



T.C.
DOKUZ EYLÜL ÜNİVERSİTESİ
EĞİTİM BİLİMLERİ ENSTİTÜSÜ



BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ EĞİTİMİ ANABİLİM DALI
BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ ÖĞRETMENLİĞİ PROGRAMI
YÜKSEK LİSANS TEZİ

CODE.ORG PLATFORMUNUN 6. SINIF ÖĞRENCİLERİNİN
PROGRAMLAMA ÖĞRENİMİNE ETKİSİ

ÜMİT KARADUMAN

İzmir
2019

**BİLGİSAYARVE ÖĞRETİM TEKNOLOJİLERİ EĞİTİMİ ANABİLİM DALI
BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ ÖĞRETMENLİĞİ PROGRAMI
YÜKSEK LİSANS TEZİ**

**CODE.ORG PLATFORMUNUN 6. SINIF ÖĞRENCİLERİNİN
PROGRAMLAMA ÖĞRENİMİNE ETKİSİ**

ÜMİT KARADUMAN

**İzmir
2019**

ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ

Yüksek lisans tezi olarak sunduğum “Code.org Platformunun 6. Sınıf Öğrencilerinin Programlama Öğrenimine Etkisi” adlı çalışmanın içerdiği fikri izinsiz başka bir yerden almadığımı; çalışmamın hazırlık, veri toplama, analiz ve bilgilerin sunumu olmak üzere tüm aşamalarında ve bölümlerinin yazımında bilimsel etik ilke ve kurallara uygun davrandığımı, tez yazım kurallarına uygun olarak hazırlanan bu çalışmada kullanılan her türlü kaynağa eksiksiz atıf yaptığımı ve bu kaynaklara kaynakçada yer verdiğimi, ayrıca bu çalışmanın Dokuz Eylül Üniversitesi tarafından kullanılan bilimsel intihal tespit programıyla tarandığını ve intihal içermediğini beyan ederim. Herhangi bir zamanda aksinin ortaya çıkması durumunda her türlü yasal sonuca razı olduğumu bildiririm.

23.05.2019

Ümit KARADUMAN



DOKUZ EYLÜL ÜNİVERSİTESİ EĞİTİM BİLİMLERİ ENSTİTÜSÜ
YÜKSEK LİSANS/DOKTORA TEZ ÇALIŞMASI ORJİNALLİK RAPORU



Tarih: 18/06/2019

Tez Başlığı:

Code.org Platformunun 6. Sınıf Öğrencilerinin Programlama Öğrenimine Etkisi

Yukarıda başlığı belirtilen tez çalışmamın a) Kapak sayfası, b) Giriş, c) Ana bölümler ve d) Sonuç kısımlarından oluşan toplam 132 sayfalık kısmına ilişkin, 18/04/2019 tarihinde tez danışmanım tarafından Dokuz Eylül Üniversitesi Kütüphane ve Dokümantasyon Daire Başkanlığı'nın sağladığı İntihal Tespit Programından (Turnitin-Tez İntihal Analiz Programı) aşağıda belirtilen **filtreleme tiplerinden biri** (uygun olanı işaretleyiniz) uygulanarak alınmış olan orijinallik raporuna göre, tezin **benzerlik oranı % 13'tür**.

- <http://www.kutuphane.deu.edu.tr/turnitin-tez-intihal-analiz-programi/> adresindeki Tez İntihal Analiz Programı Kullanım Kılavuzunu okudum ☒

Filtreleme Tipi 1(Maksimum %15) ☒

Filtreleme Tipi 2(Maksimum %30) ☐

- ☐ Kabul/Onay ve Bildirim sayfaları hariç,
- ☐ Kaynakça hariç,
- ☐ Alıntılar dâhil,
- ☐ Altı (6) kelimeden daha az örtüşme içeren metin kısımları hariç.

- ☐ Kabul/Onay ve Bildirim sayfaları hariç,
- ☐ Kaynakça dâhil,
- ☐ Alıntılar dâhil.

EK 1- İntihal Tespit Programı Raporu İLK SAYFA Çıktısı. ☒

EK 2- İntihal Tespit Programı Raporu (Tümü) Cd İçinde. ☒

Dokuz Eylül Üniversitesi Eğitim Bilimleri Enstitüsü Tez Çalışması Orijinallik Raporu Uygulama Esaslarını inceledim ve yukarıda belirtilen azami benzerlik oranlarına göre tez çalışmamın herhangi bir intihal içermediğini; aksinin tespit edileceği muhtemel durumda doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi ve yukarıda vermiş olduğum bilgilerin doğru olduğunu beyan ederim.

Gereğini saygılarımla arz ederim.

Adı Soyadı : Ümit KARADUMAN
Öğrenci No : 2013950260
Anabilim Dalı : Bilgisayar ve Öğretim Teknolojileri Eğitimi
Programı : Bilgisayar ve Öğretim Teknolojileri Öğretmenliği
Statüsü : Yüksek Lisans ☒ Doktora ☐

ÖĞRENCİ

18/06/2019
Ümit KARADUMAN

DANIŞMAN

18/06/2019
Prof. Dr. Ercan AKPINAR

Açıklamalar

- 1: Bu formu teslim etmeden önce sizden istenen bilgileri uygun kutucuğu (☐) işaretleyerek doldurunuz.
Kullanıcı şifre vb. konusunda sorun yaşanması durumunda Üniversitemiz Merkez Kütüphanesinde bulunan Turnitin yetkilisine (Ali Taş Tel: +90 (232) 3018026 veya ali.tas@deu.edu.tr) başvurunuz.
- 2: Yüksek Lisans/Doktora Tez Çalışması Orijinallik Raporu" formu tezin ciltlenmiş ve elektronik nüshalarının içerisinde ekler kısmında yer alır.
- 3: Tez savunmasında düzeltme alınması durumunda bu form güncellenerek yeniden hazırlanır.
- 4: Turnitin-Tez İntihal Analiz Programına yükleme yapılırken Dosya Başlığı (document title) olarak **tez başlığının tamamı**, Yazar Adı (author's first name) olarak **öğrencinin adı**, Yazar Soyadı (author's last name) olarak **öğrencinin soyadı** bilgisini yazınız.

YÜKSEK LİSANS TEZİ SINAV SONUÇ FORMU

Ümit KARADUMAN tarafından Prof. Dr. Ercan AKPINAR yönetiminde hazırlanan Code.org Platformunun 6. Sınıf Öğrencilerinin Programlama Öğrenimine Etkisi başlıklı tez tarafımızdan okunmuş, kapsamı ve niteliği açısından “Yüksek Lisans Tezi” olarak kabul edilmiştir.


Prof. Dr. Ercan AKPINAR
Danışman


Doç. Dr. Bahar BARAN
Jüri Üyesi


Dr. Öğr. Üyesi Fırat SARSAR
Jüri Üyesi

23/5/2019


Prof. Dr. Süha YILMAZ
Enstitü Müdürü V.

TEŞEKKÜR

İnsanların birbiriyle anlaşabilmeleri için sağlıklı bir iletişim kurmaları gerekmektedir. Benzer şekilde bu durum bilgisayarla insanlar arasında da geçerlidir. Bilgisayarın dilini öğrenerek iletişime geçebilen bireyler, onları kendi amaç ve düşünceleri için kullanabilirler. Bilgisayarları yönetebilmek söz konusu olduğunda ise erken yaşta programlama eğitimi akıllara gelmektedir. Çünkü programlama eğitimi ile çocuklar; zihinsel gelişim, problem çözme becerisi, işlerin planlanması, durumlar arası ilişki kurma ve yeni fikirler üretme konusunda daha çok gelişim göstermektedirler. Çalışmamda da programlama öğrenimine yönelik araştırmalar dile getirilmektedir.

Yüksek lisans eğitimim ve araştırmalarım süresince yol gösteren ve yardımlarını esirgemeyen danışmanım Prof. Dr. Ercan Akpınar'a, Doç. Dr. Bahar Baran'a ve Arş. Gör. Yasemin Kahyaoğlu Erdoğan'a teşekkürlerimi iletirim.

Hayatım boyunca yanımda duran, maddi ve manevi desteğini hiçbir zaman eksik etmeyen, üzerimde büyük emekleri olan değerli aileme minnettar olduğumu ifade etmek isterim.

Çalışmalarım süresince desteklerini esirgemeyen, daima motive eden, varlıklarıyla yaşamıma anlam katan kıymetli eşime ve canım kızıma sevgimi ve sonsuz teşekkürlerimi sunarım.



İÇİNDEKİLER

ÖZET	xii
ABSTRACT.....	xiv
BÖLÜM I.....	1
GİRİŞ	1
1.1. Problem Durumu.....	1
1.2. Araştırmanın Amacı ve Önemi	4
1.3. Problem Cümlesi.....	6
1.3.1. Alt problemler	7
1.4. Sayıtlar	7
1.5. Sınırlılıklar	7
1.6. Tanımlar	8
1.7. Kısaltmalar	8
BÖLÜM II	10
İLGİLİ YAYIN VE ARAŞTIRMALAR.....	10
2.1. BTY Dersi	10
2.2. Programlama	15
2.3. Programlama Öğretimi.....	18
2.4. Code.org Platformu	21
2.5. Görsel Programlama Ortamları ve İlgili Çalışmalar	23
2.6. Eğitimde Oyunun Yeri ve İlgili Çalışmalar	31
BÖLÜM III.....	34
YÖNTEM	34
3.1. Araştırma Modeli	34
3.1.1. Değişkenler ve deneysel desen	35
3.2. Araştırma Grubu	36

3.3. Veri Toplama Araçları ve Verilerin Toplanması	39
3.3.1. EBODKÖYTÖ.....	39
3.3.1.1. EBODKÖYTÖ verilerinin toplanması.....	40
3.3.2. Programlama öğrenim düzeyi belirleme aracı (PÖDA).....	40
3.3.2.1. PÖDA verilerinin toplanması.....	41
3.3.3. Code.org platformuna yönelik yarı yapılandırılmış görüşme formu (CPGF).....	42
3.3.3.1. CPGF verilerinin toplanması	42
3.4. Uygulama Süreci.....	43
3.4.1. Code.org platformu kullanımı ve yapılan ortak işlemler	43
3.4.2. DG ve KG uygulama süreci	50
3.5. Verilerin Analizi	59
3.5.1. EBODKÖYTÖ verilerinin analizi	59
3.5.2. PÖDA verilerinin analizi	61
3.5.3. CPGF verilerinin analizi	61
BÖLÜM IV	64
BULGU VE YORUMLAR.....	64
4.1. Code.org Platformunun 6. Sınıf Öğrencilerinin Programlama Öğrenim Düzeylerine Etkisi	64
4.2. Code.org Platformunun 6. Sınıf Öğrencilerinin Programlama Öğrenimlerinin Kalıcılık Düzeylerine Etkisi.....	65
4.3. Code.org Platformunun Deney ve Kontrol Grubunda Yer Alan Öğrencilerin EBODKÖY Tutumlarına Etkisi	66
4.4. Deney Gurubu Öğrencilerinin Code.org Platformuna Yönelik Görüşleri	68
BÖLÜM V.....	75
SONUÇ, TARTIŞMA VE ÖNERİLER	75
5.1. Sonuç ve Tartışma.....	75

5.1.1. Code.org platformunun 6. sınıf öğrencilerinin programlama öğrenim düzeylerine etkisi	75
5.1.2. Code.org platformunun 6. sınıf öğrencilerinin programlama öğrenimlerinin kalıcılık düzeylerine etkisi	76
5.1.3. Code.org platformunun deney ve kontrol grubunda yer alan öğrencilerin EBODKÖY tutumlarına etkisi	77
5.1.4. Deney grubu öğrencilerinin Code.org platformuna yönelik görüşleri	78
5.2. Öneriler	80
5.2.1. Uygulayıcılara yönelik öneriler	80
5.2.2. Araştırmacılara yönelik öneriler	80
5.2.3. İlgili kurumlara yönelik öneriler	81
KAYNAKÇA	82
EKLER	94
Ek 1. Araştırmacının Akademik Özgeçmişi	94
Ek 2. Etik Kurul İzin Belgesi	95
Ek 3. EBODKÖYTÖ	96
Ek 4. EBODKÖYTÖ Kullanım İzni	97
Ek 5. PÖDA	98
Ek 6. PÖDA Uzman Görüşü Formu	101
Ek 7. CPGF	102
Ek 8. Code.org Platformu Kullanım İzni	103
Ek 9. BTY Dersi 5. Sınıf “Problem Çözme ve Programlama Alanı” Kazanımları.....	104
Ek 10. BTY Dersi 6. Sınıf “Problem Çözme ve Programlama” Alanı Kazanımları.....	106
Ek 11. Örnek Günlük Ders Planları	108
Ek 12. Deney Grubuna Ait Görseller.....	110
Ek 13. Kontrol Grubuna Ait Görseller.....	112
Ek 14. Ölçeklerin Uygulanmasına Yönelik Görseller.....	114

TABLOLAR LİSTESİ

Tablo 1. Deneyisel işlem düzeni	35
Tablo 2. Öğrencilerin cinsiyetlerine göre dağılımları.....	36
Tablo 3. Öğrencilerin programlamaya ilişkin kavramların ne ifade ettiğine ilişkin betimsel istatistik sonuçları	37
Tablo 4. EBODKÖYTÖ ön test uygulama sonuçlarının gruplara göre U testi sonuçları	38
Tablo 5. PÖDA ön test uygulama sonuçlarının gruplara göre U testi sonuçları	39
Tablo 6. Deney grubu görüşme süreleri.....	43
Tablo 7. CPGF madde 1 (Örnek)	62
Tablo 8. PÖDA son test uygulama sonuçlarının gruplara göre U testi sonuçları.....	64
Tablo 9. KG ön test ve son test PÖDA sonuçlarının karşılaştırılması	65
Tablo 10. DG ön test ve son test PÖDA sonuçlarının karşılaştırılması	65
Tablo 11. KG ve DG son test-izleme testi PÖDA verileri.....	66
Tablo 12. DG ve KG izleme testi PÖDA sonuçlarının karşılaştırılması	66
Tablo 13. EBODKÖYTÖ son test uygulama sonuçlarının gruplara göre U testi sonuçları..	67
Tablo 14. KG ön test ve son test EBODKÖYTÖ sonuçlarının karşılaştırılması	67
Tablo 15. DG ön test ve son test EBODKÖYTÖ sonuçlarının karşılaştırılması	68
Tablo 16. CPGF madde 1	68
Tablo 17. CPGF madde 2.....	69
Tablo 18. CPGF madde 3.....	70
Tablo 19. CPGF madde 4.....	70
Tablo 20. CPGF madde 5.....	71
Tablo 21. CPGF madde 6.....	72
Tablo 22. CPGF madde 7	72
Tablo 23. CPGF madde 8.....	73
Tablo 24. CPGF madde 9.....	73
Tablo 25. CPGF madde 10.....	74
Tablo 26. CPGF ek madde	74

ŞEKİLLER LİSTESİ

Şekil 1. Tanımlar	8
Şekil 2. İlgili standartlar ve öğretim programları	11
Şekil 3. Bilgi ve iletişim teknolojileri yeterlikleri	11
Şekil 4. Teknoloji taksonomisi	12
Şekil 5. BTY dersi öğrenme alanları	13
Şekil 6. Program için belirlenen düzeyler.....	13
Şekil 7. Problem çözme ve programlama ünitesi.....	14
Şekil 8. BTY dersi öğretim programının özel amaçları	15
Şekil 9. Algoritma örneği	17
Şekil 10. Akış şeması örneği.....	17
Şekil 11. Code.org ekran görüntüsü.....	21
Şekil 12. Code.org öğrenci takibi ekran görüntüsü.....	22
Şekil 13. Code.org destekçilerinden bazıları	22
Şekil 14. Code.org örnek ekran görüntüleri.....	23
Şekil 15. Scratch ekran görüntüsü	25
Şekil 16. Alice ekran görüntüsü.....	26
Şekil 17. Kodu Game Lab ekran görüntüsü.....	26
Şekil 18. Hacker Can ekran görüntüsü	27
Şekil 19. mBlock ekran görüntüsü.....	27
Şekil 20. CodeCombat ekran görüntüsü	28
Şekil 21. Blockly ekran görüntüsü.....	28
Şekil 22. App Inventor ekran görüntüsü.....	29
Şekil 23. Cinsiyetlere göre dağılım grafiği.....	37
Şekil 24. Programlamaya ilişkin kavramların ne ifade ettiğine dair grafik	38
Şekil 25. PÖDA ölçme aracının ilgili olduğu kazanımlar	41
Şekil 26. Sanal sınıf öğrenci listesi.....	44
Şekil 27. Code.org platformu öğrenci girişi bilgileri.....	45

Şekil 28. Öğrenci giriş ekranı	45
Şekil 29. Bölüm ipucu video ekranı.....	46
Şekil 30. Video notları.....	46
Şekil 31. Bölüm ekranı	46
Şekil 32. Bölüm geçiş ekranı	47
Şekil 33. Kod görüntüleme ekranı	47
Şekil 34. Öğrencinin bölümler arası geçiş ekranı	47
Şekil 35. Öğrencinin genel durum takip ekranı	48
Şekil 36. Grup ayarları ekranı.....	48
Şekil 37. Öğrenci sanal sınıf izleme ekranı	49
Şekil 38. Öğrenci kodlama istatistik ekranı	49
Şekil 39. Öğrenci uygulama takip ekranı.....	50
Şekil 40. Eğitim sonrası verilen sertifika örneği.....	50
Şekil 41. DG-KG algoritma örnekleri.....	51
Şekil 42. Akış şeması sembolleri.....	51
Şekil 43. DG örnek akış şeması	52
Şekil 44. KG örnek akış şeması	52
Şekil 45. DG “Döngü” kavramına ilişkin bir örnek.....	53
Şekil 46. KG “Döngü” kavramına ilişkin bir örnek.....	53
Şekil 47. DG “Koşul” kavramına ilişkin bir örnek	54
Şekil 48. KG “Koşul” kavramına ilişkin bir örnek	55
Şekil 49. DG “Değişken” kavramı.....	55
Şekil 50. DG “Değişken” kavramına ilişkin bir örnek	55
Şekil 51. KG “Değişken” kavramı.....	56
Şekil 52. KG “Değişken” kavramına ilişkin bir örnek	56
Şekil 53. DG “Operatörler” kavramına ilişkin bir örnek	57
Şekil 54. KG “Operatörler” kavramına ilişkin bir örnek	57
Şekil 55. DG öğrenilenlere ilişkin genel bir örnek	58
Şekil 56. KG öğrenilenlere ilişkin genel bir örnek	59

ÖZET

Code.org Platformunun 6. Sınıf Öğrencilerinin Programlama Öğrenimine Etkisi

KARADUMAN, Ümit

Yüksek Lisans Tezi, Bilgisayar ve Öğretim Teknolojileri Eğitimi Anabilim Dalı

Danışman: Prof. Dr. Ercan AKPINAR

Öğrencilere hayatın içerisinde karşılaşılan problemleri çözmenin bir amaç olmadığı, asıl önemli olanın yaşanılan süreç olduğu programlama öğretimi ile aktarılmaktadır. Öğrenilmesi zor bir süreç olarak belirtilen programlamanın da daha kolay ve etkili öğretilmesi amacıyla görsel programlama ortamlarından yararlanılmaktadır. Bu araştırmada da 6. sınıf öğrencilerinin programlama öğrenimlerine Code.org platformunun etkileri araştırılmıştır.

Araştırma Milli Eğitim Bakanlığına bağlı bir ortaokulda eğitim gören öğrenciler ile yürütülmüştür. Araştırmada veri toplama aracı olarak Keçeci, Alan ve Zengin (2016) tarafından geliştirilen Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeği, Code.org Platformuna Yönelik Yarı Yapılandırılmış Görüşme Formu ve Programlama Öğrenim Düzeyi Belirleme Aracı kullanılmıştır. Deneysel araştırma desenine uygun olarak gerçekleştirilen araştırmada gerçek deneme modellerinden 2 X 3'lük (deney - kontrol X ön test - son test - izleme testi) model kullanılmıştır. Uygulamanın başlangıcında veri toplama araçları iki gruba da uygulanmış, sonrasında ise öğrenciler normal yöntemler ve Code.org platformu destekli yöntemler olmak üzere iki farklı şekilde programlama öğrenimine katılmışlardır. Deneysel işlemin bitiminden hemen sonra gruplardan son test ölçümleri alınmış ve deney grubuna görüşme formu uygulanmıştır. Deneysel işlemin tamamlanmasından beş hafta sonra da programlama öğreniminin kalıcılığına yönelik izleme testi ölçümleri alınmıştır. Görüşme formunun sonuçları içerik analizi ile betimsel olarak

yorumlanmıştır. Diğer ölçme işlemlerinin sonucunda elde edilen veriler Mann Whitney U testi, Wilcoxon İşaretli Sıralar testi, kovaryans analizi (ANCOVA) ve Bonferroni ikili karşılaştırmalar testi ile SPSS istatistik paket programı kullanılarak değerlendirilmiştir.

Araştırma sonucunda Code.org platformunun, öğrencilerin programlama öğrenimlerine ve programlama öğrenimlerinin kalıcılıklarına normal öğretim yöntemlerine göre daha olumlu etkileri olduğu saptanmıştır. Ayrıca Code.org platformu, öğrencilerin eğitsel bilgisayar oyunları destekli kodlama öğrenimine karşı olan tutumlarını olumlu yönde artırmıştır. Yürütülen yarı yapılandırılmış görüşmeler sonucunda da bu platformun öğrenci başarısı üzerinde daha pozitif bir etkisi olduğu, öğrencilerin öğrenme sürecinden daha fazla keyif aldıkları ve Bilişim Teknolojileri ve Yazılım dersine karşı daha olumlu tutumlar geliştirdikleri ortaya konulmuştur. Çalışmanın süreci ve sonuçları göz önüne alındığında; Code.org platformunun öğrenme ortamlarına entegre edilmesi ve bu alandaki deneysel çalışmaların kapsam olarak genişletilmesine yönelik yeni araştırmalar yapılması önerilmektedir.

Anahtar Sözcükler: Code.org, Görsel Programlama, Programlama Öğrenimi.

ABSTRACT

The Effect of Code.org Platform on 6th Grade Students' Learning Programming

KARADUMAN, Ümit

Master's Thesis, Department of Computer and Instructional Technology Education

Thesis Advisor: Professor Doctor Ercan AKPINAR

It is transferred to the students that the aim is not solving the problems encountered in life but the important one is the process lived by the programming teaching. Visual programming environments are benefited for the purpose of easier and more effective teaching of programming which is described as a difficult learning process. In this study, the effects of Code.org platform on the programming learning of 6th grade students were investigated.

The research was carried out with students studying in a secondary school affiliated to the Ministry of National Education. In this study, Attitude Scale for Educational Computer Games Supported Coding Learning developed by Keçeci, Alan and Zengin (2016), Semi-Structured Interview Form for Code.org Platform and Programming Learning Leveling Tool are used as data collecting tools. In the research carried out in accordance with the experimental research design, 2 X 3 (experimental - control X pre-test - final-test - re-test) model from the real trial models was used. At the beginning of the application, data collecting tools were applied to two groups, and then students participated in programming in two different ways as usual methods and Code.org platform supported methods. After the end of the experimental process, the final-test measurements were taken from the groups and the interview form was applied to the experimental group. Five weeks after the completion of the experimental procedure, re-test measurements were taken for the permanence of programming learning. The results of the interview form were interpreted descriptively with the content analysis. The data obtained from the other measurement processes were

evaluated by using Mann Whitney U test, Wilcoxon Signed Ranks test, covariance analysis (ANCOVA) and Bonferroni paired comparisons test and SPSS statistical package program.

As a result of the research, it was found that Code.org platform had more positive effects on students' programming learning and retention of programming learning than usual teaching methods. In addition, the Code.org platform positively increased the students' attitudes towards educational computer games supported coding learning. As a result of the conducted semi-structured interviews, it was revealed that this platform had a more positive effect on students' achievement; students enjoyed more of the learning process and developed more positive attitudes towards the Information Technology and Software course. Considering the process and results of the study; it is recommended to integrate the Code.org platform into learning environments and to be done new research to expand the experimental studies in this area.

Keywords: Code.org, Visual Programming, Programming Learning.

BÖLÜM I

GİRİŞ

Bu bölümde sırasıyla, araştırmanın gerekçelerini ortaya koyan problem durumu, araştırmanın amacı ve önemi, problem cümlesi, alt problemleri, sayıtları, sınırlılıkları, tanımları ve kısaltmaları yer almaktadır.

1.1. Problem Durumu

21. yüzyılda çocukların sahip olması gereken en temel beceriler; yaratıcılık, eleştirel düşünme, iletişim ve işbirliği olarak ifade edilmektedir. Teknolojinin sürekli ilerlemesi, eğitim sürecine kazandırdığı yenilikler ve beraberinde getirdiği kolay ulaşılabilirlik, öğrencilerin bu becerileri edinmesini oldukça kolaylaştırmaktadır. 21. yüzyıl becerilerinden birisi olarak görülen ve her bireyin sahip olması gereken “Bilgi İşlemsel Düşünme (Computational Thinking)” kavramı ise ‘bilgisayar bilimini kullanarak problem çözme becerisi’ şeklinde tanımlanmaktadır (ISTE, 2018). Bu kavrama göre algoritmik düşünme ile problemler küçük parçalara ayrılır, çözüm üretilerek sonuç gözlemlenir ve en başta ön görülen sonuca göre doğru adımlar atılmaya devam edilir. Sonuç olarak ise bu becerilere sahip olan kişiler derslerindeki ve günlük yaşantılarındaki problemlerini de kolaylıkla çözebileceklerdir (Küçüköğlu, 2014). Yaşantımızda önemli yeri olan problem çözme süreci ise programlama eğitiminin temel parçası konumundadır.

Programlama eğitiminin, öğrencilerin problem çözme, yaratıcılık, algoritmik ve sistematik düşünme gibi bir takım bilgi işlemsel düşünme becerilerini kazanmalarına etkisi olduğu farklı dönemlerde yapılan araştırmalarca belirtilerek bireyler ve kazanımlar açısından önemi her dönemde dile getirilmiştir (Ozoran, Çağıltay ve Topallı 2012; Çankaya, Durak, Yünkül 2017; Özcan, Ergün, Köse, Emir ve Gezgin 2017; Oluk, Korkmaz ve Oluk 2018; Dizman 2018; Erdem, 2018; Vatansever 2018). Ancak programlama eğitiminin, amaçlanan kazanımların üzerinde önemli etkileri olduğu sürekli vurgulanırken, gerçekleştirilen programlama eğitiminin şekli, hedef kitleye ilişkin kullanılan yöntem doğrultusunda bazı sorunlar oluşturabilmektedir (Çatlak, Tekdal ve Baz, 2015). Çünkü normal metotların yer aldığı eğitim-öğretim sürecinde işin tamamlanması, geçirilen sürecin kalitesinden daha önemli görülmektedir. Öğrenmenin odak olarak görüldüğü bir ortamda ise yaşanan süreç, gözlem yapabilmek için çok önemlidir. Öğrenci bireysel öğrenme hedefleri olan, bilgi ve becerilerini geliştirmeye odaklanan ve öz değerlendirme yapabilen öğrenen konumundadır.

Bu öğrenme ortamlarında öğrencilerin üst-bilişlerini etkinleştirerek eleştirel ve yaratıcı düşünen, sebep-sonuç bağlantısı kurabilen, bilgi okuryazarı olan ve hayat boyu öğrenmeyi benimseyen bireyler haline gelebilmeleri için farklı çözümler aranmaktadır (Küçüköğlu, 2014).

Cevahir ve Özdemir (2017), programlama öğreniminde, soyut düşünme becerisindeki eksiklik, ezbere dayalı öğretim, tekrar ve uygulama eksikliği gibi zorluklar yaşandığını dile getirmişlerdir. Dizman (2018) ve Şimşek (2018) de çalışmalarında; metin tabanlı kodlama yapısında öğrencilerin zorluklar yaşadığını ve programlamaya karşı önyargı oluşturduklarını belirtmişlerdir. Özmen ve Altun (2014) ise çalışmalarında; pratik eksikliği, algoritma oluşturamama ve bilgi eksikliği gibi hususları programlama eğitiminde yaşanan zorluklar olarak ortaya koymuşlardır. Programlamanın öğrenciler tarafından zor ve karmaşık olarak görülmesi, onların programlamaya ilişkin tutumlarını da olumsuz olarak etkileyebilmektedir. Ayrıca bu durum, öğrencilerde başarısızlık hissi de oluşturabilmektedir. Bu olumsuzlukların üstesinden gelmek için programlama eğitimini kolaylaştırabilmek, yaşanabilecek kaygı ve stresi azaltabilmek, derse katılımı arttırabilmek için, büyük firmaların geliştirdiği blok tabanlı görsel programlama dilleri kullanılabilir (Demir, 2015; Ozoran ve diğerleri, 2012; Genç ve Karagüç, 2011). Aytekin, Çakır, Yücel ve Kulaöz (2018); araştırılan açık kaynaklar ve alanyazın irdelendiğinde birçok temel yetinin kazanılması için kodlama derslerinde görsel programlama kaynaklardan yararlanılabileceğini, bu sayede geleneksel programlama dillerinin öğrenilmesindeki zorlukların aşılabileceğini ve mevcut oyunlaştırmanın öğrencilerin motivasyonlarına, yeni fikirler ve projeler geliştirmelerine katkı sağlayabileceğini ifade etmişlerdir. Buna paralel olarak Code.org, Alice, Kodu ve Scratch gibi programlama öğrenmeyi kolaylaştıran araçların seçilmesi ve bu sayede öğrenme ortamının eğlenceli hale getirilmesi, öğrencilerin tutumlarını da olumlu yönde etkileyebilir. Nitekim Yıldız (2018) ve Patan (2016) da gerçekleştirdiği çalışmalarında; Code.org platformu kullanımının öğrencilerin tutumunu olumlu yönde etkilediği sonucuna ulaşmışlardır.

Code.org, Alice, Kodu ve Scratch gibi görsel programlama dilleri ile kodlamanın öğretim amaçlı olarak kullanımı yaygınlaşmıştır. Bu ortamlar, erken yaştaki öğrencilerin geleneksel programlama dillerinin kompleks kod yapılarını öğrenme şartını ortadan kaldırarak, uygulamalar oluşturabilmelerini sağlamaktadır (Resnick ve diğerleri, 2009). Ayrıca görsel programlama ortamları çocuklara yönelik tasarlandıkları için onların gelişim seviyelerine uygun durumdadır (Fessakis, Gouli ve Mavroudi, 2013). Bir öğrenme

materyalinin, gerçeklik, açık uçlu tasarım görevlerindeki zorlukların dengesi, zengin ve çeşitli geri bildirimler, tartışma ve işbirliği, çoklu bağlamlar, deneyler ve araştırmalar gibi öğelere sahip olması beklenmektedir; öğrenciye verilecek tasarım görevlerinin de eğitsel yazılımlar, eğitsel oyunlar, web siteleri veya PowerPoint sunuları şeklinde olması etkililiği artırabilmektedir (Erekmekçi ve Fidan, 2012). Erasmus Von Rotterdam *“Öğretim işi, çocuklar için zevkli bir şekle sokulmalıdır. Bu nedenle öğretim, oyun ile birleştirilmelidir.”* sözü ile çocuklara verilen eğitimin oyunlaştırılmasının önemine vurgu yapmıştır (Bozan, 2014). Ayrıca eğitim ile oyun iç içe bulundurularak hem yeni kavramlar öğrenciye öğretilir, hem de bu kavramların kalıcılığı sağlanabilir (Kılınçpınar, 2014). Nitekim Savaş (2014), Yıldırım (2015) ve Koka (2018) da gerçekleştirdiği çalışmalarında; oyun öğesinin öğrenci başarısı ve kalıcılığı üzerinde olumlu etkisi olduğunu ifade etmişlerdir. Ayrıca Savaş (2014), öğretim esnasında kullanılan eğitsel oyunlar sayesinde öğrencilerin oynayarak öğrenebildiklerini, dersi ve ilgili öğretmenlerini daha çok sevdiklerini dile getirmiştir.

Görsel programlama ortamları öğrencilerin kendi etkileşimli oyunlarını, animasyonlarını, benzetim uygulamalarını ve hikâyelerini yaratmaya olanak tanıyan görsel programlama araçlarıdır ve öğrencilere bazı matematiksel düşüncelerini animasyon haline dönüştürerek yeni şeyler oluşturmalarını mümkün kılan sanal ortamlar sağlamaktadırlar (Taylor, Harlow ve Forret, 2010). Bu öğrenme ortamlardaki başlıca hedef farklı becerileri geliştirerek öğrenme çıktıları ve öğrencilerin motivasyonunu artırmaktır (Sayın ve Seferoğlu, 2016). Bu çalışmada oyunlaştırılmış görsel programlama araçları içerisinde Code.org platformu seçilmiş ve çalışmada etkinliği incelenmiştir. Code.org platformunun, ticari kaygı taşıması, her yerde her zaman öğrenme olanağı tanınması, eğlenceli ve etkileşimli yapısı, öğretmen-öğrenci kullanımı ile okul yapısına uyması gibi özellikleri bu seçimi belirleyen nedenlerden bazılarıdır. Ayrıca örgün eğitime devam eden genç ve çocukların eğlenerek öğrenmeleri için geçmişte popüler olmuş uygulamaların kullanılmasının faydalı olabileceği (Taşdemir ve Şüyun, 2016) düşünüldüğünde; Angry Birds, Flappy, Minecraft gibi oyunları içerisinde barındıran Code.org platformu öğrenciler tarafından ilgi çekici görülmektedir.

Yapılan alanyazın taraması ve analizler, kodlama eğitimine yönelik gerçekleştirilen akademik çalışmaların çok az sayıda olduğunu gösterirken birçok ülkenin eğitim müfredatlarına kodlama eğitimini dâhil ettiğini veya buna yönelik hazırlıklar yaptığını da ifade etmektedir (Sayın ve Seferoğlu, 2016). Programlama dilleri öğretimi, çeşitli ülkeler

tarafından (İngiltere, ABD, Güney Kore, Fransa, Avusturya, Bulgaristan, İspanya, Çek Cumhuriyeti, Tayvan, Danimarka, Finlandiya, Macaristan, İrlanda, Avustralya, Litvanya, Malta, İsrail, Polonya, Estonya, Portekiz, Slovakya) öğretim programlarına dâhil edilerek zorunlu statüde okutulan bir ders konumundadır (Demirer ve Sak, 2016; Sayın ve Seferoğlu, 2016). Kodlama eğitimi sadece okullarda ders olarak okutulmanın dışında “Hour of Code (Kod Saati)”, “Europe CodeWeek (Avrupa Kod Haftası)” gibi çeşitli etkinlik ve organizasyonlarla dünya genelinde tanıtılarak yaygınlaştırılmaya ve sevdirmeye çalışılmaktadır. Ayrıca Code.org gibi kâr amacı gütmeyen kuruluşlarla, büyük olarak addedilen firmaların işbirliği yaptığı bilinmekte ve bu çalışmaların geleceğin yetişmiş insan gücünü oluşturabileceği düşüncesiyle politika yapıcılar ve uygulayıcılar tarafından da desteklendiği anlaşılmaktadır. Bahsedilen bu faktörler de araştırma konusunun seçiminde etkili olan unsurlardandır.

1.2. Araştırmanın Amacı ve Önemi

Teknolojinin her yönden hayatımıza dâhil olması ve kapladığı yerin her geçen gün artması hızlı bilgi artışını, dolayısıyla toplumların da hızlı değişimini ve gelişimini beraberinde getirmektedir. Bu değişime ayak uydurmak ise ancak yeterli eğitimle mümkün olabilmektedir. Günümüzde okuma-yazma bilen, aritmetik bilgileri olan kişilere ithafen kullanılan eğitilmiş insan ifadesi yaşanan hızlı değişimlerden etkilenmiştir. Eğitimin hedeflediği temel becerilere; dijital kültürün ortak dili olan, öğrencilerin kendi geleceklerini planlamaları imkânını sunan kodlama becerisi de eklenmiştir (EBA, 2019). Türk Eğitim sisteminin temel amacı olarak Onuncu Kalkınma Planında “düşünebilme, algılayabilme, sorun çözebilme yeteneği gelişmiş, bilimi ve teknolojiyi kullanabilen ve teknoloji üretimine yatkın, bilgi toplumları için gerekli temel bilgi ve becerilere sahip, üretebilen bireyler” ifadeleri yer almaktadır (Onuncu Kalkınma Planı, 2014). İçinde bulunduğumuz bilgi çağında kodlama eğitimi, bu hedef doğrultusunda önemli bir araçtır (Aytekin ve diğerleri, 2018).

Milli Eğitim Bakanlığı (MEB) öğretim programlarında ve birçok strateji metninde bilgisayar okuryazarlığı kavramı sıklıkla ifade edilmektedir. Klasikleşen bu kavramı incelediğimizde ise genellikle bilgisayarın okuma olarak ele alındığı, yazma işinin ise göz ardı edildiği veya basit olarak kelime işlemlerde metin oluşturma şeklinde algılandığı anlaşılmaktadır. Ancak çağımızın değişen şartları yazma boyutunun yeniden yorumlanması gerekliliğini doğurmaktadır. Çünkü programlama eğitiminde olduğu gibi bilgisayar okuryazarlığı da, sayısal okuryazarlığın en önemli parçalarından birisi olarak uzun yıllardır birçok ülkenin öğretim programına entegre edilmiş ve hali hazırda uygulanmaktadır. Erken

yaştan itibaren bu konuda öğrencilere imkân sunulması çok önemli hale gelmiştir. Dolayısıyla temel eğitim seviyesinde bu dersin muhtevasının ve gerekli öğrenme araçlarının hazırlanması önemli bir eksikliği giderecek, programlama öğretimi için de sağlam bir destek ayağı sağlanmış olacaktır (Akpınar ve Altun, 2014).

Bilişim teknolojileri alanında çok önemli konuma sahip Mark Zuckerberg, Bill Gates, Jack Dorsey gibi isimlerin genelinin ortak özelliği erken yaşta programlama ile tanışmış olmalarıdır. Programlama öğrenmek bireylere; yaratıcı düşünebilme, sorun çözebilme, sistematik düşünme, olaylar arasındaki bağlantıları görebilme gibi 21.yy becerileri kazandırma ve bilişim okuryazarı olma konusunda çok büyük avantajlar sağlamaktadır (Demirer ve Sak, 2016). Endüstrileşen ülkeler, yeni sanayi devrimi konumundaki Endüstri 4.0'a daha hazır ve etkin bir şekilde geçmek için programlama öğrenmenin gerekliliği görüşünü benimsemiş (Aytekin ve diğerleri, 2018) ve bu alana yönelik zorunlu derslere öğretim programlarında yer vermiştir (Demirer ve Sak, 2016; Sayın ve Seferoğlu, 2016). Gerekli olmasının yanı sıra öğreniminin de zor bir süreç olduğu belirtilen programlamanın daha iyi öğretilmesi amacıyla görsel programlama uygulamaların kullanılması gerekliliği ifade edilmiş ancak ülkemizde yeterli seviyede kullanılmadığı fark edilmiştir (Solmaz, 2014).

Çatlak ve arkadaşlarına göre (2015), programlama ile yeni tanışan kullanıcılara yönelik, geleneksel programlama dillerinin yapılarındaki karmaşıklık sorununu çözmek için, öğrenilmesi daha kolay, görselliği ve etkileşimi ön plana alan programlama öğrenme ortamları geliştirilmiştir. Öğrencilere basit bir kullanım sunan yap-boz şeklindeki bloklardan oluşan görsel programlama araçları, problem basamakları arasındaki ilişkilerin fark edilmesini, yapılan çalışmanın test edilebilmesini, hataların minimum düzeye indirgenebilmesini sağlamaktadır (Baldwin ve Kuljis, 2000). Öğrencilere senaryo tasarlayabilecekleri bir imkân sunup, özgür bir ortam sağlayıp, animasyonlarla bu sürecin somutlaştırılmasını (Ramadhan, 2000), böylece daha ilgi çekici bir ortama dönüşmesini (Utting, Cooper, Kölling, Maloney ve Resnick, 2010) sağlayan bu araçlar, öğrencileri kodları ve kuralları ezberlemekten, söz dizimi yanlışlarından ve kod satırlarına boğulmaktan kurtarmıştır. Görsel programlama araçlarından olan Code.org platformu da bu açıdan önemli yer tutmaktadır.

Erken yaşta alınan programlama eğitiminin geleceği yakalamanın anahtarı olduğunu kavrayan programcılar bu eğitimi kolay ve etkili kılmak için Code.org platformu gibi çeşitli araçlar ve ortamlar geliştirdiler. Code.org platformu; Microsoft, Facebook, Twitter, Google

gibi alanlarında dünyanın en meşhur programcıları ve büyük teknoloji firmalarının kurucularının yer aldığı kişilerin destekleriyle programlama öğreniminin yaygınlaşması için başlatılan proje kapsamında kurulmuştur. Bu platform sahip olduğu yapı sebebiyle eğitim faaliyetleri için uygun durumdadır. Öğretmen ve öğrenci kullanımlarına müsait olması okul yapısıyla benzerlik taşımaktadır. Süreç içerisinde öğretmen öğrencileri sistem üzerinden takip edebilmekte, ilgili kursa ilişkin gerekli gördüğü müdahaleleri yapabilmektedir. Öğretmenin yanı sıra öğrenciler de kendi ilerlemelerini izleyebilmektedirler. Çalık ve Sezgin'e (2005) göre eğitim, yaşamımızın her alanında, her zaman ve her yerde gerçekleşmelidir. Sınıf içinde olduğu kadar internette de eğitim etkili şekilde verilebilmeli, evde eğitim olanakları sağlanmalıdır. Bu açıdan bakıldığında Code.org platformunun ders dışı kullanıma uyumlu olması onun etkili bir öğrenme aracı olarak kullanılabilme potansiyelini göstermektedir.

Alanyazındaki Code.org platformunun öğrencilerin programlama öğrenim düzeylerine ve tutumlarına etkisini içeren çalışmaların eksikliği bu araştırmanın önemini artırmakta, konu üzerinde çalışmak isteyen araştırmacılar için bu çalışmanın kaynak çeşitliği sağlayacağı düşünülmektedir. Ayrıca programlama konusunun öğretimine ilişkin; Code.org platformunun sahip olduğu özelliklerle birlikte Bilişim Teknolojileri ve Yazılım dersi programlama öğretimi konusunda eğitimcilere kılavuzluk ederek öğretime yardımcı olması beklenmektedir.

Genel olarak değerlendirildiğinde kodlama eğitimi, 21.yy yeterlikleriyle donatılmış, teknolojiyi çok daha etkin kullanabilen, problem çözme ve ürün geliştirme becerilerine sahip bireylerin eğitiminde önemli bir konumdadır. Kodlama öğretimine duyulan ihtiyaç ve Code.org platformunun özellikleri dikkate alındığında, kodlama öğretimine yönelik ilgili platformun çözüm getirebileceği değerlendirilerek *bu araştırmanın amacı*, Code.org platformunun ortaokul 6. sınıf öğrencilerinin programlamaya ilişkin tutumlarına ve programlama öğrenimlerine olan etkilerini araştırmak ve öğrencilerin bu platforma yönelik görüşlerini incelemek olarak belirlenmiştir.

1.3. Problem Cümlesi

Code.org platformu ile programlama öğreniminin 6. sınıf öğrencilerinin eğitsel bilgisayar oyunları destekli kodlama öğrenimlerine yönelik tutumlarına, programlama öğrenim düzeylerine, programlama öğrenimlerinin kalıcılık düzeylerine etkisi nedir ve deney grubu öğrencilerinin platforma yönelik görüşleri nelerdir?

1.3.1. Alt problemler

1. Code.org platformunun deney ve kontrol grubunda yer alan öğrencilerin programlama öğrenim düzeylerine etkisi nedir?
2. Code.org platformunun deney ve kontrol grubunda yer alan öğrencilerin programlama öğrenimlerinin kalıcılık düzeylerine etkisi nedir?
3. Code.org platformunun deney ve kontrol grubunda yer alan öğrencilerin eğitsel bilgisayar oyunları destekli kodlama öğrenimlerine yönelik tutumlarına etkisi nedir?
4. Deney gurubu öğrencilerinin Code.org platformuna yönelik görüşleri nelerdir?

1.4. Sayıtlılar

Deney ve kontrol gurupları arasında uygulamanın sonuçlarını etkileyecek bir etkileşimin olmadığı düşünülmektedir.

1.5. Sınırlılıklar

1. Bu araştırma, 2016-2017 eğitim-öğretim yılında Kütahya ili Gediz ilçesinde bulunan Altinkent Ortaokulu 6. sınıf öğrencileriyle sınırlıdır.
2. Bu çalışma, araştırma kapsamında kullanılan tutum ve başarı ölçekleri ile yarı yapılandırılmış görüşme formunun ölçtüğü niteliklerle sınırlıdır.

1.6. Tanımlar

<i>Bilgisayar Programlama</i>	Bilgisayar programlama; problemleri çözmek, insan - bilgisayar etkileşimini sağlamak ve belirli bir görevi bilgisayarlar tarafından gerçekleştirmek için çeşitli komut setleri ile yapılan uygulama ve geliştirme sürecidir. Bu komut setleri bilgisayar programları tarafından dikkate alınırlar ve bilgisayarların işlemleri kolayca yapabilmelerini sağlarlar (businessdictionary.com, 2019).
<i>Görsel Programlama</i>	Resim, grafik, blok gibi görsel öğeler kullanılarak yapılan programlama işlemine verilen isimdir.
<i>Komut Seti</i>	Bilgisayar programlama dilleri ile yazılan kaynak kodlara verilen addır (Sayın ve Seferoğlu, 2016).
<i>Makine Dili</i>	Bilgisayarların ek bir dönüştürme ihtiyacı duymaksızın doğrudan anlayabildikleri 0 ve 1'lerden oluşan komutlar ve tanımlamalar kümesine makine dili denir. (Yılmaz ve Çağultay, 2004).
<i>Programlama Dilleri</i>	Bir problemin çözümünde bilgisayarda yazılacak çözümün adım adım ve belli bir kural dizisine uygun olarak algılayan tanım kümesine verilen isimdir.

Şekil 1. Tanımlar

1.7. Kısaltmalar

<i>A-12</i>	: Anaokulundan 12. Sınıfa kadar
<i>BTY</i>	: Bilişim Teknolojileri ve Yazılım
<i>CPGF</i>	: Code.org Platformuna Yönelik Yarı Yapılandırılmış Görüşme Formu
<i>DG</i>	: Deney Grubu
<i>EBO</i>	: Eğitsel Bilgisayar Oyunları
<i>EBODKÖY</i>	: Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik
<i>EBODKÖYTÖ</i>	: Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeği
<i>KG</i>	: Kontrol Grubu

<i>MEB</i>	: Milli Eğitim Bakanlığı
<i>MEGEP</i>	: Mesleki Eğitim ve Öğretim Sistemini Güçlendirme Projesi
<i>PÖDA</i>	: Programlama Öğrenim Düzeyi Belirleme Aracı
<i>SPSS</i>	: Statistical Package for the Social Sciences (İstatistik Programı)
<i>TTKB</i>	: Talim Terbiye Kurulu Başkanlığı
<i>YÖK</i>	: Yüksek Öğretim Kurulu



BÖLÜM II

İLGİLİ YAYIN VE ARAŞTIRMALAR

Araştırmamanın bu bölümünde alanyazın desteği ile Bilişim Teknolojileri ve Yazılım (BTY) dersi, programlama, programlama öğretimi konulu çalışmalar ve Code.org platformu kavramlarına ilişkin bilgiler ve araştırmalar yer almaktadır.

2.1. BTY Dersi

2007 yılından itibaren ilköğretim kademesinde (1. - 8. sınıflarda) seçmeli olarak verilen Bilişim Teknolojileri dersi, 2012-2013 eğitim öğretim yılında 4+4+4 olarak yenilenen eğitim sistemi ile Bilişim Teknolojileri ve Yazılım dersi adını almış ve ortaokul kademesinde (5., 6., 7. ve 8. sınıflar) seçmeli olarak okutulması kararlaştırılmıştır. 2013-2014 öğretim yılı öncesinde ise ortaokul 5. ve 6. sınıflarda zorunlu, 7. ve 8. sınıflarda seçmeli olmak üzere haftalık iki ders saati olacak şekilde verilmesi planlanarak güncellenmiştir. 2018-2019 öğretim yılı itibariyle de aynı uygulama devam etmektedir.

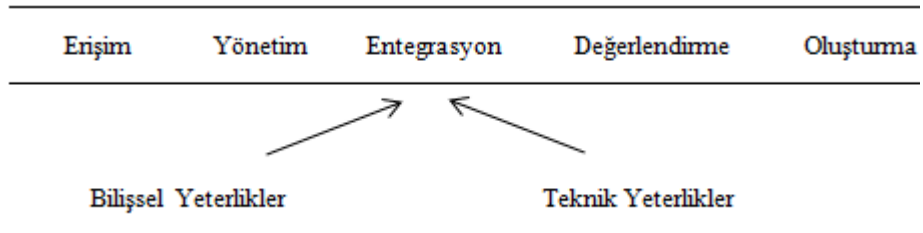
Talim ve Terbiye Kurulu Başkanlığı (TTKB) tarafından hazırlanan BTY dersi öğretim programı için ilk olarak farklı ülkelerce tespit edilen standartlar ve çerçeve öğretim programları irdelenmiş, bilişim teknolojileri kullanımına ilişkin diğer ülkelerin konuya bakış açısı incelenmiştir. Eğitim ve teknoloji konularında gerçekleştirdiği çalışmalarıyla dünya çapında bir topluluk olan “The International Society for Technology in Education (ISTE)”, “Öğrenciler için Ulusal Eğitim Teknolojileri Standartları”nı belirlemiştir (ISTE-NETS-S, 1998, 2007). Bu çalışma okul yöneticileri ve öğretmenler gibi farklı örneklemeler üzerinde de düzenlenmiştir. Öğrenciler için 6 seviye olarak standartlar belirlenmiştir (Şekil 2) (ISTE-NETS-S, 1998, 2007).

ISTE kuruluşunun çalışmasının bir benzerini İrlanda'da bulunan “National Council for Curriculum and Assessment (NCCA)” birimi gerçekleştirmiştir. NCCA (2007) bilişim teknolojilerine yönelik çerçeve olarak dört başlık üzerinde durmuştur (Şekil 2). İngiltere'de bulunan NAACE (2012) kuruluşunun hazırladığı raporda ise beş başlık belirlenmiştir (Şekil 2). Ayrıca bu raporda dijital okur-yazarlığa yönelik temel yetiler belirtilmiştir (Şekil 2). BTY dersi öğretim programı hazırlanırken tüm bu programlar ve standartlar göz önüne alınmış, kapsamlı ve ülkemize uygun yapıda bir program oluşturulmaya dikkat edilmiştir.

ISTE	NCAA (İrlanda)	NAACE (İngiltere)
Öğrenciler için Ulusal Eğitim Teknolojileri Standartları	Bilgi ve İletişim Teknolojileri Çerçevesi	Bilgi ve İletişim Teknolojileri Çerçevesi
Yaratıcılık ve Yenilik	Oluşturma, İletişim ve İşbirliği	Dijital okur-yazarlık
İletişim ve İşbirliği	Temel Bilgi, Beceri ve Kavramları Geliştirme	Beceniler
Araştırma ve Bilgi Akıcılığı	Eleştirel ve Yaratıcı Düşünme	Dünya daki Teknoloji
Eleştirel Düşünme	Bilgi ve İletişim Teknolojilerinin Sosyal ve Bireysel Etkilerini Anlama	Teknik Anlayış
Dijital Vatandaşlık		Gizlilik, Güvenlik ve Hukuk
Teknoloji İşlemleri ve Kavramları		
Dijital okur-yazarlık		
Kültürel ve sosyal anlayış geliştirme	Eleştirel düşünme ve değerlendirme	Teknoloji kullanma becerileri kazanma
	Yaratıcılık	Gizlilik ve güvenlik
	Etkili iletişim kurma	İşbirliği
		Doğru bilgiyi ayırt edebilme

Şekil 2. İlgili standartlar ve öğretim programları

Bilgi ve iletişim teknolojileri kullanımına yönelik öğrenme düzeyleri ve boyutlar belirlenirken de çeşitli çalışmalar üzerinde durulmuştur. ETS (2017) tarafından bilgi ve iletişim okuryazarlığına yönelik olarak 5 seviye dile getirilirken (Şekil 3) California Eyaletinde (2008) ise buna ek olarak “İletişim” alanına yer vermiş ve 6 boyut dile getirilmiştir.



Şekil 3. Bilgi ve iletişim teknolojileri yeterlikleri (ETS, 2007)

Fraillon ve Ainley (2011) gerçekleştirdiği çalışmalarında; bilgisayar ve bilgi okuryazarlığını, bireyin bilgisayarı araştırma, oluşturma ve iletişim amacıyla her ortamda kullanabilme becerisi olarak ifade etmişlerdir. Bu ifade kavramsal olarak temel, orta ve ileri aşama olmak üzere üç seviye ve her seviyede iki ayrı düzey olarak belirlenmiştir. BTY dersi öğretim programı çerçevesinde yeterlik düzeyleri olarak bu sınıflandırma temel alınmıştır. Bu sınıflandırmayla birlikte BTY dersi öğretim programında standartlar doğrultusunda kazanımlar oluşturulurken Tomei'nin teknoloji alanı için belirlediği taksonomiden de faydalanılmıştır (Şekil 4).

Taksonomi Düzeyi	Teknoloji Taksonomisinin Düzey Tanımları
Okuryazarlık	Teknoloji, bilgisayarlar, eğitim programları, ofis verimliliği, yazılım, internet ve bir öğrenme stratejisi olarak tüm öğelerin karşılıklı etkileşiminin etkililiği hakkında öğretmen ve öğrencilerden en alt düzeyde beklenen yeterlik
Teknolojiyi anlama	
İşbirliği	Kişiler arası etkili iletişim için teknolojiyi kullanma becerisi
Fikirleri paylaşma	
Karar verme	Teknolojiyi, yeni ve somut durumlarda analiz ile değerlendirme yapmak ve karara varmak için kullanma becerisi
Problem çözme	
Yayımla	Mevcut teknolojinin özgün öğrenme durumları için tanımlanması, kullanımı ve uygulanması
Teknolojiyle öğrenme	
Bütünleştirme	Teknolojiye dayalı yeni materyaller oluşturma, bu mümkün olmadığı takdirde öğretim için farklı teknolojileri bir araya getirme
Teknolojiyle öğretim	
Teknoloji	Teknoloji kullanımının evrensel etkisini, paylaşılan değerlerini ve sosyal doğurgularını değerlendirme ile teknoloji kullanımının öğretim ve öğrenme üzerindeki etkisini değerlendirme becerisi
Teknoloji kullanımını değerlendirme	

Şekil 4. Teknoloji taksonomisi

Dersin Milli Eğitim Bakanlığı (MEB) tarafından yayımlanan BTY öğretim programında belirlenmiş öğrenme alanları Şekil 5’te verilmiştir.

1	Bilişim Teknolojileri
2	Etik ve Güvenlik
3	İletişim, Araştırma ve İşbirliği
4	Ürün Oluşturma
5	Problem Çözme ve Programlama

Şekil 5. BTY dersi öğrenme alanları (TTKB, 2017)

Öğretim programı içerisindeki düzeyler ve bu düzeylere uygun kazanımlar tespit edilirken Fraillon ve Ainley (2011) ile Tomei (2005) tarafından önerilen sınıflandırmaların her ikisine de uygun olması durumu dikkate alınmıştır (Şekil 6).

Temel I	Temel II	Orta I	Orta II	İleri I	İleri II
Bilişim teknolojilerini kavrama	Bilgiye erişme ve değerlendirme	Bilgiyi yönetme	Bilgiyi dönüştürme	Bilgiyi oluşturma	Bilgiyi paylaşma

Şekil 6. Program için belirlenen düzeyler

Düzeyler → Standartlar ↓	Temel I Düzey: Bilişim teknolojilerini kavrama	Temel II Düzey: Bilgiye erişme ve değerlendirme	Orta I Düzey: Bilgiyi yönetme	Orta II Düzey: Bilgiyi dönüştürme	İleri I Düzey: Bilgiyi oluşturma	İleri II Düzey: Bilgiyi paylaşma
4.1. Problem Çözme Yaklaşımları	• Problem kavramını tanımlar. • BİT kullanımında karşılaşılan problemlerin farkına varır. • Problem çözmenin önemini ifade eder. • Farklı problem çözme yaklaşımlarını listeler. • Algoritma ve strateji kavramlarını tanımlar. • BİT araçları ile algoritma ve strateji kavramlarının ilişkisini farkedir.	• Problem çözme sürecinde problemin çözülebilirliği hakkında yorum yapar. • Problem çözmek için gerekli değişken ve işlemleri belirler.	Belirlenen problemin çözümüne farklı yollardan ulaşır.	Çözümü verilen probleme farklı bir çözüm yolu önerir.	• Belirlenen problemin çözümü için adımlar oluşturur. • Problem çözümü için geliştirdiği adımların geçerliğini sorgulayarak en etkili çözüme ulaşır.	Belirlenen problem için oluşturduğu çözüm önerisini ve yaklaşımını sunar.
4.2. Algoritma ve Strateji Geliştirme	• Belirlenen problemin çözümü için algoritma geliştirmenin önemini ifade eder. • Problem çözme sürecinde strateji geliştirmenin önemini açıklar.	• Farklı algoritmaları inceleyerek en hızlı ve doğru çözümü seçer. • Strateji geliştirmek için gereksinim duyduğu bilgiye erişir. • Eriştiği bilgiyi, strateji geliştirmeye uygunluk açısından değerlendirir.	Belirlenen problemin çözüm sürecinde gerekli işlemler için akış şeması hazırlar.	• Hatalı bir algoritmayı doğru çalışacak biçimde düzenler. • Farklı kaynaklardan edindiği bilgileri, geliştirdiği strateji kapsamında kullanmak üzere dönüştürür.	Günlük hayatta karşılaşılan problemleri çözmek için farklı stratejiler geliştirir.	• Belirlenen problemin çözümünü sağlayan farklı algoritmalar hakkında tartışır. • Geliştirdiği stratejilerin, belirlenen problemin çözümündeki etkisini tartışır.
4.3. Programlama	• Programlama için kullanılan kavramları tanımlar. • Programlama için kullanılan süreçleri açıklar. • Animasyon ve yazılım geliştirme konusunda temel kavramları tanımlar. • Programlama için gereken donanımı açıklar. • Programlama sürecinde uyulması gereken pedagojik ilkeleri listeler.	• Programı çalıştırmak için gerekli derleyiciyi kullanır. • Programlama mantığına ilişkin yaklaşım geliştirir. • Aynı amaç için hazırlanmış yazılımlar arasından verilmiş ölçütlere göre seçim yapar.	• Programın hatasız çalışarak sonuç üretmesini sağlar. • Animasyon için akış şeması ve öykü yapılarını hazırlar. • Hazırladığı akış şeması ve öykü yapılarını programın olası kullanıcılarının görüşlerine göre düzenler.	• Mevcut bir algoritmayı program koduna dönüştürür. • Hazır akışa göre gerekli animasyon sahnelarını oluşturur.	• Açık kaynak kodlu ortamlarda bulunan program kodlarını geliştirerek yeni bir çözüm üretir. • Belirlenen problemi çözmek için animasyon ve yazılım geliştirir. • Eğitsel oyun geliştirir.	• Program kodunu ve çalıştırılabilir dosyayı sosyal ortamlarda paylaşır. • Geliştirdiği eğitsel oyunu sosyal ortamlarda paylaşır.
4.4. Yazılım Projesi Geliştirme, Uygulama ve Yaygınlaştırma	• Proje geliştirme sürecini açıklar. • Proje döngüsünü tanımlar.	Geliştirilmiş proje fikirleri arasından belirli ölçütlere göre seçim yapar.	Projenin nasıl yaygınlaştırılabileceğine ilişkin görüş geliştirir.	Proje fikrini, proje döngüsüne dönüştürerek açıklar.	Proje dokümanını verilen bir şablona uygun biçimde hazırlar.	• Projenin uygulama sonuçlarını sosyal medya ortamında paylaşır. • Projenin yaygınlaştırılması ile ilgili görüşlerini proje paydaşları ile paylaşır.

Şekil 7. Problem çözme ve programlama ünitesi (TTKB, 2012)

TTKB tarafından hazırlanan öğretim programı içerisindeki “Problem çözme ve programlama” ünitesinin belirlenen düzeylerle ilişkilendirilmiş hali Şekil 7’de verilmiştir. Ayrıca BTY dersi öğretim programının özel amaçları Şekil 8’deki gibi maddeleştirilmiştir. TTKB (2017), BTY dersi öğretim programı içerisindeki 5. ve 6. sınıflara ait “Problem Çözme ve Programlama” alanı kazanımları ise Ek-9 ve Ek-10’da gösterilmiştir.

1	Dijital vatandaş olarak teknolojik kavramları, sistemleri ve işlemleri iyi anlayan bireyler olmaları
2	Bilişim teknolojilerini etkili ve amacına uygun kullanmaları
3	İnternet tabanlı servislere erişmelerini, araştırmalarını ve kullanmaları
4	Bilgisayar bilimine ilişkin genel bir anlayış ve teknik birikim oluşturmaları
5	Problem çözme ve bilgi-işlemsel düşünme becerileri edinmelerini ve geliştirmeleri
6	Akıl yürütme sürecini takip edebilmelerini ve değerlendirmeleri
7	Öğrenme sürecinin bir parçası olarak iş birlikli çalışma becerileri edinmelerini, sosyal ortamlardan faydalanmalarını ve öğrendiklerini paylaşmaları
8	İnternet ortamında öğrenme fırsatları aramaları
9	Algoritma tasarımına ilişkin anlayış geliştirerek sözel ve görsel olarak ifade edebilmeleri
10	Problemleri çözmek için uygun programlama yaklaşımını seçerek uygulayabilmeleri
11	Programlama konusunda teknik birikim oluşturmaları
12	Programlama dillerinden en az birini kullanabilmeleri
13	Ürün tasarımı ve yönetimi konusunda çalışmalar yürütmeleri
14	Günlük hayatta karşılaşılan sorunların (yaşlı ve engelli bireylerin karşılaştığı sorunlar vb.) çözümüne ilişkin yenilikçi ve özgün projeler geliştirmeleri
15	Yaşam boyu öğrenme konusunda bilinç kazanmaları

Şekil 8. BTY dersi öğretim programının özel amaçları (TTKB, 2017)

2.2. Programlama

Hayatımız boyunca problemlerle karşılaşırız ve çoğu zaman bu problemleri hemen çözemeyiz. Karşılaştığımız problemler arttıkça ve karmaşıktıkça çözüme ilişkin denemelerimiz ve hatalarımız da artar. Programlama; öğrencilere problem çözmenin bir amaç olmadığını, tecrübe edilen sürecin daha önemli olduğunu öğretir. Bu öğreti, kişilerin hayatın zorluklarına karşı koyabilecekleri veya ileride iş hayatlarına taşıyabilecekleri bir yaşam kabiliyetidir.

Kodlama veya programlama kavramları tanım olarak incelendiğinde, bilgisayarı yönlendirmek için komutların bir araya getirilme süreci ifadesine rastlanılmaktadır (Turan, Akça ve Küçük Kurt, 2016). Kesici ve Kocabaş (2007) ise programlamayı şöyle tanımlamıştır: Bilgisayarlar gerçekleştirecekleri işlemleri, anlayabilecekleri komut dizileri halinde getirirler. Bu komutların bir araya getirilerek derlenmesi ve çalıştırılması sürecinde ortaya çıkan işlem algoritması programlama olarak adlandırılmaktadır. Bir başka tabirle programlama mevcut bir problemi çözmek için bilgisayarın algılayabileceği bir dil ile o probleme çözüm üretmektir.

Bilgisayarın algılayabileceği dil ifadesinden de anlaşılacağı üzere kullanıcı ve bilgisayar arasındaki iletişimi gerçekleştirebilecek bir aracıya gereksinim duyulmuştur. Bu gereksinim ilk olarak makine dili ile karşılanmaya çalışılmıştır. 1950'li yıllar itibarıyla programlar, makine diliyle "0" ve "1" bilgilerine göre oluşturulmaktaydı. "0" yok veya yanlış, "1" ise var veya doğru bilgisini iletmekteydi. Sonraki süreçte programlamanın zor ve anlaşılması güç olmasından dolayı makine diline benzeyen ve işlemciler tarafından algılanabilen Assembly programlama dili kullanılmaya başlandı. 1980'li yıllardan itibaren de daha üst düzeydeki programlama dilleri yaygınlaşarak kullanılmaya başlamıştır (Tekin ve diğerleri, 2003). C, Java, C++, C#, JavaScript, PHP, HTML, Visual Basic vb. birçok programlama dili kullanıcı ile bilgisayar arasındaki köprü görevini farklı alanlarda işlevsel olarak yerine getirmiştir.

Bilgisayar programlama konusunda ilgili programlama dillerinin öğretilmesi çeşitli kavramları beraberinde getirmiştir. Programlama öğretimi konusunda dikkat çeken en önemli iki kavram algoritma ve akış şeması kavramlarıdır. Var olan bir problemin çözümü için gerekli olan basamakların adım adım hesap edilerek mantıksal bir sıraya konulması işlemi algoritma olarak adlandırılmaktadır. Kullanıcı tarafından girilen iki sayının toplamını hesaplayarak sonucu ekrana yazdıran programın algoritması Şekil 9'da verilmiştir.

Adım 1: Başla

Adım 2: İlk sayıyı gir

Adım 3. İkinci sayıyı gir

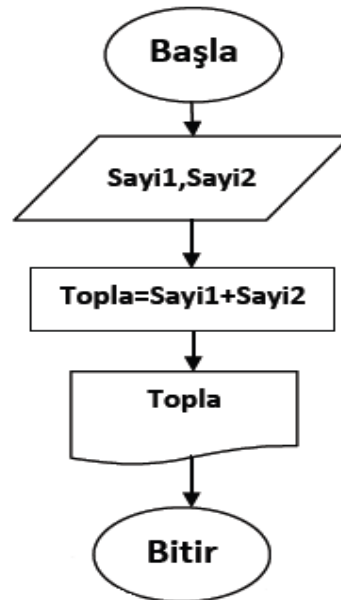
Adım 4: Sayıları topla

Adım 5: Ekran yazdır

Adım 6: Bitir

Şekil 9. Algoritma örneği

Oluşturulan algoritmaların şekillerle görselleştirilerek birbirleriyle ilişkilendirilmesi sonucu ortaya çıkan yapıya ise akış şeması ismi verilmektedir. Akış şemaları var olan problemlerin çözümü için gerekli bir aşamayı oluşturmaktadır. Kullanıcı tarafından girilen iki sayının toplamını hesaplayarak sonucu ekrana yazdıran programın akış şeması Şekil 10'da gösterilmiştir.



Şekil 10. Akış şeması örneği

2.3. Programlama Öğretimi

Yaşantımızda sürekli olarak problemlerle karşılaşmakta ve bu problemlerin çözümüyle uğraşmaktayız. Problemlerimizle akılcıca başa çıkabilmek içinse yaptığımız hataları doğru bir analiz sürecine tabi tutmalıyız. Araştırma konumuz olan programlama öğretiminin kapsadığı önemli bir kavramlardan biri hata ayıklamadır. Sorunların sebebini bulabilmek program yazma sürecinin bir parçasıdır. Öğrencilere erken dönemde nasıl hata ayıklanacağını öğretmek, iyi bir öğrenmeye olanak sağlamasının yanı sıra onları yaşamın karmaşık zorluklarına da hazırlar. Durak ve Samur (2018), programlama öğretim uygulamalarının hayat boyu öğrenme yeterliklerinin geliştirilmesinde önemli bir etkiye sahip olduğunu ifade etmektedir. Çocuklar var olan kodlar üzerinde hata ayıklayarak problemleri ilk denemesinde veya 30. denemesinde çözme öğrenmez. Burada önemli olan nokta her seferinde amaca yaklaşılacağıdır, hatalardan korkmamaktır. Ancak bu sayede karşılaşılan problemlere çözümler üretebilen, özgüveni yüksek, kendi yeteneklerini keşfedebilen nesiller yetiştirebiliriz.

Bilgisayar programlamanın, öğrencilerin bilişsel gelişimlerine katkı sağlamasına ilişkin uzun süredir araştırmalar yapılmaktadır (Papert, 1980; Clements ve Gullo, 1984; Akpınar ve Altun, 2014). Özellikle erken yaşlardaki bireylerde bu durum, incelenen bir konu olmuştur. Gorman ve Bourne (1983), on iki yaşındaki öğrencilerle gerçekleştirdikleri araştırmalarında; haftada bir saat programlama faaliyetlerinde yer alanların, bu faaliyetlerde yarım saat bulunanlara oranla kural öğreniminde daha başarılı olduklarını bildirmiştir (Akpınar ve Altun, 2014).

Öğrenciler kodlama yaparken, öncelikle var olan probleme bir çözüm üretmek, sonrasında ise bu çözümü bilgisayara iletmenin bir yolunu bulmak zorundadırlar ve bu işlem örüntüsünün tüm detaylarıyla eksiksiz olması gerekmektedir (Papert, 1980; Szilvi ve Zsako, 2006). Söz konusu işlem, öğrencinin doğal dil örüntüsünün gelişimine de katkı sağlayabilmektedir (Hromokovič, 2006). Ayrıca programlamada bir soruna çözüm bulma faaliyeti, alt parçalara ayırma ve çözümleme işlemlerini içerdiği için öğrencilerin analiz becerisini de geliştirebilmektedir (Saali, Perrenet, Jochems ve Zwaneveld, 2011). Sonuç olarak programlamayla farklı alanlardaki problem çözme yetilerinin gelişimi de sağlanmaktadır ki bu alanda yapılan araştırmalarda, ulaşılan sonucu destekleyici kuvvetli bulgular elde edilmektedir (Feurzeig ve diğerleri, 1970; Papert, 1993; Mulder, 2002; Dasso ve diğerleri, 2005; akt: Akpınar ve Altun, 2014).

Programlama, mantıksal akıl yürütme sürecinin bir parçası olarak görülmekte ve günümüzde sıklıkla rastladığımız “21. yüzyıl becerileri” olarak isimlendirilen becerilerden biri olarak dile getirilmektedir. Ayrıca günümüzde adından çok söz ettirerek eğitim kurumlarında veya iş hayatında bireyleri değerlendiren uzmanlar için anahtar bir yetkinlik olarak görülmektedir. Çünkü dijital beceriler, dünyadaki dijital dönüşüm ve ekonomik büyümenin, bireylerin refah düzeyinin yükselmesinin ve dijital teknoloji pazar stratejisinin gerçekleşebilmesinin ön şartı olarak görülmektedir (Sayın ve Seferoğlu, 2016).

Türkiye’de BTY dersi kapsamında değerlendirilen programlama öğretimi çerçevesinde paylaşmanın öne çıktığı ve birlikte geliştirmeyi temele alan sosyal kodlama ortamları önerilmektedir. Çünkü kodlama öğrenme süreci öğrencilerin kodlama yapabilmelerinin yanında onların farklı alanlardaki öğrenmelerini de etkilemektedir. Örnek olarak matematik veya bilişimsel düşünme becerileri öğrenilirken; problem çözme, proje tasarlama ve fikirler arasında iletişim kurma gibi olguların öğrenimi de desteklenmektedir. Bu beceriler bilgisayar bilimcilerinin yanı sıra ilgi alanı veya mesleklerden bağımsız olarak herkesin gereksinimi olan yetkinlikler olarak görülebilirler (Sayın ve Seferoğlu, 2016). Programlama yaparak farklı alanlarındaki kavramların ve süreçlerin derinlemesine çalışılması, daha kolay anlaşılabilmesi ve işevruruklaşabilmesi daha mümkün hale gelmektedir (Resnick ve Ocko, 1990). Örneğin Matematikteki değişken, fonksiyon, alt öğeler gibi kavramlar programlama eğitimi içerisinde daha basit anlaşılabilir veya bir kavramın oluşum sürecinin incelenmesi programlamanın yapısı gereği kolayca gerçekleştirilebilmektedir. Böylece ders tasarımcısı ve/veya öğretmeni ilgili dersin faaliyetlerini bilgi aktaran şekilde değil, bilgi oluşturulabilen şekilde düzenleyerek farklı kazanımlar elde edebilmektedir (Akpınar ve Altun, 2014).

Teknolojik gelişmelerde yaşanan hızlı değişimler, öğrenme-öğretme biçimlerini ve bilginin öğrenciye sunulmuş biçimini etkileyerek değiştirmektedir. Bu durum eğitim öğretimde de yeniliklere neden olmakta, var olan yöntem ve teknikler teknolojik değişimlerin etkisiyle gözden geçirilerek bireyin daha nitelikli eğitimi için dönüşüme uğramaktadır. Özellikle bilişim teknolojilerinin öğrenme-öğretme süreçlerindeki vazgeçilmezliğinin, araştırmacıları yeni kuramlar ve uygulama yolları arayışına yönlendirdiği bir gerçektir (Seferoğlu, 2007).

Genel olarak bilgisayar kullanımının eğitim-öğretime katkısını konu alan çalışmalarda normal metotların kullanıldığı öğretime oranla bilgisayar destekli öğretimin akademik başarıyı %10-18 daha fazla yükselttiği dile getirilmektedir (Kamacı ve Durukan, 2012). Akpınar (2006), Gürbulak (2013), Kapucu (2017) ve Karabulutlu (2018) da gerçekleştirdiği

çalışmalarında; bilgisayar destekli eğitimin öğrencilerin başarılarını olumlu yönde etkilediğini ifade etmişlerdir. Araçlarla desteklenen bir öğretim normal metotların yer aldığı ortamlara nazaran daha ilgi çekici, sürükleyici, verimli, zenginleşmiş ve ekonomik haldedir. Ayrıca eğitim araçları, gerçekleştirilen öğrenme işleminin kalıcılığı açısından da önem taşımaktadır. (Korkmaz, 1997). Alan ve Taşdemir (2016), bilgisayar destekli eğitim yöntemlerini kullanarak bilgisayar bilimlerinde algoritma yapısının öğretilmesine destek olacak üç boyutlu görsel bir oyun hazırlamışlar; bu oyun ile algoritma yapılarının eğlenceli ve akılda kalıcı bir şekilde öğrenilebileceğini söylemişlerdir.

Cobun'a (1968) göre öğrendiklerimizin; %83'ünü görme duyu organımızla edindiğimiz yaşantılar vasıtasıyla öğrenmekteyiz (Çilenti, 1985). Korkmaz (1997) ise çalışmasında zaman değişkeni sabit tutulmak kaydıyla insanların; okuduklarının %10'unu, duyduklarının %20'sini, gördüklerinin %30'unu, hem duyup hem gördüklerinin %50'sini, söylediklerinin %70'ini, yapıp söylediklerinin ise %90'ını hatırladıklarını ifade etmektedir. Ayrıca çalışmalar kapsamında derslerde çoklu ortam araçlarının kullanımına ilişkin alanyazındaki birçok araştırmada, öğrenci başarısının arttığı dile getirilmektedir (Yaman, 2005). Görüldüğü üzere çoklu ortam materyallerinin kullanıldığı, yaparak yaşayarak öğrenmeye dayalı öğrenme ortamları öğretimin etkililiği açısından çok önemli bir yere sahiptir. Görselliği ve etkileşimi ön planda tutan blok temelli programlama ortamları da bu açıdan programlama öğretiminde öne çıkmaktadır.

Geleneksel programlama dillerinin sahip olduğu karmaşık yapı ve bunun getirdiği öğrenim zorluğu, programlama ile yeni tanışan kişilerin karşılaştığı büyük bir engel konumundadır. Bu problemin giderilmesi için, öğrenilmesi daha basit ve görselliğin ön planda yer aldığı programlama öğrenme ortamları geliştirilmiştir (Çatlak ve diğerleri, 2015). Bu görsel öğrenme araçlarının, programlama öğrenimine ve öğrencilerdeki beceri gelişimine olan pozitif katkılarından yola çıkarak birçok ülkenin eğitim müfredatında yer almaya başladığı ifade edilmektedir (Saygıner ve Tüzün, 2018).

Dünya genelindeki kodlama konusunda gerçekleştirilen çalışmalar göz önüne alındığında, programlama eğitiminin özellikle erken yaşta verilmesinin gerekliliği anlaşılmaktadır. Yapılan araştırmalar, A-12 seviyesindeki kişilerin daha etkili ve basit bir şekilde programlama öğrenebilmesi için farklı öğrenme araçlarının geliştirilmekte olduğunu, çeşitli eğitim kuruluşları ve organizasyonları tarafından birçok projelerin başlatıldığını göstermektedir (Demirer ve Sak, 2016). Programlama eğitimini A-12 düzeyinde müfredatına entegre eden ülkelere bakıldığında dünyanın birçok noktasında (İspanya, Belçika, Estonya,

İngiltere, Portekiz, Fransa, Avustralya, ABD, Güney Kore, Tayvan ...) çalışmaların yapıldığı görülmektedir (Demirer ve Sak, 2016; Sayın ve Seferoğlu, 2016). Bunun yanı sıra çeşitli organizasyonlar ve yarışmalar düzenlenerek uluslararası alanda yaygınlaştırmanın sağlanması için kodlama faaliyetleri gerçekleştirilmektedir.

2.4. Code.org Platformu

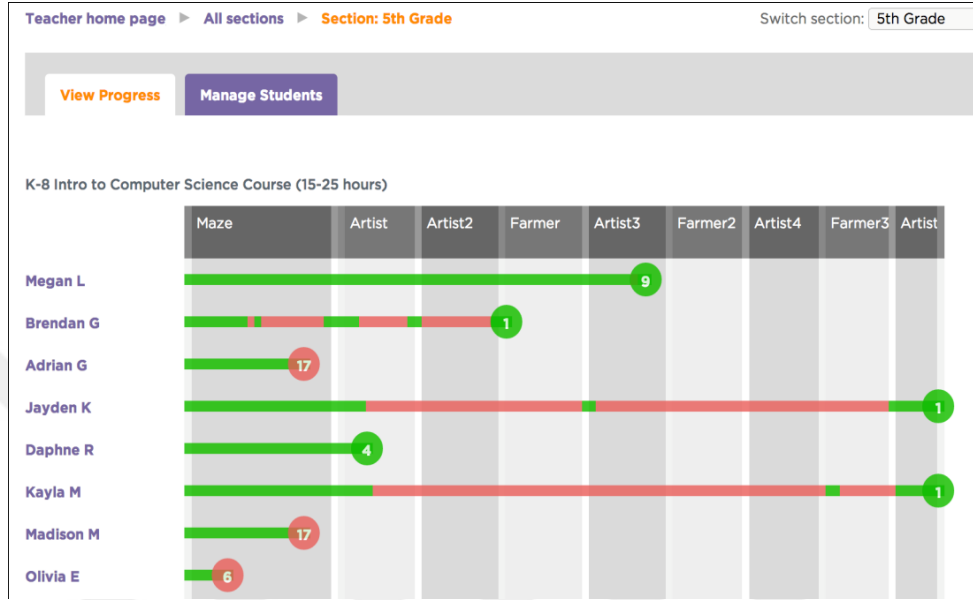
2013 yılında Ali Patrovi ve Hadi Patrovi isimli iki kardeşin öncülüğünde başlatılan Code.org projesi, öğrencilere bilgisayar bilimini öğrenme fırsatı sunmaktadır. 34 dilde sunulan site oyun olarak yapılandırılmış ve ünlü kişilerin de yer aldığı posterlerden ve video konferanslardan oluşmaktadır (Kalelioğlu, 2015). 38 milyon öğrenci tarafından kullanılmış olan Code.org platformu kapsamında, 41 milyonun üzerinde proje oluşturulmuştur (Code.org, 2019).



Şekil 11. Code.org ekran görüntüsü

ABD hükümeti, sivil toplum kuruluşları, Google, Microsoft, Facebook, Amazon gibi teknoloji ve yazılım odaklı firmaların desteğiyle eğitim kurumlarında kodlama eğitimini yaygın hale getirmek ve öğrencilerin programlama yeteneklerini geliştirmek amacıyla "Herkes kodlamayı öğrenebilir" çağrısı yapmıştır (Code.org, 2019). Code.org platformu bünyesinde kodlama eğitime yönelik müfredat oluşturulmuştur. Yaygınlığın sağlanması doğrultusunda 180 ülkeden katılımcılarla, A-12 seviyesinde her yıl Aralık ayında düzenlenen "Hour of Code" gibi etkinlikler yapılmakta, bu sayede öğrencilerin problem çözme ve kodlama yetilerinin geliştirilmesi amaçlanmaktadır.

Öğretmen-öğrenci kullanımına uygun olan platform ile sanal sınıflar oluşturulabilmekte (Şekil 12), bu sayede süreklilik sağlanarak öğrencilerin durum takibi yapılabilir.



Şekil 12. Code.org öğrenci takibi ekran görüntüsü

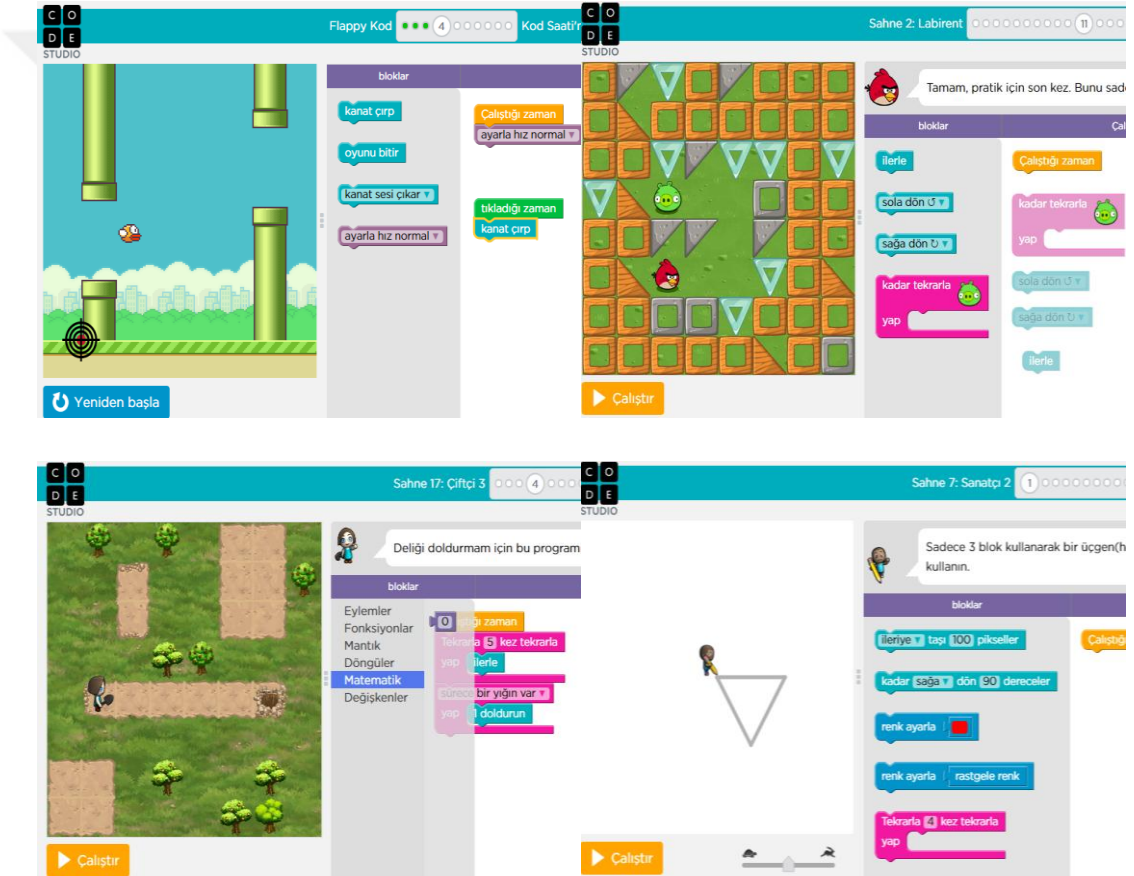
Code.org platformu içerisinde yazılım camiasından birçoğunu tanıdığımız Mark Zuckerberg, Bill Gates, will.i.am, Chris Bosh, Jack Dorsey, Elena Silenok gibi isimler eğitim süresi boyunca öğrencilere videolar aracılığıyla rehberlik etmektedir (Şekil 13).



Şekil 13. Code.org destekçilerinden bazıları

Site içeriğinde 631.929 sözcük bulunmaktadır. Bu içeriğin % 9'u (56.660 sözcük) onaylanmış olmak üzere toplamda % 34'lük kısmı (216.148 sözcük) Türkçe'ye çevrilmiştir (Code.org Translations, 2019). Çeviri işleminin tamamlanmaması sebebiyle içeriğin çoğunluğu Türkçe altyazılı olarak verilmektedir.

Çocukların ilgisini çekmek için angry birds, flappy bird gibi oyunlar ile kodlama daha eğlenceli öğrenilebilmektedir. Sürükle bırak ile oluşturulan kodların sonucunu çalıştır butonuna tıklayarak görülebilmekte böylece öğrenci nerede hata yaptığını görüp tekrar oynayarak düzeltebilmektedir.



Şekil 14. Code.org örnek ekran görüntüleri

2.5. Görsel Programlama Ortamları ve İlgili Çalışmalar

Belirtildiği üzere bu çalışma çerçevesinde Code.org platformu kullanımı incelenecektir ve önceki başlıkta platform tanıtılmıştır. Alanyazına baktığımızda Code.org ortamının programlama öğrenim düzeyine etkisini içeren veya başka bir ortamla karşılaştırıldığı çalışmaların eksikliği gözlenmektedir. Var olan çalışmalar şunlardır:

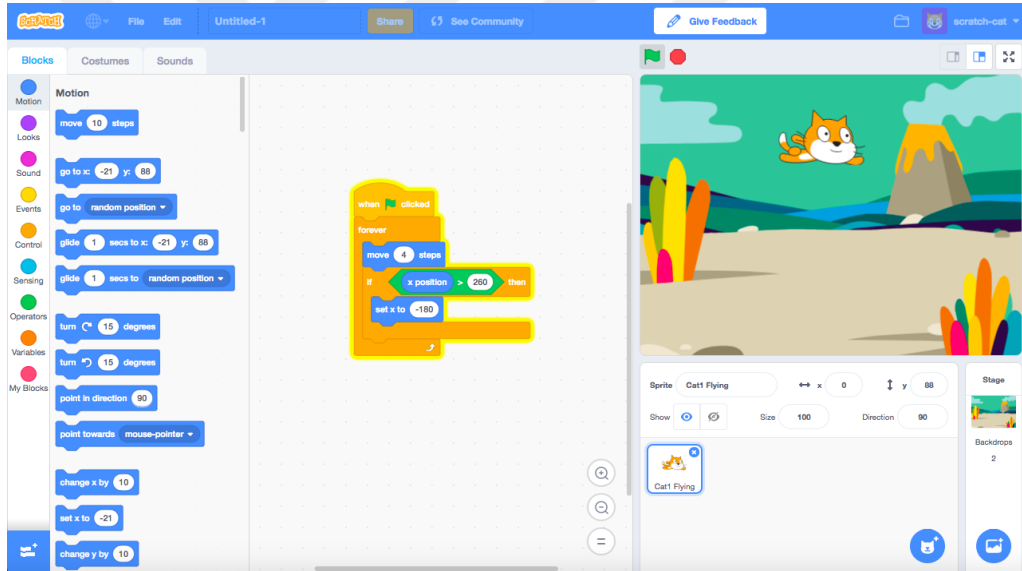
Yıldız (2018), blok temelli algoritmik düşünme aktivitelerinin kullanımının öğrencilerin tutum, katılım ve becerileri üzerindeki etkilerini incelemek amacıyla iki deney grubunda 29'ar ve bir kontrol grubunda 30 olmak üzere toplam 88 ortaokul 5. sınıf öğrencisi ile bir çalışma gerçekleştirmiştir. Karma araştırma yönteminin çeşitleme deseninin kullanıldığı çalışmada toplam 3 hafta sürmüş, öğrenciler blok temelli Code.org ortamında algoritmik aktiviteler yapmışlardır. Deney gruplarında oyunlaştırılmış öğrenme ortamında her hafta farklı bir strateji ön plana çıkarılırken (rekabet, ödül, zorluk), kontrol grubunda yalnızca Code.org öğrenme aracına yer verilmiştir. Farklı oyunlaştırma stratejilerinin ön plana çıktığı haftalarda deney grupları arasında tutum, katılım ve beceri düzeyleri arasında anlamlı bir fark bulunmazken oyunlaştırılmış ortamda öğrenim gören öğrenciler ile kontrol grubu öğrencileri arasında aynı boyutlarda anlamlı farklılıklar bulunmuştur. Nitel veriler ile elde edilen bulgularda da öğrenciler, oyunlaştırılmış öğrenme ortamlarındaki rekabeti ve heyecanı daha öğretici ve eğlenceli bulduklarını ifade etmişlerdir. Araştırma neticesinde oyunlaştırmanın dinamik, mekanik ve bileşenleri ile bir bütün olduğu ama oyunlaştırmanın sadece belli unsurlarının kullanımının da normal metotların yer aldığı eğitime göre daha eğlenceli, teşvik edici ve olumlu tutum geliştirici etkileri bulunduğu sonucuna ulaşılmıştır.

Baz (2018) gerçekleştirdiği çalışmasında; çocuklar için kodlama ortamlarından 40 tanesini farklı kriterlere (ücret durumu, hedef yaş aralığı, desteklenen dil seçenekleri, mobil cihazlara uyumlu olması, örnek projelerin kullanımı, sosyal medya/blog hesapları durumu, yardım desteği, desteklenen işletim sistemi seçenekleri) göre karşılaştırarak incelemiştir. Gerçekleştirilen analizlerin sonucunda, Code.org, Scratch ve App Inventor görsel programlama uygulamalarının diğer araçlara oranla daha fazla özellik ve fonksiyonlara sahip olduğunu ifade etmiştir. Bu açıdan bahsi geçen öğrenme araçlarına öğrencilerin programlama öğrenimlerinde yer vermenin daha faydalı olabileceği sonucuna ulaşmıştır.

Yıldız (2018), blok tabanlı kodlama ortamları ile öğrencilerin problem çözme süreçlerinde gösterdikleri davranışları belirlemek için 15 altıncı sınıf öğrencisiyle bir dönem boyunca devam eden BTY dersinde, Code.org ortamındaki problem çözme etkinliklerini gerçekleştirmiştir. Çalışmada problem çözme süreçleri betimlenerek elde edilen veriler ile sürecin temel aşamaları ortaya çıkarılmaya çalışılmaktadır. Sonuç olarak, Code.org üzerinde problem çözme süreçlerinin temel olarak Odaklanma, Sınama ve Sonuçlandırma şeklinde 3 aşamada gerçekleştiği belirlenmiştir. Bu aşamalar farklı faktörlerin etkisiyle bazı durumlarda birbirini takip eden sırada doğrusal biçimde gerçekleşirken, bazı durumlarda ise aşamalar arası döngüsel geçişler söz konusu olabildiği gözlemlenmiştir.

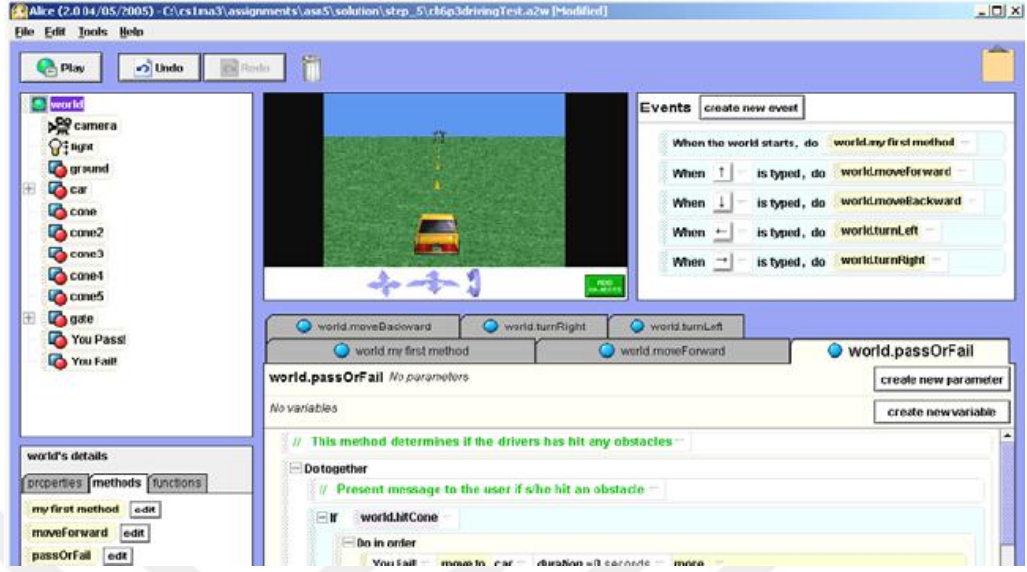
Patan (2016) yaptığı çalışmada; okul öncesi çağında bulunan 4 ve 5 yaş çocuklarının bilgi işlemsel düşünme yetilerini geliştirme amaçlı öğretim programının tasarlanması amaçlamıştır. Projede sınıf içi oyun, drama gibi etkinliklere ve ipad uygulamaları kullanımına yer verilerek bilgi işlemsel düşünmenin geliştirilmesi desteklenmiştir. Projenin ipad kullanımında programlama öğretiminde Kodable ve Code.org; animasyon tasarımında Flipboom ve dijital hikâyelemede Scratchjr kullanılmıştır. Araştırma sonucunda okul öncesi seviyesinde kodlama öğrenimine tabi tutulan öğrencilerin tutumlarında pozitif değişimler gösterdikleri tespit edilmiştir.

Code.org platformu dışındaki diğer görsel programlama ortamları incelendiğinde Scratch, Alice, Kodu Game Lab, Hacker Can, mBlock, Code Combat, Blockly gibi ortamların programlama öğretiminde kullanılabildiğini görmekteyiz.



Şekil 15. Scratch ekran görüntüsü

Scratch, 50 çeşit dil desteği ve yaklaşık 2 milyon kullanıcıya sahip, içerdiği kütüphane sayesinde kullanıcıların projelerini içerisinde barındıran basit ve anlaşılır bir platformdur (Maloney, Resnick, Rusk, Silverman ve Eastmond, 2010) (Şekil 15).



Şekil 16. Alice ekran görüntüsü

Carnegie Mellon Üniversitesince geliştirilen Alice, C ++ gibi nesne yönelimli programlama dilleri kavramlarını öğretmek için tasarlanmış 3 boyutlu bir programlama aracıdır (Şekil 16).

Kodu Game Lab görsel programlama aracı ise Microsoft firması tarafından masaüstü aygıtlar için hazırlanmış, oyun yapımında kullanılan üç boyutlu bir yazılımdır. 24 farklı dil seçeneği ile 2009 yılından itibaren kullanıcılara hizmet vermektedir (Şekil 17).



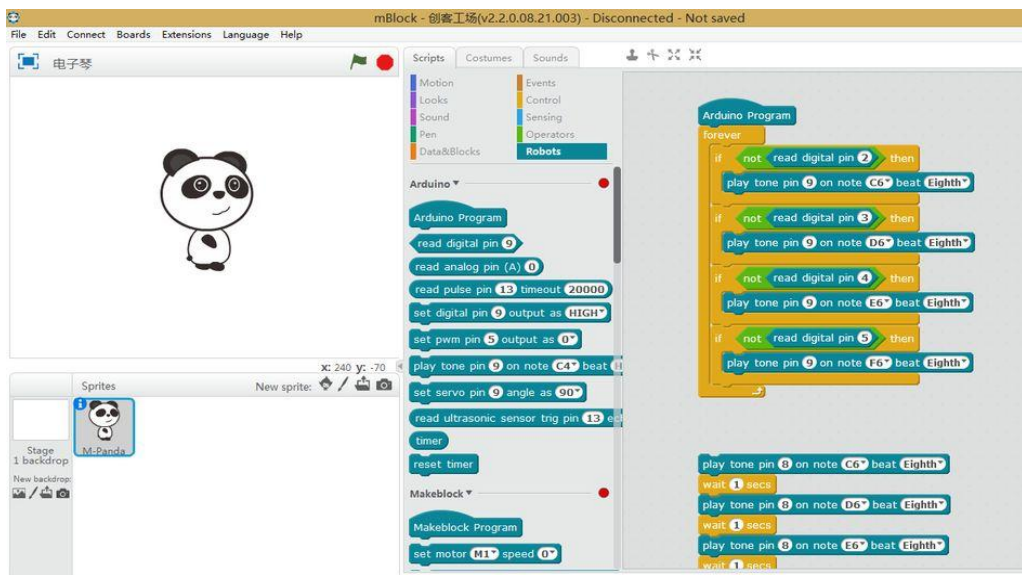
Şekil 17. Kodu Game Lab ekran görüntüsü



Şekil 18. Hacker Can ekran görüntüsü

Hacker Can %100 Türkçe olan bir görsel programlama aracıdır. Noktalama işaretlerinin ve yabancı karakterlerin en aza indirildiği CoffeeScript modelinde bir dil öğretir. Şuan dil desteği olarak Türkçe ve İngilizce seçenekleri bulunmaktadır (Şekil 18).

Scratch programı mantığıyla çalışan mBlock ise temel olarak mBot gibi robotların kontrolü için kullanılan görsel programlama ortamıdır. Scratch programına benzer şekilde blok tabanlı yapıda olan mBlock, değişik olarak Robotlar menü alanına sahiptir (Şekil 19).

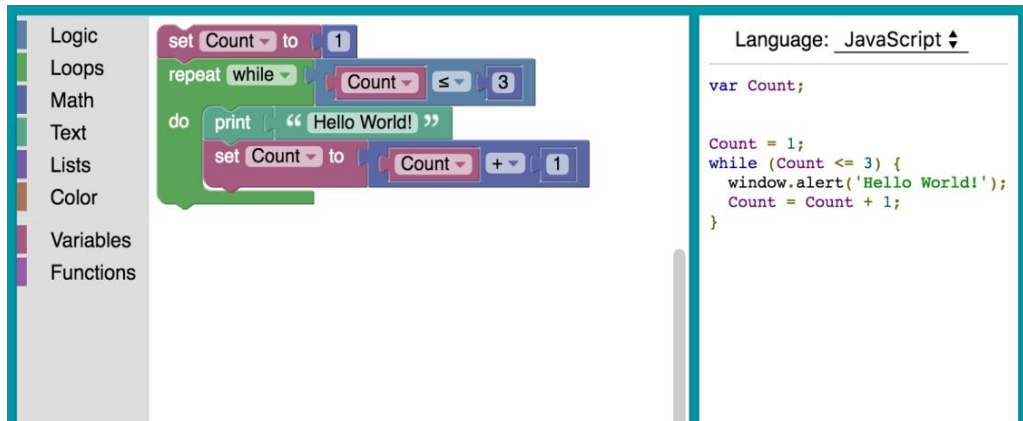


Şekil 19. mBlock ekran görüntüsü



Şekil 20. CodeCombat ekran görüntüsü

Code.org platformu ekibi tarafından geliştirilen ve web tabanlı bir strateji oyunu olan Code Combat, Lua, JavaScript ve Python programlama dillerinin öğrenimi için tasarlanmıştır. Cep telefonu, tablet veya masaüstü aygıtlardan rahatlıkla ulaşılabilir. 200'den fazla ülkede, 50'den fazla dil seçeneği ile yüksek katılımları hedefleyen Code Combat 2013 yılından itibaren hizmet vermektedir (Şekil 20).

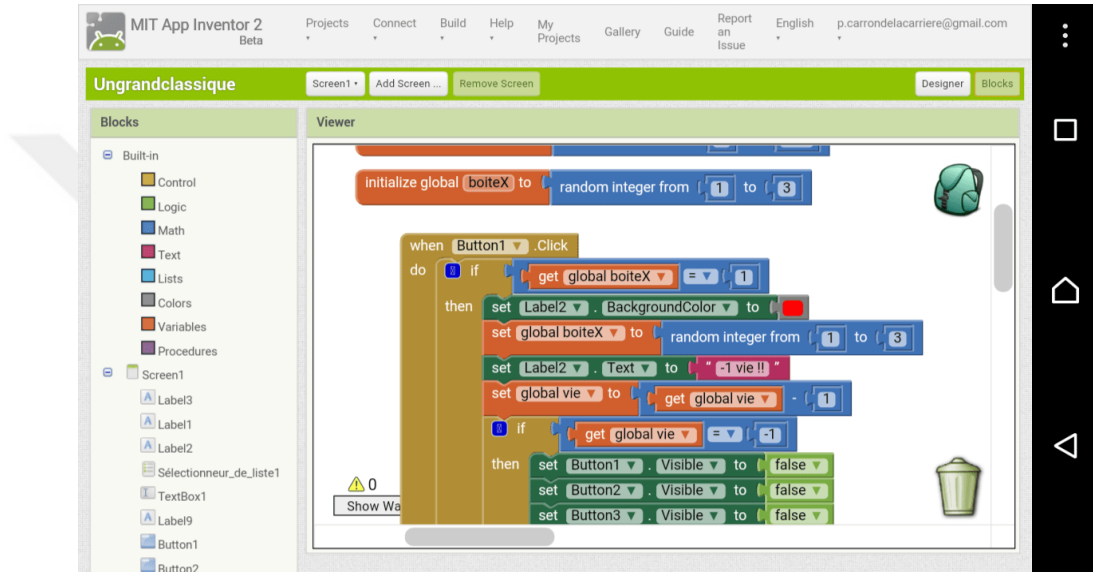


Şekil 21. Blockly ekran görüntüsü

2012 yılı Haziran ayında kullanımına başlanılan Google Blockly programı vasıtasıyla JavaScript dili öğrenilebilmektedir. Görsel biçimde kodlama öğrenilmesini hedefleyen uygulama, bir Google kütüphanesidir. Farklı uygulamalarda da rastladığımız üzere sürekli bırak yoluyla yapılan kodlama işlemi sonrası aşama başarılı bir şekilde

tamamlanırsa ekranda yapılan işlemin JavaScript kodlaması görülmekte ve bir sonraki görev ekranına geçilmektedir (Şekil 21).

App Inventor ise Google firması tarafından programlama bilgisi olmayan son kullanıcıların Android işletim sistemine sahip mobil akıllı cihazlarda uygulama geliştirebilmesi amacıyla üretilmiş bir programlama dilidir. 2011 yılında ücretsiz olarak kamu erişimine sunmuştur (Şekil 22).



Şekil 22. App Inventor ekran görüntüsü

Bahsedilen görsel programlama ortamlarıyla ilgili var olan çalışmalar bu araştırmaya da ışık tutabilir. Programlama dili öğretiminde Alice görsel programlama aracını kullanan Kayabaşı (2016) ve Solmaz (2014), görsel programlama aracının programlama öğretimine katkıda bulunduğu sonucuna ulaşmıştır. Solmaz (2014), C++ ve java dillerinde ders başarısının arttığını ve Kayabaşı (2016) da derse olan ilginin, katılımın ve ders başarısının arttığı sonucuna ulaşmıştır.

Programlama dili öğretiminde Scratch görsel programlama aracı ile ilgili birçok çalışmaya rastlanılmıştır. Bu çalışmalarda Scratch ile programlamanın daha eğlenceli olduğu (Ozoran ve diğerleri, 2012; Yıldırım, 2016); öğrencilerin programlamaya yönelik kaygı düzeyini pozitif olarak etkilediği (Genç ve Karaguş, 2011; Demir, 2015; Dinçer, 2018); programlama öğretiminin (Osman, Zakaria, Loke ve Downe, 2012; Demir, 2015; Yıldırım, 2017; Dinçer, 2018) ve oyunlaştırmanın (Yıldırım, 2016; Yükseltürk ve Altıok, 2016) öğrenci başarısını arttırdığı; işbirlikli öğrenme ortamı sağladığı (Taylor ve diğerleri, 2010);

öğrencilerin etkileşimde bulunmalarına yardımcı olduğu (Maloney ve diğerleri, 2010; Lopez, Gonzalez ve Cano, 2016); derse olan tutumu pozitif olarak etkilediği (Gülbahar ve Kalelioğlu, 2014; Yüksel, 2017); ortamın eğlenceli olmasına katkı sağladığı (Lopez ve diğerleri, 2016); öğrencilerin programlamaya olan isteklerini arttırdığı (Gülbahar ve Kalelioğlu, 2014; Kobsiripat, 2014) ve programlamayı kolaylaştırdığı (Maloney ve diğerleri, 2010) sonuçlarına ulaşılmıştır.

Saygıner (2017) gerçekleştirdiği çalışmada; öğrencilerin mantıksal düşünme, erişim ve programlamaya yönelik motivasyonlarına, metin tabanlı ve blok tabanlı görsel programlama öğretimlerinin etkilerini belirlemeyi amaçlamıştır. Analizler sonucunda; blok tabanlı görsel programlama (Scratch ile) öğretiminin yapıldığı öğrencilerin erişim, mantıksal düşünme ve motivasyon parametrelerinde istatistiksel olarak anlamlı bir artış elde edilmiş ve incelenen değişkenler için yüksek düzeyde bir etki büyüklüğü hesaplanmıştır. Metin tabanlı programlama (Small Basic ile) öğretiminin yapıldığı grupta ise erişim parametresinde yüksek, mantıksal düşünme parametresinde orta düzeyde anlamlı bir artış gözlenirken öğrencilerin motivasyonlarında istatistiksel olarak anlamlı bir artış belirlenmemiştir. Özcan ve diğerleri (2017); çalışmalarında Scratch programına ilişkin toplanan öğrenci görüşleri üzerinde durmuşlardır. Bu görüşler incelendiğinde Scratch görsel programlama aracının eğlenceli, iyi ve programlama eğitime yararlı olduğu, öğrencilerde programlama mantığı, algoritma öğrenimi ve yaratıcılık gibi kavramların gelişimine katkı sağladığı görüşlerine ulaşılmıştır. Ayrıca araştırmalarda (Ozoran ve diğerleri, 2012; Çankaya ve diğerleri, 2017; Özcan ve diğerleri, 2017; Oluk ve diğerleri, 2018; Dizman 2018; Erdem, 2018; Vatansever, 2018) Scratch programının bilgi işlemsel düşünme becerisini olumlu etkilediği ifade edilmiştir.

Programlama öğretiminde Kodu Game Lab görsel programlama ortamının kullanıldığı çalışmalara baktığımızda bu araç yardımıyla programlama eğitime tabi tutulan öğrencilerin tutumlarının olumlu yönde değiştiği, aracı kullanma isteklerinin arttığı genel olarak mutlu oldukları (Fowler, 2012); Kodu Game Lab görsel programlama aracının oyun geliştirme açısından katkı sağladığı, öğrencilere problem çözme gibi yetileri kazandırma bakımından yararlı olduğu (Hava, 2016); öğrencilerin programlama öğrenim düzeylerinde ve eğitsel bilgisayar oyunları destekli kodlama öğrenimine yönelik (EBODKÖY) tutumlarında artış olduğu (Dinçer, 2018) sonuçlarına ulaşılmıştır.

Programlama öğretiminde mBlock görsel programlama ortamının kullanıldığı Şimşek (2018)'in çalışmasında da, programlama öğrenim düzeyinin olumlu olarak etkilendiği sonucuna ulaşılmıştır. Yiğit (2016) gerçekleştirdiği çalışmada; Phyton ile

(geleneksel) programlama eğitimi alan öğrencilere nazaran Blockly görsel programlama aracıyla programlama eğitimi alan öğrencilerin programlamayı öğrenmede daha başarılı oldukları sonucuna ulaşmıştır. Ayrıca Kodable, CodeMonkey, Code The Foos, Tynker gibi birçok uygulama da öğrencilerin ilgisini çekerek programlama öğrenmelerini sağlama ve yaratıcılık, problem çözebilme, analitik düşünme vb. gibi 21.yy becerileri olarak adlandırılan kazanımları hedefleyen görsel programlama ortamlarıdır (Aytekin ve diğerleri, 2018).

2.6. Eğitimde Oyunun Yeri ve İlgili Çalışmalar

Kültür, zaman veya mekân fark etmeksizin çocuğun bulunduğu ortamda her zaman oyunun varlığından da söz edilmektedir. Oyunu, çocuk için hayatı öğrenme yolu olarak ifade edebiliriz. Çocuk için oyun esnasında elde edilen bilgiler daha etkili ve daha kalıcıdır. Bu yüzden eğitimde oyun en etkili yollardan biridir. Eğlenerek öğrenmeyi sağlayan oyunlar ile çocuklar aktif olarak öğrenme sürecine katılmaktadır. Demirel (1999) eğitsel oyunu, öğrenilen bilgilerin pekiştirilmesini ve yinelenmesini sağlayan etkinlikler olarak ifade etmiştir. Eğitsel oyunlar ile öğretim öğrenci için eğlenebileceği şekle dönüşmekte, öğrencinin derse olan ilgisinde artış sağlanmaktadır. İlgi çekiciliğinin yanı sıra bu sayede yeni olgular öğrenciye aktarılabilir ve öğrenilenlerin akılda kalma süresi uzatılabilir.

Oyun, çocuğun yaşamının bütününe etkisi olan bir faaliyettir. Çocuklar oyun esnasında öğrenmeyi öğrenmektedirler (Mesleki Eğitim ve Öğretim Sistemini Güçlendirme Projesi [MEGEP], 2007). Çocuklar oyun oynayarak tecrübe edinir ve sürekli olarak düşünür. Oyun, çocuk için etkin ve doğal öğrenme metodudur. Çocuklar oyun sayesinde yaparak ve yaşayarak öğrenebilme tecrübesi kazanırken öğrenme de doğal ve daha kalıcı olarak sağlanmış olur (MEGEP, 2007). Eğitsel oyunlar ise oyun yapısı içerisinde öğrencilerin öğretime ilişkin ders konularını öğrenmesine olanak tanıyan ya da problem çözme yetilerine katkı sağlayan yazılımlar şeklinde ifade edilmektedir (Demirel, Seferoğlu ve Yağcı, 2003). Eğitsel oyunlar normal oyunlardan farklı olarak öğretilecek konuya özgü formal bilgileri içermektedir. Çocuk oyun oynarken keyifli vakit geçirir, informal etkinlik ile formal bilgileri de öğrenmiş olur (Sarıçayır, 2007). Oyuncu oyun oynarken, aynı anda çıkan sorunun cevabını da bulmaya çalışacak ve bu şekilde oyuncuların zekâ gelişimine de katkı sağlanabilecektir.

Piaget'e göre ortaokul 6. sınıf düzeyindeki öğrenciler somut işlemler döneminde bulunmakta ve soyut kavramları öğrenmekte güçlük çekebilmektedir (Karamustafaoğlu ve Kaya, 2013). Henüz soyut düşünemeyen bu öğrencilerin eğitsel bilgisayar oyunları (EBO)

sayesinde kavramlar, karar verme ve iletişim becerileri, problem çözme ve analitik düşünme gibi öğreniminde zorluk yaşanabilecek birtakım becerileri geliştirilebilecekleri dile getirilmektedir (Akpınar, 1999). Literatür çalışmaları, EBO ile yapılan öğrenmelerin öğrencilerde daha kalıcı olduğunu ve öğrenilmesi zor ve karmaşık olan kavramların daha kolay bir şekilde öğrenilebildiğini göstermektedir (Alan, 2017). Ayrıca EBO ortamlarının öğretim esnasında kullanılan diğer yöntem ve tekniklerin destekleyicisi ya da zenginleştiricisi konumunda olduğu da ifade edilmektedir (Çankaya ve Karamete, 2008). Dile getirilen bir başka görüş ise süre dengesini koruyarak dijital oyunları oynamanın, sağlıklı yaşam biçimini desteklediği, oyunların duygusal boşalma ve rahatlama gibi olumlu katkılarının olduğudur (Irmak ve Erdoğan, 2016).

Eğitsel oyunlar, eğlenerek öğrenmek için en uygun bir araç ve eğitim için etkili bir yöntemdir. Bu şekilde bireyler anlatılan konuları daha çabuk kavrar, eğlenerek öğrendiği için eğitim süreci daha sağlıklı, verimli olmaktadır. Böylece bilgiler daha uzun süre hafızada tazeliğini koruyabilir. Bu noktalardan hareketle günümüzdeki eğitimciler, öğrencilerin öğrenme kapasitelerini geliştirmek ve onlara daha iyi öğrenme çevreleri sağlamak amacıyla bilgisayar oyunlarının sahip oldukları potansiyeli dikkate almakta ve bilgisayar oyunlarını sınıf ortamlarında kullanmaktadırlar.

Alan (2017) yaptığı araştırmada; dijital oyun tabanlı yaklaşım kullanılarak hazırlanan eğitsel bir oyunun öğrenciler üzerindeki etkinliğini incelemiştir. Araştırma sonucunda Mesleki ve Teknik Anadolu Lisesinde okuyan öğrencilerin bilgisayar programlama dersine karşı olan tutumlarında olumlu yönde bir artış gözlemlenmiştir. Ayrıca algoritma oluşturma konusunda sıkıntı yaşayan ve programlamanın mantığını anlamadığını söyleyen öğrencilerin eğitsel oyunu oynadıktan sonra bu problemleri çözebildiklerini ifade etmiş, yazılım geliştirme süreçlerinin öğretimi için bu tür oyun tabanlı yaklaşımların oldukça etkili olduğunu belirtmişlerdir. Alan ve Taşdemir (2017), dijital tabanlı eğitsel bir bilgisayar oyunu hazırlayarak uygulamalar yapmış ve araştırma sonucunda hazırlanan oyunun öğrencilerin derse karşı olan tutumlarını olumlu yönde artırdığını belirlemişlerdir. Gerçekleştirilen bir diğer çalışmada (Yue ve Wan, 2015); programlamaya ilişkin kavramların öğrenimi için dijital oyunların kullanılabilir, çoklu ortam içeriği açısından yeterli, pratik, eğlendirici ve kullanım kolaylığı açısından uygun olduğu sonucuna ulaşılmıştır. Ayrıca yapılan araştırma, akademik başarı açısından da dijital oyunların faydalı bir öğrenme aracı olduğunu ortaya koymuştur.

Savaş (2014) gerçekleştirdiği çalışmada; oyunla öğretim yönteminin öğrenci başarısı ve kalıcılığı üzerinde olumlu etkisi olduğu sonucuna ulaşmıştır. Ayrıca günlükler üzerinde yapılan nitel analiz sonucunda, oyun oynanırken öğrencilerin heyecanlı ve mutlu oldukları, genel olarak çok eğlendikleri tespit edilmiştir. Yanı sıra kullanılan eğitsel oyunların, dersin ve ilgili öğretmenin daha çok sevilmesini sağladığı, öğrencilerin öğrenimine pozitif etki ettiği sonuçlarına da ulaşılmıştır. Yıldırım (2015) gerçekleştirdiği çalışmada; öğrenmenin düzeyine ve kalıcılığına eğitsel oyun ve dönüt-düzeltilme kullanımının etkisini incelenmiştir. Araştırma neticesinde eğitsel oyun yoluyla öğretimin, mevcut öğretim programına göre öğrencilerin öğrenme düzeylerini arttırmada ve edinilen bilgilerin kalıcılığı hususunda daha etkili olduğu sonucuna ulaşmıştır. Koka (2018) ise çalışmada, öğrencilerin ders başarısına bilgisayar destekli eğitsel oyunların etkisini incelemiştir. Araştırmada elde edilen bulgulara bakıldığında öğrencilerin akademik başarılarını ve edinilen bilgilerin kalıcılığını, kullanılan bilgisayar destekli eğitsel oyunların arttırdığı tespit edilmiştir.

Bayırtepe ve Tüzün (2007) araştırmalarında; EBO ile desteklenen oyun tabanlı öğrenme ortamlarının öğrencilerin kaygı düzeylerini azalttığını; Kablan (2010) çalışmada; amacına uygun ve yeterli düzeyde hazırlanmış olan EBO ortamlarının öğrencilerin öğrenme seviyesini yükselttiğini ve dersin sıkıcılığı düşürdüğünü belirtmişlerdir. Aslan (2014) ise çalışmada öğrencilere; ilköğretim matematik dersi müfredatında yer alan olasılık problemleri konusunu, Scratch yazılımını kullanarak tasarladığı eğitsel oyunla öğretmiş, öğrencilerin olasılık problemlerini rahatlıkla anladığını ve öğrenirken bir taraftan da eğlenerek derse olan ilgilerinin arttığını tespit etmiştir.

BÖLÜM III

YÖNTEM

Bu bölümde araştırmacının modeli ve denenceleri, araştırmanın yöntemi, çalışma alanı, çalışma grupları, aşamaları ve veri toplama araçları, verilerin toplanması ve verilerin analizine yönelik açıklamalara yer verilmiştir.

3.1. Araştırma Modeli

Code.org platformu kullanımının öğrencilerin programlama öğrenim düzeylerine, programlama öğrenimlerinin kalıcılık düzeylerine ve EBODKÖY tutumlarına olan etkisini ölçmek amacıyla araştırma kapsamında deneme modellerinden gerçek deneme modeli kullanılmıştır. Deneme modelleri, sebep-sonuç ilişkilerini tespit etmeye yönelik, direkt olarak araştırmacının denetiminde, çalışma içerisinde gözlenmesi planlanan verilerin oluşturulduğu araştırma modelleridir (Karasar, 2011). Bu tip araştırmalarda araştırmacı farklı uygulamalar yapar ve bağımsız değişkenin kontrolünü elinde tutarak bağımlı değişkende meydana gelen değişiklikleri gözlemler (Sümer, Demirutku ve Özkan, 2005). Deneme modelleri, Campbell ve Stanley tarafından “deneme öncesi modeller”, “gerçek deneme modelleri” ve “yarı deneme modelleri” olmak üzere üç başlıkta ele alınmaktadır (Karasar, 2011).

Çalışma kapsamında gerçek deneme modellerinden “ön test - son test kontrol gruplu desen” kullanılmıştır. Ön test - son test kontrol gruplu araştırma modelinde, tarafsız atama ile bir araya getirilmiş iki grup yer almaktadır. Bu gruplardan biri kontrol, diğeri deney grubu olarak adlandırılır. İki gruba da uygulama öncesi ve uygulama sonrası ölçme işlemleri uygulanır. Modelde ön testlerin yer alması, grupların deneysel işlem öncesi benzerlik oranlarının bilinmesine ve elde edilen son test sonuçlarının buna göre revize edilmesine yardım etmektedir. Bu modelde bağımsız değişkenin hangi oranda etkili olduğunu tespit etmek için ön test ve son test ölçüm sonuçları birlikte kullanılmaktadır. Bu amaçla, öncelikle ön test puanları karşılaştırılmakta, arada anlamlı bir farkın olmamasıyla birlikte son test puanlarına bakılarak ortalamalar arası ayırım sınanmaktadır. Ancak, deneysel işlem öncesi yapılan ölçmenin anlamlı derecede birbirinden farklı olması, sonrasında ulaşılan bulgulara yapılacak karşılaştırmaların yorumlanmasını güçleştirebilecektir (Karasar, 2014).

Bu araştırma kapsamında öğrenciler, rastgele olarak gruplara yerleştirilmiş, gruplardan biri kontrol, diğeri deney grubu olarak isimlendirilmiştir. Araştırma süresince deney grubuna 6. sınıf BTY dersi öğretim programındaki “Problem Çözme ve Programlama” ünitesi Code.org platformu kullanılarak, kontrol grubunda ise normal öğretim yöntemleri kullanılarak işlenmiştir.

3.1.1. Değişkenler ve deneysel desen

Bu çalışmada “Code.org platformu ile programlama öğretimi” ve “Normal yöntemler ile programlama öğretimi” olmak üzere iki bağımsız değişken bulunmaktadır. Bağımsız değişken, araştırmacı tarafından ortama katılarak bağımlı değişkenin aldığı değeri etkileyen, çalışmanın sonucuna etkisi bulunan değişken olarak ifade edilebilir (Büyüköztürk, 2017). Araştırmada öğrencilerin “Programlama Öğrenim Düzeyleri”, “Programlama Öğrenimlerinin Kalıcılık Düzeyleri” ve “Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutumları” olmak üzere 3 bağımlı değişken bulunmaktadır. Bağımlı değişken, araştırmanın neticesi olarak kabul edilen, süreç içerisinde ortamda bulunan bir durumdan etkilenecek değişik sonuçlar alabilen değişken olarak ifade edilebilir (Büyüköztürk, 2017). Ön test - son test kontrol gruplu desen Tablo 1’de verildiği gibi gösterilmiştir.

Tablo 1.

Deneysel işlem düzeni

Gruplar	Ölçme	İşlem	Ölçme	İzleme
Deney Grubu (DG)	EBODKÖYTÖ PÖDA	Code.org Platformu ile Programlama Öğretimi	EBODKÖYTÖ PÖDA CPGF	PÖDA
Kontrol Grubu (KG)	EBODKÖYTÖ PÖDA	Normal Yöntemler ile Programlama Öğretimi	EBODKÖYTÖ PÖDA	PÖDA

* **CPGF:** Code.org Platformuna Yönelik Yarı Yapılandırılmış Görüşme Formu

* **PÖDA:** Programlama Öğrenim Düzeyi Belirleme Aracı

* **EBODKÖYTÖ:** Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeği

3.2. Araştırma Grubu

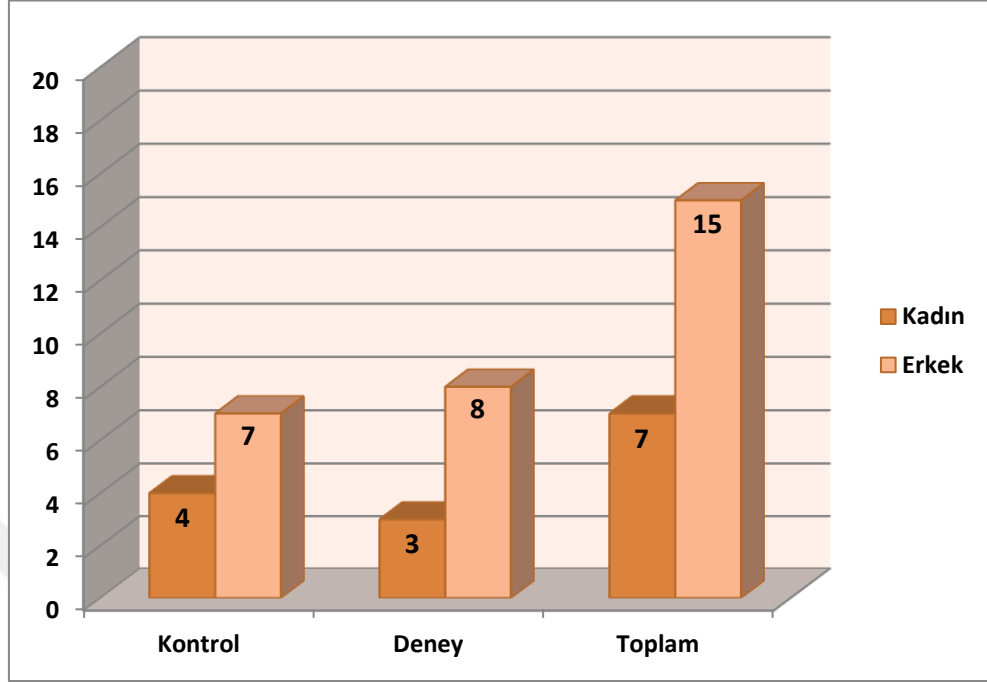
Bu araştırmada deneysel desen kullanılması planlandığından dolayı evren ve örneklem yerine çalışma grubu hakkında bilgi verilmesi daha uygun görülmektedir. Araştırmanın çalışma grubunu 2016-2017 öğretim yılının ikinci döneminde, Kütahya ili Gediz ilçesi Altinkent Ortaokulunda öğrenim gören 22 altıncı sınıf öğrencisi oluşturmaktadır. Gerçek deneme modeli çerçevesinde rastgele olarak deney ve kontrol grupları belirlenmiştir. Deney grubu 8'i erkek, 3'ü kız olmak üzere toplam 11 öğrenciden; kontrol grubu ise 4'ü kız, 7'si erkek olmak üzere toplam 11 öğrenciden oluşmaktadır. Çalışma grubunda yer alan öğrencilerin cinsiyet bilgileri Tablo 2'de verildiği gibidir.

Tablo 2.

Öğrencilerin cinsiyetlerine göre dağılımları

	Kontrol Grubu		Deney Grubu		Toplam	
Cinsiyet	n	%	n	%	n	%
Kadın	4	36.36	3	27.27	7	31.82
Erkek	7	63.64	8	72.73	15	68.18
Toplam	11	100	11	100	22	100

Tablo 2'de görüldüğü gibi, çalışma toplam 22 öğrenci ile gerçekleştirilmiş olup bu öğrencilerden %31.82'si kadın, %68.18'i ise erkektir. Deney grubunda yer alan öğrencilerin %27.27'si kadın, %72.73'ü ise erkektir. Kontrol grubunda yer alan öğrencilerin ise %36.36'sı kadın, %63.64'ü erkektir. Cinsiyetlere göre dağılım grafiği Şekil 23'te verilmiştir.



Şekil 23. Cinsiyetlere göre dağılım grafiği

Çalışma grubunda yer alan 22 öğrencinin tamamı 12 yaşındadır. Ayrıca çalışmaya katılan 22 öğrencinin hiçbirisi herhangi bir kodlama eğitimi içerisinde yer almamıştır.

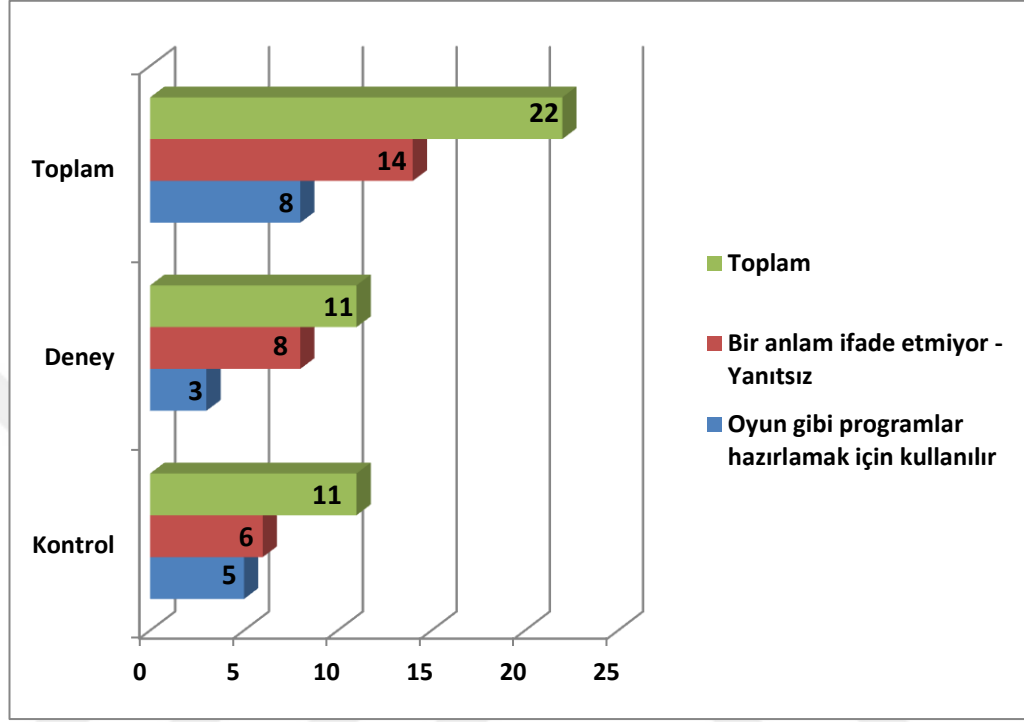
Tablo 3.

Öğrencilerin programlamaya ilişkin kavramların ne ifade ettiğine ilişkin betimsel istatistik sonuçları

Verilen Cevaplar	Kontrol Grubu		Deney Grubu		Toplam	
	n	%	n	%	n	%
Oyun gibi programlar hazırlamak için kullanılır	5	45.45	3	27.27	8	36.36
Bilmiyorum - Yanıtsız	6	54.55	8	72.73	14	63.64
Toplam	11	100	11	100	22	100

Çalışma grubunda yer alan öğrencilere programlamaya ilişkin kavramların ne ifade ettiğine dair bilgiler Tablo 3'te gösterilmiştir. Görüldüğü üzere, çalışmaya katılan 22 öğrencinin %36.36'sı "Oyun gibi programlar hazırlamak için kullanılır" yanıtını verirken

%63.64'ü ise yanıt vermemiş veya “bir anlam ifade etmiyor” yorumlarında bulunmuşlardır (Şekil 24).



Şekil 24. Programlamaya ilişkin kavramların ne ifade ettiğine dair grafik

Elde edilen verilere dayalı olarak; Tablo 4'teki Mann Whitney U testi sonuçlarına bakıldığında deney ve kontrol grubu öğrencilerinin ön test olarak uygulanan EBODKÖYTÖ puan ortalamaları arasında .05 düzeyinde anlamlı bir fark bulunamamıştır ($U=56.500$, $p>.05$).

Tablo 4.

EBODKÖYTÖ ön test uygulama sonuçlarının gruplara göre U testi sonuçları

Grup	N	Sıra Ortalaması	Sıra Toplamı	U	z	p
Kontrol	11	11.86	130.50	56.500	-.263	.793
Deney	11	11.14	122.50			
Toplam	22					

Tablo 5'teki Mann Whitney U testi sonuçlarına bakıldığında deney ve kontrol grubunda yer alan öğrencilerin ön test olarak uygulanan PÖDA ölçeği puanları arasında .05 düzeyinde anlamlı bir fark bulunamamıştır ($U=54.500$, $p>.05$).

Tablo 5.

PÖDA ön test uygulama sonuçlarının gruplara göre U testi sonuçları

Grup	N	Sıra Ortalaması	Sıra Toplamı	U	z	p
Kontrol	11	12.05	132.50	54.500	-.661	.509
Deney	11	10.95	120.50			
Toplam	22					

Code.org platformunun kullanıldığı deney grubu öğrencileri ile normal öğretim yönteminin uygulandığı kontrol grubu öğrencilerinin EBODKÖY tutumları ve programlama öğrenim düzeyleri arasında anlamlı farklılık bulunmaması alt problemlerin yorumlanması açısından önemlidir.

3.3. Veri Toplama Araçları ve Verilerin Toplanması

Araştırmada ölçme aracı olarak Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik (EBODKÖY) Tutum Ölçeği, Programlama Öğrenim Düzeyi Belirleme Aracı (PÖDA) ve öğrencilerin Code.org platformuna yönelik görüşlerinin belirlendiği yarı yapılandırılmış görüşme formu (CPGF) kullanılmış ve bu araçlar üzerinden veri toplama işlemi gerçekleştirilmiştir.

3.3.1. Eğitsel bilgisayar oyunları destekli kodlama öğrenimine yönelik tutum ölçeği (EBODKÖYTÖ)

EBODKÖYTÖ, Keçeci ve diğerleri (2016) tarafından geliştirilmiştir (Ek-3). Bu ölçme aracı ile ortaokul seviyesindeki öğrencilerin, EBO destekli kodlama öğrenimine ilişkin tutumlarının tespitinde kullanılabilecek bir veri toplama aracı geliştirilmesi hedeflenmiştir. Çalışmalar sonucunda 22 olumlu 6 olumsuz maddeden oluşan 28 maddelik veri toplama aracı oluşturulmuştur. Likert tipteki bu ölçeğin KMO değeri .84, kikare değeri anlamlı ($\chi^2_{(630)}=3087.798$; $p<.01$), güvenirlik katsayısı .833 olarak tespit edilmiştir. Toplam %41.39 oranında bileşenlerin varyansa katkısı tespit edilmiştir. EBODKÖYTÖ ölçme

aracında katılımcıların cinsiyetlerinin, yaşlarının ve sınıf düzeylerinin sorulduğu maddeler de bulunmaktadır.

Tutum ölçekleri negatif ve pozitif olarak iki kutuplu ve Likert tipteki veri toplama araçlarıdır. Tutum maddelerinin doğru veya yanlış bir yanıtı yoktur, dolayısıyla puanlamada bir doğru cevap anahtarı da söz konusu değildir. Puanlama, tutum ifadesine katılma derecelerine (tercihlerine) göre, anlamı negatif olan cümleler için 1’den 5’e doğru; pozitif olan maddeler için 5’ten 1’ e doğru puanlanır.

3.3.1.1. EBODKÖYTÖ verilerinin toplanması

EBODKÖYTÖ, deney ve kontrol grubunda yer alan öğrencilerin uygulama öncesi ve sonrası tutum seviyelerini tespit etmek amacıyla, gerçekleştirilecek öğretimden önce (Şubat 2017) ve sonra (Nisan 2017) olmak üzere iki defa sınıf ortamında uygulanmıştır. Elde edilen anketlerin sonuçları SPSS programına aktarılmıştır.

3.3.2. Programlama öğrenim düzeyi belirleme aracı (PÖDA)

Programlama Öğrenim Düzeyi Belirleme Aracı (PÖDA) kapsam geçerliliğini sağlamak amacıyla uzman görüşü alınarak hazırlanmıştır. PÖDA ölçme aracında kodlamaya öğrenimine yönelik tarama soruları da yer almaktadır. Şekil 25’te yer alan kazanımlara yönelik olarak hazırlanan ölçme aracı Çanakkale ve Kocaeli illerindeki iki ortaokulda yapılan pilot uygulamalar sonrası uzman görüşleri (Ek-6) alınarak revize edilmiştir. Tekrar pilot uygulamaya tabi tutularak uzman görüşleri alınmış, 10 soru ve yanıtları şeklinde hazır hale getirilmiştir (Ek-5).

PÖDA veri toplama aracı araştırma içerisinde, uygulama öncesi, uygulama sonrası ve kalıcılık seviyesini belirlemeye yönelik izleme aşaması olmak üzere; deney ve kontrol grubunda yer alan öğrencilerin grup içi ve gruplar arası programlama öğrenim düzeyleri arasında anlamlı bir farklılık olup olmadığını belirlemek amacıyla hazırlanmıştır.

Soru No	İlgili Kazanım
1	Belirlenen problemin çözümüne farklı yollardan ulaşır.
2	Çözümü verilen probleme farklı bir çözüm yolu önerir.
3	Belirlenen problemin çözümü için adımlar oluşturur.
4	Problem çözümü için geliştirdiği adımların geçerliğini sorgulayarak en etkili çözüme ulaşır.
5	Belirlenen problem için oluşturduğu çözüm önerisini ve yaklaşımını sunar.
6	Belirlenen problemin çözümü için algoritma geliştirmenin önemini ifade eder.
7	Farklı algoritmaları inceleyerek en hızlı ve doğru çözümü seçer.
8	Problem çözme sürecinde strateji geliştirmenin önemini açıklar.
9	Strateji geliştirmek için gereksinim duyduğu bilgiye erişir.
10	Eriştiği bilgiyi, strateji geliştirmeye uygunluk açısından değerlendirir.
11	Hatalı bir algoritmayı doğru çalışacak biçimde düzenler.
12	Farklı kaynaklardan edindiği bilgileri, geliştirdiği strateji kapsamında kullanmak üzere dönüştürür.
13	Belirlenen problemin çözüm sürecinde gerekli işlemler için akış şeması hazırlar.
14	Günlük hayatta karşılaşılan problemleri çözmek için farklı stratejiler geliştirir.
15	Belirlenen problemin çözümünü sağlayan farklı algoritmalar hakkında tartışır.
16	Geliştirdiği stratejilerin, belirlenen problemin çözümündeki etkisini tartışır.

Şekil 25. PÖDA ölçme aracının ilgili olduğu kazanımlar

3.3.2.1. PÖDA verilerinin toplanması

Programlama Öğrenim Düzeyi Belirleme Aracı, deney ve kontrol grubunda yer alan öğrencilerin programlama öğrenim düzeylerini belirlemek amacıyla, ön test (Şubat 2017), son test (Nisan 2017) ve öğrencilerin bilgilerinin kalıcılığını belirlemek için son testten beş

hafta sonra izleme testi (Mayıs 2017) olmak üzere üç defa sınıf ortamında uygulanmıştır. Elde edilen sonuçlar SPSS programına aktarılmıştır.

3.3.3. Code.org platformuna yönelik yarı yapılandırılmış görüşme formu (CPGF)

Nitel araştırma kapsamında değerlendirilen yarı yapılandırılmış görüşme ölçeği belirli miktarda standartlığı ve esnekliği aynı anda içerisinde barındırarak eğitim bilim çalışmalarında daha kullanılabilir bir yöntem konumundadır (Ekiz, 2003). Bu tekniğin araştırmacıya kazandırdığı en önemli rahatlık görüşmenin daha önceden belirlenmiş çerçevesine bağlı kalarak devam etmesi nedeniyle daha planlı ve karşılaşılabılır bilgi sunacak olmasıdır (Yıldırım ve Şimşek, 1999). Çalışmanın amacı doğrultusunda, uzman görüşü alınarak araştırmacı tarafından Code.org Platformuna Yönelik Yarı Yapılandırılmış Görüşme Formu (CPGF) hazırlanmıştır. Bu veri toplama aracının hazırlanma aşamasında Çanakkale ve Kocaeli illerindeki iki ortaokulda öğrenim gören öğrenciler ile pilot uygulama gerçekleştirilmiştir. Pilot uygulamanın ardından, ölçeğin kapsam geçerliğinin sağlanması ve daha anlaşılır hale getirilmesine yönelik uzman görüşleri alınarak görüşme formu hazır hale getirilmiştir. Son haliyle 10 sorudan oluşan görüşme formu, deney grubu öğrencilerinin deneysel işlem sonrasında Code.org platformuna yönelik görüşlerini tespit etmek amacıyla uygulamaya hazır hale gelmiştir (Ek-7).

3.3.3.1. CPGF verilerinin toplanması

Ölçme işlemine tabi tutulacak kişinin onay vermesi ve imkânların uygun olması halinde, işlem esnasında fiziki cihazlarla gerçekleştirilen sürekli kayıtlar, en sağlıklı kayıt yöntemleridir. Bu sayede hem görüşme süreci kesintiye uğramamış hem de görüşme esnasında kayıt alınmasından dolayı görüşme sonuçlarının daha sonra da detaylı olarak incelenebilmesine yönelik bir belge elde edilmiş olur (Kahn ve Cannel, 1957; Borg, 1963; Kerlinger, 1964; Madge, 1965; Rummel, 1970; Özgüven, 1980 akt: Karasar, 2004). Buradan hareketle öğrencilerin CPGF veri toplama aracında yer alan sorulara verdikleri yanıtlar ses kayıt aracıyla kaydedilmiş, bu sayede toplanan verilerin yanlış veya eksik olma ihtimaline büyük oranda engel olunmuştur. CPGF veri toplama aracıyla toplanan veriler 2017 yılı Nisan ayında, bilişim teknolojileri sınıfında deney grubunu oluşturan 11 öğrenci ile birebir yapılan görüşmeler yoluyla elde edilmiştir. Görüşme esnasında öğrenciler soruları yanıtlarken araştırmacıdan etkilenmemesine dikkat edilmiş ve görüşme süresi ortalama 8 dakika 53 saniye sürmüştür (Tablo 6). Kayıt altına alınan görüşmelerden, bir alan uzmanıyla paralel

olarak yazılı dökümler oluşturulmuş ve karşılaştırma yapılarak elde edilen verilerin güvenilirliği sağlanmıştır.

Tablo 6.

Deney grubu görüşme süreleri

Öğrenciler	Süre (dk)
Öğrenci 1	08:31
Öğrenci 2	07:38
Öğrenci 3	10:04
Öğrenci 4	08:25
Öğrenci 5	07:47
Öğrenci 6	09:24
Öğrenci 7	08:35
Öğrenci 8	08:39
Öğrenci 9	11:05
Öğrenci 10	08:42
Ortalama	08:53

3.4. Uygulama Süreci

Bu araştırma 2016-2017 eğitim-öğretim yılının ikinci döneminde, MEB'e bağlı okullardan Kütahya ilinin Gediz ilçesindeki Altınkent Ortaokulu'nda gerçekleştirilmiştir. Çalışma içerisindeki deneysel işlem, ortaokul 6. sınıf öğrencileriyle BTY dersi kapsamındaki 20 ders saati süresince yürütülmüştür. Programlama öğretimine başlamadan önce öğrencilere ön-test olarak EBODKÖYTÖ ve PÖDA veri toplama araçları uygulanmıştır.

3.4.1. Code.org platformu kullanımı ve yapılan ortak işlemler

Gerçekleştirilen programlama öğretiminden önce deney grubu için Code.org platformunun kullanıma hazır hale getirilmesinden ve genel olarak her derste yapılan ortak işlemlerden bahsedilecektir. Öğrencilerin kullanımından önce ilk olarak platforma öğretmen

üyeliği ile giriş yapıldı. Öğrencilerin programlama öğrenimine yönelik “Hızlandırılmış Bilgisayar Bilimlerine Giriş Kursu” planlandı ve öğrenci bilgileri girilerek sanal sınıf oluşturuldu (Şekil 26).

ALTINKENT ORTAOKULU - 6/A

Bölüm değiştir: 6. Sınıflar 2017/2018 - Altır ▼

İlerleme Durumu	Metin Cevaplar	Değerlendirmeler/Anketler	Projeler	İstatistikler	Öğrencileri Yönet
-----------------	----------------	---------------------------	----------	---------------	--------------------------

Birden fazla öğrenci Ekle

Öğrencileri taşı

İsim	Yaş	Cinsiyet	Gizli	Tümünü Kaydet
Öğrenci adı (zorunlu)	▼	▼	Otomatik olarak oluşturulan	Ekle
ÖĞRENCİ 1	12 ▼	Erkek ▼	Otomatik olarak oluşturulan	Kaydet İptal
ÖĞRENCİ 2	12 ▼	Erkek ▼	Otomatik olarak oluşturulan	Kaydet İptal
ÖĞRENCİ 3	12 ▼	Kadın ▼	Otomatik olarak oluşturulan	Kaydet İptal
ÖĞRENCİ 4	12 ▼	Erkek ▼	Otomatik olarak oluşturulan	Kaydet İptal
ÖĞRENCİ 5	12 ▼	Erkek ▼	Otomatik olarak oluşturulan	Kaydet İptal
ÖĞRENCİ 6	12 ▼	Erkek ▼	Otomatik olarak oluşturulan	Kaydet İptal
ÖĞRENCİ 7	12 ▼	Erkek ▼	Otomatik olarak oluşturulan	Kaydet İptal
ÖĞRENCİ 8	12 ▼	Kadın ▼	Otomatik olarak oluşturulan	Kaydet İptal
ÖĞRENCİ 9	12 ▼	Kadın ▼	Otomatik olarak oluşturulan	Kaydet İptal
ÖĞRENCİ 10	12 ▼	Erkek ▼	Otomatik olarak oluşturulan	Kaydet İptal
ÖĞRENCİ 11	12 ▼	Erkek ▼	Otomatik olarak oluşturulan	Kaydet İptal

Şekil 26. Sanal sınıf öğrenci listesi

Öğrencilerin platforma girerken kullanacakları internet adresi, kullanıcı adı ve şifre bilgileri çıktı alındı ve öğrencilere dağıtıldı (Şekil 27). Öğrenciler verilen yönerge ile platforma girdi, isimlerinin üzerine gelerek şifre sözcüklerini yazdı ve oturumlarını başlattı (Şekil 28).

URL https://studio.code.org/sections/FQWLZN İsim ÖĞRENCİ 1 Gizli Sözcük company rule	URL https://studio.code.org/sections/FQWLZN İsim ÖĞRENCİ 2 Gizli Sözcük from noun
URL https://studio.code.org/sections/FQWLZN İsim ÖĞRENCİ 3 Gizli Sözcük travel century	URL https://studio.code.org/sections/FQWLZN İsim ÖĞRENCİ 4 Gizli Sözcük whole ready
URL https://studio.code.org/sections/FQWLZN İsim ÖĞRENCİ 5 Gizli Sözcük save make	URL https://studio.code.org/sections/FQWLZN İsim ÖĞRENCİ 6 Gizli Sözcük animals wish
URL https://studio.code.org/sections/FQWLZN İsim ÖĞRENCİ 7 Gizli Sözcük than entire	URL https://studio.code.org/sections/FQWLZN İsim ÖĞRENCİ 8 Gizli Sözcük expect really

Şekil 27. Code.org platformu öğrenci girişi bilgileri

ALTINKENT ORTAOKULU - 6/A bölümüne hoşgeldiniz

Adınızı seçin*

ÖĞRENCİ 1

ÖĞRENCİ 10

ÖĞRENCİ 11

ÖĞRENCİ 2

ÖĞRENCİ 3

ÖĞRENCİ 4

ÖĞRENCİ 5

ÖĞRENCİ 6

ÖĞRENCİ 7

ÖĞRENCİ 8

ÖĞRENCİ 9

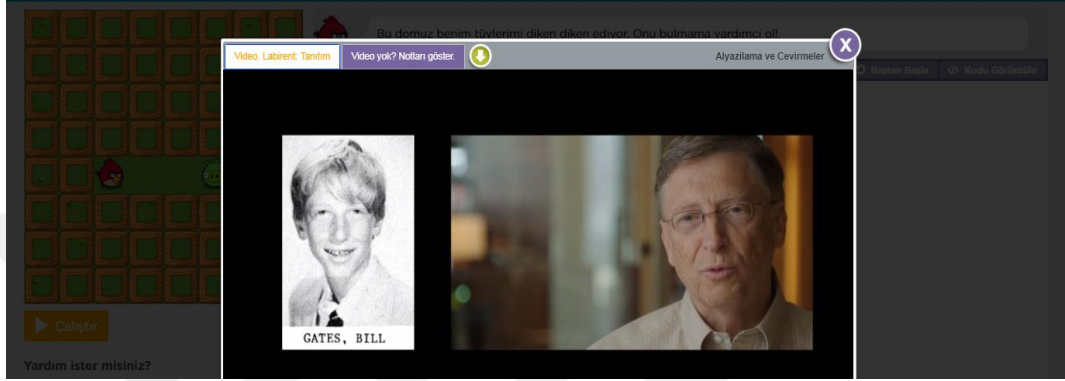
*Adınızı neden görmediğiniz ile ilgili bilgi [burada](#).

Giriş Yap

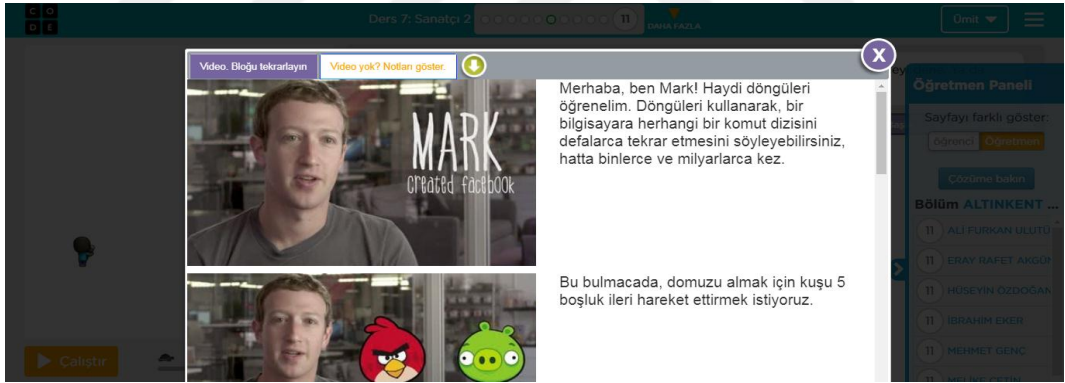
Şimdi gizli kelimelerinizi girin

Şekil 28. Öğrenci giriş ekranı

Daha sonra Code.org platformu arayüzü ve kullanıma ilişkin site haritası öğrencilere tanıtıldı. Menüler ve kod blokları gösterildi, nasıl kullanmaları gerektiği anlatıldı. Ardından derse geçildi ve öğrenciler geçmek istedikleri bölüm öncesi bilgilendirme videosunu izledi (Şekil 29), ilgili notları okudu (Şekil 30), ipucu satırını inceleyerek deneme-yanılma metoduyla görevini gerçekleştirdi (Şekil 31).



Şekil 29. Bölüm ipucu video ekranı

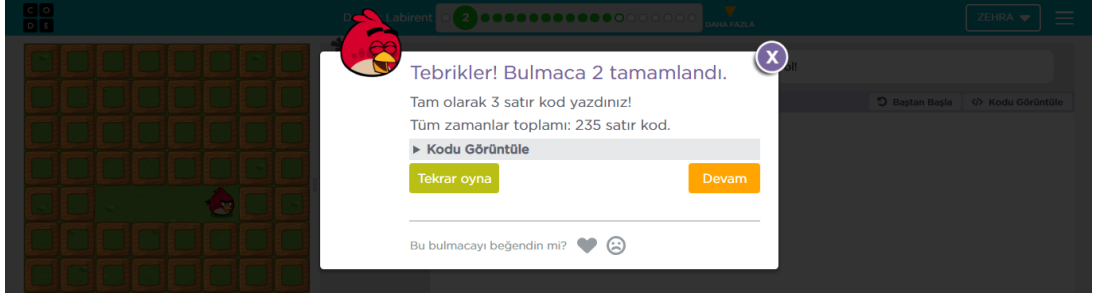


Şekil 30. Video notları

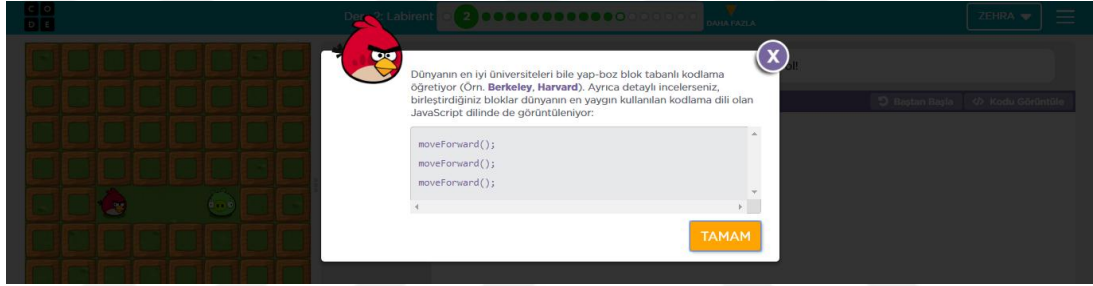


Şekil 31. Bölüm ekranı

Bölümün tamamlanmasının ardından öğrenci tarafından bölüme ilişkin oluşan kodlar görüntülenebildi (Şekil 33).

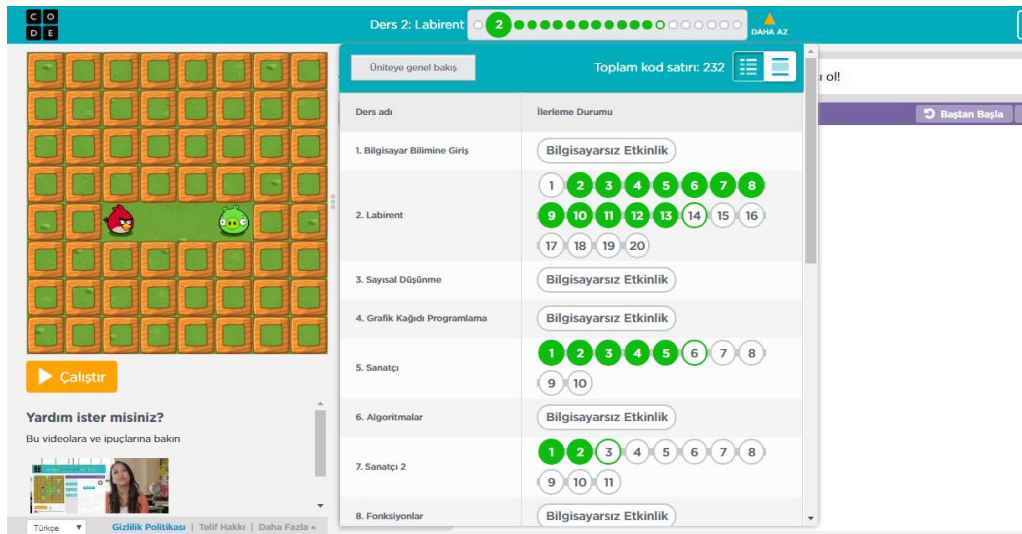


Şekil 32. Bölüm geçiş ekranı



Şekil 33. Kod görüntüleme ekranı

Kodların görüntülenmesinin ardından öğrenci tarafından genel ilerleme durumu izlenerek bölüm geçişi sağlandı.



Şekil 34. Öğrencinin bölümler arası geçiş ekranı

Hızlandırılmış Bilgisayar Bilimlerine Giriş Kursu

Bu 20 saatlik ders temel bilgisayar bilimi ve programlama kavramlarını tanıtır. Ders, K-8 dersliklerinde kullanılmak üzere tasarlanmıştır, ama her yaştan öğrenmek için eğlencelidir.

Devam

Yardım Al

Ders adı	İlerleme Durumu
1. Bilgisayar Bilimine Giriş	Bilgisayarsız Etkinlik
2. Labirent	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
3. Sayısal Düşünme	Bilgisayarsız Etkinlik
4. Grafik Kağıdı Programlama	Bilgisayarsız Etkinlik
5. Sanatçı	1 2 3 4 5 6 7 8 9 10
6. Algoritmalar	Bilgisayarsız Etkinlik
7. Sanatçı 2	1 2 3 4 5 6 7 8 9 10 11
8. Fonksiyonlar	Bilgisayarsız Etkinlik
9. Çiftçi	1 2 3 4 5 6 7 8 9 10 11
10. Koşullar	Bilgisayarsız Etkinlik
11. Sanatçı 3	1 2 3 4 5 6 7 8 9 10 11

Şekil 35. Öğrencinin genel durum takip ekranı

Öğretmen Ana Sayfa > Öğrenci hesapları ve ilerleme

Yeni bölüm

Bölüm	Oturum açma türü	Sınıf	Kurs	Ektralar	Öğrenciler	Bölge Kodu	
ALTINKENT ORTAOKULU - 6/A İlerleme Durumu Öğrencileri Yönet	word	6	Hızlandırılmış Kurs	Evet	11	FWWLZN	Düzenle Sertifika yazdır

Şekil 36. Grup ayarları ekranı

Öğretmen girişi ile sürecin sürekli takibi sağlandı. İlerlemeleri izlenerek gerektiği takdirde dönütler verildi. Kullanım istatistikleri incelenerek öğrencilere tamamladıkları seviyeler ve yazmış oldukları kodlar hakkında bilgilendirmeler yapıldı (Şekil 38).

Bir kurs ya da ünite seçin

Hızlandırılmış Bilgisayar Bilimlerin

Derse geçin

5: Sanatçı

[Kursu incele](#)

Ders																					3
Seviye türü																					
ÖĞRENCİ 1	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	Bilgisayarsız		
ÖĞRENCİ 10	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	Bilgisayarsız		
ÖĞRENCİ 11	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	Bilgisayarsız		
ÖĞRENCİ 2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	Bilgisayarsız		
ÖĞRENCİ 3	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	Bilgisayarsız		
ÖĞRENCİ 4	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	Bilgisayarsız		
ÖĞRENCİ 5	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	Bilgisayarsız		
ÖĞRENCİ 6	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	Bilgisayarsız		
ÖĞRENCİ 7	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	Bilgisayarsız		
ÖĞRENCİ 8	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	Bilgisayarsız		
ÖĞRENCİ 9	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	Bilgisayarsız		

Şekil 37. Öğrenci sanal sınıf izleme ekranı

ALTINKENT ORTAOKULU - 6/A

Bölüm değiştir: 6. Sınıflar 2017/2018 - Altın

İlerleme Durumu

Metin Cevaplar

Değerlendirmeler/Anketler

Projeler

İstatistikler

Öğrencileri Yönet

İsim	Tamamlanan seviyeler	Kod satırları
ÖĞRENCİ 1	56	829
ÖĞRENCİ 10	52	411
ÖĞRENCİ 11	35	363
ÖĞRENCİ 2	52	495
ÖĞRENCİ 3	35	235
ÖĞRENCİ 4	28	183
ÖĞRENCİ 5	41	464
ÖĞRENCİ 6	39	394
ÖĞRENCİ 7	43	541
ÖĞRENCİ 8	30	331
ÖĞRENCİ 9	29	246

Şekil 38. Öğrenci kodlama istatistik ekranı



Şekil 39. Öğrenci uygulama takip ekranı

Sürecin ardından kursu tamamlayan öğrencilere sertifikaları verildi (Şekil 40).



Şekil 40. Eğitim sonrası verilen sertifika örneği

3.4.2. DG ve KG uygulama süreci

Programlama öğretimine iki gruba da benzer şekilde Algoritma konusu işlenerek başlanmıştır. Öğrencilerden hayatın içinde yer alan olayları sırasıyla adım adım düşünerek algoritma haline dönüştürmeleri istenmiştir. Derste yer verilen bazı örnekler Şekil 41’de gösterilmiştir.

Adım 1: Kalem tıraşı çıkar.

Adım 2: Kalemi al.

Adım 3: Çöp kovasının yanına git.

Adım 4: Kalemin ucunu aç.

Adım 5: Sırana geri dön.

Adım 6: Yazmaya devam et.

Adım 1: Başla

Adım 2: Yoğurdu kaba koy.

Adım 3: Su ekle.

Adım 4: Çırp.

Adım 5: Tuz koy.

Adım 6: Bardağa doldur.

Adım 7: Bitir.

Şekil 41. DG-KG algoritma örnekleri

Daha sonra her iki grupta da Akış Şemaları konusu işlenmiştir. Öncelikle semboller tanıtılmıştır (Şekil 52).

Simge	İşlev
	Başla/Bitir
	Giriş
	Atama/İşlem
	Denetim (Karar)
	Çıkış
	Döngü
	Akış Yönü

Şekil 42. Akış şeması sembolleri

Daha sonraki süreçte deney grubuyla gerçekleştirilen öğretimde öğrencilerden Code.org platformu üzerinden yaptıkları blok programlama örneklerinin Akış Şemalarını oluşturmaları istenmiştir. Derste yer verilen bir örnek Şekil 43'te gösterilmiştir.



Şekil 43. DG örnek akış şeması

Kontrol grubuyla gerçekleştirilen öğretimde de öğrencilerden algoritmalarını yazdıkları örneklerin Akış Şemalarını oluşturmaları istenmiştir. Derste yer verilen bir örnek Şekil 44'te gösterilmiştir.



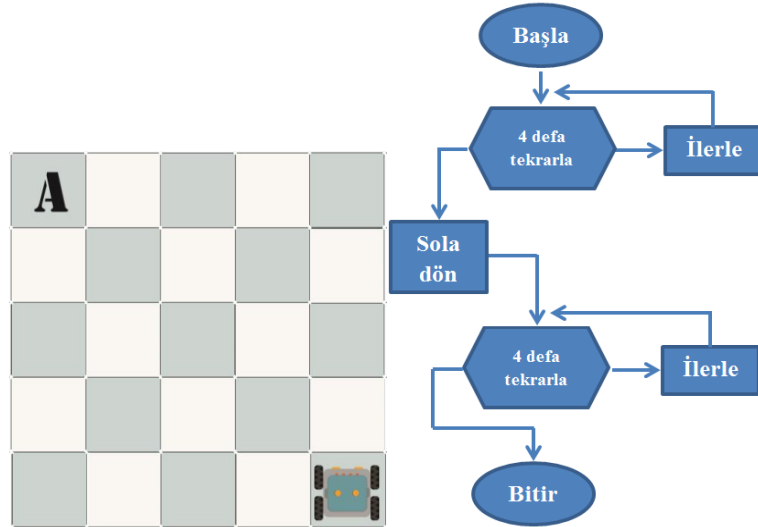
Şekil 44. KG örnek akış şeması

Daha sonra deney gurubuyla programlama dilinde döngüler konusuna geçilmiştir. Programlama içerisinde tekrar eden davranışlara ayrı ayrı yer vermek yerine döngü yapısı ile daha basit ve sade şekilde ilerlenebileceği ifade edilmiştir. Daha sonra öğrencilerin bu konuya yönelik örnekler yapmaları sağlanmıştır. Döngüler konusuna yönelik derste yapılan bir çalışma Şekil 45'te verilmiştir.



Şekil 45. DG “Döngü” kavramına ilişkin bir örnek

Paralel şekilde kontrol grubuyla da programlama dilinde döngüler konusuna geçilmiştir. Aynı şekilde programlama içerisinde döngü yapısı kullanılarak tekrar eden davranışların daha basit şekilde yapılacağı bu gruba da ifade edilmiş ve bu konuya yönelik örnekler yapmaları sağlanmıştır. Döngüler konusuna yönelik derste yapılan bir çalışma Şekil 46’da verilmiştir.



Şekil 46. KG “Döngü” kavramına ilişkin bir örnek

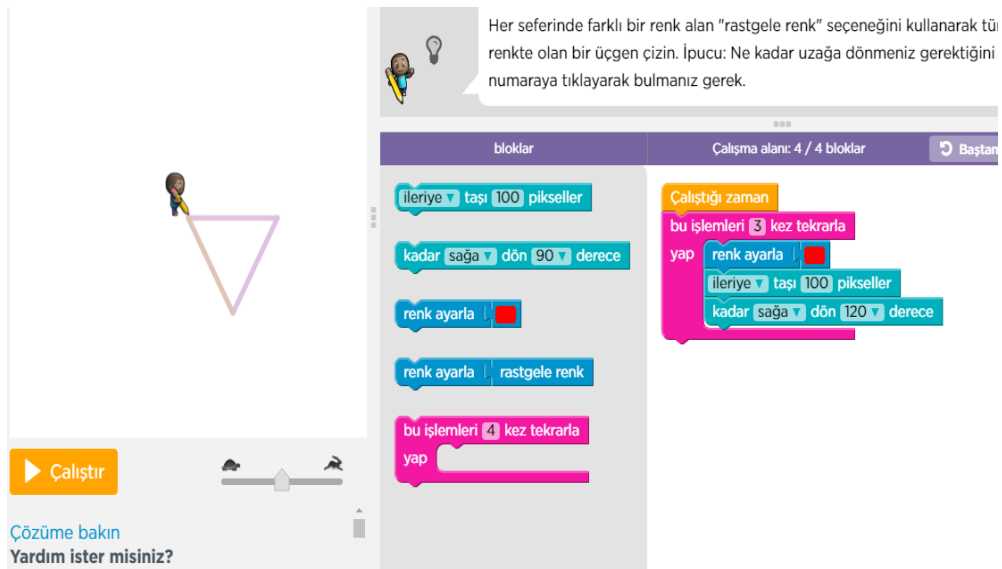
Sonraki süreçte deney gurubu ile koşul kavramı ele alınmıştır. Bu kavramla ilgili öğrencilere günlük hayatta yapılan seçimlerden ve verilen kararlardan örnekler verilerek farkındalık oluşturulmuştur. Ayrıca programa ilişkin alınan her kararın farklı sonuçlar

Şekil 48. KG “Koşul” kavramına ilişkin bir örnek

Bir sonraki derste deney gurubuyla değişken kavramı işlenmiştir. Bu kavrama ilişkin öğrencilere, bir değişkenin değişebilen veya sabit olarak kalan değerler alabileceği ifade edilmiştir (Şekil 49). Değişken kullanımına yönelik derste işlenen bir örnek Şekil 50’de verilmiştir.

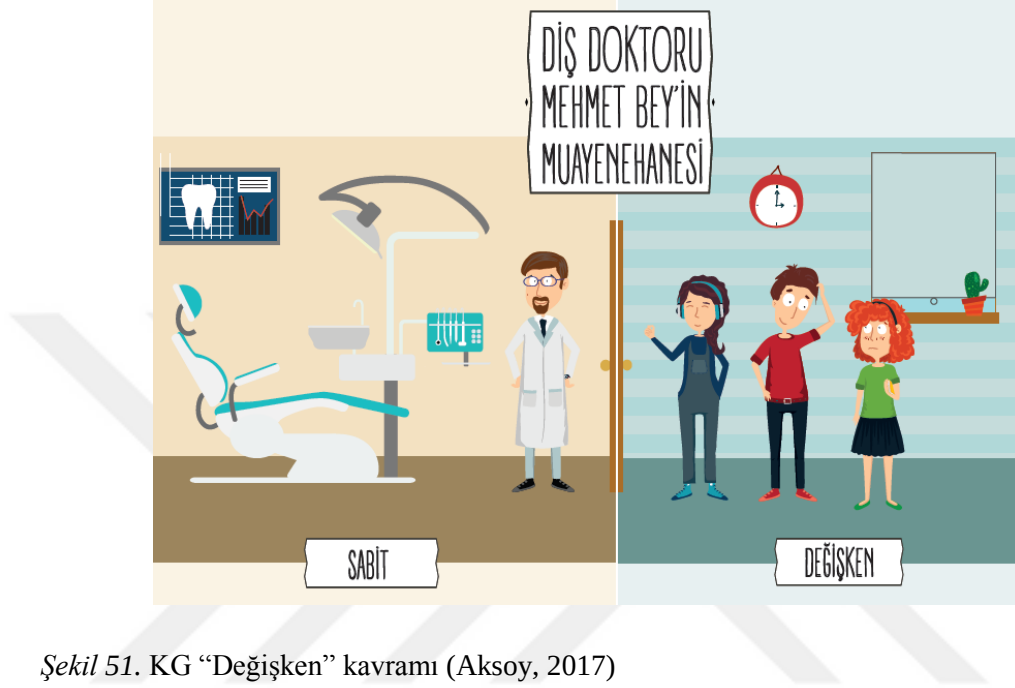


Şekil 49. DG “Değişken” kavramı (Aksoy, 2017)

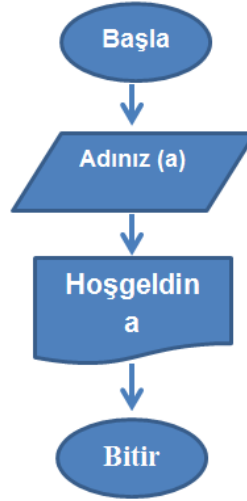


Şekil 50. DG “Değişken” kavramına ilişkin bir örnek

Kontrol grubuyla da programlama dilinde değişken kavramı işlenmiştir. Değişkenin sabit bir değer veya değişen farklı değer alabileceği öğrencilere gösterilerek (Şekil 51), değişken kullanımına yönelik derste yer verilen bir örnek Şekil 52’de verilmiştir.



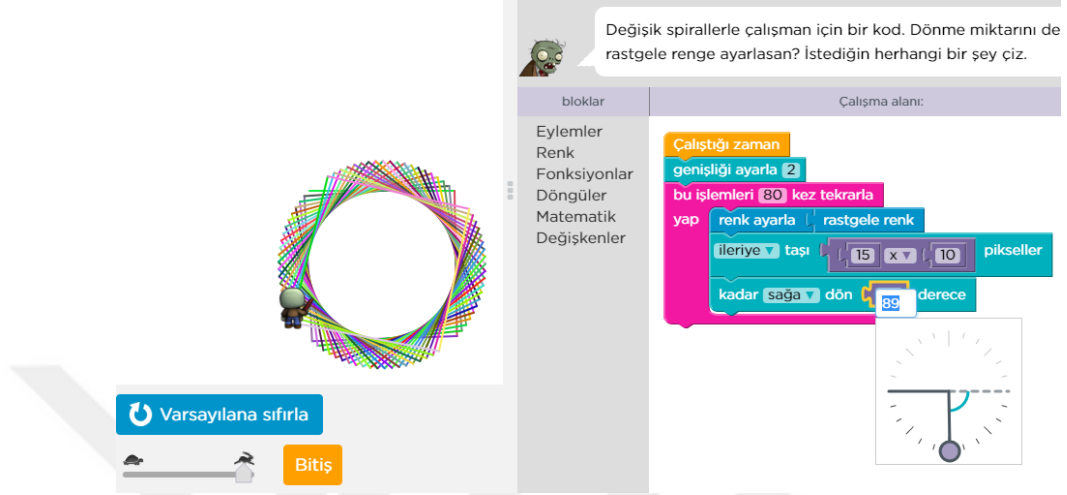
Şekil 51. KG “Değişken” kavramı (Aksoy, 2017)



Şekil 52. KG “Değişken” kavramına ilişkin bir örnek

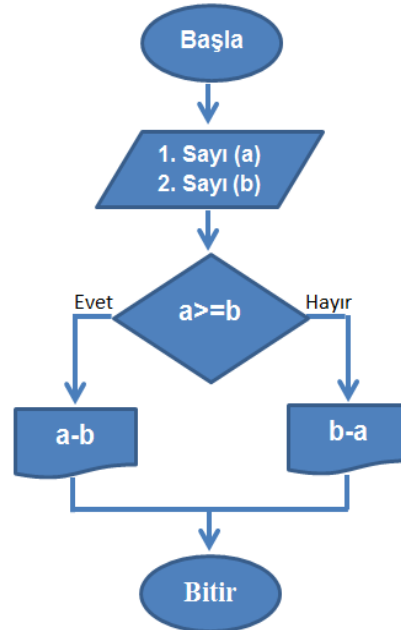
Deney grubuyla işlenen bir diğer derste programlama dilinde mantıksal ve matematiksel operatörler kavramları anlatılmıştır. Bu kavramlarla ilgili olarak “+, ×, -, /, <,”

$\leq$, $>$, $\geq$, ve, veya, değil” işlemleri aktarılmıştır. Bu konuya ilişkin deney grubuyla işlenen bir örnek Şekil 53’te verilmiştir.



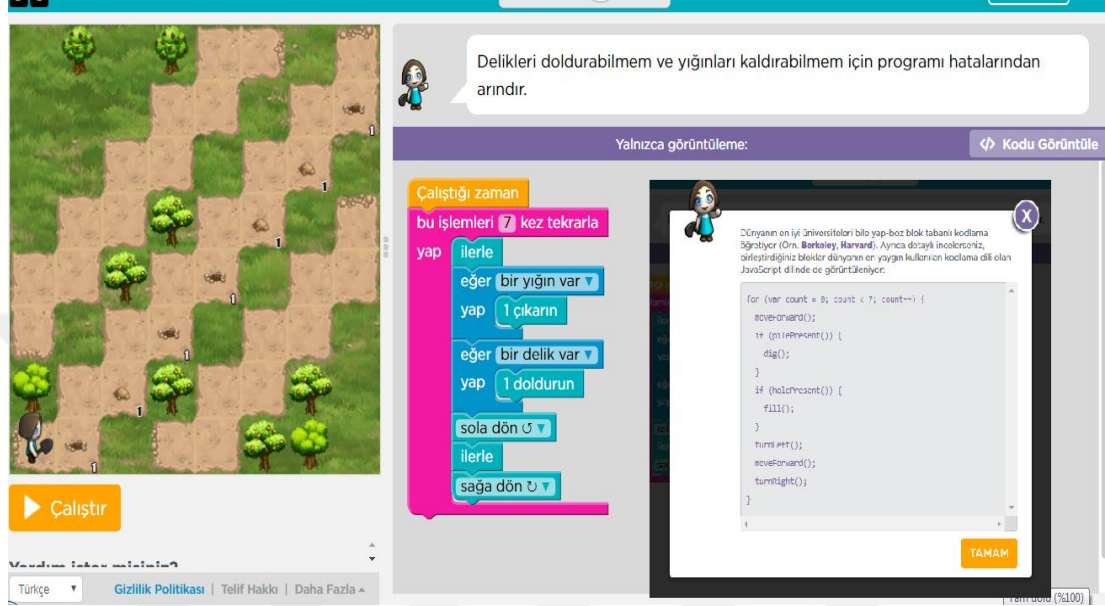
Şekil 53. DG “Operatörler” kavramına ilişkin bir örnek

Kontrol grubuyla işlenen sonraki derste de programlama dilinde mantıksal ve matematiksel operatörler kavramları anlatılmış, aynı şekilde bu kavramlarla ilgili olarak “ $+$, $\times$, $-$, $/$, $<$, $\leq$, $>$, $\geq$, ve, veya, değil” işlemleri aktarılmıştır. Bu konuya ilişkin kontrol grubuyla işlenen bir örnek Şekil 54’te verilmiştir.



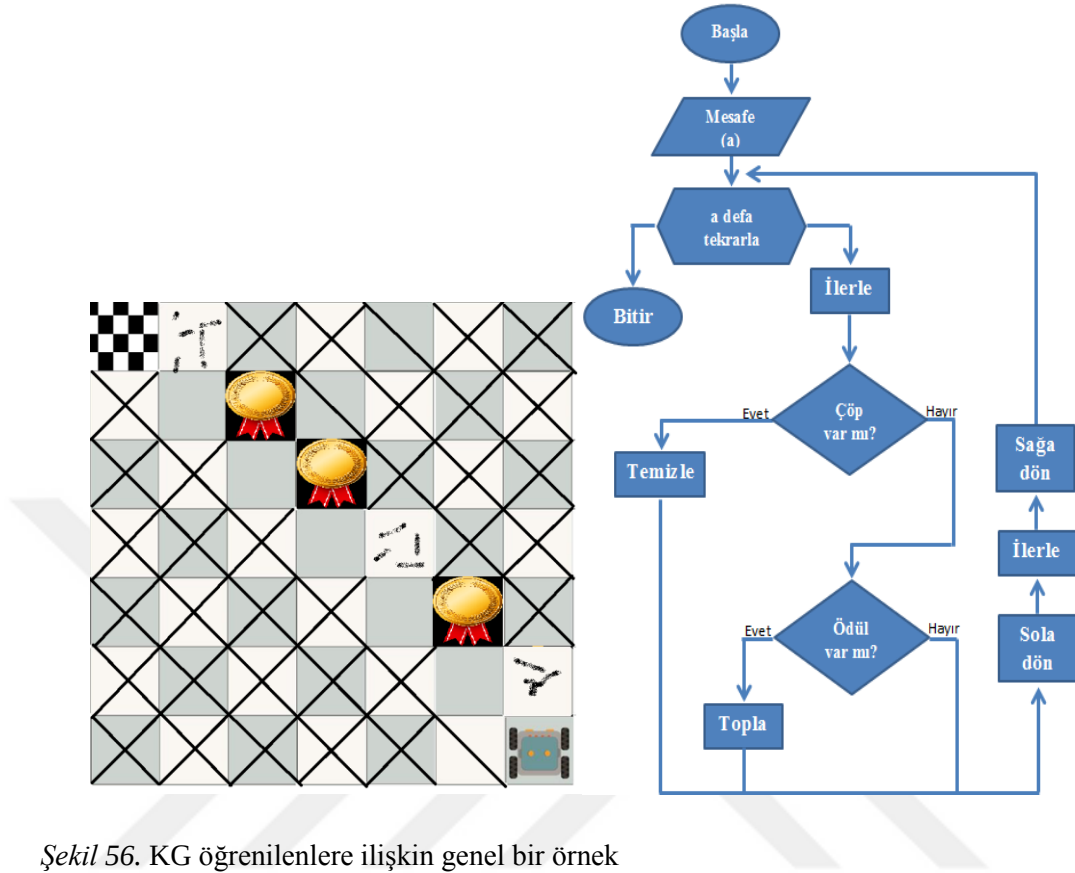
Şekil 54. KG “Operatörler” kavramına ilişkin bir örnek

Son olarak deney gurubuyla gerçekleştirilen öğretimde programlama dilinde öğrencilerin şuana kadar öğrendiği kavramlara yönelik genel örnekler üzerinde durulmuştur. Derste işlenen örneklerden biri Şekil 55’te verilmiştir.



Şekil 55. DG öğrenilenlere ilişkin genel bir örnek

Benzer şekilde kontrol grubuyla da programlama dilinde öğrencilerin şuana kadar öğrenilen kavramların yer aldığı örnekler üzerinde durulmuştur. Derste işlenen örneklerden biri Şekil 56’da verilmiştir.



Şekil 56. KG öğrenilenlere ilişkin genel bir örnek

3.5. Verilerin Analizi

Çalışma çerçevesinde belirtilen ölçme araçları (EBODKÖYTÖ, PÖDA ve CPGF) uygulanmış ve analiz sürecine geçilmiştir.

3.5.1. EBODKÖYTÖ verilerinin analizi

Öğrencilerin EBO Destekli Kodlama Öğrenimine yönelik işlem öncesi ve işlem sonrası tutumlarını belirlemek amacıyla uygulama öncesinde ve sonrasında EBODKÖYTÖ uygulanmıştır. Elde edilen anketlerin sonuçları aşağıda yer aldığı şekilde SPSS programına aktarılmıştır. Öğrencilerin ölçekteki “Kesinlikle Katılıyorum”, “Katılıyorum”, “Kararsızım”, “Katılmıyorum”, “Kesinlikle Katılmıyorum” seçeneklerinden bir tanesini işaretlemesi istenmiştir.

Tutum ölçeklerinin puanlamasında boş bırakılan maddelere sıfır puan verilmeyip cevabın ortalama, standart sapma ve yüzde hesaplamalarında dikkate alınmaması gerekmektedir (Turgut ve Baykul, 2012). Özgün yapısıyla likert tipi bir ölçme aracında her bir madde için verilecek yanıt beş seçenekten birisidir. Bu seçenekler için birden beşe doğru

bir puan dağılımı olmakta, her maddede üç puan kararsızlık durumunu ifade etmektedir. Ayrıca bu çalışma içerisinde öğrenciler tarafından iki veya daha fazla seçeneğin işaretlendiği ya da boş bırakılan seçeneklere üç puan verilmiştir. Üç puan madde ölçek çerçevesinde SPSS programına kayıp değer olarak işlenmiştir. Bir puan negatif uçtaki tutumu, beş puan pozitif uçtaki tutumu temsil etmektedir. Puanlama yolu ölçek boyunca aynı kalmalıdır. Boş bırakılmış cevap bulunmamalıdır (Tezbaşaran, 2008).

EBODKÖYTÖ verilerinin çözümlenmesi sürecinde parametrik olmayan testlerden (Mann-Whitney U testi, Wilcoxon işaretli sıra sayıları testi) faydalanılmıştır. Parametrik istatistiklerin kullanımı için dağılımın normalliği varsayımına gerek duyulmaktadır. Eğer araştırmacının dağılımın normalliğine ilişkin yeterli kanıtı yoksa veya mevcut deneysel çalışmada az denek bulunuyorsa parametrik olmayan istatistikler kullanılmalıdır (Büyüköztürk, 2006). Mann-Whitney U testi bağlantısız ölçme işlemlerinin olduğu, denek sayısı az olan deneysel çalışmalarda ilişkisiz t-testinin yerine kullanılabilir (Büyüköztürk, 2006). Mann-Whitney U testi bağımsız örneklerle yönelik kullanılan t-testinin non-parametrik alternatifidir (Karagöz, 2010). Wilcoxon işaretli sıralar testi, çoğunlukla denek sayısı az olarak gerçekleştirilen gruplar içi araştırmalarda kullanılmaktadır. Aynı deneklerden oluşan gruba iki ayrı zamanda yapılan ölçme işlemi sonuçlarının kullanıldığı işaretli sıralar testi, ilişkili t-testinin yerine kullanılmaktadır (Büyüköztürk, 2006). Araştırmadaki veri sayısının 30'dan az olduğu nicel veriler varlığında (n=22) parametrik olmayan testler tercih edilmelidir (Yılmaz, 2007).

EBODKÖYTÖ kapsamında toplanan veriler SPSS yazılımına girilmiştir. Deney ve kontrol grubunda yer alan öğrencilerin ön test ve son test ölçüm sonuçları arasında anlamlı bir farka rastlanılma durumu ilişkisiz örneklemeler için Mann-Whitney U testi yapılarak test edilmiştir. Analizlerde ölçüm sonuçları arasındaki farkın anlamlılığı .05 düzeyinde değerlendirilmiştir. Grupların her biri için, deneysel işlem öncesi ölçülen ön test puanları ve deneysel işlem sonrası ölçülen son test puanları arasında anlamlı bir fark olup olmadığı bağımlı örneklemeler için Wilcoxon işaretli sıralar testi yapılarak test edilmiştir. Analizlerde ölçüm sonuçları arasındaki farkın anlamlılığı .05 düzeyinde değerlendirilmiştir.

3.5.2. PÖDA verilerinin analizi

PÖDA verilerinin çözümlenmesi sürecinde de kovaryans analizi, ikili karşılaştırmalar testi, betimsel istatistikler ve parametrik olmayan testlerden (ANCOVA, Bonferroni, Mann-Whitney U testi, Wilcoxon işaretli sıra sayıları testi) faydalanılmıştır. PÖDA kapsamında öğrencilerin açık uçlu sorulara verdiği yanıtlardan oluşan veriler üç ayrı alan uzmanı tarafından pilot çalışma sürecinde oluşturulan cevap anahtarına göre ($10 \times 10=100$ puan) değerlendirilmiş, ulaşılan sonuçların ortalamaları alınarak elde edilen veriler SPSS yazılımına girilmiştir.

Deney ve kontrol grubunda yer alan öğrencilerin ön test ve son test ölçüm sonuçları arasında anlamlı bir farka rastlanılma durumu ilişkisiz örneklem için Mann-Whitney U testi yapılarak test edilmiştir. Grupların izleme testi sonuçları ise kovaryans analizi (ANCOVA) ile karşılaştırma yapılarak irdelenmiştir. ANCOVA deseni, ortak değişkene göre ayarlanmış grup ortalamalarının, birbirinden anlamlı bir şekilde farklılık gösterip göstermediğini test etmektedir (Büyüköztürk, 2015). Örneklem büyüklüğünün az olduğu durumlarda etkili sonuçlar verebilmektedir (Kalaycı, 2010). Kovaryans analizi sonucunda ortaya çıkan farkın kaynağını belirlemek amacıyla Bonferroni ikili karşılaştırmalar testi uygulanmıştır. Analizlerde ölçüm sonuçları arasındaki farkın anlamlılığı .05 düzeyinde değerlendirilmiştir.

Deney ve kontrol gruplarının her biri için, deneysel uygulama öncesi ölçülen ön test puanları, deneysel uygulama sonrası ölçülen son test puanları ve izleme testi puanları arasında anlamlı bir fark olup olmadığı bağımlı örneklem için Wilcoxon işaretli sıralar testi yapılarak test edilmiştir. Analizlerde ölçüm sonuçları arasındaki farkın anlamlılığı .05 düzeyinde değerlendirilmiştir.

3.5.3. CPGF verilerinin analizi

CPGF ölçme araçlarından toplanan verilerin çözümlenmesinde içerik analizinden faydalanılmıştır. İçerik analizinin kullanım amacı, ulaşılan verileri ifade edebilecek kavramlara ve bağlantılara erişmektir. Betimsel analizde özetlenerek ve yorumlanarak elde edilen veriler, içerik analizi yapılarak daha detaylı bir incelemeden geçirilmekte, bu sayede betimsel bir bakış açısıyla fark edilemeyen kavram ve kategorilerin keşfi mümkün olmaktadır (Yıldırım ve Şimşek, 2013). Görüşmede alınan ses kaydı ve notlar, uygulamanın gerçekleştirilmesinde etkili olan unsurları dikkate alınarak değerlendirilmiştir. Araştırmada

görüşme formunda yer verilen sorular birer tema olarak kabul edilmiş ve görüşmeye katılan deney grubu öğrencilerinin bu sorulara verdikleri cevaplardan kategoriler oluşturularak sunulmuştur. Betimsel analiz yapılan bir araştırmanın geçerliğinin sağlanması için görüşülen bireylerin ifadelerine doğrudan yer vermek ve bu ifadelerin doğrultusunda sonuçları açıklamak önem taşımaktadır (Yıldırım ve Şimşek, 2008). Bu çalışmada da görüşme yapılan öğrencilerin görüşlerine yönelik doğrudan alıntılara yer verilmiştir. Ayrıca, kategorilerin hangi sıklıkta tekrar ettiği belirlenmiş ve tablo haline getirilerek sunulmuştur. Yapılan çalışmaya yönelik bir örnek aşağıdaki gibidir:

Tablo 7.

CPGF madde 1 (Örnek)

Kategoriler	f	Örnek Alıntılar
Eğlenceli	9	Ö5 Daha eğlenceli, oyun gibi kullandık Ö6 Daha iyi, etkili, güzel öğrenebildik.
Uygulamalı	8	Ö4 Yaparak, görerek öğrenebildik. Ö1 Kendi oluşturduğumuz oyunları oynayabiliriz.
Öğrenmeyi kolaylaştırıcı	8	Ö10 Aşamalı şekilde kolaydan zora ilerledik. Ö9 Daha hızlı öğrenebildik. Ö3 Öğretimi kolaylaştırdı.

*Bir öğrencinin birden fazla görüşü yer alabilmektedir

Araştırma kapsamında öğrencilere “Code.org platformunun kodlamayı öğrenmen üzerinde (Akış Şeması, Döngüler, Koşul Yapıları vb.) etkili olduğunu düşünüyor musun? Etkili ise hangi yönlerden etkili oldu? Ne kazandırdı?” sorusu sorulmuştur. Öğrencilerin tamamı etkili olma konusunda “evet” cevabını vermiştir. Nasıl etkili olduğu ve sürecin öğrencilere kazandırdıkları ile ilgili verilen cevaplar betimsel analize tabi tutulmuş, yukarıdaki örnekte olduğu gibi kategoriler belirlenmiştir. İlgili kategorilere verilen örnek yanıtlardan alıntılar yapılmıştır.

CPGF ölçme araçlarından elde edilen veriler; birbirinden bağımsız iki uzman kişi tarafından ayrı ayrı kodlanmış ve güvenilirliği tespit etmek amacıyla kodlayıcılar arası uyum yüzdesi (Miles, Huberman ve Saldana, 2014) hesaplanmıştır. Bu değer hesaplanmasında Miles ve Huberman’ın (1994) aşağıdaki uyum yüzdesi formülü kullanılmıştır.

$$Uyum\ yüzdesi\ (P) = \frac{Na\ (Görüş\ birliği)}{Na\ (Görüş\ birliği) + Nd\ (Görüş\ ayrılığı)} \times 100$$

Çalışmanın güvenilir sıfatını alabilmesi için bu oranın %85 ve üzeri olması beklenmektedir (Miles ve diğerleri, 2014). Yukarıdaki denkleme göre CPGF kodlayıcıları arası uyum yüzdesi 90,91 olarak hesaplanmıştır.



BÖLÜM IV

BULGU VE YORUMLAR

Çalışmanın bu kısmında araştırmanın denencelerinin test edilmesi sonucu ulaşılan bulgular sunulmuş ve buna bağlı olarak yorumlar yapılmıştır. Elde edilen bulgulara PÖDA (ön test - son test - izleme testi), EBODKÖYTÖ (ön test - son test) ve CPGF (deney grubu, son test sonrası) veri toplama araçları vasıtasıyla erişilmiştir.

4.1. Code.org Platformunun 6. Sınıf Öğrencilerinin Programlama Öğrenim Düzeylerine Etkisi

Tablo 8’deki Mann Whitney U testi sonuçları dikkate alındığında ön test sonuçları arasında anlamlı farklılık bulunmayan deney ve kontrol grubu öğrencilerinin son test olarak uygulanan PÖDA ölçeği puanları arasında .05 düzeyinde deney grubu lehine anlamlı bir fark bulunmuştur ($U=28.000$, $p<.05$).

Tablo 8.

PÖDA son test uygulama sonuçlarının gruplara göre U testi sonuçları

Grup	N	Sıra Ortalaması	Sıra Toplamı	U	z	p
Kontrol	11	8.55	94.00	28.000	-2.14	.032
Deney	11	14.45	159.00			
Toplam	22					

Tablo 9’a göre yapılan Wilcoxon İşaretili Sıralar testi sonuçları dikkate alınarak kontrol grubunda yer alan öğrencilerin ön test ve son test PÖDA ölçeği puanları arasında .05 düzeyi baz alınarak son test lehine anlamlı bir fark bulunmuştur ($p<.05$).

Tablo 9.

KG ön test ve son test PÖDA sonuçlarının karşılaştırılması

Ön Test-Son Test	n	Sıra Ortalaması	Sıra Toplamı	z	p
Azalma	0	.00	.00		
Artma	11	6.00	66.00	-2.934*	.003
Eşit	0	-	-		

*Negatif sıralar temeline dayalı

Tablo 10'a göre yapılan Wilcoxon İşaretli Sıralar testi sonuçlarına bakıldığında deney grubunda yer alan öğrencilerin ön test ve son test PÖDA ölçeği puanları arasında .05 düzeyi baz alınarak son test lehine anlamlı bir fark bulunmuştur ($p < .05$).

Tablo 10.

DG ön test ve son test PÖDA sonuçlarının karşılaştırılması

Ön Test-Son Test	n	Sıra Ortalaması	Sıra Toplamı	z	p
Azalma	0	.00	.00		
Artma	11	6.00	66.00	-2.936*	.003
Eşit	0	-	-		

*Negatif sıralar temeline dayalı

4.2. Code.org Platformunun 6. Sınıf Öğrencilerinin Programlama Öğrenimlerinin Kalıcılık Düzeylerine Etkisi

DG ve KG arasında PÖDA son test puanları kontrol altına alındığında izleme testi puanları açısından gruplar arasında istatistiksel olarak anlamlı fark olup olmadığını belirlemek için, son testin uygulanmasından 5 hafta sonra izleme testi uygulanmıştır. Deney ve kontrol gruplarındaki öğrencilerin PÖDA son test ve izleme testi puanlarının istatistiki verileri Tablo 11'de sunulmuştur.

Tablo 11.

KG ve DG son test-izleme testi PÖDA verileri

Gruplar	N		Toplam Puanlar		Düzeltilmiş Ortalamalar	
			$\bar{X}$	SS	$\bar{X}$	SH
Deney	11	Son test	62.91	14.29		
		İzleme testi	63.18	19.12	63.51	5.06
Kontrol	11	Son test	49.27	12.07		
		İzleme testi	41.27	10.07	40.95	5.06

Kovaryans analizi sonuçları, son test toplam puanları kontrol altına alındığında, grupların düzeltilmiş izleme testi toplam puanları açısından anlamlı derecede farklılaştığını göstermiştir. Farklılaşmanın hangi grup lehine olduğunu istatistiksel olarak belirlemek için Bonferroni ikili karşılaştırmalar testi yapılmış, deney grubu lehine anlamlı farklılık tespit edilmiştir ($p=.008<.05$) (Tablo 12).

Tablo 12.

DG ve KG izleme testi PÖDA sonuçlarının karşılaştırılması

Gruplar	Ort. Farklar (DG-KG)	Std. Hata	p	Bonferroni
DG KG	22.555	7.595	.008	DG > KG

4.3. Code.org Platformunun Deney ve Kontrol Grubunda Yer Alan Öğrencilerin EBODKÖY Tutumlarına Etkisi

Tablo 13'teki Mann Whitney U testi sonuçları doğrultusunda ön test sonuçları arasında anlamlı farklılık bulunmayan deney ve kontrol grubu öğrencilerinin son test olarak uygulanan EBODKÖYTÖ puanları arasında .05 düzeyinde deney grubu lehine anlamlı bir fark bulunmuştur ($U=30.000$, $p<.05$).

Tablo 13.

EBODKÖYTÖ son test uygulama sonuçlarının gruplara göre U testi sonuçları

Grup	N	Sıra Ortalaması	Sıra Toplamı	U	z	p
Kontrol	11	8.73	96.00	30.000	-2.004	.045
Deney	11	14.27	157.00			
Toplam	22					

Tablo 14'e göre yapılan Wilcoxon İşaretli Sıralar testi sonuçlarına bakıldığında kontrol grubunda yer alan öğrencilerin ön test ve son test EBODKÖYTÖ puanları arasında .05 düzeyinde anlamlı bir farkın oluşmadığı görülmektedir ($p > .05$).

Tablo 14.

KG ön test ve son test EBODKÖYTÖ sonuçlarının karşılaştırılması

Ön Test-Son Test	n	Sıra Ortalaması	Sıra Toplamı	z	p
Azalma	5	6.40	32.00	-.089*	.929
Artma	6	5.67	34.00		
Eşit	0	-	-		

*Negatif sıralar temeline dayalı

Tablo 15'e göre yapılan Wilcoxon İşaretli Sıralar testi sonuçları doğrultusunda deney grubunda yer alan öğrencilerin ön test ve son test EBODKÖYTÖ puanları arasında .05 düzeyi baz alınarak son test lehine anlamlı bir fark bulunmuştur ($p < .05$).

Tablo 15.

DG ön test ve son test EBODKÖYTÖ sonuçlarının karşılaştırılması

Ön Test-Son Test	n	Sıra Ortalaması	Sıra Toplamı	z	p
Azalma	1	2.00	2.00		
Artma	9	5.89	53.00	-2.599*	.009
Eşit	1	-	-		

*Negatif sıralar temeline dayalı

4.4. Deney Gurubu Öğrencilerinin Code.org Platformuna Yönelik Görüşleri

“Deney gurubu öğrencilerinin Code.org platformuna yönelik görüşleri nelerdir?” problem cümlesine cevap aramak amacıyla, deney grubu öğrencileriyle yapılan yarı yapılandırılmış görüşme ölçeğine verdikleri cevaplar kategorize edilmiş ve bu kategorilere ait frekans değerleri sırayla verilmiştir.

Tablo 16.

CPGF madde 1

<i>Kategoriler</i>	<i>f</i>	<i>Örnek Alıntılar</i>
Eğlenceli	9	Ö5 Daha eğlenceli, oyun gibi kullandık. Ö6 Daha iyi, etkili, güzel öğrenebildik.
Uygulamalı	8	Ö4 Yapararak, görerek öğrenebildik. Ö1 Kendi oluşturduğumuz oyunları oynayabiliriz.
Öğrenmeyi kolaylaştırıcı	8	Ö10 Aşamalı şekilde kolaydan zora ilerledik. Ö9 Daha hızlı öğrenebildik. Ö3 Öğretimi kolaylaştırdı.

*Bir öğrencinin birden fazla görüşü yer alabilmektedir

Araştırma kapsamında öğrencilere “Code.org platformunun kodlamayı öğrenmen üzerinde (Akış Şeması, Döngüler, Koşul Yapıları vb.) etkili olduğunu düşünüyor musun? Etkili ise hangi yönlerden etkili oldu? Ne kazandırdı?” sorusu sorulmuştur. Öğrencilerin tamamı etkili olma konusunda “evet” cevabını vermiştir. Nasıl etkili olduğu ve sürecin öğrencilere kazandırdıkları ile ilgili verilen cevaplar betimsel analize tabi tutulmuş, Tablo

16'daki gibi kategoriler belirlenmiştir. İlgili kategorilere verilen örnek yanıtlardan alıntılar yapılmıştır.

Tablo 17.

CPGF madde 2

<i>Kategoriler</i>	<i>f</i>	<i>Örnek Alıntılar</i>
Uygulamalı	8	Ö4 Çünkü görerek ve yaparak öğrenebildim.
Eğlenceli	6	Ö8 Çünkü oyun oynar gibi öğrenebildim. Ö1 Çünkü eğlenceliydi.
Esnek öğrenme ortamı	1	Ö9 Çünkü hem okulda hem evde her zaman kullanabiliyoruz.

*Bir öğrencinin birden fazla görüşü yer alabilmektedir

Araştırma kapsamında öğrencilere “Code.org platformunu kullanmanın kodlamayı daha kalıcı olarak öğrenmene etki edeceğini düşünüyor musun? Açıklayabilir misin?” sorusu sorulmuştur. Öğrencilerin tamamı kalıcı öğrenme konusunda “evet” cevabını vermiştir. Platformun, kalıcılığa nasıl etki ettiğine ilişkin öğrencilerin verdiği cevaplar betimsel analize tabi tutulmuş, Tablo 17’deki gibi kategoriler belirlenmiştir. İlgili kategorilere verilen örnek yanıtlardan alıntılar yapılmıştır.

Tablo 18.

CPGF madde 3

<i>Kategoriler</i>	<i>f</i>	<i>Örnek Alıntılar</i>
Eğlenceli	12	Ö3 Oyun oynar gibi kullanabiliyorduk. Ö7 Bölümler (Çiftçi, Angry Birds vb.)güzeldi. Ö2 Çeşitli bölümler vardı. Ö6 Eğlenceliydi.
Uygulamalı	6	Ö1 Kendi hesabımızla oturum açıp kullanıyorduk.
Yönlendirici	4	Ö5 İlerleme durumumuzu, tamamlanan, yarım kalan, yapılmamış bölümlerimizi renkli olarak takip ediyorduk. Ö9 Bölümlerden önce bilgilendirici videolar vardı.
Öğrenmeyi kolaylaştırıcı	2	Ö10 Aşamalı olarak kolaydan zora doğru gidiyordu bölümler.

*Bir öğrencinin birden fazla görüşü yer alabilmektedir

Araştırma kapsamında öğrencilere “Code.org platformunun en beğendiğin tarafları nelerdi? Açıklar mısın?” sorusu sorulmuştur. Öğrencilerin platforma yönelik beğenilerini dile getirdiği cevaplar betimsel analize tabi tutulmuş, Tablo 18’deki gibi kategoriler belirlenmiştir. İlgili kategorilere verilen örnek yanıtlardan alıntılar yapılmıştır.

Tablo 19.

CPGF madde 4

<i>Kategoriler</i>	<i>f</i>	<i>Örnek Alıntılar</i>
Yardım alma	9	Ö1 Zorlandığım zamanlarda arkadaşlarımdan yardım aldım.
Olumsuzluğa kapılma	8	Ö1 Zorlandığım zamanlarda canım sıkıldı, olumsuz hissettim.
Olumlu hissetme	3	Ö6 Zorlandığım zamanlarda pek olumsuz hissetmedim. Ö6 Zorlandığım zamanlarda farklı şekillerde deneyerek devam ettim. Başka şeyler de öğrenmiş oldum.
Diğer	1	Ö3 Biraz daha kolay olsa daha iyi olurdu.

*Bir öğrencinin birden fazla görüşü yer alabilmektedir

Araştırma kapsamında öğrencilere “Code.org platformunu kullanırken karşılaştığın problemler nelerdir? Bu problemlerle nasıl başa çıktın? Bu problemler öğrenme isteğinde olumsuz yönde etkili oldular mı?” sorusu sorulmuştur. Öğrencilerin %9.1’i ($f=1$) “bir problemle karşılaşmadım” yanıtını verirken diğer öğrenciler (%90 $f=10$) “zorlandığım bölümlerde problem yaşadım” cevabını iletiler. Problem yaşadığını ifade eden öğrencilerin, görüşlerini açıklarken verdikleri cevaplar betimsel analize tabi tutulmuş, Tablo 19’daki gibi kategoriler belirlenmiştir. İlgili kategorilere verilen örnek yanıtlardan alıntılar yapılmıştır.

Tablo 20.

CPGF madde 5

<i>Kategoriler</i>	<i>f</i>	<i>Örnek Alıntılar</i>
Zorluk	3	Ö10 Bir bölümü geçerken zorlandığımızda ekstra ipucu videoları olabilir. Ö1 Bölümlerde hamle sınırı olmasa daha iyi olurdu. Ö3 Bazı zor olan bölümler olmasa daha iyi olurdu.
Anlaşılabilirlik	2	Ö11 Görüntülenen kodlar Türkçe olabilirdi.
Diğer	2	Ö1 Bölümlerde farklı karakterler olabilirdi (zombi - domuz vb. yerine). Ö11 Code.org platformu Türkiye’ye ait olsa daha iyi olurdu.

*Bir öğrencinin birden fazla görüşü yer alabilmektedir

Araştırma kapsamında öğrencilere “Aynı uygulama bir kez daha yapılsa nelerin eklenmesini veya değiştirilmesini isterdin?” sorusu sorulmuştur. Öğrencilerin %63.6’sı ($f=7$) “aynı şekilde kalmalı” cevabı ile bir değişiklik istemediklerini belirtirken diğer öğrenciler (%36.3 $f=4$) bazı değişikliklerin yapılabileceğini ifade etmişlerdir. Öğrencilerin, değişmesini istedikleri hususları açıklarken verdikleri cevaplar betimsel analize tabi tutulmuş, Tablo 20’deki gibi kategoriler belirlenmiştir. İlgili kategorilere verilen örnek yanıtlardan alıntılar yapılmıştır.

Tablo 21.

CPGF madde 6

<i>Kategoriler</i>	<i>f</i>	<i>Örnek Alıntılar</i>
Eğlenceli	11	Ö11 Çünkü çok eğlenceli. Ö6 Çünkü çok beğendim, çok güzel bir uygulama.
Öğrenmeyi kolaylaştırıcı	7	Ö5 Daha akılda kalıcı öğreniliyor. Ö2 Daha kolay öğreniliyor. Ö8 Daha hızlı öğreniliyor. Ö6 Daha öğretici bir uygulama.

*Bir öğrencinin birden fazla görüşü yer alabilmektedir

Araştırma kapsamında öğrencilere “Bu platform ile kodlama öğrenmeyi bir arkadaşına önerir misin? Neden?” sorusu sorulmuştur. Öğrencilerin tamamı platformu bir başkasına önerme konusunda “evet” cevabını vermiştir. Bu önerilerin sebeplerini açıklarken öğrencilerin verdiği cevaplar betimsel analize tabi tutulmuş, Tablo 21’deki gibi kategoriler belirlenmiştir. İlgili kategorilere verilen örnek yanıtlardan alıntılar yapılmıştır.

Tablo 22.

CPGF madde 7

<i>Kategoriler</i>	<i>f</i>	<i>Örnek Alıntılar</i>
Çalışma	10	Ö11 Çok çalışırsam kendi oyunumu oluşturabilirim. Ö8 Zor, çalışsam bile başaramayabilirim.
Yardım alma	1	Ö2 Yardım alarak kendi oyunumu oluşturabilirim.

*Bir öğrencinin birden fazla görüşü yer alabilmektedir

Araştırma kapsamında öğrencilere “Kodlama ile kendi oyununu kendin yazmak ister miydin? Sence bunu başarabilir misin?” sorusu sorulmuştur. Öğrencilerin tamamı “kendi oyunumu yazmak isterdim” cevabını vermiştir. Bu isteklerini açıklarken verilen cevaplar betimsel analize tabi tutulmuş, Tablo 22’deki gibi kategoriler belirlenmiştir. İlgili kategorilere verilen örnek yanıtlardan alıntılar yapılmıştır.

Tablo 23.

CPGF madde 8

<i>Kategoriler</i>	<i>f</i>	<i>Örnek Alıntılar</i>
Diğer	7	Ö6 Kodlama her yerde öğretiliyor mu? (Yurtiçi-yurtdışı) Ö1 Kodlama nereden yapılıyor? Ö8 Oynadığımız bilgisayar oyunları nasıl yapılıyor? Ö10 Niye Türkçe dili kullanılmıyor?

*Bir öğrencinin birden fazla görüşü yer alabilmektedir

Araştırma kapsamında öğrencilere “Kodlama alanında merak ettiğin konular neler? Açıklar mısın?” sorusu sorulmuştur. Öğrencilerin %36.4’ü ($f=4$) “merak ettikleri bir şey olmadığı” cevabını iletirken diğer öğrenciler (%63.6 $f=7$) merak ettikleri konuları ifade etmişlerdir. Öğrencilerin, merak ettikleri hususları açıklarken verdikleri cevaplar betimsel analize tabi tutulmuş, Tablo 23’teki gibi kategoriler belirlenmiştir. İlgili kategorilere verilen örnek yanıtlardan alıntılar yapılmıştır.

Tablo 24.

CPGF madde 9

<i>Kategoriler</i>	<i>f</i>	<i>Örnek Alıntılar</i>
Eğlenceli	11	Ö3 Çünkü sevdim, güzel ve eğlenceliydi. Ö5 Kendi oluşturduğum oyunu sunmak güzel olurdu. Ö2 Başkalarına öğretebilmek güzel olurdu. Ö6 Bazı değişikliklerle çocuklar için kodlamayı daha ilgi çekici hale getirebilirdim.
Kazançlı	2	Ö1 Bu şekilde çok para kazanılabilir.

*Bir öğrencinin birden fazla görüşü yer alabilmektedir

Araştırma kapsamında öğrencilere “Kodlama ile ilgili bir iş alanında çalışmak ister miydin? Açıklar mısın?” sorusu sorulmuştur. Öğrencilerin %72.7’si ($f=8$) “çalışmak isterdim” cevabını, %18.2’si ($f=2$) “ilgimi çeken, sevebileceğim bir iş olursa çalışmak isterdim” cevabını ve %9.1’i ($f=1$) de “bilgisayar başında fazla oturmak istemediğim için çalışmak istemezdim” cevabını vermiştir. “çalışmak isterdim” yanıtını veren öğrencilerin,

görüşlerini açıklarken verdikleri cevaplar betimsel analize tabi tutulmuş, Tablo 24'teki gibi kategoriler belirlenmiştir. İlgili kategorilere verilen örnek yanıtlardan alıntılar yapılmıştır.

Tablo 25.

CPGF madde 10

<i>Kategoriler</i>	<i>f</i>	<i>Örnek Alıntılar</i>
Beğeni	15	Ö5 Bilişim Teknolojileri dersini artık daha çok seviyorum. Daha zevkli, eğlenceli ve güzel geçiyor dersler. Ö8 Daha fazla bilgisayar kullanabiliyoruz. Ö1 Daha fazla Bilişim Teknolojileri dersi işlemek istiyorum. Ö9 Ders daha hızlı bitiyor gibi geliyor.

*Bir öğrencinin birden fazla görüşü yer alabilmektedir

Araştırma kapsamında öğrencilere “Code.org platformunu kullanarak kodlama öğrenme sürecin, BTY dersi ile ilgili düşüncelerin değiştirdi mi? Eğer değiştirdiyse olumlu mu olumsuz yönde mi? Örnek verebilir misin?” sorusu sorulmuştur. Öğrencilerin tamamı “platform düşüncelerimi olumlu yönde değiştirdi” cevabını vermiştir. Bu olumlu değişimi açıklarken verilen cevaplar betimsel analize tabi tutulmuş, Tablo 25'teki gibi kategoriler belirlenmiştir. İlgili kategorilere verilen örnek yanıtlardan alıntılar yapılmıştır.

Tablo 26.

CPGF ek madde

<i>Kategoriler</i>	<i>f</i>	<i>Örnek Alıntılar</i>
Diğer	2	Ö4 Code.org'u çok beğendim. Ö9 Her okulda kullanılmalı.

Araştırma kapsamında öğrencilere “Eklemek istediğin herhangi bir şey var mı?” sorusu sorulmuştur. Öğrencilerin %81.8'i (f=9) “hayır” cevabını vererek ekleyecekleri bir şey olmadıklarını belirtmişlerdir. Diğer öğrencilerin verdiği yanıtlar Tablo 26'da gösterilmiştir.

BÖLÜM V

SONUÇ, TARTIŞMA VE ÖNERİLER

Çalışmanın bu kısımda araştırma çerçevesindeki bulgulardan yola çıkarak elde edilen sonuçlar açıklanmış, alanyazın incelemesi sonucu ulaşılan kaynaklarla bağlantı kurularak tartışılmıştır. Ayrıca ulaşılan bulgu ve sonuçların ışığında öneriler sunulmuştur.

5.1. Sonuç ve Tartışma

Araştırmanın bu kısmında iki farklı şekilde programlama öğrenimi gören öğrencilerin programlama öğrenimlerine, programlama öğrenimlerinin kalıcılık düzeylerine, EBODKÖY tutumlarına ve deney grubu öğrencilerinin Code.org platformuna dair görüşlerine yönelik ölçme sonuçları dile getirilmiş, alanyazındaki çalışmalarla arasındaki ilişkiler göz önüne alınarak tartışılmıştır.

5.1.1. Code.org platformunun 6. sınıf öğrencilerinin programlama öğrenim düzeylerine etkisi

- Deney ve kontrol grubunda yer alan öğrencilerin uygulama öncesi programlama öğrenim düzeylerinin (PÖDA Ön Test) farklı olmadığı sonucuna ulaşılmıştır.
- BTY dersi çerçeve programındaki “Problem Çözme ve Programlama” ünitesi; deney grubuyla Code.org platformu desteği ile kontrol grubuyla ise normal öğretim yöntemleri ile işlenmiştir. Ardından gerçekleştirilen ölçme sonucunu (PÖDA Son Test) ön test sonuçları ile karşılaştırdığımızda ise her iki grubun öğrenim düzeyinde de anlamlı bir artışın olduğu gözlenmiştir.
- Ayrıca araştırma probleminin tespiti için yapılan grup karşılaştırmasında deney grubu lehine anlamlı bir fark gözlenmiştir. Bu durumda Code.org platformunun programlama öğretiminde kullanılmasının normal yöntemlere göre başarılı olduğu sonucuna ulaşılmıştır. Alanyazında Code.org platformunun programlama öğrenim düzeyine ilişkin farklı bir platform ile karşılaştırıldığı bir çalışmaya rastlanılmamıştır. Genel olarak görsel programlama ortamlarının öğrencilerin programlama öğrenim düzeylerine etkisini içeren araştırmalar incelendiğinde ise alanyazındaki çalışmalarla bu araştırmanın benzer sonuçlara ulaştığı görülmektedir (Osman ve diğerleri, 2012; Solmaz, 2014; Demir, 2015; Yıldırım, 2016; Kayabaşı,

2016; Yiğit, 2016; Yüksel, 2017; Dinçer, 2018; Şimşek, 2018). Bu sonuçlar dikkate alınarak programlama öğretiminin Code.org platformu ile gerçekleştirilmesinin programlama öğrenim düzeyleri üzerinde daha etkili olduğu sonucuna ulaşılabilir. Alanyazında programlama eğitimi sürecinde oyunlaştırmanın öğrencilerin programlama öğrenim düzeyleri üzerinde etkili olabileceğini ifade eden çalışmalar da yer almaktadır (Yue ve Wan, 2015; Yıldırım, 2016; Yükseltürk ve Altıok, 2016; Alan, 2017). Code.org platformunun da programlama öğrenim düzeyini olumlu yönde etkilemesinde, yapısının hareketli ve oyun temelli olmasının etkili olduğu gösterilebilir. Derslerde çoklu ortam araçlarının kullanımına dair alanyazında yer alan çalışmalarda, öğrenci başarısının arttığı ifade edilmiştir (Yaman, 2005). Code.org platformunun başarılı olması da bu sonuçla örtüşmektedir. Ayrıca bilgi işlemsel düşünme becerilerinin alt boyutlarından biri olan problem çözme becerilerinin programlama öğrenimi üzerinde etkisi dikkate alındığında görsel programlama platformu olan Code.org platformunun da bilgi işlemsel düşünme becerisini olumlu yönde etkilediği, bilgi işlemsel düşünme becerilerinin yüksek olmasının programlama öğrenim düzeyini geliştirmiş olabileceği söylenebilir (Ozoran ve diğerleri, 2012; Çankaya ve diğerleri, 2017; Özcan ve diğerleri, 2017; Oluk ve diğerleri, 2018; Dizman, 2018; Erdem, 2018; Vatansever, 2018). Gorman ve Bourne'in (1983) araştırmasında programlama öğretiminin öğrencilerin kural öğrenimine katkısı (Akpınar ve Altun, 2014); Durak ve Samur'un (2018) araştırmasında da programlama öğretim uygulamalarının hayat boyu öğrenme yeterliklerinin geliştirilmesine olumlu etkisi ifade edilmektedir. Buradan hareketle Code.org platformunun kural öğrenimine ve hayat boyu öğrenme becerisine katkısından bahsedilebilir.

5.1.2. Code.org platformunun 6. sınıf öğrencilerinin programlama öğrenimlerinin kalıcılık düzeylerine etkisi

- Programlama öğretiminden beş hafta sonra gerçekleştiren ölçümdeki sonuçları (PÖDA İzleme) PÖDA son test sonuçları kontrol altına alınarak karşılaştırdığımızda anlamlı bir farklılaşma gözlenmiş, Code.org platformunun, öğrencilerin programlama öğrenimlerinin kalıcılığında normal yöntemlere nazaran daha etkili olduğu sonucuna ulaşılmıştır. Alanyazında Code.org platformunun programlama öğreniminin kalıcılık düzeyine etkisini içeren bir çalışmaya rastlanılmamıştır. Genel olarak görsel programlama ortamlarının programlama öğreniminin kalıcılık düzeyine

etkisini içeren arařtırmalar incelendiğinde ise alanyazındaki çalışmaların bu arařtırma sonucunu desteklediğı görölmektedir (Yıldırım, 2016; Yüksel, 2017). Bu bulgudan yola çıkılarak programlama öğretiminin Code.org platformu ile gerçekleştirilmesinin programlama öğreniminin kalıcılık düzeyi üzerinde daha etkili olduğı sonucuna ulaşılabilir. Ayrıca platformun oyun teması üzerine kurulu olduğı düşünöldüğünde bu durumun da kalıcılığa olumlu etki ettiğı görüşü savunulabilir (MEGEP, 2007; Kılınçpınar, 2014; Savaş, 2014; Yıldırım, 2015; Koka, 2018).

5.1.3. Code.org platformunun deney ve kontrol grubunda yer alan öğrencilerin EBODKÖY tutumlarına etkisi

- Deney ve kontrol grubunda yer alan öğrencilerin uygulama öncesi tutum oranlarının (EBODKÖYTÖ Ön Test) farklı olmadığı sonucuna varılmıştır.
- Uygulamanın ardından gerçekleştirilen ölçme sonuçlarını (EBODKÖYTÖ Son Test) ön test sonuçlarıyla karşılařtırdığımızda ise kontrol grubunda anlamlı bir değıřime rastlanılmazken deney grubunda anlamlı bir farka rastlanılmış, öğrencilerde olumlu yönde değıřiklik gözlenmiştir.
- Ayrıca arařtırma probleminin tespiti için yapılan grup karşılařtırmasında deney grubu lehine anlamlı bir farklılık tespit edilmiştir. Bu durumda Code.org platformunun EBODKÖY tutumlarına normal yöntemlere nazaran daha olumlu etkiler meydana getirdiğı kanısına varılmıştır. Bu kanı alanyazındaki Code.org platformunun kullanıldığı deneysel çalışmalarca desteklenmektedir (Patan, 2016; Yıldız, 2018). Ayrıca diğerk görsel programlama ortamlarının öğrencilerin tutumlarına etkisini içeren arařtırmalar incelendiğinde ise benzer şekilde alanyazındaki çalışmalarla bu arařtırma sonucu paralellik göstermektedir (Genç ve Karaguş, 2011; Ozoran ve diğerkleri, 2012; Yıldırım, 2016; Alan, 2017; Alan ve Tařdemir, 2017; Saygıner, 2017; Dinçer, 2018). Bu sonuçlar dikkate alınarak programlama öğretiminin Code.org platformu ile gerçekleştirilmesinin öğrencilerin tutumları üzerinde daha etkili olduğı sonucuna ulaşılabilir. Alanyazında programlama eğitimi sürecinde oyunlařtırmanın öğrencilerin tutumları üzerinde etkili olabileceğini ifade eden çalışmalar da yer almaktadır (Yükseltürk ve Altıok, 2016; Yıldırım, 2016; Alan, 2017). Code.org platformunun da tutumu olumlu yönde etkilemesinde, yapısının hareketli ve oyun temelli olmasının etkili olduğı gösterilebilir.

5.1.4. Deney gurubu öğrencilerinin Code.org platformuna yönelik görüşleri

- Deney grubu öğrencilerinin uygulama sonrası platforma yönelik görüşleri incelenmiştir. Öğrencilerin tamamı Code.org platformunun öğrenmelerini olumlu olarak etkilediğini ifade etmiştir. Öğrencilerin çoğunluğu görüşlerini açıklarken platformun görselliğini ve kendilerinin kullanmasını ön plana almışlardır. Bu sonuç, öğrencilerin bilgisayar destekli öğrenme ortamlarında daha başarılı olabileceğini ifade eden (Kamacı ve Durukan, 2012; Akpınar, 2006; Gürbulak, 2013; Kapucu, 2017; Karabulutlu, 2018) araştırma sonuçlarıyla benzerlik göstermektedir.
- Öğrencilerin tamamı Code.org platformunun öğrenmelerini daha kalıcı hale getirdiklerini bildirmiştir. Buna gerekçe olarak da yüksek oranla görerek ve yaparak öğrenebilmelerini göstermişlerdir. Bu sonucu araçlarla desteklenen bir öğretimin normal metotların yer aldığı ortamlara nazaran daha etkili olduğu ve öğrenmenin kalıcılığı açısından çok önemli olduğunu belirten (Korkmaz, 1997; Alan ve Taşdemir, 2016) araştırma sonuçlarının desteklediği görülmektedir.
- Öğrenciler Code.org platformunu beğendiklerini dile getirirken platformun öğrencilere kendi hesaplarını kullanabilme, öğrenmelerini takip edebilme gibi olanaklar sunması öne çıkan taraflar olmuştur. Ayrıca sürecin eğlenceli olması, aşamalı ilerlemesi, çeşitli bölümleri içermesi belirtilen diğer görüşlerdendir. Görsel programlama ortamları ile programlama öğretiminin, öğrenme ortamının eğlenceli olmasına katkı sağladığına (Lopez ve diğerleri, 2016) yönelik çalışmalar vardır. Code.org platformu da bu çerçevede değerlendirilebilir.
- Öğrencilerin tamamına yakını süreç içerisinde karşılaştıkları problem olarak platformun bazı bölümlerindeki zorlanışlarını ifade etmişlerdir. Bu durumda yüksek oranda olumsuz hisse kapıldıklarını, bu durumla arkadaşlarından yardım alarak başa çıktıklarını belirtmişlerdir. Bu sonuç görsel programlama ortamlarının sınıf içi etkileşimini artırdığını (Maloney ve diğerleri, 2010) ve işbirlikli çalışmayı olumlu yönde etkilediğini (Taylor ve diğerleri, 2010) ifade eden araştırmaların sonuçlarıyla paralellik gösterdiği söylenebilir.
- Öğretimin tekrarlanması halinde istenen değişiklikler konulu maddede öğrenciler yüksek oranda bir değişiklik istemediklerini iletmışlerdir. Kodlama dilinin Türkçe olması, zorlanılan bölümlerin yer almaması gibi istekler de dile getirilmiştir.

- Öğrencilerin tamamı Code.org platformu ile kodlama öğrenimini başkalarına tavsiye edebileceklerini belirtmişlerdir. Bu görüşlerine gerekçe olarak ise platformun çok güzel olmasını, öğretimi daha eğlenceli, daha akılda kalıcı, daha kolay, daha öğretici ve daha hızlı hale getirmesini göstermişlerdir. Bu sonuç görsel programlama ortamları ile programlama öğreniminin kolay olduğunu ve programlama öğrenim düzeyini arttırdığı (Maloney ve diğerleri, 2010; Genç ve Karaguş, 2011; Özcan ve diğerleri, 2017) sonucuna ulaşan araştırmaların sonuçlarıyla benzerlik gösterdiği söylenebilir.
- Öğrencilerin tamamı kodlama ile kendi oyunlarını oluşturmayı istediklerini belirtmişlerdir. Öğrencilerin tamamına yakını bu isteklerini çok çalışmaları halinde gerçekleştirebileceklerini ifade etmişlerdir. Görsel programlama ortamlarının öğrencilerin programlamaya karşı olan isteklerini arttırdığına değinen çalışmalarla bu açıdan bir benzerlik göze çarpmaktadır (Gülbahar ve Kalelioğlu, 2014; Kobsiripat, 2014).
- Kodlama alanında merak edilenler konulu maddede öğrenciler; genel olarak merak ettikleri bir şey olmadıklarını söylemişlerdir. Bu öğretimin her yerde (yurtiçi-yurtdışı) gerçekleştirilme durumu, oyunların nereden, nasıl yapıldığı, dilin Türkçe olmaması merak edilen konular olarak ifade edilmiştir.
- Öğrencilerin çoğunluğu kodlamayla ilgili bir iş alanında çalışmak istediklerini belirtmiş, bu isteklerine gerekçe olarak da kodlamayı sevmelerini, güzel ve eğlenceli bulmalarını göstermişlerdir. Görsel programlama ortamlarının öğrencilerin programlamaya karşı olan isteklerini arttırdığı sonucuna ulaşan çalışmalar bu sonucu desteklemektedir (Gülbahar ve Kalelioğlu, 2014; Kobsiripat, 2014).
- Öğrencilerin tamamı Code.org platformunun BTY dersine karşı olan tutumlarını olumlu yönde değiştirdiğini; dersi artık daha çok sevdiklerini, dersin daha güzel, daha zevkli ve daha eğlenceli bir hal aldığını ifade etmişlerdir. Bu sonuç da görsel programlama ortamlarının öğrencilerin tutumlarını olumlu etkilediğini belirten araştırmalarca desteklenmektedir (Genç ve Karaguş, 2011; Fowler, 2012; Ozoran ve diğerleri, 2012; Aslan, 2014; Yıldırım, 2016; Patan, 2016; Alan, 2017; Alan ve Taşdemir, 2017; Saygıner, 2017; Dinçer, 2018; Yıldız, 2018).

- Ayrıca Code.org platformunun çok beğenilmesi ve her okulda kullanılmasının gerekliliği, tekrarlanan ve eklenen görüşlerdendir.

5.2. Öneriler

Araştırma kapsamında erişilen bulgu ve sonuçlardan hareketle birtakım öneriler geliştirilmiştir. Bu öneriler, “Uygulayıcılara Yönelik Öneriler”, “Araştırmacılara Yönelik Öneriler” ve “İlgili Kurumlara Yönelik Öneriler” başlıkları altında toplanmıştır.

5.2.1. Uygulayıcılara yönelik öneriler

- BTY dersi müfredatında yer alan programlama öğretiminde Code.org platformundan yararlanılabilir.
- Öğitmenler Code.org platformundaki kurs planlamalarını öğrencileri için aşamalılık ilkesi çerçevesinde basitten karmaşığa doğru yapabilir.
- Kurs öncesi platformun tanıtımı (kullanımı, kapsamı, takip mekanizması, çevrimiçi yapısı, sertifikasyon ödülü vb.) yapılabilir.
- Süreç içerisinde işbirlikli öğrenmeye yönelik öğrenciler arası grup çalışmalarına (akran öğretimi, ikili denetim, karşılıklı sorgulama vb. yöntemler) yer verilebilir.
- Eğitim çerçevesinde harmanlanmış (hibrid-karışık-karma) öğrenme ortamlarına ilişkin metotlara yer verilebilir.
- Süreç içerisinde konuya ilişkin problemlerin çözümüne dair kontrollü rekabet ortamları (farklı yollarla çözebilme, hızlı çözebilme vb.) oluşturulabilir.

5.2.2. Araştırmacılara yönelik öneriler

- Var olan çalışmalar incelendiğinde, Code.org platformuna ilişkin veya bu platformu içeren çalışmaların sayısının az olduğu görülmüştür. Bu alanda daha fazla deneysel çalışmalar yapılması önerilebilir.
- Code.org platformu ile programlama eğitimi gerçekleştiren öğretmenlere yönelik deneysel çalışmalar yapılması önerilebilir.

- Bu araştırma 6. sınıf öğrencileri ile gerçekleştirilmiştir. Aynı çalışma farklı sınıf düzeylerindeki öğrencilere de uygulanarak genişletilebilir.
- Farklı sosyoekonomik özelliklere sahip daha geniş örneklemeler üzerinde çalışmalar yapılabilir.
- Code.org platformunun ilkokul kademesindeki durumuna yönelik sınıf öğretmenleri ile araştırmalar yapılabilir.

5.2.3. İlgili kurumlara yönelik öneriler

- Code.org platformu üzerinden programlama eğitimini yürütecek eğitim kurumları, teknolojik materyallerini ve laboratuvarlarını amaca uygun olarak düzenlenmelidir.
- MEB, BTY dersi başta olmak üzere ilgili müfredatlarını Code.org platformunun derslerde kullanılabilmesi için direkt olarak uyumlu hale getirebilir.
- Eğitim Fakültelerindeki ilgili öğretmen adayları programlama eğitime yönelik Code.org platformunu içeren uygulamalı çalışmalara tabi tutulabilir.

KAYNAKÇA

- Akpınar, E. (2006). *Fen öğretiminde soyut kavramların yapılandırılmasında bilgisayar desteği: Yaşamımızı yönlendiren elektrik ünitesi*. Dokuz Eylül Üniversitesi, İzmir, Erişim adresi: <https://dspace.deu.edu.tr/xmlui/handle/12345/6913>
- Akpınar, Y. (1999). *Bilgisayar Destekli Öğretim ve Uygulamalar*. Ankara: Anı Yayıncılık.
- Akpınar, Y. & Altun, A. (2014). Bilgi toplumu okullarında programlama eğitimi gereksinimi. *İlköğretim Online*, 13(1).
- Aksoy, N. (2017). *Bilişim Teknolojileri ve Yazılım 6. Sınıf Ders Kitabı*. Erişim adresi: <http://kesfetprojesi.org/files/uploads/b73ccdaa0e22abf77e29ed8e67b9bba40.pdf>
- Alan, D. (2017). *Dijital oyun tabanlı yaklaşım ile yazılım geliştirme öğretimi* (Yayınlanmamış yüksek lisans tezi). Selçuk Üniversitesi Fen Bilimleri Enstitüsü: Konya (No:467392).
- Alan, D. & Taşdemir, Ş. (2016). An Example Study For Teaching Algorithm Structure Using Games In Computer Sciences, *International Conference on Education in Mathematics, Science & Technology (ICEMST)*, Bodrum/Türkiye.
- Alan, D. & Taşdemir, Ş. (2017). A Training Game Design For Helping The Coding Learning Program Of The Computer Department Students, *International Conference on Education in Mathematics, Science & Technology (ICEMST)*, Ephesus - Kuşadası, Türkiye.
- Aslan, Ü. (2014). *Olasılık Öğreniminin Oyun Programlama Yöntemiyle Geliştirilmesi* (Yayınlanmamış yüksek lisans tezi). Boğaziçi Üniversitesi Fen Bilimleri Enstitüsü: İstanbul (No:356973).
- Aytekin, A., Çakır, F. S., Yücel, Y. B., & Kulaöz, İ. (2018). Geleceğe Yön Veren Kodlama Bilimi ve Kodlama Öğrenmede Kullanılabilecek Bazı Yöntemler. *Avrasya Sosyal ve Ekonomi Araştırmaları Dergisi*, 5(5), 24-41.
- Baldwin, L. P. & Kuljis, J. (2000). Visualization techniques for learning and teaching programming. *Journal of Computing and Information Technology*, 8(4), 285-291.

- Bayırtepe, E., & Tüzün, H. (2007). The effects of game based learning environments on students' achievement and self efficacy in a computer course. *Hacettepe University Journal of Education*, 33, 41-54
- Baz, F. Ç. (2018). Çocuklar İçin Kodlama Yazılımları Üzerine Karşılaştırmalı Bir İnceleme. *Curr Res Educ*, 4(1), 36-47.
- businessdictionary.com (2019). [Erişim adresi: <http://www.businessdictionary.com/definition/computer-programming.html>], Erişim tarihi: 15.01.2019.
- Bozan, N. (2014). *Okul öncesi eğitimde oyunun öğretmen görüşlerine göre değerlendirilmesi* (Yayınlanmamış yüksek lisans tezi). Atatürk Üniversitesi Eğitim Bilimleri Enstitüsü: Erzurum (No: 381625).
- Büyüköztürk, Ş. (2006). *Sosyal bilimler için veri analizi el kitabı* (6. Baskı). Ankara: Pegem A Yayıncılık.
- Büyüköztürk, Ş. (2015). *Sosyal bilimler için veri analizi el kitabı: İstatistik, araştırma deseni. SPSS uygulamaları ve yorum* (Gözden geçirilmiş yirmi birinci baskı). Ankara: Pegem Yayıncılık.
- Büyüköztürk, Ş. (2017). *Sosyal bilimler için veri analizi el kitabı* (23. Baskı). Ankara: Pegem Akademi.
- Cevahir, H. & Özdemir, M. (2017). Programlama öğretiminde karşılaşılan zorluklara yönelik öğretmen görüşleri ve çözüm önerileri. *Conference: 11. Uluslararası Bilgisayar ve Öğretim Teknolojileri Eğitimi Sempozyumu sunulan bildiri*. İnönü Üniversitesi, Malatya. Erişim adresi: <http://icits2017.inonu.edu.tr/dosya/1493636695061912000.pdf>
- Clements, D. H. & Gullo, D. F. (1984). *Effects of computer programming on young children's cognition*. *Journal of educational psychology*, 76(6), 1051.
- Code.org (2019). [Erişim adresi: <https://code.org/about>].
- Code.org Translations (2019). [Erişim adresi: <https://crowdin.com/project/codeorg>], Erişim tarihi: 17.01.2019.

- Çalık, T. & Sezgin, F. (2005). Küreselleşme, bilgi toplumu ve eğitim. *Kastamonu Eğitim Dergisi*, 13(1), 55-66.
- Çankaya, S., Durak, G. & Yünkül, E. (2017). Robotlarla programlama eğitimi: öğrencilerin deneyimlerinin ve görüşlerinin incelenmesi. *Turkish Online Journal of Qualitative Inquiry*, 8(4), 428-445.
- Çatlak, Ş., Tekdal, M. & Baz, F. Ç. (2015). Scratch Yazılımı İle Programlama Öğretiminin Durumu: Bir Doküman İnceleme Çalışması. *Journal of Instructional Technologies ve Teacher Education*, 4(3).
- Çilenti, K. (1985). *Fen eğitimi teknolojisi*. Ankara: Kadioğlu Matbaası.
- Demir, F. (2015). *Programlama öğretiminde eğitsel programlama dilinin farklı kullanımlarının programlama başarısı ve kaygısına etkisi* (Yayınlanmamış doktora tezi). Atatürk Üniversitesi Eğitim Bilimleri Enstitüsü: Erzurum (No:429631).
- Demirel, Ö. (1999). *Öğretme Sanatı*. Ankara: Pegem Yayınları.
- Demirel, Ö., Seferoğlu, S. S. & Yağcı, E. (2003). *Öğretim Teknolojileri ve Materyal Geliştirme*, 4. Baskı, Ankara: Pegem A Yayıncılık.
- Demirer, V. & Sak, N. (2016). Dünyada ve Türkiye'de programlama eğitimi ve yeni yaklaşımlar. *Eğitimde Kuram ve Uygulama*, 12(3), 521-546.
- Dinçer, A. (2018). *6.sınıf öğrencilerine scratch ve kodu game lab programlama dillerinin öğretiminde öğrencilerin tutum, öz yeterlilik ve akademik başarılarının karşılaştırılması* (Yayınlanmamış yüksek lisans tezi). Dokuz Eylül Üniversitesi Eğitim Bilimleri Enstitüsü: İzmir (No:512965).
- Dizman, A. (2018). *Kodlama, robotik, 3D tasarım ve oyun tasarımı eğitiminin 11-14 yaş grubu öğrencilerinin problem çözme becerileri ve üstbilişsel farkındalık düzeyine etkisi* (Yayınlanmamış yüksek lisans tezi). Bahçeşehir Üniversitesi Eğitim Bilimleri Enstitüsü: İstanbul (No:523935).
- Durak, H. Y. & Samur, Y. (2018). Kodlama Eğitiminin Öğretmen Adaylarının Yaşam Boyu Öğrenme Yeterliliklerinin Geliştirmesine Katkısının İncelenmesi. *Ege Eğitim Teknolojileri Dergisi*, 2(2) 55-67.

EBA (2019), [Erişim adresi: <https://www.eba.gov.tr/kod>].

Educational Testing Service (ETS) (2007). i-Skills: Digital Transformation A Framework for ICT Literacy. Erişim adresi: http://www.ets.org/Media/Tests/Information_and_Communication_Technology_Literacy/ictreport.pdf

Ekiz, D. (2003). *Eğitimde Araştırma Yöntem ve Metotlarına Giriş*. Ankara.

Erdem, E. (2018). *Blok tabanlı ortamlarda programlama öğretimi sürecinde farklı öğretim stratejilerinin çeşitli değişkenler açısından incelenmesi* (Yayınlanmamış yüksek lisans tezi). Başkent Üniversitesi Eğitim Bilimleri Enstitüsü: Ankara (No:509354).

Erekmekçi, M. & Fidan, Ş. (2012). Oyunun Tasarım Platformları: Oyunun Eğitim ve Kültüre Etkisi. *Journal of Life Sciences*, 1(1). [Erişim adresi: <http://www.yasambilimleridergisi.com/makale/pdf/1356289798.pdf>, Erişim Tarihi: 19.12.2016].

Fessakis, G., Gouli, E. & Mavroudi, E. (2013). Problem solving by 5–6 years old kindergarten children in a computer programming environment: A case study. *Computers & Education*, 63, 87-97.

Fowler, A. (2012, Kasım). Enriching student learning programming through using Kodu. *3rd Annual Conference of Computing and Information Technology Research and Education New Zealand (CITRENZ2012)'da sunulan bildiri*. Christchurch, New Zealand. Erişim adresi: http://www.citrenz.ac.nz/conferences/2012/pdf/2012CITRENZ_Fowler01Kodu.pdf

Fraillon, J. & Ainley, J. (2011). *The IEA International Study of Computer and Information Literacy (ICILS)*. International Association for Evaluation of Educational Achievement, Erişim adresi: http://www.icdl.com.au/assets/documents/ACER_whitepaper.pdf

Genç, Z. (2003). “C” ile programlamaya giriş. [Adobe Acrobat Reader sürümü]. Erişim adresi: http://web.firat.edu.tr/enfders/file/c_programlama/c.pdf

Genç, Z. & Karakuş, S. (2011). *Tasarımla Öğrenme: Eğitsel Bilgisayar Oyunları Tasarımında Scratch Kullanımı*. 5th International Computer & Instructional Technologies Symposium (ICITS). Elazığ, Türkiye.

- Gorman Jr, H. & Bourne Jr, L. E. (1983). Learning to think by learning LOGO: Rule learning in third-grade computer programmers. *Bulletin of the Psychonomic Society*, 21(3), 165-167.
- Gülbahar, Y. & Kalelioğlu, F. (2014). The effects of teaching programming via Scratch on problem solving skills: A discussion from learners' perspective. *Informatics in Education-An International Journal*, 13(1), 33-50.
- Gürbulak, N. (2013), *Okul Öncesi Öğrencilerine Renk Kavramını Kazandırmada Geçerli ve Yeterli Bir Eğitsel Yazılım Geliştirme ve Ölçme Çalışması* (Yayınlanmamış yüksek lisans tezi). Bahçeşehir Üniversitesi Fen Bilimleri Enstitüsü: İstanbul (No:341262).
- Hava, K. (2016). *Tasarım tabanlı öğrenme ortamında bilgisayar oyunu geliştirme etkinliğinin öğretim sürecinde kullanılması* (Yayınlanmamış doktora tezi). Gazi Üniversitesi Eğitim Bilimleri Enstitüsü: Ankara (No:429379).
- Irmak, A. Y., & Erdoğan, S. (2016). Ergen ve genç erişkinlerde dijital oyun bağımlılığı: güncel bir bakış. *Türk Psikiyatri Dergisi*, 27(2), 128-37.
- ISTE-NETS-S (1998, 2007). The International Society for Technology in Education (ISTE), National Educational Technology Standards for Students. Erişim adresi: <http://www.iste.org/standards/nets-for-students.aspx>
- ISTE (2018). ISTE Standards for Students. Erişim adresi: <http://www.iste.org/standards/for-students>
- Kablan, Z. (2010). Öğretim sürecinde bilgisayara dayalı alıştırma amaçlı oyun kullanılmasının eğitim fakültesi öğrencilerinin akademik başarısına etkisi. *Kuram ve Uygulamada Eğitim Bilimleri*, 10(1), 335-364.
- Kalaycı, Ş. (2010). *SPSS uygulamalı çok değişkenli istatistik teknikleri* (Vol. 5). Ankara, Türkiye: Asil Yayın Dağıtım.
- Kalelioğlu, F. (2015). A new way of teaching programming skills to K-12 students: Code.org. *Computers in Human Behavior*, 52, 200-210.

- Kamacı E, Durukan E. (2012). Araştırma Görevlilerinin Eğitimde Tablet Bilgisayar Kullanımına İlişkin Görüşleri Üzerine Nitel Bir Araştırma (Trabzon Örneği), *Uluslararası Türkçe Edebiyat Kültür Eğitim Dergisi*, 3203-215.
- Kapucu, T. (2018). *Bilgisayar destekli eğitimin 8. sınıf öğrencilerinin permütasyon-kombinasyon-olasılık başarısına ve öğrencilerin bilgisayar destekli eğitime ilişkin tutumlarına etkisi* (Yayınlanmamış yüksek lisans tezi). Orta Doğu Teknik Üniversitesi Eğitim Bilimleri Enstitüsü: Ankara (No:463551).
- Karabulutlu, L. (2018). *Okul öncesi fen eğitiminde analogilerin ve bilgisayar destekli eğitimin akademik başarı açısından değerlendirilmesi* (Yayınlanmamış yüksek lisans tezi). Kafkas Üniversitesi Eğitim Bilimleri Enstitüsü: Kars (No:519638).
- Karagöz, Y. (2010). Nonparametrik Tekniklerin Güç ve Etkinlikleri. *Elektronik Sosyal Bilimler Dergisi*, 33(33).
- Karamustafaoğlu, O. & Kaya, M. (2013). Eğitsel oyunlarla ‘yansıma ve aynalar’ konusunun öğretimi: Yansımali koşu örneği. *Araştırma Temelli Etkinlik Dergisi*, 3(2), 41-49.
- Karasar, N. (2004). *Bilimsel Araştırma Yöntemi*. (12.Basım), Ankara: Nobel yayıncılık.
- Karasar, N. (2011). *Bilimsel Araştırma Yöntemi*. (22., 26. bs), Ankara: Nobel Yayımcılık.
- Kayabaşı, E. (2016). *Öğretmen adaylarının alice deneyimi: 3b ortamda programlama* (Yayınlanmamış yüksek lisans tezi). Uludağ Üniversitesi Eğitim Bilimleri Enstitüsü: Bursa (No:429742).
- Keçeci, G., Alan, B. & Zengin, F. K. (2016). Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeği: Geçerlilik ve Güvenirlik Çalışması. *Educationsciences*, 11(4), 184-194.
- Kert, S. B. & Uğraş, T.(2009). *Programlama eğitiminde sadelik ve Eğlence: Scratch Örneği*. The First International Congress of Educational Research, Çanakkale, Türkiye.
- Kesici, T. & Kocabaş, Z. (2007). *Bilgisayar 2 Ders Kitabı*. (2. Baskı), Ankara: Semih Ofset.
- Kılınçpınar, P. (2014). [Erişim adresi: http://pelinkilincpınar.blogspot.com/2014_09_01_archive.html], Yazı tarihi: 02.09.2014, Erişim tarihi: 16.01.2019.

- Kobsiripat, W. (2015). Effects of the media to promote the scratch programming capabilities creativity of elementary school students. *Procedia-Social and Behavioral Sciences*, 174, 227–232. Erişim adresi: <https://www.sciencedirect.com/science/article/pii/S1877042815007028?via%3Dihub>
- Koka, V. (2018). *Sosyal bilgiler dersinde kullanılan bilgisayar destekli eğitsel oyunların öğrencilerin ders başarısına olan etkisi* (Yayınlanmamış yüksek lisans tezi). İnönü Üniversitesi Eğitim Bilimleri Enstitüsü: Malatya (No:505026).
- Korkmaz, H. (1997). *İlkokul fen öğretiminde araç-gereç kullanımı labaratuvar uygulamaları açısından öğretmen yeterlikleri* (Yayınlanmamış yüksek lisans tezi). Hacettepe Üniversitesi Eğitim Bilimleri Enstitüsü: Ankara (No:63847).
- Küçüköğlu, Ö. (2014). *21.yüzyıl becerileri için en etkin oluşumlar: “Kod okur-yazarlığı” ve “FLL”*. [Erişim adresi: <https://www.weebly.com/>, Erişim Tarihi:14.09.2015].
- Lopez, J. M., Gonzalez, M. R. & Cano, E. V. (2016). Visual programming languages integrated across the curriculum in elementary school: A two year case study using “Scratch” in five schools. *Computers & Educations*, 97, 129-141. Erişim adresi:<https://www.sciencedirect.com/science/article/pii/S0360131516300549?via%3Dihub>
- Maloney, J., Resnick, M., Rusk, N., Silverman, B. & Eastmond, E. (2010). The Scratch programming language and environment. *Trans. Comput. Educ.*, 10(4), 1-15. Erişim adresi: <https://ieeexplore.ieee.org/document/6504380>
- MEGEP (2007). *Çocuk Gelişimi ve Eğitimi, Oyun Etkinliği – I*. Ankara.
- Miles, M. B., Huberman, A. M. (1994). *Qualitative data analysis: An expanded sourcebook*. (2nd Edition). California: SAGE Publications.
- Miles, M., Huberman, A. M., & Saldana, J. (2014). *Qualitative data analysis: A methods sourcebook*. European Journal of Science Education. Los Angeles: Sage Puplicaton, Thousand Oaks.
- NAACE (2012). Draft Naace Curriculum Framework Information and Communication Technology (ICT) Key Stage 3. Erişim adresi: <http://www.naace.co.uk/ks3ictcurriculum>

- NCCA (National Council for Curriculum and Assessment. (2007). ICT Framework: A Structured Approach to ICT in Curriculum and Assessment (Revised Framework). Erişim adresi: <http://www.ncca.ie/uploadedfiles/publications/ict%20revised%20framework.pdf>
- Oluk, A., Korkmaz, Ö. & Oluk, H. A. (2018). Scratch'ın 5. Sınıf Öğrencilerinin Algoritma Geliştirme ve Bilgisayarca Düşünme Becerilerine Etkisi. *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, 1-1.
- Onuncu Kalkınma Planı (2014-2018), Türkiye Cumhuriyeti Kalkınma Bakanlığı, Erişim adresi: http://www.kalkinma.gov.tr/Lists/Kalkinma%20Planlar/Attachments/12Onuncu_Kalkinma_Planı.pdf, [Erişim Tarihi: 10 Haziran 2018].
- Osman, M. A., Zakaria, M. N., Loke, S. P. & Downe, A. G. (2012, Aralık). *Secondary students' perfectionism and their response to different programming learning tools. Paper presented at the Humanities, Science and Engineering (CHUSER). 2012 IEEE Colloquium on* sunulan bildiri. Kota Kinabalu, Malezya. Erişim adresi: <https://ieeexplore.ieee.org/document/6504380>
- Ozoran, D., Çağıltay, N. & Topallı, D. (2012). Using Scratch In Introduction to Programming Course for Engineering Students. *In 2nd International Engineering Education Conference (IEEC2012)*, (pp. 125-132).
- Özcan, S., Ergün, K., Köse, Ö., Emir, N. & Gezgın, D. (2017). Bilgisayar Programlama Eğitiminde Scratch Programı Kullanımına İlişkin Lise Öğrencilerinin Görüşleri. *2nd International Scientific Researches Congress on Humanities and Social Sciences (IBAD-2017)*, İstanbul.
- Özmen, B. & Altun, A. (2014). Undergraduate Students' Experiences in Programming: Difficulties and Obstacles. *Turkish Online Journal of Qualitative Inquiry*, 5(3), 9-27.
- Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. Basic Books, Inc..
- Patan, B. (2016). *Okul öncesi kodlama öğretim programının geliştirilmesi* (Yayınlanmamış yüksek lisans tezi). Bahçeşehir Üniversitesi Eğitim Bilimleri Enstitüsü: İstanbul (No:436081).

- Ramadhan, H. A. (2000). Programming by discovery. *Journal of Computer Assisted Learning*, 16, 83-93.
- Resnick, M., Maloney, J., Hernandez, A., Easmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B. & Kafai, Y. (2009). Scratch: programming for all. *Communications of the ACM*, 52(11), 60-67. Eriřim adresi: <https://dl.acm.org/citation.cfm?doid=1592761.1592779>
- Saeli, M., Perrenet, J., Jochems, W. M. & Zwaneveld, B. (2011). Teaching programming in secondary school: a pedagogical content knowledge perspective. *Informatics in Education*, 10(1).
- Sarıçayır, H. (2007). *Kimya Eđitiminde Kimyasal Tepkimelerde Denge Konusunun Bilgisayar Destekli ve Laboratuar Temelli Öğretiminin Öğrencilerin Kimya Başarılarına, Hatırlama Düzeylerine ve Tutumlarına Etkisi* (Yayınlanmamış yüksek lisans tezi). Marmara Üniversitesi Eğitim Bilimleri Enstitüsü: İstanbul (No:206212).
- Savaş, E. (2014). *Oyunla Öğretim Yöntemi Uygulamasının Başarı ve Kalıcılık Üzerindeki Etkisi (Deneyisel Bir Çalışma)* (Yayınlanmamış yüksek lisans tezi). Onsekiz Mart Üniversitesi Eğitim Bilimleri Enstitüsü: Çanakkale (No: 356358).
- Saygıner, Ş. (2017). *Blok tabanlı görsel ve metin tabanlı programlama öğretimlerinin eriş, mantıksal düşünme ve motivasyona etkileri* (Yayınlanmamış yüksek lisans tezi). Hacettepe Üniversitesi Eğitim Bilimleri Enstitüsü: Ankara (No:454912).
- Saygıner, Ş., & Tüzün, H. (2018). Programlama Eğitimi Üzerine Bir İnceleme: Yaşanan Zorluklar, Mevcut Uygulamalar ve Güncel Yaklaşımlar. *Eđitim Teknolojileri Okumaları*, 671-682.
- Sayın, Z. & Seferođlu, S. S. (2016). Yeni Bir 21. Yüzyıl Becerisi Olarak Kodlama Eğitimi ve Kodlamanın Eğitim Politikalarına Etkisi. *Akademik Biliřim 2016*, Adnan Menderes Üniversitesi, Aydın
- Seferođlu, S. (2007). “İlköğretim Bilgisayar Dersi Öğretim Programı: Eleřtirel Bir Bakıř ve Uygulamada Yaşanan Sorunlar”. *Eđitim Arařtırmaları Dergisi*, Ankara.
- Solmaz, E. (2014). *Programlama dili öğretiminde Alice yazılımının ders başarısı, eleřtirel düşünme ve problem çözme becerileri ile üstbiliřsel farkındalık düzeyine etkisi*

- (Yayınlanmamış doktora tezi). Gazi Üniversitesi Eğitim Bilimleri Enstitüsü: Ankara (No:381475).
- Sümer, N., Demirutku, K. & Özkan, T. (2005). *Araştırma teknikleri*. İstanbul: Morpa Kültür Yayınları, 40.
- Szlávi, P. & Zsakó, L. (2006, November). *Programming versus application*. In Issep (pp. 48-58).
- Şimşek, E. (2018). *Programlama öğretiminde robotik ve scratch uygulamalarının öğrencilerin bilgi işlemsel düşünme becerileri ve akademik başarılarına etkisi* (Yayınlanmamış yüksek lisans tezi). Ondokuz Mayıs Üniversitesi Eğitim Bilimleri Enstitüsü: Samsun (No:519321).
- Taşdemir, Ş., & Şüyun, S. B. (2016). Bilgisayar Oyun Tasarımı ve Eğitsellik Kazandırılmasına Yönelik Bir Yaklaşım. *Selçuk-Teknik Dergisi*, 15(2), 113-124.
- Taylor, M., Harlow, A. & Forret, M. (2010). Using a computer programming environment and an interactive whiteboard to investigate some mathematical thinking. *Procedia-Social and Behavioral Sciences*, 8, 561-570. Erişim adresi:<https://linkinghub.elsevier.com/retrieve/pii/S187704281002183X>
- Tekin, A., Akbal, A., Sevinç. B., Ertam, F., Tuzsuzoğlu, H. H., Serhatlıoğlu, İ., Balıkçı, K., Talu, M. F., Çıbuk, M., Özdemir, O., Daş, R., Akbulut, Y. & Genç, Z. (2003), “C” ile *Programlamaya Giriş*. Elazığ.
- Tezbaşaran, A. (2008). *Likert Tipi Ölçek Hazırlama Kılavuzu*. Üçüncü sürüm e-kitap, Mersin Üniversitesi Eğitim Fakültesi Eğitim Bilimleri Bölümü, Eğitimde Ölçme ve Değerlendirme ABD, Mersin.
- Tomei, L. A. (2005). *Taxonomy for the Technology Domain*. USA: Information Science Publishing.
- TTKB, (2012). Bilişim Teknolojileri ve Yazılım Dersi Öğretim Programı. https://www.bilgisayarbilisim.net/attachments/bilisimteknolojileri_ortaokul-rar.29815/ adresinden 2017 tarihinde erişilmiştir.

- TTKB, (2017). Bilişim Teknolojileri ve Yazılım Dersi Öğretim Programı. <http://bilgisayarbilim.com/wp-content/uploads/2017/01/Bili%C5%9FimTeknolojileri-ve-az%C4%B1l%C4%B1m-Dersi-%C3%96%C4%9Fretim-Program%C4%B1.pdf> adresinden 2018 tarihinde erişilmiştir.
- Turan, İ. , Akça, M., Küçükkurt, M. (2016). *Temel Kodlama Eğitimi 1-2*. İstanbul: Pusula Yayıncılık.
- Turgut, M. F. & Baykul, Y. (2012). *Eğitimde ölçme ve değerlendirme*. Pegem Akademi.
- Utting, I., Cooper, S., Kölling, M., Maloney, J. & Resnick, M. (2010). Alice, greenfoot, and scratch--a discussion. *ACM Transactions on Computing Education (TOCE)*, 10(4), 17.
- Vatansever, Ö. (2018). *Scratch İle Programlama Öğretiminin Ortaokul 5. ve 6. Sınıf Öğrencilerinin Problem Çözme Becerisi Üzerindeki Etkisinin İncelenmesi* (Yayınlanmamış yüksek lisans tezi). Uludağ Üniversitesi Eğitim Bilimleri Enstitüsü: Bursa (No:501053).
- Yaman, M. (2005). Solunum Zinciri Konusunda Simülasyonla Desteklenmiş Bir Bilgisayar Programının Öğrenme ve İlgiye Etkisi. *Hacettepe Üniversitesi Eğitim Fakültesi Dergisi*, 29, 222-228.
- Yıldırım, A. & Şimşek, H. (1999). *Sosyal bilimlerde nitel araştırma teknikleri*. Ankara: Seçkin Yayınları.
- Yıldırım, A. & Şimşek, H. (2008). *Sosyal bilimlerde nitel araştırma yöntemleri*. Ankara: Seçkin Yayıncılık.
- Yıldırım, A. & Şimşek, H. (2013). *Sosyal bilimlerde nitel araştırma yöntemleri*. Ankara: Seçkin Yayıncılık.
- Yıldırım, B. (2015). *Eğitsel Oyun ve Dönüt-Düzeltilmenin Öğrenme Düzeyi ve Kalıcılığa Etkisi* (Yayınlanmamış yüksek lisans tezi). Necmettin Erbakan Üniversitesi Eğitim Bilimleri Enstitüsü: Konya (No:407553).
- Yıldırım, E. (2016). Dijital oyun tasarım programlarının eğitimde önemi. *Mesleki Bilimler Dergisi*, 5 (2), 12 – 19. Erişim adresi: <http://dergipark.gov.tr/mbd/issue/34074/377099>

- Yıldırım, E. (2017). *Scratch programlama dili eğitimine yönelik bir mobil uygulamanın geliştirilmesi* (Yayınlanmamış yüksek lisans tezi). Onsekiz Mart Üniversitesi Eğitim Bilimleri Enstitüsü: Çanakkale (No:471031).
- Yıldız, M. (2018). *Oyunlaştırılmış blok tabanlı algoritmik düşünme etkinliklerinin öğrencilerin programlamaya yönelik tutum, katılım ve becerilerine etkisi* (Yayınlanmamış yüksek lisans tezi). Atatürk Üniversitesi Eğitim Bilimleri Enstitüsü: Erzurum (No:513250).
- Yıldız, S. (2018). *Blok tabanlı kodlama ortamında problem çözme süreçlerinin incelenmesi* (Yayınlanmamış yüksek lisans tezi). Karadeniz Teknik Üniversitesi Eğitim Bilimleri Enstitüsü: Trabzon (No:494311).
- Yılmaz, H. (2007). *Örneklem Büyüklüğünün saptanması ve İstatistiksel Testler*. Manisa.
- Yiğit, F. M. (2016). *Görsel programlama ortamı ile öğretimin öğrencilerin bilgisayar programlamayı öğrenmesine ve programlamaya karşı tutumlarına etkisinin incelenmesi* (Yayınlanmamış yüksek lisans tezi). Ondokuz Mayıs Üniversitesi Eğitim Bilimleri Enstitüsü: Samsun (No:442990).
- Yue, W. S., & Wan, W. L. (2015). The Effectiveness of Digital Game for Introductory Programming Concepts. *The 10th International Conference for Internet Technology and Secured Transactions (ICITST-2015)*, [Erişim adresi:<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7412134&tag=1>, Erişim Tarihi: 05.04.2016].
- Yükseltürk, E. & Altıok, S. (2016). Investigation of pre-service information technology teachers' game projects prepared with Scratch. *SDU International Journal of Educational Studies*, 3(1), 59-66. Erişim adresi: <http://dergipark.gov.tr/sduijes/issue/20865/223885>
- Yüksel, S. (2017). *Scratch programı öğretiminde ayrılıp birleşme tekniği kullanımının öğrencilerin derse yönelik tutumuna, akademik başarısına ve kalıcılığa etkisi* (Yayınlanmamış yüksek lisans tezi). Adnan Menderes Üniversitesi Eğitim Bilimleri Enstitüsü: Aydın (No:472231).

EKLER

Ek 1. Araştırmacının Akademik Özgeçmişi

Kişisel Bilgiler			
Adı ve Soyadı	Ümit KARADUMAN		
E-postası/Web Sayfası	umit.karaduman@ogr.deu.edu.tr / umitkaraduman.tr.gg		
Bildiği Yabancı Diller	İngilizce		
Uzmanlık Alanı	Bilişim Teknolojileri		
Öğrenim Bilgileri			
	Üniversite	Bölüm	Yıl
Lisans	Dokuz Eylül Üniversitesi	Bilgisayar ve Öğretim Teknolojileri Öğretmenliği	2013
Tez Başlığı	Code.org Platformunun 6. Sınıf Öğrencilerinin Programlama Öğrenimine Etkisi		
Tez Danışmanı	Prof. Dr. Ercan AKPINAR		
Akademik Eserler			
-			
-			
Alanıyla İlgili Bilimsel Kuruluşlara Üyelikler			
International Journal of Computer Science Education in Schools Eğitim ve Bilim			
Alanıyla İlgili Aldığı Ödüller			
-			

Ek 2. Etik Kurul İzin Belgesi



T.C.
KÜTAHYA VALİLİĞİ
İl Millî Eğitim Müdürlüğü

Sayı : 53490996-44-E.1971902
Konu : Ümit KARADUMAN'ın
Tez Uygulaması

15/02/2017

VALİLİK MAKAMINA

İlgi : a) MEB. Yenilik ve Eğitim Teknolojileri Genel Müdürlüğünün 2012/13 nolu Genelgesi.
b) Dokuz Eylül Üniversitesi Rektörlüğünün 11/01/2017 tarihli ve 0041 sayılı yazısı.

Bakanlığımızın ilgi (a) Genelgesi doğrultusunda, Dokuz Eylül Üniversitesi Rektörlüğünün ilgi (b) yazısında; Eğitim Bilimleri Enstitüsü, Bilgisayar ve Öğretim Teknolojileri Eğitimi Anabilim Dalı Bilgisayar ve Öğretim Teknolojileri Öğretmenliği Yüksek Lisans Programı öğrencisi Ümit KARADUMAN'ın "Code.org Platformunun 6. Sınıf Öğrencilerinin Programlama Öğrenimine Etkisi" konulu tez çalışması kapsamında İlimiz Gediz İlçesi Altınkent Ortaokulu 6.sınıf öğrencilerine uygulama yapma isteği belirtilmektedir.

İl Millî Eğitim Müdür Yardımcısı Hamdi SARIOĞ'ün başkanlığında toplanan değerlendirme komisyonu yapmış olduğu inceleme sonucunda söz konusu anket çalışmasının okullarda uygulanabilir olduğu karar vermiş olup, sportif etkinlikler ile ölçüm yapılması, veli izinlerinin alınması, yaralanmalara karşı tedbir alınması, eğitim- öğretimi aksatmadan, konunun dışına çıkmamaları, bütün sorumluluğun ilgililere ve okul müdürlüğüne ait olmak üzere yukarıda belirtilen anket çalışmasının tamamlandıktan sonra bir örneğinin Müdürlüğümüze verilmek üzere yapılmasını;

Makamlarınızca da uygun görülmesi halinde olurlarınıza arz ederim.

Sabahattin DÜLGER
İl Millî Eğitim Müdürü

OLUR
15/02/2017

Arif YALÇIN
Vali a.
Vali Yardımcısı

İl Millî Eğitim Müdürlüğü/KÜTAHYA
Elektronik Ağ: kutahya.meb.gov.tr
e-posta: stratejigelistirme43@meb.gov.tr

Ayrıntılı bilgi için: Filiz ÖRNEK- VHKİ
Tel: (0 274) 2236241/159
Faks: (0 274) 2236254

Bu evrak güvenli elektronik imza ile imzalanmıştır. <http://evraksorgu.meb.gov.tr> adresinden 9e60-9fa4-31d0-bfa1-d91b kodu ile teyit edilebilir.

Ek 3. EBODKÖYTÖ

Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeği

Sevgili öğrenciler bu test eğitsel bilgisayar oyunları destekli kodlama öğrenimine karşı tutumlarınızı tespit etmek için kullanılabilecek bir ölçme aracı geliştirilmesi amacıyla gerçekleştirilmiştir. Ölçekteki maddeler kodlama öğrenimine karşı istek, bilgisayar oyunlarının derslerde eğitim amaçlı kullanımına yönelik ilgi ve bilgisayarın asosyalleştirmesine yönelik endişeleri kapsayacak şekilde hazırlanmıştır. Ölçeğe cevap verirken aşağıdaki hususlara dikkat ediniz.

1. Ölçekte geçen bilgisayar oyunu sadece bilgisayarda oynanan oyunlar değil tablet, telefon vs. gibi oynanan oyunlar olarak da düşünülmelidir.
2. Kodlama: Bilgisayar ve benzeri düzeneklere bir işlem yaptırmak için verilen komut dizisidir.
3. Sınıf: Cinsiyet: Yaş :

	Maddeler	Kesinlikle Katılmıyorum	Katılmıyorum	Kararsızım	Katılmıyorum	Kesinlikle Katılmıyorum
1	Bilgisayarla (tablet, telefon vb.) oyun oynamak okul başarıma katkı sağlar.					
2	Bilgisayar oyunlarından yararlanılarak ders çalışmak faydalıdır.					
3	Dersleri bilgisayarla oyun oynayarak işlemeyi isterim.					
4	Bilgisayar oyunlarındaki kahramanların dış görünüşünü kendim ayarlamak isterim.					
5	Bilgisayarda kendi oyunumu tasarlamak isterim.					
6	Bilgisayarda kodlama yapmayı öğrenmek isterim.					
7	Bilgisayar ile ilgili meslek sahibi olanlar <u>aktif değildir.</u>					
8	Oyunların seviyeli olması ilgimi çeker.					
9	Oyunlardaki seviyelerin çokluğu oyuna olan ilgimi artırır.					
10	Oyun hazırlarken kahramanın başına neler gelebileceğini kendim belirlemek isterim.					
11	Kendi hazırladığım bilgisayar oyununu arkadaşlarımla da oynamasını isterim.					
12	Bilgisayarda kendi oyunumu hazırlamak ilgimi <u>çekmez.</u>					
	Maddeler	Kesinlikle Katılmıyorum	Katılmıyorum	Kararsızım	Katılmıyorum	Kesinlikle Katılmıyorum
13	Arkadaşlarımla tasarladığım oyunları oynamak eğlenceli olabilir.					
14	Bilgisayarda kod yazmak benim için <u>zordur.</u>					
15	Bilgisayarda kod yazmayı öğrenmek <u>istemem.</u>					
16	Sokakta oynanan oyunlar benim için tehlikelidir.					
17	Kodlama öğreniminin benim için faydalı olacağını düşünüyorum.					
18	Bilgisayar oyunları ile eğitim öğretim <u>olmaz.</u>					
19	Kodlama öğrenimi problem çözme becerimi geliştireceği için sınavlarda başarımla artar.					
20	Kendi oyunumu tasarlamak yaratıcılığımı geliştirecektir.					
21	Bilgisayarla oyun oynamak boş zamanları değerlendirmek için idealdir.					
22	Dışarıda (futbol, basketbol, evcilik vb.) oynanan oyunlardan bilgisayarla oynanan oyunlar daha iyidir.					
23	Arkadaşlarımla oyun oynamaktansa bilgisayar oyunu oynamayı tercih ederim.					
24	Bilgisayar oyunlarında ikili oyunları daha çok severim.					
25	Kodlama öğrenimi zeka geliştirir.					
26	Bilgisayarda oyun oynamak bireyleri <u>tembelleştirir.</u>					
27	Derslerin bilgisayar oyunları ile işlenmesi derse olan ilgimi artırır.					
28	Bilgisayarda oyun kodlamayı öğrenirsem derslerim de başarımla artar.					

Ek 4. EBODKÖYTÖ Kullanım İzni

Subject: Re: Ölçek Kullanım İzni
From: Fikriye ZENGİN <fkzengin@gmail.com>
Date: Fri, November 18, 2016 14:39
To: umit.karaduman@ogr.deu.edu.tr
Priority: Normal
Options: [View Full Header](#) | [View Printable Version](#)

Merhabalar

"Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeği" mizi kullanabilirsiniz. İyi çalışmalar

Prof. Dr. Fikriye Kırbağ Zengin

18 Kasım 2016 12:32 tarihinde <umit.karaduman@ogr.deu.edu.tr> yazdı:

> İyi çalışmalar.
 >
 > Dokuz Eylül Üniversitesi Bilgisayar ve Öğretim Teknolojileri Öğretmenliği
 > yüksek lisans öğrencisiyim. Ekte yer alan tezim hususunda "Eğitsel
 > Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeği"nizi
 > kullanmak istiyorum. Uygun gördüğünüz takdirde düzenlenmiş halini cevaben
 > atabilirsiniz çok sevinirim.
 >
 > Saygılarımla...
 > Ümit KARADUMAN
 > Bilişim Teknolojileri Öğretmeni

Ek 5. PÖDA

ALTINKENT ORTAOKULU
BİLİŞİM TEKNOLOJİLERİ VE YAZILIM DERSİ 6. SINIF SINAVI

Adı – Soyadı: _____ Numarası: _____ Notu: _____

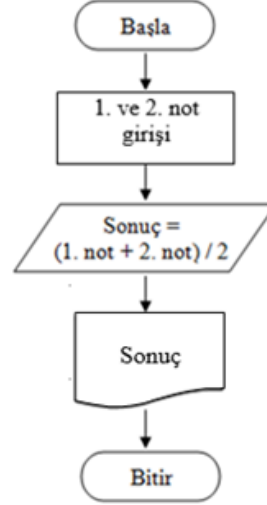
Daha önce programlama (Kodlama) eğitimi aldınız mı?	Evet _____	Hayır _____
Programlama, kodlama, akış şeması gibi kavramlar size ne anlam ifade ediyor, sizce hangi amaçlar için kullanılabilir?		

SORULAR

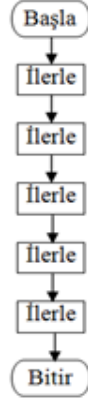
1. Resimdeki çocuğumuzun ay çiçeğine ulaşması için gerekli programlama akış şemasını çiziniz.



2. BTY dersindeki iki sınav sonucunun ortalamasını hesaplayarak notunu öğrenmek isteyen öğrenci akış şemasında hatalar yapmıştır. Hatalı kısımları düzelterek tekrar çiziniz.



3. Resimdeki kuşun domuzcuğa ulaşması için gerekli akış şeması verilmiştir. Bu şemayı daha az satırla daha sade olarak çiziniz?



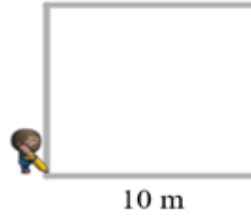
4. Resimdeki kuşun domuzcuğa ulaşması için gerekli olan programlama akış şemasını en az satırla en sade şekilde çiziniz.



5. Aşağıda çizili olan programlama akış şemasının görevi ne olabilir? Adım adım yazarak açıklayınız.



6. Aşağıdaki minik ressamımızın bir kenarı 10 metre olan kareyi çizebilmesi için gerekli programlama akış şemasını çiziniz.



7. Keloğlan eğer su sıcaksa deniz girmek, değilse güneşlenmek istiyor. Gerekli programlama akış şemasını çiziniz.



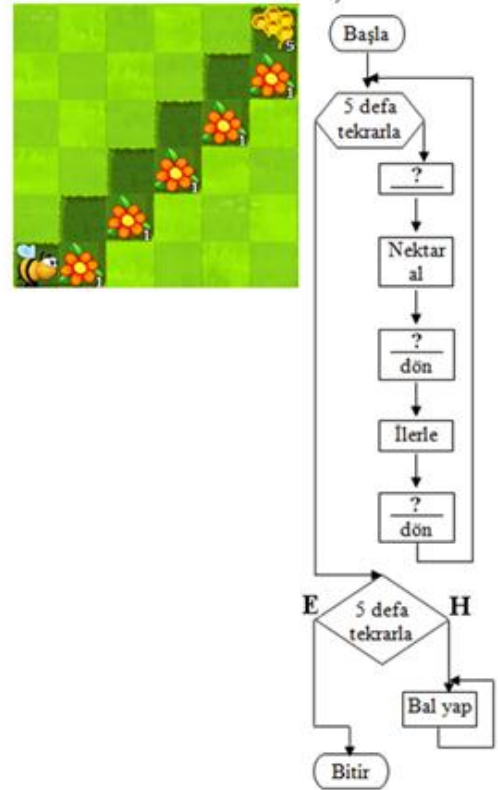
8. Çiftçimiz yol olduğu sürece ilerleyip toprak yığınlarını tarlasından çıkarmak, çukurları da doldurmak istiyor. Gerekli programlama akış şemasını çiziniz.



9. Keloğlan yeşil ışık yandığında dört adım ilerleyip okuluna girmek istiyor. Gerekli programlama akış şemasını çiziniz.



10. Arımız beş çiçekten de nektar alıp son kutuya ulaştığında ise beş kovana da bal yapmak istiyor. Akış şemasındaki hatalı ve eksik kısımları düzelterek tekrar çiziniz.



Ek 6. PÖDA Uzman Görüşü Formu

Testin amacı: Bu ölçek 6. Sınıf Bilişim Teknolojileri ve Yazılım dersi programlama akademik başarı tespiti için kullanılacaktır.

Hedef kitle: Ortaokul 6. Sınıf

Ders Konuları ve Soruların Dağılımı

Soru No	Konu	İlgili Kazanım	Sorunun Uygunluğu (Soruların uygun olup olmadığı sorular üzerine de işaretleyebilirsiniz.)
1	Temel Algoritma	- Belirlenen problemin çözümü için adımlar oluşturur. - Belirlenen problemin çözüm sürecinde gerekli işlemler için akış şeması hazırlar.	Uygun () Uygun değil () Öneri:.....
2	Hata Ayıklama - Akış Şeması Sembolleri	- Hatalı bir algoritmayı doğru çalışacak biçimde düzenler. - Geliştirdiği stratejilerin, belirlenen problemin çözümündeki etkisini tartışır.	Uygun () Uygun değil () Öneri:.....
3	Temel Algoritma	- Belirlenen problemin çözümüne farklı yollardan ulaşır. - Çözümü verilen probleme farklı bir çözüm yolu önerir. - Problem çözümü için geliştirdiği adımların geçerliğini sorgulayarak en etkili çözüme ulaşır. - Farklı algoritmaları inceleyerek en hızlı ve doğru çözümü seçer.	Uygun () Uygun değil () Öneri:.....
4	Döngü	- Belirlenen problemin çözümü için adımlar oluşturur. - Belirlenen problemin çözüm sürecinde gerekli işlemler için akış şeması hazırlar. - Belirlenen problemin çözümünü sağlayan farklı algoritmalar hakkında tartışır. - Farklı algoritmaları inceleyerek en hızlı ve doğru çözümü seçer.	Uygun () Uygun değil () Öneri:.....
5	Akış Şeması Yorumlama	- Geliştirilen stratejilerin, belirlenen problemin çözümündeki etkisini tartışır.	Uygun () Uygun değil () Öneri:.....
6	Döngü	- Belirlenen problemin çözümü için adımlar oluşturur. - Belirlenen problemin çözüm sürecinde gerekli işlemler için akış şeması hazırlar.	Uygun () Uygun değil () Öneri:.....
7	Koşul (Eğer - Değilse)	- Belirlenen problemin çözümü için adımlar oluşturur. - Belirlenen problemin çözüm sürecinde gerekli işlemler için akış şeması hazırlar.	Uygun () Uygun değil () Öneri:.....
8	Koşul (Eğer - Değilse) - Döngü	- Belirlenen problemin çözümü için adımlar oluşturur. - Belirlenen problemin çözüm sürecinde gerekli işlemler için akış şeması hazırlar.	Uygun () Uygun değil () Öneri:.....
9	Koşul (Eğer - Değilse) - Döngü	- Belirlenen problemin çözümü için adımlar oluşturur. - Belirlenen problemin çözüm sürecinde gerekli işlemler için akış şeması hazırlar. - Belirlenen problemin çözümünü sağlayan farklı algoritmalar hakkında tartışır.	Uygun () Uygun değil () Öneri:.....
10	Hata Ayıklama - Akış Şeması Sembolleri - Eksik Tamamlama	- Hatalı bir algoritmayı doğru çalışacak biçimde düzenler. - Geliştirdiği stratejilerin, belirlenen problemin çözümündeki etkisini tartışır.	Uygun () Uygun değil () Öneri:.....

Ek 7. CPGF

Bugün _____ saat _____ Code.org platformuna yönelik öğrenci görüşlerini almak üzere _____ ile görüşme yapmak amacıyla bilişim teknolojileri laboratuvarındayım. Merhaba _____ BTY dersinde kullandığın Code.org platformu hakkında bazı sorular soracağım, bu yaptıklarınızla ilgili düşüncelerini anlamak istiyorum. Ama bu soruları cevaplamak zorunda değilsin, istemezsen burada kesebiliriz. Eğer “cevaplarım” diyorsan konuştuklarımızı kaydetmem gerekiyor. Kaydettiğimiz her şeyi araştırmam için kullanacağım, ama başka hiç kimse ile kesinlikle paylaşmayacağım. Ne dersin, bunları kabul ediyor musun, devam edelim mi?

Teşekkürler, o halde sorulara geçebiliriz.

Görüşme Soruları:

1. Code.org platformunun kodlamayı öğrenmen üzerinde (Akış Şeması, Döngüler, Koşul Yapıları vb.) etkili olduğunu düşünüyor musun? Etkili ise hangi yönlerden etkili oldu? Ne kazandırdı?
2. Code.org platformunu kullanmanın kodlamayı daha kalıcı olarak öğrenmene etki edeceğini düşünüyor musun? Açıklayabilir misin?
3. Code.org platformunun en beğendiğin tarafları nelerdi? Açıklar mısın?
4. Code.org platformunu kullanırken karşılaştığın problemler nelerdir? Bu problemlerle nasıl başa çıktın? Bu problemler öğrenme isteğinde olumsuz yönde etkili oldular mı?
5. Aynı uygulamayı bir kez daha yapılsa nelerin eklenmesini veya değiştirilmesini isterdin?
6. Bu platform ile kodlama öğrenmeyi bir arkadaşına önerir misin? Neden?
7. Kodlama ile kendi oyununu kendin yazmak ister miydin? Sence bunu başarabilir misin?
8. Kodlama alanında merak ettiğin konular neler? Açıklar mısın?
9. Kodlama ile ilgili bir iş alanında çalışmak ister miydin? Açıklar mısın?
10. Code.org platformunu kullanarak kodlama öğrenme sürecin, BTY dersi ile ilgili düşüncelerin değiştirdi mi? Eğer değiştirdiyse olumlu mu olumsuz yönde mi? Örnek verebilir misin?

Ek 8. Code.org Platformu Kullanım İzni

From: "Josh S. (Code.org)" <support@code.org>
Date: Mon, February 29, 2016 19:59
To: "Umit Karaduman" <umit.karaduman@ogr.deu.edu.tr>
Priority: Normal
Options: [View Full Header](#) | [View Printable Version](#)

Hi,

All of our content is entirely free to use worldwide, so there is no need for formal permission or partnership to use Code.org in a school setting. Thanks for reaching out!

Best,
Josh

[image: Umit Karad]

Umit Karaduman

Feb 29, 05:45

Hello

First of all thank you for the service you offer this platform. I am an ICT teacher and graduate student. Using this platform, together with my students want to do graduate work. I'd like your permission. Good work.

Umit Karaduman
Dokuz Eylul University

Ek 9. BTY Dersi 5. Sınıf “Problem Çözme ve Programlama Alanı” Kazanımları

5.5.1.	5.5.1. Problem Çözme Kavramları ve Yaklaşımları
5.5.1.1.	Günlük hayatta karşılaştığı problemlere çözüm önerileri getirir.
5.5.1.2.	Verilen bir problemi uygun adımları kullanarak çözer. <i>Bir problemi çözmek için farklı çözümlerinin tasarlanabileceği vurgulanır.</i>
5.5.1.3.	Problem çözmede temel kavramları tanımlayarak problem türlerini açıklar.
5.5.1.4.	Problem çözme sürecinde takip edilmesi gereken adımları fark eder.
5.5.1.5.	Verilen bir problemi analiz eder.
5.5.1.6.	Problemi çözmek için gerekli değişken, sabit ve işlemleri açıklar.
5.5.1.7.	Problem çözümünde kullanılabilecek operatörlere örnek verir.
5.5.1.8.	Problem çözümünde ifa de ve eşitliklere örnek verir.
5.5.1.9.	Problem çözümünde işlem önceliğine örnek verir.
5.5.1.10.	Verilen bir problemin çözümünde operatörleri kullanır.
5.5.1.11.	Verilen bir problemde ifa de ve eşitlikleri kullanarak çözüm üretir.
5.5.1.12.	Algoritma kavramını açıklar.
5.5.1.13.	Bir problemin çözümü için algoritma geliştirir.
5.5.1.14.	Akış şeması bileşenlerini ve işlevlerini açıklar. Bir algoritma için akış şeması çizer.
5.5.1.15.	<i>Akış şemasının elektronik ortandaki çizimi için kelime işlemci programları veya diğer çizim programları kullanılır.</i>

5.5.1.16.	Bir algoritmayı test ederek hataları ayıklar.
5.5.1.17.	Matematik ve bilgisayar bilimi arasındaki ilişkiyi tespit eder.
5.5.2.	Programlama
5.5.2.1.	Programlamayla ilgili temel kavramları açıklar. <i>Program, program yazmanın amacı, programlama dili üzerinde durulur.</i>
5.5.2.2.	Blok tabanlı programlama aracının arayüzünü ve özelliklerini tanıır. <i>Açık kaynak kodlu veya ücretsiz erişilebilen programlama araçları kullanılır.</i>
5.5.2.3.	Blok tabanlı programlama ortamında sunulan hedeflere ulaşmak için doğru algoritmayı oluşturur. <i>Blok tabanlı programlama aracındaki basit örnekler üzerinden algoritma işlemleri yaptırılır.</i>
5.5.2.4.	Doğrusal mantık yapısını açıklar.
5.5.2.5.	Doğrusal mantık yapısını kullanan algoritmalar geliştirir.
5.5.2.6.	Karar yapısını ve işlevlerini açıklar.
5.5.2.7.	Karar yapıları içeren algoritmalar geliştirir.
5.5.2.8.	Döngü yapısını ve işlevlerini açıklar. <i>Tekrarlanan işlemler için döngü yapılarının gerekliliği üzerinde durulur.</i>
5.5.2.9.	Döngü yapısı içeren algoritmalar oluşturur.
5.5.2.10.	Farklı yapılar için oluşturduğu algoritmaların sonucunu yordayarak hatalarını ayıklar.

Ek 10. BTY Dersi 6. Sınıf “Problem Çözme ve Programlama” Alanı Kazanımları

6.5.1.	Problem Çözme Kavramları ve Yaklaşımları
6.5.1.1.	Verileri toplayarak türlerine göre sınıflandırır.
6.5.1.2.	Sabitleri ve değişkenleri problem çözümünde kullanır.
6.5.1.3.	Bir problemi alt problemlere böler.
6.5.1.4.	Temel fonksiyonları problem çözme sürecinde kullanır.
6.5.1.5.	Problemin çözümü için bir algoritma geliştirir.
6.5.1.6.	Bir algoritmanın çözümünü test eder.
6.5.1.7.	Farklı algoritmaları inceleyerek en hızlı ve doğru çözümü seçer.
6.5.1.8.	Hatalı bir algoritmayı doğru çalışacak biçimde düzenler.
6.5.1.9.	Problemin çözümünü benzer problemler için geneller.
6.5.1.10.	Matematik ve bilgisayar bilimi arasındaki ilişkiyi tartışır.
6.5.2.	Programlama
6.5.2.1.	Blok tabanlı programlama aracının arayüzünü ve özelliklerini tanır. <i>Açık kaynak kodlu veya ücretsiz erişilebilen programlama platformları kullanılabilir.</i>
6.5.2.2.	Blok tabanlı programlama aracında sunulan bir programın işlevlerini açıklar.
6.5.2.3.	Blok tabanlı programlama aracında sunulan bir programın hatalarını ayıklar.
6.5.2.4.	Blok tabanlı programlama aracında sunulan bir programı verilen ölçütlere göre geliştirerek düzenler.

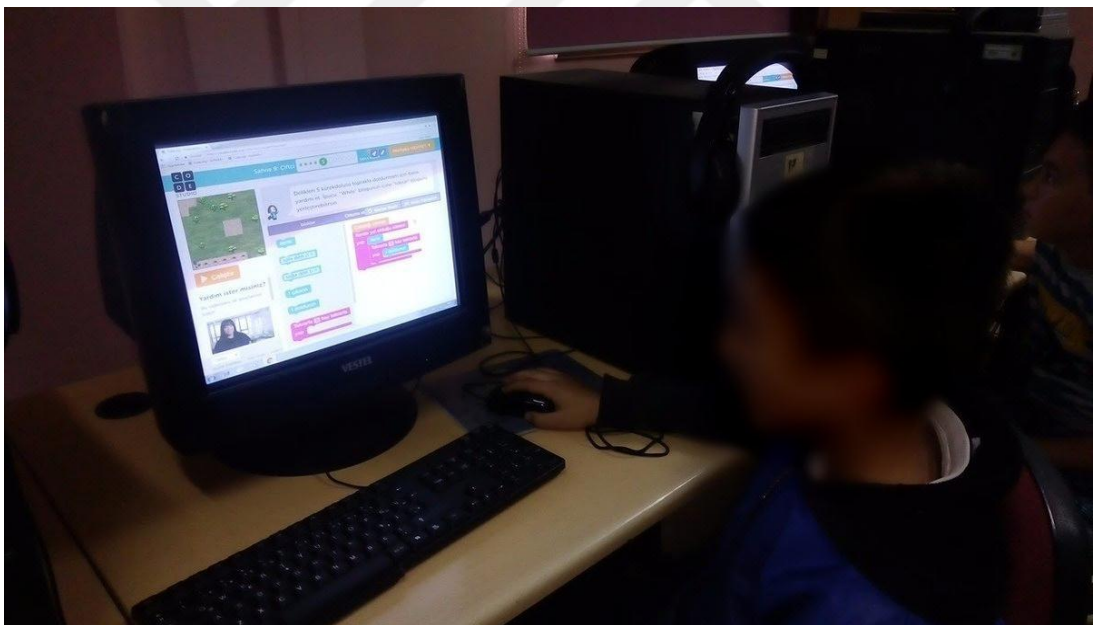
6.5.2.5.	Doğrusal mantık yapısını içeren programlar oluşturur.
6.5.2.6.	Doğrusal mantık yapısını içeren programları test ederek hatalarını ayıklar.
6.5.2.7.	Karar yapısını içeren programlar oluşturur.
6.5.2.8.	Karar yapısını içeren programları test ederek hatalarını ayıklar.
6.5.2.9.	Çoklu karar yapıları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

Ek 11. Örnek Günlük Ders Planları

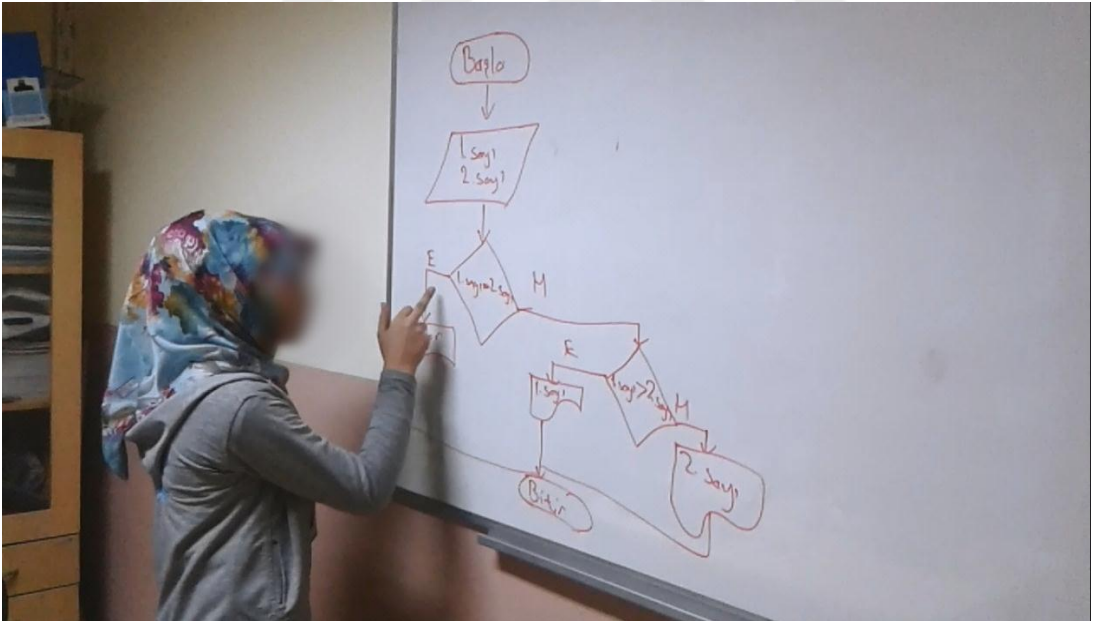
(Kontrol Grubu) Altınkent Ortaokulu 2016-2017 Eğitim-Öğretim Yılı 2. Dönem Bilişim Teknolojileri ve Yazılım Dersi 6. Sınıflar Günlük Planı	
Dersin Adı	Bilişim Teknolojileri ve Yazılım
Sınıf	6. Sınıflar
Ünite Adı	4. ÜNİTE: Problem Çözme, Programlama ve Özgün ürün geliştirme
Konu	4.1. Problem Çözme Yaklaşımları
Süre	40 Dakika + 40 Dakika
Kazanımlar	• Problem kavramını tanımlar. • BİT kullanımında karşılaşılan problemlerin farkına varır. • Problem çözmenin önemini ifade eder.
Ünite Kavramları ve Sembolleri / Davranış Örüntüsü	Problem, Analiz, Çözme Yaklaşımları
Öğretme - Öğrenme Yöntem ve Teknikleri	Anlatım, soru-cevap, tartışma, gösterip yaptırma
Kullanılan Eğitim Tek, Araç - Gereçler ve Kaynakça	Bilgisayar, projeksiyon, internet, görseller
Öğretme - Öğrenme Etkinlikleri	
• Dikkat çekme • Güdüleme • Gözden geçirme • Derse geçiş • Bireysel öğrenme etkinlikleri • Grupla öğrenme etkinlikleri • Özet	Ayakkabı bağlamayı bilmediği varsayılan bir öğrenci tahtaya çıkarılır. Diğer öğrencilerden sadece sözlü yönlendirme yaparak bu öğrenciye ayakkabı bağlama işlemini yaptırmaları istenir. Anlatma işleminde adımlar atlandığında veya sırası karıştırıldığında ayakkabının bağlanmasının imkânsız olduğunu öğrencilerin anlaması sağlanır. Buradan yola çıkarak bilgisayarlarında hiç bir şey bilmediğini, kendi başına hareket etmediğini, ona bir şey yaptırmak istediğimizde tüm isteklerimizi basit ve belli sırayla anlatmamız gerektiği vurgulanır. Son olarak BİT kullanımında karşılaşılan problemler sorulur. Örnek olarak yazıcıdan çıktı almak için gerekli olan adımlar sorulur ve sınıfla birlikte tüm adımlar tahtaya yazılır.
Ölçme - Değerlendirme	
• Bireysel öğrenme etkinliklerine göre ölçme - değerlendirme • Grupla öğrenme etkinliklerine göre ölçme ve değerlendirme	Öz değerlendirme ölçütleri • Problem kavramını tanımlarım. • BİT kullanımında karşılaşılan problemleri bilirim. • Problem çözmenin önemini ifade ederim.
Planın Uygulanmasına İlişkin Açıklamalar	Bu plan 6. sınıflara 13.02.2017 - 17.02.2017 tarihleri arasında uygulanacaktır.

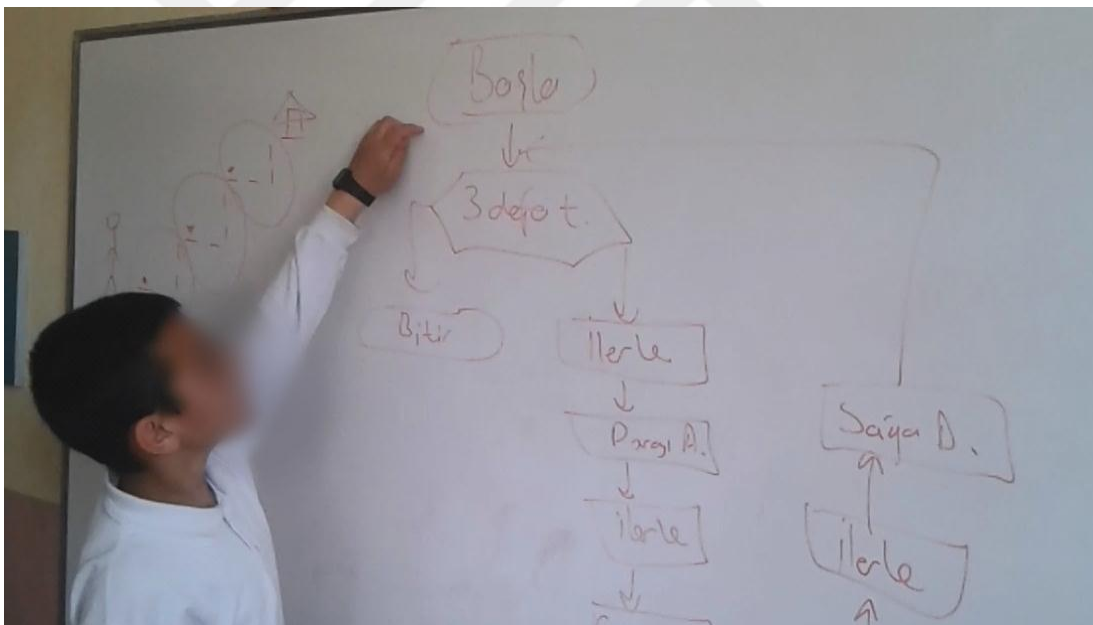
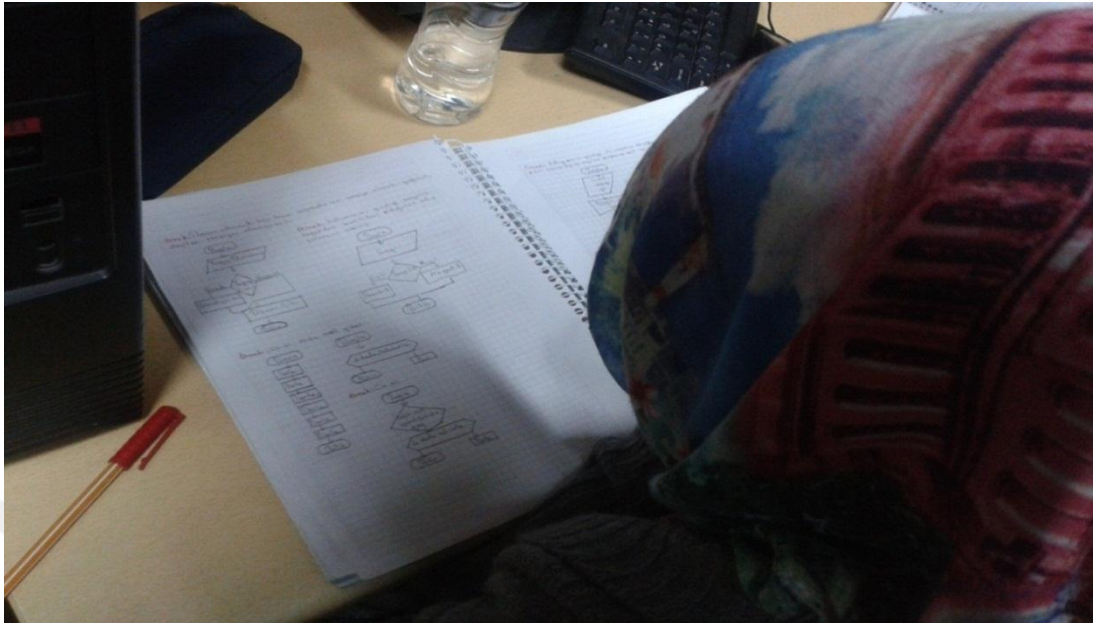
(Deney Grubu) Altınkent Ortaokulu 2016-2017 Eğitim-Öğretim Yılı 2. Dönem Bilişim Teknolojileri ve Yazılım Dersi 6. Sınıflar Günlük Planı	
Dersin Adı	Bilişim Teknolojileri ve Yazılım
Sınıf	6. Sınıflar
Ünite Adı	4. Ünite: Problem Çözme, Programlama ve Özgün ürün geliştirme
Konu	4.2. Algoritma ve Strateji Geliştirme
Süre	40 Dakika + 40 Dakika
Kazanımlar	• Belirlenen problemin çözümü için algoritma geliştirmenin önemini ifade eder.
Ünite Kavramları ve Sembolleri / Davranış Örüntüsü	Algoritma, Strateji geliştirme, www.code.org
Öğretme - Öğrenme Yöntem ve Teknikleri	Anlatım, soru-cevap, tartışma, gösterip yapma
Kullanılan Eğitim Tek, Araç - Gereçler ve Kaynakça	Bilgisayar, projeksiyon, internet, Code.org platformu
Öğretme - Öğrenme Etkinlikleri	
• Dikkat çekme • Güdüleme • Gözden geçirme • Derse geçiş • Bireysel öğrenme etkinlikleri • Grupla öğrenme etkinlikleri • Özet	www.code.org web sitesi açılır. Öğrenci ilerlemeleri görüntülenir ve öğrencilerin hangi bölümde olduğu, hangi bölümleri doğru yapıp hangilerini yanlış yaptıklarının öğretmen kontrol panelinden görüntülenebilir olduğu belirtilir. Öğrencilerin yaparken en çok zorlandıkları bölümler tespit edilerek bu bölümler açılır. Sınıf önünde bu bölümler incelenir. Problemin çözümünde bir adım eksik yazılması bir adım fazla yazılması halinde problemin çözülmediği gözlemlenmiş olur. Uygulamada verilen problemin çözülmesi için tüm adımların doğru ve eksiksiz bir şekilde yazılması gerektiği, yani problemi çözerken algoritma geliştirmemiz gerektiği belirtilir.
Ölçme - Değerlendirme	
• Bireysel öğrenme etkinliklerine göre ölçme - değerlendirme • Grupla öğrenme etkinliklerine göre ölçme ve değerlendirme	Öz değerlendirme ölçütleri • Belirlenen problemin çözümü için algoritma geliştirmenin önemini ifade ederim.
Planın Uygulanmasına İlişkin Açıklamalar	Bu plan 6. sınıflara 20.02.2017 - 24.02.2017 tarihleri arasında uygulanacaktır.

Ek 12. Deney Grubuna Ait Görseller



Ek 13. Kontrol Grubuna Ait Görseller





Ek 14. Ölçeklerin Uygulanmasına Yönelik Görseller

