LED’ler üzerinde çıktıyı göstermesi için çıktının veri türü aşağıdakilerden hangisi olmalıdır?

a) numeric ()

b) float()

c) str()

d) int()

```
1 from microbit import *
2
3 while True:
4     if button_a.is_pressed():
5         display.show(Image.HAPPY)
6     elif button_b.is_pressed():
7         break
8     else:
9         display.show(Image.SAD)
10
11 display.clear()
```

17. Yandaki resimde yer alan program çalıştığında Micro:bit elektronik kartı üzerinden B butonuna basılırsa sonuç ne olur?

- a) Döngüden çıkar
- b) LED'lerin üzerinde mutlu bir yüz belirir
- c) Döngü devam eder
- d) LED'lerin üzerinde üzgün bir yüz belirir

18. Resimdeki programda 11. satırda yer alan **display.clear()** komutunun görevi nedir?

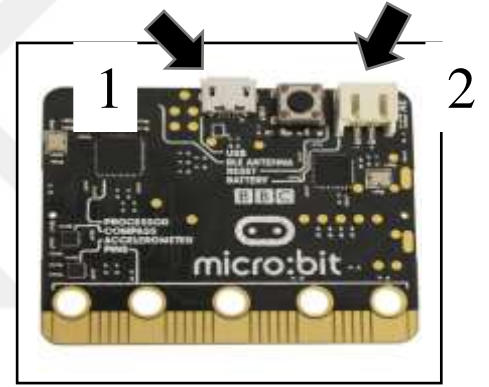
- a) LED'lerin üzerinde resim oluşturur
- b) Döngüden çıkarır
- c) Döngü başlatır
- d) LED'lerin üzerindeki çıktıyı temizler

19. Resimde, Micro:bit elektronik kartının üzerinde 1 numara ile gösterilen bileşen aşağıdakilerden hangisidir?

- a) Reset Butonu
- b) Bluetooth
- c) Micro USB
- d) Pil Girişi

20. Resimde, Micro:bit elektronik kartının üzerinde 2 numara ile gösterilen bileşen aşağıdakilerden hangisidir?

- a) Reset Butonu
- b) Bluetooth
- c) Micro USB
- d) Pil Girişi



21. Micro:bit elektronik kartında geçerli hareketin algılanması için aşağıdaki yöntemlerden hangisi kullanılır?

- a) accelerometer ()
- b) accelerometer.current
- c) accelerometer.current_gesture()
- d) gesture

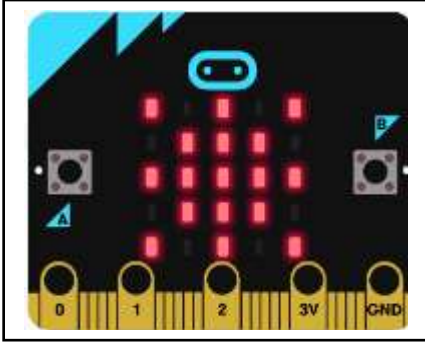
22. Micro:bit elektronik kartının üzerinde A butonuna basıldığında True değerini döndüren düğme işlevi aşağıdakilerden hangisidir?

- a) button_a.was_pressed()
- b) button_pressed
- c) button_a_pressed
- d) a_pressed

23. Micro:bit ile ilgili aşağıdaki bilgilerden hangisi yanlıştır?

- a) Micro:bit elektronik kartı harici olarak birçok bileşen ve modül ile kullanılabilir.
- b) Üzerinde A, B, C olmak üzere 3 tane buton bulunmaktadır.
- c) Bilgisayara Micro USB ile de bağlanabilir.
- d) Micro:bit elektronik kartı Python, Javascript, Makecode, Scratch gibi editörler ile kodlanabilir.

24. Micro:bit elektronik kartında LED’ler üzerinde resimdeki görseli oluşturan doğru kod sıralaması aşağıdakilerden hangisidir?



- a) `display.show(Image("99999: 99990: 99999: 09990: 90909"))`
- b) `display.show(Image("90909: 09990: 99999: 09990: 99999"))`
- c) `display.show(Image("99999: 09990: 99999: 09990: 90909"))`
- d) `display.show(Image("90909: 09990: 99999: 09990: 90909"))`

25. Micro:bit Python ile elemanları “Matematik, Fen, Bilişim Teknolojileri” olan “dersler” adında bir liste oluşturulmak istenmektedir. Aşağıdakilerden hangisi doğru bir liste tanımlaması olur?

- a) `dersler = ("Matematik, Fen, Bilişim Teknolojileri")`
- b) `dersler = ["Matematik, Fen, Bilişim Teknolojileri"]`
- c) `dersler = ["Matematik", "Fen", "Bilişim Teknolojileri"]`
- d) `dersler = ("Matematik", "Fen", "Bilişim Teknolojileri")`

EK 3: Haftalık Değerlendirme Tablolar ve Sonuçlar

Birinci Haftanın Değerlendirilmesi

Soru No	Sorular	Katılıyorum (%)	Ne katılıyorum / ne katılmıyorum (%)	Katılmıyorum (%)
1	"Python Programlama Dili Nedir?" tanımlayabilirim.	87.5	12.5	0
2	Micro:bit programlama aracının ne olduğunu açıklayabilirim.	100		0
3	Micro:bit bileşenlerini bilirim.	64.7	35.3	0
4	Micro:bit bileşenlerinin ne işe yaradığını açıklayabilirim.	64.7	35.3	0
5	Micro:bit ile bilgisayar bağlantısını tek başıma yapabilirim.	82.4	17.6	0

İkinci Haftanın Değerlendirilmesi

Soru No	Sorular	Katılıyorum (%)	Ne katılıyorum / ne katılmıyorum (%)	Katılmıyorum (%)	Öğrencilerin olumlu cevap verme yüzdesi (%)
1	Micro:bit ile bilgisayar bağlantısını tek başıma doğru bir şekilde bağlayabilirim.	88,9	11,1	0	
2	Micro:bit python editörünü tek başıma açabilirim.	94,4	5,6	0	
3	Micro:bit python'da kodlama yaparken aşağıdaki kod komutlarını kullanabilirim.				
		display.scroll()			94,1
		display.show()			94,1
		While True:			94,1
4	"# (hashtag) işareti programla ilgili açıklama yapıldığı anlamına gelir. Bir kod değildir." ifadesi için aşağıdakilerden hangisi doğrudur?				
		Doğru bir açıklamadır.			83,3
		Yanlış bir açıklamadır.			16,7

Üçüncü Haftanın Değerlendirilmesi

Soru No	Sorular		%
1	Python programlama dilinde bir liste oluşturabilirim;	Evet.	94,4
		Hayır.	5,6
2	Yazmış olduğumuz kod parçasının bir liste olduğunu aşağıdakilerden hangisiyle python anlayabilir?	Köşeli parantez ile yazılır.	72,2
		# işareti ile yazılır.	0
		Liste içindeki öğeler ";" (noktalı virgül) ile ayrılarak yazılır.	27,8
		Parantez içinde yazılır.	0
3	Micro:bit üzerinde yer alan input/output kavramlarının ne olduğunu bilirim?	Evet.	88,9
		Hayır.	11,1
4	Micro:bit üzerinde yer alan GND pininin ne işe yaradığını bilirim ve kodlayabilirim.	Evet.	72,2
		Hayır.	27,8

Dördüncü Haftanın Değerlendirilmesi

Hafta	Soru No	Sorular		%
4	1	Micro:bit üzerindeki A ve B butonlarını kodlaya bilirim.	Evet.	100
			Hayır.	0
	2	Aşağıda yer alan komutların anlamlarını bilir ve bir program içinde kullanabilirim;	button_a.was_pressed()	100
			button_b.was_pressed()	100
			sleep ()	94,4
			get_presses ()	94,4
			if something is True:	94,4
			elif some other thing is True:	83,3
			else:	94,4
			import random	88,9
			random.choice	83,3
			random.randint ()	44,4
			random.randrange()	38,9

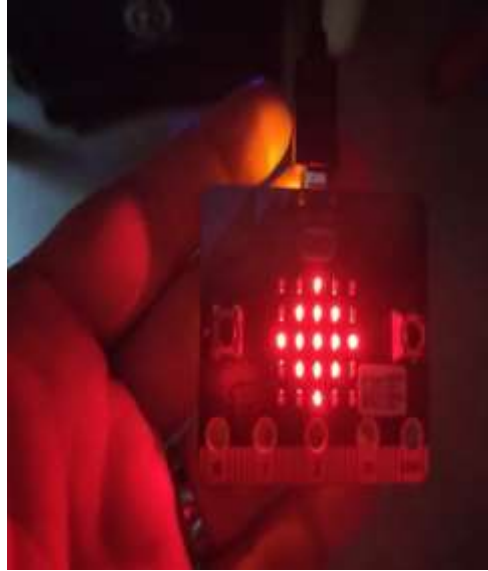
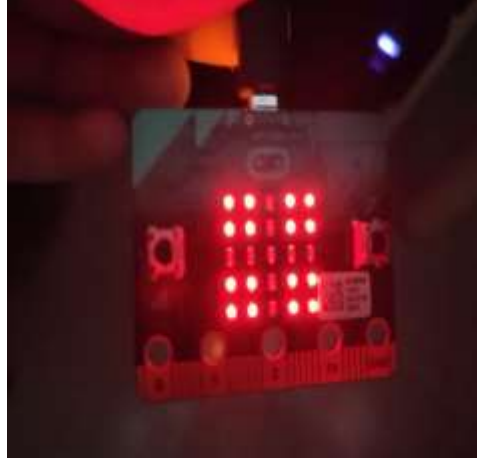
Beşinci Haftanın Değerlendirilmesi

Soru No	Sorular		%
1	Micro:bit üzerinde bir ivme ölçer olduğunu bilirim ve kod yazarken kullanabilirim.	Evet.	66,7
		Hayır.	33,3
2	Aşağıda yer alan komutları kullanabilirim;	Acceleration (mg)	85,6
		get_x	77,8
		accelerometer.current_gesture	72,2
		while True:	100
3	Micro:bit üzerinde yer alan accelerometer bileşeninin amacını bilirim.	Evet.	83,3
		Hayır.	16,7
4	Micro:bit ile kaç eksen ölçebiliriz?	2 eksen.	8,8
		3 eksen.	83
		4 eksen.	5
		5 eksen.	3,2

Altıncı Haftanın Değerlendirilmesi

Soru No	Sorular		%
1	Micro:bit ile python kodlama yaparken micro:bit üzerinde yer alan ledlerin parlaklığını ayarlayabilirim.	Evet.	100
		Hayır.	0
2	Micro:bit üzerinde kaç kaçlık LED vardır?	2*2	0
		3*3	0
		5*5	94,4
		10*10	5,6
3	Micro:bit üzerinde koordinat kodları yazabilirim.	Evet.	100
		Hayır.	0

EK 4: Ders Anlatım – Çalışma Görüntüleri



EK 5: Ders Gözlem Formu

Öğrencinin Adı Soyadı:

Değerlendirme Kriterleri	Micro:bit Aracı ve Python Programlama Yeterlik Gözlem Formu												
	Micro:bit aracının parçalarını bilebilme.	Micro:bit aracını doğru şekilde bilgisayara bağlayabilme.	MicroPython editörünü açabilme.	MicroPython editöründe kodlama yapabilme.	MicroPython editöründe çalışma adını değiştirebilme.	MicroPython editöründe python programlama diliyle kod parçası yazabilme.	MicroPython editöründe yazdığı kod parçasını micro:bit aracına yükleyebilme.	MicroPython editöründe yazdığı kod parçasını micro:bit aracında çalıştırabilme.	MicroPython editöründe yazdığı kod parçasında bir hatayla karşılaştığında çözebilme.	Karşılaştığı bir problemin daha önce karşılaştığı problemlere benzeyip benzemediğini ayırt edebilme. (Transfer yapabilme)	Yaratıcı programlar oluşturabilme.	Proje Oluşturabilme.	Ödevini Yapabilme.
1.Hafta													
2.Hafta													
3.Hafta													
4.Hafta													
5.Hafta													
6.Hafta													

EK 6 : Etik Komisyonu Onay Bildirimi

Evrak Tarih ve Sayısı: 08.12.2021-84787



1993

BAŞKENT ÜNİVERSİTESİ

Akademik Değerlendirme Koordinatörlüğü

Sayı : E-62310886-302.14.01-84787 Konu : Tez Önerisi (Ezgi Arzu Yurdakök)

EĞİTİM BİLİMLERİ ENSTİTÜSÜ MÜDÜRLÜĞÜNE İlgili : 22.11.2021 tarih ve 80029 sayılı yazınız.

Enstitünüz Bilgisayar ve Öğretim Teknolojileri Eğitimi Tezli Yüksek Lisans Programı öğrencisi Ezgi Arzu Yurdakök'ün, Doç. Dr. Filiz Kalelioğlu danışmanlığında yürütmeyi planladığı "Fiziksel Programlama Aracı Destekli Metin Tabanlı Programlama Öğretiminin Çeşitli Değişkenler Açısından İncelenmesi" başlıklı tez önerisi değerlendirilmiş ve bilgilerinize ekte sunulmuştur.

08.12.2021

Prof. Dr. M. Abdülkadir VAROĞLU Kurul Başkanı

Ek: Değerlendirme Formu

Sayı : 17162298.600-205
Konu : Tez Çalışması

26 KASIM 2021

İlgili Makama

Üniversitemiz Eğitim Bilimleri Enstitüsü Bilgisayar ve Öğretim Teknolojileri Eğitimi Tezli Yüksek Lisans Programı öğrencisi Ezgi Arzu Yurdakök'ün, Doç. Dr. Filiz Kalelioğlu danışmanlığında yürütmeyi planladığı "Fiziksel Programlama Aracı Destekli Metin Tabanlı Programlama Öğretiminin Çeşitli Değişkenler Açısından İncelenmesi" başlıklı tez önerisi değerlendirilmiş ve yapılmasında bir sakınca olmadığı tespit edilmiştir.

Bilgilerinize saygılarımızla sunarız.

Baskent Üniversitesi Sosyal ve Beşeri Bilimler ve Sanat Araştırma Kurulu

Ad, Soyad	Değerlendirme	İmza
Prof. Dr. M. Abdülkadir Varoğlu	Olumlu/ Olumsuz	
Prof. Dr. Kudret Güven	Olumlu/Olumsuz	
Prof. Ali Sevgi	Olumlu/Olumsuz	
Prof. Dr. Işıl Bulut	Olumlu/Olumsuz	
Prof. Dr. Sadegül Akbaba Altun	Olumlu/ Olumsuz	
Prof. Dr. Can Mehmet Hersek	Olumlu/ Olumsuz	
Prof. Dr. Özcan Yağcı	Olumlu/ Olumsuz	