

T.C.
NECMETTİN ERBAKAN ÜNİVERSİTESİ
EĞİTİM BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ EĞİTİMİ
ANABİLİM DALI
BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ EĞİTİMİ
BİLİM DALI

TEKNOLOJİK PEDAGOJİK ALAN BİLGİSİ (TPAB)
ÇERÇEVESİ İLE OLUŞTURULMUŞ PROGRAMLAMA
EĞİTİMİNİN ÖĞRENME ÇIKTILARI ÜZERİNE
ETKİLERİ

Handan ATUN

YÜKSEK LİSANS TEZİ

Danışman
Doç. Dr. Ertuğrul USTA

Konya, 2018

T.C.
NECMETTİN ERBAKAN ÜNİVERSİTESİ
EĞİTİM BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ EĞİTİMİ
ANABİLİM DALI
BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ EĞİTİMİ
BİLİM DALI

TEKNOLOJİK PEDAGOJİK ALAN BİLGİSİ
(TPAB) ÇERÇEVESİ İLE OLUŞTURULMUŞ
PROGRAMLAMA EĞİTİMİNİN ÖĞRENME
ÇIKTILARI ÜZERİNE ETKİLERİ



Handan ATUN

YÜKSEK LİSANS TEZİ

Danışman

Doç. Dr. Ertuğrul USTA

Konya, 2018

 KONYA	T.C. NECMETTİN ERBAKAN ÜNİVERSİTESİ Eğitim Bilimleri Enstitüsü Müdürlüğü	 NECMETTİN ERBAKAN ÜNİVERSİTESİ EĞİTİM BİLİMLERİ ENSTİTÜSÜ
--	--	---

Bilimsel Etik Sayfası

Öğrencinin	Adı Soyadı	Handan ATUN		
	Numarası	158305011006		
	Ana Bilim / Bilim Dalı	Bilgisayar ve Öğretim Teknolojileri Eğitimi Ana Bilim Dalı/ Bilgisayar ve Öğretim Teknolojileri Eğitimi Bilim Dalı		
	Programı	Tezli Yüksek Lisans	<input checked="" type="checkbox"/>	
		Doktora	<input type="checkbox"/>	
Tezin Adı	Teknolojik Pedagojik Alan Bilgisi (TPAB) Çerçevesi İle Oluşturulmuş Programlama Eğitiminin Öğrenme Çıktıları Üzerine Etkileri			

Bu tezin hazırlanmasında bilimsel etiğe ve akademik kurallara özenle riayet edildiğini, tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada başkalarının eserlerinden yararlanılması durumunda bilimsel kurallara uygun olarak atıf yapıldığını bildiririm.


Handan ATUN

 KONYA	T.C. NECMETTİN ERBAKAN ÜNİVERSİTESİ Eğitim Bilimleri Enstitüsü Müdürlüğü	 NECMETTİN ERBAKAN ÜNİVERSİTESİ EĞİTİM BİLİMLERİ ENSTİTÜSÜ
---	---	---

YÜKSEK LİSANS TEZİ KABUL FORMU

Öğrencinin	Adı Soyadı	Handan ATUN
	Numarası	158305011006
	Ana Bilim Dalı	Bilgisayar ve Öğretim Teknolojileri Eğitimi
	Bilim Dalı	Bilgisayar ve Öğretim Teknolojileri Eğitimi
	Programı	Tezli Yüksek Lisans
	Tez Danışmanı	Doç. Dr. Ertuğrul USTA
	Tezin Adı	Teknolojik Pedagojik Alan Bilgisi (TPAB) Çerçevesi İle Oluşturulmuş Programlama Eğitiminin Öğrenme Çıktıları Üzerine Etkileri

Yukarıda adı geçen öğrenci tarafından hazırlanan “Teknolojik Pedagojik Alan Bilgisi (TPAB) Çerçevesi İle Oluşturulmuş Programlama Eğitiminin Öğrenme Çıktıları Üzerine Etkileri” başlıklı bu çalışma 06/08/2018 tarihinde yapılan savunma sınavı sonucunda oybirliği/oyçokluğu ile başarılı bulunarak, jürimiz tarafından yüksek lisans tezi olarak kabul edilmiştir.

	Ünvanı Adı Soyadı	İmza
Danışman	Doç. Dr. Ertuğrul USTA	
Jüri Üyesi	Doç. Dr. Sami ŞAHİN	
Jüri Üyesi	Dr.Öğr. Üyesi Şemseddin GÜNDÜZ	

TEŞEKKÜR

Araştırmamın başından itibaren bana desteğini ve yardımını esirgemeyen, sorunlarıma hızlı çözümler sunan ve sonuna kadar motivasyonumu kırmadan ilerlememi sağlayan tez danışmanım Doç. Dr. Ertuğrul USTA hocama sonsuz teşekkürlerimi sunarım. Ayrıca her zaman olduğu gibi bu süreçte de yanımdan bir an olsun ayrılmayan, maddi manevi destekleriyle beni her zaman başarıların üstünde taşıyan anneme ve babama teşekkürü bir borç bilirim. Siz olmasanız başaramazdım. Ayrıca bana bu süreçte yol göstermeyi ihmal etmeyen canım ablama ve bilgeliğiyle benim sevgi hocam olmayı başarmış Sensei' ye teşekkür ederim.

T.C.
NECMETTİN ERBAKAN ÜNİVERSİTESİ
Eğitim Bilimleri Enstitüsü Müdürlüğü

Öğrencinin	Adı Soyadı	Handan ATUN
	Numarası	158305011006
	Ana Bilim / Bilim Dalı	Bilgisayar ve Öğretim Teknolojileri Ana Bilim Dalı
	Programı	Tezli Yüksek Lisans <input checked="" type="checkbox"/> Doktora <input type="checkbox"/>
	Tez Danışmanı	Doç. Dr. Ertuğrul USTA
	Tezin Adı	Teknolojik Pedagojik Alan Bilgisi (TPAB) Çerçevesi İle Oluşturulmuş Programlama Eğitiminin Öğrenme Çıktıları Üzerine Etkileri

ÖZET

Bu çalışmanın amacı Bilişim Teknolojileri ve Yazılım (BTY) dersi kapsamında TPAB çerçevesi ile oluşturulmuş programlama eğitiminin ortaokul öğrencilerinin öğrenme çıktıları üzerine etkilerini araştırmaktır. Araştırmanın örneklemini tamamı ortaokul 6. sınıfta öğrenim gören 41 öğrenci oluşturmaktadır. Ön-test son-test deney ve kontrol gruplu yarı deneysel desenle yürütülen bu çalışmada hem nicel hem de nitel verilerin toplandığı karma araştırma yöntemi kullanılmıştır. TPAB çerçevesiyle oluşturulmuş programlama öğretiminin öğrencilerin akademik başarısına, problem çözme becerisi algılarına, bilgisayarca düşünme beceri düzeylerine etkileri ile bu değişkenlerin arasında anlamlı ilişki olup olmadığı ve ölçülen bu değerlerin cinsiyete göre farklılaşıp farklılaşmadığı incelenmiştir. Aynı zamanda görüşme sorularıyla öğrencilerden TPAB çerçevesiyle oluşturulmuş programlama dersi üzerine görüşleri alınmıştır. Araştırma sonuçlarına göre deney grubunun akademik başarı puanı ortalamaları ile problem çözme envanteri ve bilgisayarca düşünme beceri düzeyleri ölçeğinden aldıkları ortalama puan kontrol grubunun puanlarından anlamlı derecede yüksek çıkmıştır. Buna göre TPAB çerçevesiyle oluşturulmuş öğretimin öğrencilerin akademik başarıları, problem çözme becerisi algıları ve bilgisayarca düşünme beceri düzeyleri üzerinde olumlu etkisi olmuştur. Korelasyon analizi sonucunda ise öğrencilerin akademik başarıları ile problem çözme ve bilgisayarca düşünme becerileri arasında pozitif yönlü anlamlı ilişkiye rastlanmıştır. Ayrıca öğrencilerin akademik başarıları, problem çözme ve bilgisayarca düşünme becerileri cinsiyete göre farklılık

göstermemektedir. Aynı zamanda öğrencilerin görüşlerinden alınan bilgilere göre bu sonuçları desteklemektedir. Öğrencilerin TPAB çerçevesiyle oluşturulmuş programlama dersi üzerine görüşleri genel olarak olumludur.



T.C.
NECMETTİN ERBAKAN ÜNİVERSİTESİ
Eğitim Bilimleri Enstitüsü Müdürlüğü

Öğrencinin	Adı Soyadı	Handan ATUN
	Numarası	158305011006
	Ana Bilim / Bilim Dalı	Bilgisayar ve Öğretim Teknolojileri Ana Bilim Dalı
	Programı	Tezli Yüksek Lisans <input checked="" type="checkbox"/> Doktora <input type="checkbox"/>
	Tez Danışmanı	Doç. Dr. Ertuğrul USTA
	Tezin İngilizce Adı	The Effects of Programming Education Planned with Technological Pedagogical Content Knowledge (TPACK) Framework on Learning Outcomes

SUMMARY

This study aims at investigating the effects of Programming Education Planned with TPACK Framework on middle school students' learning outputs within the ITS (Informational Technology and Software) course. The sample of the research consists of 41 6th grade level students who studies in middle school. This study used a quasi-experimental research design that compared pre-test and post-test results for experimental and control groups. Data were collected through both qualitative and quantitative scales, which is called mixed research method. The effects of programming education planned with TPACK framework on students' academic achievement, perception of problem solving skills and computational thinking skills, the relationship between those variables and the impact of the gender on those variables are investigated. Furthermore, qualitative data related to programming lesson planned with TPACK framework is collected through interview questions from students. According to the results, the means of academic achievement score, problem solving inventory score and computational thinking skill scale score of experimental group are significantly higher than those of control group. In accordance with this result it can be said that lessons planned with TPACK framework has a positive impact on students' academic achievement, problem solving and computational skills. Using correlation analysis, a significant positive directed correlation is found between students' academic achievement, problem solving skills and computational thinking skills scores. There is no difference between those variables related to gender.

Moreover, these results are supported by students' opinions, which are mostly positive on programming lesson planned with TPACK framework.



İÇİNDEKİLER

	<u>Sayfa No</u>
Bilimsel Etik Sayfası.....	II
Yüksek Lisans Tezi Kabul Formu.....	III
Teşekkür.....	IV
Özet.....	V
Summary.....	VII
İçindekiler.....	IX
Kısaltmalar.....	XI
Tablolar Listesi.....	XIII
1. Giriş.....	1
1.1. Araştırmanın Konusu.....	3
1.2. Araştırmanın Amacı.....	4
1.3. Araştırmanın Önemi.....	5
1.4. Varsayımlar.....	6
1.5. Sınırlılıklar.....	6
1.6. Tanımlar.....	6
2. Teorik Çerçeve.....	8
2.1. Teknolojinin Eğitimde Kullanılması.....	8
2.1.1. Eğitim/Öğretim Teknolojileri.....	8
2.1.2. Bilgi ve İletişim Teknolojileri (BİT) ve Eğitim.....	9
2.1.3. Teknolojinin Eğitime Entegrasyonu ve TPAB.....	14
2.2. Bilgisayar Eğitimi.....	16
2.2.1. Türkiye’de Eğitim Teknolojileri ve Bilgisayar Öğretimi.....	16
2.2.2. Bilgisayar Eğitiminin Dünyadaki Yeri.....	20
2.2.3. Bilgisayar Eğitimi ve TPAB.....	23
2.3. TPAB Çerçevesinin K12 Eğitiminde Kullanılması.....	24
3. Yöntem.....	26
3.1. Araştırmanın Modeli.....	26
3.2. Çalışma Grubu.....	28

3.3. Veri Toplama Araçları.....	28
3.3.1. Programlama Başarı Testinin Geliştirilmesi.....	28
3.3.2. Bilgisayarca Düşünme Beceri Düzeyleri (BDBD) Ölçeği	31
3.3.3. İlköğretim Düzeyindeki Çocuklar İçin Problem Çözme Envanteri (ÇPÇE).....	32
3.3.4. Görüşme Soruları.....	32
3.3.5. Ders Planının Hazırlanması.....	32
3.4. Veri Analizi.....	38
4. Bulgular ve Yorum.....	41
4.1. Katılımcıların Betimsel Özellikleri.....	41
4.2. Araştırma Sorularının Cevapları.....	41
4.2.1. Öğrencilerin akademik başarılarına göre ön-test ve son- test puanları arasındaki farka yönelik bulgular.....	41
4.2.2. Öğrencilerin problem çözme becerilerine göre ön-test ve son-test puanları arasındaki farka yönelik bulgular.....	42
4.2.3. Öğrencilerin bilgisayarca düşünme becerilerine göre ön- test ve son-test puanları arasındaki farka yönelik bulgular.....	43
4.2.4. Öğrencilerin akademik başarısı, problem çözme becerisi algısı ve bilgisayarca düşünme becerisi arasındaki ilişkiye yönelik bulgular.....	44
4.2.5. Öğrencilerin akademik başarısı, problem çözme becerisi algısı ve bilgisayarca düşünme becerisi üzerinde cinsiyetin etkisine yönelik bulgular.....	46
4.2.6. Öğrenci görüşlerine yönelik bulgular.....	48
5. Sonuç ve Tartışma.....	56
5.1. Öneriler.....	58
6. Kaynakça.....	59
7. Ekler.....	75
8. Özgeçmiş.....	119

KISALTMALAR

AB	: Alan Bilgisi
BDBD	: Bilgisayarca Düşünme Beceri Düzeyleri
BDE	: Bilgisayar Destekli Eğitim
BİT	: Bilgi ve İletişim Teknolojileri
BÖTE	: Bilgisayar ve Öğretim Teknolojileri Eğitimi
BT	: Bilişim Teknolojileri
BTR	: Bilişim Teknolojileri Rehberliği
BTY	: Bilişim Teknolojileri ve Yazılım
ÇPÇE	: Çocuklar için Problem Çözme Envanteri
EBA	: Eğitim Bilişim Ağı
ERG	: Eğitim Reformu Girişimi
ET	: Etkileşimli Tahta
FATİH	: Fırsatları Artırma ve Teknolojiyi İyileştirme Hareketi
ICT	: Information and Communication Technologies
ISTE	: International Society for Technology in Education
IT	: Informational Technologies
K12	: Okul Öncesi, İlk ve Ortaöğretim
MEB	: Millî Eğitim Bakanlığı
ODTÜ	: Orta Doğu Teknik Üniversitesi
PAB	: Pedagojik Alan Bilgisi
PB	: Pedagojik Bilgi
RTI	: Research Triangle Institute
TAB	: Teknolojik Alan Bilgisi
TB	: Teknolojik Bilgi
TPAB	: Teknolojik Pedagojik Alan Bilgisi
TPACK	: Technological Pedagogical Content Knowledge
TPB	: Teknolojik Pedagojik Bilgi
TTKB	: Talim ve Terbiye Kurulu Başkanlığı
TÜBİTAK	: Türkiye Bilimsel ve Teknolojik Araştırma Kurumu
YEĞİTEK	: Yenilik ve Eğitim Teknolojileri

YÖK: : Yükseköğretim Kurulu



TABLOLAR LİSTESİ

	<u>Sayfa No</u>
Tablo-1: Araştırma Modeli Grupları	27
Tablo-2: Programlama Başarı Testi Belirtke Tablosu	29
Tablo-3: Programlama Başarı Testinin Madde Analiz Dereceleri	30
Tablo-4: Öğrenci Becerileri ve Bilgi Türleri	34
Tablo-5: Ders Planında Kullanılan TPAB Öğrenme Aktiviteleri	36
Tablo-6: Katılımcıların Betimsel Özellikleri	41
Tablo-7: Akademik Başarıya Göre Deney ve Kontrol Gruplarının Ön-Test Analizi	41
Tablo-8: Akademik Başarıya Göre Deney ve Kontrol Gruplarının Son-Test Analizi	42
Tablo-9: Problem Çözme Becerisine Göre Deney ve Kontrol Gruplarının Ön-Test Analizi	42
Tablo-10: Problem Çözme Becerisine Göre Deney ve Kontrol Gruplarının Son-Test Analizi	43
Tablo-11: Bilgisayarca Düşünme Becerisine Göre Deney ve Kontrol Gruplarının Ön-Test Analizi	43
Tablo-12: Bilgisayarca Düşünme Becerisine Göre Deney ve Kontrol Gruplarının Son-Test Analizi	44
Tablo-13: Akademik Başarı ve Problem Çözme Becerisi Algısı Arasındaki İlişki	45
Tablo-14: Problem Çözme Becerisi Algısı ve Bilgisayarca Düşünme Becerisi Arasındaki İlişki	45
Tablo-15: Akademik Başarı ve Bilgisayarca Düşünme Becerisi Arasındaki İlişki	46
Tablo-16: Cinsiyete Göre Akademik Başarı	46
Tablo-17: Cinsiyete Göre Problem Çözme Becerisi Algısı	47
Tablo-18: Cinsiyete Göre Bilgisayarca Düşünme Becerisi	47

Tablo-19:	Görüşme Sorularına Verilen Cevapların Kategori, Tema ve Kodları	48
Tablo-20:	Öğrencilerin 1. Görüşme Sorusuna Verdikleri Cevaplar	49
Tablo-21:	Öğrencilerin 2. Görüşme Sorusuna Verdikleri Cevaplar	51
Tablo-22:	Öğrencilerin 3. Görüşme Sorusuna Verdikleri Cevaplar	52
Tablo-23:	Öğrencilerin 4. Görüşme Sorusuna Verdikleri Cevaplar	53
Tablo-24:	Öğrencilerin 5. Görüşme Sorusuna Verdikleri Cevaplar	54



1. GİRİŞ

Artan teknoloji gelişiminin, Bilgi ve İletişim Teknolojileri çağında hayatın her alanında kullanılır hale gelmesi kaçınılmazdır. Eğitim ve öğretim de Bilgi ve İletişim Teknolojilerinin etkisinin güçlü olduğu alanlardandır, hatta teknolojinin öğrenmeyi artırdığı birçok araştırmada görülmüştür (Polly, Mims, Shepherd ve İnan, 2010). Bu durum, öğretmenlerin teknolojik ürünleri bilgece kullanacakları yeni öğretim yöntemleri üretmelerini gerekli kılmaktadır (Oster-Levinz ve Klieger, 2010). Son yıllarda öğretmenlerin teknolojiyi derslerinde etkili kullanımını artırmak için birçok lisans programı “Eğitim Teknolojisi” dersini içeriklerine dahil etmiştir. Buna karşın bu eğitim yeterli gelmemiş öğretmenlerin eğitimde teknolojiyi kullanma becerileri yeterli gelişim göstermemiştir (Polly, Mims, Shepher ve İnan, 2010). Bunun nedeni öğretmenlerin sahip oldukları pedagoji bilgilerini alan bilgileriyle başarılı bir şekilde harmanlayamamalarında yatıyor olabilir. Shulman (1987)’ a göre bir öğretmen en azından; içerik bilgisi, genel pedagojik bilgi, program bilgisi, pedagojik içerik bilgisi, öğrenci bilgisi, eğitimsel içerik bilgisi ve eğitsel çıktılar, amaçlar ve değerler bilgisine sahip olmalıdır. Shulman’ın bulgularına teknoloji bilgisi de eklenirse çağımızın gerekliliklerine uygun öğretmenin sahip olması gereken bilgiler tamamlanmış olacaktır. Mishra ve Koehler (2006) buna çözüm olarak Teknolojik Pedagojik Alan Bilgisi- TPAB kavramını oluşturmuşlardır (Şekil.1). Uluslararası anlamda TPACK olan bilinen Teknolojik Pedagojik Alan Bilgisinin kısaltması ilk çıktığında TPCK idi. Sonraları okunuşunun kolaylığını ve akılda kalırlığını artırmak amacıyla TPCK, TPACK’ e dönüştürülmüştür (Thompson ve Mishra, 2007). TPAB çerçevesi teknolojiyi ayrı bir alan olarak düşünmek yerine, öğretmenlerin alan, pedagoji ve teknoloji bilgilerinin birbirleriyle olan ilişkilerine, etkileşimlerine ve birbirlerine sağladıkları kolaylık ve kısıtlamalara dikkat çekmektedir. Bu modelde alan bilgisi (AB), pedagojik bilgi (PB) ve teknolojik bilgi (TB) öğretmenlerin gelişimi için gereken ana bilgilerdir. Bunun yanında bu alanlar çift olarak birleştiklerinde; Teknolojik Pedagojik Bilgi (TPB), Teknolojik Alan Bilgisi (TAB) ve Pedagojik Alan Bilgisi (PAB) oluşurken, üçü bir araya geldiklerinde TPAB’ yi oluştururlar (Mishra ve Koehler, 2006). Pedagojik Alan Bilgisi, ders içeriğine en uygun olan öğrenme yaklaşımı ve materyal seçimi ile ilgilidir (Angeli ve Valanides, 2005; Cox ve Graham,

Vatandaşlık” yeterliliğine sahip öğrenciler internet üzerinde ulaştıkları bilginin yasallık ve doğruluğuna önem verirler. İşbirliğini öğrenmeyi ve üretmeyi destekleyen teknolojiyi kullanmaya karşı olumlu tutuma sahiptirler. Aynı şekilde öğrencilerin teknolojik uygulamaları kullanabilmeleri, kavramları anlayabilmeleri ve yeni çıkan bir teknolojiyi kullanmayı eski bilgilerini transfer ederek öğrenebilmeleri gerekmektedir.

Belirlenen bu standartların öğrencilere verilmesinin bir yolu da TPAB çerçevesi ile geliştirilmiş ders planı kullanmaktır. Öğretmenin dersi planlarken pedagojiyi yani dersi anlatırken kullanacağı yöntemleri, dersin içeriğini ve derste kullanacağı teknolojiyi birlikte düşünmesi önemlidir. Bu üç önemli bileşenin birlikte uyumu düşünüldüğünde; bu uyum ne kadar güçlüyse öğrenme o kadar etkili ve akılda kalıcıdır. Bu yüzden TPAB çerçevesi öğrenciler için etkili ders planı yapabilmeyi ve onların teknoloji, pedagoji ve içeriğin dahil edildiği ders planlarıyla öğrenmeleri başarılı ve üretken bireylere dönüşmelerini sağlar (Lingenfelter, 2015).

1.1. Araştırmanın Konusu

Teknolojinin gelişmesiyle birlikte öğrencilerin derslere karşı beklentileri, öğrenme stilleri, becerileri ve çağın onlardan beklediği yeterlilikler değişmiştir. Bu çağın öğrencileri için geleneksel eğitim anlayışı yetersiz gelmekte ve teknolojinin dâhil edilmediği eğitim programlarının yetiştirdiği öğretmenler öğrencilerin bu beklentilerine cevap verememektedirler. Öğretmenler teknolojiyi eğitimle doğru bir şekilde bütünleştirememektedirler (Cox, 2008). Bunun sonucu olarak Mishra ve Koehler (2006) teknoloji, pedagoji ve alan bilgisinin öğretmenlere ya da öğrencilere bir arada verilmesi gerektiğini savunan TPAB çerçevesini geliştirdi.

Alan yazın incelendiğinde TPAB ile ilgili öğretmenlere ve akademisyenlere yönelik birçok araştırma olmasına rağmen öğrencilere yönelik olan araştırmalar sınırlıdır. Aynı zamanda Bilişim Teknolojileri dersinin TPAB çerçevesiyle hazırlanması ile ilgili bir araştırmaya rastlanmamıştır. Bu nedenle çağımızın gerekliliklerinden olan eğitimde teknoloji kullanımı ve bilgisayar eğitimi düşünüldüğünde “Bilişim Teknolojileri ve Yazılım Dersi Kapsamında TPAB Çerçevesi İle Oluşturulmuş Programlama Eğitiminin Ortaokul Öğrencilerinin Öğrenme Çıktıları Üzerine Etkileri” araştırma konusu olarak belirlenmiştir. Bu

çalışmanın Bilişim Teknolojileri ve TPAB alanında yapılacak olan çalışmalara ışık tutacağı düşünülmektedir.

1.2. Araştırmanın Amacı

Bu araştırmanın amacı, “Bilişim Teknolojileri ve Yazılım Dersi Kapsamında TPAB Çerçevesi İle Oluşturulmuş Programlama Eğitiminin Ortaokul Öğrencilerinin Öğrenme Çıktıları Üzerine Etkileri” ni belirlemektir.

Bu amaç doğrultusunda aşağıdaki sorulara yanıt aranacaktır:

1. Öğrencilerin akademik başarılarına göre ön-test ve son-test puanları arasında anlamlı bir fark var mıdır?
 - 1.1. Deney ve kontrol grubunun akademik başarılarına göre ön-testleri arasında fark var mıdır?
 - 1.2. Deney ve kontrol grubunun akademik başarılarına göre son-testleri arasında fark var mıdır?
2. Öğrencilerin problem çözme becerilerine göre ön-test ve son-test puanları arasında anlamlı bir fark var mıdır?
 - 2.1. Deney ve kontrol grubunun problem çözme becerisi algılarına göre ön-testleri arasında fark var mıdır?
 - 2.2. Deney ve kontrol grubunun problem çözme becerisi algılarına göre son-testleri arasında fark var mıdır?
3. Öğrencilerin bilgisayarca düşünme becerilerine göre ön-test ve son-test puanları arasında anlamlı bir fark var mıdır?
 - 3.1. Deney ve kontrol grubunun bilgisayarca düşünme becerilerine göre ön-testleri arasında fark var mıdır?
 - 3.2. Deney ve kontrol grubunun bilgisayarca düşünme becerilerine göre son-testleri arasında fark var mıdır?
4. Öğrencilerin akademik başarısı, problem çözme becerisi algısı ve bilgisayarca düşünme becerisi arasında anlamlı bir ilişki var mıdır?
 - 4.1. Öğrencilerin akademik başarısı ve problem çözme becerisi algısı arasında anlamlı bir ilişki var mıdır?

- 4.2. Öğrencilerin problem çözme becerisi algısı ve bilgisayarca düşünme becerisi arasında anlamlı bir ilişki var mıdır?
- 4.3. Öğrencilerin akademik başarısı ve bilgisayarca düşünme becerisi arasında anlamlı bir ilişki var mıdır?
5. Öğrencilerin akademik başarısı, problem çözme becerisi algısı ve bilgisayarca düşünme becerisi cinsiyete göre farklılaşıyor mu?
 - 5.1. Öğrencilerin akademik başarısı cinsiyete göre farklılaşıyor mu?
 - 5.2. Öğrencilerin problem çözme becerisi algısı cinsiyete göre farklılaşıyor mu?
 - 5.3. Öğrencilerin akademik bilgisayarca düşünme becerisi cinsiyete göre farklılaşıyor mu?
6. Öğrencilerin TPAB çerçevesiyle oluşturulmuş Programlama dersine ilişkin görüşleri nelerdir?

1.3. Araştırmanın Önemi

Günümüzde teknoloji hızla gelişmekte ve etki alanı gitgide genişlemektedir. Özellikle gelişmiş ülkelerde öğrencilere teknoloji destekli eğitim verilmesi yaygın olan faaliyetlerdendir. Ülkemizde de bu çalışmalara yönelik, yani teknolojiyle eğitimin birleştirilmesiyle ilgili çalışmalar yapılmaktadır. Özellikle Fırsatları Arttırma ve Teknolojiyi İyileştirme Hareketi (FATİH) projesinin hayata geçirilmesiyle birlikte öğretmenlere ve öğrencilere teknolojik açıdan verilen imkânlar hayli artmıştır. Öte yandan FATİH projesinin getirdiği yenilikleri etkili kullanmama sorunu da ortaya çıkmıştır. Bununla birlikte son günlerde gündemi meşgul eden Programlama dersi eğitiminin müfredata konması konusu da beraberinde “Bilişim Teknolojileri öğretmenleri bu ders için yeterli donanımlara sahip mi” “Bilgisayar olmayan okullarda bu ders nasıl okutulacak?” “Bu ders için öğrencilerin hazır bulunuşluluk düzeyi yeterli mi?” gibi sorular getirmiştir.

Bu bağlamda bir Bilişim Teknolojileri öğretmeni olarak Programlama eğitimini Bilişim Teknolojileri ve Yazılım dersi kapsamında öğrencilere kazandırılması düşünülmektedir. Teknoloji araçlarını pedagojik yöntemlerle uyum içinde ya da pedagojiden bağımsız olarak kullanarak ders planı oluşturulmasının farkının bilinmesi

Bilişim Teknolojileri ve Yazılım dersinin öğretilmesindeki yöntem ve tekniklerin doğru belirlenmesi açısından önemlidir. Bu nedenle bu araştırmanın gelecekte karşılaşılabilecek sorunları önceden görmeye katkı sağlayacağı düşünülmektedir.

1.4. Varsayımlar

Bu tez çalışmasında aşağıda verilen varsayımlara göre hareket edilmesi planlanmaktadır.

1. Çalışma grubunun programlama dilleri hakkında herhangi bir bilgisi yoktur.
2. Deney grubunun temel bilgisayar kullanma bilgilerine sahip olduğu düşünülmektedir.

1.5. Sınırlılıklar

Yapılacak olan bu çalışma;

1. 2017-2018 eğitim öğretim yılında 6 haftalık bir dönemle sınırlıdır.
2. Konya Kulu Ahmet Yesevi İmam Hatip Ortaokulu ve Kozanlı 75.yıl Birol Polat Ortaokulu'nda 6.sınıflarda öğrenim gören 41 öğrenciyle sınırlıdır.
3. Bilişim Teknolojileri ve Yazılım dersi "Problem Çözme ve Programlama" ünitesi "Programlama" konusu ile sınırlıdır.
4. Araştırma kapsamında toplanacak verilerle sınırlıdır.

1.6. Tanımlar

Alan Bilgisi: Alan bilgisi öğretilecek ya da öğrenilecek konu hakkındaki bilgidir. Alan hakkındaki temel bilgiler, kavramlar, teoriler; fikirleri düzenleyen ve birbirine bağlayan açıklayıcı çerçeveler; kanıt ve ipucu kurallarını içerir.

Pedagojik Bilgi: Öğrenme ve öğretme yöntemleri, pratiği ve süreci ve öğrenme kazanımlarının nasıl verileceği hakkındaki bilgidir. Öğrencinin öğrenmesi, sınıf yönetimi, ders planı geliştirme gibi bilgileri içerir. Sınıf içinde uygulanabilecek

yöntem ve teknikler, hedef kitlenin özellikleri ve öğrenciyi ölçmek için uygulanan stratejileri kapsar.

Teknolojik Bilgi: Teknolojik araçların işletim sistemi ve bilgisayar donanımı bilgisi, kelime işlemci, elektronik çizelge, web tarayıcıları ve e-posta programları gibi standart yazılımları kullanabilme bilgisidir. Çevresel aygıtları, programları kurma ve kaldırma ve dosya yönetimi gibi bilgileri de içerir. Teknoloji sürekli geliştiği için sürekli güncellenmesi gerekir.

Pedagojik Alan Bilgisi: alan ve pedagoji bilgisinin kesişimidir. Anlatılacak içeriğe uygun olan öğrenme-öğretme yaklaşımlarını ya da öğrenme-öğretme yaklaşımına uygun içeriği bilmektir. Öğrencinin giriş bilgilerine göre ve kavramın özelliklerine göre anlatımın şekli değişecektir.

Teknolojik Alan Bilgisi: İçeriğe uygun teknolojiyi kullanabilme bilgisidir. Bazı dersler için oluşturulan yazılımlar o derslerin içeriğinin daha kolay anlaşılmasını sağlıyor.

Teknolojik Pedagojik Bilgi: Teknoloji kullanımının öğrenmeyi nasıl etkileyeceğinin bilinmesidir. Derse katılım, değerlendirme, sohbet odaları ve tartışma ortamları gibi derse yararlı uygulamalar sağlayan teknolojik araç gereçlerin bilgisini içerir.

Teknolojik Pedagojik Alan Bilgisi: Üç bilgiyi de içine alan bilgi türüdür. Bir öğretmenin teknoloji kullanımı, öğretim yöntemleri ve alan bilgisinin karışımı birbirleriyle olan ilişkilerini kapsar.

2. TEORİK ÇERÇEVE

Bu bölümde araştırmanın konusunun alanyazında hangi aşamalardan geçip oluştuğuyla ilgili bilgilere yer verilmiştir. Teknolojinin Eğitimde Kullanılmasının ve Bilişim Teknolojileri dersinin okutulması ile ilgili alanyazın incelenmiş ve bu bölüm altında toplanmıştır.

2.1 Teknolojinin Eğitimde Kullanılması

Teknolojinin Eğitimde Kullanılması; “Eğitim/Öğretim Teknolojileri”, “Bilgi ve İletişim Teknolojileri (BİT) ve Eğitim”, “Teknolojinin Eğitime Entegrasyonu ve TPAB” olmak üzere 3 ayrı bölümde incelenmiştir.

2.1.1 Eğitim/Öğretim Teknolojileri

İlk olarak 1963’te görsel-işitsel iletişim cihazlarının eğitim-öğretim sürecinde kullanılmasıyla başlayan “Eğitim Teknolojisi” terimi AECT tarafından (Association for Educational Communications and Technology), uygun teknolojik süreçleri ve kaynakları oluşturmak, kullanmak ve yönetmek suretiyle öğrenmeyi kolaylaştırmak ve performansı arttırmak için yapılan çalışma ve etik uygulamalar bütünü olarak tanımlanır (Januszewski ve Persichitte, 2008).

Richey ve Seels (1994)’ e göre Öğretim teknolojileri, öğrenme kaynaklarının öğrenciye sunulmasını tasarım, geliştirme, kullanım/uygulama, yönetim, değerlendirme ve analiz olmak üzere 6 ana bölüme ayrılan süreçte gerçekleştirilmesini sağlayan teori ve uygulamalardır. Buna göre teori ve pratik birbirinden ayrı düşünülemez ve ÖT terimi öğretim için kullanılan tüm kaynakları ve bu kaynakların kullanıldığı eğitim-öğretim sürecini kapsar. Yani gelişen teknolojinin gücünden en iyi düzeyde yarar sağlamak öğretim tasarımcılarının yaratıcılıklarıyla medyayı kullanma yeteneklerine ve bu ikisi arasındaki ilişkinin iyi anlaşılmasına bağlıdır (Kozma, 1994).

Genelde Öğretim Tek. terimi alanın dışındaki kişilerce “öğretim medyası” yani eğitim-öğretimde kullanılan görsel-işitsel cihazları tanımlamak için kullanılmakta ve alanın bilim adamlarının tanımlarda kullandıkları süreç yani medyanın eğitimde nasıl kullanıldığı unutulmaktadır. Bu nedenle bu şekilde kullanım eksik kalmaktadır çünkü

öğretim teknolojisi hem sistem yaklaşımı hem de görsel-işitsel cihazları kapsamaktadır (Reiser, 1987). Tanımdaki bu karmaşıklığı netleştirmek üzere Reiser, Öğretim Teknolojilerini “öğretim medyası” ve “öğretim tasarımı” olmak üzere 2 ana başlık altında toplamıştır. (Reiser, 2007). Öğretim medyasını “öğretimi sunmak için kullanılan öğretmen, defter ve tahta hariç tüm fiziksel araç-gereçler” olarak, Öğretim tasarımı ise “öğretim medyasını sunma metot ve yöntemleri” olarak tanımlamıştır. Reiser’e göre öğretim medyasını etkili bir şekilde kullanmak ancak iyi planlanmış bir öğretimle mümkündür ve iyi bir medyayla eşleştirilen bir öğretim tasarımı eğitim/öğretimin gelişimi için gereklidir.

Öğretim teknolojisinin tanımı dönemden döneme farklılık gösterir. Her dönemde öne çıkan uygulamaların bu farklılığa neden olduğu söylenebilir. Buna göre Öğretim teknolojilerinin dönemlerini şu şekilde sıralayabiliriz:

1. Görsel hareket dönemi
2. Kayıtlı ses dönemi
3. Hareketli görüntü dönemi
4. Sınırlı etkileşim dönemi
5. Çoklu ortam dönemi
6. Sayısal ağlar dönemi

Bu bağlamda içinde bulunduğumuz dönem, sayısal ağlar dönemi olup öne çıkan uygulamaları web 2.0 teknolojileridir ve felsefi akım olarak yapılandırmacı yaklaşım ve performans teknolojileri önem kazanmıştır (Turan vd., 2016).

2.1.2 Bilgi ve İletişim Teknolojileri (BİT) ve Eğitim

Eğitimde bilgisayar kullanımı uygun maliyetli mikrobilgisayarların üretiminin yaygınlaşmasıyla 1980’lerde popüler olmuştur. Bu süreçte birçok teknoloji firması kendi bilgisayarını üretmiş ve okullara dağıtmaya başlamıştır. 1980’lerin sonunda bilgisayarlar bilgi alımını sağladıkları için “bilgisayar” teriminin yerini BT (Bilgi Teknolojileri) terimi almıştır. Daha sonra BT yerine BİT (Bilgi ve İletişim Teknolojileri) terimi kullanılmıştır (Pelgrum ve Law, 2003). "BİT" terimi, bilgisayarları, İnternet’i ve elektronik dağıtım sistemlerini içerir ve günümüz eğitim

alanında yaygın olarak kullanılmaktadır (Chuang, Weng ve Huang, 2015). Pelgrum ve Law (2003)'a göre eğitimde BİT kullanımı 3 farklı şekilde olmaktadır. Bunlar: “BİT’i öğrenmek”, bu günümüz BT derslerine karşılık gelmektedir, “BİT’le öğrenmek”, herhangi bir dersi öğretirken teknolojiden yararlanmak ve sonuncusu “BİT tabanlı öğrenmek” yani BİT’i eğitimde yardımcı kaynak olarak değil de eğitimin tamamını BİT altyapısıyla öğrenciye sunmak. BİT tabanlı öğrenmek deyimi yerine bugün BİT’in eğitime entegre edilmesi kullanılmaktadır. “Entegre” kelimesi iki ya da daha fazla şeyi daha etkili bir yapı oluşturmak amacıyla bir araya getirmek anlamına gelmektedir. Türk Dil Kurumu’na göre “entegrasyon” bütünleşme ve uyum demektir. Eğitimde Teknoloji Entegrasyonu ise teknoloji ile eğitimi bütünleştirmek yani “İstendik öğrenme çıktılarını gerçekleştirmek için eğitim teknolojisinin etkili bir şekilde uygulanması” (Davies ve West, 2013) şeklinde açıklanabilir.

BİT’in, okullarda eğitime entegrasyonu sürecinin nasıl ilerlediğini belirlemek ve olası sorunlara çözümler üretmek amacıyla birçok çalışma yapılmıştır. ACOT (Apple Classrooms of Tomorrow/Yarının Apple Sınıfları) bunlardan ilki sayılabilir. Bu çalışma kapsamında öğretmen ve öğrencilere tarihte ilk kişisel bilgisayar dağıtımı gerçekleştirilmiştir. Sonrasında LoTİ (Levels of Technology Integration/ Teknoloji Entegrasyon Düzeyleri) çalışması yapılmış ve öğretmenlerin genel anlamda teknolojiyi sadece aktiviteleri zenginleştiren ilave bir araç olarak kullandığı ortaya çıkmıştır. Benzer şekilde NCES (Eğitsel İstatistikler Ulusal Merkezi)’nin yaptığı araştırmada da öğretmenlerin çoğunluğunun bilgisayarda kelime işlemci ve elektronik tablola gibi basit programlar kullandığı belirlenmiştir. CRITO (Bilişim Tek. Araştırma Merkezi ve Organizasyonları)’nın yaptığı araştırmada ise bilgisayarların daha çok teknolojiyle alakalı derslerde kullanıldığı diğer derslerde uzak kaldığı sonucuna varılmıştır (Barron, Kemker, Harnes ve Kalaydjian, 2003).

BİT’in eğitime entegre edilmesini yaygınlaştırmak amacıyla çok sayıda kâr amacı gütmeyen organizasyonlar kurulmuştur. Bunlardan dünya çapında üne ulaşmış olanlar arasında; ISTE (International Society for Technology in Education), ACM (Association for computing machinery), AECT (Association for educational communications and technology), NMC (New Media Consortium) ve NCATE sayılabilir.

ISTE vakfı, her öğretmenin bilmesi gereken ve teknolojiyle neler yapabileceği gerektiğine dair bir dizi kılavuz oluşturmuştur. Eğitimcilerin, teknolojiyi sınıfta bulunan diğer standart araçlar gibi kullanmaları için ihtiyaç duydukları becerileri açıklamıştır (Wiebe ve Taylor, 1997). Bu beceriler ISTE standartları olarak tanımlanır ve bu standartlara kısaca NETS (National Educational Technology Standards) denir. Bu standartları geliştirmeye yönelik çabalar 1989 yılında başlamıştır. Öğretmenlik okuyan öğrencilerin teknoloji kullanmadaki yeterliliklerini belirlemek amacıyla araştırma yapılmış, bu araştırma sonucunda elde edilen bilgiler bilgisayar ve teknoloji alanındaki farklı dereceler ve resmi kuralların belirlenmesi için hangi kriterlerin kullanılacağını belirlemek amacıyla kullanılmıştır (Handler ve Strudler, 1997). Bu standartlar, teknoloji temelli araçlardan yararlanan gelişmiş öğrenme deneyimleri sağlamak için tüm öğretmen adaylarının ve öğretmenlerin ihtiyaç duyduğu becerilerin bir açıklamasını sağlar (Friske, Knezek, Taylor, Thomas ve Wiebe, 1995). Bu bağlamda CAEP (Council for the Accreditation of Education Program/ Eğitim Programının Akreditasyonu Konseyi) (CAEP, 2016), CCSSO (Council of Chief State School Officers/ Uzman Devlet Okulu Görevlileri Konseyi) (CCSSO, 2013) ve ISTE gibi kuruluşlar öğretmenler için internet nesline uygun standartlar geliştirmişlerdir. ISTE 21. Yüzyıl eğitimcilerinin sahip olması gereken özellikleri aşağıdaki gibi sıralamıştır:

1. Öğrenci

Eğitimciler, pratik bilgilerini; başkalarından ve başkalarıyla öğrenerek ve yükselen teknolojiyle birlikte kanıtlanmış, gelecek vaat eden uygulamaları öğrencilerinin öğrenmelerini artırmak için keşfederek sürekli geliştirirler.

2. Lider

Eğitimciler, öğrenciyi güçlendirmek, başarısını desteklemek ve öğretme ve öğrenmeyi geliştirmek için liderlik fırsatları ararlar.

3. Vatandaş

Eğitimciler, öğrencilere dijital dünyaya olumlu katkıda bulunma ve sorumlu bir şekilde katılma konusunda ilham verir.

4. İşbirlikçi

Eğitimciler, uygulamayı geliştirmek, kaynaklarla fikirleri keşfetmek ve paylaşmak ile sorunları çözmek için hem meslektaşları hem de öğrencilerle iş birliği yapar.

5. Tasarımcı

Eğitimciler, öğrenci farklılığını tanıyan otantik, öğrenci merkezli etkinlikler ve ortamlar tasarlar.

6. Kolaylaştırıcı

Eğitimciler, öğrencilerin ISTE Standartlarında başarısını desteklemek için teknoloji ile öğrenmeyi kolaylaştırır.

7. Analist

Eğitimciler, eğitimlerini sürdürmek ve öğrencileri öğrenme hedeflerine ulaşmalarında desteklemek için verileri anlar ve kullanır (ISTE, 2017).

Öğretmenlerin yanında bugünün öğrencileri için de artık çeşitli tanımlamalar ve standartlar mevcuttur. 21. Yüzyıl öğrencileri Prensky (2001) tarafından bilgisayar, video oyunları ve İnternet'in dilinden doğuştan anladıkları için "dijital yerli" olarak adlandırılmışlardır. Dijital Yerliler bilgiyi hızlı almaya alışkındırlar. Paralel işlem ve çoklu görevden hoşlanırlar. Konu ile ilgili grafikleri yazılı metinlerden önce görmeyi tercih ederler. Rastgele erişimi tercih ederler (Bağlantı-Köprü gibi). Birlikte bir ağa bağlandıklarında en iyi şekilde çalışırlar. Anında tatmin ve sık ödüller kazanma beklentisi içindedirler. Oyun oynamayı "ciddi" çalışmalara tercih ederler (Prensky, 2001). Benzer şekilde Z-Kuşağı (Strauss ve Howe, 1991) olarak da adlandırılan bu grup Tapscott (1999) tarafından İnternet'in içinde doğan ve çoğu işlerini çevrimiçi yapmayı tercih ettikleri için Net Generation yani "İnternet Kuşağı" olarak tanımlanmıştır. Net Gen üyeleri bilgisayar teknolojisini bütün hayatlarına koyduklarından dolayı daha önceki nesillere göre teknolojiye daha yatkınlardır. Bu, alışılanın aksine genç kuşakların yaşlılara rehberlik etmesini ve onları eğitmesini gerektiren bir durum ortaya çıkarmıştır. Net Gen üyeleri, medyayı öncekilerden farklı kullanmaktadırlar. Yani, internetteki bilgiye ulaşan Pasif alıcılardan ziyade, Net

Genler daha aktiftir, kendi içeriklerini oluştururlar (Tapscott, 2009). Bu bağlamda, günümüz öğrencilerinin sahip olması gereken beceriler aşağıda verilmiştir:

1. Öğrenmeyi Güçlendirme

Öğrenciler, öğrenme hedeflerini seçme, gerçekleştirme ve yeterliliklerini göstermede aktif rol almak için teknolojiden yararlanırlar.

2. Dijital Vatandaşlık

Öğrenciler birbirine bağlı bir dijital dünyada yaşama, öğrenme ve çalışma haklarını, sorumluluklarını ve fırsatlarını tanırlar ve güvenli, yasal ve etik olan şekillerde hareket ederler.

3. Bilgiyi Yapılandırma

Öğrenciler bilgiyi inşa etmek, yaratıcı eserler üretmek ve kendileri ve başkaları için anlamlı öğrenme deneyimleri yapmak için dijital araçları kullanarak çeşitli kaynakları araştırırlar.

4. Yenilikçi Tasarımcılık

Öğrenciler yeni, kullanışlı ya da yaratıcı çözümler oluşturarak problemleri tanımlamak ve çözmek için bir tasarım süreci içerisinde çeşitli teknolojiler kullanırlar.

5. Bilgisayarca Düşünme

Öğrenciler, problemleri anlamak ve çözmek için teknolojik yöntemlerin gücünü kullanan şekillerde problemleri anlama ve çözme stratejileri geliştirir ve kullanır.

6. Yaratıcı İletişim

Öğrenciler, amaçlarına uygun platformları, araçları, stilleri, formatları ve dijital medyayı kullanarak çeşitli amaçlarla kendilerini açıkça ve yaratıcı bir şekilde ifade eder.

7. Küresel İş birliği

Öğrenciler, kendi perspektiflerini genişletmek ve başkalarıyla iş birliği yapıp yerel ve küresel olarak ekipler halinde etkin bir şekilde çalışarak öğrenmelerini

zenginleştirmek için dijital araçları kullanırlar (ISTE, 2016). Yukarıda bahsedildiği üzere, bu standartlar gösteriyor ki öğrencilerin ve öğretmenlerin çağa ayak uydurması için teknoloji ile öğrenmek ve öğretmek oldukça önemlidir. Eğitimi bu standartlarla uyum içerisinde geliştirmek için birçok çalışma yapılmıştır. Bunun için Bilgi ve İletişim Teknolojilerinin eğitime entegrasyonu araştırmacılar tarafından çoğunlukla incelenmiş ve bulunan sorunlara çözüm aranmıştır.

2.1.3 Teknolojinin Eğitime Entegrasyonu ve TPAB

Teknolojinin eğitime dahil edilmesi ile ilgili süreçte çeşitli sorunlarla karşılaşmıştır. Bunlar içinde Bilgi ve İletişim Teknolojileri (BİT)'nin eğitimde bir kavram olarak faydalarının anlaşılması, değişime direnç gösterme, uygulamalar için gerekli bilgi ve beceri eksikliği, oluşturulan materyallerin hedef ve amaçlara uygunsuzluğu, zamanında güncelleme yapılmaması, ekonomik kaynakların BİT eğitime değil sadece donanım için harcanması, BİT müfredatının batıdan yayılmasından dolayı doğu ülkelerinde milli ihtiyaçları karşılayamaması, öğrenci merkezli öğretimin uygulanmaması, teknolojiye sınırlı erişim, değerlendirme yazılımı eksikliği, sınavlara yönelik eğitim, sınavların yüksek seviye becerileri değerlendirememesi sayılabilir (Laferriere, Hamel ve Searson 2013). Benzer şekilde Türkiye'de teknolojiyi eğitime entegre etmede karşılaşılan temel sorunlar arasında; sınıflarda BİT ekipmanlarının bulunmaması, BİT tabanlı öğretim kaynaklarının eksikliği, öğretmenlerin inançları ve uygulamaları, geleneksel yaklaşımların etkileri, öğretmenlerin hizmet içi eğitimi ile ilgili sorunlar, zaman eksikliği ile ilgili problemler sayılabilir (Unal ve Ozturk, 2012).

Bunun yanı sıra teknolojinin eğitimde gelişiminin bir sonucu olarak etkileşimli tahtalar ortaya çıkmaya başlamış ancak onları etkili bir şekilde kullanabilen bireyler yetiştirilememiştir. Bunun sonucunda etkileşimli tahtalar geleneksel görsel-işitsel araçların yerini alsa da hala dijital teknoloji araçlarının yoğun halde bulunduğu okullarda basılı materyaller en sık kullanılan eğitim-öğretim aracıdır (Area, Hernández ve Sosa, 2016).

Aslında; okul yönetimi, okul ortamı ve öğretmenlerin teknolojiye yeteri kadar adapte olamamalarından dolayı teknolojik araçlardan okullarda gerektiği şekilde ve

miktarda yararlanılamamaktadır (Zhao ve Frank, 2003). Öğretmenler teknolojiyle yüksek oranda etkileşimde olsa bile; teknolojiyi nadiren kullanmakta ve kullandıklarında da yenilikçi eğitim-öğretim yazılımları yerine kelime işlemci ve sunum programları gibi basit programlar için kullanmaktadırlar (Cuban, Kirkpatrick ve Peck 2001; Area, Hernandez ve Sosa, 2016; Kazu ve Yavuzalp, 2010; Seferoğlu, Akbıyık ve Bulut 2008). Oysa teknoloji entegrasyonunun verimli bir şekilde gerçekleşmesi için en önemli etken öğretmenlerdir çünkü öğretmenler değişimin öncüleridir ve önce onların gerekli desteği alarak değişime adapte olmaları gerekir. Bu bağlamda öğretmenlere teknolojik donanımına sahip ortam sağlanması gerektiği gibi onların da eğitim teknolojisi yani hangi teknolojiyi hangi pedagojiyle birlikte kullanmaları gerektiği bilgilerini geliştirmek gerekir (Dexter, Anderson ve Becker, 1999). Yani, teknolojinin eğitimle etkili bir şekilde bütünleşmesini sağlamak için öncelikle öğretmenleri eğitmek gerekir (Inan, Lowther, Ross ve Strahl, 2010; Zhao ve Frank, 2003).

Pierson (2001), teknoloji entegrasyonundaki sorunları ortadan kaldırmak amacıyla, öğretmenlerin belirli bir konuyu öğretmenin en etkili yolunu bilmesi anlamına gelen pedagojik içerik bilgisine teknolojinin dahil edilmesi gerektiğini belirtmiştir. Teknolojiyi etkili bir şekilde eğitimle bütünleştiren öğretmen, kapsamlı bir şekilde içerik ve pedagojik alanlardan yararlanabilecektir ve bu bilgi birikimiyle teknolojik bilgiyi bir araya getirecektir. Pierson (2001) üç bilgi alanının kesişimini yani teknolojik-pedagojik içerik bilgisini, etkili teknoloji entegrasyonu olarak tanımlar.

Pierson (2001)' a göre ihtiyacımız olan öğretmenler sadece bilgisayarları kullanmayı bilenler değil, her türlü eğitim-öğretim aracını öğrencilerin yararı için etkin bir şekilde kullanmayı bilen öğretmenlerdir. Bu da Mishra ve Koehler (2006)'in çerçeve olarak yayımladığı Teknolojik Pedagojik Alan Bilgisi (TPAB) ile mümkündür.

2.2. Bilgisayar Eğitimi

2.2.1. Türkiye’de Eğitim Teknolojileri ve Bilgisayar Öğretimi

Türkiye’de Eğitim Teknolojileri ile ilgili tartışmalar ilk olarak 1970’li yıllarda başlamıştır. Bu yıllarda, teknoloji ürünü araç ve gereçlerin yeni bir yaklaşımla öğrenme-öğretme sürecine dahil edilmesi durumu gündeme gelmiştir. Aynı zamanda üniversitelerde de, eğitim teknolojisi alanında araştırmalar yapılmış ve bu alana uygun öğretim programları uygulamaya konmuştur (Akkoyunlu ve İmer, 1998). 1984 yılından itibaren MEB tarafından teknoloji kullanımını artırmak amacıyla çeşitli projeler ve girişimler yapılmaya başlanmıştır (Eryılmaz ve Salman, 2014). 1989 yılında eğitim niteliğinin yükseltilmesini sağlamak amacıyla hazırlanan “Altıncı Beş Yıllık Kalkınma Planı” ve 1996 yılında hazırlanan “Yedinci Beş Yıllık Kalkınma Planı” kapsamında bilim ve teknolojiye gelişmeler ışığında öğretim programlarının güncellenmesi için gerekli hedefler belirlenmiş ve çeşitli projeler yoluyla bu hedeflere ulaşılmaya çalışılmıştır (Sezer, 2011).

Bu projelerden en önemlileri arasında BDO (Bilgisayar Deneme Okulu), BLO (Bilgisayar Laboratuvar Okulu) ve MLO (Müfredat Laboratuvar Okulları) sayılabilir. Bu projelerin sonucunda yapılan araştırmalara göre okulların donanımsal eksiklikleri giderilse de öğretmenler, okul müdürleri ve müfredatın eğitimde teknoloji kullanımı konusunda yetersiz olduğu ve Bilişim Teknolojileri (BT) sınıflarının amacına uygun kullanılmadığı ortaya çıkmıştır (Akbaba-Altun, 2004; Akbaba-Altun, 2005; Özdemir, 2010).

1993 yılında ODTÜ ve TÜBİTAK iş birliğiyle Türkiye’nin internet bağlantısı sağlanmıştır (Wolcott ve Cagiltay, 2001). Okullara internet bağlantısı sağlanması ilk olarak Temel Eğitim Projesi’nde belirtilmiş ve projenin birinci fazı kapsamında 2.802 İlköğretim okuluna bilgisayar, yazıcı, televizyon, tepegöz ve bilgisayar yazılımlarının yer aldığı, internet erişimi bulunan Bilgi Teknolojisi (BT) sınıfları oluşturulmuş ve bu sınıflara video, video kasetleri ve tepegöz saydamı sağlanmıştır (MEB, 2001). World Links Projesi’nde öğretmen ve öğrenciler arasında uluslararası iletişimi sağlayarak iş birliğine dayalı projeler oluşturulup internetin öğretmen ve öğrenciler tarafından kullanımının öğrenilmesi ve yaygınlaştırılması amaçlanmıştır (Sezer, 2011). Türk

Eđitim Sisteminin davranışçı özellikleri de o yıllarda masaya yatırılmış ve artık yapılandırmacı yaklaşımın benimsenmesi gerektiđi önerilmiştir (Özden ve Şimşek, 1998).

Temel Eđitim projesi kapsamında 2006-2007 eğitim-öđretim yılından itibaren bilgisayar okuryazarlığı ilköđretim birinci sınıftan itibaren “Bilgisayar” dersi adı altında seçmeli olarak öđrencilere kazandırılmaya başlanmıştır (MEB, 2007). 1998 yılında uygulamaya konulan programda ise, “Seçmeli Bilgisayar” dersi 4, 5, 6, 7 ve 8. sınıflarda okutulmaktaydı (TTKB, 2006). 2007 yılında alınan bir kararla dersin adı “Bilişim Teknolojileri” olarak deđiştirilmiş ve ders saati 4. ve 5. sınıflarda 2 saate çıkarılmıştır (TTKB, 2007). 1993 yılında ilk kez oluşturulan BT sınıfları bulunan okullarda öđretmenleri teknoloji kullanımı hakkında bilgilendirmek amacıyla ortaya çıkan BT koordinatör öđretmeni yani “formatör öđretmen” (MEB, 1993) kavramı Temel Eđitim projesi ile birlikte yaygınlaşmıştır. Proje kapsamında, belirtilen konularda il milli eğitim müdürlüklerine danışmanlık ve rehberlik yapmak üzere 250 eğitimci bilgisayar formatör öđretmene eğitim verilmiştir. Bakanlık tarafından bilgi teknolojileri alanında eğitilen eğitimci bilgisayar formatör öđretmenlerinden, kazandıkları bilgi ve beceriyi, il genelinde formasyon kursları düzenleme, BT sınıflarının denetimi için plan hazırlama, uygulama ve deđerlendirme raporları hazırlama, yazılımların kullanılmasını sağlama gibi konularda hem bilgisayar formatör öđretmenleri ile paylaşması hem de il milli eğitim müdürlüklerine danışmanlık yapması istenilmiştir. BT sınıfı bulunan her okulda en az bir formatör öđretmenin olmasını hedefleyen Bakanlık, bilgisayar formatör öđretmenlerini “deđişimin öncüleri” olarak isimlendirmiştir (MEB, 2007).

1998 yılında müfredata konulan “ilköđretim seçmeli bilgisayar” dersini okutacak öđretmen ihtiyacını karşılamak amacıyla aynı yıl Bilgisayar ve Öđretim Teknolojileri Eğitimi (BÖTE) bölümü kurulmuştur (YÖK, 1998). BÖTE bölümünün Yüksek Öđretim Programı tarafından belirlenen ilk öđretim programında bölüm dışı dersler fazlasıyla yer almıştır. Bunu deđiştirmek amacıyla YÖK, alanla ilgili derslere yoğunluk vererek öđretim programını 2007 yılında güncellemiştir (YÖK, 2007). 2002 yılından bu yana mezun veren BÖTE programları, üniversitelerin BÖTE bölümlerinde akademisyen, Millî Eğitim Bakanlığı’na bađlı devlet ve özel statülü okullarda

öğretmen, danışman, eğitim programı danışmanı yetiştirilmektedir. Ayrıca BÖTE mezunları, öğretim tasarımı, eğitim yazılım ve materyali geliştirme ve değerlendirme alanlarında eğitim teknolojisi olarak görev alabilirler (Altun ve Ateş, 2008). Ancak “formatör öğretmen” kavramının ortaya çıkışı zaman içerisinde BÖTE mezunlarından yukarıda sayılan niteliklerinden farklı özelliklere sahip olması gerektiği algısına ve bu doğrultuda okullarda BT öğretmenin farklı görevler (BDE görevlisi, e-okul yöneticisi, webmaster, ödev yardımcısı, vb.) üstlenmesine neden olmuştur (Topu ve Göktaş, 2009).

2003 yılında MEB’e bağlı okul ve kurumlara kesintisiz internet bağlantısı sağlamak amacıyla “Okullara İnternet Projesi” kapsamında Türk Telekom A.Ş ile MEB ve Ulaştırma Bakanlığı bir protokol imzalamıştır. Bu projenin amacı sağlanan internet sayesinde öğretmen ve öğrencilerin e-devlet uygulamalarını kullanımını kolaylaştırmak ve eğitimde kaliteyi artırmak olmuştur (Sezer, 2011).

Buraya kadar bahsedilen projeler eğitimde teknoloji alanında ilerleme sağlasa da BİT’in eğitim politikaları içerisinde tam olarak yer almasının sağlanması 2003 yılında internetin yaygınlaşmasından sonra ortaya konulan E-Türkiye çalışmaları ile olmuştur ve bu tarihten itibaren BİT’e yönelik somut politika ve hedefler uygulamaya konulmuştur (Bayrakçı, 2005). Bunlardan en geniş kapsama sahip olanı kuşkusuz “Fırsatları Arttırma ve Teknolojiyi İyileştirme Hareketi” bilinen adıyla FATİH projesidir.

FATİH projesi 22.12.2010 Tarihinde Millî Eğitim Bakanlığı ve Ulaştırma Bakanlığı arasında imzalanan bir protokolle başlatılmıştır (Akgün, Yılmaz ve Seferoğlu, 2011). Projenin amacı, eğitim ve öğretimde fırsat eşitliğini sağlamak ve okullarımızdaki teknolojiyi iyileştirmek amacıyla bilişim teknolojileri araçlarının öğrenme-öğretme sürecinde daha fazla duyu organına hitap edecek şekilde, derslerde etkin kullanımını sağlamaktır (MEB, 2011). Proje kapsamında tüm okullara etkileşimli tahta (ET), tablet bilgisayar ve internet ağı altyapısı sağlanması öngörülmektedir (Okulöncesi ve ilkökul düzeyinde ET’ler, ortaokul ve lise düzeyinde ise ET’ler ve tablet bilgisayarlar sağlanacak). Bu sayede, eğitimde fırsat eşitliğinin yanı sıra, eğitim ve öğretim süreçlerinde BT kullanımının geliştirilmesi amaçlanmıştır. FATİH ile

Türkiye çapında 40.000 okul ve 620.000 sınıfa BT donanımı kurulacaktır. İlk olarak ortaöğretim düzeyinde uygulamaya konulan, 2011 ve 2019 yılları arasındaki süreçte ise tüm eğitim düzeylerine yaygınlaştırılması planlanmıştır (RTI ve ERG, 2013). FATİH projesinin başlıca bileşenleri; donanım, erişim, veri merkezi, yönetim yazılımları, içerik yazılımı, içerik, sosyal paylaşımlar, destek ve öğretmen eğitimidir (MEB, 2011). Projenin “Eğitsel e-içeriğin sağlanması ve yönetilmesi bileşeni” kapsamında Eğitim Bilişim Ağı (EBA) Yenilik ve Eğitim Teknolojileri Genel Müdürlüğü (YEĞİTEK) tarafından tasarlanarak 26.12.2012 tarihinde internette yayına başlamıştır (Fiş Erümit, Gedik ve Göktaş, 2016). EBA sayesinde öğrenme öğretme süreci okul içiyle sınırlı kalmayıp okul dışına da taşınmış, sunulan çeşitli, zengin ve eğitici içerikler yardımıyla öğrenci, öğretmen ve velilerin ihtiyaçları karşılanmış ve sosyal ağ yapısı yardımıyla birbirleriyle bilgi paylaşımı yapmaları sağlanmıştır (Tüysüz ve Çümen, 2016).

2012 yılında temel eğitim projesiyle uygulamaya geçen 8 yıllık zorunlu eğitim kaldırılmış 12 yıllık zorunlu eğitim sistemine geçilmiştir. Buna göre öğrencilere 4 yıllık ilköğretim 4 yıllık ortaokul ve 4 yıllık lise eğitimleri sonunda diploma verilmesi kararlaştırılmıştır (MEB, 2012a). Aynı yıl akademisyenler tarafından devlete sunulan Fatih Projesi Çalıştay Raporu’nda Bilişim Teknolojileri dersinin yeniden zorunlu kredili ders olarak programlara konması gerektiği belirtilmiştir (Yardımcı vd., 2012). Bunun üzerine bilişim teknolojileri dersinin ismi “Bilişim Teknolojileri ve Yazılım” olarak değiştirilmiş ve program içeriğinde programlama ve yazılım konuları daha yoğun olarak işlenmiştir. Ünite adları genel anlamda değiştirilmiş ve “Problem Çözme, Programlama ve Özgün Ürün Geliştirme” adlı ünite eklenmiştir (TTKB, 2012). 2013 yılında Bilişim Teknolojileri ve Yazılım dersi ilk kez 5. ve 6. Sınıflarda zorunlu olmuştur ve 7. ve 8. Sınıflarda seçmeli olarak okutulmaya devam edilmiştir (BTE, 2013).

Yine 2012 yılında üzerinde ağır görev yükü bulunan okul bilişim teknolojileri formatör öğretmenliği kaldırılmış (MEB, 2012c), Bilişim Teknolojileri Rehberliği görevlendirmesi yapılmaya başlanmıştır (MEB, 2012b). 2015 yılında Bilişim Teknolojileri Rehber Öğretmenliği (BTR) adı Fatih Projesi BT Rehberliği olarak değiştirilmiştir (MEB, 2015).

2017-2018 eğitim öğretim yılında son BT ve Yazılım dersi öğretim programı kabul edilmiştir (TTKB, 2018). Bu programda “Programlama” konusunun üzerinde bir önceki programa göre daha fazla durulmuştur. Son yayınlanan BTY dersi öğretim programı hakkında araştırmacıların fikirleri bu öğretim programına uyacak şekilde BÖTE lisans eğitim programının da değişmesi ve hali hazırda çalışan öğretmenler içinse hizmet içi eğitimler düzenlenmesi gerektiği yönündedir (Mercimek ve İlic, 2017).

Aynı zamanda ülkemizde teknoloji ve eğitim üzerine gelişen olaylar dikkate alındığında BT öğretmenlerinin görev sorumluluklarının diğer öğretmenlere verecekleri eğitsel yazılım tasarlama, programlama, geliştirme ve değerlendirme eğitimlerini içerecek şekilde yeniden düzenlenmesi, Programlama dersinin daha erken yaşlarda verilmeye başlanması ve bu sayede tüketen değil üreten bireyler yetiştirilmesi araştırmacılar tarafından önerilmektedir (Demirer ve Sak, 2015).

2.2.2. Bilgisayar Eğitiminin Dünyadaki Yeri

Bilgisayar dersiyle ilgili son zamanlarda yapılan araştırmalar incelendiğinde bilgisayar dersinin içeriğinin bilgi teknolojilerinden bilgisayar bilimine, programlamaya, oyun geliştirmeye ve robotik kodlamaya kaydığı görülür. Bu bağlamda İngiltere’de BT (Bilişim Teknolojileri) dersinin Programlama (Computing) dersine dönüşmesi üzerine lise öğrencileriyle Lego robotları ve Scratch arayüzü kullanılarak yürütülen araştırma sonucunda öğrencilerin çoğunun bu şekilde ders işlemekten keyif aldıkları ve bilgisayar bilimlerine olan ilgilerinin arttığı belirlenmiştir ve hatta bazı öğrencilerin bilgisayar bilimiyle hobi olarak ilgilenmek istedikleri görülmüştür (Grout ve Houlden, 2013). Flip adında bir kodlama diliyle 12-13 yaşlarındaki öğrencilere uygulanan 8 haftalık eğitim sonucunda öğrenciler kendi etkileşimli oyunlarını üretebilmişlerdir. Araştırmanın sonucunda, öğrencilerin bilgisayarca iletişim becerisinde önemli derecede bir gelişim olduğu, kızların erkeklere göre daha karmaşık kodlar yazdığı ve kız öğrencilerin erkek öğrencilere göre daha başarılı olduğu ortaya çıkmıştır (Howland ve Good, 2014). Benzer şekilde Ortaokul kız öğrencilerinin “Stagecast Creator” programıyla oyun oluşturmalarının sağlandığı araştırma sonucunda oyun geliştirmenin öğrencilerin bilgisayar bilimlerini

anlamalarını kolaylaştırdığı ve desteklediği ortaya çıkmıştır (Denner, Werner ve Ortiz, 2011). Ortaokul öğrencileriyle Robotik Kodlama dersinin işlendiği bir araştırmada öğrencilerin bu dersten zevk aldıkları ve derse karşı olumlu tutum geliştirdikleri ortaya çıkmıştır (Zaharija, Mladenovic ve Boljat, 2013). 10-12 yaş arası öğrencilerle basit oyun ve animasyon yazılımlarının (HANDS, Stagecast) işlendiği bilgisayar dersinde yapılan araştırma sonucunda öğrencilerin programlamayı öğrendikleri ve öğrenirken zevk aldıkları, velilerin de programlama öğretilmesini destekledikleri belirlenmiştir (Lin, Yen, Yang ve Chen, 2005). Yaşları ve sınıfları farklı lise öğrencilerinin Bilgisayarca düşünme becerisinin robotik dersiyle ölçüldüğü araştırmada ise yaşa ve cinsiyete göre sonuçların değişmediği görülmüştür (Atmatzidou ve Demetriadis, 2016). Buna karşın lise öğrencilerinin BİT (ICT) mi yoksa Programlama mı tercih ettikleriyle ilgili yapılan araştırmada Programlama öğrencilerinin Öz-Yeterlikleri haricinde diğer ölçülen (Performans Beklentisi, Memnuniyet, Sosyal Etki, Algılanan Davranış Kontrolü ve Davranışsal Amaç) değerlerinin daha düşük olduğu görülmüştür (Giannakos, Hubwieser ve Chrisochoides, 2013). Bunun nedeni bilgisayar bilimlerinin geleceği hakkında fazla bilgiye sahip olmamaları olabilir. Nitekim, lise öğrencilerinin bilgisayar alanını seçmelerini sağlamak ve bilgisayara karşı ilgilerini artırmak amacıyla yürütülen bir çalışmada öğretmenlere, öğrencilere ve rehber öğretmenlere bilgisayar bilimi dersleri verilmiştir. Araştırma sonucunda öğretmenlerin bilgisi artarken öğrencilerin bu alana karşı hem ilgileri hem bilgileri artmıştır (Munson, Moskal, Harriger, Lauriski-Karriker ve Heersink, 2011).

Bilgisayar eğitimine erken yaşta başlanması gerektiği de yine son araştırmalarda işlenen bir diğer konudur. Anaokulunda bilgisayar kullanımının öğretilmesi hakkında yapılan bir araştırmada çocukların anaokulundan 3. Sınıfa kadar olan dönemdeki becerilerini geliştirmede anaokulunda bilgisayar bulunmasının önemli bir yordayıcı etken olduğu görülmüştür. Bilgisayar kullanma becerileri açısından erkek ve kız öğrencilerin arasında anlamlı bir fark yoktur fakat kızların bilgisayar beceri ortalaması erkeklerinkinden daha yüksek çıkmıştır (Saçkes, Trundle ve Bell, 2011). Okul öncesi öğrencilerle yapılan TengibleK Robotik Programının kullanıldığı araştırmada öğrencilerin programlamada başarılı ve ilgili olduğu ve bilgisayarca düşünme, robotik,

problem çözüme, hata ayıklama becerilerini kazandıkları görülmüştür (Bers, Flannery, Kazakoff ve Sullivan, 2014).

Programlama öğretiminin blok tabanlı görsel öğreticilerle daha kolay, anlaşılır ve akılda kalıcı olduğu söylenebilir. Blok tabanlı programlama araçlarının kullanıldığı bir araştırmada bu araçların problem çözmeye ve programlamayı öğrenmeye yardımcı olduğu ortaya çıkmıştır (Chao, 2016). Aynı şekilde 5 ve 6. Sınıf öğrencileriyle yapılan bir çalışmada öğrencilerin blok tabanlı programlama aracı olan Scratch'i hem bilgisayar dersinde hem de sosyal bilgiler ve resim derslerinde kullanmaktan memnun oldukları belirlenmiştir. Programlama kavramlarını, mantığını ve hesaplamalı bilgisayar uygulamalarını öğrenmeleri artmıştır. Öğrenciler tarafından Blok Tabanlı Programlama aracı eğlenceli, motive edici, hevesli, bağlayıcı, kullanışlı ve anlamayı kolaylaştırıcı bulunmuştur (Sáez-López, Román-González ve Vázquez-Cano, 2016). Lise öğrencileriyle yapılan bir araştırmada bir gruba geleneksel programlama (pascal) diğer gruba blok tabanlı programlama aracı (Scratch) ile programlama dersi verilmiştir. Scratch ile öğrenim gören grup diğer gruba göre daha çok motive olmuş ve çalışmalarına devam etmek istemişlerdir. Scratch ile öğrenim gören grubun %65'i çalışmalarına programlama alanında devam etmek isterken diğer grubun sadece %10'u programlamaya devam etmek istemiştir (Ouahbi, Kaddari, Darhmaoui, Elachqar ve Lahmine, 2015). Buna karşın lisede programlamaya yardımcı araçların (scratch, b#, robomind) kullanıldığı bir araştırmada bu araçları kullanmanın Programlama öğretimine hiçbir etkisinin olmadığı belirlenmiştir. Öğrenciler bu araçların aşırı kolay olduğundan ve söz diziminin (syntax) programlama dillerine uymamasından yakınmıştır. (Koorse, Cilliers ve Calitz, 2015).

Dersin işlenişinin dijital tabanlı olması gibi değerlendirmesinin de dijital tabanlı olması gerektiği görüşleri vardır. Lise uygulamalı bilişim teknolojileri dersinin değerlendirmesinin dijital portfolyo veya bilgisayar üzerinden sınavla gerçekleştirilmesi üzerine yapılan bir çalışmada; bu yöntemlerin öğrenci ve öğretmenler tarafından geleneksel yöntemlere göre değerlendirme açısından daha başarılı görüldüğü ortaya çıkmıştır (Newhouse, 2011).

Sonuç olarak, dünya geneline bakıldığında Bilgisayar öğretiminin Bilgi ve İletişim Teknolojileri eğitiminden “Bilgisayar Bilimi/Programlama” eğitimine dönüştüğü görülmektedir. Bu değişim gösteriyor ki uygun öğretmen eğitimi, bilgisayarca düşünme becerisinin K12 müfredatına temel olarak konulması ve Programlama öğretiminin verilmesi iyi bir gelecek için kesinlikle gereklidir (Hubwieser, Armoni, Giannakos ve Mittermeir, 2014).

2.2.3. Bilgisayar Eğitimi ve TPAB

Bilgisayar öğretiminde ve eğitiminde TPAB çerçevesinin kullanılmasıyla ve Bilgisayar/Bilişim öğretmenlerinin TPAB’lerinin ölçülmesi ilgili literatürde çeşitli çalışmalar mevcuttur. Araştırmalar incelendiğinde Bilgisayar Bilimi ve Bilişim Teknolojileri öğretmenlerinin Teknolojik Pedagojik Alan Bilgisi ve bileşenlerinin (TB, AB, PB, TAB, TPB, PAB) yüksek (~4,2/5) olduğu görülmüştür (Giannakos, Doukakis, Pappas, Adamopoulos, ve Giannopoulou, 2015; Doukakis vd., 2010; Albayrak Sarı, Canbazoğlu Bilici, Baran, ve Özbay, 2016). Benzer şekilde BÖTE bölümündeki öğretmen adaylarının teknoloji kullanım yeterliliklerinin TPAB düzeyine etkisi incelendiğinde diğer alanlardaki öğretmen adaylarına kıyasla anlamlı bir biçimde yüksek olduğu (Tatlı ve Akbulut, 2017), BT öğretmeni adaylarının eğitimde bilgi ve iletişim teknolojilerini kullanma konusunda kendilerini yeterli algıladıkları (Şad ve Nalçacı, 2015) ve BT öğretmeni adaylarının Web-TPAB’lerinin yüksek olduğu (Kaya, Özdemir, Emre ve Kaya, 2011) görülmüştür.

TPAB çerçevesinin bilgisayar bilimi dersinin öğretim tasarımı sürecinde kullanıldığı bir çalışmada ise TPAB çerçevesinin özellikle anlaşılması güç konularda öğrencilerin yanlış anlamalarını düzeltmede etkili olduğu görülmüştür (Ioannou ve Angeli, 2013). Aynı zamanda Programlama öğretimi için Bilişim Teknolojileri dersine PAB’nin yerleştirilmesinin önündeki problemlere çözüm aranmıştır (Saeli, Perrenet, Jochems ve Zwanveld, 2011). Ayrıca Almanya’da Bilgisayar bilimi için PAB’ nin kavramsallaştırılmasının ve tüm ülkedeki okullara yayılmasının sağlanması için çalışılmaktadır (Hubwieser, Magenheim, Mühlning ve Ruf, 2013).

2.3. TPAB Çerçevesinin K12 Eğitiminde Kullanılması

TPAB'in öğretmen bilgisi üzerine kurulmuş bir çerçeve olmasından dolayı TPAB ile ilgili araştırmaların çoğunluğu (Lee ve Tsai, 2010; Niess, 2005; Doering, Veletsianos, Scharber ve Miller, 2009; Baran, Chuang ve Thompson, 2011; Archambault, Wetzel, Faulger ve Williams, 2010) öğretmen eğitimi üzerinedir. Ancak araştırmacılar tarafından TPAB çerçevesinin K12 eğitiminde öğrenci ve öğretmenler için kullanılabileceği vurgulanmıştır (Angeli ve Valanides, 2009; Harris ve Hofer, 2009).

Öğretmenlerin Teknolojik Pedagojik Alan Bilgisi düzeyinin öğrenciler üzerine etkisini inceleyen bir araştırmada TPAB'i yüksek olan öğretmen adaylarının K12 öğrencilerinin üst düzey düşünme becerilerini (Eleştirel düşünme, yaratıcılık ve işbirlikçi iletişim) geliştirmelerini sağlayacak seviyede karmaşık aktiviteleri oluşturmada daha başarılı oldukları ortaya çıkmıştır (Brown, Neal ve Fine, 2011).

Bir diğer araştırmada ise TPAB çerçevesiyle oluşturulan ve robotik platformunun kullanıldığı ders planlarıyla ortaokul Fen Bilimleri ve Matematik derslerindeki enerji, adaptasyon ve aritmetik dizi konuları için öğretim programı oluşturulmuştur. Sonuçta TPAB çerçevesinin ders planı tasarlanmasında kullanılması, anlaşılması güç soyut STEM konularının öğretilmesi ve öğrenilmesinde başarılı olmuştur (Brill, Listman ve Kapila, 2015). Benzer şekilde 6.sınıflarla yapılan bir çalışmada TPAB çerçevesinde yürütülen ders planının öğrenmeyi artırdığı, öğrencilerin ders içeriğine uygun teknolojik araçları kullanmayı öğrenmelerini sağladığı ve öğrencilerin aktif öğrenci olmaları için gerekli içerik ve teknoloji bilgisine sahip olmalarını sağladığı kanıtlanmıştır (Wetzel ve Marshall, 2011). Başka bir araştırmada da K12 eğitiminde coğrafya öğretiminde kullanılan TPAB çerçevesinin öğretmenlerin teknoloji entegrasyon pratiğini geliştirdiği gibi öğrencilerin de derse katılımını teşvik ettiği ve coğrafya kavramlarını anlamalarını geliştirdiği kanıtlanmıştır (Doering, Koseoglu, Scharber, Henrickson ve Lanegran, 2014). Benzer şekilde 3.sınıf öğrencilerine TPAB modeliyle bilişim tabanlı olarak verilen eğitim neticesinde öğrencilerin çoğunluğu (%92) akademik açıdan başarılı olmuştur ve öğrencilerin eleştirel düşünme ve bilgisayar okuryazarlığı becerileri artmıştır (Aisyah, 2013). Öğretim planında Harris ve Hofer (2009)'in geliştirdiği TPAB öğrenme

aktivitelerinin kullanılması öğretmenlerin, teknolojik olarak entegre edilmiş planlamalarıyla ilgili daha fazla bilgi sahibi olmalarına ve kendilerine güvenmelerini sağlamaya yardımcı olmuştur (Hofer ve Harris, 2010). Başka bir araştırmada aynı TPAB tabanlı öğrenme aktiviteleriyle çalışan öğretmenlerin önceki ve sonraki oluşturdukları ders materyalleri karşılaştırılmış ve sonra oluşturdukları materyallerin daha bilinçli, stratejik ve çeşitli olduğu planlamalarının daha öğrenci-merkezli olduğu ve Eğitimde Teknoloji Entegrasyonunun kalitesinin yükseldiği gözlenmiştir (Harris ve Hofer, 2014).

Ortaokul özel eğitim öğretmenleriyle gerçekleştirilen, derslerde eğitim teknolojilerinin kullanılmasının TPAB'e etkisinin araştırıldığı bir çalışmada ise öğretmenlerin TPAB'inin arttığı ve öğrencilerin de anlama kabiliyetinin geliştiği ortaya çıkmıştır (Ciampa, 2017). Buna karşın Teknolojik araç kullanımının öğretmen ve öğrencilere etkisini araştıran bir çalışmada ise öğretmenlerin pedagojik yaklaşımlarının çok az değişim gösterdiği, öğrencilerinse öğrenme stillerini değiştirmeye yeteri kadar güvenleri olmadığı ortaya çıkmıştır. Öğretmenlerin TPAB'inin değişime direnç gösterdiği öğrencilerinse TPAB'inin yeterince gelişmediği görülmüştür (Kontkanen vd., 2017).

Lise matematik öğretmenleriyle yapılan bir araştırmada öğretmenlerin ürettikleri yazılı ders materyallerinden aldıkları TPAB puanlarıyla pratik öğretimleri paralellik göstermiştir. Yüksek TPAB'e sahip öğretmenlerin öğrencilerinin akademik başarıları yüksektir (Lyublinskaya ve Tournaki, 2011). Buna karşın bir diğer araştırmada K12 öğretmenlerinin TPAB'i ve öğrenci başarıları arasında herhangi bir anlamlı ilişki bulunamamıştır (Farrell ve Hamed, 2017).

3. YÖNTEM

Bu bölümde araştırmanın modeli, çalışma grubu, veri toplama araçları, araştırmada izlenen yol ve toplanan verilerin çözümlenmesinde kullanılan tekniklere yer verilmiştir.

3.1 Araştırmanın Modeli

Bilişim Teknolojileri ve Yazılım dersinin öğretiminde TPAB destekli öğretimin farklı değişkenler açısından etkisini inceleyen bu araştırma, yarı deneysel desen çerçevesinde karma yöntemle yürütülmüştür. Karma yöntem nicel ve nitel araştırma verilerini tek bir çalışma altında toplamaktır (Andrew ve Halcomb, 2009). Karma yöntemler sadece nicel ya da sadece nitel yöntemlerin kullanıldığı araştırmalara göre daha güvenilir sonuçlar sağlar (Amaratunga, Baldry, Sarshar ve Newton, 2002). Bu araştırmada karma araştırma türlerinden Sıralı-Açıklayıcı tasarım kullanılmıştır. Sıralı-Açıklayıcı tasarımda önce nicel, sonra nitel veri toplanır ve baskın olan nicel veridir (Creswell, 2003).

Araştırmanın nicel bölümü yarı deneysel desenle yürütülmüştür. Deneysel desenli araştırmalar, neden-sonuç ilişkilerini belirlemeye çalışmak amacı ile doğrudan araştırmacının kontrolü altında, gözlenmek istenen verilerin üretildiği araştırma modelleridir (Karasar, 2015). Deneysel çalışmaları diğer araştırma biçimlerinden ayıran şey, araştırmacının katılımcıları kontrol grubu ve deney grubu olarak ayırıp, değişkenlerden nasıl etkilendiğini ölçmesidir. Teknik açıdan, araştırmacı bir veya daha fazla bağımsız değişkeni ekleyerek ya da çıkartarak bağımlı değişkenin ya da çalışmanın sonucu üzerindeki etkisini incelemektedir. Bağımsız değişken, kontrol ve deney grubu katılımcılarının nasıl farklılaştığını ifade eden değişkendir. Eğitim alanında, bağımsız değişken, müfredat materyalleri, öğretim stilleri veya özel eğitim (ör., Eğitim alma veya almama) olabilir (Lodico, Spaulding ve Voegtle, 2010).

Bu çalışmada üç tane “bağımlı değişken” vardır. Bağımlı değişkenler “Programlama öğrenme başarısı”, “Bilgisayarca Düşünme Becerisi” ve “Problem Çözme Becerisi Algısı” dir. Araştırmadaki “bağımsız değişken” ise uygulanan “öğretim yöntemi” dir. Bağımsız değişkenin iki düzeyi vardır. Bunlar; “TPAB çerçevesiyle öğretim yöntemi” ile “geleneksel öğretim yöntemidir”. Araştırmada

deneysel modellerden “Öntest – Sontest Kontrol Gruplu Yarı Deneysel Desen” kullanılmıştır. Yarı deneysel desen deneysel desene benzer fakat gruplar burada gelişigüzel oluşturulur (Karasar, 2015). Bu çalışmada da deneysel yerine yarı deneysel desen kullanılmasının nedeni deney ve kontrol gruplarının önceden belirlenmiş iki ayrı sınıf oluşu ve sonradan müdahale edilemeyeşidir. Araştırma modelinde yer alan simgeler ve anlamları tabloda verilmiştir.

Tablo-1: Araştırma Modeli Grupları

Grup	Ön Test	Bağımsız Değişken	Son Test
Deney Grubu (GD)	T1, T2, T3	TPAB Çerçevesiyle Öğretim	T4, T5, T6
Kontrol Grubu (GK)	T7, T8, T9	Geleneksel Öğretim	T10, T11, T12

GD = Deney Grubu (TPAB Tabanlı öğretim uygulanan grup)

GK = Kontrol Grubu (Geleneksel öğretim uygulanan grup)

T1 = Deney Grubu, Akademik Başarı Ön-Testi

T2 = Deney Grubu, Problem Çözme Envanteri Ön-Testi

T3 = Deney Grubu, Bilgisayarca Düşünme Becerisi Ön-Testi

T4 = Deney Grubu, Akademik Başarı Son-Testi

T5 = Deney Grubu, Problem Çözme Envanteri Son-Testi

T6 = Deney Grubu, Bilgisayarca Düşünme Becerisi Son-Testi

T7 = Kontrol Grubu, Akademik Başarı Ön-Testi

T8 = Kontrol Grubu, Problem Çözme Envanteri Ön-Testi

T9 = Kontrol Grubu, Bilgisayarca Düşünme Becerisi Ön-Testi

T10 = Kontrol Grubu, Akademik Başarı Son-Testi

T11 = Kontrol Grubu, Problem Çözme Envanteri Son-Testi

T12 = Kontrol Grubu, Bilgisayarca Düşünme Becerisi Son-Testi

3.2 Çalışma Grubu

Araştırma grubunu, Konya ili Kulu ilçesi Kozanlı 75. Yıl Birol Polat ve Ahmet Yesevi İmam Hatip Ortaokulu'nda 2017-2018 eğitim-öğretim yılında 6.sınıflarda 2 şubede okuyan 41 öğrenci oluşturmaktadır. Yansız olarak sınıflardan birisi kontrol diğeri deney grubu olarak seçilmiştir. Deney ve kontrol gruplarında aynı öğretmen ile öğretim yapılmıştır.

3.3 Veri Toplama Araçları

Bu araştırmada belirlenen amaca ulaşabilmek için hem nicel hem de nitel veri toplama araçlarından yararlanılmıştır. Programlama Akademik Başarı Testi, Problem Çözme Envanteri ve Bilgisayarca Düşünme Beceri Ölçeği nicel veri toplama aracı olarak kullanılmıştır. Nitel veri toplama aracı olarak ise görüşme soruları kullanılmıştır.

3.3.1 Programlama Başarı Testinin Geliştirilmesi

Araştırmada kullanılacak ön-test ve son-test için başlangıçta araştırmacı tarafından 40 soruluk Problem Çözme ve Programlama ünitesi ile ilgili bir test geliştirilmiştir. Bu test kapsam ve görünüş geçerliliği için alanlarında uzmanlaşmış öğretim görevlilerinden uzman görüşleri alınmıştır. Uzman görüşleri sonucunda test 25 soruya düşürülmüştür.

Başarı testinde toplam 14 hedef davranış vardır (TTKB, 2018):

BT.6.5.2.1. Blok tabanlı programlama aracının arayüzünü ve özelliklerini tanır.

Scratch (scratch.eba.gov.tr), Kodadı2023, Code.org, Codecademy vb. programlama platformları kullanılabilir.

BT.6.5.2.2. Blok tabanlı programlama aracında sunulan bir programın işlevlerini açıklar.

BT.6.5.2.3. Blok tabanlı programlama aracında sunulan bir programın hatalarını ayıklar.

BT.6.5.2.4. Blok tabanlı programlama aracında sunulan bir programı verilen ölçütlere göre geliştirerek düzenler.

BT.6.5.2.5. Doğrusal mantık yapısını içeren programlar oluşturur.

BT.6.5.2.6. Doğrusal mantık yapısını içeren programları test ederek hatalarını ayıklar.

BT.6.5.2.7. Karar yapısını içeren programlar oluşturur.

BT.6.5.2.8. Karar yapısını içeren programları test ederek hatalarını ayıklar.

BT.6.5.2.9. Çoklu karar yapıları içeren programlar oluşturur.

BT.6.5.2.10. Çoklu karar yapısını içeren programları test ederek hatalarını ayıklar.

BT.6.5.2.11. Döngü yapısını içeren programlar oluşturur.

BT.6.5.2.12. Döngü yapısını içeren programları test ederek hatalarını ayıklar.

BT.6.5.2.13. Bir algoritmayı uyarlamak için en uygun karar yapılarını seçer.

BT.6.5.2.14. Farklı programlama yapılarını kullanarak karmaşık problemlere çözüm üretir.

25 soruluk başarı testindeki maddelerin belirtke tablosu Tablo-2' de verilmiştir.

Tablo-2: Programlama Başarı Testi Belirtke Tablosu

Bilişsel Alan- Kazanımlar	Bilgi	Kavrama	Uygulama	Analiz	Sentez	Toplam
Blok Tabanlı Programlama (BT.6.5.2.1-2-3-4)	1, 16, 17, 21, 23, 24		22	-	-	7
Doğrusal Mantık Yapısı (BT.6.5.2.5-6)	25	-	20	2	15	4
Karar Yapısı (BT.6.5.2.7-8)	19	5	10		14	4

Çoklu Karar Yapısı (BT.6.5.2.9-10-13)	-	7	-	-	12	2
Döngü Yapısı (BT.6.5.2.11-12)	18	6	3	4, 13		5
Karmaşık Problemler (BT.6.5.2.14)	-	-	-	8, 11	9	3
Toplam	8	4	4	5	4	25

Uzman görüşleri ile 25 soruya düşürülen test Kulu Kozanlı 75. Yıl Birol Polat Ortaokulu ve Ahmet Yesevi İmam Hatip Ortaokulu 6. sınıf öğrencilerine, “Bilişim Teknolojileri ve Yazılım” dersi programında yer alan “Problem ve Çözme ve Programlama” ünitesindeki “Programlama” konusu üzerindeki öğrenmelerini ölçmek amacıyla uygulanmıştır. Akademik başarı testinin güvenilirliğini belirlemek amacıyla SPSS (Statistical Package Program for Social Studies) programıyla güvenilirlik analizi uygulanmış, analiz sonucunda testin Cronbach’s Alpha katsayısı 0.93 olarak bulunmuştur.

Bir testteki öğelerin kalitesini bulmak için, onları deneysel olarak denemek ve analiz etmek gerekir. Bu analizi yapmak için iki temel gösterge, maddenin zorluk seviyesi (güçlüğü) ve ayırmacılık gücüdür (Escudero, Reyna ve Morales, 2000). Maddelerin güçlük ve ayırt edicilik kat sayıları Tablo-3’te verilmiştir.

Tablo-3: Programlama Başarı Testinin Madde Analiz Dereceleri

Madde No	Madde güçlüğü (p)	Madde ayırt ediciliği (d)
1	0,79	0,64
2	0,88	0,44
3	0,48	0,7
4	0,36	0,67
5	0,42	0,34
6	0,79	0,51
7	0,73	0,38
8	0,36	0,78
9	0,33	0,77
10	0,82	0,67
11	0,52	0,8
12	0,45	0,78

13	0,82	0,36
14	0,88	0,57
15	0,82	0,61
16	0,85	0,53
17	0,58	0,35
18	0,36	0,5
19	0,3	0,43
20	0,79	0,33
21	0,79	0,7
22	0,82	0,65
23	0,73	0,54
24	0,67	0,79
25	0,55	0,54

Bütün bunların sonucunda geliştirilen test; 8 bilgi, 4 kavrama, 4 uygulama, 5 analiz ve 4 sentez düzeyinde olmak üzere toplam 25 tane çoktan seçmeli soru içermektedir. Bu başarı testinin kullanılma amacı, öğrencilerin “Programlama, Doğrusal Mantık Yapısı, Karar Yapısı, Çoklu Karar Yapısı, Döngü Yapısı, Karmaşık Problem Yapısı” konularıyla ilgili bilgilerini ön-test ve son-test şeklinde kontrol ederek, uygulanan yöntemlerden kaynaklanabilecek gruplar arası bilişsel düzeydeki farklılıkları ortaya koymaktır.

3.3.2 Bilgisayarca Düşünme Beceri Düzeyleri (BDBD) Ölçeği

Öğrencilerin bilgisayarca düşünme becerilerini ölçmek için Korkmaz, Çakır ve Özden (2015) tarafından geliştirilen “Ortaokul Düzeyi İçin Bilgisayarca Düşünme Beceri Düzeyleri Ölçeği” kullanılmıştır. Ölçek beş dereceli Likert tipinde geliştirilmiş 22 maddeden oluşmaktadır. Ölçek “Yaratıcılık”, “Algoritmik Düşünme”, “İşbirliklilik”, “Eleştirel Düşünme” ve “Problem Çözme” olmak üzere 5 ayrı faktörden oluşmaktadır. Ölçeğin orijinali Korkmaz, Çakır ve Özden (2017) tarafından üniversite öğrencileri için geliştirilen Computational Thinking Scales (CTS) olup ortaokul düzeyindeki öğrenciler için yeniden uyarlanmıştır. Ölçeğin iç tutarlılık katsayısı 0,81 olarak belirlenmiştir.

3.3.3 İlköğretim Düzeyindeki Çocuklar İçin Problem Çözme Envanteri (ÇPÇE)

Öğrencilerin problem çözme becerisi algılarını ölçmek için Serin, Bulut Serin ve Saygılı (2010) tarafından geliştirilen “İlköğretim Düzeyindeki Çocuklar İçin Problem Çözme Envanteri (ÇPÇE)” kullanılmıştır. Ölçek beş dereceli Likert tipinde geliştirilmiş 24 maddeden oluşmaktadır. Ölçek “Problem Çözme Becerisine Güven”, “Öz Denetim” ve “Kaçınma” olmak üzere 3 ayrı faktörden oluşmaktadır. Ölçeğin iç tutarlılık katsayısı 0,80 olarak belirlenmiştir. Ölçek bu alanda ülkemizde geliştirilen özgün ve ilk envanter olma özelliğini taşımaktadır.

3.3.4 Görüşme Soruları

Öğrencilerin TPAB çerçevesi ile oluşturulmuş Programlama dersiyle ilgili görüşlerini almak için araştırmacılar tarafından geliştirilen ve uzman görüşü alınmış yarı yapılandırılmış form kullanılmıştır. Sorular Bilişim Teknolojileri ve Yazılım dersi Programlama konusuna göre uyarlanmıştır. Ölçek Bilişim Teknolojileri ve Yazılım dersi Programlama konusunun TPAB çerçevesiyle oluşturulması ile ilgili öğrenci görüşleri hakkında 5 sorudan oluşmaktadır.

3.3.5 Ders Planının Hazırlanması

Araştırmanın uygulamasına başlamadan önce öğrencilere Bilişim Teknolojileri ve Yazılım dersi Programlama ünitesi için 12 saatlik (6 hafta) bir öğretim programı hazırlanmıştır. Deney grubu için TPAB destekli öğretim yöntem ve teknikleri kullanılırken, kontrol grubu için geleneksel yöntem ve teknikler kullanılmıştır. Ders planları her 2 grup için de Gagne'nin Öğretim Durumları Modeli düzeninde oluşturulmuştur. Gagne'nin öğretim Durumları Modeli 9 ana aşamadan oluşur. Bunlar: dikkat çekme, hedeften haberdar etme, ön öğrenmeleri hatırlatma, uyarıcı materyal kullanma, öğrenmede rehber olma, performansı ortaya çıkarma, dönüt sağlama, değerlendirme ve kalıcılığı ve transferi sağlamadır (Karaağaçlı ve Erden, 2008).

Hedef kitle, yaşları 11-13 arasında değişen 6. Sınıf öğrencileridir. Piaget'in Bilişsel Gelişim Kuramına göre 11 -15 yaş arasında oldukları için Formal işlevsel evre (soyut işlemler)'ye dahil olmaktadır. Bu aşamada, çocuğun bilişsel yapıları en üst

düzyeyde gelişim seviyesine ulaşır ve çocuğun doğrudan yaşantısından veya soyut göndermeden soyut sorunlar da dahil olmak üzere çocuğun bütün sınıf problemlerine mantıksal mantık uygulayabilmesi mümkün olur (Piaget, 1963).

Aynı zamanda bu öğrenciler yaşları itibariyle Erken Ergenlik Öğrencileri (11-14 Yaş) grubuna girerler. Ortaokul ve lise yıllarında, öğrenciler hayatın diğer zamanlarında yaşananlardan çok daha dramatik gelişimsel değişiklikler geçirir ve çok farklı zaman çizelgeleri üzerinde çalışırlar. Erken dönem ergen, çeşitli deneyimlerle uğraşmayı öğrenmelidir:

-Değişen ve genellikle önceden tahmin edilemez bir vücutta ortaya çıkan cinsellik;

-Bir süredir bilişsel bir düzlüğe ulaşma ve daha sonra yeni, yetişkin entelektüel araçları bulma;

-Çıkarların çoğalması ve hızla değişmesi;

-Akışkan ve esnek bir benlik konsepti;

-Yetişkinlerle kişilerarası ilişkilerin yeniden işe alınması ihtiyacı;

-Çalkantılı duygular;

-Aşırı idealizm; bağımsızlık taahhüdünde bulunma ihtiyacı ve

-Güçlü bir akran grubu.

Bu yaştaki çocuklar için tüm okullaşmanın temel amacı, olumlu ilişkilerin teşviki ve kendi kendine imajın pozitif olmasıdır. Ortaokul öğrencileri, akademik disiplinlerin taleplerine bir girişin yanı sıra kapsamlı keşif imkânlarına da ihtiyaç duyar (Schurr, 1989).

Öğretim programı hazırlanırken Teknolojik Pedagojik Alan Bilgisi ile ilgili çalışmalardan yararlanılmıştır. Harris ve Hofer (2009)'a göre TPAB destekli geliştirilen öğretim programı hazırlanırken yapılması gerekenler;

-Öğrenme hedeflerini seçme

-Öğrenme deneyiminin doğasıyla ilgili pratik pedagojik kararlar alma

-Öğrenme deneyimini oluşturmak için birleştirilecek uygun etkinlik türlerini seçme ve sıralama

-Öğrencilerin nerde ve ne kadar iyi öğrendiklerini açığa çıkartacak biçimlendirme ve özet değerlendirme stratejilerinin seçilmesi

-Öğrencilerin planlanan öğrenme deneyiminden yararlanmalarına en iyi yardımcı olacak araçları ve kaynakları seçmek

Araştırmalara göre (Giannakos vd., 2015; Doukakis vd., 2010; Albayrak Sarı vd., 2016) BT öğretmenlerinin TPAB' inin yüksek olması nedeniyle TPAB çerçevesine uygun ders planının hazırlanması için gerekli olan bilginin BT öğretmenlerinde mevcut olduğu söylenebilir.

6 haftalık TPAB tabanlı gerçekleştirilen Programlama ders planında öğrencilere kazandırılması planlanan beceriler ve ait olduğu bilgi türü (Teknolojik Bilgi, Pedagojik Bilgi, Alan Bilgisi, Pedagojik Alan Bilgisi, Teknolojik Alan Bilgisi, Teknolojik Pedagoji Bilgisi, Teknolojik Pedagojik Alan Bilgisi) aşağıdaki tabloda verilmiştir:

Tablo-4: Öğrenci Becerileri ve Bilgi Türleri

Bilgi	Öğrenci Becerileri	Açıklama
TB	Scratch programını kullanır. Blockly oyunları kullanır. Code.org programlama ortamını kullanır. Touch Develop programlama ortamını kullanır. Microsoft Small Basic programını kullanır.	Öğrenci Teknoloji Bilgisi, Teknoloji araçlarının, programlarının ve uygulamalarının başarılı bir şekilde kullanımını öğrencilerin çeşitli formatlardaki dosyalarla (örn. PowerPoint, Prezi, multi-medya sunumları, videolar, vb.) çalışabilmesine olanak tanır (Pilgrim ve Martinez, 2015).
PB	Etkinliklerde işbirlikli çalışır. Etkinliklerde yapacaklarını organize eder.	Öğrenci Pedagojik Bilgisi, öğretim ihtiyaç ve kararlarını içeren öğrenme ve araştırma sürecine seçilen ve uygulanan yöntemleri ifade eder. Öğrencilerin öğretim teknolojisini kullanmada başarılı olma yetenekleri, öğretmenin destekli araştırma becerilerine ve öğrencilerin kendi öğrenme süreçlerini anlamalarına ilişkin kendi üstbilişlerine dayanır (Pilgrim ve Martinez, 2015).
AB	Bloklarla ve kodlarla çalışır. Blokları ve kodları önceden öğrendikleri algoritma konusuyla ilişkilendirir. Blokları yazılı kodlarla ilişkilendirir.	Öğrenci Alan Bilgisi, öğrencilerin ön öğrenmelerinden yani şemalarından etkilenir. Bu bilgi, kavramlar, teoriler ve örgütsel çerçeveler geliştirmek için bir temel sağlar. İçerik bilgisi, öğrencilerin en önemli / alakalı içeriği bulmak için bilgi sentezleme yeteneğini içerir (Pilgrim ve Martinez, 2015).

PAB	Öğrenciler kendi istekleri doğrultusunda ya etkinliklerdeki yönergeleri okur ya öğretmenin elindeki öğretim materyallerine tekrar göz atar ya da kendisi araştırarak ne yapması gerektiğini bulur.	Öğrenci Pedagojik İçerik Bilgisi, öğrenme içeriğinin dönüşümüne dair bir kavramdır. Çevrimiçi kaynaklar basılı metnin elektronik formatları olabilir, ancak etkileşimli olabilirler; Böylece öğrencinin öğrenim ihtiyaçları ve ilgi alanı komutu olarak içeriğe (gömülü bağlantılar, videolar, işitsel bileşenler, vb.) katılmasına izin verir. Öğretmenin öğrencinin ne öğrendiğini ve öğrencinin nasıl öğrendiğini ve bağlantı kurduğunu değerlendirmesini sağlar (Pilgrim ve Martinez, 2015).
TAB	Öğrenciler öğretmenin verdiği ödevleri yapmak için en uygun programlama ortamını seçerler.	Öğrenci Teknolojik İçerik Bilgisi, öğrencinin perspektifinden konuyla ilgili bilgileri elde etmek ve iletmek için belirli teknolojilerden (çevrimiçi formatlar, programlar, uygulamalar, vb.) uygun olanını seçmektir. Bu öğrencilerin herhangi bir amaç için yanlış teknolojiyi kullanıp kullanmadığını da anlamlarını sağlar (Pilgrim ve Martinez, 2015).
TPB	Öğrenciler programlama için kullandıkları ortamın elverdiği şekilde bireysel ya da işbirlikli çalışırlar.	Öğrenci Teknolojik Pedagoji Bilgisi, öğrencinin araştırma araçlarını, konuları ve web sitesi seçimine göre, öğretme ve öğrenmenin nasıl değişebileceğinin anlaşılmasıdır. Ayrıca, öğrenilen içeriğin öğrenci tarafından yaygınlaştırılması, sınıfın genel öğrenmesine katkıda bulunmalıdır (Pilgrim ve Martinez, 2015).
TPAB	6 haftanın sonunda öğrencilerden istedikleri programlama ortamını kullanarak istedikleri konuyla ilgili (Karar yapısı, Döngü, Karmaşık) bireysel ya da grup çalışması yapmaları istenir.	Öğrenci Teknolojik Pedagojik Alan Bilgisi, teknolojik temelli materyalleri öğrenme kaynaklarını oluşturan içerik, format, amaç ve pedagojik değerlendirmelerin anlaşılmasını gerektiren teknoloji ile etkin öğrenmenin temelidir. Öğretmen eğitimi kolaylaştırdıkça öğrenciler öğrenme deneyimlerinde aktif katılımcı olarak yer alırlar (Pilgrim ve Martinez, 2015).

Ders planında öğrencilere kazandırılması planlanan Bilgisayar Dersi Standartları (CSTA, 2016):

1B-A-5-4 Bir problemi çözmek ya da bir eylem yaratmak için, sıralama, olaylar, döngüler, koşullu, paralellik ve değişkenleri içeren, blok tabanlı bir görsel programlama dili ya da metin tabanlı bir dil kullanarak hem bireysel hem de iş birliği içinde çalışır.

1B-A-3-6 Bireysel ya da işbirlikli olarak büyük bir problemi daha küçük alt problemlere böler.

1B-A-3-7 Bir bilgisayar aygıtıyla veya aygıtsız olarak hem bireysel hem de işbirlikli çalışarak sıralama, döngü ve koşullandırma içeren bir algoritma oluşturur ve çalıştırır.

1B-A-6-8 Diziler, olaylar, döngüler, koşullar, paralellik ve değişkenleri içeren bir algoritmayı analiz eder ve hatalarını ayıklar (düzeltir).

2-A-7-4 Algoritma akışını yorumlar ve sonuçlarını tahmin eder.

2-A-5-6 Hem işbirlikçi hem de bireysel olarak iç içe döngü içeren programlar geliştirir.

Aynı zamanda ders planında Harris ve Hofer'in (2009) TPAB Öğrenme aktivitelerinden esinlenerek yazılan "Bilgisayar Bilimi için TPAB Öğrenme Aktiviteleri"nden (Carton, 2017) yararlanılmıştır.

Deney Grubu ile yapılan TPAB öğrenme aktiviteleri:

Tablo-5: Ders Planında Kullanılan TPAB Öğrenme Aktiviteleri

Aktivite Adı	Tanımı	Etkinlik	Adı	Açıklama
		/Kullanılan Platform		
Akran Dönütü	Öğrenciler programlama ve projeleri hakkında dönüt alır ve verirler (Carton, 2017).	İlk Yazıyorum/ Scratch	Kodumu	Öğrenciler Scratch'a giriş yaparlar. Yaptıkları etkinliği istedikleri şekilde değiştirip proje olarak kaydeder ve birbirlerine Scratch üzerinden yorum yazarlar. (TPB)
Problemleri Tanımlama	Öğrenciler, bilgisayar ortamında çözülebilen gerçek dünya problemlerini tespit eder (Carton, 2017).	Hem Hem	Kodluyorum/ Oynuyorum/ Blockly Oyunlar	Öğrenciler hayvan sınıflandırmasının bilgisayarda yapılabildiğine dair bir fikir kazanır. (TAB)

Hata Ayıklama/Sorun Giderme	Öğrenciler bilgisayar sistemlerinde sorun giderir ve kod dizinlerindeki hataları sistematik olarak ayıklar (Carton, 2017).	Kuşun Domuzu Yakalamasına Yardım Ediyorum ve Arının İşini Kolaylaştırıyorum/ Code.org	Öğrenciler etkinlikte öğretmenlerinin yaptığı hatayı bulur ve düzeltir. (TAB)
Artifaktları Etme	Test Öğrenciler, kriterler ve kısıtlamaların karşılanıp karşılanmadığını belirlemek için hesaplamalı eserleri sistematik olarak test eder (Carton, 2017).	Yönümü Buluyorum/ Blockly Oyunlar Kediye Yönlendiriyorum/ Scratch	Öğrenciler yazmaları gereken kodu kendileri tahmin eder, test eder ve düzeltirler. (TPAB)
Karmaşık Basitleştirme	Kodu Öğrenciler, sonuçta herhangi bir değişikliğe yol açmadan kodun bir bölümünü daha az kod içerecek şekilde değiştirirler (Carton, 2017).	Kuşun Domuzu Yakalamasına Yardım Ediyorum 2/ Code.org	Öğrenciler önce döngüsüz yazdıkları uzun kodu döngülü olarak kısa biçime çevirir. (TAB)
Problemleri Ayırıştır	Öğrenciler problemleri daha yönetilebilir alt problemlere ayırır (Carton, 2017).	Döngüyle Buluyorum/ Oyunlar	Öğrenciler gereken kodu yazmak için önce bir döngüde hangi kodların tekrarladığını belirler sonra kaç döngüye girdiğini tespit eder. (TPAB)
Problemleri Değerlendir	Öğrenciler, hesaplanabilir şekilde çözülebileceklerini belirlemek için problemleri değerlendirir (Carton, 2017).	Algoritmayı Hatırlıyorum, Algoritmamı Bloklarına Çeviriyorum	Öğrenciler algoritması verilen problemin kodlara çevrilip çevrilmeyeceğini değerlendirir. (PAB)

Problemleri Tartış	Öğrenciler bir problemin hesaplamalı bir yaklaşımla çözülme kabiliyeti hakkında açıklayıcı sorular sorup tartışırlar (Carton, 2017).	Algoritmanı Kod Bloklarına Çeviriyorum/ Code.org	Öğrenciler algoritması verilen kodlar üzerine grup olarak tartışır. (PAB)
Artifakt Geliştir	Öğrenciler problemleri çözmek, kendilerini ifade etmek veya görevleri tamamlamak için hesaplamalı eserler oluştururlar (Carton, 2017).	Kaplumbağa ile Kare Çiziyorum/ Develop Kodla Çizim Yapıyorum/ Code.org	Öğrenciler kodlarla bir şekil oluştururlar. (TPAB)
Modüller Tasarla/ Oluştur	Öğrenciler etkileşim ve soyutlama sistemleri tasarlar ve oluşturur (Carton, 2017).	Bilgisayarım Benimle Konuşuyor/ Microsoft Small Basic	Öğrenciler bilgisayarın ekranına duruma göre değişen yazı yazdıran kod oluştururlar. Bütün öğrendiklerini kapsayan bir proje oluştururlar. (TPAB)

Araştırmaya katılan deney ve kontrol grubu öğrencileri bilgisayar kullanımı konusunda yeterli bilgi ve beceriye sahip olduklarından dolayı, bilgisayar kullanımı konusunda ön çalışmaya gerek duyulmamıştır. Uygulama 6 hafta/12 saatlik bir dönemi kapsamaktadır. Uygulama bilgisayar, hoparlör, etkileşimli tahta ve internet bağlantısının bulunduğu Bilgisayar Laboratuvarında gerçekleştirilmiştir.

3.4 Veri Analizi

Veri analizi, verileri toplama, düzenleme ve istatistiksel işlemler uygulayarak anlamlı kararlar verebilme ve geçerli sonuçlar çıkarabilme süreci olarak tanımlanabilir (Büyüköztürk, Kılıç Çakmak, Akgün, Karadeniz ve Demirel, 2012). Bir araştırmada, bağımlı değişken üzerinde etkisi gözlenecek değişken sayısı tek ise, tek faktörlü analizler (örneğin, t-testi, tek faktörlü ANOVA, basit regresyon analizi vb.) kullanılır. Bağımlı değişken üzerinde iki ya da daha çok değişkenin etkisinin, eş zamanlı test

edilmesi söz konusu ise, çok faktörlü analizlerin (örneğin iki faktörlü ANOVA, çoklu regresyon analizi vb.) kullanılması gerekir. (Büyüköztürk vd., 2012).

Birbirinden bağımsız ve ortalamaları eşit olan 2 grup arasındaki farklılaşmayı incelemek için “Bağımsız örneklem için t-testi” kullanılır (Ross, 2014). Nicel ölçme araçları uygulama okulunda bulunan 6. sınıfın tüm şubelerine uygulanmıştır. Başarı durumları birbirinin hemen hemen hepsinin aynı çıkmasından ötürü kura çekimi ile A şubesi deney grubu, B şubesi ise kontrol grubu olarak seçilmiştir. Deney ve kontrol grubu öğrencilerinin akademik başarı, problem çözme envanteri ve bilgisayarca düşünme becerisi ölçekleri ön-test, son-test puanlarının t testi ile analizi yapılmadan önce parametrik testler için gerekli olan normallik varsayımı test edilmiştir.

Normallik dağılımını ölçmek için Shapiro-Wilk, Kolmogrov-Smirnov gibi testler mevcuttur. Ancak bu testlerden kimisi küçük gruplar kimi büyük gruplar için idealdir ve aynı veri üzerinden farklı sonuçlar vermektedirler. Bu nedenle normallik dağılımını ölçmek için büyük ya da küçük grupların her ikisiyle de çalışmaya uygun olan ve daha doğru sonuçlar veren çarpıklık (Skewness) ve basıklık (Kurtosis) katsayılarına bakmak gerekir (Kim, 2013). Deney grubunun akademik başarı testi ön-testi için çarpıklık (Skewness) ve basıklık (Kurtosis) katsayısı sırasıyla 0,375 ve -0,966; kontrol grubu için sırasıyla 0,453 ve -0,224’tür. Deney grubunun problem çözme envanteri ön-testi için çarpıklık (Skewness) ve basıklık (Kurtosis) katsayısı sırasıyla 0,257 ve -0,914; kontrol grubu için sırasıyla 0,768 ve -0,597’dir. Deney grubunun bilgisayarca düşünme beceri ölçeği ön-testi için çarpıklık (Skewness) ve basıklık (Kurtosis) katsayısı sırasıyla 0,837 ve 0,719; kontrol grubu için sırasıyla 0,465 ve -0,802’dir. Bu katsayının 0 dolaylarında olması puanların normal dağıldığını göstermektedir (Brown, 1996). Buna göre ön-test puanlarının normal bir dağılım gösterdiği söylenebilir.

Deney grubunun akademik başarı testi son-testi için çarpıklık (Skewness) ve basıklık (Kurtosis) katsayısı sırasıyla -0,499 ve -0,983; kontrol grubu için sırasıyla 0,062 ve -0,848’dir. Deney grubunun problem çözme envanteri ön-testi için çarpıklık (Skewness) ve basıklık (Kurtosis) katsayısı sırasıyla 0,826 ve 0,378; kontrol grubu için sırasıyla 0,980 ve 0,754’tür. Deney grubunun bilgisayarca düşünme beceri ölçeği ön-

testi için çarpıklık (Skewness) ve basıklık (Kurtosis) katsayısı sırasıyla 0,001 ve -0,907; kontrol grubu için sırasıyla 0,279 ve -0,849'dur. Buna göre son-test puanlarının normal bir dağılım gösterdiği söylenebilir

TPAB destekli öğretim programının ön-test son-test kontrol gruplu desende etkisini test etmek amacıyla SPSS istatistik programı kullanılarak bağımsız örneklem için t-testi ile analiz edilmiştir. Bağımsız örneklem için t-testi ile iki ilişkisiz örneklem ortalamaları arasındaki farkın anlamlı olup olmadığı araştırılmıştır. T-testi analizi öncesinde varyansların homojenliğinin sağlanıp sağlanmadığı ölçülmüştür. Bunun için p değerinin 0,05'ten büyük olması gerekmektedir. Buna göre deney ve kontrol gruplarının tüm ön-test ve son-test sonuçlarına bakıldığında varyansların homojen olduğu sonucuna varılmıştır ($p > .05$).

Analizde deney ve kontrol gruplarının fark puanları ortalamaları arasında anlamlı bir farkın olup olmadığına "p" değerinden bakılmıştır. $p < .05$ olduğunda "anlamlı bir fark vardır" ve $p > .05$ olduğunda "anlamlı bir fark yoktur" sonucuna varılmıştır.

Nitel ölçme aracı (görüşme soruları) yalnızca deney grubuna araştırma uygulaması (ders anlatımı) sonrasında uygulanmıştır. Bu çalışmada nicel verilerle beraber nitel verilerin de kullanılmasının amacı toplanan veri ve teoriyle öğrenme çıktılarının etkisi arasındaki ilişkiyi ortaya koymaktır (Creswell ve Clark, 2007).

Görüşme sonucu elde edilen veriler içerik analizi yöntemiyle analiz edilmiştir. İçerik analizinde, veri toplama araçları aracılığıyla elde edilen verileri işleme ve kodlama, temaları bulma, kodların ve temaların düzenlenmesi, bulguların tanımlanması ve yorumlanması olmak üzere dört aşama vardır (Yıldırım ve Şimşek, 2006; Büyüköztürk vd., 2012). Bu çalışmada da veriler kodlanmış ve kodlanan veriler temalara ve alt temalara ayrılmıştır.

4. BULGULAR VE YORUM

4.1. Katılımcıların Betimsel Özellikleri

Katılımcılar yaşları 11-13 arasında değişen 22'si kız 19'u erkek olmak üzere toplam 41 adet 6. Sınıf öğrencisinden oluşmaktadır.

Tablo-6: Katılımcıların Betimsel Özellikleri

Cinsiyet	Deney Grubu	Kontrol Grubu	Toplam
Erkek	9	10	19
Kız	11	11	22
Toplam	20	21	41

4.2. Araştırma Sorularının Cevapları

4.2.1. Öğrencilerin Akademik Başarılarına Göre Ön-Test ve Son-Test Puanları Arasındaki Farka Yönelik Bulgular

Öğrencilerin akademik başarılarına göre ön-test ve son-test puanları arasında anlamlı bir fark var mıdır?

1. Deney ve kontrol grubunun akademik başarılarına göre ön-testleri arasında fark var mıdır?

Tablo-7: Akademik Başarıya Göre Deney ve Kontrol Gruplarının Ön-Test Analizi

Grup	N	X	S	sd	t	p
Deney	20	3,55	1,36	39	0,058	0,954
Kontrol	21	3,52	1,54			

Tablo-7 incelendiğinde öğrencilerin ön test ortalama puanları arasında anlamlı bir fark olmadığı görülmektedir ($t=0,058$, $p=.954>.05$). Deney grubunun akademik başarı ön-testinde doğru cevapladıkları soru sayısı ortalaması $\bar{x} = 3,55$, kontrol grubunun akademik başarı ön-testinde doğru cevapladıkları soru sayısı ortalaması $\bar{x} = 3,52$ ' dir. 25 soruluk başarı testinin her bir sorusu 4 puan olarak hesaplandığında öğrencilerin aldıkları ortalama puan deney grubunun $\bar{x} = 14,20$, kontrol grubunun $\bar{x} = 14,08$ olarak bulunmuştur.

Bu bulgulara dayanarak deney ve kontrol grubu öğrencilerinin akademik başarı yönünden birbirlerinden farklı olmadığı ve ortalamalarının hemen hemen aynı olduğu söylenebilir.

2. Deney ve kontrol grubunun akademik başarılarına göre son-testleri arasında fark var mıdır?

Tablo-8: Akademik Başarıya Göre Deney ve Kontrol Gruplarının Son-Test Analizi

Grup	N	X	S	sd	t	p
Deney	20	18,55	4,80	39	2,45	0,019
Kontrol	21	14,29	6,21			

Tablo-8 incelendiğinde öğrencilerin son test ortalama puanları arasında anlamlı bir fark olduğu görülmektedir ($t=2,45$, $p=.019<.05$). Deney grubunun akademik başarı son-testinde doğru cevapladıkları soru sayısı ortalaması $\bar{x} = 18,55$, kontrol grubunun akademik başarı son-testinde doğru cevapladıkları soru sayısı ortalaması $\bar{x} = 14,29$ ' dur. 25 soruluk başarı testinin her bir sorusu 4 puan olarak hesaplandığında kontrol grubunun ortalama puanı $\bar{x} = 74,20$, kontrol grubunun ortalama puanı $\bar{x} = 57,16$ olarak bulunmuştur.

Bu bulgulara dayanarak deney grubunun ortalaması anlamlı olarak kontrol grubunun ortalamasından yüksek olduğundan dolayı TPAB destekli öğretimin öğrencilerin akademik başarısını artırdığı söylenebilir.

4.2.2. Öğrencilerin Problem Çözme Becerilerine Göre Ön-Test ve Son-Test Puanları Arasındaki Farka Yönelik Bulgular

1. Deney ve kontrol grubunun problem çözme envanteri ön-testleri arasında fark var mıdır?

Tablo-9: Problem Çözme Becerisine Göre Deney ve Kontrol Gruplarının Ön-Test Analizi

Grup	N	X	S	sd	t	p
Deney	20	2,34	0,67	39	-1,46	0,152
Kontrol	21	2,71	0,92			

Tablo-9 incelendiğinde öğrencilerin problem çözme envanteri ön test ortalama puanları arasında anlamlı bir fark olmadığı görülmektedir ($t=-1,46$, $p=.152>.05$). Deney grubunun “problem çözme envanteri” ne verdikleri cevapların ortalaması $\bar{x} = 2,34$, kontrol grubunun “problem çözme envanteri” ne verdikleri cevapların ortalaması $\bar{x} = 2,71'$ dir.

Bu bulgulara dayanarak deney ve kontrol grubu öğrencilerinin problem çözme becerisi yönünden birbirlerinden farklı olmadığı söylenebilir.

2. Deney ve kontrol grubunun problem çözme envanteri son-testleri arasında fark var mıdır?

Tablo-10: Problem Çözme Becerisine Göre Deney ve Kontrol Gruplarının Son-Test Analizi

Grup	N	X	S	sd	t	p
Deney	20	3,63	0,60	39	2,97	0,005
Kontrol	21	2,93	0,87			

Tablo-10 incelendiğinde öğrencilerin ön test ortalama puanları arasında anlamlı bir fark olduğu görülmektedir ($t=2,97$, $p=.005<.05$). Deney grubunun “problem çözme envanteri” ne verdikleri cevapların ortalaması $\bar{x} = 3,63$, kontrol grubunun “problem çözme envanteri” ne verdikleri cevapların ortalaması $\bar{x} = 2,93'$ tür.

Bu bulgulara dayanarak deney grubunun ortalaması anlamlı olarak kontrol grubunun ortalamasından yüksek olduğundan dolayı TPAB destekli öğretimin öğrencilerin problem çözme beceri algısını artırdığı söylenebilir.

4.2.3. Öğrencilerin Bilgisayarca Düşünme Becerilerine Göre Ön-Test ve Son-Test Puanları Arasındaki Farka Yönelik Bulgular

1. Öğrencilerin bilgisayarca düşünme becerisine göre ön test puanları arasında anlamlı bir fark var mıdır?

Tablo-11: Bilgisayarca Düşünme Becerisine Göre Deney ve Kontrol Gruplarının Ön-Test Analizi

Grup	N	X	S	sd	t	p
Deney	20	2,56	0,73	39	-2,00	0,058
Kontrol	21	2,97	0,83			

Tablo-11 incelendiğinde öğrencilerin bilgisayarca düşünme becerisi ön test ortalama puanları arasında anlamlı bir fark olmadığı görülmektedir ($t=-2,00$ $p=.058>.05$). Deney grubunun “bilgisayarca düşünme beceri ölçeği” ne verdikleri cevapların ortalaması $\bar{x} = 2,56$, kontrol grubunun “bilgisayarca düşünme beceri ölçeği” ne verdikleri cevapların ortalaması $\bar{x} = 2,97'$ dir.

Bu bulgulara dayanarak deney ve kontrol grubu öğrencilerinin bilgisayarca düşünme becerisi yönünden birbirlerinden farklı olmadığı söylenebilir.

2. Öğrencilerin bilgisayarca düşünme becerisine göre son test puanları arasında anlamlı bir fark var mıdır?

Tablo-12: Bilgisayarca Düşünme Becerisine Göre Deney ve Kontrol Gruplarının Son-Test Analizi

Grup	N	X	S	sd	t	p
Deney	20	3,87	0,69	39	2,55	0,015
Kontrol	21	3,29	0,77			

Tablo-12 incelendiğinde öğrencilerin bilgisayarca düşünme becerisi son test ortalama puanları arasında anlamlı bir fark olduğu görülmektedir ($t=2,55$ $p=.015<.05$). Deney grubunun “bilgisayarca düşünme beceri ölçeği” ne verdikleri cevapların ortalaması $\bar{x} = 3,87$, kontrol grubunun “bilgisayarca düşünme beceri ölçeği” ne verdikleri cevapların ortalaması $\bar{x} = 3,29'$ dur.

Bu bulgulara dayanarak deney grubunun ortalaması anlamlı olarak kontrol grubunun ortalamasından yüksek olduğundan dolayı TPAB destekli öğretimin öğrencilerin bilgisayarca düşünme becerisini artırdığı söylenebilir.

4.2.4. Öğrencilerin Akademik Başarısı, Problem Çözme Becerisi Algısı ve Bilgisayarca Düşünme Becerisi Arasındaki İlişkiye Yönelik Bulgular

1. Öğrencilerin akademik başarısı ve problem çözme becerisi algısı arasında anlamlı bir ilişki var mıdır?

Tablo-13: Akademik Başarı ve Problem Çözme Becerisi Algısı Arasındaki İlişki

		Akademik Başarı	Problem Çözme Envanteri
Akademik Başarı	Pearson Correlation	1	,506**
	p		,001
	N	41	41
Problem Çözme Envanteri	Pearson Correlation	,506**	1
	p	,001	
	N	41	41

Tablo-13 incelendiğinde öğrencilerin programlama akademik başarı testinden aldıkları ortalama puanla problem çözme envanterinden aldıkları ortalama puan arasında orta düzeyde pozitif yönlü anlamlı bir ilişkinin olduğu söylenebilir ($r=0,506$; $p<0,01$). Bu sonuca göre öğrencilerin akademik başarısı arttıkça problem çözme becerisi algıları da artmaktadır.

2. Öğrencilerin problem çözme becerisi algısı ve bilgisayarca düşünme becerisi arasında anlamlı bir ilişki var mıdır?

Tablo-14: Problem Çözme Becerisi Algısı ve Bilgisayarca Düşünme Becerisi Arasındaki İlişki

		Problem Çözme Envanteri	Bilgisayarca Düşünme Becerisi
Problem Çözme Envanteri	Pearson Correlation	1	,947**
	p		,000
	N	41	41
Bilgisayarca Düşünme Becerisi	Pearson Correlation	,947**	1
	p	,000	
	N	41	41

Tablo-14 incelendiğinde öğrencilerin problem çözme envanterinden aldıkları ortalama puanla bilgisayarca düşünme becerisi ölçeğinden aldıkları ortalama puan arasında pozitif yönlü çok yüksek düzeyde anlamlı bir ilişkinin olduğu söylenebilir

($r=0,947$; $p<0,01$). Bu sonuca göre öğrencilerin problem çözme becerisi algısı arttıkça bilgisayarca düşünme becerisi de artmaktadır.

3. Öğrencilerin akademik başarısı ve bilgisayarca düşünme becerisi arasında anlamlı bir ilişki var mıdır?

Tablo-15: Akademik Başarı ve Bilgisayarca Düşünme Becerisi Arasındaki İlişki

		Akademik Başarı	Bilgisayarca Düşünme Becerisi
Akademik Başarı	Pearson Correlation	1	,484**
	p		,001
	N	41	41
Bilgisayarca Düşünme Becerisi	Pearson Correlation	,484**	1
	p	,001	
	N	41	41

Tablo-15 incelendiğinde öğrencilerin programlama akademik başarı testinden aldıkları ortalama puanla bilgisayarca düşünme becerisi ölçeğinden aldıkları ortalama puan arasında orta düzeyde pozitif yönlü anlamlı bir ilişkinin olduğu söylenebilir ($r=0,484$; $p<0,01$). Bu sonuca göre öğrencilerin akademik başarısı arttıkça bilgisayarca düşünme becerisi de artmaktadır.

4.2.5 Öğrencilerin akademik başarısı, problem çözme becerisi algısı ve bilgisayarca düşünme becerisi üzerinde cinsiyetin etkisine yönelik bulgular

1. Öğrencilerin akademik başarısı cinsiyete göre farklılaşıyor mu?

Tablo-16: Cinsiyete Göre Akademik Başarı

Cinsiyet	N	X	S	sd	t	p
Erkek	19	15,89	6,27	39	-0,47	0,641
Kız	22	16,77	5,68			

Tablo-16 incelendiğinde kız ve erkek öğrencilerin programlama akademik başarı testi ortalama puanları arasında anlamlı bir fark olmadığı görülmektedir ($t=-0,47$, $p=.641>.05$). Erkek öğrencilerin programlama akademik başarı testinde doğru

cevapladıkları soru sayısı ortalaması $\bar{x} = 15,89$, kız öğrencilerin programlama akademik başarı testinde doğru cevapladıkları soru sayısı ortalaması $\bar{x} = 16,77$ ' dir. 25 soruluk başarı testinin her bir sorusu 4 puan olarak hesaplandığında erkek öğrencilerin aldıkları ortalama puan $\bar{x} = 63,56$, kız öğrencilerin $\bar{x} = 67,08$ olarak bulunmuştur.

Bu bulgulara dayanarak kız öğrencilerin ortalamasının yüksek olmasına karşın öğrencilerin programlama akademik başarı testi ortalama puanlarının cinsiyete göre farklılaşmadığı söylenebilir.

2. Öğrencilerin problem çözme becerisi algısı cinsiyete göre farklılaşıyor mu?

Tablo-17: Cinsiyete Göre Problem Çözme Becerisi Algısı

Cinsiyet	N	X	S	sd	t	p
Erkek	19	3,17	0,93	39	-0,71	0,482
Kız	22	3,36	0,73			

Tablo incelendiğinde kız ve erkek öğrencilerin problem çözme envanteri ortalama puanları arasında anlamlı bir fark olmadığı görülmektedir ($t=-0,71$ $p=.482>.05$). Erkek öğrencilerin problem çözme envanterinde bir likert tipi soruya verdikleri ortalama puan $\bar{x} = 3,17$, kız öğrencilerin problem çözme envanterinde bir likert tipi soruya verdikleri ortalama puan $\bar{x} = 3,36$ ' dir.

Bu bulgulara dayanarak kız öğrencilerin ortalamasının yüksek olmasına karşın öğrencilerin problem çözme becerisi algısının cinsiyete göre farklılaşmadığı söylenebilir.

c. Öğrencilerin bilgisayarca düşünme becerisi cinsiyete göre farklılaşıyor mu?

Tablo-18: Cinsiyete Göre Bilgisayarca Düşünme Becerisi

Cinsiyet	N	X	S	sd	t	p
Erkek	19	3,45	0,85	39	-0,89	0,379
Kız	22	3,67	0,71			

Tablo-18 incelendiğinde kız ve erkek öğrencilerin bilgisayarca düşünme becerisi ölçeği ortalama puanları arasında anlamlı bir fark olmadığı görülmektedir ($t=0,89$ $p=.379>.05$). Erkek öğrencilerin bilgisayarca düşünme becerisi ölçeğinde bir likert tipi soruya verdikleri ortalama puan $\bar{x} = 3,45$, kız öğrencilerin bilgisayarca düşünme becerisi ölçeğinde bir likert tipi soruya verdikleri ortalama puan $\bar{x} = 3,67$ dir.

Bu bulgulara dayanarak kız öğrencilerin ortalamasının yüksek olmasına karşın öğrencilerin bilgisayarca düşünme becerisinin cinsiyete göre farklılaşmadığı söylenebilir.

4.2.6 Öğrenci Görüşlerine Yönelik Bulgular

TPAB çerçevesinde hazırlanan Programlama dersine ilişkin öğrenci görüşlerini incelemek amacıyla 12 deney grubu öğrencisine aşağıdaki görüşme soruları yöneltilmiştir:

1. Daha önceki Bilişim Teknolojileri ve Yazılım dersi ünitelerinin işlenişleri ile Programlama konusunun işlenişini karşılaştırır mısın?
2. Programlama konusunun işlenişinde kullanılan TPAB destekli öğrenme yönteminin sana ne gibi etkileri olduğunu düşünüyorsun? Neden?
3. Programlama konusunun işlenişinde kullanılan TPAB destekli öğrenme yöntemi hoşuna gitti mi? Neden?
4. Programlama ünitesinde kullanılan uygulamalar ilgili olarak neler düşünüyorsun?
5. Diğer derslerde de TPAB destekli öğrenme yönteminin kullanılmasını ister miydin? Neden?

Sorulara verilen cevaplar kategori, tema ve kodlara ayrılmıştır.

Tablo-19: Görüşme Sorularına Verilen Cevapların Kategori, Tema ve Kodları

Kategori	Tema	Kod
TPAB destekli dersin diğer ünitelerdeki derslerde n farkları	TPAB destekli programlama dersi	Zevkli Motive edici Açık/Anlaşılır

		Kalıcı
		Sıkıcı/Tekdüze
	Diğer ünitelerdeki dersler	Yorucu
		Geçici
TPAB destekli öğretimin öğrenci üzerine etkileri	Akademik başarı	Diğer derslerdeki başarı BT dersindeki başarı
	Kişisel gelişim	Öz-güven Öz-yeterlilik Öz-denetim
	Öğrenme tekniği	İşbirlikli Öğrenme Bireysel Öğrenme
TPAB destekli öğrenmenin katkıları	Düşünme teknikleri	Yaratıcı Düşünme Algoritmik Düşünme Eleştirel Düşünme
	Bilgi/Beceri	İş birliği Problem Çözme
TPAB destekli ders etkinliklerinin özellikleri	Olumlu	Zevkli Kolay Etkileşimli Yaşama Yakınlık
	Olumsuz	Yetersiz zaman Uzun Zor
TPAB destekli öğretimin diğer derslerde kullanılması	Olumlu	Paylaşım Bireysel çalışma İşbirlikli çalışma Bireysel farklılıklar
	Olumsuz	Çalıntı olasılığı Dersin ciddiyetinin bozulması

Öğrencilerin “Daha önceki Bilişim Teknolojileri ve Yazılım dersi ünitelerinin işlenişleri ile Programlama ünitesinin işlenişini karşılaştırır mısınız?” sorusuna verdiği cevaplar uygun tema ve kodlara ayrılmış, her bir kodun frekansı Tablo-20’de belirtilmiştir.

Tablo-20: Öğrencilerin 1. Görüşme Sorusuna Verdikleri Cevaplar

Tema	Kod	f
------	-----	---

TPAB destekli programlama dersi	Zevkli/eğlenceli	11
	Motive edici	8
	Açık/Anlaşılır	10
	Kalıcı	5
Diğer ünitelerdeki dersler	Sıkıcı/Tekdüze	10
	Yorucu	11
	Geçici	7

Öğrencilerin 1. görüşme sorusuna verdikleri cevaplar “TPAB destekli programlama dersi” ve “diğer ünitelerdeki dersler” olmak üzere 2 temaya ayrılmıştır.

Öğrencilerin “TPAB destekli programlama” kategorisine ait görüşleri incelendiğinde TPAB dersinin diğer ünitelerdeki derslere göre daha zevkli/eğlenceli (n=11), motive edici (n=8), açık/anlaşılır (n=10) olduğu ve bu derste öğrenilen bilgilerin kalıcı (n=5) olduğu görüşlerine sahip oldukları görülür.

Bu sonuçları destekleyen öğrenci cevaplarına aşağıda yer verilmiştir:

Ö2: “Bu ders diğer derslere göre daha zevkli geçtiği için çabuk bitti. Daha doyamadım...”

Ö8: “Bu derse her geldiğimde ne öğreneceğimi merak ederek geldim. Normalde hiç merak etmem.”

Öğrencilerin “Diğer ünitelerdeki dersler” temasına ait görüşleri incelendiğinde Bilişim Teknolojileri ve Yazılım dersinde işlenen diğer derslerin daha sıkıcı/tekdüze (n=10), yorucu (n=11) ve bu derslerde öğrenilen bilgilerin geçici (n=7) olduğu görüşlerine sahip oldukları görülür.

Bu sonuçları destekleyen öğrenci cevaplarına aşağıda yer verilmiştir:

Ö1: “Önceki derslerde sürekli yazı yazıyorduk kendimiz pek bir şey yapmıyorduk, o yüzden eve gidince öğrendiklerimi hemen unutuyordum.”

Ö3: “Önceki dersler hep aynı geçiyordu, öğretmenimiz anlatırdı, biz yazardık sonra da bilgisayarda bir şeyler yazardık, sıkıcıydı.”

Öğrencilerin “Programlama konusunun işlenişinde kullanılan TPAB destekli öğrenme yönteminin sana ne gibi etkileri olduğunu düşünüyorsun? Neden?” sorusuna verdiği cevaplar uygun tema ve kodlara ayrılmış, her bir kodun frekansı Tablo-21’de belirtilmiştir.

Tablo-21: Öğrencilerin 2. Görüşme Sorusuna Verdikleri Cevaplar

Tema	Kod	f
Akademik başarı	Diğer derslerdeki başarı	3
	BT dersindeki başarı	10
Kişisel gelişim	Öz-güven	11
	Öz-yeterlilik	9
	Öz-denetim	5
Öğrenme tekniği	İşbirlikli Öğrenme	9
	Bireysel Öğrenme	8

Öğrencilerin 2. Görüşme sorusuna verdikleri cevaplar “Akademik Başarı”, “Kişisel Gelişim” ve “Öğrenme Tekniği” olmak üzere 3 temaya ayrılmıştır.

Öğrencilerin “Akademik Başarı” temasına ait görüşleri incelendiğinde öğrencilerin TPAB dersinin diğer derslerde (n=3) ve BT dersinde (n=10) akademik başarılarını artırdığı görüşüne sahip oldukları görülür.

Bu sonuçları destekleyen öğrenci cevaplarına aşağıda yer verilmiştir:

Ö8: “Bu dersi işledikten sonraki matematik dersindeki problemleri daha kolay çözmeye başladım, etkisinin olduğunu düşünüyorum.”

Ö9: “Normalde BTY ders notlarım hep 40-45 civarındayken programlama konusundan 72 aldım. Notlarımı yükselttiğini düşünüyorum.”

Öğrencilerin “Kişisel Gelişim” temasına ait görüşleri incelendiğinde öğrencilerin TPAB dersinin özgüvenlerini (n=11), öz-yeterliliklerini (n=9) ve öz-denetimlerini (n=5) artırdığı görüşüne sahip oldukları görülür.

Bu sonuçları destekleyen öğrenci cevaplarına aşağıda yer verilmiştir:

Ö7: “Artık matematik gibi zorlandığım dersleri asla anlayamayacağımı düşünmüyorum, anlayabileceğim konusunda kendime güveniyorum.”

Ö12: “Artık bir sorununun birden fazla çözüm yolu olduğunu öğrendim ve herhangi bir sorunla karşılaştığımda bu çözümleri de düşünüyorum.”

Öğrencilerin “Öğrenme Tekniği” temasına ait görüşleri incelendiğinde öğrencilerin TPAB dersinin işbirlikli öğrenme (n=9) ve bireysel öğrenme (n=8) tekniklerinde başarılı olma olasılığını artırdığı görüşüne sahip oldukları görülür.

Bu sonuçları destekleyen öğrenci cevaplarına aşağıda yer verilmiştir:

Ö3: “Eve gittiğimde artık arkadaşlarımla birlikte oyun oynamak yerine kodlama çalışıyoruz ve hepimiz birbirimize yardım ediyoruz.”

Ö4: “Normalde ders esnasında yapılan etkinliklerde hiç başarılı olmadığım için dolayı yapmaya çekinirim hep başka arkadaşlara sorarım. Ama bu derste bilgisayar beni yönlendirdiği için kendi başıma yapabildim.”

Öğrencilerin “Programlama konusunun işlenişinde kullanılan TPAB destekli öğrenme yöntemi hoşuna gitti mi? Neden?” sorusuna verdiği cevaplar uygun tema ve kodlara ayrılmış, her bir kodun frekansı Tablo-22’de belirtilmiştir.

Tablo-22: Öğrencilerin 3. Görüşme Sorusuna Verdikleri Cevaplar

Tema	Kod	f
Düşünme Teknikleri	Yaratıcı Düşünme	9
	Algoritmik Düşünme	12
	Eleştirel Düşünme	5
Bilgi/Beceri	İş Birliği	8
	Problem Çözme	10

Öğrencilerin 3. görüşme sorusuna verdikleri cevaplar “Düşünme Teknikleri” ve “Bilgi/Beceri” olmak üzere 2 temaya ayrılmıştır.

Öğrencilerin “Düşünme Teknikleri” temasına ait görüşleri incelendiğinde öğrencilerin TPAB dersinin yaratıcı düşünme (n=9) ve algoritmik düşünme (n=12) ve eleştirel düşünme (n=5) tekniklerini geliştirdiği görüşüne sahip oldukları görülür.

Bu sonuçları destekleyen öğrenci cevaplarına aşağıda yer verilmiştir:

Ö5: “Bu dersi sevdim çünkü görsel olduğu için matematiksel sembol ve kavramları daha iyi anlamamı sağladı.”

Ö2: “Kodlarla uğraşırken belirli bir plana göre gitmek, karmaşık problemleri çözmeye çalışmak hoşuma giden şeylerdendi.”

Öğrencilerin “Bilgi/Beceri” temasına ait görüşleri incelendiğinde öğrencilerin TPAB dersinin iş birliği (n=8) ve problem çözme (n=10) becerilerini geliştirdiği görüşüne sahip oldukları görülür.

Bu sonuçları destekleyen öğrenci cevaplarına aşağıda yer verilmiştir:

Ö1: “Dersi sevmemin nedeni arkadaşlarımdan fikirlerinden yararlanmayı bana öğretmesi...”

Ö12: “Bu dersin sayesinde problemleri daha iyi çözebileceğime dair inancım arttı.”

Öğrencilerin “Programlama ünitesinde kullanılan uygulamalar ilgili olarak neler düşünüyorsun?” sorusuna verdiği cevaplar uygun tema ve kodlara ayrılmış, her bir kodun frekansı Tablo-23’te belirtilmiştir.

Tablo-23: Öğrencilerin 4. Görüşme Sorusuna Verdikleri Cevaplar

Tema	Kod	f
Olumlu	Zevkli	12
	Kolay	7
	Etkileşimli	3
	Yaşama Yakınlık	3
Olumsuz	Yetersiz zaman	3
	Uzun	4
	Zor	5

Öğrencilerin 4. görüşme sorusuna verdikleri cevaplar “Olumlu” ve “Olumsuz” olmak üzere 2 temaya ayrılmıştır.

Öğrencilerin “Olumlu” temasına ait görüşleri incelendiğinde öğrencilerin TPAB dersi uygulamalarının zevkli (n=12), kolay (n=7), etkileşimli (n=3) ve yaşama yakın (n=3) oldukları görüşüne sahip oldukları görülür.

Bu sonuçları destekleyen öğrenci cevaplarına aşağıda yer verilmiştir:

Ö5: “Etkinlikler çok zevkli geçti özellikle hareketli olmaları hoşuma gitti. Eve giderken artık sanki oyundaymışım gibi düşünüyorum”

Ö9: “Etkinlikler çok kolaydı, kısa sürede çözdüm.”

Öğrencilerin “Olumsuz” temasına ait görüşleri incelendiğinde öğrencilerin TPAB dersi uygulamalarının yetersiz zamanlı (n=3), uzun (n=4) ve zor (n=5) oldukları görüşüne sahip oldukları görülür.

Bu sonuçları destekleyen öğrenci cevaplarına aşağıda yer verilmiştir:

Ö1: “Etkinlikler bana çok uzun geldi bu yüzden bazılarında zamanı yetiştiremedim.”

Ö2: “Etkinlikler bana çok karmaşık ve zorlayıcı geldi.”

Öğrencilerin “Diğer derslerde de TPAB destekli öğrenme yönteminin kullanılmasını ister miydin? Neden?” sorusuna verdiği cevaplar uygun tema ve kodlara ayrılmış, her bir kodun frekansı Tablo-24’te belirtilmiştir.

Tablo-24: Öğrencilerin 5. Görüşme Sorusuna Verdikleri Cevaplar

Tema	Kod	f
Olumlu	Paylaşım	9
	Bireysel çalışma	5
	İşbirlikli çalışma	11
	Bireysel Farklılıklar	4
Olumsuz	Çalıntı olasılığı	2
	Dersin ciddiyetinin bozulması	3

Öğrencilerin 5. görüşme sorusuna verdikleri cevaplar “Olumlu” ve “Olumsuz” olmak üzere 2 temaya ayrılmıştır.

Öğrencilerin “Olumlu” temasına ait görüşleri incelendiğinde öğrencilerin TPAB destekli öğretimin diğer derslerde kullanılmasının paylaşımı (n=9), bireysel çalışmayı (n=5), işbirlikli çalışmayı (n=11) desteklediğini ve bireysel farklılıklara uygun (n=4) olduğu görüşüne sahip oldukları görülür.

Bu sonuçları destekleyen öğrenci cevaplarına aşağıda yer verilmiştir:

Ö4: “Başka dersler de bu şekilde işlenirse internet üzerinden her yaptığımızı paylaşabilir ve arkadaşlarımızın fikirlerini alabiliriz.”

Ö8: “Diğer dersleri bu şekilde işleyebilsek görsel öğeler fazla olduğu için bana daha çok hitap ederdi.”

Öğrencilerin “Olumsuz” temasına ait görüşleri incelendiğinde öğrencilerin TPAB destekli öğretimin diğer derslerde kullanılmasının çalıntı (n=2), ve dersin ciddiyetinin bozulması (n=3) olasılığının artırabileceği görüşüne sahip oldukları görülür.

Bu sonuçları destekleyen öğrenci cevaplarına aşağıda yer verilmiştir:

Ö2: “Her şeyi internette paylaşabildiğimiz için başkaları bizim projemizden ya da ödevimizden kopya çekebilir.”

Ö6: “Her dersi bu şekilde işlersek derse öğrenciler ders gözüyle bakmayabilir, gevşeyebilirler.”

5. SONUÇ VE TARTIŞMA

Eğitim teknolojilerinin hayatımıza girmesiyle birlikte teknolojinin eğitime entegrasyonu terimi ortaya çıkmıştır. Teknolojinin eğitime entegrasyonu, sadece eğitime teknolojiyi eklemek değil, eğitimin amaçlarına uygun olarak teknolojiyi eğitimin ayrılmaz bir parçası haline getirmek, eğitimi teknolojiyle planlamak, değerlendirmek ve uygulamaktır. Teknoloji entegrasyonunun istendik şekilde ilerlemesi öğretmenlerin teknoloji, pedagoji ve alan bilgilerinin üçüne de hakim olmalarına bağlıdır (Pierson, 2001). Bu nedenle, Mishra ve Koehler (2006) Teknolojik Pedagojik Alan Bilgisi (TPAB) çerçevesini oluşturmuşlardır. Araştırmacılar tarafından TPAB çerçevesinin teknoloji bazlı öğretim programı planlamaya olumlu etki ettiği (Angeli ve Valanides, 2009) ve hatta TPAB çerçevesi esas alınarak oluşturulan öğrenme/öğretme aktivitelerinin (Harris ve Hofer, 2009), TPAB'in sadece öğretmen eğitiminde değil ilk ve ortaöğretim eğitiminde de hem öğrenci hem öğretmenlerin yararına kullanılabileceğini ortaya çıkarmıştır. Literatürde genel anlamda TPAB çerçevesiyle oluşturulmuş ders planlarının öğretmen ve öğrenci üzerine etkilerini araştıran çalışmalar yer almaktadır (Brill, Listman ve Kapila, 2015; Kontkanen vd., 2017; Hofer ve Harris, 2010; Wetzel ve Marshall, 2011; Doering vd., 2014; Aisyah, 2013). Literatürde ortaokul düzeyinde öğrencilerle bilgisayar/bilişim dersinde TPAB çerçevesiyle yapılan öğretim programlarının kullanılmasıyla ilgili çalışmalara rastlanmamıştır. Bu araştırma boşluğunu doldurmak amacıyla bu çalışmada “TPAB çerçevesiyle oluşturulan Bilişim Teknolojileri ve Yazılım dersinin öğrenciler üzerine etkileri” araştırılmıştır. Programlama dersinin öğrenciler tarafından genelde anlaşılması en güç derslerden görüldüğü için (Yadin, 2011) uygulamanın “Problem Çözme ve Programlama” ünitesinin “Programlama” konusu için yapılması kararlaştırılmıştır. Ön-test, son-test deney ve kontrol grubu yarı-deneysel desenle yürütülen araştırmanın sonucunda şu yargılara ulaşılmıştır:

Deney ve kontrol grubunun akademik başarılarına göre ön-testleri arasında anlamlı bir fark yoktur, bu iki grubun programlama konusu akademik başarı düzeyi açısından birbirine eş olduğunu gösterir. Deney ve kontrol grubunun akademik başarılarına göre son-testleri arasında yapılan bağımsız örneklem t-testi analizinde deney grubunun lehine anlamlı bir fark bulunmuştur. Deney grubunun akademik

başarı testinden aldığı ortalama puan 74,20 iken kontrol grubunun ortalaması 57,16' dır. Benzer şekilde yürütülen araştırmalarda TPAB çerçevesiyle kurulmuş ders planlarının matematik ve fen bilgisi soyut konuların öğretiminde (Brill, vd., 2015) ve coğrafya dersinin öğretiminde (Doering, vd., 2014) etkili olduğu ortaya çıkmıştır. Deney ve kontrol grubunun problem çözme becerilerine göre ön-testleri arasında anlamlı bir fark yoktur, bu iki grubun problem çözme becerileri açısından birbirine eş olduğunu gösterir. Deney ve kontrol grubunun akademik başarılarına göre son-testleri arasında yapılan bağımsız örneklem t-testi analizinde deney grubunun lehine anlamlı bir fark bulunmuştur. Deney grubunun problem çözme envanterinden aldığı ortalama puan 3,63 iken kontrol grubunun ortalaması 2,93 tür. Deney ve kontrol grubunun bilgisayarca düşünme becerilerine göre ön-testleri arasında anlamlı bir fark yoktur, bu iki grubun bilgisayarca düşünme becerileri açısından birbirine eş olduğunu gösterir. Deney ve kontrol grubunun akademik başarılarına göre son-testleri arasında yapılan bağımsız örneklem t-testi analizinde deney grubunun lehine anlamlı bir fark bulunmuştur. Deney grubunun bilgisayarca düşünme becerisi ölçeğinden aldığı ortalama puan 3,87 iken kontrol grubunun ortalaması 3,29 dur. Problem çözme ve bilgisayarca düşünme gibi üst düzey becerilerin TPAB çerçevesiyle yürütülen derse katılan öğrencilerde daha yüksek çıkması yapılan bazı araştırmalarla benzer çıkmıştır. Bu araştırmalardan birinde TPAB modelinin kullanılması öğrencilerin eleştirel düşünme ve bilgisayar okuryazarlığını artırdığı (Aisyah, 2013), bir diğerinde ise eleştirel düşünme, yaratıcılık ve işbirlikçi iletişim gibi becerileri geliştirdiği (Brown, vd., 2011) belirlenmiştir. Deney ve kontrol grubunun akademik başarısı ve problem çözme becerisi algısı arasında yapılan korelasyon analizinde orta seviyede pozitif yönlü anlamlı bir ilişki bulunmuştur. Deney ve kontrol grubunun problem çözme becerisi algısı ve bilgisayarca düşünme becerisi arasında yapılan korelasyon analizinde yüksek seviyede pozitif yönlü anlamlı bir ilişki bulunmuştur. Deney ve kontrol grubunun akademik başarısı ve bilgisayarca düşünme becerisi arasında yapılan korelasyon analizinde orta seviyede pozitif yönlü anlamlı bir ilişki bulunmuştur. Kız ve erkek öğrencilerin programlama akademik başarıları, problem çözme becerisi ve bilgisayarca düşünme becerisi arasında anlamlı bir fark yoktur. Bu sonuca benzer şekilde bilgisayar kullanma becerilerinin ölçüldüğü bir araştırmada ise kız öğrenciler ve erkek öğrenciler arasında fark çıkmamıştır (Saçkes, vd., 2011). Bunun tersine başka

bir arařtırmada ise bilgisayarca dūřünme becerisinde kızların erkeklere göre daha başarılı olduđu ortaya çıkmıřtır (Howland ve Good, 2014). Öğrencilerin görüşme sorularına verdikleri cevaplarından toplanan nitel veriler, nicel verilerle paralellik göstermiştir. Görüşme sorularına verilen cevaplara bakıldığında öğrencilerin TPAB çerçevesiyle yürütölen dersten genel olarak memnun oldukları, bu dersin algoritmik dūřünme ve problem çözüme becerisi gibi üst düzey becerilerini geliřtirdiđini dūřündükleri ve akademik başarılarını artırdıđı ortaya çıkmıřtır.

5.1 Öneriler

1. Arařtırma yalnızca toplamda 41 kiřiyle sınırlı olduđu için gelecekteki arařtırmalarda örneklemin daha geniş tutulması önerilmektedir. Aynı zamanda arařtırma yalnızca 6.sınıfta okuyan öğrencilerle sınırlı olduđu için gelecekteki arařtırmalarda yař aralıđının geniş tutulması yararlı olacaktır.

2. Arařtırmada öğrencilerin problem çözüme becerileri ve bilgisayarca dūřünme becerilerinin biliřim teknolojileri ve yazılım dersindeki akademik başarılarıyla iliřkisi arařtırılmıřtır. Gelecekteki arařtırmalarda öğrencilerin başka derslerdeki akademik başarılarının problem çözüme ve bilgisayarca dūřünme becerilerine olan etkileri analiz edilebilir.

3. Programlama dersini ilk defa alan öğrencilerin eleřtirel dūřünme, iřbirlikçi iletiřim, yaratıcılık gibi üst düzey dūřünme becerilerinin geliřip geliřmediđi ölçülebilir. Aynı řekilde programlama dersinin diđer derslerdeki akademik başarıyla iliřkisi ölçülebilir.

4. TPAB çerçevesinde ders planı oluřturmanın öğretmenler üzerine etkisi arařtırılabilir.

6. KAYNAKÇA

- Aisyah, A. R. (2013). The Development of Working Design through Characterized Technology Pedagogy and Content Knowledge in the Elementary Schools' Instructional. *Procedia-Social and Behavioral Sciences*, 103, 1016-1024.
- Akbaba-Altun, S. (2004). Information technology classrooms and elementary school principals' roles: Turkish experience. *Education and Information Technologies*, 9(3), 255-270.
- Akbaba-Altun, S. (2005). Bilgi Teknolojisi Sınıflarında Denetim. *Eurasian Journal of Educational Research (EJER)*, (18).
- Akgün, E., Yılmaz, E. O., ve Seferoğlu, S. S. (2011). Vizyon 2023 strateji belgesi ve fırsatları artırma ve teknolojiyi iyileştirme hareketi (FATİH) projesi: Karşılaştırmalı bir inceleme. *Akademik Bilişim*, 2(4), 115-122.
- Akkoyunlu, B., ve İmer, D. G. (1998). Türkiye'de eğitim teknolojisinin görünümü. *Çağdaş Eğitimde Yeni Teknolojiler, Anadolu Üniversitesi, Eskişehir*, 160-168.
- Albayrak Sarı, A., Canbazoğlu Bilici, S., Baran, E., ve Özbay, U. (2016). Farklı branşlardaki öğretmenlerin teknolojik pedagojik alan bilgisi (TPAB) yeterlikleri ile bilgi ve iletişim teknolojilerine yönelik tutumları arasındaki ilişkinin incelenmesi. *Eğitim Teknolojisi Kuram ve Uygulama; Cilt 6, Sayı 1*.
- Altun, E., ve Ateş, A. (2008). The problems and future concerns of computer and instructional technologies preservice teachers. *Elementary Education Online*, 7(3), 680-692.
- Amaratunga, D., Baldry, D., Sarshar, M., & Newton, R. (2002). Quantitative and qualitative research in the built environment: application of "mixed" research approach. *Work study*, 51(1), 17-31.

- Andrew, S., and Halcomb, E. J. (2009). *Mixed methods research for nursing and the health sciences*. John Wiley & Sons.
- Angeli, C., and Valanides, N. (2005). Preservice elementary teachers as information and communication technology designers: An instructional systems design model based on an expanded view of pedagogical content knowledge. *Journal of Computer Assisted Learning*, 21(4), 292-302.
- Angeli, C., & Valanides, N. (2009). Epistemological and methodological issues for the conceptualization, development, and assessment of ICT–TPCK: Advances in technological pedagogical content knowledge (TPCK). *Computers & education*, 52(1), 154-168.
- Archambault, L., Wetzel, K., Foulger, T. S., and Kim Williams, M. (2010). Professional development 2.0: Transforming teacher education pedagogy with 21st century tools. *Journal of Digital Learning in Teacher Education*, 27(1), 4-11.
- Area, M., Hernández, V., and Sosa, J. (2016). Models of educational integration of ICT in the classroom. *Revista Comunicar*, 24(47), 79-87.
- Atmatzidou, S., and Demetriadis, S. (2016). Advancing students' computational thinking skills through educational robotics: A study on age and gender relevant differences. *Robotics and Autonomous Systems*, 75, 661-670.
- Baran, E., Chuang, H. H., and Thompson, A. (2011). TPACK: An emerging research and development tool for teacher educators. *Turkish Online Journal of Educational Technology-TOJET*, 10(4), 370-377.
- Barron, A. E., Kemker, K., Harnes, C., and Kalaydjian, K. (2003). Large-scale research study on technology in K–12 schools: Technology integration as it relates to the National Technology Standards. *Journal of Research on Technology in Education*, 35(4), 489-507.
- Bayrakçı, M. (2005). Avrupa Birliği ve Türkiye eğitim politikalarında bilgi ve iletişim teknolojileri ve mevcut uygulamalar. *Milli Eğitim*, 33(167), 20.

- Bers, M. U., Flannery, L., Kazakoff, E. R., and Sullivan, A. (2014). Computational thinking and tinkering: Exploration of an early childhood robotics curriculum. *Computers & Education*, 72, 145-157.
- Brill, A. S., Listman, J. B. and Kapila, V. (2015). Using robotics as the technological foundation for the TPACK framework in K-12 classrooms. In *2015 ASEE Annual Conference & Exposition* (pp. 26-1679).
- Brown, C. A., Neal, R. E. and Fine, B. (2011). Using 21st Century Thinking Skills Applied to the TPACK Instructional Model. *Paper presented at the annual meeting of the AECT International Convention, Hyatt Regency Jacksonville Riverfront, Jacksonville, FL, Nov 08, 2011*
- Brown, J. D. (1996). *Testing in language programs*. Upper Saddle River, NJ: Prentice Hall.
- BTE (Bilişim Teknolojileri Eğitimcileri Derneği). (2013). Türkiye’de ilk ve ortaokullarda (ilköğretim) okutulan bilişim teknolojileri derslerinin tarihi.
- Büyüköztürk, S., Kiliç Çakmak, E., Akgün, Ö. E., Karadeniz, S., ve Demirel, F. (2012). Bilimsel Arastırma Yöntemleri (18. Baskı). *Ankara: Pegem Akademi Yayıncılık*.
- CAEP (Council for the Accreditation of Education Program). (2016). *2013 CAEP Standards*. <http://www.ncate.org/~media/Files/caep/standards/caep-standards-one-pager-061716.pdf?la=en>
- Carton, R. (2017). TPACK learning activity types for secondary computer science courses" (2017). *Graduate Research Papers*. 139.
- CCSSO (Council of Chief State School Officers). (2013, April). Interstate Teacher Assessment and Support Consortium *InTASC Model Core Teaching Standards and Learning Progressions for Teachers 1.0: A Resource for Ongoing Teacher Development*. Washington, DC: Author.

- Chao, P. Y. (2016). Exploring students' computational practice, design and performance of problem-solving through a visual programming environment. *Computers & Education, 95*, 202-215.
- Chuang, H. H., Weng, C. Y., and Huang, F. C. (2015). A structure equation model among factors of teachers' technology integration practice and their TPACK. *Computers & Education, 86*, 182-191.
- Ciampa, K. (2017). Building bridges between technology and content literacy in special education: Lessons learned from special educators' use of integrated technology and perceived benefits for students. *Literacy Research and Instruction, 56*(2), 85-113.
- Cox, S. M. (2008). *A Conceptual Analysis of Technological Pedagogical Content Knowledge*. Doctor of Philosophy, BRIGHAM YOUNG UNIVERSITY Department of Instructional Psychology & Technology, Provo.
- Cox, S., and Graham, C. R. (2009). Using an elaborated model of the TPACK framework to analyze and depict teacher knowledge. *TechTrends, 53*(5), 60-69.
- Creswell, J. W. (2003). *Research design: Qualitative, quantitative, and mixed methods approaches* (2nd ed.). Thousand Oaks, CA: Sage.
- Creswell, J. W., and Clark, V. L. P. (2017). *Designing and conducting mixed methods research*. Sage publications.
- CSTA (Computer Science Teachers Association). (2016). K-12 Computer science standards. *Computer Science Teachers Association*.
- Cuban, L., Kirkpatrick, H. and Peck, C. (2001). High access and low use of technologies in high school classrooms: Explaining an apparent paradox. *American educational research journal, 38*(4), 813-834.

- Davies, R. S., and West, R. E. (2014). Technology integration in schools. *Handbook of research on educational communications and technology*, 841-853. Springer, New York, NY.
- Demirer, V., ve Sak, N. (2015). Türkiye’de bilişim teknolojileri (BT) eğitimi ve BT öğretmenlerin değişen rolleri. *Uluslararası Eğitim Bilimleri Dergisi*, 2(5), 434-448.
- Denner, J., Werner, L., and Ortiz, E. (2012). Computer games created by middle school girls: Can they be used to measure understanding of computer science concepts?. *Computers & Education*, 58(1), 240-249.
- Dexter, S., Anderson, R. E., and Becker, H. J. (1999). Teachers’ views of computers as catalysts for changes in their teaching practice. *Journal of Research on Computing in Education*, 31 (3), 221-239.
- Doering, A., Koseoglu, S., Scharber, C., Henrickson, J., and Lanegran, D. (2014). Technology integration in K–12 geography education using TPACK as a conceptual model. *Journal of Geography*, 113(6), 223-237.
- Doering, A., Veletsianos, G., Scharber, C., and Miller, C. (2009). Using the technological, pedagogical, and content knowledge framework to design online learning environments and professional development. *Journal of Educational Computing Research*, 41(3), 319-346.
- Doukakis, S., Psaltidou, A., Stavradi, A., Adamopoulos, N., Tsiotakis, P., and Stergou, S. (2010). Measuring the technological pedagogical content knowledge (TPACK) of in-service teachers of computer science who teach algorithms and programming in upper secondary education. *Readings in technology and education: Proceedings of ICICTE*, 442-452.
- Eryılmaz, S., ve Salman, Ş. (2014). FATİH Projesi kapsamında yer alan öğretmen ve öğrencilerin projeden beklentileri ve bilişim teknolojileri kullanımına karşı algıları. *Elektronik Mesleki Gelişim Ve Araştırmalar Dergisi*, 2(1), 46-63.

- Escudero, E. B., Reyna, N. L., and Morales, M. R. (2000). The level of difficulty and discrimination power of the Basic Knowledge and Skills Examination (EXHCOBA). *Revista electrónica de investigación educativa*, 2(1), 2.
- Farrell, I. K., and Hamed, K. M. (2017). Examining the Relationship Between Technological Pedagogical Content Knowledge (TPACK) and Student Achievement Utilizing the Florida Value-Added Model. *Journal of Research on Technology in Education*, 49(3-4), 161-181.
- Fiş Erümit, S. Gedik ve N. Göktaş, Y. (2016). Türkiye’de öğretim teknolojilerinin gelişimi: 1984—2015 Dönemi. *Öğretim Teknolojilerinin Temelleri: Teoriler, Araştırmalar, Eğilimler*. Ankara: Pegem Akademi.
- Friske, J., Knezek, D., Taylor, H., Thomas, L., and Wiebe, J. (1995). ISTE’s technology foundation standards for all teachers: Time for a second look?. *Journal of Computing in Teacher Education*, 12(2), 9-12.
- Giannakos, M. N., Doukakis, S., Pappas, I. O., Adamopoulos, N., and Giannopoulou, P. (2015). Investigating teachers’ confidence on technological pedagogical and content knowledge: an initial validation of TPACK scales in K-12 computing education context. *Journal of Computers in Education*, 2(1), 43-59.
- Giannakos, M., Hubwieser, P., and Chrisochoides, N. (2013). How students estimate the effects of ICT and programming courses. In *Proceeding of the 44th ACM technical symposium on Computer science education* (pp. 717-722). ACM.
- Grout, V., and Houlden, N. (2014). Taking computer science and programming into schools: The Glyndŵr/BCS Turing project. *Procedia-Social and Behavioural Sciences*, 141, 680-685.
- Handler, M. G., and Strudler, N. (1997). The ISTE foundation standards: Issues of implementation. *Journal of Computing in Teacher Education*, 13(2), 16-23.
- Harris, J., and Hofer, M. (2009). Instructional planning activity types as vehicles for curriculum-based TPACK development. In *Society for Information Technology &*

Teacher Education International Conference (pp. 4087-4095). Association for the Advancement of Computing in Education (AACE).

Harris, J. B., and Hofer, M. J. (2011). Technological pedagogical content knowledge (TPACK) in action: A descriptive study of secondary teachers' curriculum-based, technology-related instructional planning. *Journal of Research on Technology in Education*, 43(3), 211-229.

Harris, J. B., and Hofer, M. J. (2011). Technological pedagogical content knowledge (TPACK) in action: A descriptive study of secondary teachers' curriculum-based, technology-related instructional planning. *Journal of Research on Technology in Education*, 43(3), 211-229.

Hofer, M., and Harris, J. (2010). Differentiating TPACK development: Using learning activity types with inservice and preservice teachers. In *Society for Information Technology & Teacher Education International Conference* (pp. 3857-3864). Association for the Advancement of Computing in Education (AACE).

Howland, K., and Good, J. (2015). Learning to communicate computationally with Flip: A bi-modal programming language for game creation. *Computers & Education*, 80, 224-240.

Hu, C., and Fyfe, V. (2010). Impact of a new curriculum on pre-service teachers' Technical, Pedagogical and Content Knowledge (TPACK). *Curriculum, technology and transformation for an unknown future: Proceedings ascilite Sydney*, 185-189.

Hubwieser, P., Armoni, M., Giannakos, M. N., and Mittermeir, R. T. (2014). Perspectives and visions of computer science education in primary and secondary (K-12) Schools. *ACM Transactions on Computing Education (TOCE)*, 14(2), 7.

Hubwieser, P., Magenheimer, J., Mühling, A., and Ruf, A. (2013, August). Towards a conceptualization of pedagogical content knowledge for computer science. In *Proceedings of the ninth annual international ACM conference on International computing education research* (pp. 1-8). ACM.

- Inan, F. A., Lowther, D. L., Ross, S. M., and Strahl, D. (2010). Pattern of classroom activities during students' use of computers: Relations between instructional strategies and computer applications. *Teaching and Teacher Education*, 26(3), 540-546.
- Ioannou, I., and Angeli, C. (2013). Teaching computer science in secondary education: A technological pedagogical content knowledge perspective. In *Proceedings of the 8th Workshop in Primary and Secondary Computing Education* (pp. 1-7). ACM.
- ISTE (International Society for Technology in Education). (2007). National educational technology standards and performance indicators for students. Eugene, OR: International Society for Technology in Education
- ISTE (International Society for Technology in Education). (2016). *ISTE Standards for Students* http://www.iste.org/docs/Standards-Resources/iste-standards_students-2016_one-sheet_final.pdf?sfvrsn=0.23432948779836327
- ISTE (International Society for Technology in Education). (2017). *ISTE Standards for Educators* <https://www.iste.org/standards/for-educators>
- Januszewski, A., and Persichitte, K. A. (2008). A history of the AECT's definitions of educational technology. *Educational technology: A definition with commentary*. New York and London: Routledge, 259-282.
- Karaağaçlı, M., ve Erden, O. (2008). İnternet destekli uzaktan eğitimde dokuz aşamalı öğretim durumunun tasarımı. *Bilişim Teknolojileri Dergisi*, 1(2).
- Karasar, N. (2015). *Bilimsel Araştırma Yöntemi: Kavramlar İlkeler Teknikler*. Ankara: Nobel.
- Kaya, Z., Özdemir, T. Y., Emre, İ., ve Kaya, O. N. (2011). Exploring preservice information technology teachers' perception of self-efficacy in web-technological pedagogical content knowledge. In *6th International Advanced Technologies Symposium (IATS'11)*, Elazığ.

- Kazu, İ. Y., ve Yavuzalp, N. (2010). Öğretim yazılımlarının kullanımına ilişkin öğretmen görüşleri. *Eğitim ve Bilim*, 33(150).
- Kim, H. Y. (2013). Statistical notes for clinical researchers: assessing normal distribution (2) using skewness and kurtosis. *Restorative dentistry & endodontics*, 38(1), 52-54.
- Koehler, M., and Mishra, P. (2009). What is technological pedagogical content knowledge (TPACK)?. *Contemporary Issues in Technology And Teacher Education*, 9(1), 60-70.
- Kontkanen, S., Dillon, P., Valtonen, T., Eronen, L., Koskela, H., and Väisänen, P. (2017). Students' experiences of learning with iPads in upper secondary school—a base for proto-TPACK. *Education and Information Technologies*, 22(4), 1299-1326.
- Koorsse, M., Cilliers, C., and Calitz, A. (2015). Programming assistance tools to support the learning of IT programming in South African secondary schools. *Computers & Education*, 82, 162-178.
- Korkmaz, Ö., Çakır, R., ve Özden, M. Y. (2016). Bilgisayarca Düşünme Beceri Düzeyleri Ölçeğinin (Bdbd) Ortaokul Düzeyine Uyarlanması. *Gazi Eğitim Bilimleri Dergisi*, 1(2).
- Korkmaz, Ö., Çakır, R., ve Özden, M. Y. (2017). A validity and reliability study of the Computational Thinking Scales (CTS). *Computers in Human Behavior*, 72, 558-569.
- Kozma, R. B. (1994). Will media influence learning? Reframing the debate. *Educational technology research and development*, 42(2), 7-19.
- Laferrière, T., Hamel, C. and Searson, M. (2013). Barriers to successful implementation of technology integration in educational settings: A case study. *Journal of Computer Assisted Learning*, 29(5), 463-473.

- Lee, M. H., and Tsai, C. C. (2010). Exploring teachers' perceived self efficacy and technological pedagogical content knowledge with respect to educational use of the World Wide Web. *Instructional Science*, 38(1), 1-21.
- Lin, J. M. C., Yen, L. Y., Yang, M. C., and Chen, C. F. (2005, June). Teaching computer programming in elementary schools: A pilot study. In *National educational computing conference*.
- Lingenfelter, B. (2015). Building Lessons with Web 2.0 & TPACK. *American College of Education*.
- Lodico, M. G., Spaulding, D. T., and Voegtler, K. H. (2010). *Methods in educational research: From theory to practice* (Vol. 28). John Wiley & Sons.
- Lyublinskaya, I., and Tournaki, N. (2011, March). The effects of teacher content authoring on TPACK and on student achievement in algebra: Research on instruction with the TI-Nspire handheld. In *Society for Information Technology & Teacher Education International Conference* (pp. 4396-4401). Association for the Advancement of Computing in Education (AACE).
- MEB. (1993). Bilgisayar öğretmenleri ile bilgisayar koordinatör öğretmenlerinin seçimi ve görevleri. *2378 Sayılı Tebliğler Dergisi*. 215-217
- MEB. (2001). "Bilgi Teknolojilerinin Kullanımı" Konulu 27.06.2001 tarih ve 5985 sayılı genelge.
- MEB. (2007). Temel Eğitim Projesi II. Fazı: BT Entegrasyonu Temel Araştırması Raporu. Ankara: Bilgitek Eğitim Danışmanlık ve Taahhüt A.Ş.
- MEB. (2011). <http://fatihprojesi.meb.gov.tr/proje-hakkinda/>
- MEB. (2012a). "12 Yıllık Zorunlu Eğitime Yönelik Uygulamalar" Konulu 09.05.2012 tarih ve 401 sayılı genelge.
- MEB. (2012b). "Bilişim Teknolojileri Rehberliği Görevi" Konulu 28.09.2012 tarih ve 16791 sayılı resmi yazı.

- MEB (2012c). “Bilişim Teknolojileri Rehberliği Görevi (Tereddüt edilen hususlar)” Konulu 27.12.2012 tarih ve 248065 sayılı resmi yazı.
- MEB. (2015). “Fatih Projesi BT Rehberliği Görevi” konulu 30.12.2015 tarih ve 13507730 sayılı resmi yazı.
- Mercimek, B., ve İlic, U., (2017). Bilişim Teknolojileri ve Yazılım Dersi Öğretim Programı Güncelleme Önerisine Yönelik Bir Değerlendirme. *Academia Eğitim Araştırmaları Dergisi*, 2(1).
- Mishra, P. and Koehler, M. (2006). Technological pedagogical content knowledge: A framework for teacher knowledge. *The Teachers College Record*, 108(6), 1017-1054.
- Munson, A., Moskal, B., Harriger, A., Lauriski-Karriker, T. and Heersink, D. (2011). Computing at the high school level: Changing what teachers and students know and believe. *Computers & Education*, 57(2), 1836-1849.
- Naughton, J. (2012). Why all our kids should be taught how to code. *Observer New Review*, 31(3), 12.
- Newhouse, C. P. (2011). Using IT to assess IT: Towards greater authenticity in summative performance assessment. *Computers & Education*, 56(2), 388-402.
- Niess, M. L. (2005). Preparing teachers to teach science and mathematics with technology: Developing a technology pedagogical content knowledge. *Teaching and teacher education*, 21(5), 509-523.
- Oblinger, D., Oblinger, J. L. and Lippincott, J. K. (2005). *Educating the net generation*. Brockport: Educause.
- Oster-Levinz, A., and Klieger, A. (2010). Online tasks as a tool to promote teachers' expertise within the Technological Pedagogical Content Knowledge (TPACK). *Procedia-Social and Behavioral Sciences*, 2(2), 354-358.

- Ouahbi, I., Kaddari, F., Darhmaoui, H., Elachqar, A., and Lahmine, S. (2015). Learning basic programming concepts by creating games with scratch programming environment. *Procedia-Social and Behavioral Sciences*, 191, 1479-1482.
- Özdemir, S. (2010). 'To err is human, but to persist is diabolical': Loss of organizational memory and e-learning projects. *Computers & Education*, 55(1), 101-108.
- Özden M.Y. ve Şimşek, H. (1998). Davranışçılıktan Oluşturmacılığa: "Öğrenme" Paradigmasının Dönüşümü ve Türk Eğitimi. *BT/Bilgi ve Toplum, Türk Dünyası Araştırmaları Vakfı*, 71-82.
- Pelgrum, W. J. and Law, N. W. Y. (2003). ICT in education, some major concepts and a short historical overview. *ICT in education around the world: Trends, problems and prospects*. UNESCO: International Institute for Educational Planning, 19-31.
- Piaget, J. (1963). The attainment of invariants and reversible operations in the development of thinking. *Social research*, 283-299.
- Pierson, M. E. (2001). Technology integration practice as a function of pedagogical expertise. *Journal of research on computing in education*, 33(4), 413-430.
- Pilgrim, J. and Martinez E. (2015). Web literacy and technology integration: Moving beyond TPACK with student-centered instruction. *Journal of Literacy and Technology*, 16(2).
- Polly, D., Mims, C., Shepherd, C. E., and Inan, F. (2010). Evidence of impact: Transforming teacher education with preparing tomorrow's teachers to teach with technology (PT3) grants. *Teaching and Teacher Education*, 26(4), 863-870.
- Prensky, M. (2001). Digital natives, digital immigrants part 1. *On the horizon*, 9(5), 1-6.
- Reiser, R. A. (1987). Instructional technology: A history. *Instructional technology: foundations*, 11-48.

- Reiser, R. A. (2007). What field did you say you were in. *Trends and issues in instructional design and technology*, 2-9.
- Richey, R. C. and Seels, B. (1994). The 1994 Definition of the Field. *Instructional Technology: The Definition and Domains of the Field*. Bloomington: AECT, 1-22.
- Ross, S. M. (2014). Introduction to probability and statistics for engineers and scientists. Academic Press.
- RTI (Research Triangle Institute International) ve ERG (Eğitim Reformu Girişimi). (2013). FATİH Projesi Eğitimde Dönüşüm İçin Bir Fırsat Olabilir mi? Politika Analizi ve Önerileri. *ERG Raporları*.
- Saçkes, M., Trundle, K. C., and Bell, R. L. (2011). Young children's computer skills development from kindergarten to third grade. *Computers & Education*, 57(2), 1698-1704.
- Saeli, M., Perrenet, J., Jochems, W. M., and Zwaneveld, B. (2011). Teaching programming in secondary school: a pedagogical content knowledge perspective. *Informatics in Education*, 10(1).
- Sáez-López, J. M., Román-González, M., and Vázquez-Cano, E. (2016). Visual programming languages integrated across the curriculum in elementary school: A two year case study using "Scratch" in five schools. *Computers & Education*, 97, 129-141.
- Schurr, S. (1989). *Dynamite in the classroom: A how-to handbook for teachers*. National Middle School Association.
- Seferoğlu, S. S., Akbıyık, C., ve Bulut, M. (2008). İlköğretim öğretmenlerinin ve öğretmen adaylarının bilgisayarların öğrenme/öğretme sürecinde kullanımı ile ilgili görüşleri. *Hacettepe Üniversitesi Eğitim Fakültesi Dergisi*, 35(35).
- Serin, O., Bulut Serin, N. B., ve Saygılı, G. (2010). İlköğretim düzeyindeki Çocuklar için Problem Çözme Envanteri'nin (ÇPÇE) geliştirilmesi. *İlköğretim Online*, 9(2).

- Sezer, B. (2011). Bilişim teknolojilerinin eğitime kaynaştırılması: önem, engeller ve ülkemizde gerçekleştirilen projeler. *XVI. "Türkiye'de İnternet" Konferansı Bildirileri Kitapçığı*. 30 Kasım-2 Aralık 2011. İstanbul: Gamze yayıncılık.131-139.
- Shulman, L. (1987). Knowledge and teaching: Foundations of the new reform. *Harvard Educational Review*, 57(1), 1-23.
- Strauss, W., and Howe, N. (1991). Generations: The history of America's future, 1584 to 2069. *New York*.
- Sukamolson, S. (2010). Fundamentals of quantitative research. *Language Institute, Chulalongkorn University*.
- Şad, S. N. ve Nalçacı, Ö. İ. (2015). Öğretmen adaylarının eğitimde bilgi ve iletişim teknolojilerini kullanmaya ilişkin yeterlilik algıları. *Mersin Üniversitesi Eğitim Fakültesi Dergisi*, 11(1).
- Tapscott, D. (1999). Educating the net generation. *Educational leadership*, 56(5), 6-11.
- Tapscott, D. (2009). *Grown up digital* (Vol. 361). New York: McGraw Hill.
- Tatlı, Z., ve Akbulut, H. İ. (2017). Öğretmen Adaylarının Alanda Teknoloji Kullanımına Yönelik Yeterlilikleri. *Ege Eğitim Dergisi*, 18(1), 31-55.
- Thompson, A. D., and Mishra, P. (2007). Breaking news: TPCK becomes TPACK! *Journal of Computing in Teacher Education*, 24(2), 38, 64.
- Topu, F. B., ve Göktaş, Y. (2009). İlköğretim Bilişim Teknolojileri (BT) Öğretmenlerinin Aldıkları Eğitimle Görev Yaptıkları Okullarda Onlara Yüklenen Görev ve Sorumlulukların Karşılaştırılması, 3. *Uluslararası Bilgisayar ve Öğretim Teknolojileri Sempozyumu (bots2009)*, Trabzon, Türkiye.
- TTKB. (2006). "İlköğretim Okulu Haftalık Ders Çizelgesi ve İlköğretim Okulu Seçmeli Dersler" 26.08.1998 tarih ve 180 sayılı karar.

- TTKB. (2007). “İlköğretim Okulları Haftalık Ders Çizelgesi” 04.06.2007 tarih ve 111 sayılı karar.
- TTKB (2012). Ortaokul ve İmam hatip ortaokulu Bilişim teknolojileri ve yazılım dersi (5, 6, 7 ve 8. Sınıflar) öğretim programı.
- TTKB (2018). Bilişim teknolojileri ve yazılım dersi öğretim programı (ortaokul 5 ve 6. Sınıflar).
- Turan, Z., Yılmaz, R. M., Durdu, L. ve Göktaş, Y. (2016). Öğretim teknolojilerinin tarihsel gelişimi. (Editör: Kürşat Çağıltay, Yüksel Göktaş). *Öğretim Teknolojilerinin Temelleri: Teoriler, Araştırmalar, Eğilimler*. Ankara: Pegem Akademi, 25-40.
- Tüysüz, C. ve Çümen, V. (2016). Eba Ders Web Sitesine İlişkin Ortaokul Öğrencilerinin Görüşleri. *Uşak Üniversitesi Sosyal Bilimler Dergisi*, 9(27).
- Unal, S., and Ozturk, I. H. (2012). Barriers to ITC integration into teachers' classroom practices: Lessons from a case study on social studies teachers in Turkey. *World Applied Sciences Journal*, 18(7), 939-944.
- Wetzel, K., and Marshall, S. (2011). TPACK goes to sixth grade: Lessons from a middle school teacher in a high-technology-access classroom. *Journal of Digital Learning in Teacher Education*, 28(2), 73-81.
- Wiebe, J. H., and Taylor, H. G. (1997). What should teachers know about technology? A revised look at the ISTE foundations. *Journal of Computing in Teacher Education*, 13(4), 5-9.
- Wolcott, P., and Çağıltay, K. (2001). Telecommunications, Liberalization, and the Growth of the Internet in Turkey. *Information Society*, 17(2), 133-142.
- Yadin, A. (2011). Reducing the dropout rate in an introductory programming course. *ACM inroads*, 2(4), 71-76.

- Yardımcı, A., Dolanbay, C., Karadağ, L., Çubuk, M., Kuleyin, N., Gunizi, S., İşler, V., Biroğul, S., Taşçı, E., Niğdelioğlu, A. ve Antunok, E. (2012 Nisan). Akademisyenlerden hükümete fatih projesi raporu. *Bilişim Dergisi*, 144, 60-65.
- Yıldırım, A. ve Şimşek, H. (2006). Sosyal bilimlerde nitel araştırma yöntemleri. Seçkin yayıncılık.
- YÖK. (1998). Eğitim Fakültesi Öğretmen Yetiştirme Lisans Programları. Ankara.
- YÖK. (2007). Öğretmen Yetiştirme ve Eğitim Fakülteleri (1982-2007) (Öğretmenin Üniversitede Yetiştirilmesinin Değerlendirilmesi), s. 62, Ankara.
- Zaharija, G., Mladenović, S., and Boljat, I. (2013). Introducing basic programming concepts to elementary school children. *Procedia-social and behavioral sciences*, 106, 1576.
- Zhao, Y., and Frank, K. A. (2003). Factors affecting technology uses in schools: An ecological perspective. *American educational research journal*, 40(4), 807-840

7. EKLER


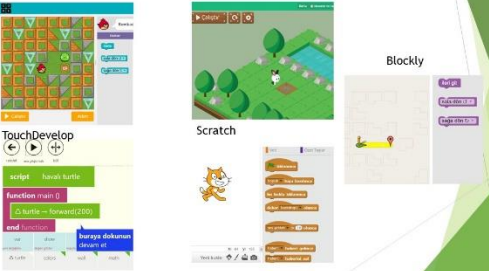

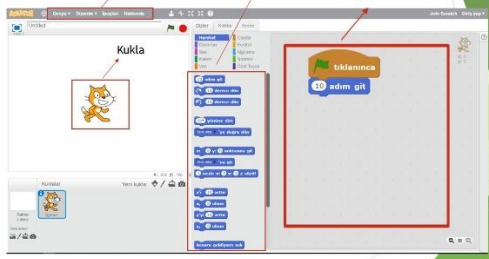
Ek-1: Günlük Ders Planları

1. Hafta	
Ders Adı:	Bilişim Teknolojileri ve Yazılım
Ünite Adı:	Problem Çözme ve Programlama
Süre:	40+40 Dakika
Kazanımlar:	
<p>6.5.2. Programlama</p> <p>6.5.2.1. Blok tabanlı programlama aracının ara yüzünü ve özelliklerini tanıtır. Scratch (scratch.eba.gov.tr), Kodadı2023, Code.org, Codecademy vb. programlama platformları kullanılabilir.</p> <p>6.5.2.2. Blok tabanlı programlama aracında sunulan bir programın işlevlerini açıklar.</p>	
Materyaller:	
<ul style="list-style-type: none"> • “Kodlamaya Giriş” videosu • “Blok Tabanlı Programlama” sunusu • Scratch • Blocky Oyunlar • “İlk Kodumu Yazıyorum” etkinliği • “Hem Kodluyorum Hem Oynuyorum” etkinliği 	
Yöntem ve Teknikler:	
<p>Öğretim Etkinlikleri Modeli (Gagne)</p> <p>Deney Grubu İçin:</p> <p>Akran Dönütü: Öğrenciler programlama ve projeleri hakkında birbirlerine geribildirim verir ve birbirlerinden geri bildirim alırlar (Carton, 2017).</p> <p>Problemleri Tanımla: Öğrenciler, bilgisayar ortamında çözülebilen gerçek dünya problemlerini tespit eder (Carton, 2017).</p> <p>Çeşitli Girdiler Araştır/ Girdileri Analiz Et: Öğrenciler diğer öğrencilerin çeşitli fikirlerini araştırır ve analiz eder (Carton, 2017)</p> <p>Kontrol Grubu İçin:</p> <p>Dönüt/Düzeltilme</p>	

Anlatım	
Tartışma	
Ders Akışı:	
Giriş (15 dk.)	
Sunum ve Uygulama (50 dk.)	
Değerlendirme (15 dk.)	
Dersin İşlenişi	
1. Giriş	
1.1. Dikkat	Öğrencilere çekme: https://www.touchdevelop.com/app/#list:vjsseqks/scripts:script:flspb:o verview web adresindeki önceden indirilmiş “Kodlamaya Giriş” videosu öğrencilere izlettirilir. Öğretmen videoyu izletirken gerekli yerlerde durdurur, öğrencilerin fikirlerini sorar ve videoda geçen önemli yerleri tekrarlar.
	
Şekil 1: “Kodlamaya Giriş” videosu ekran görüntüsü	
1.2. Kazanımlardan haberdar etme: Öğrencilerden kendi bilgisayarları üzerinden http://scratch.eba.gov.tr/ adresine girmeleri istenir. “Örnek Projeleri Görüntüle” linkine tıklayıp “Bilgisayar Parçaları” projesini incelemeleri istenir. Böyle bir projenin nasıl yapıldığı sorulur ve bunun Scratch gibi bir “Blok tabanlı Programlama” aracıyla kolay olduğu söylenir. Bugün “Blok tabanlı Programlama” konusuna giriş yapacakları söylenir.	
1.3. Ön öğrenmelerle ilişkilendirme: Geçen sene öğrendikleri algoritmayı bu şekilde kodlara çevirebilecekleri söylenir.	
2. Sunum ve Uygulama	

2.1. İçeriği Sunma ve Rehberlik Yapma: Blok Tabanlı Programlama sunusu öğrencilere izlettirilir ve anlatım yapılır.

 <h2 style="text-align: center;">BLOK TABANLI PROGRAMLAMA</h2> <p style="text-align: right;">Handan ATUN</p>	Slayt 1
<h3>Programlama Dilleri</h3> <ul style="list-style-type: none"> ▶ Programlama dili, insanların bilgisayara çeşitli işlemler yaptırmasına imkân veren her türlü sembol, karakter ve kurallar grubudur. ▶ Yani bilgisayarın anladığı dildir. ▶ Aynı her milletin kendine has dili (Türkçe, Fransızca, İngilizce vb.) olduğu gibi bilgisayarların da anlayabileceği birden çok programlama dili vardır. ▶ Bunlardan bazıları <ul style="list-style-type: none"> ▶ C++ ▶ Javascript ▶ Phyton ▶ Visual Basic 	Slayt 2
<h3>Programlama Dilleri</h3> <ul style="list-style-type: none"> ▶ Sadece yazıdan oluşur, uğraştırıcıdır, bütün kodları öğrenmesi zaman alır. <pre style="background-color: black; color: white; padding: 10px;"> <pre> </pre> </pre>	Slayt 3
<h3>Blok Tabanlı Programlama</h3> <ul style="list-style-type: none"> ▶ Programlamayı yeni öğrenenler için geliştirilmiş, kodların hazır bloklar halinde bulunduğu yapboz biçimli programlama türüdür. ▶ Öğrenmesi zevkli, açık ve anlaşlırdır. ▶ Çeşitli blok tabanlı programlama araçları vardır. ▶ Bunlardan bazıları: <ul style="list-style-type: none"> ▶ Kodris ▶ Code.org ▶ Scratch ▶ Touch Develop ▶ Blockly 	Slayt 4



<h3>Blok Tabanlı mı Normal Programlama mı?</h3> <p>Blok tabanlı kodlamada kodlar diğer programlama dillerine göre daha anlaşılır ve kolaydır.</p> <p>Code.org Javascript</p> 	Slayt 5
<h3>Blok Tabanlı Programlama Araçları</h3> <p>Code.org Kodris Blockly</p>  <p>Scratch</p> 	Slayt 6
<h3>Scratch</h3> <p>Menü Kod Bloklar Kod Yazma Bölümü</p> 	Slayt 7
<h3>Scratch</h3> <p>Programı Başlat-Programı Durdur</p> <p>Çoğalt-Sil-Büyüt-Küçült</p> <p>Kukla Kütüphanesi -Kukla Çiz-Kuklayı Bilgisayardan Seç-Kameradan Yeni Kukla</p> <p>tıklanınca Yeşil bayrağa tıkladığında bu blok altında sıralanmış kod kümesini çalıştırır.</p> <p>0 derece→Yukarı 180 derece→Aşağı 90 derece→Sağ -90 derece→Sol</p>	Slayt 8

Slayt 9

Neler Öğrendik

- ▶ Programlama dili nedir?
- ▶ Programlama dilleri nelerdir?
- ▶ Blok tabanlı programlama nedir?
- ▶ Blok tabanlı ve normal programlama arasındaki farklar nelerdir?
- ▶ Blok tabanlı programlama araçları nelerdir?
- ▶ Scratch programının arayüzü nasıl kullanılır?
- ▶ Scratch programında butonların görevleri nelerdir?
- ▶ Scratch programında yönler nelerdir?

2.2. Performansı ortaya çıkarma ve dönüt sağlama: “İlk Kodumu Yazıyorum” ve “Hem Kodluyorum Hem Oynuyorum” etkinlikleri yapılır. Deney Grubu öğrencilerinden scratch üzerinde hesap oluşturmaları istenir.

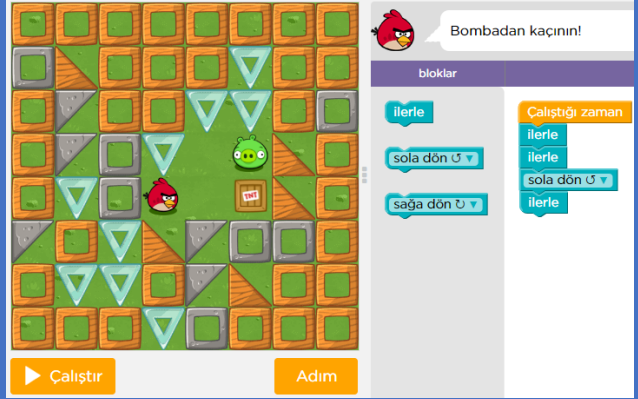
Etkinlik No	1
Etkinlik Adı	İlk Kodumu Yazıyorum
Süre	15 dk.
Yönergeler	<ol style="list-style-type: none"> 1. https://scratch.mit.edu/projects/editor/ web sayfasını açınız. 2. “Olaylar” kod sekmesinden  kod bloğunu kod yazma alanına sürükleyiniz. 3. “Hareket” kod sekmesinden 4 adet  kod bloğunu kod yazma alanına sürükleyiniz. 4. Kuklanın (Kedinin) sırasıyla yukarı, aşağı, sağa ve sola dönmesi için kod bloklarında gerekli düzenlemeleri yapınız. 5. Programı başlatınız ve yaptıklarınızı kontrol ediniz. 6. Doğruluğundan emin olunca yaptıklarınızı kaydediniz. 7. Arkadaşlarınızın yaptığı kayıtlı projelere göz atınız ve yorum yazınız (Akran Dönütü)
Etkinlik No	2
Etkinlik Adı	Hem Kodluyorum Hem Oynuyorum
Süre	15 dk.

Yönergeler	<ol style="list-style-type: none"> 1. http://kod.eba.gov.tr/tr/index.html web sayfasını açınız. 2. “Bulmaca” adlı kod oyununa tıklayınız 3. Her hayvanın (yeşil blok), resmini, ayak sayısını ve özelliklerini (mor blok) blokların uygun yerlerine takınız. 4. Hepsi bitince “Yanıtları Kontrol Et” e tıklayınız. 5. Deney Grubu: Bitiren arkadaşlarınızla bilgisayarda bunun gibi gerçek hayattan başka ne tür etkinlikler yapılabilir, tartışınız (Problemleri Tanımlama). Kontrol Grubu: Doğruluğundan emin olunca yaptıklarınızı öğretmeninize gösteriniz.
3. Değerlendirme	
<p>3.1. Performans değerlendirme:</p> <p>Deney Grubu: Öğrencilere birbirlerinin etkinliklerine göz atmaları ve değerlendirmeleri söylenir. Aynı zamanda bu değerlendirmeye okul dışı ortamda scratch hesaplarından devam etmeleri ve başkalarının fikirlerine göz atmaları istenir (Çeşitli girdiler araştır/analiz et).</p> <p>Kontrol Grubu: Öğretmen etkinlikleri değerlendirir.</p>	
<p>3.2. Öğrenilenlerin kalıcılığını sağlama ve transferi güçlendirme:</p> <p>Öğretmen “Blok Tabanlı Programlama” sunusunda “Neler Öğrendik” sayfasını açar. Bunlara ek olarak öğretmen Scratch ve Blockly araçlarından öğrencilere neler yaptıklarını sorar. Blok kavramının ve kod sırasının üzerinde durulur. Deney grubu öğrencilerinin tartışmaya dersin blog sayfası (bilisimdersimizkozanli.blogspot.com) üzerinden devam edebilecekleri söylenir.</p>	

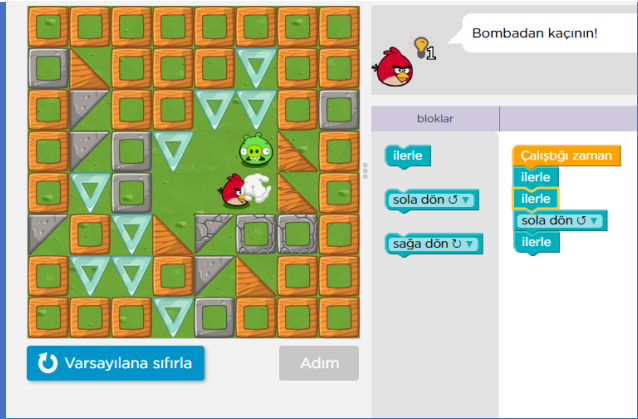
2. Hafta	
Ders Adı:	Bilişim Teknolojileri ve Yazılım
Ünite Adı:	Problem Çözme ve Programlama
Süre:	40+40 Dakika
Kazanımlar:	
6.5.2. Programlama	

<p>6.5.2.3. Blok tabanlı programlama aracında sunulan bir programın hatalarını ayıklar.</p> <p>6.5.2.4. Blok tabanlı programlama aracında sunulan bir programı verilen ölçütlere göre geliştirerek düzenler.</p> <p>6.5.2.5. Doğrusal mantık yapısını içeren programlar oluşturur.</p> <p>6.5.2.6. Doğrusal mantık yapısını içeren programları test ederek hatalarını ayıklar.</p>	
Materyaller:	
<ul style="list-style-type: none"> • Hatalı kod içeren şablon • Code.org • Blocky Oyunlar 	
Yöntem ve Teknikler:	
<p>Öğretim Etkinlikleri Modeli (Gagne)</p> <p>Deneysel Grubu:</p> <p>Hata Ayıklama / Sorun giderme: Öğrenciler bilgisayar sistemlerinde sorun giderir ve hesaplama yapılarındaki hataları sistematik olarak ayıklar (Carton, 2017).</p> <p>Artifaktları Test et: Öğrenciler, kriterler ve kısıtlamaların karşılanıp karşılanmadığını belirlemek için hesaplamalı eserleri sistematik olarak test eder (Carton, 2017).</p> <p>Kontrol Grubu:</p> <p>Anlatım</p> <p>Gösterip Yaptırma</p>	
Ders Akışı:	
<p>Giriş (15 dk.)</p> <p>Sunum ve Uygulama (50 dk.)</p> <p>Değerlendirme (15 dk.)</p>	
Dersin İşlenişi	
<p>1. Giriş</p> <p>1.1. Kazanımlardan haberdar etme ve ön öğrenmeleri hatırlatma: Öğrencilere önceki ders ne yaptıkları sorulur, sınıfta tartışma ortamı yaratılır. Tartışmaya her öğrencinin katılımı sağlanır. Önceki derste önemli kavramların anlamları sorulur: “Blok tabanlı programlama nedir?”, “Hangi araçlarda kullanılır?”, “Programlama dili nedir?” vb. Öğrencilere bugün code.org ve blockly’de yazılmış bazı kodların hatalarını bulacakları ve bu kodları verilene göre geliştirip düzenleyecekleri söylenir.</p>	
<p>2. Sunum ve Uygulama</p>	

2.1. İçeriği sunma ve rehberlik sağlama: Öğretmen akıllı tahtada code.org’u açar ve mantığının diğer blok tabanlı kodlama sayfalarıyla aynı olduğunu söyler. Daha sonra daha önceden yapmış olduğu hatalı kod içeren bir sayfa açar.



1. Öğretmen “Size yeşil domuzu yakalamak için yazılan bu kod doğru mu?” diye öğrencilere sorar.
2. Öğrencilerin fikirlerini belirtmesine izin verir. “Evet” ve “Hayır” diyenlerin fikirlerini ayrı ayrı alır.
3. Öğretmen “Çalıştır” butonuna basar.



1. Öğrenciler kuşun domuzla varmadan TNT’nin üstüne geldiğinde patladıklarını görürler.
2. Öğretmen öğrencilere kuşun hangi koddan sonra havaya uçtuğunu sorar.
3. Öğrencilerden alınan çeşitli cevaplardan sonra doğru cevap söylenir: 2. İlerle kodundan sonra.

4. Öğretmen 1 ilerle komutunun 1 kare ilerlettiği ve dönme komutlarında sadece dönüldüğü ilerleme olmadığı açıklamasını yapar.
5. Öğretmen bu kodu düzeltmek için ne yapmak gerektiğini öğrencilere sorar.




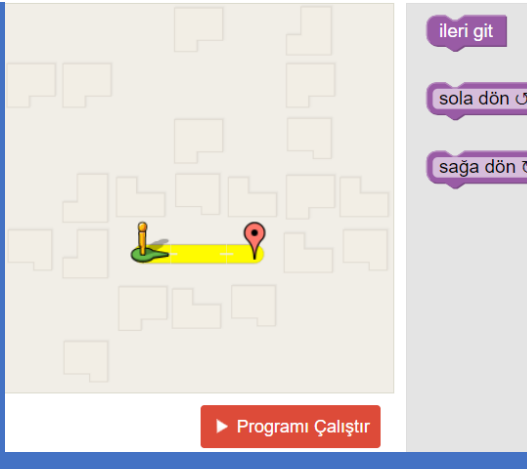
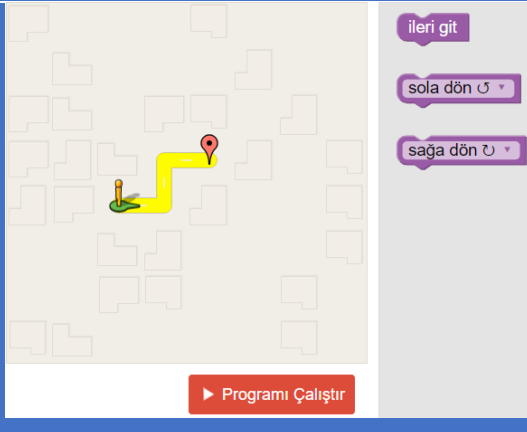
Öğretmen kodların çalışırken yazıldığı sırayı takip ettiklerini ve her kodun çalışma sırası geldiğinde yandığına dikkat etmelerini söyler.
Buna “doğrusal mantık yapısı” dendiğini söyler.

1.2. Performansı ortaya çıkarma ve dönüt sağlama

Deney Grubu öğrencilerinden code.org üzerinde hesap oluşturmaları istenir. “Kuşun Domuzu Yakalamasına Yardım Ediyorum” ve “Yönümü Buluyorum” etkinlikleri yapılır.

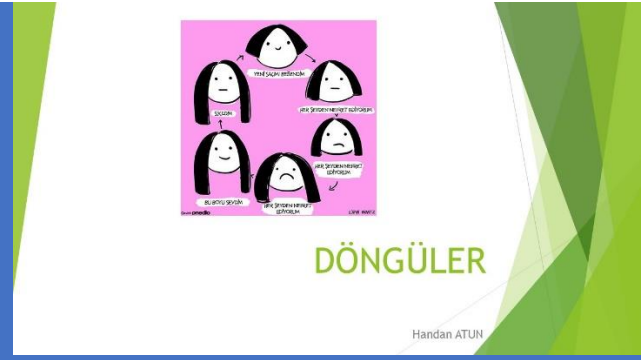
Etkinlik No	3
Etkinlik Adı	Kuşun Domuzu Yakalamasına Yardım Ediyorum
Süre	15 dk.
Yönergeler	<ol style="list-style-type: none"> 1. https://code.org/ sayfasını açınız. 2. “Öğrenciler” sekmesine basınız.

	<p>3. “Kurs 4” sekmesine basınız.</p> <p>4. Deney Grubu: Açılan sayfada 2. Etkinlik olan “Labirent ve Arı”nın 1 numaralı dersine tıklayınız. Öğretmeninizin yanlış yaptığı uygulamayı düzelterek yapınız (Hata ayıklama/Sorun Giderme).</p> <p>Kontrol Grubu: Kendiniz yapınız.</p>
	<p>5. 2. Etkinliği yapınız.</p>
	<p>6. Deney Grubu: Yaptıklarınızı hesabımıza kaydediniz.</p> <p>Kontrol Grubu: Yaptıklarınızı öğretmeninize gösteriniz.</p>
Etkinlik No	4
Etkinlik Adı	Yönümü Buluyorum

Süre	15 dk.
Yönergeler	<ol style="list-style-type: none"> 1. http://kod.eba.gov.tr/tr/index.html web sayfasını açınız. 2. “Labirent” adlı kod oyununa tıklayınız.
	<ol style="list-style-type: none"> 3. Sarı adamın kırmızı işaretli yere gitmesi için gereken kodu yazınız. Programı çalıştırınız. Deney Grubu: Yaptıklarınızın doğruluğunu test ediniz (Artifaktları Test Et).
	<ol style="list-style-type: none"> 4. 2. Etkinliği yapınız. Deney Grubu: Yaptıklarınızın doğruluğunuzu test ediniz (Artifaktları Test Et).
	<ol style="list-style-type: none"> 5. Deney Grubu: Yaptığımız 2 etkinliği de arkadaşlarınızinkilerle karşılaştırınız. Kontrol Grubu: Yaptıklarınızı öğretmeninize gösteriniz.
2. Değerlendirme	
3.3. Performans değerlendirme:	

<p>Deney Grubu: Öğrencilere birbirlerinin etkinliklerine göz atmaları ve değerlendirmeleri söylenir. Öğretmen de en son cevapları blog’da yayınlacağını söyler.</p> <p>Kontrol Grubu: Öğretmen yapılan etkinlikleri değerlendirir.</p>
<p>3.4. Öğrenilenlerin kalıcılığını sağlama ve transferi güçlendirme: Öğretmen Code.org’un Scratch’a ve Blockly’ye benzeyen yönlerini sorar. 3’ünü karşılaştırmalarını ister. Öğretmen kodlardaki hataları artık bulmaya başladıklarını ve istenen biçimde kod dizisi oluşturabildiklerini söyler. Dersi bitirir.</p>

3. Hafta	
Ders Adı:	Bilişim Teknolojileri ve Yazılım
Ünite Adı:	Problem Çözme ve Programlama
Süre:	40+40 Dakika
Kazanımlar:	
<p>6.5.2. Programlama</p> <p>6.5.2.11. Döngü yapısını içeren programlar oluşturur.</p> <p>6.5.2.12. Döngü yapısını içeren programları test ederek hatalarını ayıklar.</p>	
Materyaller:	
<ul style="list-style-type: none"> • “Döngüler” sunusu • “Dans Döngüsü” etkinliği • “Kuşun Domuzu Yakalamasına Yardım Ediyorum 2” etkinliği (Code.org) • “Döngüyle Yönümü Buluyorum” etkinliği (Blockly Oyunlar) 	
Yöntem ve Teknikler:	
<p>Öğretim Etkinlikleri Modeli (Gagne)</p> <p>Deney Grubu: Karmaşık Kodu Basitleştirme: Öğrenciler, herhangi bir farklılığı hesaba katmak için değişkenler kullanan tek bir bölüm için bir kodun bölümlerini değiştirirler (Carton, 2017). Problemleri Ayırıştır: Öğrenciler gerçek dünya sorunlarını daha yönetilebilir alt problemlere ayırır (Carton, 2017).</p> <p>Kontrol Grubu: Örnek verme Aksine örnek verme Anlatım</p>	
Ders Akışı:	
<p>Giriş (15 dk.) Sunum ve Uygulama (50 dk.)</p>	

Değerlendirme (15 dk.)	
Dersin İşlenişi	
1. Giriş	
<p>1.1. Dikkat çekme ve ön öğrenmeleri hatırlatma: Öğrencilere önceki ders ne yaptıkları sorulur, sınıfta tartışma ortamı yaratılır. Tartışmaya her öğrencinin katılımı sağlanır. Önceki derste kuşa verdikleri komutlar tartışılır. Öğretmen “Eğer domuz kuştan 1000 blok ötede olsaydı ne olurdu? 1000 kere ilerle mi yazardık?” diye sorar, fikirlerini alır ve bunun bir kısa yolu olduğunu ve adına da “döngü” dendiğini söyler.</p> <p>1.2. Kazanımlardan haberdar etme: Öğrencilere bugün döngüler konusunu işleyecekleri ve içeriğinde döngü bulunan kodlar yazacakları söylenir.</p>	
2. Sunum ve Uygulama	
2.1. İçeriği sunma: Öğretmen akıllı tahtada “Döngüler” sunusunu açar ve anlatım yapar.	
	
<p>Döngü Nedir?</p> <ul style="list-style-type: none"> ▶ Döngüler, günlük hayatta sürekli karşımıza çıkar. ▶ Acıktım → Yemek hazırlayım → Yemek yiyeyim → Bulasık yıkayım → Acıktım.... ▶ Kirlendim → Yıkayayım → Dışarı Çıkayım → Kirlendim.... ▶ Yazıldan düşük aldım → Moralim bozuk → Oyun oynayayım → Ders çalışmadım → Yazıldan düşük aldım.... ▶ Döngü: Bir veya birden fazla olayın belirli bir düzen dahilinde tekrarlanmasıdır. 	
<p>Slayt1</p> <p>Slayt2</p>	

<h3>Döngüler</h3> <ul style="list-style-type: none"> ► Aşağıdaki -Banyo Döngüsü- örneğini inceleyelim. ► Bu banyo döngüsünün günde 3 kere tekrarlandığını farz edelim. ► O zaman kodumuz nasıl olurdu? <div style="display: flex; align-items: center; justify-content: center;"> <div style="text-align: center;"> <p>Banyo Döngüsü</p> </div> <div style="margin-left: 20px;"> <p>Tekrarla(3)[kirlen, yıkan, dışarı çık]</p> </div> </div>	Slayt3
<h3>Etkinlik 1: Dans Döngüsü</h3> <p>Elinizdeki kağıtta dans hareketlerine bakarak bunları tekrarla komutunun içine kod olarak yazınız.</p> <p>Ben 3. döngüyü sizin için yaptım.</p> <p>Hangi hareketlerin kaç kere tekrarlandığına dikkat ediniz.</p>	Slayt4
<h3>Döngü Ne İşe Yarar?</h3> <ul style="list-style-type: none"> ► Programlama yaparken birçok yazı yazmak ya da birçok blok kullanmak gerekebilir. ► Döngü kullanmak bu yükü azaltır. ► Kod kalabalığından bizi kurtarır. ► Daha az blokla daha çok şey başarabiliriz. ► Aşağıdaki 2 kod bloğunun da görevi aynı. Biri döngüyle, biri döngüsüz yazılmış. Farkı görebildiniz mi? <div style="display: flex; align-items: center; justify-content: center;"> <div style="border: 1px solid black; padding: 5px; margin-right: 20px;"> <p>Çalıştır zamanı</p> <p>başla</p> <p>başla</p> <p>başla</p> <p>başla</p> <p>başla</p> </div> <div style="margin-right: 20px;">=</div> <div style="border: 1px solid black; padding: 5px;"> <p>Çalıştır zamanı</p> <p>bu işlemi 3 kez tekrarla</p> <p>başla</p> </div> </div>	Slayt5
<h3>Döngü Blokları</h3> <p>Döngü blokları içine aldıkları kodları parantez içindeki rakam kadar tekrarlatır.</p> <div style="display: flex; flex-wrap: wrap;"> <div style="width: 50%;"> <p>Code.org</p> </div> <div style="width: 50%;"> <p>Scratch</p> </div> <div style="width: 50%;"> <p>Blockly</p> </div> <div style="width: 50%;"> <p>TouchDevelop</p> </div> </div>	Slayt6

ETKİNLİK: DANS DÖNGÜSÜ

Tabloda dans hareketlerine bakarak bunları tekrarla komutunun içine kod olarak yazınız.

Ben 3. döngüyü sizin için yaptım.

Hangi hareketlerin kaç kere tekrarlandığına dikkat ediniz.

Dans figürlerinin anlamları aşağıda verilmiştir.

	Dans Figürleri						Cevap
1. Döngü							
2. Döngü							
3. Döngü						Tekrarla(3) [eller belde, kıvr]	
4. Döngü							

Dans Figürü					
Anlamı	Kollar önde	Eller arkada	Eller belde	Kollar çapraz	Kıvr

Etkinlik No	6
Etkinlik Adı	Kuşun Domuzu Yakalamasına Yardım Ediyorum 2
Süre	15 dk.
Yönergeler	<ol style="list-style-type: none"> https://code.org/ sayfasını açınız. “Öğrenciler” sekmesine basınız. “Kurs 2” sekmesine basınız. Açılan sayfada 6. Etkinlik olan “Labirent: Döngüler” in 1 numaralı dersine tıklayınız.

	<p>5. Deney Grubu: 1. Etkinliği döngü kullanmadan yapınız. Kontrol Grubu: 2. Etkinliği yapınız.</p>
	<p>6. Deney Grubu: 2. Etkinlik 1. Etkinliğin aynısıdır fakat bunu döngü kullanarak çözmeniz gerekiyor (Karmaşık Kodu Basitleştirme).</p>
	<p>7. 3. Etkinliği döngü kullanarak çözünüz.</p>
<p>Etkinlik No 7</p>	<p>Etkinlik Adı Döngüyle Yönümü Buluyorum</p>
<p>Süre 15 dk.</p>	<p>Yönergeler</p> <ol style="list-style-type: none"> 1. http://kod.eba.gov.tr/tr/index.html web sayfasını açınız. 2. “Labirent” adlı kod oyununa tıklayınız.

	<p>3. Deney Grubu: 4. Etkinliği “tekrar et “bloğu kullanarak çözüünüz. Adamın gittiği yolu bulurken ileri gidip sürekli dönme hareketlerini tekrar ederek zikzak çizdiğine dikkat ediniz. Hangi kodları hangi sırada tekrarladığına dikkat ediniz (Problemleri Ayırıştır).</p> <p>Kontrol Grubu: 3. Etkinliği tekrar et bloğu kullanarak çözüünüz.</p>
	
	<p>4. 5. Etkinliği yapınız.</p> <p>5. Adamın dönme hareketlerini tekrar etmediğine dikkat ediniz.</p>
	<p>6. Deney Grubu: Yaptıklarınızı arkadaşlarınızla paylaşınız.</p> <p>Kontrol Grubu: Yaptıklarınızı öğretmeninize gösteriniz.</p>
<p>3. Değerlendirme</p>	
<p>3.1. Performans değerlendirme:</p> <p>Deney Grubu: Öğrencilere birbirlerinin etkinliklerine göz atar ve değerlendirir, yorum yazarlar.</p>	

Kontrol Grubu: Öğretmen etkinlikleri değerlendirir.
3.2. Öğrenilenlerin kalıcılığını sağlama ve transferi güçlendirme: Öğretmen döngülerle ilgili öğrencilere fikirlerini sorar. Sevdikleri ya da sevmedikleri yanlarını belirtmelerini söyler. Deney grubu hayattan döngü örneklerini dersin bloğunda paylaşır.

4. Hafta	
Ders Adı:	Bilişim Teknolojileri ve Yazılım
Ünite Adı:	Problem Çözme ve Programlama
Süre:	40+40 Dakika
Kazanımlar:	
6.5.2. Programlama	
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.	
Materyaller:	
<ul style="list-style-type: none"> • “Koşullandırıcılar” sunusu • “Arının İşini Kolaylaştırıyorum” etkinliği (Code.org) • “Kediyi Yönlendiriyorum” etkinliği (Scratch) 	
Yöntem ve Teknikler:	
Öğretim Etkinlikleri Modeli (Gagne)	
Deney Grubu:	
Hata Ayıklama/Sorun Giderme: Öğrenciler bilgisayar sistemlerinde sorun giderir ve kod dizinlerindeki hataları sistematik olarak ayıklar (Carton, 2017).	
Atrifaktları Test Etme: Öğrenciler, kriterler ve kısıtlamaların karşılanıp karşılanmadığını belirlemek için hesaplamalı eserleri sistematik olarak test eder (Carton, 2017).	
Kontrol Grubu:	
Beyin Fırtınası	
Anlatım	
Soru-Cevap	
Ders Akışı:	
Giriş (15 dk.)	
Sunum ve Uygulama (50 dk.)	
Değerlendirme (15 dk.)	
Dersin İşlenişi	
1. Giriş	








1.1. Dikkat çekme ve ön öğrenmeleri hatırlatma: Öğrencilere önceki ders ne yaptıkları sorulur, sınıfta tartışma ortamı yaratılır. Tartışmaya her öğrencinin katılımı sağlanır. Önceki derste kodları döngüye koyarak kısalttıklarına dikkat çekilir. Öğretmen “Eğer kuşun domuz varsa yakalaması yiyecek varsa yemesi gerektiğini yazmak isteseydik ne yazardık?” diye sorar, fikirlerini alır ve bunun için bir komut olduğunu ve adına da “koşullandırıcı” ya da “karar yapısı” dendiğini söyler.

1.2. Kazanımlardan haberdar etme: Öğrencilere bugün koşullandırıcılar konusunu işleyecekleri ve karar yapısı içeren kodlar yazacakları söylenir.

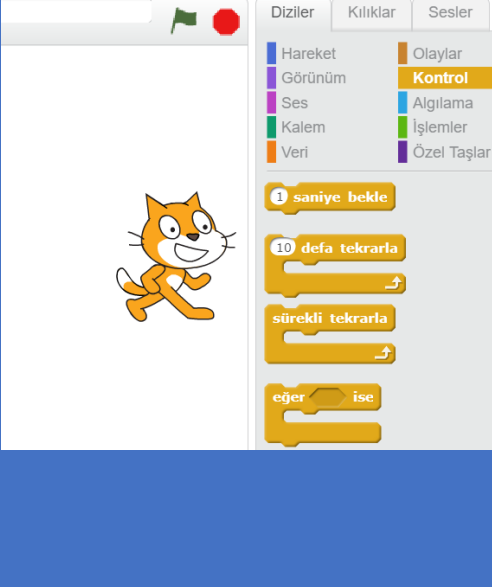
2. Sunum ve Uygulama

2.1. İçeriği sunma: “Koşullandırıcılar” sunusu izlettirilir ve anlatım yapılır.

 <p style="text-align: center;">KOŞULLANDIRICILAR (KARAR YAPISI)</p> <p style="text-align: right;">Handan ATUN</p>	Slayt 1
<p>Koşullar (EĞER)</p> <ul style="list-style-type: none"> ► Koşulları da döngüler gibi, günlük hayatta sürekli kullanırız. <ul style="list-style-type: none"> ► Hava sıcaksa → Dışarı Çıkayım Değilse → Evde oturayım ► Hafta içiyse → İşe gideyim ► Yazın → Tatile çıkayım ► Koşul: Bir veya birden fazla olayın gerçekleşmesi için gereken durum.  <p style="text-align: center;">Eğer (Grip Olursan) [Antibiyotik Kullan]</p>	Slayt 2
<p>Karar Yapısı</p> <ul style="list-style-type: none"> ► Koşullar sadece Eğer bloğuyla olabileceği gibi Eğer ve Değilse/Yoksa bloğuyla da olur. Buna karar yapısı denir. Yani durumun ne olduğuna karar veriyoruz. ► Aşağıdaki -Padişah- örneğini inceleyelim. ► Karar → Padişah ben miyim, sen mi?  <p style="text-align: center;">Eğer (Padişah=sen) [Kendi ordunun başına geç.] Eğer (Padişah=ben) [Ordunun başına geç.]</p> <p style="text-align: center;">=</p> <p style="text-align: center;">Eğer (Padişah=sen) [Kendi ordunun başına geç.] Değilse/Yoksa [Ordunun başına geç.]</p>	Slayt 3

<h3>Eğer Blokları</h3> <p>Eğer blokları içine aldıkları kodları -eğer- in yanındaki durum gerçekleşirse çalıştırır. Eğer durum gerçekleşmezse kod çalışmaz.</p> <p>Code.org</p>  <p>Scratch</p>  <p>Blockly</p>  <p>TouchDevelop</p> 	Slayt 4
<h3>Karşılaştırım</h3> <p>Döngü</p> <ul style="list-style-type: none"> ▶ Anahtar kelime=Tekrarla ▶ İçine aldığı kodu yazılan sayı kadar tekrarlar.  <p>Karar Yapısı</p> <ul style="list-style-type: none"> ▶ Anahtar kelime=Eğer ▶ İçine aldığı kodu üstteki durum gerçekleşirse çalıştırır. ▶ -Değilse- adında bir koşullandırıcı daha vardır. O da üstteki durum gerçekleşmezse çalıştırır. 	Slayt 5
<h3>Etkinlik: Arının İşini Kolaylaştırıyorum</h3> <ul style="list-style-type: none"> ▶ Etkinliği bilgisayarınızda yapınız. ▶ Dikkat hem döngü hem de karar yapısını kullanacağız. 	Slayt 6
<h3>Nerde Hata Yaptım?</h3> 	Slayt 7
<p>2.2. Performansı ortaya çıkarma ve dönüt sağlama: “Arının İşini Kolaylaştırıyorum” ve “Kediyi Yönlendiriyorum” etkinlikleri yapılır.</p>	
Etkinlik No	8
Etkinlik Adı	Arının İşini Kolaylaştırıyorum
Süre	20 dk.

<h2>Yönergeler</h2>	<ol style="list-style-type: none"> 1. https://code.org/ sayfasını açınız. 2. “Öğrenciler” sekmesine basınız. 3. “Kurs 4” sekmesine basınız. 4. Açılan sayfada 2. Etkinlik olan “Labirent ve Arı”nın 7 numaralı dersine tıklayınız.
	<ol style="list-style-type: none"> 5. Etkinliği yapınız. Önce tekrarlar bloğuyla ilerleyin sonra eğer bloğuyla bulutun altında çiçek varsa nektar toplayın.
	<ol style="list-style-type: none"> 6. Deney Grubu: 8 numaralı etkinliği öğretmeniniz yapmıştır. Öğretmeninizin hatasını düzelterek kodu doğru hale getirin (Hata Ayıklama/Sorun Giderme). Kontrol Grubu: 8 numaralı etkinliği yapınız.

	<p>7. 9 numaralı etkinliği iç içe eğer ve döngü bloklarını kullanarak etkinliği yapınız.</p>
<p>Etkinlik No 9</p>	
<p>Etkinlik Adı</p>	<p>Kediyi Yönlendiriyorum</p>
<p>Süre</p>	<p>15 dk.</p>
<p>Yönergeler</p>	<ol style="list-style-type: none"> 1. https://scratch.mit.edu/projects/editor/ web sayfasını açınız. 2. Kedinin yeşil bayrağa tıklanınca yukarı ok tuşuna basılıyken 10 adım gitmesini yoksa hmm diye düşünmesini sağlayan kodu yazınız.
	<ol style="list-style-type: none"> 3. Tıklanınca bloğu → Olaylar Eğer bloğu → Kontrol Yukarı ok basılı mı bloğu → Algılama Hmm düşünme bloğu → Görünüm 4. Deney Grubu: Kodunuzu test ediniz. Bütün kodları sürekli tekrarla bloğu içine koyup tekrar test ediniz (Artifaktları Test Etme). 5. Kontrol Grubu: Kodunuzu çalıştırınız.
	<ol style="list-style-type: none"> 6. Deney Grubu: Yaptıklarınızı proje olarak kaydediniz.

Kontrol Grubu: Yaptıklarınızı öğretmeninize gösteriniz.
3. Değerlendirme
<p>3.1. Performans değerlendirme:</p> <p>Deney Grubu: Öğrencilere birbirlerinin etkinliklerine göz atar ve yorum yazarlar.</p> <p>Kontrol Grubu: Öğretmen etkinlikleri değerlendirir.</p>
<p>3.2. Öğrenilenlerin kalıcılığını sağlama ve transferi güçlendirme:</p> <p>Deney Grubu: Öğretmen öğrencilerden karar ve döngü yapısı içeren kodlar kullanarak scratch, code.org ya da blockly üzerinde çalışma yapmalarını ve yaptıkları çalışmanın ekran görüntüsünü dersin blog'una eklemelerini ister.</p> <p>Kontrol Grubu: Öğretmen Karar Yapısı ve Döngüler le ilgili öğrencilere neler öğrendiklerini sorar. Öğrencilerden kendi cümleleriyle açıklamaları istenir. Öğretmen “Arının işini kolaylaştırıyorum” etkinliğinde bulutun içinde çiçek olmasa ne olurdu diye sorar. Öğrencilerden alınan yanıtlarla konu toparlanır.</p>

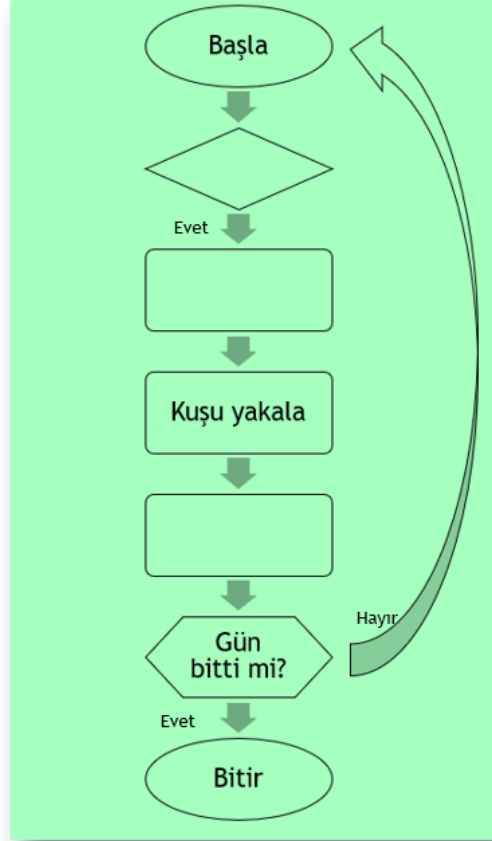
5. Hafta	
Ders Adı:	Bilişim Teknolojileri ve Yazılım
Ünite Adı:	Problem Çözme ve Programlama
Süre:	40+40 Dakika
Kazanımlar:	
<p>6.5.2. Programlama</p> <p>6.5.2.13. Bir algoritmayı uyarlamak için en uygun karar yapılarını seçer.</p> <p>6.5.2.14. Farklı programlama yapılarını kullanarak karmaşık problemlere çözüm üretir.</p>	
Materyaller:	
<ul style="list-style-type: none"> • “Programlamaya Giriş” sunusu • “Algoritmayı Hatırlıyorum” etkinliği • “Algoritmamı Kod Bloklarına Çeviriyorum” etkinliği 	
Yöntem ve Teknikler:	
<p>Öğretim Etkinlikleri Modeli (Gagne)</p> <p>Deney Grubu:</p> <p>Problemleri Değerlendir: Öğrenciler, hesaplanabilir şekilde çözülebileceklerini belirlemek için problemleri değerlendirir (Carton, 2017).</p>	

<p>Problemleri Tartış: Öğrenciler bir problemin hesaplamalı bir yaklaşımla çözülme kabiliyeti hakkında açıklayıcı sorular sorup tartışırlar (Carton, 2017).</p> <p>Kontrol Grubu: Soru-Cevap Anlatım Tartışma</p>	
Ders Akışı:	
<p>Giriş (30 dk.) Sunum ve Uygulama (45 dk.) Değerlendirme (5 dk.)</p>	
Dersin İşlenişi	
<p>1. Giriş</p> <p>1.1. Dikkat çekme ve ön öğrenmeleri hatırlatma: Öğrencilere geçen yıldan algoritma hakkında ne hatırladıkları sorulur ve “Algoritmayı hatırlıyorum” etkinliği yapılır.</p> <p>Deney Grubu: “Algoritmayı hatırlıyorum” etkinliğindeki algoritmayı code.org’ta kullanıp kullanamayacakları hakkında öğrencilerden fikir alınır (Problemleri Değerlendir).</p>	
Etkinlik No	10
Etkinlik Adı	Algoritmayı Hatırlıyorum
Süre	15 dk.

ETKİNLİK: ALGORİTMAYI HATIRLIYORUM

Aşağıdaki hikâyenin algoritmasını akış şemasına yerleştiriniz.

“Kafesteki kuş, kafesin kapısı açılırsa kaçıyor, kaçınca perdeye konuyormuş. Perdeye konan kuşu sahibi yakalayıp kafese koyuyor ve aynı şeyler gün bitene kadar devam ediyormuş.”



1.2. Kazanımlardan haberdar etme: Bugün yazılı bir algoritmayı önce bloklu kodlara sonra yazılı kodlara dönüştürmeyi öğrenecekleri söylenir ve “Algoritmamı Kod Bloklarına Çeviriyorum” etkinliği yapılır.

Deney Grubu: Öğretmen önce sadece algoritma kısmını gösterir ve öğrencilerden bu algoritmanın code.org’a uyarlanıp uyarlanmayacağı hakkında fikir alır (Problemleri Değerlendir).

Etkinlik No	11		
Etkinlik Adı	Algoritmamı	Kod	Bloklarına
	Çeviriyorum		
Süre	20 dk.		

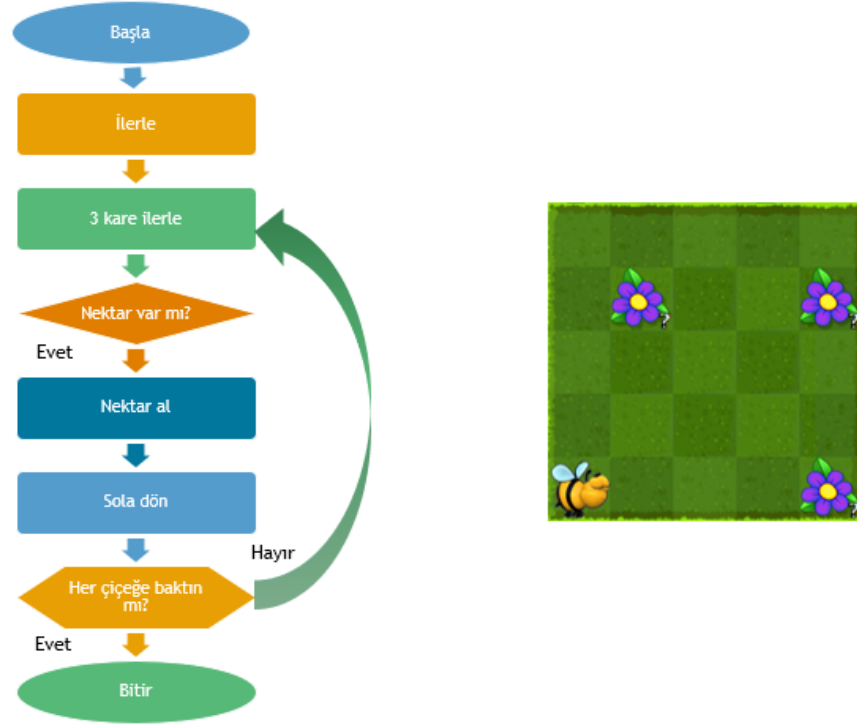
ETKİNLİK: ALGORİTMAMI KOD BLOKLARINA ÇEVİRİYORUM

Aşağıdaki algoritmayı verilen kod bloklarını kullanarak bloklukod yapısına çevirip code.org. Kurs-2 ders-3 etkinlik-8'e uyarlayınız.

Kodunuzu çalıştırıp kontrol ediniz.


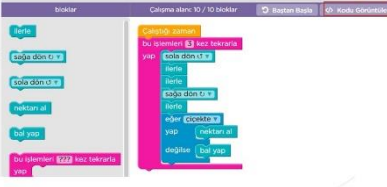
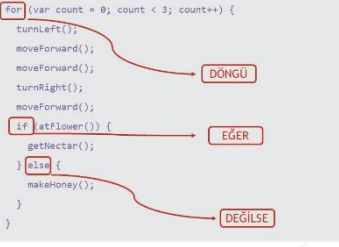
Kod Blokları

Döngü/ karar	bu işlemleri ??? kez tekrarla yap	eğer nektar = 1 yap	
Eylem	ilerle	sola dön	nektarı al



2. Sunum ve Uygulama

2.1. İçeriği sunma: "Programlamaya Giriş" sunusu izlettirilir ve anlatım yapılır.

 <h2 style="text-align: center;">PROGRAMLAMAYA GİRİŞ</h2> <p style="text-align: right;">Handan ATUN</p>	Slayt 1
<h3>Blok Tabanlı Programlamada Saklı Kodlar</h3> <ul style="list-style-type: none"> ► Bugüne kadar kodlarımızı bloklar yardımıyla yazdık. ► Ancak blok tabanlı programlama araçları arka planında yazılı kodlar saklar. ► Bunun için "kodu görüntüle" yazan yere tıklamalıyız. 	Slayt 2 : Burada öğrencilere önceden code.org'ta yapmış oldukları etkinliklerin yazılı kod kısmına bakmaları sağlanır. Öğrenciler kendileri araştırır.
<h3>Blok Tabanlı Programlamada Saklı Kodlar</h3> <ul style="list-style-type: none"> ► JavaScript dilindeki kod görüntülenir: 	Slayt 3
<h3>FOR Döngüsü</h3> <ul style="list-style-type: none"> ► FOR döngüsünün genel yapısı: <p>► FOR (i=0 ; i<[] ; i++)</p> <p style="text-align: center;">0'dan başla Buradan yazan Arttır Sayıya kadar</p> <p>► { }</p> <p>Tekrar etmesi gereken kod (döngü içindeki kod) süslü parantez içine yazılır.</p>	Slayt 4

<p>FOR Döngüsü Örneği</p> <ul style="list-style-type: none"> ▶ 4 kare ilerlemek için (ilerle=moveForward()); ▶ FOR (i=0; i<4; i++) ▶ {moveForward()} ▶ Her 1 kare ilerlediğinde sağa dönmek ve bunu 3 kere tekrarlamak için (sağadön=turnRight()); ▶ FOR (i=0; i<3; i++) ▶ {moveForward(); ▶ turnRight();} <p>1'den fazla kod tekrar ediyorsa araya noktalı virgül ; konur</p>	<p>Slayt 5: Öğrenciler code.org kurs 2 – arı:döngüler’de daha önceden yazdıkları kodları incelerler.</p>
<p>IF ELSE Koşullandırması</p> <ul style="list-style-type: none"> ▶ IF Else genel yapısı: ▶ IF (.....) Parantez içine gerçekleşmesi gereken durum yazılır. ▶ {.....} Süslü parantez içine üstteki durum gerçekleşirse olacak eylem yazılır. ▶ ELSE ▶ {.....} Süslü parantez içine en üstteki durum gerçekleşmezse olacak eylem yazılır. 	<p>Slayt 6</p>
<p>IF ELSE Örneği</p> <ul style="list-style-type: none"> ▶ Eğer çiçeğe geldiyse nektar al, peteğe geldiyse bal yap: ▶ Nektar al=getNectar ; Çiçek= flower ; petek= honeyComb; bal yap= makeHoney <pre> IF (flower) {getNectar} ya da IF (honeyComb) {makeHoney} ELSE {makeHoney} ELSE {getNectar} </pre>	<p>Slayt 7: Öğrenciler code.org kurs 2 – arı:koşullandırıcılar’da daha önceden yazdıkları kodları incelerler.</p>

3. Değerlendirme

3.1. Öğrenilenlerin kalıcılığını sağlama ve transferi güçlendirme:

Deney Grubu: Öğretmen öğrencileri 3-4 kişilik gruplara ayırır. Öğrencilerden grup olarak scratch, code.org ya da blockly üzerinde belirledikleri bir etkinliği algoritmaya çevirmelerini ister (Problemleri Tartış).

Kontrol Grubu: Öğretmen yazılı kodlarla ilgili öğrencilere fikirlerini sorar. Gelecek derste kod yazacakları ve artık bloklardan kurtulacaklarını söyler. Evde bu kodları araştırmalarının onlara hazırlıklı gelmelerini kolaylaştıracağını belirtir.

6. Hafta

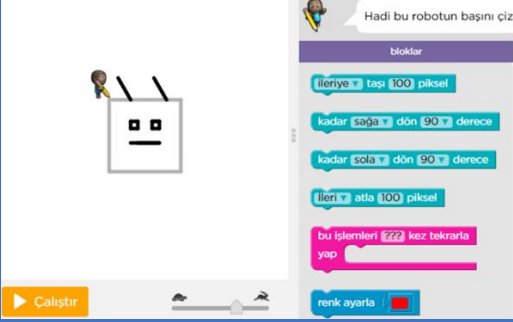
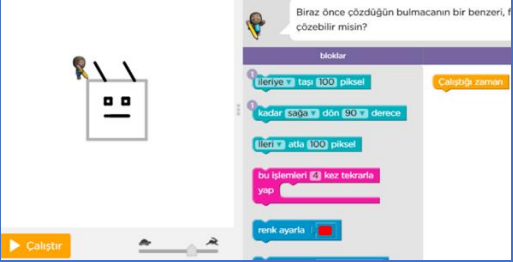

Ders Adı:	Bilişim Teknolojileri ve Yazılım
Ünite Adı:	Problem Çözme ve Programlama


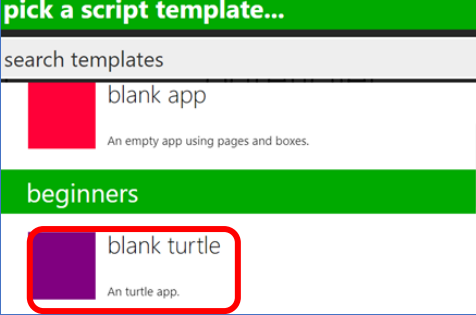
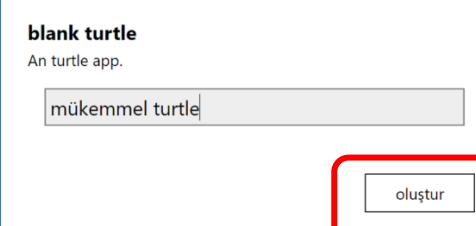
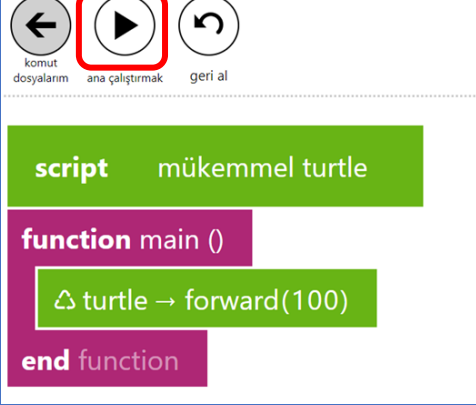
Süre:	40+40 Dakika
Kazanımlar:	
<p>6.5.2. Programlama</p> <p>6.5.2.5. Doğrusal mantık yapısını içeren programlar oluşturur.</p> <p>6.5.2.9. Çoklu karar yapıları içeren programlar oluşturur.</p> <p>6.5.2.11. Döngü yapısını içeren programlar oluşturur.</p> <p>6.5.2.14. Farklı programlama yapılarını kullanarak karmaşık problemlere çözüm üretir.</p>	
Materyaller:	
<ul style="list-style-type: none"> • “Programlamaya Giriş” sunusu • “Kodla Çizim Yapıyorum” etkinliği (Code.org) • “Kaplumbağa ile Kare Çiziyorum” etkinliği (Touch Develop) • “Bilgisayarım Benimle Konuşuyor” etkinliği (Microsoft Small Basic) 	
Yöntem ve Teknikler:	
<p>Öğretim Etkinlikleri Modeli (Gagne)</p> <p>Deney Grubu:</p> <p>Artifakt Geliştir: Öğrenciler problemleri çözmek, kendilerini ifade etmek veya görevleri tamamlamak için hesaplamalı eserler oluştururlar (Carton, 2017).</p> <p>Modüller Tasarla / Oluştur: Öğrenciler etkileşim modülleri ve soyutlama sistemleri tasarlar ve oluşturur (Carton, 2017).</p> <p>Kontrol Grubu:</p> <p>Gösterip Yaptırma</p> <p>Anlatım</p> <p>Tartışma</p>	
Ders Akışı:	
<p>Giriş (30 dk.)</p> <p>Sunum ve Uygulama (45 dk.)</p> <p>Değerlendirme (5 dk.)</p>	
Dersin İşlenişi	
<p>1. Giriş</p> <p>1.1. Dikkat çekme ve ön öğrenmeleri hatırlatma: Öğrencilere bir önceki dersten ne hatırladıkları sorulur. Öğrenciler “Programlamaya Giriş” sunusunda 7. Slayda kadar kısaca göz gezdirirler.</p> <p>1.2. Kazanımlardan haberdar etme: Öğrencilere bu derste artık yazılı kodlamaya geçecekleri ve karmaşık problemlere çözüm üretecekleri söylenir.</p>	
<p>2. Sunum ve Uygulama</p>	

2.1. İçeri Sunma: “Programlamaya Giriş” sunusu 8. Slayttan itibaren izlettirilir ve anlatım yapılır.

<h3>Kodla Çizim Yapıyorum</h3> <ul style="list-style-type: none"> Hazır blok kodlardan ayrılmış yazılı kodlara geçmeden önce code.org üzerinden bir alıştırmaya daha yapalım. 	Slayt 8												
<h3>Kaplumbağa ile Kare Çiziyorum</h3> <ul style="list-style-type: none"> Touch Develop'ta, “Kaplumbağa ile Kare Çiziyorum” etkinliği yapılır. Bu etkinliği hazır kod bloklarını kullanmadan yapacağız ancak kodları tamamen yazıyla yazmamız da gerekmiyor. Bu etkinlik -Blok tabanlı kodlama-dan -Yazılı kodlama-ya geçiş etkinliği olarak görülebilir. 	Slayt 9												
<h3>Microsoft Small Basic’le Birlikte Kod Yazalım</h3> <ul style="list-style-type: none"> Bu etkinliği birlikte yapacağız. Bilgisayarımızın yazı ekranına bilgisayar saatine göre iyi günler ya da iyi geceler yazmasını sağlayacağız. 1. Masaüstündeki Microsoft Small Basic simgesine tıklayalım. Burada 2 değişik kod göreceğiz: THEN ve ENDIF IF (Clock.Hour>18) THEN → Alta yazan satırı yap TextWindow.WriteLine(“iyi geceler”) ELSE TextWindow.WriteLine(“iyi günler”) ENDIF → Eğer Koşulu burada bitiyor 	Slayt 10												
<h3>Aşağıdaki kodu yazınız ve çalıştırınız:</h3> <table border="0"> <thead> <tr> <th>KOD</th> <th>ANLAMI</th> </tr> </thead> <tbody> <tr> <td>IF (Clock.Hour>18) THEN</td> <td>Eğer saat 18'den büyükse</td> </tr> <tr> <td>TextWindow.WriteLine(“iyi geceler”)</td> <td>Yazı ekranına iyi geceler yaz.</td> </tr> <tr> <td>ELSE</td> <td>Saat 18'den büyük değilse</td> </tr> <tr> <td>TextWindow.WriteLine(“iyi günler”)</td> <td>Yazı ekranına iyi günler yaz.</td> </tr> <tr> <td>ENDIF</td> <td>Kod burada bitiyor.</td> </tr> </tbody> </table>	KOD	ANLAMI	IF (Clock.Hour>18) THEN	Eğer saat 18'den büyükse	TextWindow.WriteLine(“iyi geceler”)	Yazı ekranına iyi geceler yaz.	ELSE	Saat 18'den büyük değilse	TextWindow.WriteLine(“iyi günler”)	Yazı ekranına iyi günler yaz.	ENDIF	Kod burada bitiyor.	Slayt 11
KOD	ANLAMI												
IF (Clock.Hour>18) THEN	Eğer saat 18'den büyükse												
TextWindow.WriteLine(“iyi geceler”)	Yazı ekranına iyi geceler yaz.												
ELSE	Saat 18'den büyük değilse												
TextWindow.WriteLine(“iyi günler”)	Yazı ekranına iyi günler yaz.												
ENDIF	Kod burada bitiyor.												

<p>Ekranında Ne Yazıyor?</p> <ul style="list-style-type: none"> ▶ Kodu yazdıktan sonra çalıştırma basıp ne olacağını görün. ▶ Kodunuz doğru çalışıyor mu? ▶ Bilgisayarınız size ne dedi? ▶ Bu koda göre bilgisayarınız ekrana saat kaçla kaç arası -iyi günler-, kaçla kaç arası -iyi geceler- yazar? 	Slayt 12
<p>Bilgisayarım Benimle Konuşuyor</p> <ul style="list-style-type: none"> ▶ Microsoft Small Basic'te -Bilgisayarım Benimle Konuşuyor- etkinliğini kendiniz yapınız. ▶ Burada fazladan ELSEIF komutu kullanacağız. ▶ Yapısı IF komutuyla aynıdır. ▶ Ancak ELSEIF 2'den fazla durum varsa kullanılır. ▶ Sabahları günaydın, öğleleri tünaydın, akşamları iyi akşamlar, geceleri iyi geceler demek gibi. (4 durum) 	Slayt 13
<p>Neler Öğrendik?</p> <ul style="list-style-type: none"> ▶ Programlama ▶ For Döngüsü ▶ If Else Koşullandırması ▶ Code.org-Kodla Şekil Çizimi ▶ TouchDevelop- Kodla Şekil Çizimi ▶ Microsoft Small Basic- Metin Penceresine Yazı Yazdırma 	Slayt 14
<p>2.2. Performansı ortaya çıkarma ve dönüt sağlama: “Kodla Çizim Yapıyorum”, “Kaplumbağa ile Kare Çiziyorum” ve “Bilgisayarım Benimle Konuşuyor” etkinlikleri yapılır.</p>	
<p>Etkinlik No</p>	12
<p>Etkinlik Adı</p>	Kodla Çizim Yapıyorum
<p>Süre</p>	15 dk.
<p>Yönergeler</p>	<ol style="list-style-type: none"> 1. https://code.org/ sayfasını açınız. 2. “Öğrenciler” sekmesine basınız. 3. “Kurs 2” sekmesine basınız.

	8. Açılan sayfada 7. Etkinlik olan “Sanatçı: Döngüler” in 1 numaralı dersine tıklayınız.
	9. Sanatçının kare çizmesini sağlayan kodu yazın.
	<p>10. 2. Etkinliği yapınız.</p> <p>11. 2. etkinlik 1. Etkinliğin aynısıdır. Fakat bu sefer döngü yapmak zorundasınız.</p> <p>12. Tekrarla bloğunun içine hangi kodları koyacağımıza dikkat edin.</p>
	<p>13. Deney Grubu: Yaptıklarınızı proje olarak kaydediniz (Artifakt Geliştir).</p> <p>Kontrol Grubu: Yaptıklarınızı öğretmeninize gösteriniz.</p>
<p>Etkinlik No</p>	13
<p>Etkinlik Adı</p>	Kaplumbağa ile Kare Çiziyorum
<p>Süre</p>	20 dk.
<p>Yönergeler</p>	<p>1. https://www.touchdevelop.com/ web sayfasını açınız.</p>
	<p>2. Launch Touch Develop'a tıklayınız.</p>

 <p>komut dosyalarım</p> <p>Tüm komut dosyalarım</p> <p>doğüstü turtle Ben</p> <p>Komut Dosyası Oluştur</p> <p>havalı turtle Ben</p> <p>3 ★ in 27 öğretici öğrenme durumu</p> <p>çarpıcı turtle Ben</p> <p>acayip bird Ben</p>	<p>3. Komut Dosyası oluştur'a tıklayınız.</p>
 <p>pick a script template...</p> <p>search templates</p> <p>blank app</p> <p>An empty app using pages and boxes.</p> <p>beginners</p> <p>blank turtle</p> <p>An turtle app.</p>	<p>4. Beginners bölümünden blank turtle' a tıklayınız.</p>
 <p>blank turtle</p> <p>An turtle app.</p> <p>mükemmel turtle</p> <p>oluştur</p>	<p>5. Çalışmanıza istediğiniz ismi vererek oluştur'a tıklayınız.</p>
 <p>← komut dosyalarım ana çalıştırmak geri al</p> <p>script mükemmel turtle</p> <p>function main ()</p> <p>turtle → forward(100)</p> <p>end function</p>	<p>6. Açılan sayfadaki kodu bu haliyle çalıştırıp ne olduğunu görün.</p> <p>7. Kodla çizim yapıyorum etkinliğinde yaptığımız kodu buraya uyarlayarak kaplumbağanın kare çizmesini sağlayın.</p>

	<p>8. Kod eklemek için turtle <input type="checkbox"/> forward(100) bloğuna tıklayınız. + işaretine tıklayınız.</p>
	<p>9. Altta açılan pencerede turtle 1 seçiniz</p>
	<p>1. Sonra Kaplumbağanın doğru yöne dönmesini sağlayan kodu seçiniz. left turn= sola dön Right turn=sağa dön</p>
	<p>2. Kareyi for döngüsüyle çizdirmeyi unutmayınız. (+ ya tıklayınız ve açılan penceren for'u seçiniz)</p>
	<p>3. Aynı diğer blok tabanlı kodlama araçları gibi kod blokları taşınabilir ve for bloğunun içine konulabilir. 4. For bloğundaki sayılara dikkat edin. Tekrar sayısı buna bağlı. 5. Deney Grubu: Yaptıklarınızı proje olarak kaydediniz ve</p>

	<p>farklı kareler çizmeyi deneyiniz (Artifakt Geliştir).</p> <p>Kontrol Grubu: Yaptıklarınızı öğretmeninize gösteriniz.</p>
--	--

Etkinlik No	14
Etkinlik Adı	Bilgisayarım Benimle Konuşuyor
Süre	20 dk.

ETKİNLİK: BİLGİSAYARIM BENİMLE KONUŞUYOR

Bilgisayarınızın yazı ekranına
Saat 17'den sonra "iyi akşamlar",
Saat 12 ile 17 arası "iyi öğlenler" ve
Saat 0 ile 12 arası "günaydın" yazmasını sağlayınız.



Kullanılacak komutlar:

Komut	Anlamı
<code>TextWindow.WriteLine</code>	Ekrana yazı yaz
<code>Clock.Hour</code>	Bilgisayar Saati
<code>IF</code>	Eğer
<code>THEN</code>	Yap
<code>ELSEIF</code>	Eğer Ya da
<code>ENDIF</code>	Eğer'in Sonu

Deney Grubu: Komutları yazdığımızda yaptıklarımızı bilgisayarınıza dosya olarak kaydediniz ve kod düzenini bozmayacak değişiklikler yapınız (Modüller Tasarla/Oluştur).

3. Değerlendirme

3.1. Performans değerlendirme: Öğrencilere birbirlerinin etkinliklerine göz atmaları ve değerlendirmeleri söylenir.

3.2. Öğrenilenlerin kalıcılığını sağlama ve transferi güçlendirme:

Deney Grubu: Dersin sonunda öğrencilerden istedikleri programlama ortamını kullanarak karar ve döngü yapıları içeren bireysel ya da grup çalışması yapmaları istenir.

Kontrol Grubu: Öğretmen sununun “Neler Öğrendik” sayfasını açar ve bugünün sonrasında son 6 haftanın özetini yapar. Soru-Cevap ve Tartışma tekniğiyle ders sona erer.



Ek-2: Bilgisayarca Düşünme Beceri Düzeyleri Ölçeği (Ortaokul Düzeyi İçin)

	Hiçbir Zaman	Nadiren	Bazen	Genellikle	Her zaman
1. Kararlarının çoğundan emin olan insanları severim	1	2	3	4	5
2. Yeni bir durumla karşılaştığımda ortaya çıkabilecek sorunları çözebileceğime inancım vardır.	1	2	3	4	5
3. Bir sorunumu çözmek üzere plan yaparken o planı yürütebileceğime güvenirim.	1	2	3	4	5
4. Bir sorunla karşılaştığımda, başka konuya geçmeden önce durur ve o sorun üzerinde düşünürüm.	1	2	3	4	5
5. Bir problemin çözümünü verecek denklemi hemen kurabilirim.	1	2	3	4	5
6. Matematiksel sembol ve kavramlar yardımıyla yapılan anlatımları daha kolay öğrendiğimi düşünürüm	1	2	3	4	5
7. Sayılar arasındaki ilişkileri kolaylıkla yakalayabildiğime inanırım.	1	2	3	4	5
8. Sözel olarak ifade edilen bir matematik problemini sayısallaştırabilirim.	1	2	3	4	5
9. Grup arkadaşlarımla birlikte işbirlikli öğrenme deneyimleri yaşamaktan hoşlanırım.	1	2	3	4	5
10. İşbirlikli öğrenmede, grupla çalıştığım için daha başarılı sonuçlar elde ettiğimi/edeceğimi düşünüyorum.	1	2	3	4	5
11. İşbirlikli öğrenmede grup arkadaşlarımla birlikte grup projesi ile ilgili problemleri çözmekten hoşlanırım.	1	2	3	4	5
12. İşbirlikli öğrenmede daha çok fikir ortaya çıkıyor.	1	2	3	4	5
13. Karmaşık problemlerin çözümüne yönelik düzenli planlar geliştirmede iyiyimdir.	1	2	3	4	5
14. Karmaşık problemleri çözmeye çalışmak eğlencelidir.	1	2	3	4	5
15. Zorlayıcı şeyler öğrenmeye istekliyimdir.	1	2	3	4	5
16. Elimdeki seçenekleri karşılaştırırken ve karar verirken kullandığım sistematik bir yöntem vardır.	1	2	3	4	5
17. Problemin çözümünü zihnimde canlandırma konusunda sıkıntı yaşarım.	1	2	3	4	5
18. Problem çözümünde X, Y gibi değişkenleri nerede ve nasıl kullanmam gerektiği konusunda sıkıntı yaşarım.	1	2	3	4	5
19. Tasarladığım çözüm yollarını sırasıyla aşamalı bir şekilde uygulayamam.	1	2	3	4	5
20. Bir soruna yönelik olası çözüm yollarını düşünürken çok fazla seçenek üretemem.	1	2	3	4	5
21. İşbirlikli öğrenme ortamında kendi düşüncelerimi geliştiremem.	1	2	3	4	5
22. İşbirlikli öğrenme grup arkadaşlarıma bir şeyler öğretmeye çalışmak beni yoruyor.	1	2	3	4	5


Ek-3: İlköğretim Düzeyindeki Çocuklar İçin Problem Çözme Envanteri (ÇPÇE)



	Hiçbir Zama	Nadir en	Bazen	Genellikle	Her zaman
1. Sorunlarımdan kaçma yerine sorunumu çözmeye çalışırım.	1	2	3	4	5
2. Ne zaman sorun yaşasam içimde hep bir karamsarlık olur ve kendimi kolay kolay toplayamam.	1	2	3	4	5
3. Karşıma sorunlar çıktığında sakin olmaya çalışırım.	1	2	3	4	5
4. Kafama bir şeyler takıldığında sinirli olurum ve istemediğim sözler söylerim.	1	2	3	4	5
5. Yaşadığım problemlerin herkesin başına gelebileceğine inanırım.	1	2	3	4	5
6. Başıma bir problem geldiğinde çabucak üzülürüm.	1	2	3	4	5
7. Sorun yaşadığımda onu çözmek için bulduğum çözüm yolu işe yarayana kadar vazgeçmem.	1	2	3	4	5
8. Sorun yaşadığımda uzun süre etkisinden kurtulamam.	1	2	3	4	5
9. Sorunlarım olduğunda hep kendi kendime sorular sorarım ve çözüm yolları ararım.	1	2	3	4	5
10. Sorunlarımı çözemediğim zaman her şeyden soğurum.	1	2	3	4	5
11. Karşılaştığım sorunlardan kurtulmak için vazgeçmeden bütün çözüm yollarımı denerim.	1	2	3	4	5
12. Sorun yaşadığımda kendimi kolay kolay derse veremem.	1	2	3	4	5
13. Öncelikle sorunlarımın neden kaynaklandığını bulmaya çalışırım.	1	2	3	4	5
14. Arkadaşlarımla sorun yaşadığımda konuşmak yerine kavga ederim.	1	2	3	4	5
15. Sorunlardan kaçmak yerine işe yarayan bir çözüm yolu bulana kadar uğraşırım.	1	2	3	4	5
16. İş ve sorumluluklarımdan kaçmak için birçok bahane uydururum.	1	2	3	4	5
17. Sorunlar karşısında oldukça sabırlı ve kararlı davranırım.	1	2	3	4	5
18. Bir sorunum olduğunda ne yaparsam yapayım çözülmeyeceğini düşünürüm.	1	2	3	4	5
19. Sorunlarımı çözemediğimde zamanlarda ailemden ya da arkadaşlarımdan yardım isterim.	1	2	3	4	5
20. Sorunlarımı çözme konusunda genellikle başarılı değilimdir.	1	2	3	4	5
21. Sorunlarım karşısında genellikle yaratıcı ve etkili çözüm yolları bulurum.	1	2	3	4	5
22. Sorunlarım olduğunda küçük çocuk gibi davranmak beni rahatlatır.	1	2	3	4	5
23. Bir sorunla karşılaştığımda tüm çözüm yollarını düşünerek çözeceğime inanırım.	1	2	3	4	5
24. Bir sorunum olduğunda çözüm yolları aramak yerine her şeyi oluruna bırakırım.	1	2	3	4	5



Ek-4: Programlama Başarı Testi ve Cevap Anahtarı

1) Aşağıdakilerden hangisi blok tabanlı programlama aracı değildir?
 A) Scratch B) Code.org
 C) Kodris D) C++


2) Kuşun TNT'ye çarpmadan domuzu yakalayabilmesi için aşağıdakilerden hangi kod bloğunu kullanması doğru olur?







A)  B) 

C)  D) 


3) Kuşun TNT'ye çarpmadan domuzu yakalayabilmesi için aşağıdakilerden hangi kod bloğunu kullanması doğru olur?







A)  B) 

C)  D) 

4) Kuşun TNT'ye çarpmadan domuzu yakalayabilmesi için aşağıdakilerden hangi kod bloğunu kullanması doğru olur?



A)  B) 

C)  D) 

5,6 ve 7. Soruları aşağıdaki cümlelere göre cevaplayınız.

I. Zıpızp adlı kurbağa hava güneşliyse gölde yüzer, diğer durumlarda yiyecek bulmaya gider.
 II. Zıpızp adlı kurbağamız her gün dikdörtgen şeklinde bir patikayı kullanarak işe gitmektedir.
 III. Zıpızp eve geldiği zaman yorgunsa dinlenir, değilse; yaz mevsiminde tarlaya, kış mevsiminde ormana gider.

5) I. Cümle için bir program yazacak olursak aşağıdaki yapılardan hangisini kullanmamız gerekir?
 A) Döngü yapısı B) Çoklu Karar Yapısı
 C) Karar yapısı D) Doğrusal Mantık Yapısı

6) II. Cümle için bir program yazacak olursak aşağıdaki yapılardan hangisini kullanmamız gerekir?
 A) Döngü yapısı B) Çoklu Karar Yapısı
 C) Karar yapısı D) Doğrusal Mantık Yapısı

7) III. Cümle için bir program yazacak olursak aşağıdaki yapılardan hangisini kullanmamız gerekir?

- A) Döngü B) Çoklu Karar Yapısı
yapısı
C) Karar D) Doğrusal Mantık Yapısı
yapısı

8) Aşağıdaki arının çiçeklerin hepsinden nektar toplaması için aşağıdaki kod dizisinin hangisi yazılmalıdır? [İlerle =forward();Nektar toplama=getNectar()]

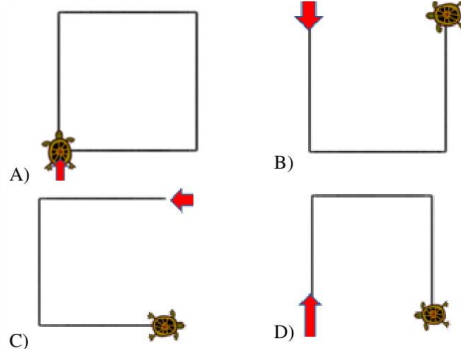
- A) for(var i=0; i<6; i++){
getNectar();forward();}
B) for(var i=0; i<6;
i++){forward(); getNectar();}
C) for(var i=0; i<5; i++){
getNectar();forward();}
D) for(var i=0; i<5;
i++){forward(); getNectar();}



```
for(var i=0; i<3;
i++){
forward();turnRight
();}
```

9) Yukarıdaki kaplumbağa yere boya sürerek ilerlemektedir. Yukarıda yazılan koda göre kaplumbağanın çizdiği şekil aşağıdakilerden hangisinde doğru olarak verilmiştir?

[forward()=İlerle; turnRight()=Sağa dön] Not: Kaplumbağanın başladığı nokta kırmızı okla gösterilmiştir.



10) Resimdeki arının bulutun altında çiçek varsa nektar toplamasını, bal varsa petek yapmasını sağlayan kod dizisi aşağıdakilerden hangisidir?



A)



C)

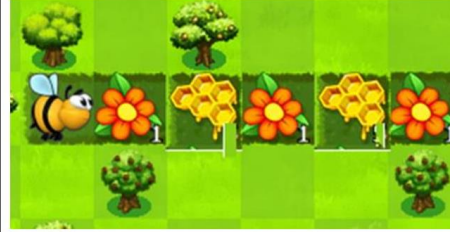


B)



D)

11) Aşağıdaki arının çiçeğe geldiğinde nektar almasını peteğe geldiğinde bal yapmasını sağlayan kod dizisi aşağıdakilerden hangisidir?



İlerle=forward(); çiçek=flower(); nektar al=getNectar();
balyap=makeHoney() petek=honeyComb()

- A) for(var i=0; i<5;
i++){forward();
if (flower()) {
makeHoney();
} else {
getNectar();}
B) for(var i=0; i<5;
i++){forward();
if (flower()) {
getNectar();
} else {
makeHoney ();}
C) for(var i=0; i<5;
i++){forward();
if (honeyComb ()) {
getNectar ();
} else {
makeHoney();}
D) for(var i=0; i<5; i++){
if (flower()) {
getNectar ();
} else {
makeHoney();}
forward();}

```

IF (Clock.hour<13) THEN
Textwindow.writeline("İYİ GÜNLER")
ELSE IF (Clock.hour<17) THEN
Textwindow.writeline("İYİ ÖĞLENLER")
ELSE
Textwindow.writeline("İYİ AKŞAMLAR")
ENDIF

```

12) Yukarıdaki kod dizisi Microsoft Small Basic programında yazıldığında aşağıdaki durumlardan hangisi oluşmaz?

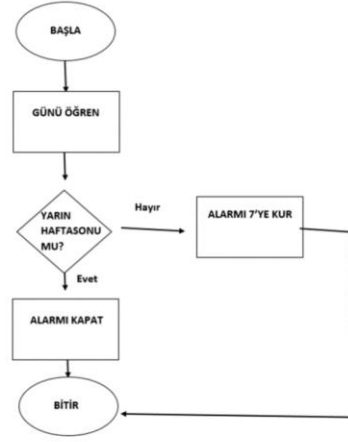
- A) Saat 17'den önceki durumlarda ekranda "İYİ ÖĞLENLER" yazısı çıkar.
 B) Saat 17'den sonraki durumlarda ekranda "İYİ AKŞAMLAR" yazısı çıkar.
 C) Saat 17'den önceki ve 13'ten sonraki durumlarda ekranda "İYİ ÖĞLENLER" yazısı çıkar.
 D) Saat 13'ten önceki durumlarda ekranda "İYİ GÜNLER" yazısı çıkar.

13) Aşağıdaki kodda bulunan hatayı düzeltmek için seçeneklerden hangisi yapılmalıdır?



- A) "İlerle" ve "nektarı" al komutları yer değiştirmeli.
 B) Kodlar "sola dön" değil "sağa dön" le başlamalı.
 C) Nektarı al işlemi "2" yerine "3" kez tekrarlanmalı.
 D) Tüm işlemler "3" yerine "4" kez tekrarlanmalı.

14) Aşağıdaki algoritmanın koda çevrilmesi seçeneklerden hangisinde doğru verilmiştir?



Tomorrow_weekend= yarın haftasonu

Setalarm= alarmı ayarla

Tomorrow_weekday=yarın haftaici

Closealarm=alarmı kapat

- A)

```
if (tomorrow_weekend())
{setAlarm(7); }
else { closeAlarm(); }
```

 B)

```
for (tomorrow_weekend())
{setAlarm(7); }
for (tomorrow_weekday)
{closeAlarm(); }
```

 C)

```
if (tomorrow_weekend()
{closeAlarm(); }
else {setAlarm(7); }
```

 D)

```
if (tomorrow_weekend()
{closeAlarm(); }
if {setAlarm(7); }
```

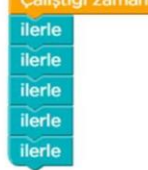
15) Aşağıdaki döngü yapısıyla yazılmış olan kodun doğrusal yapıya dönüştürülmüş hali aşağıdaki seçeneklerden hangisinde doğru olarak verilmiştir?





A)



B)




C) 

D) 

16) Yanda verilen Scratch programının simgelerinin görevleri hangi seçenekte doğru olarak verilmiştir?


A) Çoğalt-Kes- Karakter büyüt -Karakter küçült
B) Sil-Karakter büyüt-Çoğalt-Karakter Küçült
C) Çoğalt-Sil-Karakter büyüt- Karakter küçült
D) Çoğalt-Karakter küçült-Kes-Karakter büyüt

17) Yandaki kod bloğunun açıklaması aşağıdakilerden hangisidir?




A) Klavyede sağ yön tuşuna basıldığında sağa doğru dön ve 10 adım git.
B) Klavyede boşluk tuşuna basıldığında sağa doğru dön ve 10 adım git.
C) Klavyede aşağı yön tuşuna basıldığında aşağı doğru dön ve 10 adım git.
D) Klavyede yukarı yön tuşuna basıldığında sağa dön ve 10 adım git.

18) Yandaki blok ne işe yarar?




A) Karar yapısı oluşturmaya
B) Döngü yapısı oluşturmaya
C) Doğrusal yapı oluşturmaya
D) Algoritma yapısı oluşturmaya

19) Yandaki blok ne işe yarar?



A) Karar yapısı oluşturmaya
B) Döngü yapısı oluşturmaya
C) Doğrusal yapı oluşturmaya
D) Algoritma yapısı oluşturmaya

20) Aşağıdaki koddaki hatayı düzeltmek için seçeneklerden hangisi yapılmalıdır?





A) "Sağa dön" komutu yerine "sola dön" yazılmalı
B) "Sola dön" ve "sağa dön" komutları yer değiştirmeli.
C) "Sola dön" komutu yerine "sağa dön" yazılmalı.
D) Sondaki "ilerle" komutu silinmeli.


21) Yanda verilen Scratch programının simgelerinin görevleri hangi seçenekte doğru olarak verilmiştir?


A) Küçük ekran-Tam ekran
B) Yenile- Tekrarla
C) Programa Giriş -Programdan çıkış
D) Programı başlat- Programı durdur

22) Scratch programında karakterin sırasıyla yukarı, aşağı, sağa ve sola dönmesini sağlayan kod dizisi aşağıdakilerden hangisidir?










23) Yandaki Scratch komutunun görevi aşağıdakilerden hangisidir?



A) Komut bloklarını tekrarlatır.
B) Komutları istenilen süre kadar bekletir.
C) Bütün kod kümelerini durdurur.
D) Yeşil bayrağa tıkladığında bu blok altında sıralanmış kod kümesini çalıştırır.

24) Legolara benzer bir şekilde, içe içe geçen ve arka arkaya sıralanabilen yapılardan oluşan programlamayı kolaylaştırmak için tasarlanmış araçlara ne ad verilir?

A) Nesne tabanlı programlama araçları
B) Blok tabanlı programlama araçları

- C) Doğrusal tabanlı programlama araçları
D) Algoritma tabanlı programlama araçları
25) “İşlemler tek yönlü olarak sırayla ilerler, üstte yazılan kod bilgisayar tarafından alttakinden önce okunur.” Bu açıklama aşağıdaki terimlerden hangisine aittir?
A) Dairesel Mantık Yapısı
B) Sıralı Mantık Yapısı
C) Doğrusal Mantık Yapısı
D) Döngüsel Mantık Yapısı

CEVAP ANAHTARI

- 1) D
- 2) C
- 3) A
- 4) B
- 5) C
- 6) A
- 7) B
- 8) D
- 9) D
- 10) A
- 11) B
- 12) A
- 13) A
- 14) C
- 15) B
- 16) C
- 17) A
- 18) A
- 19) B
- 20) B
- 21) D
- 22) C
- 23) D
- 24) B
- 25) C



T.C.
NECMETTİN ERBAKAN ÜNİVERSİTESİ
Eğitim Bilimleri Enstitüsü Müdürlüğü

Özgeçmiş

Adı Soyadı:	Handan ATUN
Doğum Yeri:	Polatlı
Doğum Tarihi:	16.07.1989
Medeni Durumu:	Bekar

Öğrenim Durumu

Derece	Okulun Adı	Program	Yer	Yıl
İlköğretim	Mustafa Necati İÖO		Kırıkkale	1995-2003
Ortaöğretim	Kırıkkale Fen Lisesi	Sayısal	Kırıkkale	2003-2004
	Özel Kızılırmak Lisesi	Sayısal	Kırıkkale	2004-2007
Lisans	ODTÜ	BÖTE	Ankara	2007-2013
İş Deneyimi:	MEB, Öğretmen			2014-Devam ediyor
İlgi Alanları:	TPAB (TPACK), Bilgisayar Bilimleri, Web 2.0, Eğitim			
Tel:	05348316023			
Adres:	Camikebir Mah. Ankara Cad. No: 2078/5 Kulu/Konya			
E-posta:	handanatun@gmail.com			