

T.C.
ONDOKUZ MAYIS ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ EĞİTİMİ ANA BİLİM
DALI



PROGRAMLAMA EĞİTİMİNE BAŞLAMADA BLOK
TEMELLİ PROGRAMLAMA İLE METİN TEMELLİ
PROGRAMLAMANIN KARŞILAŞTIRILMASI

Yüksek Lisans Tezi

Yunus Emre ÇANAKCI

Danışman

Doç. Dr. Polat ŞENDURUR

SAMSUN
2022

TEZ KABUL VE ONAYI

Yunus Emre ÇANAKCI tarafından, **Doç. Dr. Polat ŞENDURUR** danışmanlığında hazırlanan “**PROGRAMLAMA EĞİTİMİNE BAŞLAMADA BLOK TEMELLİ PROGRAMLAMA İLE METİN TEMELLİ PROGRAMLAMANIN KARŞILAŞTIRILMASI**” başlıklı bu çalışma, jürimiz tarafından 29.8.2022 tarihinde yapılan sınav sonucunda oy birliği ile başarılı bulunarak Yüksek Lisans Tezi olarak kabul edilmiştir.

	Unvanı Adı Soyadı Üniversitesi Ana Bilim/Ana Sanat Dalı	İmza	Sonuç
Başkan	Dr. Öğr. Üyesi Neşet MUTLU Kayseri Erciyes Üniversitesi Okul Öncesi Ana Bilim Dalı		<input checked="" type="checkbox"/> Kabul <input type="checkbox"/> Ret
Üye	Dr. Öğr. Üyesi Hatice Gökçe BİLGİÇ DOĞAN Ondokuz Mayıs Üniversitesi Bilgisayar ve Öğretim Teknolojileri Ana Bilim Dalı		<input checked="" type="checkbox"/> Kabul <input type="checkbox"/> Ret
Üye	Doç. Dr. Polat ŞENDURUR Ondokuz Mayıs Üniversitesi Bilgisayar ve Öğretim Teknolojileri Ana Bilim Dalı		<input checked="" type="checkbox"/> Kabul <input type="checkbox"/> Ret

Bu tez, Enstitü Yönetim Kurulunca belirlenen ve yukarıda adları yazılı jüri üyeleri tarafından uygun görülmüştür.

ONAY

... / ... / ...

Prof. Dr. Ali BOLAT

Enstitü Müdürü

BİLİMSEL ETİĞE UYGUNLUK BEYANI

Hazırladığım Yüksek Lisans tezinin bütün aşamalarında bilimsel etiğe ve akademik kurallara riayet ettiğimi, çalışmada doğrudan veya dolaylı olarak kullandığım her alıntıya kaynak gösterdiğimi ve yararlandığım eserlerin Kaynaklar'da gösterilenlerden oluştuğunu, her unsurun enstitü yazım kılavuzuna uygun yazıldığını ve TÜBİTAK Araştırma ve Yayın Etiği Kurulu Yönetmeliği'nin 3. bölüm 9. maddesinde belirtilen durumlara aykırı davranılmadığını taahhüt ve beyan ederim.

Etik Kurul Gerekli mi?

Evet (Gerekli ise ekler kısmına ekleyiniz)

Hayır

İmza

10/10/2022

Yunus Emre ÇANAKÇI

TEZ ÇALIŞMASI ÖZGÜNLÜK RAPORU BEYANI

Tez Başlığı: PROGRAMLAMA EĞİTİMİNE BAŞLAMADA BLOK TEMELLİ PROGRAMLAMA İLE METİN TEMELLİ PROGRAMLAMANIN KARŞILAŞTIRILMASI

Yukarıda başlığı belirtilen tez çalışması için şahsım tarafından 22.06.2022 tarihinde intihal tespit programından alınmış olan özgünlük raporu sonucunda;

Benzerlik oranı : % 11

Tek kaynak oranı : % 3 çıkmıştır.

İmza

10/10/2022

Doç. Dr. Polat ŞENDURUR

ÖZET

PROGRAMLAMA EĞİTİMİNE BAŞLAMADA BLOK TEMELLİ PROGRAMLAMA İLE METİN TEMELLİ PROGRAMLAMANIN KARŞILAŞTIRILMASI

Yunus Emre ÇANAKCI
Ondokuz Mayıs Üniversitesi
Lisansüstü Eğitim Enstitüsü
Bilgisayar ve Öğretim Teknolojileri Eğitimi Ana Bilim Dalı
Yüksek Lisans, Ağustos/2022
Danışman: Doç. Dr. Polat ŞENDURUR

Bu araştırmada ortaokul öğrencilerinin Bilişim Teknolojileri ve Yazılım dersi kapsamında aldıkları doğrusal programlama eğitimlerinin sonuçlarına odaklanılmıştır. Araştırmada Ağrı-Patnos Değirmendüzü Ortaokulu 6 ve 7. Sınıf öğrencileri ile çalışılmıştır. Ülkemizde ortaokul kademesindeki tüm öğrencilere programlama becerileri blok tabanlı programlama araçları ile kazandırılmaktadır. Ancak blok tabanlı programlama eğitimi ile programlama becerilerinin temellerini oluşturan öğrencilerin metin tabanlı programlamaya geçişte bu becerilerinin yeterli olmayacağı varsayılmaktadır. Araştırmada odaklanılan problem durumu; blok tabanlı programlama araçları ile temel programlama eğitimi alan öğrencilerin metin tabanlı programlamanın mantığını ve çalışma prensiplerini anlamak, uygulamak konusunda bazı problemler yaşama olasılığıdır. Blok tabanlı programlama araçlarının doğrusal programlama mantığının dışında, belli açılardan doğrusal olmayan programlama mantığını içermesi doğrusal programlama mantığına göre çalışan metin tabanlı programlama araçlarının kullanımında problemler ortaya çıkarmaktadır. Buna göre tezin amacı temel programlama becerilerini blok tabanlı programlama eğitimi ile alan öğrenciler ile temel programlama eğitimini metin tabanlı programlama araçları ile alan öğrencilerin ileri düzey programlama eğitimleri sonucunda elde edecekleri beceriler arasındaki farkın ortaya çıkarılmasıdır.

Araştırmaya göre gruplar deney ve kontrol grubu olarak ikiye ayrılmıştır. Gruplar 12 öğrenciden oluşmaktadır. 6. Sınıf kademesinde çalışmaya dahil olan bu iki grubun okulun ve bireysel imkanların kısıtlı olmasından dolayı 5. Sınıf kademesinde bilişim teknolojileri ile ilgili herhangi bir ders almamışlardır. Gruplar çalışmanın başlangıcına kadar henüz programlama eğitimi almamış kişilerden oluşmaktadır. 6. Sınıf kademesinde kontrol grubu blok tabanlı programlama eğitimi alırken deney grubu blok tabanlı programlama araçlarını hiç görmeden metin tabanlı programlama etkinliklerine başlamıştır. Etkinlikler sonucunda öğrencilerden programlama kavramlarını tanımlamaları istenmiştir. Analizler öğrenciler arasında bir farklılığa işaret etmemiştir. Çalışma, iki gruba da metin temelli daha ileri düzey programlama eğitimleri verilmesi şeklinde devam etmiştir. Gerçekleştirilen başarı testi sonucunda programlama eğitimine blok temelli programlama eğitimi almadan doğrudan metin temelli programlama etkinlikleri ile başlayan grubun programlama alanında akademik ve pratik olarak daha başarılı olabileceği, temel programlama becerilerini metin tabanlı araçlarda hayata geçirebilme konusunda blok tabanlı eğitimi ile başlayan gruba göre daha başarılı oldukları ortaya çıkmıştır. Bu sonuçlara programlama becerileri değerlendirme ölçeği ile ulaşılmıştır.

Anahtar Sözcükler: Blok tabanlı kodlama, Metin tabanlı kodlama, Scratch, Python,
Ortaokul öğrencileri



ABSTRACT

COMPARISON OF BLOCK-BASED PROGRAMMING AND TEXT-BASED PROGRAMMING FOR BEGINNING TO PROGRAMMING EDUCATION

Yunus Emre ÇANAKCI
Ondokuz Mayıs University
Institute of Graduate Studies

Department of Computer Education and Instructional Technology Computer
Master, August/2022

Advisor: Assoc. Prof. Dr. Polat ŞENDURUR

In this research, the results of linear programming trainings taken by secondary school students within the scope of Information Technologies and Software course are focused. In the research, Ağrı-Patnos Değirmendüzü Secondary School 6 and 7. Classroom students were studied. In Türkiye, all students at the secondary school level are provided with programming skills with block-based programming tools. However, it is assumed that the text-based programming skills of the students who form the foundations of programming skills with block-based programming education will not be sufficient. The focus of the research is on the problem situation; it is the fact that students who receive basic programming education with block-based programming tools have some problems in understanding and applying the logic and working principles of text-based programming. The fact that block-based programming tools contain nonlinear programming logic other than linear programming logic creates problems in the use of text-based programming tools that work according to linear programming logic. Accordingly, the aim of the thesis is to reveal the difference between the group that receives basic programming skills with block-based programming education and the real software skills of the groups that receive basic education with text-based programming tools.

According to the research, the groups were divided into two as experimental and control groups. The groups consist of 12 students. 6. Due to the fact that these two groups involved in the work in the grade cadaver have limited school and individual opportunities, 5. The course could not be processed at the grade level. The groups consist of people who have not yet received programming training until the beginning of the study. At the 6th grade level, the control group received block-based programming training, while the experimental group started text-based programming activities without ever seeing the block-based programming tools. According to the first concept evaluation scale, there is no significant difference between the groups. After the studies, it was revealed that the experimental group that started programming training with text-based programming activities without receiving block-based programming training could be more successful in real software, and that they were more successful than the group that started with block-based training in terms of implementing basic programming skills in text-based tools. These results were reached by the programming skills assessment scale.

Keywords: Block-based coding, Text-based coding, Scratch, Python, Middle school students

ÖNSÖZ VE TEŞEKKÜR

Bu çalışmanın öncesinde tez aşamasına gelene kadar her konuda destek ve yol göstermesinin yanı sıra, tez aşamasında her problemime çözüm üreten, saat ve gün fark etmeksizin görüşme sağlayan danışmanım Sayın Doç. Dr. Polat ŞENDURUR hocama emek ve anlayışı için teşekkürlerimi sunarım.

Yunus Emre ÇANAKCI



İÇİNDEKİLER

TEZ KABUL VE ONAYI	i
BİLİMSEL ETİĞE UYGUNLUK BEYANI	ii
TEZ ÇALIŞMASI ÖZGÜNLÜK RAPORU BEYANI	ii
ABSTRACT	v
ÖNSÖZ VE TEŞEKKÜR	vi
İÇİNDEKİLER	vii
SİMGELER VE KISALTMALAR	viii
ŞEKİLLER DİZİNİ	ix
TABLolar DİZİNİ	x
1. GİRİŞ	1
1.1. Amaç	2
1.2. Problem Durumu	2
1.3. Araştırma Soruları	3
1.4. Çalışmanın Önemi	3
1.5. Sınırlılıklar	4
1.6. Varsayımlar	5
1.7. Tanımlar	5
2. KURAMSAL ÇERÇEVE	7
2.1. Pedagojik Açından Programlama Eğitimi.....	11
2.1.1. Pedagojik Yaklaşımlar	14
2.2. Programlama Eğitimindeki Sorunlar	21
3. YÖNTEM	28
3.1. Araştırma Deseni.....	28
3.2. Ölçme Araçları	29
3.2.1. Kavram Değerlendirme Ölçeği	29
3.2.2. Programlama Becerileri Ölçeği.....	30
3.3. Katılımcılar	31
3.4. Süreç	31
3.5. Veri Analizi	34
3.6. Araştırmacının Rolü.....	35
4. BULGULAR	36
4.1. Kavram Değerlendirme	36
4.1.1. Döngü	36
4.1.2. Fonksiyon	37
4.1.3. Eğer Sorgusu.....	38
4.1.4. Eğer İse Değilse	39
4.1.5. Ve-Veya.....	40
4.1.6. Değişken.....	40
4.1.7. Başlatıcı	41
4.1.8. Doğrusal, Doğrusal Olmayan Programlama	42
4.2. Programlama Becerileri Değerlendirme	43
5. SONUÇ, TARTIŞMA VE ÖNERİLER	47
KAYNAKÇA	47
EKLER	57
Öz Geçmiş	69

SİMGELER VE KISALTMALAR

BTP	: Blok Tabanlı Programlama
MTP	: Metin Tabanlı Programlama
TPE	: Temel Programlama Eğitimi
MEB	: Millî Eğitim Bakanlığı
TTK	: Talim Terbiye Kurumu



ŞEKİLLER DİZİNİ

ŞEKİL 2.1. SCRATCH PROGRAMI ÖRNEK UYGULAMA.....	25
ŞEKİL 2.2. BAYRAĞA TIKLANDIĞINDA	25
ŞEKİL 2.3. TUŞ İLE BAŞLAMA	26
ŞEKİL 2.4. KUKLA ÜZERİNE TIKLAMA İLE BAŞLAMA.....	26
ŞEKİL 2.5. DEKOR DEĞİŞİMİ İLE BAŞLAMA	26
ŞEKİL 2.6. SES VE ZAMANLAYICI SORGUSU İLE BAŞLAMA	26
ŞEKİL 2.7. HABERLEŞME İLE BAŞLATMA.....	27



TABLolar DİZİNİ

TABLO 2. 1 347 SAYILI KARAR İLE İLKÖĞRETİM HAFTALIK BİLGİSAYAR DERSİ SAATLERİ.....	7
TABLO 2. 2. PROGRAMLAMA DERS İÇERİKLERİNİN BLOOM TAKSONOMİSİNE GÖRE ÖRNEK KAZANIMLARI.....	12
TABLO 2. 3. PROGRAMLAMA DERS İÇERİKLERİNİN DUYUŞSAL ALAN ÖRNEK KAZANIMLARI .	14
TABLO 2. 4. PROGRAMLAMA ÖĞRETİMİNDE YAŞANAN ZORLUKLARIN NEDENLERİ.....	18
TABLO 3. 1. KATILIMCI SAYISI.....	31
TABLO 4. 1. DÖNGÜ KAVRAMINA AİT İFADELER	37
TABLO 4. 2. FONKSİYON KAVRAMINA AİT İFADELER.....	37
TABLO 4. 3. EĞER SORGUSU KAVRAMINA AİT İFADELER.....	38
TABLO 4. 4. EĞER İSE DEĞİLSE KAVRAMINA AİT İFADELER.....	39
TABLO 4. 5. VE OPERATÖRÜNE AİT İFADELER.....	40
TABLO 4. 6. VEYA OPERATÖRÜNE AİT İFADELER	40
TABLO 4. 7. DEĞİŞKEN KAVRAMINA AİT İFADELER	41
TABLO 4. 8. DEĞİŞKEN OLUŞTURMA KURALLARINA AİT İFADELER	41
TABLO 4. 9. BAŞLATICI KAVRAMINA AİT İFADELER	41
TABLO 4. 10. DOĞRUSAL PROGRAMLAMA KAVRAMINA AİT İFADELER.....	42
TABLO 4. 11. DOĞRUSAL OLMAYAN PROGRAMLAMA KAVRAMINA AİT İFADELER.....	42
TABLO 4. 12. DEĞİŞKEN KURALI YAZMA VE ÖRNEK VEREBİLME, DOĞRU DEĞİŞKEN SEÇİMİ (SORU 1-5)	43
TABLO 4. 13. ARİTMETİK OPERATÖRLER VE ANLAMLARI (SORU 2).....	44
TABLO 4. 14. İLİŞKİSEL OPERATÖRLER VE ANLAMLARI (SORU 3)	44
TABLO 4. 15. ATAMA OPERATÖRLERİ VE ANLAMLARI (SORU 4)	45
TABLO 4. 16. KODLARIN ÇIKTISINI ALABİLME (SORU 6 – SORU 7).....	45
TABLO 4. 17. EKRANA YAZDIRMA KOMUTUNU KULLANMA (SORU 8- SORU 9- SORU 10)	45
TABLO 4. 18. \n KOMUTU VE \t KOMUTUNUN GÖREVİ	46
TABLO 4. 19. VERİLEN PROBLEMİN ÇÖZÜM KODLARINI YAZMA.....	46

1. GİRİŞ

Ortaokul kademelerinde programlama eğitimi bilgisayarsız programlama ve blok tabanlı programlama (BTP) ile başlamaktadır. BTP eğitimi sözdizimi kurallarından arındırılmış, hazır kod bloklarının birbirine eklenmesi mantığı ile yapılan programlama etkinliklerinden oluşmaktadır. BTP ortamları yazım kurallarının detaylarını içermeyen, günlük konuşma dilinde programlama yapılmasını sağlayan programlama ortamlarıdır. Scratch, Mblock, Code.org gibi platformlar bunlara örnektir. Bu ortamlar sınıf düzeylerine göre farklı zorluk derecelerine olanak sağlamaktadır. Blok tabanlı programlama temelde basit komutlar ile karakter yönlendirme gibi görevleri içerisinde barındırır da mobil uygulamalar, masaüstü uygulamalar, oyunlar gibi ürünler ortaya çıkarmaya da imkân sağlamaktadır. Metin tabanlı programlama (MTP) ortamları, farklı programlama dillerinin farklı syntax kurallarını barındıran, farklı koşturma platformları gerektiren, programlama dillerine göre kendi kurallarını barındıran ortamlardır. MTP günlük konuşma dilini içerisinde barındırmamaktadır. BTP de hazır kod bloklarının eklenmesi temel programlama yöntemi iken MTP de tüm kodlar yazılımcı tarafından yazılmak zorundadır.

Çalışmada BTP’de bulunan ancak metin tabanlı programlamada (MTP) karşılığı bulunmayan kavram yanılgılarına değinilmiştir. BTP eğitimi alan öğrencilerin MTP eğitimine geçişte öğrenilen kavramların tam anlamı ile karşılığının bulunmaması halinde öğrenilen kavramların yerine yenisi yerleştirilememektedir. Çalışmada öğrencilerin BTP eğitimi almadan doğrudan MTP eğitimi alması, BTP eğitiminden MTP eğitimine geçişin doğurabileceği kavram yanılgılarının önüne geçeceği ve gerçek programlama çalışmalarında daha başarılı sonuçlar elde edebileceği varsayılmaktadır. Belirlenen iki gruptan birine BTP eğitimi yıllık plana uygun zamanda verilmektedir. Diğer gruba ise aynı tarihte başlayacak şekilde doğrudan MTP eğitimi verilmeye başlanmıştır. MTP eğitimi ileri seviye olmayacak şekilde temel programlama becerileri üzerinde planlanmıştır. Ortaokul 6. Sınıf kademesinde yapılacak uygulamanın sonucunda bir üst sınıfa geçen gruplara yıllık plan ile paralel olacak tarihte ön test uygulanacaktır. Ön testte MTP eğitimi alanların daha başarılı çıkacakları varsayılmıştır. Ön test temel programlamaya ait kavramları içeren açık uçlu sorulardan oluşmaktadır. Gruplar ön test sonrasında paralel olacak şekilde MTP eğitimine başlayacaktır. Eğitimin sonunda son test uygulanarak “BTP ile programlama eğitimine başlayan grup ile MTP ile programlama eğitimine başlayan grup arasında anlamlı fark

var mıdır?" sorusuna cevap verecektir. Çalışmada MTP ile programlama eğitimine başlayan grubun daha başarılı sonuçlar elde edeceği varsayılmıştır.

1.1. Amaç

Kodlama eğitimi çocukluk döneminden itibaren verilmeye başlanmaktadır. Kodlama eğitiminde temel eğitim basamaklarının başında içerik olarak Scratch uygulaması gösterilmektedir. Çocuklar üzerinde düşünme, öz-düzenleme, keşfetme becerilerinin geliştirilmesine katkıda bulunduğu ortaya çıkmaktadır. Kodlama içeriğinin çocukluk döneminde verilecek olması içeriklerin bu kademedeki öğrencilere uygun olması gerekmektedir. Programlamanın karmaşık düzeninin dışında temel amaç olarak problemi tanıma, analiz yapma ve yaratıcılık kısımlarına odaklanması gerekmektedir. MIT tarafından geliştirilen Scratch yazılımı; küçük yaş grupları için karmaşıklıktan uzak, ticari amaç gütmeyen(ücretsiz) ve öğrenciler için eğlenceli hale getirecek şekilde tasarlanmıştır (Kert & Uğraş, 2009).

Ancak kodlama eğitiminde kullanılan Scratch uygulamasının blok tabanlı kodlama mantığının metin tabanlı kodlama mantığı ile bire bir örtüşmediğini düşünülmektedir. Blok tabanlı programlama ortamında yürütülen kod yazma süreçleri, blok tabanlı programlamada yer alıp metin tabanlı programlama ortamında tam karşılığı olmayan komut, işlemler ve aynı zamanda bu ortamların doğrusallığı arasındaki farklılıklar yukarıda bahsedilen varsayımın temelini oluşturmaktadır. Bu bağlamda çalışmanın temel amacı ilgili varsayımı destekleyen ya da bu varsayımın geçerliliğini açıklamaya yarayacak sonuçları ortaya koymaktır. Bu durumun temel programlama eğitimi süreçlerinde, eğitimin doğal akışını bozmadan gerçekleştirilmesi hedeflenmektedir.

1.2. Problem Durumu

Ortaokul kademesi öğrencilerinin Bilişim Teknolojiler ve Yazılım dersi kapsamında eğitimini aldığı programlama eğitimi farklı içerikler ile verilebilmektedir. 5. Sınıf ile temel eğitime başlanan programlama eğitimi, blok temelli programlama araçları ile verilmektedir. 6. Sınıf kademesinde blok temelli programlama araçlarından ağırlıklı olarak Scratch, Code.org, Mblock kullanılmaktadır. 7. Sınıf kademesine geçildiğinde müfredat metin temelli programlama araçlarını zorunlu kılacak içeriklere sahiptir. Bu nedenle blok temelli programlama araçlarından metin temelli programlama araçlarına geçiş sağlanmaktadır. Tez çalışmasına konu olan problem durumu bu aşamada ortaya çıkmaktadır. Programlama mantığının temellerini blok

temelli programlama araçları ile öğrenen öğrencilerde metin temelli programlamaya geçiş yaptığında bazı kavramsal yanlışlar yaşanmaktadır. Blok temelli programlama ile metin temelli programlama araçları arasında birbiri ile örtüşmeyen kısımlar bulunmaktadır. Örnek olarak; blok temelli programlamada tek bir kodlama ekranında birbirinden ayrı, yan yana, ayrı başlatıcılar barındıran kod blokları oluşturulabilirken metin temelli programlamada tek bir kodlama alanında kodlar alt alta yazılması; farklı başlatıcılar mümkün olmasına rağmen, mantığının blok temellide bulunandan oldukça farklı olması verilebilir. Bunu gibi durumlar tez çalışmasına konu olan blok temelli programlamadan metin temelli programlama geçişte yaşanan kavramsal yanlışların öğrencilerin programlama becerilerine olumsuz etkileri olacağı problem durumunu ortaya koymaktadır.

1.3. Araştırma Soruları

Çalışmanın amacı blok tabanlı ve metin tabanlı ortamlarda temel programlama eğitimi alan öğrencilerin ileri programlamaya geçişte yaşadıkları farklılıkları ortaya çıkarmaktır. Bu bağlamda iki gruba ayrılan öğrencilerden birinci gruba blok, ikinci gruba ise metin tabanlı temel programlama eğitimi verilmiştir. Devamında ise iki grupta yer alan öğrencilerin tamamıyla metin temelli programlama eğitimleri ile devam edilmiş ve ders kapsamındaki başarıları test edilerek karşılaştırılmıştır. Bu gruplar metin tabanlı programlama (MTP) grubu ve blok tabanlı programlama (BTP) grubu olarak adlandırılan gruplar ile aşağıdaki sorulara cevap aranmaya çalışılmıştır;

1. Temel programlama eğitimi sonucunda MTP grubu ile BTP grubu arasında programlama kavramlarının tanımlanması açısından bir fark var mıdır?
2. İleri programlama eğitimi sonucunda MTP grubu ile BTP grubu arasında programlamaya yönelik akademik başarı farkı var mıdır?

1.4. Çalışmanın Önemi

Tüm bilim dallarında başlangıcın yani temelin doğru ve sağlam olması alandaki bireysel başarıda oldukça etkili olduğu bilinmektedir (Arabacıoğlu, Bülbül, & Filiz, 2007). Bu nedenle programlama eğitimine başlangıcın oldukça önemli olduğu bilinmektedir. Dünya Ekonomik Forumu'nun yayınladığı "Mesleklerin Geleceği" raporunda 2025 yılına kadar en çok ihtiyaç duyulacak beceriler arasında programlama becerileri de yer almıştır (Zurnacı & Turan, 2022). Çalışma sonucuna göre programlama becerilerinin temellerinin nasıl atılacağına öngörü oluşturmuştur.

Programlama becerilerine sahip bireylerin sayısının artması açısından çalışma önem kazanmıştır. Ülkemizde ortaokul öncesinde bilgisayarsız programlama eğitimleri verilmektedir. Ortaokul kademesinde bilgisayarlı programlama eğitimleri verilmektedir (MEB, 2018).

Ortaokul kademesi programlama eğitiminde BTP çalışmaları yürütülmektedir. Temel Programlama Eğitiminde (TPE) blok tabanlı programlama eğitimi tamamlanmasının ardından MTP eğitime geçilmektedir. Bu çalışmaya yön veren temel hipoteze göre BTP eğitiminin MTP eğitimi ile örtüşmeyen kavramsal farklılıklar bulunmaktadır ve bu kavram farklılıkları MTP da ilerlemeyi yavaşlatmaya öğrenme problemlerine temel teşkil eden kavram yanılgılarına sebep olmaktadır. Çalışmada yer alan gruplardan biri eğitim programlarında ön görülen geleneksel eğitime devam ederek BTP eğitimini alırken, diğer grup BTP eğitimini almadan MTP üzerinden tasarlanan temel programlama eğitimine giriş yapacaktır. Çalışma iki çalışma ortamı arasındaki detaylara odaklanmasından dolayı önem arz etmektedir. Çalışma temel programlama eğitiminin dışında olan bu uygulama ile programlama eğitimine yeni bir bakış açısı getirecektir. Temel programlama eğitimi açısından ilk ve orta kademeli eğitim kurumlarındaki programların gözden geçirilmesine dair öneriler sunma potansiyeli bulunmaktadır. Öğrencilerin BTP ile öğrenmeye başladıkları programlama, algoritmik düşünme ve problem çözme becerilerinin çalışma sonucuna göre MTP ile öğrenmeye başlamaları halinde daha başarılı sonuçlar elde etme ihtimalleri bulunmaktadır. Buna göre programlama eğitimi programlarında değişime gidilmesine ve yeni içerikler geliştirilmene neden olacak sonuçların ortaya çıkması olasılıklar arasındadır.

1.5. Sınırlılıklar

- Araştırma 2020-2021 ve 2021-2022 eğitim öğretim yılında Ağrı ili Patnos ilçesinde Değirmendüzü köyünde bulunan Değirmendüzü Ortaokulu ile sınırlıdır.
- Örneklem 6. Sınıf kademesinde ve çalışma devamında 7. Sınıf olan öğrenciler ile sınırlıdır.
- BTP ortamı Scratch ile sınırlıdır
- MTP ortamı Python ile sınırlıdır.

1.6. Varsayımlar

Araştırmanın varsayımları;

- Araştırma kapsamında tercih edilen Scratch programlama ortamının BTP ortamlarının özelliklerini yansıtan bir programlama ortamı olduğu varsayılmaktadır.
- Araştırma kapsamında tercih edilen Python programlama ortamının MTP ortamlarının özelliklerini yansıtan bir programlama ortamı olduğu varsayılmaktadır.
- Araştırma grubunda bulunan iki grubun akademik başarılarının benzer olduğu varsayılmıştır.
- Çalışmada kullanılan ölçeklerin hedeflenen değişkenleri doğru şekilde ölçtüğü varsayılmaktadır.

1.7. Tanımlar

Scratch: Kullanıcıların etkileşimli projeler oluşturabilmesine olanak sağlayan programlama ortamıdır (Maloney, Resnick, Rusk, Silverman, & Eastmond, 2010). MIT'nin geliştirdiği proje 2003 yılında başladı, 2007 yılında halka ücretsiz olarak sunuldu. Farklı dil tercihlerine olanak sağlaması oldukça fazla ilgi görmesine sebep oldu. Projenin temel amacı programlama bilgisi bulunmayan kişilere programlamayı tanıtmaktır. Bu amaç doğrultusunda metin tabanlı programlama tercih edilmemiş bunu yerine görsel blok dili tercih edilmiştir.

Python: Nesne yönelimli ve etkileşimi yüksek programlama dilidir. Basit söz dizimi sayesinde öğrenimi oldukça kolaydır. Kısa zamanda vakit kaybettiren detaylar olmadan programlama yapılmasına olanak sağlamaktadır. Genel olarak tüm platformlarda çalışabilmektedir. Tasarımcısı Guido Van Rossum tasarladığı programlama diline sevdiği komedi grubunun bir gösterisinin adını vermiştir.

Programlama Dili: Bilgisayarın fonksiyonel beyni olmadığı için bilgisayara özgü komutlar yazılmalıdır. Bilgisayara özgü komutlara programlama dili denir. Farklı isimlerde ve özelliklerde programlama dilleri mevcuttur. Birçok ortak özellikleri olmasına rağmen kendine özgü özelliklerde barındırmaktadır.

Blok Temelli Programlama: Hazır kod parçalarının bulunduğu programlama araçlarının amacı kullanılan kod blokları ile bilgisayar ve kullanıcı arasında köprü

görevi görmektir. Günlük dilde ekle çıkar mantığı ile oluşturulan kod bütünlerini bilgisayarın anlayacağı biçimde çevirerek programlamayı sağlar.

Metin Temelli Programlama: Kelimelerle ve komutlarla yapılan programlamadır. Her programlama dilinin kendine özgü olan söz dizim kurallarına uygun olarak kullanılan kodlarla programlama sağlanmaktadır.

Algoritma: Belirli bir problemi çözmek için gerekli olan sıralı mantıksal adımlara denir. Algoritmalar sonlanmak zorundadırlar ve tüm ihtimalleri ele alabilecek kadar geniş çerçeveye sahip olmalıdırlar.

Akış Şeması: Algoritmaları ifade etmek için kullanılan evrensel şekillerdir.



2. KURAMSAL ÇERÇEVE

Avrupa ile Türkiye’de bilgisayar kullanımının başlangıcı 1980 yıllarına dayanmaktadır. Bilgisayarın donanım olarak ülkede yaygınlaşması yeterli görülmemiştir. MEB tarafından 1991 yılında okullarda teknolojinin etkin kullanılması, teknoloji kullanabilen bireylerin arttırılması amacıyla üniversite eğitim fakültelerinde bilgisayar formatör öğretmenliği eğitimleri vermeye başlanmıştır (Akpınar & Altun, 2014). Eğitim içeriği olarak; bilgisayar kullanımı, eğitim yazılımları seçme ve değerlendirme, teknolojinin eğitimde kullanılması konularına yer verilmiştir. Ancak verilen formatör eğitimlerinin yeterli olmayacağı görüşü nedeniyle ilk ve orta öğretimde görev yapacak öğretmenleri yetiştirmek amacıyla Bilgisayar öğretmenliği anabilim dalı açılması gerektiği düşünülmüştür. İlk olarak ODTÜ’de Fen Bilimleri Eğitimi Bölümü altında Bilgisayar Öğretmenliği anabilim dalı kurulmuş ve 1997 fakülte yapılandırılması ardından anabilim dalının adı Bilgisayar ve Öğretim Teknoloji Eğitimine dönüştürülmüştür.

Papert programlama öğretiminde öğrenenlerin düşünme ve soyutlama becerilerinin geliştiğini belirtmiştir (Papert, 1980). 28.08.2006 tarih ve 347 sayılı karar ile TTK başkanlığınca kabul edilen İlköğretim Seçmeli Bilgisayar (1-8) Dersi Öğretim Programı içerisine eklenmiştir (MEB, 2006). Seçmeli bilgisayar (1-8) dersi öğretim programı; 3 temel yapı üzerine kurulmuştur. Bunlar hedef, öğrenme-öğretme süreçleri ve değerlendirmedir. Bu 3 temel yapı sayesinde daha düzenli ve sistematik öğrenme ortamı sağlanacağı düşünülmüştür. 2007 yılında bilgisayar dersi isim değiştirerek “Bilişim Teknolojileri” adını almıştır. Derse ayrılan süre aşağıdaki tabloda açıklanmıştır.

Tablo 2. 1 347 sayılı karar ile İlköğretim haftalık bilgisayar dersi saatleri

Sınıf	Haftalık Ders Saati
1-2-3	1 Saat
4-5	2 Saat
6-7-8	1 Saat

Ders saatleri Avrupa’ya kıyasla ilgili dönemde iki katı fazla durumdadır. Ancak öğretim ortamlarındaki eksiklikler, derse hazırlık ve diğer etkinliklerle ders süresinin bir kısmının harcanmasıyla kaybedilen zaman Avrupa’dan fazla olan ders sayısını anlamlı olmaktan çıkarmaktadır (Seferoğlu, 2007).

2006 yılında programa dâhil edilen Bilişim Teknolojileri dersi, 05.09.2012 tarih ve 150 sayılı karar ile yenilenmiş ve Ortaokul ve İmam Hatip Ortaokulu Bilişim Teknolojileri ve Yazılım Dersi (5-8) adını almıştır. Program içerisinde dört temel yeterlilik belirlenmiştir. Bunlar;

- Bilişim okur yazarlığı
- BT kullanarak iletişim kurma, bilgi paylaşma, kendini ifade etme
- Araştırma, iş birlikçi çalışma
- Problem çözüme, programlama ve özgün ürün geliştirme (MEB, 2012)

Bilişim teknolojileri dersi bilgisayar okuryazarlığı içerikleri ve kazandırdıkları beceriler ile öğrencileri saf tüketici olmalarının önüne geçmeyi hedeflemiştir. Öğrenenleri tüketen olmaktan çıkarıp üreten bireyler haline getirmek bu programla hedeflenen temel amaçlardan biridir.

5. ve 6. sınıf Bilişim Teknolojileri ve Yazılım dersi kazanımlarında programlama konularına da yer verilmiştir. Programlama öğretimi farklı disiplinleri içerisinde barındırmaktadır. Öğrencilerin temel becerilerinin güçlenmesine ve bilgi işlemsel düşünme becerilerini günlük hayatta kullanmalarına katkı sağlayacağı öngörülmektedir. Programlama temel olarak insan ile makinelerin iletişimini sağlayan söz dizilerini doğru biçimde yazma işlemidir ve ilk olarak askeri amaçlarla ortaya çıkmıştır (Gülbahar & Karal, 2018). Basit olarak programlama bilgisayarın donanımlarına ne yapması gerektiğinin söylenmesi durumudur. Zaman zaman programlama ve kodlama kavramları birbiri yerine kullanılabilir. Ancak programlama algoritma ve kodlama işlemlerinin genel adıdır. Yani programlama bu iki kavramı da içerisinde barındırmaktadır.

Algoritma bir sorunu çözmek ya da bir amaca ulaşmak için tasarlanan yol, işlem basamaklarına denir. Programlamanın temelini algoritma kavramı oluşturmaktadır. Algoritmanın ilk örneklerine Harezmi tarafından MS. 780 yılında yazılan kitaplarda rastlanmaktadır (Aytekin, Sönmez Çakır, Yücel, & Kulaozü, 2018). El-Harezmi'nin "Hisab-el Cebir ve El Mukabala" kitabında ilk algoritma örneklerine rastlanmaktadır. Alt alta yazılan işlem adımlarının evrensel kabul görmüş şekiller ile sembolize edilmesine akış diyagramı adı verilmektedir. Akış diyagramları, algoritması oluşturulan işlem adımlarını görselleştirmek amacıyla kullanılır.

Programlamanın kapsayıcı olduğu diğer alan ise kodlamadır. Kodlama; problem çözümlerinin algoritma olarak yazılması, devamında akış diyagramına dönüştürülmesinin ve tüm bu işlemlerin ardından programlama dillerine çevrilmesidir. Programlama dilleri bilgisayar ile insanlar arasında çevirmen olarak görev yapmaktadır. Bilgisayar yalnızca 0 ve 1 rakamlarından oluşan makine kodlarını anlamaktadır. Programlama dilleri kendi söz dizimi kuralları çerçevesinde insanların anlayabileceği şekilde oluşturduğu komutları 0 ve 1 rakamları ile ifade edilen makine diline çeviren araçlardır. Her bir işlemci için 0 ve 1'lerden oluşan kombinasyonların farklı anlamları olabilmektedir. O nedenle çeviriciler (programlama dilleri) için kurulum esnasında işlemci seçimi gerektiren bazı diller bulunmaktadır. 0-1 kombinasyonları bilgisayarın elektronik aksamaları ile alakalıdır. Görüldüğü gibi bilgisayar programlama süreçlerinin kendi içinde karmaşık bir yapıya sahip olması küçük yaşta öğretilmesi konusunda bazı engelleri de beraberinde getirmiştir. İlerleyen teknoloji ile kolaylaşan programlama ve yazım dilleri sayesinde bu durum günümüzde daha az zorluk barındıran bir alan dönüşmüştür. Dolayısıyla ülkelerin temel eğitim sistemlerinde yerini almaya başlamıştır.

Dünyada programlama öğretimi incelendiğinde farklı ülkelerin farklı tarihlerde bu konuya odaklandıkları görülmektedir. Estonya 2012 yılında 7 yaş ve üzeri bireylere programlama eğitimi vermeye başlamıştır. Güney Kore'de 2017'de ilkökul düzeyinde, 2018'de ise lise düzeyinde yazılım dersleri eğitim programlarına eklenmiştir. İngiltere'de alışlagelmiş bilgisayar dersi içeriklerinin (Ofis programları, grafik programları vb.) sıkıcı olduğu ve öğrenenlerin gelişmesine olanak sağlamadığını düşünülmüştür. Bu nedenle 2013 yılında ders programı esnek hale getirilerek müfredata programlama içerikleri eklenmiştir. Hedeflenene göre öğrenciler 5 yaşından itibaren programlamanın ne olduğunu ve nasıl geliştirildiğini, 11 yaşından itibaren ise programlama dillerini kullanmayı öğrenebileceklerdir. Avrupa birliği konunun bilincine vararak 2013 yılında Code-Week etkinlikleri düzenlemeye başlamıştır.

Programlama alanında ihtiyacın günden güne artışı karşısında hükümetler tarafından da bazı adımlar atılmıştır. İngiltere hükümeti Sun gazetesi ile "Çocuklar Kod Öğrenelim" kampanyası başlatmıştır. Kampanyaya web sitelerinden ve TV kanallarından destek gelmiştir. Amerika'da 2013 yılında farklı eyalet okullarında programlama dersleri verilmeye başlanmıştır. Dünyada erken yaşta programlama eğitiminin önemi giderek arttığı bu örneklerden de anlaşılmaktadır (Demirer & Sak,

2016). Amerika Birleşik Devletleri Google, Microsoft gibi büyük şirketlerin desteği ile okullarda kodlama eğitimi alanında birçok çalışma yapmaktadır. En çok bilineni “code.org” platformudur. Bu platform öğrenenlerin yazılım ile ilgili birçok kavramı bireysel olarak öğrenebildiği ve kendi programlarını yazabildikleri görülmüştür (Saygıner & Tüzün, 2017). Hindistan’da ilkokulda algoritmanın temelleri üzerine eğitim verilmektedir. Ortaokul seviyesinde Basic programlama dili, sabitler ve değişkenler üzerine dersler yer almaktadır. Avustralya 2015 yılı itibariyle birinci sınıftan başlayarak iki yıl boyunca temel programlama dilleri eğitimi verilmesini planlamıştır. Fransa’da 2015 yılında yapılan çalışmada programlama eğitiminin en erken kaç yaşında başlayabileceği araştırılmış ve okul öncesi eğitim çağında başlanabileceğine karar verilmiştir. Yapılan plan doğrultusunda amaçlananın öğrencilerin sezgi ve görsel düşünme becerilerini geliştirmek olduğu belirtilmiştir. Macaristan’da “Programcı Kızlar” adında proje ile ortaokulda öğretim gören kız öğrencilere yönelik programlama eğitimi verilmektedir. Çin’de temel programlama eğitimlerinin daha erken yaşlarda kart oyunları ile verildiği bilinmektedir. Farklı ülkeler incelendiğinde programlama eğitiminin oldukça küçük yaşlarda verildiği ve programlama eğitimine verilen önem görülmektedir.

Ülkemizde 2012 yılında “Bilgisayar” dersinin adında değişikliğe gidilmiş ve dersin adı “Bilişim Teknolojileri ve Yazılım” olmuştur. Yazılım kelimesinin dersin adına eklenmesi sonrasında müfredata eklenen algoritma ve programlama dilleri ile ilgili konuların eklenmiştir (Saygıner & Tüzün, 2017). Ülkemizde MEB ile çeşitli şirketler ve üniversiteler ile farklı projeler yürütülmektedir. EBA portal Millî Eğitim Bakanlığının bu amaçta oluşturduğu platformdur. EBA portal öğrenci ve öğretmenlerin özgün kod yazabilmelerine ayrıca yazılmış kodlara ulaşabilmelerine yardımcı olmaktadır. 2014 yılında Türkiye Bilişim Derneği ile destekçi olan üniversiteler “Bilgisayar Programlama Çocuk Oyuncağı” adıyla etkinlik organize etmişlerdir. Etkinliğin amacı ilk, orta ve lise öğrencilerinin kendi programlarını yazabilmenin kolaylığının farkına varmalarını sağlamaktır.

Ülkemiz çapında Bilgi İletişim Teknolojilerine yönelik en geniş kapsamlı proje FATİH projesidir. “Fırsatları Arttırma ve Teknolojiyi İyileştirme Hareketi” açılımı olan FATİH projesi 22.12.2010 tarihinde Millî Eğitim Bakanlığı ve Ulaştırma Bakanlığı iş birliği ile yürürlüğe girmiştir (Atun, 2018). Projede amaç fırsat eşitliğini sağlamak ve okullardaki teknolojik alt yapıyı iyileştirmektir. Bununla beraber

öğrenme-öğretme süreçlerinde öğrencilerin daha fazla duyusuna hitap edecek şekilde kullanımı sağlanmaktadır. Tüm okullara etkileşimli tahta, tablet ve internet ağ yapısı sağlanması planlanmıştır.

21. yüzyıla gelindiğinde programlama becerilerinin erken yaştan itibaren bireylere kazandırılması gerektiği vurgulanmaktadır. Bireylerin programlama becerilerine sahip olması kendilerini farklı alanlarda geliştirebilme imkânı bulmalarını sağlayacaktır. Problemlere çözüm üretmek, farklı yollar denemek, hataları ayıklamak gibi becerilere sahip olacaklardır (Yecan, Özçınar, & Tanyeri, 2017).

Programlama, karşılaşılan problemlerin çözüm adımlarının çeşitli diller kullanılarak bilgisayara aktarılmasıdır. Problemlerin çözümleri adım adım sıralanmalıdır. Adım adım sıralanan çözümlere algoritma adı verilmektedir. Günlük hayatta karşılaşılan problemlerde algoritma ile çözülmektedir. Konuşma dilinde yazılan bu adımların bilgisayarın anlayacağı şekilde yazılması programlama dilleri ile mümkündür. Algoritması verilen problem çözümünün evrensel olarak ortaya koyulması akış diyagramları ile sağlanabilmektedir. Tüm bu süreçlerin öğrenimini basit ve eğlenceli hale getirmek için bazı programlama araçları bulunmaktadır. Scratch, Mblock, Code.org, Alice ve App Inventor bunlara örnektir. Bu araçlar ile programlamanın zor ve detayı fazla olan söz dizimi kurallarından uzak, günlük konuşma dilinde hazır kod bloklarını ekleyerek programlama imkânı sunmaktadır (Demirer & Sak, 2016).

Programlama öğrenme sürecinde kazanılması gereken üç tür programlama bilgisi vardır (1988). Yazımsal, kavramsal, problem çözme bu üç bilgiyi oluşturmaktadır. Araştırmada öğrencilerin programlama eğitimi öncesi yazımsal ve kavramsal bilgileri ölçülürken, araştırmanın sonunda problem çözme bilgileri ölçülmüştür.

2.1. Pedagojik Açıdan Programlama Eğitimi

Bilişsel öğrenme alanının sınıflanması ilk olarak Benjamin Bloom ve çalışma arkadaşları tarafından gerçekleştirilmiştir. Çalışmaları 1956 yılında “Taxonomy of Educational Objectives: Handbook I: Cognitive Domain” adı altında bir kitap olarak yayımlanmıştır. 20’den fazla dile çevrilmiş kitap üzerinden 65 yıl geçmesine rağmen önemini korumaktadır (Yüksel, 2007).

Bloom Taksonomisi;

- Bilgi
- Kavrama
- Uygulama
- Analiz
- Sentez
- Değerlendirme

Basamaklarından oluşmaktadır. Bilişsel alan taksonomisi eğitim camiasında genel olarak kabul görmekle birlikte, bu sınıflamayı eksik gören araştırmacılar üzerinde çalışmalar yaparak farklı sınıflamalar ortaya koymuşlardır. Bu taksonomi programlama öğretimi ve öğrenimi açısından önem arz etmektedir. Programlama ders içeriklerinin Bloom taksonomisine göre örnek kazanımları aşağıda verilmiştir.

Tablo 2. 2. Programlama ders içeriklerinin Bloom taksonomisine göre örnek kazanımları

Basamak	Kazanım
Bilgi	Programlama dilinin söz dizimi ve yapılarını ezberler.
Kavrama	Adım adım programlamanın nasıl çalıştığını belirtir.
Uygulama	Verilen eksik kısımları tamamlar.
Analiz	Hata ayıklama işlemlerini yürütür.
Sentez	Probleme çözüm olan algoritmanın kodlamasını yapar.
Değerlendirme	Başkaları tarafından yazılan programları gözden geçirir ve yorumlar yapar, daha sonra öğretmen bu yorumları puanlandırır.

Bloom taksonomisinin günümüz toplumundaki değişimler, ihtiyaçlar, bilgi ve bilgiye ulaşımın değişmesi nedeniyle güncellenmesine ihtiyaç duyulmuştur. Taksonomi Anderson ve Kratwohl tarafından 2001 yılında güncellenerek son halini almıştır. Güncel halinde bilgi boyutu ve bilişsel süreç boyutu basamakları olarak iki ana başlık altında ayrıştırılmıştır. Bilgi boyutu alt basamakları; olgusal bilgi, kavramsal bilgi, işlemsel bilgi ve üst bilişsel bilgi olarak dört basamaktan oluşmaktadır. Olgusal bilgi öğrencinin bir konu alanında bilmesi gereken temel öğeler olarak tanımlanmaktadır. Kavramsal bilgi geniş bir yapının temel öğelerini oluşturan kavramların birlikte hareket etmesini sağlayan ilişkilerdir. İşlemsel bilgi bir görevin

nasıl yapılacağı hakkında teknik, yöntem vb. tüm birimleri tanımlayan kavramdır. Üst bilişsel bilgi kişinin kendi öğrenme stillerini bilerek bilgi sahibi olması durumudur (Çelik, Kul, & Çalık Uzun, 2017).

Bilişsel süreç basamakları “Öğretim nasıl sağlanır?” sorusuna cevap veren taksonomi basamaklarıdır. Alt basamaklar olarak hatırlama, anlama, uygulama, çözümlenme, değerlendirme ve yaratma olarak belirlenmiştir. Hatırlatma kısa süreli bellekten uzun süreli belleğe atılan bilgilere erişmek olarak tanımlanmıştır. Anlama verilen bilgilerden anlam çıkarma olarak tanımlanmıştır. Uygulama basamağında öğrenilen bilgilerin kullanılması olarak tanımlanmıştır. Çözümlenme basamağı bir materyalin parçalara ayrılması ve bütünü ile ilişkilerin belirlenmesidir. Değerlendirme ise ölçütlere dayanarak sonuçlara ulaşmak olarak tanımlanmıştır. Yaratma basamağı kişilere özgün ürün üretme olarak tanımlanmıştır (Anderson & Krathwohl, 2001).

Duyuşsal alan basamaklarını Krathwohl tarafından oluşturulmuştur. Basamaklar aşağıdaki gibidir.

- Alma
- Tepkide bulunma
- Değer verme
- Örgütlenme
- Kişilik haline getirme

Programlama ders içeriklerinin duyuşsal alan basamaklarına göre örnek kazanımları aşağıdaki gibidir.

Tablo 2. 3.Programlama ders içeriklerinin duyuşsal alan örnek kazanımları

Basamak	Kazanım
Alma	Metin tabanlı programlamayla ilgili bir uyarı (kavram veya etkinlik) algılayabilme
Tepkide Bulunma	Düzenli olarak metin tabanlı programlamayla ilgili bir uyarana yanıt verilir.
Değer Verme	Metin tabanlı programlamayla yönelik bir inanç veya tutum özelliğini gösterecek kadar sebatkâr bir şekilde tutarlı bir davranış gösterme.
Örgütlenme	Metin tabanlı programlamayla ilgili etkinlikler düzenleme.
Kişilik Haline Getirme	Bireyler, tutarlı bir şekilde nesne tabanlı programlamayla ilgili bir değer ve ilkeler kümesine göre hareket eder.

2.1.1. Pedagojik Yaklaşımlar

Programlama eğitiminde ilgili hedef ve kazanımlara, hedef kitleye, içeriğe, programlama ortamına ve bazı diğer unsurlara göre farklı pedagojik yaklaşımlardan söz edilebilir. Bu yaklaşımlar programlama öğretiminin nasıl gerçekleştirilmesi gerektiğine dair ipuçları sunmaktadır. Bu bağlamda programlama eğitiminde kullanılan pedagojik yaklaşımlar devam eden bölümde açıklanacaktır.

Programlama öğretiminin kuramsal temelleri içerisinde en fazla öneme sahip olan kuram yapılandırmacılıktır. Yapılandırmacı öğrenme kuramı öğrenme durumunu öğrenen açısından inceler. Merkeze öğrenciyi alır, öğretmen rehberlik eden konumundadır. Öğrenme öğrenenlerin gerçek yaşantıları sonucu ortaya çıkan değişim olarak tanımlanır. Bilgiyi üreten konumunda öğrenci bulunmaktadır.

Programlama öğretimindeki yaklaşımlar araç odaklı ve öğrenen odaklı olmak üzere iki başlık altında incelenebilir. Araç odaklı yaklaşımlar;

- Bilgisayarsız programlama
- Blok temelli programlama
- Metin temelli programlama
- Hybrid programlama
- Fiziksel araçlarla programlama

Ve öğrenen odaklı yaklaşımlar;

- Eşli programlama
- Oyun temelli programlama
- Proje temelli programlama
- Sorgulama temelli programlama
- Disiplinler arası yaklaşım, şeklinde listelenmiştir.

Eşli Programlama

Öğrenen odaklı yaklaşımlardan eşli programlama (pair programming), iki öğrenenin bir bilgisayar ile çalışması olarak tanımlanmaktadır (Beck & Andres, 2004). Literatür incelendiğinde eşli programlamanın temelleri 1970 yıllarına dayanmaktadır. Eşli programlama; bilgisayar bilimlerinde çevik programlama (agile programming) ve aykırı programlama (extreme programming) yaklaşımlarının uygulamalarındandır. Eşli programlama grup ile çalışma prensibine sahiptir. Gruplarda iki ve daha fazla kişi yer almalıdır. Eşli programlamada sayı en az iki ile sınırlandırılmıştır. Grupların amacı öğrenenlerin birbirlerini etkileyecek şekilde etkileşimlerde bulunmalarını sağlamaktır. Eşli programlamada birçok farklı gerekçe ile grup yapılarının bozulabildiği gözlenmektedir. Gruplar iki kişi ile sınırlandırıldığında olumsuz herhangi bir nedenle öğrenenlerden birinin gruptan ayrılması eşli programlamayı etkisiz kılacaktır (Demir, 2019). İki öğrenen eşli programlamada farklı roller üstlenebilmektedirler. Zaman zaman bir öğrenen kodu yazan, zaman zamanda fikir veren ve hata bulan rollerini yerine getirebilmektedir

Oyunlaştırma

Oyun temelli programlama, diğer deyişle oyunlaştırma bireylerin eğlenerek ve motivasyonlarını arttırarak problem çözümüne ulaşmalarını sağlamaktadır. 2010 yılında ortaya çıkan oyunlaştırma kavramı, içerikleri ilgi çekici hale getirerek öğrenme ortamı unsurlarını oyun unsurları ile harmanlamaktadır (Fiş Erümit & Kalelioğlu, 2019). Bu sayede öğrenme ortamının eğitici yönü ile oyun ortamının motive edici unsurları bir arada kullanılmaktadır. Küçük yaş gruplarına daha çok hitap etmektedir. Çocuklar oyun oynarken problem çözme, keşfetme, akıl yürütme, iletişim kurma, öz düzenleme gibi becerilerini geliştirmektedir. Oyun temelli programlama öğretimi amacıyla kullanılan bazı platformlar aşağıda verilmiştir.

- Code.org
- CodeCombat
- Tynker
- Lightbot
- TechRocket
- All Can Code
- CodeAcademy

Proje Temelli Öğrenme

Proje temelli öğrenme; öğrenenlerin günlük yaşamlarındaki problemlere bilimsel ve mantıklı çözümler üretmelerini amaçlamaktadır. Proje tabanlı öğrenmenin uygulama basamakları aşağıdaki gibidir.

1. Problem konusu belirlemek
2. Çalışma için zaman planlaması yapılması
3. Literatür taraması
4. Gözlem ve veri toplama
5. Veri analizi
6. Proje yazımı, değerlendirilmesi ve sunumu (Gündüz, 2007)

Proje tabanlı öğrenme etkinlikleri sonucunda öğrenen somut ürün ortaya koymalıdır. İşlem basamakları incelendiğinde süreçte aktif ve başlıca rol alan öğrenendir. Öğretmen rehber konumundadır. Genel özelliklerine bakıldığında yapılandırmacı yaklaşımı benimsemiş öğrenen odaklı yaklaşım olduğu görülmektedir. Ortaokul düzeyinde Bilişim Teknolojileri ve Yazılım dersi kapsamında problem çözümlerine yönelik proje görevleri verilerek proje temelli öğretim yaklaşımı kullanılabilir (Saracaoğlu & Çelik, 2018).

FeTeMM

Proje temelli yaklaşımın yanı sıra problem çözümü üzerinde odaklanan öğrenen odaklı bir diğer yaklaşım ise FeTeMM (STEM) yaklaşımıdır. FeTeMM; Fen, Teknoloji, Mühendislik, Matematik harflerin temsil ettiği alanlardır. FeTeMM uygulamaları gün geçtikçe önem kazanmaktadır. Ülkemiz 2023 vizyonu kapsamında FeTeMM uygulamalarının öğretim programlarında yer alma konusu gündeme gelmiştir (Yükseltürk, 2020). Millî Eğitim Bakanlığı 2016 yılında FeTeMM Eğitimi Raporu yayımlamıştır. FeTeMM uygulamalarının genel amacı öğrenenin merkezde

olduđu uygulamalar ile farklı disiplinler arası bağlantı kurmasını, faydalanmasını ve problemlere çözüm bulmasını sağlamaktır.

Bilgisayarsız Programlama

Araştırmanın konusu blok temelli programlamadan metin tabanlı programlamaya geçiş aşamasında yaşanan problemlere odaklanmaktadır. Blok temelli ve metin temelli programlamayı içerisinde barındıran araç odaklı yaklaşımlardan bilgisayarsız bilgisayar bilimi (B³-CS Unplugged) bilgisayar olmadan yapılan kodlama etkinlikleri olarak bilinmektedir. Temel mantık küçük yaş grubu öğrenenlere hikâye, senaryo üzerinden problem çözümü yaptırılmasıdır. Etkinliklere ulaşılabilecek örnek siteler aşağıda verilmiştir.

- Tospaa.org
- Csunplugged.org
- Code.org
- Kesfetprojesi.org

Bilgisayarsız kodlama programlamanın temelleri açısından küçük yaş grupları için oldukça önemlidir. İlerleyen programlama etkinlikleri olan blok tabanlı ve metin tabanlı programlamada küçük yaş grupları için söz dizimi kuralları, programın ara yüzü gibi öğrenilmesi gereken ek durumlar ortaya çıkmaktadır. Buna karşılık bilgisayarsız kodlama etkinlikleri ile tüm yaş gruplarına programlamanın sözdizimi karmaşasından uzak problem çözme becerileri kazandırılmaktadır. Bu tarz etkinlikler ile öğrenenlerin odak noktaları problem çözme, düşünme, karar verme, mantık ilişkisi kurma alanlarına ağırlık verecektir. Bir programlama dilinin temel mantık kurallarına ve sözdizimlerine uymaya odak ayırmalarına gerek kalmayacaktır. Bilgisayarsız kodlama etkinlikleri öğrenenleri blok temelli ve metin temelli programlamaya hazır hale getirmektedir.

Blok Temelli Programlama

Blok temelli programlama, hazır bulunan blok kodların sıralanması, birleştirilmesi ile programlama işlemidir. Blok temelli programlama ortaya çıkışı itibari ile MIT, Google gibi şirketlerin desteği ile hızla kullanımı artmış ve kendisine programlama alanında yer edinmiştir. Blok temelli programlamanın bu kadar hızlı artarak programlama araçları arasında kendisine yer edinmesindeki en büyük etken; her yaşa hitap edebilen ve günlük konuşma dili ile anlaşılabilen bir kodlama imkânı sunmasıdır.

Bu iki özelliği nedeniyle günümüzde okul öncesi ve ilköğretim seviyelerinde programlamaya başlangıçta sıklıkla kullanılmaktadır. Blok temelli programlamada kullanılan örnek araçlar aşağıda verilmiştir.

- Scratch
- Code-org
- Kodu Game Lab
- Mblock
- App Inventor
- Bubble
- Alice

Gomes ve Mendes programlama eğitiminde yaşanan zorlukları açıklamışlardır ve bu zorluklar Tablo 2.4'te listelenmiştir (Yükseltürk & Curaoğlu, 2019).

Tablo 2. 4. Programlama öğretiminde yaşanan zorlukların nedenleri

Başlıklar	Maddeler
Öğrenme Metotları	<ul style="list-style-type: none">• Öğretim kişiselleştirilmemektedir• Öğrenme stillerine uymayan öğretim stratejiler kullanılabilir• Statik materyallerle dinamik kavramlar açıklanmaya çalışılmaktadır.• Problem çözmek yerine bir programlama dilini ve sözdizimini öğretmeye odaklanılmaktadır.
Çalışma Metotları	<ul style="list-style-type: none">• Öğrenenler yanlış çalışma metotları seçmektedir.• Öğrenenler sıkı çalışmamaktadır.
Öğrencilerin Beceri ve Davranışları	<ul style="list-style-type: none">• Öğrenenler problemi nasıl çözeceklerini bilmemektedir.• Matematik ve mantıksal bilgi açısından zayıftırlar• Programlamaya özgü bilgiler yetersizdir
Programlamanın Doğası	<ul style="list-style-type: none">• Programlama soyuttur.• Karmaşık söz dizimlerine sahiptir.
Psikolojik Etkiler	<ul style="list-style-type: none">• Öğrenen motivasyonu düşük olabilir

Bu tabloya göre programlama öğreniminde zorlukları beş başlık altında açıklanmıştır. Tablo incelendiğinde öğretimin kişiselleştirilememesi, öğrenme stillerine uymayan stratejiler kullanılması, statik materyaller ile dinamik konular anlatılmaya çalışılması, genel hatlarıyla problemi çözmek yerine teorik olarak programlama dilini öğrenme üzerinde odaklanması öğrenme metotları açısından karşılaşılan zorluklardır. Öğrencilerin çalışma konusundaki eksiklikleri, problemi nasıl çözeceklerini bilmemeleri, matematik ve mantıksal açıdan zayıf olmaları gibi öğrenci odaklı sorunları çalışma metotları ve öğrencilerin becerileri başlığı altında incelenmiştir. Bu üç ana başlık ve psikolojik etkiler öğretmen ve öğrenci açısından programlama öğreniminin zorluklarına sebep olan alanları incelemiştir. Bu başlıklar haricinde kalan zorluk yaratan durumlar programlama dillerinin kendilerine has söz dizimleri ve soyut oluşu gösterilmiştir.

Metin Temelli Programlama

Metin temelli programlama gerçek yazılım olarak tanımlanabilir. Farklı programlama dillerinin farklı sözdizimi kuralları ve farklı kod kombinasyonları bulunmaktadır. Bu nedenle metin temelli programlama öğretimi diğerlerine göre daha zor olarak görülebilir. Blok temelli programlamadan metin temelli programlamaya geçişte yaşanan zorlukların sebeplerinden biri de budur. Öğrenenlerin söz dizimi kuralları, noktalama işaretleri gibi ince detaylara dikkat etmesine gerek olmadığından blok temelli programlama daha kolay öğrenilmektedir. Metin temelli programlama dillerine örnekler aşağıda verilmiştir.

- PHP
- Java
- C
- C++
- C#
- Python
- Kotlin

Araştırmada kullanılan metin temelli programlama dili Python olarak belirlenmiştir. Python dilinin seçilmesindeki etkenler; sözdizimi kurallarının diğerlerine göre daha az oluşu, lokal çalışmada Türkçe karakter problemi çıkarmaması ve açık kaynak oluşudur. Python tasarımcıları kolay okunabilir olması açısından küme

parantezleri yerine girintiler kullanarak sözdizimi konusunda oldukça basit olmasını sağlamışlardır. Bu sayede hızlı ve doğru kod yazımı mümkün kılınmıştır. Ayrıca ek derleyici program gerektirmemektedir. Kendi derleyicisi olan Python IDLE ile tüm uygulamalar yapılabilmektedir. Python dili programlama tarihine 1994-2000 yılları arasında girmiştir. Hemen hemen tüm platformlarda çalışmaktadır (Linux, Mac, Windows, Symbian). Guido Van Rossum (Amsterdam) 1990 yılında Python dilini geliştirmeye başlamıştır. Metin temelli bu programlama dili adını yılandan değil geliştirici Guido'nun sevdiği İngiliz komedi grubu Monty Python'dan almıştır. Python dili geliştirme görevini Python Yazılım Vakfı etrafında toplanan gönüllüler üstlenmektedir. Python 1.0 sürümü 1994 Ocak, 2.0 sürümü 2000 Ekim'de yayınlanmıştır (Lutz, 2001). 2008 Aralık itibariyle 3 serisi yayınlanmaya başlamıştır. Python dili kullanan platform ve şirketler incelendiğinde başlıca dikkat çeken aşağıdaki örneklere rastlanmaktadır.

- Google
- NASA
- CERN
- Django
- Zope
- Youtube

Python dilinin kullanılacağı alanlar farklılık gösterebilmektedir. Makine öğrenmesi, bilimsel durumlar, web uygulamaları ve veri analizleri gibi alanlar mevcuttur. Makine öğrenmesi (Machine Learning) sınıflama, resim tanıma, görüntü işleme gibi alanlarda verileri kullanarak algoritmalarla yeni bilgi edinme çabasıdır. Özellikle yapay zekâ alanında en çok adını duyuran programlama dilidir. Yapay zekâ alanında öncü konumda olan Google TensorFlow kütüphanesi Python diline aittir. Kütüphane derleme konusunda oldukça performanslı durumdadır. Görselleştirme alanında katkıları bulunmaktadır. GPU ve CPU ile oldukça uyumlu çalışması nedeniyle Google'ın favori kütüphanesi durumundadır. Makine öğrenmesi yapay zekanın bir parçasıdır ve bu iki kavramı en iyi şekilde bir araya getiren programlama dili Python'dur.

Python dili ile ilgili bahsedilen olumlu yönlerin haricinde tüm programlama dillerinde olabilen bazı olumsuz yanları da mevcuttur. Programlama dillerinin düzeyleri vardır. Python dili düzey olarak yüksek kategoride yer almaktadır ve bu

nedenle sistem düzeyinde kalan uygulamalar için uygun değildir. İşletim sistemleri için tasarlanan uygulamalarda verimli olma konusunda diğer programlama dillerinden geride kalmaktadır. Python en büyük artısı hızlı oluşudur. Ancak ağır programlar için yer yer yetersiz kalabilmektedir.

Hybrid Programlama

Araç odaklı yaklaşımlardan bir diğeri hybrid programlamadır. Hybrid programlama aslında mobil uygulama geliştirme için kullanılan yaklaşımdır. Hybrid programlama ortamları HTML5, JavaScript gibi web dilleri ile geliştirilen ve birçok platformda kullanılabilen uygulamalar geliştirme işlemine olanak sağlamaktadır. Hybrid uygulamalar bu özelliği ile “Çapraz (Cross) Platform” uygulamaları olarak adlandırılır. Hybrid uygulama geliştirme için bazı kütüphaneler kullanılmaktadır. Bu kütüphanelere Xamarin, Mobil Angular UI, Onsen örnektir. Hybrid ve Native arasında yazılımcı ve kullanıcı açısından farklı faydalar bulunmaktadır. Native programlama performans, kullanıcı deneyimi, güvenlik ve işlevsellik alanında öne çıkarken, Hybrid programlama düşük maliyet ve kolay öğrenilebilirlik ile ön plana çıkmaktadır.

Fiziksel Programlama

Fiziksel programlama diğer adıyla gömülü yazılım ve donanım birlikte kullanılarak dış dünya ile iletişime geçen fiziksel sistemler tasarlama işidir. Diğer bir adıyla robotik kodlama etkinlikleri de denilebilir. Örnek olarak Arduino, EV3, Lego WeDo, MBot verilebilir. Gömülü olarak çeşitli mikroişlemciler barındıran bu sistemler sensor ve motor gibi eklentiler ile daha modüler hale getirilebilir ve projeler çeşitlendirilebilir.

2.2. Programlama Eğitimindeki Sorunlar

Korhonen programlama eğitiminde algoritmanın görselleştirilmesinin ve öğrencilerinin aktif katılımını vurgularken, Crews, Murphy ve Ziegler’in yaptığı araştırmada ise programlama eğitiminde konuların genel hatlarını vermek yerine teorik ve pratik etkinliklerin iç içe olmasının daha başarılı olacağını vurgulamıştır (Crews & Murphy, 2004).

Kodlama eğitimi ülkemizde ortaokul, ilkokul ve okul öncesi kademelerinde kendine yer bulmaya başlamıştır. Kodlama kazanımları ortaokul kademesinde 5 ve 6.

sınıfta zorunlu olarak okutulan Bilişim Teknolojileri ve Yazılım dersi öğrenme alanlarında karşımıza çıkmaktadır.

Kazanım olarak; BT.5.5. Problem Çözme ve Programlama (5. Sınıf 5. Ünite), BT.5.5.2 Programlama (5. Sınıf 5. Ünite 2. Konu) konu başlıkları bulunmaktadır. 5. Sınıfta blok tabanlı kodlama etkinlikleri verilen konu başlıkları altında kazanım olarak verilmektedir. Kazanımlar şöyledir;

- BT.5.5.2.1. Programlamayla ilgili temel kavramları açıklar.
- BT.5.5.2.2. Blok tabanlı programlama aracının ara yüzünü ve özelliklerini tanımlar.
- BT.5.5.2.3. Blok tabanlı programlama ortamında sunulan hedeflere ulaşmak için doğru algoritmayı oluşturur.
- BT.5.5.2.4. Doğrusal mantık yapısını açıklar.
- BT.5.5.2.5. Doğrusal mantık yapısını kullanan algoritmalar geliştirir.
- BT.5.5.2.6. Karar yapısını ve işlevlerini açıklar.
- BT.5.5.2.7. Karar yapıları içeren algoritmalar geliştirir.
- BT.5.5.2.8. Döngü yapısını ve işlevlerini açıklar.
- BT.5.5.2.9. Döngü yapısı içeren algoritmalar oluşturur.
- BT.5.5.2.10. Farklı yapılar için oluşturduğu algoritmaların sonucunu yorumlayarak hatalarını ayıklar.

6. sınıflar için belirlenen kazanımlardan BT.6.5. Problem Çözme ve Programlama (6. Sınıf 5. Ünite), BT.6.5.2. Programlama (6. Sınıf 5. Ünite 2. Konu) konu başlıklarını bulunmaktadır. Blok tabanlı kodlama ile ilgili kazanımlar şöyledir (Millî Eğitim Bakanlığı, 2018);

- BT.6.5.2.1. Blok tabanlı programlama aracının ara yüzünü ve özelliklerini tanımlar.
- BT.6.5.2.2. Blok tabanlı programlama aracında sunulan bir programın işlevlerini açıklar.
- BT.6.5.2.3. Blok tabanlı programlama aracında sunulan bir programın hatalarını ayıklar.
- BT.6.5.2.4. Blok tabanlı programlama aracında sunulan bir programı verilen ölçütlere göre geliştirerek ve düzenler.
- BT.6.5.2.5. Doğrusal mantık yapısını içeren programlar oluşturur.

- BT.6.5.2.6. Doğrusal mantık yapısını içeren programları test ederek hatalarını ayıklar.
- BT.6.5.2.7. Karar yapısını içeren programlar oluşturur.
- BT.6.5.2.8. Karar yapısını içeren programları test ederek hatalarını ayıklar.
- BT.6.5.2.9. Çoklu karar yapıları içeren programlar oluşturur.
- BT.6.5.2.10. Çoklu karar yapısını içeren programları test ederek hatalarını ayıklar.
- BT.6.5.2.11. Döngü yapısını içeren programlar oluşturur.
- BT.6.5.2.12. Döngü yapısını içeren programları test ederek hataları ayıklar.
- BT.6.5.2.13. Bir algoritmayı uyarlamak için en uygun karar yapılarını seçer.
- BT.6.5.2.14. Farklı programlama yapılarını kullanarak karmaşık problemlere çözüm üretir.
- BT.6.5.2.15. Tüm programlama yapılarını içeren özgün bir proje oluşturur (Millî Eğitim Bakanlığı, 2018).

İlkokul kademesinde ise MEB tarafından sınıf öğretmenlerine verilen kodlama eğitimleri sayesinde bu kademedeki kodlama eğitimi çalışmalarına rastlamaktayız. 2023 vizyon belgesinde 77. Sayfasında Hedef 2 Dijital Becerilerin Gelişmesi İçin İçerik Geliştirilecek ve Öğretmen Eğitimi Yapılacak Başlığı altında;

1. İlkokul derslerinin kazanımı haline getirilmiş olan güvenli internet, siber güvenlik, siber zorbalık ve veri güvenliği gibi kavramların izleme ve değerlendirme çalışmalarıyla erişim ve edinimleri takip edilecek, gerekli iyileştirmeler yapılacaktır.
2. Sınıf öğretmenlerinin bilgisayarsız ortamda algoritmik düşünce öğretimine yönelik, yüz yüze hizmet içi eğitimler düzenlenecektir
3. Öğrencilerimizle birlikte, kendilerine bilişimle üretim becerileri kazandırmaya yönelik olarak kodlama ve 3D tasarım etkinlikleri yürütülecektir.
4. Öğretmenlerimizin dijital eğitim konusunda kendilerini geliştirmelerine yönelik olarak, istedikleri zaman faydalanabilecekleri içerik videoları geliştirecek ve çevrimiçi atölyeler düzenlenecektir.
5. Matematik, Fen bilimleri, Fizik, Kimya, Biyoloji, Türkçe, Sosyal bilgiler, Coğrafya gibi derslerin öğretmenlerine, disiplinler arası proje yapımı, 3D tasarım ve akıllı cihaz gibi alanlarda yüz yüze atölye eğitimleri verilecektir.

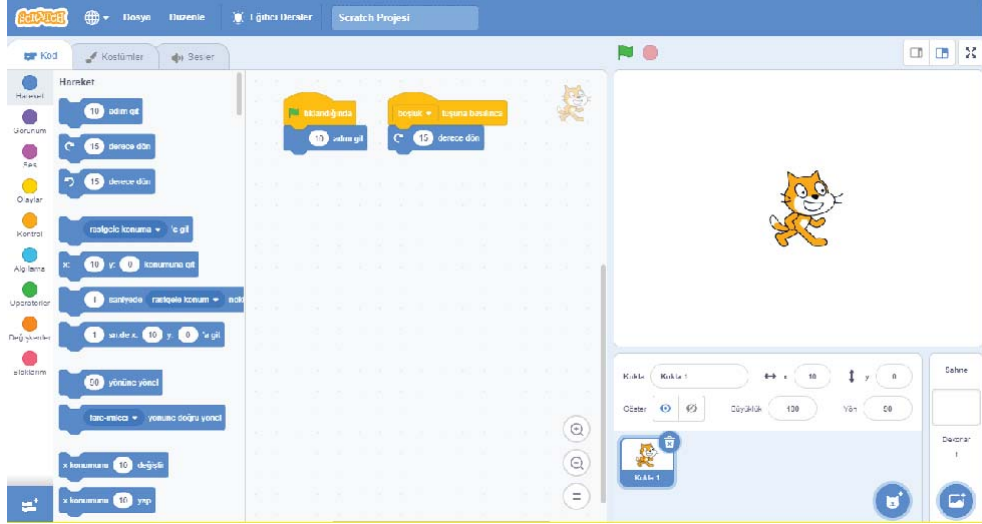
5 hedef verilmiştir (MEB, 2018). İlkokul ve okul öncesi kademelerde bilgisayarsız kodlama etkinlikleri ile öğrencilere kodlama, algoritmik düşünme becerilerinin temelleri verilmektedir.

Ortaokul kademesine ulaşan öğrenciler 5.sınıf ikinci dönem son aylarında blok tabanlı kodlama çalışmalarına başlamaktadır. 6.sınıf ikinci dönem ortalarında blok tabanlı kodlama temelleri atılmaktadır. 7.ve 8. Sınıflarda ise öğrencilere blok tabanlı kodlamanın ardından metin tabanlı kodlama etkinliklerine geçilmektedir. Kademelere göre blok tabanlı kodlama eğitim içerikleri olarak genel kabul görmüş Scratch, Bloklyeba, Kodlabüyü vb. uygulamalar içermektedir.

Kodlama mantığını öğrencilere öğretmek için yukarıda belirtilen gibi birçok uygulama bulunmaktadır. Uygulamaların içeriklerine bakıldığında küçük yaş gruplarını kendine çeken çizgi karakterler, oyunlar, animasyonlar barındırmaktadırlar. Uygulamalar öğrencilere kodlama kısmının verilen problemlere çözüm üreteceğini ve bu çözümleri kendileri oluşturmaları gerektiğini sunmaktadır. Öğrenciler çizgi karakterlerin problemlerine ya da yapmak istedikleri görevleri yapmalarına yardımcı olacaklardır. Öğrencilere kodlama içeriğinden çok problem çözme becerisi ve algoritma mantığı kazandırmak asıl amaçtır. Bu tarz blok tabanlı uygulamalar küçük yaş grupları için metin tabanlı programlama dillerinin karmaşık yapısını öğrenme zorunluluğu olmadan kodlama yapabilme imkânı sağlamaktadır (Aytekin, Sönmez Çakır, Yücel, & Kulaozü, 2018).

Araştırmada blok tabanlı programlama eğitimi araçlarından Scratch kullanılacaktır. Scratch programı 2003 yılında MIT tarafından geliştirilmeye başlanmıştır. MIT açılımı, Massachusetts Institute of Technology'dir. Scratch ücretsiz görsel (blok tabanlı) programlama aracıdır. Farklı dil seçenekleri de bulunmaktadır. Çalışma mantığı sürükle-bırak şeklinde işlemektedir. Kodlama bilmeyenler için oldukça uygun bir araçtır (Mitchel Resnick ve Arkadaşları, 2009).

Bu araştırmanın problem durumunu metin tabanlı ve blok tabanlı programlama ortamları arasındaki kavramsal farklılıklar oluşturmaktadır. Blok tabanlı programlama ortamında oluşturulan bazı kodlama yaklaşımları metin tabanlı programlamaya geçişte aynı şekilde uygulanamamaktadır.



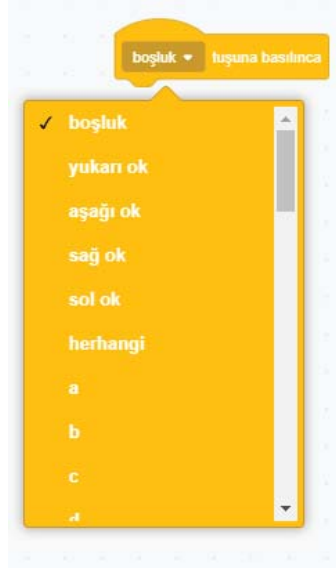
Şekil 2.1. Scratch Programı Örnek Uygulama

Şekil 1’de örneği bulunan 2 ayrı başlatıcı mantığının metin tabanlı kodlama da herhangi bir karşılığı bulunmamaktadır.

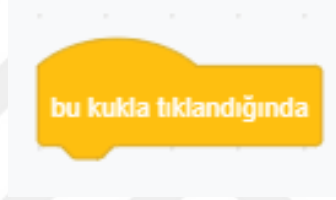
Metin tabanlı programlama dillerinde editörler yardımı ile yazılmış kod satırlarını bilgisayar en üst satırdan başlayarak okumaktadır. Ortadan ya da rastgele bir satırdan başlangıç yapmamaktadır. Ancak Scratch programında farklı başlatıcı, tetikleyiciler altında yazılmış kodların sıralaması önemsiz konumdur. Çünkü aynı başlatıcı birden fazla kullanılarak farklı görevler atanabilir. Scratch tetikleyicileri;



Şekil 2.2. Bayrağa Tıklandığında



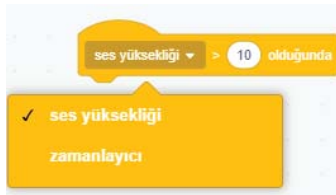
Şekil 2.3. Tuş ile Başlama



Şekil 2.4. Kukla Üzerine Tıklama ile Başlama



Şekil 2.5. Dekor Değişimi ile Başlama



Şekil 2.6. Ses ve Zamanlayıcı Sorgusu ile Başlama



Şekil 2.7. Haberleşme ile Başlatma

Bahsedilen iki programlama ortamı arasındaki farklar birinden diğerine geçişte kavram yanılgılarına ve kodlama öğretiminde bazı sorunlara neden olabileceği düşünülmektedir.

3. YÖNTEM

3.1. Araştırma Deseni

Araştırma nicel ve nitel araştırma yöntemlerinin beraber paralel olarak kullanıldığı karma araştırma desen şeklinde oluşturulmuştur. Bu yaklaşıma Cresswell (2012) yakınsak karma araştırma deseni adını vermiştir. Yakınsak karma araştırma deseninde temel amaç nicel ve nitel veri toplama süreçlerinin beraber yürütülerek araştırma sorularına cevap bulmak için birlikte kullanılmasıdır. Bu sayede nicel ve nitel araştırma süreçlerinin barındırdığı kendilerine özgü zayıflıklar mümkün olduğunca en aza indirilmektedir. Araştırmacı nitel ve nicel verileri toplar ve her bir veri setini ilgili analiz yöntemleri ile inceler. Sonuç olarak iki farklı koldan elde edilen bulgular beraberce yorumlanarak araştırma sorusu cevaplandırılmaya çalışılır.

Bu çalışmanın nicel boyutu son test kontrol gruplu yarı deneysel desen şeklinde tasarlanmıştır. Buna göre çalışmaya katılan öğrenciler iki grup halinde temel programlama eğitimi almışlardır. Gruplardan ilki temel programlama eğitimini metin temelli programlama araçları üzerinden almıştır. Bu süreçte Python programlama dili üzerinden temel programlama kavramları öğrencilere aktarılmaya çalışılmıştır. İkinci grup öğrencilere ise blok temelli programlama ortamlarından olan Scratch kullanılarak temel programlama eğitimi verilmiştir. İki grupta eğitimler tamamlandıktan sonra öğrencilerin programlama kavramlarına dair gelişimleri ölçülmüştür. Daha sonrasında temel programlama eğitiminin üzerine eklenen yeni konu ve kavramlara yönelik olarak eğitimlerine devam etmişlerdir. Bu eğitimler metin temelli ortamlarda gerçekleştirilmiş ve sonunda programlama içeriklerine yönelik olarak öğrencilerin başarıları ölçülmüştür.

Nitel boyutta da öğrencilerin hem başarı testindeki cevapları içerik analizine tabi tutulmuş hem de programlamaya dair kavram oluşumları incelenmeye çalışılmıştır. Bu sayede hem akademik başarıları hem de kavram yanılgıları ile çalışmada yer alan programlamaya giriş eğitimlerinin ilişkisi açıklanmıştır. Bu süreçte iki ölçümde elde edilen veriler hem nicel olarak analiz edilmiş hem de öğrencilerin verdiği cevaplar derinlemesine içerik analizine tabi tutulmuştur.

3.2. Ölçme Araçları

3.2.1. Kavram Değerlendirme Ölçeği

Kavram değerlendirme ölçeği geliştirilirken araştırmacı soru havuzu oluşturmuştur. Soru havuzu oluşturulurken literatürde bulunan temel programlama kavramları araştırmacıya yol göstermiştir. Sonrasında alan uzmanı desteği alınarak bazı sorular elenmiş ve ölçekten çıkarılmıştır. Alan uzmanı öğrencilerin durumuna göre soruların ağır geleceğini düşünen araştırmacının birkaç soru elemesini tavsiye etmiş ve araştırma bu yol ile bazı soruları elemiştir. Son halini alan ölçek soruları yazılı sınav şeklinde öğrencilere uygulanabilir hale getirilmiştir. Toplam 12 sorudan oluşmaktadır.

Kavram değerlendirme ölçeğinde öğrencilerin programlama öğrenimi öncesinde temel kavramlar hakkında bilgileri ölçülmüştür. Bu ölçek öğrencilerin programlama eğitimi öncesinde uygulanmıştır. Ölçeğin içerisindeki sorular öğrencilerin temel programlama kavramlarını açıklamalarına yönelik olarak hazırlanmıştır. Ayrıca açıklamanın yanında her bir kavrama dair örnekler vermeleri de istenmiştir. Bu sayede kavrama yönelik öğrencilerin mevcut durumu anlaşılmasına çalışılmıştır. Aşağıda bu ölçekte yer alan sorulara yer verilmiştir (Bkz. Ek 11):

- Döngü nedir?
- Fonksiyon nedir?
- Eğer sorgusu, eğer ise değilse sorgusu nedir nasıl çalışır?
- Ve veya sorgusu nedir nasıl çalışır?
- Değişken nedir?
- Doğrusal, doğrusal olmayan programlama nedir?
- Bilgisayarın soru sormasını hangi yapılar ile sağlarsınız?

Bu ölçeğin uygulandığı öğrenciler daha önce resmi olarak programlama eğitimi almamıştır. Teknoloji ile ilgili olarak temel bilgisayar kullanımı eğitimi almışlardır. Bu eğitim içerisinde bilgisayarın açılıp kapatılması, fare ve klavye kullanımı, klasör oluşturma, kaydetme işlemleri, görev yöneticisi kullanımı, arama yapma gibi temel konular anlatılmaktadır.

3.2.2. Programlama Becerileri Ölçeği

Ölçek hazırlanırken yıl içerisinde öğrencilere anlatılan başlıca konular seçilmiştir. Konu ve soru seçiminde en çok üzerinde durulan ve sıklıkla tekrar edinilen konular dikkate alınmış ve bu konular Python programlama dili ile çözülebilecek sorular haline dönüştürülmüştür. Bu sayede Python programlama dili aracılığıyla öğrencilerin programlama becerilerini ölçebilecek şekilde temel kavramsal sorular ve pratik kod yazma kısımları barındıran ölçek yazılı sınav olarak uygulanacak hale getirilmiştir. Alan uzmanı ile görüşmeler yapılmış ve öneriler doğrultusunda bazı sorular ölçekten çıkarılmış ve yerine başka farklı eklenmiştir. Ölçeğin son halinde alan yazında sıkça tercih edilen soruların yanında araştırmacı tarafından geliştirilen soruların bulunduğu bir ölçme aracı elde edilmiştir. Ölçeğin son versiyonu alan uzmanı tarafından tekrar incelenmiş ve iki adet soru ölçekten çıkarılmıştır. Bu sorular kod yazma yeteneklerini derinlemesine ölçmeye çalışan sorular olmasına rağmen; cevabın oldukça uzun oluşu ve cevaplama da yaşanacak zorluklar nedeniyle çalışmanın geçerliliğini etkileyeceği gerekçeleri soruların temel gerekçe olmuştur. Tüm düzenlemelerden sonra 13 sorudan oluşan programlama becerileri ölçeği elde edilmiştir (Bkz. Ek 12).

Program becerileri ölçeği çalışma sonunda iki gruba da uygulanmıştır. Ölçek ile deney grubu metin tabanlı programlama eğitimini, kontrol grubu ise müfredatta yer alan blok tabanlı programlama eğitimini tamamladıktan sonra son test uygulanmıştır. Bu ölçekte bulunan soruların amacı programlama becerileri kapsamında hata ayıklama, kod yazma ve analitik düşünme gibi becerileri ölçmektedir. Aşağıda bu ölçekte yer alan bazı sorulara örnekler verilmiştir.

- Değişken oluşturma kurallarını yazınız ve 1 örnek veriniz.
- Verilen kodların ekrana çıktısını yazınız.
 - şehir='Ankara'
 - şehir='İstanbul'
 - print ('Şehir: ', şehir)
- Print fonksiyonunu kullanarak yaş, ad, merhaba yazdırınız.

Verilen problemi Python dili ile çözecek kodları yazınız.

- Kullanıcıdan iki sayı alan ve toplamlarını ekrana yazdıran programı yazınız.

3.3. Katılımcılar

Çalışma Ağrı ili Patnos ilçesi Değirmendüzü köyünde bulunan Değirmendüzü Ortaokulunda gerçekleştirilmiştir. Okul ortalama 450 öğrenci ve 27 öğretmen ile eğitime devam etmektedir. Farklı köylerden taşınmalı olarak okula gelen öğrenciler de bulunmaktadır.

Tablo 3. 1. Katılımcı sayısı

Grup	Katılımcı sayısı
Deney	12
Kontrol	12

Çalışmaya farklı köylerden taşınmalı olarak okula gelen ve 7. Sınıf kademesini 17 Haziran 2022 tarihi ile bitirecek olan öğrenciler katılmıştır. Çalışma sürecinde çalışmadan çıkma ve yüksek düzeyde devamsızlık yapma ihtimali olan öğrenciler çalışmanın veri seti dışında bırakılmıştır. Sağlıklı veri toplanabileceği düşünülen 24 öğrenci çalışmaya dâhil edilmiştir. Gruplar eşit olacak şekilde 12'şer öğrenciden oluşmaktadır. Çalışılan gruplar 5. Sınıf kademesinde temel bilgisayar eğitimi almışlardır. Programlama eğitimi almayan bu gruplar 6. Sınıf kademesinde temel programlama eğitim almaya başlamışlar ve aynı süreçte çalışmaya dâhil olmuşlardır. Genel akademik başarı açısından iki grup eşdeğer başarıya sahiptir. Gruplar kendi içerisinde homojen olarak dağılım göstermektedir. Öğrenciler 5. Sınıf kademesinde temel bilgisayar eğitimi almış, 6.sınıf kademesinin büyük bir kısmında pandemi nedeniyle uzaktan eğitime devam edilmiştir. Köy okulu öğrencileri maddi yetersizlikten dolayı derslere katılım gösterememiş; köy okullarının açık olduğu sürelerde gruplar belirlenen eğitimleri almıştır.

3.4. Süreç

Çalışmaya katılım gösteren öğrenciler iki eğitim-öğretim yılı boyunca çalışmada yer almıştır. Öğrenciler 6 ve 7. Sınıf kademesinde eğitim-öğretime devam ederken tez çalışması yürütülmüştür. Çalışma öncesinde, pandemi sürecinin içerisinde ortaokula başlayan öğrenci grupları uzun süre yüz yüze eğitim görememişlerdir. Uzaktan eğitim için köy okulunda eğitim gören bu öğrenciler yeterli düzeyde teknolojik alt yapıya sahip değildirler. Köyde internet alt yapısı bulunmadığından dolayı evlerde internet bulunmamaktadır. Bu nedenle öğrenciler uzaktan eğitimde yeterli katılım gösterememişlerdir. Kardeş sayısının oldukça fazla olduğu aile yapısına sahip olan

Değirmendüzü köyünde öncelik yılsonunda sınava girecek öğrencilere verilmiş ve bir ailede sınav grubu öğrencilerin uzaktan eğitime katılımı sağlanmaya çalışılmıştır. Bu durumdan ötürü 5. Sınıf kademesinde uzaktan eğitim döneminde çalışmada yer alan öğrenciler hiçbir programlama eğitimi almamışlardır. Köy okullarının kısa bir süre yüz yüze eğitime başladığı süreçte öğrenciler temel bilgisayar kullanımı ve teorik içerikler verilmeye çalışılmıştır.

6. Sınıf kademesine geçiş yapılan yeni eğitim öğretim yılında çalışmanın uygulandığı köy okulu yüz yüze eğitim vermeye başlamıştır. 15 Şubat 2021 tarihinde köy okulları bu uygulamaya geçiş yapmıştır. Çalışmaya başlanmadan önce öğrencilere yazılı izin dilekçeleri verilerek aileleri tarafından doldurulması istenmiştir. Evrakları araştırmacıya teslim eden öğrenciler arasından eşit sayıda ve homojen dağılımlı olarak belirlenmiştir. Deney grubu 12 kontrol grubu 12 olmak üzere toplam 24 öğrenci çalışma için belirlenmiştir. Belirlemede kriter olarak okula devam durumu ve akademik başarı durumu temel alınmıştır. Gruplar akademik olarak homojen olacak şekilde ve iki grupta birbirine yakın seviyede öğrencilerden oluşturulmuştur. Gruplar hiçbir programlama eğitimi görmeyen öğrencilerden oluşmaktadır. Kontrol grubu müfredata paralel olarak programlama eğitimine blok temelli programlama araçları ile başlamıştır. Deney grubu blok temelli programlama araçlarını hiç görmeden doğrudan metin temelli programlama eğitimi ile programlama eğitimi görmeye başlamışlardır. Gruplar 6. Sınıf kademesinin sonuna gelene kadar programlama eğitimini almaya devam etmişlerdir. Deney ve kontrol grubu için eğitimin içeriğini temel programlama etkinlikleri oluşturmuştur. Kontrol grubu için blok temelli programlama araçlarından Scratch uygulamasını kullanılmıştır. Programın temel kullanımı ile başlayan eğitim hazır kod bloklarının belirli başlıklarda ayrıldığı menüler ve içerikleri öğretilmiştir. Bu menüler aşağıdaki gibidir.

- Hareket
- Görünüm
- Ses
- Olaylar
- Kontrol
- Algılama
- Operatörler
- Değişkenler

Deney grubu için temel seviyede programlama mantığı ve Python içerikleri konuları işlenmiştir. Söz dizimi kuralları, derleyici kurulumu, klavye ile satır düzeninde kod yazımı ve ekrana isim yazdırma seviyesinde basit kodlar konuların içerikleridir.

6. Sınıf kademesinin sonuna gelindiğinde iki gruba kavram değerlendirme testi uygulanmıştır. Kavram değerlendirme testi temel programlama kavramlarını içerisinde barındıran sorulardan oluşmaktadır (bkz. Ek 11). Kavram değerlendirme testi sonuçları benzer cevaplar tema olarak belirlenerek tablo halinde veri analizi yapılmıştır.

Deney ve kontrol grubu öğrenciler 7. Sınıf kademesine geçiş yaptıktan sonra çalışmanın ikinci aşamasına geçilmiştir. İkinci aşamada deney ve kontrol grubu öğrencilerinin metin temelli programlama içerikleri ile eğitimine başlanmıştır. Gruplar metin temelli programlama eğitimini Python programlama dili üzerinden almışlardır. Yüksek seviyeli programlama dilleri arasında bulunan ancak öğrenim ve öğretim bakımından kolay olduğundan dolayı metin tabanlı programlama eğitimi için Python dili seçilmiştir. Ayrıca açık kaynaklı oluşu ve internet üzerinden çalıştırılabilmesi de seçilmesinde etkili olmuştur (Çobanoğlu, 2018). 6. Sınıf kademesinde metin tabanlı programlama eğitimi gören deney grubu içerik olarak temel kodlar ve fonksiyonlar, kaydetme ve koşturma gibi konulara odaklanmıştır. Genel programlama müfredatına paralel bir şekilde konu başlıkları aşağıdaki gibidir (Akbıyık ve diğerleri, 2020).

1. Python Programlama Dili

- a. Bilgisayar programları nasıl çalışır?
- b. Programlama dilleri
- c. Derleyici ve yorumlayıcılar
- d. Python nedir?
- e. Neden Python?
- f. Python kurulumu
- g. Etkileşimli kabuk
- h. İlk kodlama

2. Yorum satırı

3. Değişkenler

- a. Değişken oluşturma kuralları
- b. Değişken oluşturma standartları

4. Veri Tipleri
 - a. int
 - b. str
5. Operatörler
 - a. Aritmetik
 - b. İlişkisel
 - c. Atama
 - d. Mantıksal
6. Temel Fonksiyonlar
 - a. print()
 - i. Ters taksim (\)
 - ii. Alt satır başı (\n)
 - iii. Sekme (\t)
 - iv. Çift tırnak ("")
 - b. input()

Gruplar metin temelli programlama içeriklerini verilen sıra ile tamamladıktan sonra yıl sonuna gelindiğinde öğrencilere programlama becerileri testi uygulanmıştır (bkz. Ek 12). Uygulanan iki ölçekte yazılı sınav şeklinde uygulanmıştır. Verilen cevaplar içerik analizine tabi tutulmuş ve ayrıca her bir sorunun cevabının doğru olup olmadığı cevap anahtarı aracılığıyla tespit edilerek frekans verilerine dönüştürülmüştür.

3.5. Veri Analizi

Öğrencilerin kavram değerlendirme ölçeğine verdiği cevaplar içerik analizine tabi tutulmuştur. Her bir öğrencinin ilgili kavrama yönelik tanımlar incelenerek tanım içerisindeki ifadeler kategorize edilmiştir. Bu kategorilerin tekrar etme sıklığı hesaplanmış ve deney ile kontrol grubundaki öğrencilerin cevapları karşılaştırılmıştır. Çalışma sonunda oluşturulan programlama başarı ölçeği dersin içeriği ile ilgili olarak öğretmenin çalışma sonuna kadar işlediği konuları kapsayan soruları içermektedir. Bu sorulara verilen cevaplar doğru olma durumlarına göre incelenmiş ve soruya cevap veren öğrencilerin cevapları aynı zamanda kategorize edilerek karşılaştırılmıştır. Bu sayede nicel ve nitel veriler beraberce incelenerek birbirini destekleyecek şekilde

kullanılmıştır. Elde edilen nicel veriler parametrik analizleri gerçekleştirilmesine olanak sağlamadığı için frekans değerleri üzerinden çalışma tamamlanmıştır.

3.6. Araştırmacının Rolü

Bilişim Teknolojileri ve Yazılım öğretmeni olarak görev alan araştırmacı okulda deney ve kontrol gruplarının Bilişim Teknolojileri ve Yazılım derslerine girmektedir. Grupların 6 ve 7. Sınıf derslerine girmekte olan araştırmacı 5. Sınıf kademesinde öğrencilerin dersine girmemiştir. Araştırmada verilen blok tabanlı ve metin tabanlı programlama eğitimini araştırmacı vermektedir. Ölçekleri uygulayan, veri analizi yapan araştırmacının kendisidir. Aynı araştırmacının iki gruba da eğitimi vermesi, ölçekleri uygulaması öğretmenler arası farklılığın ortaya çıkmasını engellemiştir.



4. BULGULAR

Bu bölümde araştırma sonunda elde edilen bulgular iki başlık halinde ele alınmıştır. Birinci başlıkta araştırmanın başlangıcında uygulanan kavram değerlendirme ölçeğinin sonuçlarına yer verilmiştir. Kavram değerlendirme ölçeği ile ilgili sunulan bulgular içerisinde temel programlama kavramlarının tanımları yer almıştır. İkinci başlıkta ise programlama becerileri değerlendirme ölçeğinin sonuçlarına yer verilmiştir.

4.1. Kavram Değerlendirme

Kavram değerlendirme başlığı altında çalışmaya katılan ve temel programlama eğitimini metin ve blok tabanlı ortamlarda alan öğrencilerin eğitim sonunda programlama kavramlarına yönelik gerçekleştirdikleri kavramsal tanımlara yer verilmiştir. Bu bağlamda ölçekte bulunan sorulara verilen yanıtlar üzerinden elde edilen bulgular bu bölümü oluşturmuştur. Araştırmanın başlangıcında deney ve kontrol gruplarına uygulanan kavram değerlendirme uygulanan ile temel kavramlar konusundaki durum anlaşılmasına çalışılmıştır. Yapılan değerlendirme sonrası deney ve kontrol gruplarının verdiği cevaplar içerik analizine tabi tutularak yorumlanmıştır. Aşağıda kavram değerlendirme ölçeğinde yer alan her bir kavrama ait bulgular ayrı ayrı olacak şekilde sunulmuştur.

4.1.1. Döngü

Programlamada en sık kullanılan yapılardan biri döngülerdir. Döngülerin amacı belirli şartlar gerçekleşinceye kadar veya gerçekleşmeye devam ettikçe bir komut parçasını çalıştırmaktır. Döngüler sonsuz defa tekrarda edebilir ya da belirlenmiş sayıda da tekrar edebilir.

Çalışmaya katılan öğrencilere temel programlama eğitimini tamamlamalarını takiben sorulan ve tanımlamaları istenen ilk kavram “Döngü” kavramı olmuştur. Tablo-5 öğrencilerin döngü kavramları içerisinde yer alan ve en çok tekrar eden üçer ifadeyi içermektedir.

Tablo 4. 1. Döngü kavramına ait ifadeler

Grup	İfade	<i>f</i>
Kontrol	Dönmek	3
	Yol almak	1
	Geri çağırma	2
Deney	Tekrar kullanma	2
	Tersine çevirme	2
	Dönmek	6

Yapılan değerlendirme sonrasında “Dönmek” cevabını kontrol grubundan 3, deney grubundan 6 kişi vermiştir. “Geri çağırma”, “Tekrar kullanma”, “Tersine çevirme” cevapları verilen diğer cevaplar arasında en fazla sayıda tekrar eden cevaplardır. Döngü kavramının açıklanması istenen bu soruda açıklamayı örnekler ile yapabilecekleri belirtilmiş ve Scratch programından örnek veren öğrenciler olmuştur. Verilen cevaplar incelendiğinde öğrencilerin kelimenin sözlük tanımına yönelik anlam çıkarmaya çalıştıkları görülmüştür. Bu nedenle döngü kelimesinden dönmek, geri çağırma çağrışimleri üzerinden tanımlarını gerçekleştirmişlerdir. Döngü kelimesinin anlam olarak net bir anlam taşıması cevabı kelime anlamından çıkarmaları konusunda yardımcı olmuştur. Programlama alanında bu kavramın doğru açıklamasına en yakın cevap “tekrar kullanım” cevabı olmuştur.

4.1.2. Fonksiyon

Daha önce programlama eğitimi almayan katılımcılara sorulan “Fonksiyon nedir?” sorusuna verilen cevaplar Tablo 6 da verilmiştir.

Tablo 4. 2. Fonksiyon kavramına ait ifadeler

Grup	İfade	<i>f</i>
Kontrol	İşlem	8
	Yazılanların tersi	1
	Bitkilerle alakalı	1
Deney	Programın çalışmasından sorumlu	1
	Eğer sorgusu	2
	İşlem	4

“Fonksiyon nedir?” sorusuna verilen yanıtlar içerisinde en fazla tekrar eden yanıt “İşlem” yanıtı olmuştur. Bu yanıt kontrol grubundan 8, deney grubundan 4 öğrenci tarafından verilmiştir. Kontrol grubunun bu cevabı daha fazla veriyor olmasının yanında kontrol grubunda “Yazılanların tersi” cevabın da verilmiştir. Bu durumun

olmasının Matematik dersinin bu cevaplar üzerinde etkili olabileceğini fikrini ortaya çıkarmaktadır. Ancak ölçeğin uygulandığı tarihe kadar öğrencilerin aldıkları Matematik dersinde fonksiyonlar konusu işlenmemiştir. Ayrıca verilen cevaplardan “işlem” cevabı Matematik dersinden, “bitkilerle alakalı” cevabı Fen Bilimleri dersinden akılda kalanlarla verilmiş cevaplar gibi görünmektedir. Kelime olarak bilgisayar ve yazılım ile alakalı bağlantı kuran katılımcı bulunmamaktadır.

4.1.3. Eğer Sorgusu

Programlamada eğer sorguları belirli şartların oluşması koşulu ile belirli kodları çalıştırmak için kullanılır. Farklı programlama dillerinde yapısı değişmemekle beraber kullanımı değişebilmektedir. Python programlama dilinde eğer sorgusu için kullanılan yapı “if-elif-else” olarak sıralanmaktadır. “if” eğer anlamı taşıırken, “elif” eğer ise, “else” değilse anlamı taşımaktadır. Eğer sorgusunun karşılığı olan “if” komutu yalnızca bir defa ve sorgunun en üst sırasında olarak kullanılmaktadır. Eğer sorgusu öğrencilere daha sonraki süreçte bilgisayarın soru sorması şeklinde benzetmeyle anlatılmıştır. Türkçe ve dersinde bulunan koşul, şart cümleleri konusundan dolayı öğrenciler bu soruya yüksek oranda doğru cevaplar vermişlerdir.

Tablo 4. 3. Eğer sorgusu kavramına ait ifadeler

Grup	İfade	<i>f</i>
Kontrol	Koşul	5
	Şart	2
	Cevap doğruysa çalış	3
Deney	Koşul	8
	Baklava dilimi	4
	Sorgu	5

Verilen cevaplar incelendiğinde en çok tekrar eden cevap “Koşul” olmuştur. Deney grubu 8, kontrol grubu 5 kişi ile “Koşul” yanıtını vermiştir. Ancak kontrol grubun vermiş olduğu “Şart” cevabı koşul ile aynı anlama geldiğinden verilen 2 “Şart” cevabını “Koşul” olarak kabul edilmiştir. Buna göre verilen cevap sayıları kontrol grubunda 7, deney grubunda 8 kişi olmuştur. Bu cevaplar haricinde en çok tekrar eden cevap “Sorgu” ve “Baklava dilimi” olmuştur. Baklava dilimi cevabı deney grubundan 4 öğrencinin akış şemaları hakkında bilgi sahibi olduğunu göstermektedir.

Eğer sorgusunu gruplar daha önceki yıllarda duyduklarını, kelime olarak anlamı ile örtüştüğünden dolayı verilen cevaplar doğruya yakındır. Cevaplar ağırlıklı olarak

“Koşul, şart” olarak verilmiştir. Akış şemasını bilen öğrencilerden ş “baklava dilimi” cevabı gelmiştir. Bilgisayarın soru sormasını sağlayan sorgu “Eğer” sorgusudur. Bu iki soru ile arada bağlantı kurabilme durumlarına bakılmıştır. Ancak verilen cevaplar başarılı değildir.

4.1.4. Eğer-ise-değilse

Python programlama dilinde “elif” komutunun karşılığı olan eğer ise değilse komutu kendisinden önceki tüm şartlar sağlanamadığında kullanılmaktadır. Oluşturulan “if-elif-else” komut sıralamasında “if” komutu en başta ve bir defa kullanılırken “elif” komutu istenilen sayıda alt alta kullanılabilir.

Önceki soruda sorulan eğer sorgusundan farklı olarak koşulun çalışmaması durumunda programın ne yapacağını belirle imkânı sunan “Eğer ise değilse” komutuna verilen cevaplar aşağıda incelenmiştir.

Tablo 4. 4. Eğer ise değilse kavramına ait ifadeler

Grup	İfade	f
Kontrol	Evet mi? Hayır mı?	1
	Kararsızlık	1
	Eğer sorgusunun tersi	7
Deney	Eğer ise yap değilse bitir	1
	Soru sormak	2
	Eğer sorgusu	11

Yapılan değerlendirmeye göre “Eğer ise değilse sorgusu nedir?” sorusuna verilen cevaplar arasında en çok tekrar eden cevap “Eğer sorgusu” olmuştur. Ancak verilen bu cevabın soruya verilmiş doğru cevap olarak kabul edilmemiştir. Soru içerisinde bulunan “Eğer” kelimesine tutunarak verilmiş bu cevap doğru sayılmamaktadır. Verilen cevaplar arasında doğru açıklamaya en yakın cevap “Eğer ise yap değilse bitir” cevabıdır. Bunun sebebi değilse kısmında farklı görevlerin yapılabileceğine dair oluşturulan ifadedir. Bu soruya verilen bu cevabın sayısı sadece bir kişidir. Eğer sorgusuna verilen cevapların aksine “Eğer ise değilse” sorgusuna verilen doğru yanıtlar oldukça azdır. Verilen cevaplar “Eğer” sorgusuna verilen yanıtlarla benzerlik göstermektedir. Buna göre “Eğer” kelime anlamı olarak çağrışımında bulunmalarına olanak sağlarken “değilse” kısmında kelime anlamı yeterli olmamıştır.

4.1.5. Ve-Veya

Ve-Veya bağlaçları Türkçe dersi ile bağlantı kurarak cevap verebilecek bir konu olarak nitelendirilebilir. Bu soruda katılımcılar doğru açıklama yapmakta zorlanırken doğru örnekler verebilmiştir.

Tablo 4. 5. Ve operatörüne ait ifadeler

Grup	İfade	<i>f</i>
Kontrol	Tıklanınca çalışır	2
	İki kelimenin arasına yazılır	2
Deney	3 adım git ve sola dön vb. Scratch örneği	4
	Sorguların yanlarına konur	2

Tablo 4. 6. Veya operatörüne ait ifadeler

Grup	İfade	<i>f</i>
Kontrol	Kararsızlık	1
Deney	Programı aralamak	1

“Ve-veya” sorguları için açıklama yapamayanların örnek ile açıklayabilecekleri belirtilmiştir. Kelime anlamı üzerinden verilen cevaplar ağırlıktadır. “Ve” için verilen doğru cevap ve örnek sayısı “veya” ile karşılaştırıldığında oldukça fazladır. “Veya” öğrenciler için açıklanabilir durumda değildir.

4.1.6. Değişken

Tablo 4.7 ve 4.8 incelendiğinde değişkeni açıklamaları istenen soruda akış şemasında değişken oluşturmak için kullanılan geometrik şekil ile verilen cevaplar bu soruda verilen doğru yanıttır. Açıklama konusunda problem yaşayan katılımcılar örnek verebilme imkânına sahip olmalarına rağmen örnek vermemişlerdir. Kural yazmaları istenilen soruda grupların içerisinde yalnızca bir cevap doğru verilmiştir. Bu öğrencinin programla konularına dair özel bir ilgi ya da deneyime sahip olduğu görülmektedir. Değişken oluşturma kurallarından yalnızca “Birden fazla oluşturulabilir” cevabı yakın kabul edilebilir nitelikte bir ifade olarak değerlendirilmiştir. Diğer cevaplar doğru yanıtlardan oldukça uzaktır.

Tablo 4. 7. Değişken kavramına ait ifadeler

Grup	İfade	<i>f</i>
Kontrol	Program	1
	Soruyu değiştirme	1
Deney	Dikdörtgen (Akış şeması)	2
	Programı değişken olarak kullanmak	1

Tablo 4. 8. Değişken oluşturma kurallarına ait ifadeler

Grup	İfade	<i>f</i>
Kontrol	Karar değiştirmek	1
	Soruyu değiştirmek	1
Deney	Birden fazla oluşturulabilir	1
	Programa sorular sormak	2

4.1.7. Başlatıcı

Başlatıcının görevlerini ve nerede kullanılacağı sorusuna ait verilen cevaplar Tablo 4.9’da incelenmiştir. Daha önce programlama eğitimi görmeyen çalışma gruplarının yapılan çalışmanın programlama ile ilgili olmasından dolayı “Kodlamayı başlatır”, “Programı başlatan sorgu” cevaplarını verebilmiştir.

Tablo 4. 9. Başlatıcı kavramına ait ifadeler

Grup	İfade	<i>f</i>
Kontrol	Başla	4
	Bilgisayarı başlatan	2
Deney	Kodlamayı çalıştırır	5
	Programı başlatan sorgu	1

Öğrencilerin bu soruya doğru yanıt verilme oranı olarak en yüksek orana sahip soru “Başlatıcı nedir?” sorusu olmuştur. Kelime anlamı ve ders anlatımında verilen örneklerden dolayı iki grupta doğru yanıtlar ve örnekler verebilmiştir. Deney ve kontrol grubunun “Başla” kelimesine odaklanarak cevaplar vermesi doğru yanıtı ulaştırmış olabilir. Deney grubunda verilen “Kodlamayı çalıştırır” cevabı en fazla sayıda verilmiştir.

4.1.8. Doğrusal, Doğrusal Olmayan Programlama

Araştırma kapsamında verilen programlama eğitimleri doğrusal programlama mantığı içerisindedir. Ancak Scratch programının doğrusal olmayan programlama mantığına uyan kısımları da bulunmaktadır. Bu nedenle bu iki kavram çalışma gruplarına sorulmuş ve cevaplarına ilişkin bulgular Tablo 4.10 ve 4.11’de özetlenmiştir.

Tablo 4. 10. Doğrusal programlama kavramına ait ifadeler

Grup	İfade	<i>f</i>
Kontrol	Doğru olan programlama	4
	Doğru yol	1
	Hata yapmadan yazılan kodlar	1
Deney	Güvenilir olan	3
	Cevapları doğru kabul eden program	1

Tablo 4. 11. Doğrusal olmayan programlama kavramına ait ifadeler

Grup	İfade	<i>f</i>
Kontrol	Doğru olmayan	4
	Doğrusal Matematik	1
Deney	Güvensiz	3
	Uygun olmayan	1
	Yanlış cevap veren program	1

Doğrusal ve doğrusal olmayan programlama arasındaki fark araştırmanın ana konusuna ilham olduğu için bu soru sorulmuştur. Scratch ile programlama eğitimine başlayan grup kısmen doğrusal olmayan programlama ortamında çalışmışlardır. Ancak ortaokul ve lise kademesinde gösterilecek olan metin tabanlı programlama dillerinde bu mantık yapısı doğrusal programlamaya dönmek zorundadır. Bu nedenle Scratch programının getirmiş olduğu doğrusal olmayan programlama mantığı ileride yaşanabilecek bazı sorunlara zemin oluşturma potansiyeline sahiptir. Doğrusal programlama sorusuna verilen yanıtlarda kelime anlamından yola çıkarak cevaplar verilmiştir. En fazla verilen cevap “Doğru olan programlama” ile kontrol grubundan verilmiştir. Deney grubundan verilen yanıtlar arasında en fazla tekrar eden cevap “Güvenilir olan” olmuştur. Doğrusal olmayan programlama sorusunda deney grubundan aynı sayıda “Güvensiz” cevabı verilmiştir. Aynı öğrencilere ait olabileceğini düşündüğüm bu cevaplar doğru yanıt değildir.

Eđitim-öđretim yılı bařında gruplar Scratch ve metin tabanlı programlama olarak iki farklı ierikle eđitime devam etmiřtir. Uygulanan kavram deđerlendirme öleđi sonrasında bir üst kademeye geiř yapılmıřtır. 7. Sınıf kademesinde gruplar metin temelli programlama araçları ile eđitim görmüřtür. Bu eđitimlerin sonunda yapılan son deđerlendirme verileri ařađıda açıklanmıřtır.

4.2. Programlama Becerileri Deđerlendirme

Tablo 4. 12. Deđerřen kuralı yazma ve örnek verebilme, dođru deđerřen seimi (Soru 1-5)

Grup	Kural verme	Örnek	Dođru Deđerřen
Deney	12	8	9
Kontrol	4	2	3

Deney ve kontrol grubunda yer alan öđrencilere deđerřen oluřturma kurallarından bildiklerini yazmaları ve bir deđerřen örneđi vermeleri istenmiřtir. Yapılan deđerlendirmeler sonucunda deney grubunda yer alan öđrencilerden 12 tanesi en az bir tane dođru kural ifade ederken, kontrol grubunda bu sayı 4 olmuřtur. Deney ve kontrol grubunda yer alan öđrenciler aynı zamanda deđerřen örnekleri oluřturmuřlardır. Buna göre deney grubundaki öđrencilerin oluřturduđu dođru örneklerin (n=8) kontrol grubuna göre (n=2) daha fazla olduđu gözlemlenmiřtir. Ařađıda deney ve kontrol grubuna ait dođru ve yanlıř örneklerden bazıları yer almaktadır. Bu soruya paralel olarak 5. Soruda öđrencilerin verilen 4 seenekten kurallara uygun oluřturulmuř deđerřkeni bulmaları istenmiřtir. Deney grubu 9 dođru cevap verirken kontrol grubu 3 dođru cevap verebilmiřtir.

Deney grubuna ait dođru örnekler;

- *sayı1*
- *eyup*
- *degisken*
- *okul*
- *araba*
- *isim*
- *soyad*

Kontrol grubuna ait dođru örnekler;

- *sayı1*
- *isim*
- *yas*

Deney grubuna ait yanlış örnekler;

- *Sayı 1*
- *Isayı*

Kontrol grubuna ait yanlış örnekler;

- *Benim adım*

Tablo 4. 13. Aritmetik operatörler ve anlamları (Soru 2)

Kural	Deney	Kontrol
-	12	12
*	12	12
/	11	12
**	4	0
//	4	0

Deney ve kontrol grubunda yer alan öğrencilere aritmetik operatörlerin görevleri sorulmuştur. Soruda “+” operatörünün görevi örnek olarak verilmiştir. Yapılan değerlendirmeler sonucunda; “-” operatörüne deney grubundan 12, kontrol grubundan 12 kişi doğru yanıt vermiştir. “*” operatörüne deney grubundan 12, kontrol grubundan 12 kişi doğru yanıt vermiştir. “/” operatörüne deney grubundan 11, kontrol grubundan 12 kişi doğru yanıt vermiştir. “**” operatörüne deney grubundan 4 doğru yanıt verilirken kontrol grubundan doğru yanıt verilmemiştir. “//” deney grubundan 4 doğru yanıt verilirken kontrol grubundan doğru yanıt verilememiştir.

Tablo 4. 14. İlişkisel operatörler ve anlamları (Soru 3)

Kural	Deney	Kontrol
!=	7	6
>	12	12
<	12	12
>=	5	7
<=	5	7

Deney ve kontrol grubunda yer alan öğrencilere ilişkisel operatörlerin görevleri sorulmuştur. Soruda “= =” operatörünün görevi örnek olarak verilmiştir. Yapılan değerlendirmeler sonucunda; “!=” operatörüne deney grubundan 7, kontrol grubundan 6 kişi doğru yanıt vermiştir. “>” operatörüne deney grubundan 12, kontrol grubundan 12 kişi doğru yanıt vermiştir. “<” operatörüne deney grubundan 12, kontrol grubundan 12 kişi doğru yanıt vermiştir. “>=” operatörüne deney grubundan 5, kontrol grubundan 7 kişi doğru yanıt vermiştir. “<=” operatörüne deney grubundan 5, kontrol grubundan 7 kişi doğru yanıt vermiştir.

Tablo 4. 15. Atama operatörleri ve anlamları (Soru 4)

Kural	Deney	Kontrol
-=	12	12
*=	12	11
/=	12	10

Deney ve kontrol grubunda yer alan öğrencilere atama operatörlerin görevleri sorulmuştur. Soruda += operatörünün görevi örnek olarak verilmiştir. Yapılan değerlendirmeler sonucunda; “-=” operatörüne deney grubundan 12, kontrol grubundan 12 kişi doğru yanıt vermiştir. “*=” operatörüne deney grubundan 12, kontrol grubundan 11 kişi doğru yanıt vermiştir. “/=” operatörüne deney grubundan 12, kontrol grubundan 10 kişi doğru yanıt vermiştir.

Tablo 4. 16. Kodların çıktısını alabilme (Soru 6 – Soru 7)

Grup	Soru 6	Soru 7
Deney	10	4
Kontrol	7	0

Deney ve kontrol gruplarının hazır verilen kodların çıktısını alabilme yetenekleri sorgulanmıştır. Bilgisayar gibi düşünebilme, problem adımlarını çözebilme konusundaki yeterlilikleri bu sorular ile ölçülmüştür. 6.soruya deney grubundan 10 kişi doğru cevap verirken, kontrol grubu 7 kişi ile doğru cevap verebilmiştir. 7. Soruya deney grubu 4 kişi ile doğru cevap verirken kontrol grubundan bu soruya doğru cevap verilememiştir.

Tablo 4. 17. Ekran yazdırma komutunu kullanma (Soru 8- Soru 9- Soru 10)

Grup	Soru 8	Soru 9	Soru 10
Deney	12	12	10
Kontrol	12	10	7

Soru 8, 9 ve 10 grupların eğitim sürecinde en çok kullandıkları fonksiyon olan print komutunu kullanabilme yeterlilikleri sorgulanmıştır. Sırasıyla ekrana “Merhaba Dünya”, kendi isimleri ve yaşlarını yazdırmaları istenmiştir. “Merhaba Dünya” yazdırma görevinde deney ve kontrol grubu 12 kişi ile doğru cevap vermiştir. Kendi isimlerini yazdırma görevinde deney grubu 12, kontrol grubu 10 kişi ile doğru cevap vermiştir. Yaşlarını yazdırma görevinde tırnak işaretini kullanarak görevi eksik olarak

yerine getiren kontrol grubunda 7 kişi doğru yanıtı vermiştir. Deney grubu 10 kişi ile görevi doğru yerine getirmiştir.

Tablo 4. 18. \n komutu ve \t komutunun görevi

Grup	\n	\t
Deney	7	5
Kontrol	6	3

Deney grubu \n komutunun görevini yazma sorununa 7 kişi ile doğru cevap verirken, kontrol grubu 5 kişi ile doğru cevap vermiştir. \t komutunun görevine deney grubu6, kontrol grubu 3 kişi ile doğru cevap vermiştir.

Tablo 4. 19. Verilen problemin çözüm kodlarını yazma

Grup	Doğru	Yanlış	Boş
Deney	6	6	0
Kontrol	2	6	4

Öğrencilerin problem çözmek için algoritma oluşturma ve kod yazma konusundaki yeterliliklerini sorgulayan soruya deney grubu 6, kontrol grubu 2 doğru yanıt vermiştir. Yanlış sayısı doğru yanıtlardan fazla olan bu soruda deney grubu 6, kontrol grubu 6 yanlış cevap vermiştir. Belirgin farkın görüldüğü boş bırakma oranında deney grubu soruya yanıt vermeyi tercih ederken, kontrol grubu 4 kişi ile boş bırakmıştır.

Gruplardan blok tabanlı programlama (Scratch) eğitimi ile müfredat ile paralel devam eden grubun, metin tabanlı programlama (Python) ile eğitime başlayan ve devam eden gruba göre yapılan son değerlendirmede; operatörlere hâkim olma, hata ayıklama, kodları çalıştırma, problem çözme ve temel kavramları açıklamada daha başarısız olmuşlardır.

5. SONUÇ, TARTIŞMA VE ÖNERİLER

Programlama eğitimi son yıllarda çok daha erken yaşlar için verilmeye başlanmış ve birçok ülke eğitim programlarında programlamaya yönelik eğitimlere yer vermiştir. Çalışmada ortaokul öğrencileri yer almıştır. Katılımcılar aynı sınıf kademesinde benzer özellik gösterecek şekilde iki gruba ayrılmıştır. Ortaokul 6. Sınıf kademesinde teorik derslerin tamamlanmasının ardından uygulamalı kodlama eğitimlerine geçişte çalışmanın uygulama ve veri toplama süreçleri başlamıştır. Kontrol grubu Scratch ile temel programlama eğitimine başlamış ve devam etmiştir. Deney grubunda Scratch programlama aracı hiç kullanılmadan doğrudan metin tabanlı programlama içeriklerine giriş yapmıştır. 7. Sınıf kademesinde gösterilecek olan Python eğitimleri temel ve basit düzeyde olacak şekilde deney grubuna 6. Sınıf kademesinde verilmeye başlanmıştır.

Yapılan ön değerlendirme grupların programlama terimleri hakkında bilgi düzeylerinin birbirine yakın olduğunu göstermiştir. Bunun sebebi iki grubun homojen olarak dağıtılması ve birbirine yakın özellikler taşıması olabilir. Katılımcılar arasında istisna olarak programlamaya karşı özel ilgisi öğrencilerde bulunmaktadır. Yukarıda bahsedilen eğitimler tamamlandıktan sonra (7. Sınıf sonu) son değerlendirme yapılmıştır. Uygulanan bu ölçek temel kavram bilgisi, hata ayıklama ve kodları çalıştırma gibi becerileri sorgulamaktadır. Çıkan sonuçlara göre deney grubu (metin tabanlı programlama eğitimi alan grup) aldıkları eğitim sonrasında temel kavramları açıklamada, hatalı kodları düzeltmede ve kodları çalıştırma konusunda kontrol grubuna üstünlük sağlamıştır. Literatür incelendiğinde 2020 de yapılan araştırmada metin temelli programlama öğretimi sonrası yapılan başarı testlerinde öğrenci grubunun %41,3 lük kısmı “Çok İyi” derecesiyle başarılı olmuştur. Yalnızca %4 lük kısım başarısız olmuştur (Karaçam Duman, 2020). Buna göre metin temelli programlama öğretiminin programlama becerilerinin ölçüldüğü başarı testinde olumlu sonuçlar verdiği görülmüştür. Değişken ile ilgili ilk uygulanan ölçekte iki grupta yanlış cevaplar verirken uygulanan son ölçekte durum farklılaşmıştır. Her iki grupta kendi alanlarında blok değişken oluşturma kurallarını öğrenmiş ve kodlama süreçlerinin içerisinde değişkeni uygun şekilde kullanmıştır. Bu durum sonucunda yapılan son değerlendirme deney grubunun değişken oluşturma ve açıklama konusunda daha başarılı olduğu izlenimini kuvvetlendirmiştir. Örnek vermeleri istenen son değerlendirmede deney grubu 9, kontrol grubu 3 doğru örnek vermiştir.

Bu durum metin tabanlı programlama gören grubun “Değişken” konusunda daha başarılı olduğunu göstermektedir.

Ancak aritmetik operatörlere verilen cevaplar incelendiğinde grupların doğru cevap sayılarının genelde eşit olduğu görülmüştür. Çıkarma (-) ve çarpma (*) operatörlerinde kontrol grubu daha fazla doğru yanıt vermiştir. Ancak çıkarma ve çarpma operatörlerinin kullanımı matematik ile birebir örtüştüğünden kodlama açısında tek başına net ölçüt olmamaktadır. Bölme (/) operatöründe verilen doğru yanıt sayıları eşittir. Programlama özgü olan üst alma (**) ve kalansız bölme (//) operatörlerine verilen yanıtlarda deney grubu doğru yanıtlara rastlanırken kontrol grubundan bu operatörlere doğru yanıt verilememiştir.

Deney ve kontrol gruplarının ölçek sonuçlarına göre arasındaki farklara bakıldığında metin tabanlı programlama eğitimi ile başlayan ve devam eden deney grubunun daha başarılı olduğu ortaya çıkmıştır. Scratch programının doğrusal programlama ve doğrusal olmayan programlama mantığını içerisinde barındırması bu program ile programlama öğrenen bireylerin metin tabanlı programlamaya geçişte problem yaşama ihtimallerinin daha yüksek olduğunu göstermektedir. Ortaokul ve lise kademelerinde gösterilen metin tabanlı programlama dilleri doğrusal programlama mantığına göre çalışmaktadır. Öğrenciler Scratch programında temelini oluşturdukları programlama alt yapısını metin tabanlıya aktarmada bu nedenlerden dolayı problem yaşamaktadır. Blok tabanlı programlama ortamlarında hazır bulunan operatörleri metin tabanlıda kendileri yazmak zorundadırlar. Bu nedenle kalıcı ve doğru öğrenme konusunda metin tabanlı programlama etkinlikleri daha başarılı sonuçlar vermiştir. Hata ayıklama ve kodları çalıştırma becerileri incelendiğinde benzer şekilde deney grubu öğrencilerinin daha başarılı sonuçlar elde ettiği görülmüştür. Bu durumun başlıca sebebi olarak blok tabanlı programlamada hata ayıklama ve kodları çalıştırma süreçlerinde programlama ortamının kod yazımında hata yapma olasılığını sınırlandırması olarak gösterilebilir. Bahsedilen durumun temel programlama eğitiminden ileri programlamaya geçişte özellikle hata tespiti ve ayıklamayı olumsuz etkileme olasılığından söz edilebilir. Metin tabanlı programlamada yazım düzeninden mantıksal bütünlüğe kadar tüm detaylara dikkat etmesi gerektiği için bu durumun daha başarılı sonuçlara yol açması beklenebilir. Koorsse ve diğerleri 2015 yılında Delphi programlama dilinin öğretildiği üç grup üzerinde araştırma yapmışlardır. Çalışmaya göre Delphi'ye ek olarak Scratch, B# ve Robomind programlama ortamlarından biri

öğretim materyali olarak kullanılmış bunun yanında kontrol grubu yalnızca Delphi programlama dili ile öğretime devam etmiştir. Çalışmanın sonucuna göre Scratch, B# ve Robomind ortamları öğrenci başarısını arttırmış ancak kontrol grubu ile arasında anlamlı fark olmadığı görülmüştür (Koorse, Cilliers, & Calitz, 2015). Elde ettiğimiz verilere göre Scratch ortamında programlama eğitimi gören kontrol grubu, metin temelli programlama araçları ile programlama eğitimi gören deney grubuna oranla programlama becerileri açısından daha başarısız olmuştur.

Döngü kavramına verilen yanıtlar incelendiğinde çalışma gruplarının “Dönmek” ifadesi üzerinde durduğu görülmektedir. Deney grubu “Tekrar kullanma” ve “Tersine çevirme” ifadelerine yer vermiştir. Kontrol grubu “Dönmek” ifadesinin dışında “Yol almak”, “Geri çağırmak” ifadelerini cevap olarak vermiştir. İki grubun soruyu cevaplama oranlarına bakıldığında deney grubunun bu soruya daha fazla öğrenci ile yanıt verdiği görülmektedir. 12 kişiden 10’u deney grubundan bu soruya yanıt verebilmiştir. Ancak kontrol grubunda yanıt veren öğrenci sayısı 6 ile sınırlı kalmıştır. Bu kısımda gruplar arasındaki farkın sahip oldukları programlama becerisi üzerinden değil kelime anlamından çıkarımda bulunma yeteneklerine göre farklılık gösterdiği düşünülmektedir. Bu düşüncenin ortaya çıkmasında grupların programlama becerilerinin araştırmacı tarafından gözlenmesi ve diğer kavramlara verilen yanıtlarda farklı grupların öne çıkması etkili olmuş olabilir.

Fonksiyon kavramına verilen yanıtlar incelendiğinde “İşlem” cevabı ön plana çıkmaktadır. Gruplar arasında kontrol grubu 8 öğrenci ile bu cevabı verirken deney grubu 4 öğrenci ile bu cevabı verebilmiştir. Yanıtların doğru ve yeterince açıklayıcı olmadığı bu soruda kontrol grubu 10, deney grubu 7 öğrenci ile cevap verebilmiştir. Öğrencilerin Matematik dersi ile bağlantı kurarak bu cevabı verdiği düşünülmektedir.

Eğer sorgusunun sorulduğu soruda grupların cevap verme oranı diğer sorulara kıyasla oldukça yüksektir. Bu soruda bazı öğrenciler birden fazla cevap vermiş ve bu cevaplar tabloda sayısal olarak belirtilmiştir. “Koşul” cevabı iki grupta toplamda 13 kişi tarafından verilirken kelime olarak farklı ancak anlam olarak aynı olan “Şart” cevabı da 2 öğrenci tarafından verilmiştir. Grupların doğru cevap verme oranı yüksek olan bu soruda öğrencilerin kelime anlamı ve Türkçe dersinde “Koşul-şart cümleleri” konusundan yaptıkları çıkarımların etkili olduğu düşünülmektedir. Eğer ise değilse sorusunda gruplar arasında üst düzey bir farklılık ortaya çıkmaktadır. Bu farklılık doğru cevap verme açısından değil bu soruyu cevaplama açısından ortaya çıkmıştır.

Deney grubundan 11 öğrenci bu soruya “Eğer sorgusu” cevabını vermiştir. Ancak bu cevap dikkate alınmamaktadır. Kontrol grubu 7 öğrenci ile “Eğer sorgusunun tersi” cevabı vermiştir.

Öğrencilerin Türkçe dersinden çıkarımda bulunabilecekleri bir başka soru “Ve-veya” operatörleridir. Cevaplarda deney grubundan 4 öğrencinin benzer şekilde verdiği örnekler öne çıkmaktadır. Bu örnekler Scratch programındaki blok kodlardan bazılarıdır. Bunun haricinde verilen cevaplarda doğru açıklama bulunmamaktadır.

Değişken ve değişken oluşturma kuralları sorularında öğrencilerden doğru yanıtlar gelmemiştir. Yanıtların arasında dikkat çeken bir durum deney grubundan gelen “Dikdörtgen” yanıtı olmuştur. Akış şemasında değişken tanımlamanın şematik gösterimi olan dikdörtgen cevabı doğru olarak nitelendirilmiştir. 2 öğrenci tarafından verilen bu cevabın özel ilgisi olan öğrenciler olduğu düşünülmektedir. Öğrenciler ölçeğin uygulandığı zamana kadar örgün eğitimde programlama eğitimi verilmemiş akış şeması konusu okutulmamıştır.

Başlatıcı kavramına verilen cevapların sadece kelime anlamına göre verilmiş cevaplar olduğu görülmektedir. Deney grubundan verilen “Kodlamayı çalıştırır” cevabı 5 öğrenci tarafından verilirken bu cevap verilen cevaplar arasında “Programı başlatan sorgu” ile doğru yanıtı en yakın cevaptır. Doğrusal ve doğrusal olmayan programlamanın sorulduğu soruda öğrencilerin tamamen kelime anlamı ve tahmin ile verdiği cevaplar bulunmaktadır.

Çalışmanın ilk bölümünün kavram değerlendirme süreciyle tamamlanmasından sonra ikinci aşamanın sonucunda Programlama becerileri değerlendirme ölçeği her iki grupta yer alan katılımcılara ortak olarak uygulanmıştır. Uygulamadan önce iki grupta yer alan öğrenciler aynı programlama eğitimine tabi tutulmuşlardır.

Değişken oluşturma kuralı ve bir değişken örneği istenen soruda gruplar arasında dikkate değer bir fark ortaya çıkmıştır. Deney grubu öğrencilerinin tamamı en az bir doğru değişken oluşturma kuralı verebilmiştir. Örnek değişken oluşturma konusunda deney grubunda 3 öğrenci cevap verememiştir. Kural veren öğrencilerden bir kural vererek beraberinde örnek ile kuralı anlatmaya çalışan katılımcılar tespit edilmiştir. Ancak bu şekilde yol izleyen öğrencilerin verdikleri cevapların tamamı doğru değildir. Kontrol grubu incelendiğinde yalnızca 4 öğrenci en az bir kural

yazabilmiştir. Doğru değişken örneği verebilen öğrenci sayısı 3 ile sınırlı kalmıştır. Buna göre deney grubunun değişken oluşturma kurallarına hâkim ve örnek değişken oluşturabilme becerisine daha iyi seviyede sahip olduğu görülmektedir. Kontrol grubunun değişken oluşturma kuralları ve örnek değişken oluşturma becerisinde deney grubuna oranla oldukça geride kaldığı görülmektedir. Grupların programlama eğitiminin temelini farklı platformlarda alıyor olması bu duruma net bir şekilde sebep olduğu düşünülmektedir.

Aritmetik, ilişkisel ve atama operatörleri incelendiğinde gruplar arasında belirgin bir fark görülmemektedir. “-“, “*”, “/” operatörlerine iki grupta 12’şer öğrenci ile doğru yanıtlar verebilmiştir. Yalnızca deney grubunda “/” operatörüne 1 öğrenci yanlış cevap verdiği görülmektedir. Ancak aritmetik operatörlerde “**” ve “//” operatörlerine deney grubu 4 öğrenci ile cevap verirken kontrol grubu bu iki operatöre cevap verememiştir. İlişkisel operatörlerde benzer sayıda doğru yanıtlar verildiği görülmektedir. Deney grubunun geride kaldığı “>=” ve “<=” operatörlerinde kontrol grubu 7, deney grubu 5 öğrenci ile doğru cevap verebilmiştir. Kontrol grubunun bu iki operatörde sayıca fazla doğru yanıt vermesi blok tabanlı programlama aracında hazır blok olarak bulunmamasından dolayı şaşırtıcı sonuç olmuştur. Atama operatörlerinde ise gruplar arasında belirgin bir fark görülmemektedir.

Verilen kodların çıktısını alabilme becerisinin sorgulandığı soruda deney ve kontrol grupları arasındaki fark diğer durumlara göre daha belirginleşmiştir. İki soruda toplam 14 doğru cevap veren deney grubunu kontrol grubu yalnızca 7 cevapla takip edebilmiştir. Bu durum kodların çalışma mantığını ve adım adım kodları çalıştırma becerisine deney grubunun hâkim olduğunu göstermektedir.

Ekranı “Hello World”, öğrencilerin kendi isimleri ve yaşlarının yazdıran Python kodlarını yazmaları istenen soruda “Hello World” yazdırma görevini iki grupta tam sayıda öğrenci ile doğru şekilde yapabilmişlerdir. Kendi isimlerini yazdırma görevinde deney grubu 12 ile tam sayıda doğru yanıt verirken kontrol grubunda sadece 2 öğrenci tırnak işareti ve parantez hatası yapmışlardır. Yaşlarının yazdırılması görevinde deney grubu 12 öğrenci ile tam sayıda doğru yanıt vermiştir. Ancak kontrol grubunda 7 öğrenci doğru yanıt verebilmiştir. Grupta yanlışların sebebi sayısal değer için tırnak içine yazılması, parantezin unutulmasıdır. Bu 3 soru ile öğrencilerin basit bir görevi yerine getirme konusundaki becerileri ölçülürken deney grubunun tırnak, parantez gibi metin tabanlı programlamanın syntax (sözdizim) kurallarına dikkat

konusunda kontrol grubunun önüne geçtiği görülmektedir. Bu farka temel programlama eğitiminin deney grubuna metin tabanlı programla ile verilmesinden kaynaklandığı düşünülmektedir. 2014 yılında yapılan çalışmaya göre Scratch programının düz anlatıma oranla algoritma öğretimi konusunda anlamlı bir fark bulunamamıştır (Tekerek, Altan, & Gündüz, 2014). Araştırmada Scratch programı ile algoritma kavramı anlatılmıştır. Buna ek olarak görsel programlama araçları kullanılmadan düz anlatımla algoritma kavramı anlatılmıştır. İki grup arasında anlamlı bir fark bulunamamıştır. Buna karşılık çalışmanın sonucuna göre metin temelli programlama araçları ile eğitim gören grup, blok temelli programlama ortamında eğitim gören kontrol grubuna oranda kod yazabilme, problem çözümü için algoritma oluşturabilme konusunda daha başarılı olabilmektedir. Elde edilen bulguların belirli oranda literatürden ayrıldığı söylenebilir.

Verilen bir problemin çözümü için gerekli kodların yazılması istenen soruda bazı öğrenciler algoritmasını yazarak cevap vermişlerdir. Deney grubu 6 öğrenci ile doğru yanıt verirken soruyu boş bırakan öğrenci olmamıştır. Kontrol grubu 4 kişi ile bu soruyu boş bırakırken yalnızca 2 doğru cevap verebilmişlerdir. Bu durumdan yola çıkarak deney grubun programlama ve problem çözme becerilerinin kontrol grubuna oranla daha ileri düzeyde olduğu ortaya çıkmaktadır. 2020 yılında yapılan çalışmaya göre 4-5 yaş okul öncesi çocuklarda uygulanan blok temelli programlama eğitimi sonrası sonuçlara göre 4 yaş grubu problem çözme becerilerinde anlamlı düzeyde gelişim gösterirken, 5 yaş grubu çocuklarda herhangi bir farklılık oluşmadığı gözlenmiştir (Çiftci & Bildiren, 2020). Ortaya koyulan bu sonuçlara göre blok temelli programlama eğitimi 5 yaş ve üzeri gruplarda istatistiksel olarak problem çözme becerilerinde anlamlı bir fark yaratmamaktadır.

Öneriler

1. Ortaokul kademesinde programlama eğitimi blok temelli araçlar yerine metin temelli araçlar ile verilmelidir.
2. Metin temelli araçlar ile verilen eğitim programlamanın temel mantığı, söz dizim kuralları gibi temel bilgiler ile başlamalıdır.
3. Kademe ilerledikçe sınıf düzeyine göre detaylara girilmeli, uygulamalı olarak programlama eğitimi verilmelidir.

4. Metin temelli programlama araçları ile programlama eğitimi verilirken blok temelli araçlar ödüllendirme ve oyunlaştırma adına kısmen eğitime dahil edilebilir.
5. Metin temelli programlama araçları öğrenciler için sıkıcı olabilir ancak eğitimin başında öğrencilere sonuçta ulaşabilecekleri becerilerin sonucunda yapabilecekleri örnekler gösterilebilir. Hedefe ulaşmak adına sıkıcılık ikinci planda kalacaktır.
6. Basit ve öğrencilerin başarıya ulaşabilecekleri etkinlikler planlanmalıdır.



KAYNAKÇA

- Akbıyık, S., Altun, D., Bal, N., Bütüner, R., Şahin, S., & Koçali, M. (2020). *Herkes İçin Python Programlama Dili*. Millî Eğitim Bakanlığı Öğretmen Yetiştirme ve Geliştirme Genel Müdürlüğü.
- Akpınar, Y., & Altun, A. (2014). Bilgi Toplumu Okullarında Programlama Eğitimi Gereksinimi. *Elementary Education Online*, 1-4.
- Anderson, L., & Krathwohl, D. (2001). *A Taxonomy for Learning, Teaching, and Assessing*. New York: Pearson Education.
- Arabacıoğlu, T., Bülbül, H., & Filiz, A. (2007). Bilgisayar Programlama Öğretiminde Yeni Bir Yaklaşım. *Akademik Bilişim* (s. 193-197). Kütahya: Dumlupınar Üniversitesi.
- Atun, H. (2018). Teknolojik pedagojik alan bilgisi (tpac) çerçevesi ile oluşturulmuş programlama eğitiminin öğrenme çıktıları üzerine etkileri (Order No. 28676185). Available from ProQuest Dissertations & Theses Global. (2572152853). Retrieved from <https://www.proquest.com/dissertations-theses/teknolojik-pedagojik-alan-bilgisi-tpac-çerçevesi/docview/2572152853/se-2>
- Aytekin, A., Sönmez Çakır, F., Yücel, Y. B., & Kulaozü, İ. (2018). Geleceğe Yön Veren Kodlama Bilimi ve Kodlama Öğrenmede Kullanılabilecek Bazı Yöntemler. *Avrasya Sosyal ve Ekonomi Araştırmaları Dergisi*, 24-41.
- Bayman, P., & Mayer, R.E. (1988). Using conceptual models to teach BASIC computer programming. *Journal of Educational Psychology*, 80, 291-298.
- Beck, K., & Andres, C. (2004). *Extreme Programming Explained: Embrace Change by Kent Beck*. Boston: Addison-Wesley.
- Çelik, S., Kul, Ü., & Çalık Uzun, S. (2017). Ortaokul Matematik Dersi Öğretim. *Abant İzzet Baysal Üniversitesi Eğitim Fakültesi Dergisi*, 775-795.
- Crews, T., & Murphy, C. (2004). *Programmig Right From The Start with Visual Basic.NET*. New Jersey: Upper Saddle River, NJ : Pearson/Prentice Hall.
- Çiftci, S., & Bildiren, A. (2020). The effect of coding courses on the cognitive abilities and problem-solving skills of preschool children. *Computer Science Education*, 3-21.
- Çobanoğlu, B. (2018). *Herkes için Python*. İstanbul: Pusula 20 Teknoloji ve Yayıncılık A.Ş.
- Demir, Ö. (2019). Eşli Programlamada Çeşitli Bireysel Farklılıkların Grup Uyumu, Eşli Programlamada Çeşitli Bireysel Farklılıkların Grup Uyumu,. Ankara: Hacettepe Üniversitesi Eğitim Bilimleri Enstitüsü.
- Demirer, V., & Sak, N. (2016). Programming Education And New. *Eğitimde Kuram ve Uygulama*, 521-546. <https://dergipark.org.tr/tr/pub/eku/issue/26697/280853> adresinden alındı
- Fiş Erümit, S., & Kalelioğlu, F. (2019). Programlama Öğretiminde Oyunlaştırma. 7. *Uluslararası Öğretim Teknolojileri ve Öğretmen Eğitimi Empozyumu* (s. 568-581). Antalya: Trabzon Üniversitesi-Karadeniz Teknik Üniversitesi.
- Gülbahar, P. D., & Karal, P. D. (2018). *Kuramdan Uygulamaya Programlama Öğretimi*. Ankara: Pegem Akademi.

- Gündüz, S. (2007). *Matematik Projeleri ve Sınıf Etkinlikleri*. Toroslu Kitaplığı.
- Karaçam Duman, N. (2020, Ocak). Metin Temelli Programlama Öğretimi: Ortaokul Öğrencilerinin Bilgisayarca Düşünme Becerileri ve Akademik Başarılarının İncelenmesi. İstanbul, Bahçeşehir.
- Kert, S., & Uğraş, T. (2009). Programlama Eğitiminde Sadelik ve Eğlence: Scratch Örneği. *In The First International Congress of Educational Research*. Çanakkale.
- Koorsse, M., Cilliers, C., & Calitz, A. (2015). Programming assistance tools to support the learning of IT programming in South African secondary schools. *Computer & Education*, 162-178.
- Lutz, M. (2001). *Programming Python, Second Edition*. Sebastopol: O'REILLY.
- Maloney, J., Resnick, M., Rusk, N., Silverman, B., & Eastmond, E. (2010). The Scratch Programming Language and Environment. *ACM*, 1-15.
- MEB. (2006). TTK Başkanlığı 28.08.2006 tarihi 347 sayılı kararı. *Tebliğler Dergisi*, 69.
- MEB. (2012). TTK Başkanlığı 05.09.2012 tarihi 150 sayılı kararı. *Tebliğler Dergisi*, 75.
- MEB. (2018). (2018). [meb.gov.tr. 2023vizyonu.meb.gov.tr: https://2023vizyonu.meb.gov.tr/doc/2023_EGITIM_VIZYONU.pdf](https://2023vizyonu.meb.gov.tr/doc/2023_EGITIM_VIZYONU.pdf) adresinden alındı
- Millî Eğitim Bakanlığı. (2018). *Bilişim Teknolojileri ve Yazılım Dersi Öğretim Programı (Ortaokul 5 ve 6. Sınıflar)*. Ankara: MEB.
- Mitchel Resnick ve Arkadaşları. (2009). *web.media.mit.* [https://web.media.mit.edu/ https://web.media.mit.edu/~mres/papers/Scratch-CACM-final.pdf](https://web.media.mit.edu/~mres/papers/Scratch-CACM-final.pdf) adresinden alındı
- Papert, S. (1980). *Children, Computers, and Powerful Ideas*. New York: Basic Books, Inc., Publishers.
- Saracaoğlu, A., & Çelik, B. (2018). Web Tasarımı ve Programlama Dersi Öğretiminde Proje Tabanlı Öğrenme Yaklaşımı Kullanımının Öğrenci Görüşlerine Göre Değerlendirilmesi. *Kastamonu Education Journal*, 323-333.
- Saygıner, Ş., & Tüzün, H. (2017). İlköğretim Düzeyinde Programlama Eğitimi: Yurt Dışı ve Yurt İçi Perspektifinden Bir Bakış. *Akademik Bilişim Konferansı* (s. 1-5). Ankara: Hacettepe Üniversitesi.
- Seferoğlu, S. S. (2007). İlköğretim Bilgisayar Dersi Öğretim Programı: *Eurasian Journal of Educational Research*, 99-11.
- Tekerek, M., Altan, T., & Gündüz, İ. (2014). FATİH Projesinde Tablet PC Kullanımına Yönelik Öğrenci Tutumlarının İncelenmesi. *Journa of Information Technologies*, 21-27.
- Yecan, E., Özçınar, H., & Tanyeri, T. (2017). Bilişim Teknolojileri Öğretmenlerinin Görsel Programlama Öğretimi Deneyimleri. *Elementary Education Online*, 377-393.
- Yüksel, S. (2007). Bilişsel Alanın Sınıflamasında (Taksonomi) Yeni. *Türk Eğitim Bilimleri Dergisi*, 479-509.
- Yükseltürk, P. (2020). Arduino Programlama ile Stem Etkinlikleri Semineri: Öğretmen Adaylarının Stem Hakkındaki Görüşleri. *13. Uluslararası Eğitim Camiası*

Sempozyumu Tam Metinleri (s. 685-690). Rize: 13. Uluslararası Eğitim Camiası Sempozyumu Tam Metinleri.

Yükseltürk, P., & Curaoğlu, D. (2019). Programlama Eğitiminde Blok Tabanlı Araç Kullanmanın Zorlukları. *Uluslararası Eğitim Bilimleri Sempozyumu* (s. 10-14). Alanya: Asos Yayınları.

Zurnacı, B., & Turan, Z. (2022). Türkiye'de Okul Öncesinde Kodlama Eğitimine İlişkin Yapılan Çalışmaların İncelenmesi. *Kocaeli Üniversitesi Eğitim Dergisi*, 258-286.



EKLER

Ek 1: Araştırma İzni Talebi



T.C.
AĞRI VALİLİĞİ
İl Millî Eğitim Müdürlüğü

Sayı : E-78971437-20-40224379
Konu : Araştırma Uygulama İzni
(Yunus Emre ÇANAKCI)

31/12/2021

VALİLİK MAKAMINA

İlgi: Ağrı Valiliği 33309172 sayılı ve 29.09.2021 tarihli yazı

Ondokuz Mayıs Üniversitesi Bilgisayar ve Öğretim Teknolojileri Eğitimi Ana Bilim Dalı yüksek lisans öğrencisi Yunus Emre ÇANAKCI, Doç. Dr. Polat ŞENDURUR danışmanlığında yürüttüğü "Temel Programlama Eğitimine Blok Temelli Programlama ve Metin Temelli Programlama ile Başlamanın İleri Programlamaya Etkisi" başlıklı tez çalışmasını Ağrı Patnos ilçesinde müdürlüğümüze bağlı ekli listede belirtilen okullarda uygulanması uygun görülmüştür. Makamlarınızca da uygun görülmesi halinde olurlarınıza arz ederim.

Hasan KÖKREK
İl Millî Eğitim Müdürü

OLUR
Kerem TÜRKER
Vali a.
Vali Yardımcısı

EK:
- İl MEM Komisyon tutnağı
-İlgide Kayıtlı Yazı ve Ekleri



İl Millî Eğitim Müdürlüğü Merkez Ağrı A-Ge Birimi

İ. No : 04472/2009441
E-posta : icm@meb.gov.tr
Web : icm.meb.gov.tr

Bu belge güvenli elektronik imza ile onaylanmıştır.

Belge Doğrulama Adresi : <https://www.sakipya.gov.tr/mem-ebys>

İmza Adresi : Faks : _____

Bu belge güvenli elektronik imza ile onaylanmıştır. <https://www.sakipya.gov.tr/mem-ebys> dbca-45dc-3239-8a04-0e67 kodu ile teyit edilebilir.

Ek 2: Valilik Oluru



T.C.
PATNOS KAYMAKAMLIĞI
İlçe Milli Eğitim Müdürlüğü

Sayı : E-76917568-605.01-40434456
Konu : Araştırma Uygulama İzin
(Yunus Emre ÇANAKÇI)

04.01.2022

PATNOS KAYMAKAMLIĞI
(İlçe Milli Eğitim Müdürlüğü)

İlgili: Ağrı Valiliği'nin 31/12/2021 tarihli ve 40224379 sayılı olur yazısı.
İl Milli Eğitim Müdürlüğü'nün 03.01.2022 tarih ve 40331488 sayılı yazısı.

İlgide kayıtlı olur yazısına istinaden Ondokuz Mayıs Üniversitesi Bilgisayar ve Öğretim Teknolojileri Eğitimi Ana Bilim Dalı yüksek lisans öğrencisi Yunus Emre ÇANAKÇI, Doç. Dr. Polat ŞENDURUR danışmanlığında yürüttüğü "Temel Programlama Eğitimine Blok Temelli Programlama ve Metin Temelli Programlama ile Başlamanın İleri Programlamaya Etkisi" başlıklı tez çalışmasını Ağrı Patnos ilçesinde müdürlüğümüze bağlı ekli listede belirtilen okullarda uygulanması hususunda olur yazısı ekinizde sunulmuştur. Denetimi okul müdürlüklerince yapılmak üzere gerekli kolaylıkların sağlanması hususunda;

Gereğini bilgilerinize rica ederim.

Yakup TAŞDEMİR
Milli Eğitim Müdürü

Ek:
- Valilik Oluru
-İlgide Kayıtlı Yazı ve Ekleri

Dağıtım:

Adres:

Telefon No:

E-Posta:

İnternet Adresi: www.meb.gov.tr

Bu belge güvenli elektronik imza ile oluşturulmuştur.

Belge Doğrulama Adresi: <https://www.turkiye.gov.tr/meb-ebys>

Bilgi için:

Ünvan / Maaş:

İzmir Adresi / Yaka:

Bu belge güvenli elektronik imza ile oluşturulmuştur. Belge doğrulama için <https://www.turkiye.gov.tr/meb-ebys> adresinden 0960 - 90 00 3997 - 5466 - 1048 kodu ile bilgi alınabilir.



Ek 3: İzin Komisyonu

AĞRI İL MİLLİ EĞİTİM MÜDÜRLÜĞÜ
ARAŞTIRMA UYGULAMA İZİN KOMİSYONU BAŞKANLIĞI

Müdürlüğümüze Araştırma Uygulama İzni için 29.09.2021 tarihli ve 33309172 sayılı yazı/dilekçe ile başvuran Ondokuz Mayıs Üniversitesi/Okulu Yunus Emre ÇANAKÇI adlı öğrencinin ekte belirtilen 2020/2 sayılı Milli Eğitim Bakanlığı Yenilik ve Eğitim Teknolojileri Genel Müdürlüğü Araştırma Uygulama İzinleri Genelgesi doğrultusunda Müdürlüğümüze gönderilen evraklar komisyonumuzca incelenmiş olup herhangi bir eksiklik ve aykırılık olmadığından araştırma uygulama izni verilmesi uygun görülmüştür. İşbu komisyonumuzca tutanak tutulup imzalanmıştır.

Nesim ZARİÇ
Öğretmen
Üye

Mehmet Baki DURSUN
Şef
Üye

Selahattin BİLGİÇ
Komisyon Başkanı
(Şube Müdürü)

Ek 4: Enstitü İzin Başvurusu



T.C.
ONDOKUZ MAYIS ÜNİVERSİTESİ
Lisansüstü Eğitim Enstitüsü Müdürlüğü

Sayı : E-72975315-044-123999
Konu : Ökçek İzni (Yunus Emre ÇANAKCI)

AĞRI İL. MİLLİ EĞİTİM MÜDÜRLÜĞÜNE

İlgi : Bilgisayar ve Öğretim Teknolojileri Eğitimi Ana Bilim Dalı Başkanlığı 27.09.2021 tarihli ve E-13886282-100-123848 sayılı yazısı.

Enstitümüz Bilgisayar ve Öğretim Teknolojileri Eğitimi Ana Bilim Dalı yüksek lisans öğrencisi Yunus Emre ÇANAKCI, Doç. Dr. Polat ŞENDURUR danışmanlığında yürüttüğü " Temel Programlama Eğitimine Blok Temelli Programlama ve Metin Temelli Programlama ile Başlamanın İleri Programlamaya Etkisi" başlıklı tez çalışmasını veri toplamak amacıyla Müdürlüğünüz bünyesindeki Patnos İlçesi Değirmentezli Ortaokulunda öğrenim gören 6. ve 7. sınıf öğrencileriyle ökçek çalışması yapmak istediğini bildirmiştir.

İlgili öğrencinin danışmanı Doç. Dr. Polat ŞENDURUR'a ait dilekçe, başvuru formları, tez önerisi, çalışma verileri ve Etik Kurul Kararı ekte olup söz konusu çalışmanın yapılabilmesi için izin verilmesini arz ederim.

Prof. Dr. Ali BOLAT
Enstitü Müdürü

Ekte:
1- Başvuru Formları, Tez Önerisi, Etik Kurul Kararı (10 Sayfa)
2- Yunus Emre ÇANAKCI-Ökçek-1 (2 Sayfa)
3- Yunus Emre ÇANAKCI-Ökçek-2 (2 Sayfa)

[Bu belge, güvenli elektronik imza ile onaylanmıştır.](#)

Belge Doğrulama Kodu : BMG5-C15G-05TM

Belge Doğrulama Adresi : <http://ehyocpa.omu.edu.tr>

Adres: Lisansüstü Eğitim Enstitüsü Müdürlüğü

Bilgi İçin: Yunus ŞİMŞEK DİLBER

Karapınar/SAMSUN

Bilgisayar İşletmeni

Tel/Fax No :

Fax No :

e-Posta :

İnternet Adresi : <http://www.omu.edu.tr/>

Tel/Fax No :



Ek 5: Katılım Belgesi



Ek 6: Etik Kurul



ONDOKUZ MAYIS ÜNİVERSİTESİ
SOSYAL VE BEŞERİ BİLİMLER ARAŞTIRMALARI ETİK KURUL KARARLARI

KARAR TARİHİ	TOPLANTI SAYISI	KARAR SAYISI
25.06.2021	06	2021/567

KARAR NO: 2021-567
KARAR TARİHİ: 25.06.2021
Üniversitemiz Lisansüstü Eğitim Enstitüsü öğrencisi Yunus Emre ÇANAKCI'nın Doç. Dr. Polat ŞENDURUR danışmanlığında " Temel Programlama Eğitimine Blok Temelli Programlama ve Metin Temelli Olarak Başlamamanın İleri Programlamaya Etkisi" isimli yüksek lisans tezine ilişkin ölçek ve mülakat çalışmalarını içeren 18391 sayılı dilekçesi okunarak görüşüldü.

Üniversitemiz Lisansüstü Eğitim Enstitüsü öğrencisi Yunus Emre ÇANAKCI'nın Doç. Dr. Polat ŞENDURUR danışmanlığında " Temel Programlama Eğitimine Blok Temelli Programlama ve Metin Temelli Olarak Başlamamanın İleri Programlamaya Etkisi" isimli yüksek lisans tezine ilişkin ölçek ve mülakat çalışmalarının kabulüne oy birliği ile karar verildi.

Ek 7: Tez Benzerlik Raporu

Tez-Benzerlik-Raporu

ÖZETLENEN RAPORU

% 11	% 10	% 1	%
BENZERLİK ENDEKSİ	İNTERNET KAYNAKLARI	YAYINLAR	ÖĞRENCİ ÖDEVLERİ

BİRİNCİL KAYNAKLAR

1	9lib.net İnternet Kaynağı	% 3
2	ittes.org.tr İnternet Kaynağı	% 3
3	bilgebt.com İnternet Kaynağı	% 1
4	fatihbutun.com İnternet Kaynağı	% 1
5	efdergi.ibu.edu.tr İnternet Kaynağı	<% 1
6	dspace.ankara.edu.tr İnternet Kaynağı	<% 1
7	dergipark.org.tr İnternet Kaynağı	<% 1
8	www.doczz.it İnternet Kaynağı	<% 1
9	acikbilim.yok.gov.tr İnternet Kaynağı	<% 1

Ek 9: Kavram Deęerlendirme Ölçeęi

1. Döngü nedir?

Cevap:

○ Bir örnek verir misiniz?

Cevap:

2. Fonksiyon nedir?

Cevap:

○ Bir örnek verir misiniz?

Cevap:

3. Eğer sorgusu nedir?

Cevap:

○ Bir örnek verir misiniz?

Cevap:

4. Eğer ise değilse sorgusu nasıl çalışır?

Cevap:

○ Bir örnek verir misiniz?

Cevap:

5. Ve sorgusu nasıl çalışır?

Cevap:

○ Bir örnek verir misiniz?

Cevap:

6. Veya sorgusu nasıl çalışır?

Cevap:

○ Bir örnek verir misiniz?

Cevap:

7. Deęişken nedir?

Cevap:

○ Bir örnek verir misiniz?

Cevap:

8. Değişken oluşturma kuralları nedir?

Cevap:

- Bir örnek verir misiniz?

Cevap:

9. Başlatıcı nedir?

Cevap:

- Bir örnek verir misiniz?

Cevap:

10. Doğrusal programlama nedir?

Cevap:

- Bir örnek verir misiniz?

Cevap:

11. Doğrusal olmayan programlama sence ne olabilir?

Cevap:

- Bir örnek verir misiniz?

Cevap:

12. Bilgisayarın soru sormasını hangi yapılar ile sağlarsınız?

Cevap:

- Bir örnek verir misiniz?

Cevap:

Ek 10: Programlama Becerileri Ölçeği

1. Değişken isimlendirme kuralları nelerdir? Doğru bir değişken ismi yazınız.
2. Aritmetik operatörlerin anlamlarını yanlarına yazınız.

İşaret	İşlem
+	Toplama
-	
*	
/	
**	
//	

3. İlişkisel operatörlerin anlamlarını yanlarına yazınız.

İşaret	İşlem
==	Eşit mi?
!=	
>	
<	
>=	
<=	

4. Atama operatörlerinin anlamlarını yanlarına yazınız.

İşaret	İşlem
+=	Arttırarak atama
-=	
*=	
/=	

5. Aşağıdaki değişken adlandırmalarından hangisi yanlıştır?

_sinav_puani1=80

sinavPuani=90

1sinavpuani=100

SınavPuani=70

6. Aşağıdaki kod çalıştırıldığında ekrana “True” veya “False” değerlerinden hangisi yazılır? $sayi1=8*5/2$

```
print (sayi1>20)
```

7. Aşağıdaki kod çalıştırıldığında ekrana ne yazılır?

```
sehir='Ankara'
```

```
sehir='İstanbul'
```

```
print ('Şehir: ', sehir)
```

8. Ekrana “Merhaba Dünya” yazan programı yazınız.

9. print fonksiyonu ile adınızı ekrana yazdırınız.

10. print fonksiyonu ile yaşınızı rakam olarak ekrana yazdırınız.

11. $\backslash n$ ifadesinin görevi nedir?

12. $\backslash t$ ifadesinin görevi nedir?

13. Kullanıcıdan iki sayı alan ve toplamlarını ekrana yazdıran programı yazınız.

Öz Geçmiş

Yunus Emre ÇANAKCI, Samsun Onur Ateş Anadolu Lisesi'ni bitirdikten sonra Eskişehir Anadolu Üniversitesi Eğitim Fakültesi Bilgisayar ve Öğretim Teknolojileri Eğitimi Öğretmenliği bölümünden 27.05.2017 tarihinde mezun oldu. İkinci olarak Anadolu Üniversitesi İktisat Fakültesi Uluslararası İlişkiler bölümünden 10.09.2020 tarihinde mezun oldu. 2019 yılında OMÜ Bilgisayar ve Öğretim Teknolojileri Eğitimi Ana Bilim Dalı Yüksek Lisans programına girdi. Proje Danışmanlığı, Zekâ Oyunları, Yazılım Geliştirme Uzmanlığı, Kotlin ile Mobil Uygulama Geliştirme, Python, Robotik Kodlama Eğitmenliği, Unity, Network CCNA-1, Arduino, Robot Tasarımı, E-Sınav izleme sorumlusu sertifikaları bulunmaktadır. İlgi alanı olarak bağlama ve yüzme ile ilgilenmektedir.

İletişim Bilgileri

ORCID ID : 0000-0002-1785-0867

Yayımlar:

1. Çanakci, Y. E., & Şendurur, P. (2021). Temel Programlama Eğitimine Blok Temelli ve Metin Temelli Araçlar. 8. *Uluslararası Öğretim Teknolojileri ve Öğretmen Eğitimi Sempozyumu*, (s. 117-118). Trabzon