

**T.C.**  
**NECMETTİN ERBAKAN ÜNİVERSİTESİ**  
**EĞİTİM BİLİMLERİ ENSTİTÜSÜ**  
**EĞİTİM BİLİMLERİ ANABİLİM DALI**  
**EĞİTİM PROGRAMI VE ÖĞRETİM BİLİM DALI**

**6.SINIF ÖĞRENCİLERİNE PROGRAMLAMA DİLİ**  
**ÖĞRETİLİRKEN KULLANILAN SCRATCH**  
**PROGRAMININ ÖĞRENCİLERİN PROBLEM ÇÖZME**  
**BECERİLERİNE VE TUTUMLARINA ETKİSİ**

**Rukiye Berna BALA**

**YÜKSEK LİSANS TEZİ**

**Danışman**

**Prof. Dr. Füsun Gülderen ALACAPINAR**

**Konya-2019**



**T.C.**  
**NECMETTİN ERBAKAN ÜNİVERSİTESİ**  
**EĞİTİM BİLİMLERİ ENSTİTÜSÜ**  
**EĞİTİM BİLİMLERİ ANABİLİM DALI**  
**EĞİTİM PROGRAMI VE ÖĞRETİM BİLİM DALI**

**6.SINIF ÖĞRENCİLERİNE PROGRAMLAMA DİLİ**  
**ÖĞRETİLİRKEN KULLANILAN SCRATCH**  
**PROGRAMININ ÖĞRENCİLERİN PROBLEM ÇÖZME**  
**BECERİLERİNE VE TUTUMLARINA ETKİSİ**

**Rukiye Berna BALA**

**YÜKSEK LİSANS TEZİ**

**Danışman**

**Prof. Dr. Füsun Gülderen ALACAPINAR**

**Konya-2019**

**BİLİMSEL ETİK SAYFASI**

<b>Öğrencinin</b>	Adı Soyadı	Rukiye Berna BALA
	Numarası	148301031005
	Ana Bilim Dalı	Eğitim Bilimleri Ana Bilim Dalı
	Bilim Dalı	Eğitim Programı ve Öğretim Bilim Dalı
	Programı	Tezli Yüksek Lisans
	Tezin Adı	6.Sınıf Öğrencilerine Programlama Dili Öğretilirken Kullanılan Scratch Programının Öğrencilerin Problem Çözme Becerilerine ve Tutumlarına Etkisi

Bu tezin proje safhasından sonuçlanmasına kadarki bütün süreçlerde 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.

22/02/2019

*Rukiye Berna BALA*  
**Öğrencinin**  
**Adı Soyadı İmzası**  
**Rukiye Berna BALA**

 <b>KONYA</b>	<b>T.C.</b> <b>NECMETTİN ERBAKAN ÜNİVERSİTESİ</b> <b>Eğitim Bilimleri Enstitüsü Müdürlüğü</b>	 <b>NECMETTİN ERBAKAN</b> <b>ÜNİVERSİTESİ</b> <b>EĞİTİM BİLİMLERİ</b> <b>ENSTİTÜSÜ</b>
---	---	---

### YÜKSEK LİSANS TEZİ KABUL FORMU

<b>Öğrencinin</b>	Adı Soyadı	Rukiye Berna BALA
	Numarası	148301031005
	Ana Bilim Dalı	Eğitim Bilimleri Anabilim Dalı
	Bilim Dalı	Eğitim Programı Ve Öğretim Bilim Dalı
	Programı	Tezli Yüksek Lisans
	Tez Danışmanı	Prof. Dr. Füsün Gülderen ALACAPINAR
	Tezin Adı	6.Sınıf Öğrencilerine Programlama Dili Öğretilirken Kullanılan Scratch Programının Öğrencilerin Problem Çözme Becerilerine Ve Tutumlarına Etkisi

Yukarıda adı geçen öğrenci tarafından hazırlanan 6.Sınıf Öğrencilerine Programlama Dili Öğretilirken Kullanılan Scratch Programının Öğrencilerin Problem Çözme Becerilerine Ve Tutumlarına Etkisi başlıklı bu çalışma 08/02/2019 tarihinde yapılan savunma sınavı sonucunda oybirliği/oyçokluğu ile başarılı bulunarak, jürimiz tarafından yüksek lisans tezi olarak kabul edilmiştir.

	Ünvanı Adı Soyadı	İmza
Danışman	Prof. Dr. Füsün Gülderen ALACAPINAR	
Jüri Üyesi	Dr. Öğr. Üyesi Süleyman Alpaslan SULAK	
Jüri Üyesi	Dr. Öğr. Üyesi Gülçin ZEYBEK	

## TEŞEKKÜR

Yüksek lisans eğitimim boyunca ve tezimin her aşamasında bilgisini ve birikimini paylaşan, desteğini veren, zamanını ayıran, danışmanım Sayın Prof. Dr. Füsun Gülderen ALACAPINAR hocama teşekkür ederim. Süreçte yardımcı olan, görüşlerinden yararlandığım Sayın Dr. Öğr. Üyesi Gülçin ZEYBEK, Sayın Dr. Öğr. Üyesi Şemseddin GÜNDÜZ ve Sayın Dr. Öğr. Üyesi Güngör KESKİNKILIÇ YUMUŞAK hocalarıma teşekkür ederim.

Sürecin her aşamasında fikirlerini ve desteklerini benden esirgemeyen, yol arkadaşlarım Ayşe YENİEL ve Büşra ERİM' e teşekkürlerimi iletirim. Tezimin yazım aşamasında adeta bir anne şefkatiyle yanımda olan, beni destekleyen güzel arkadaşım Sedef APAYDIN' a ayrıca teşekkür ederim.

Tabi ki her zaman her durumda yanımda olan geniş ailemin hakkını ödeyemem. Onlara ne kadar teşekkür etsem azdır. Özellikle tezimin en yorucu ve zorlayıcı anlarında annemin verdiği destek ve motivasyon sayesinde bitirdiğimi de eklemek isterim. Anne CAN' ır.

Rukiye Berna BALA

Konya, 2019



**T. C.**  
**NECMETTİN ERBAKAN ÜNİVERSİTESİ**  
**Eğitim Bilimleri Enstitüsü Müdürlüğü**



**Öğrencinin**

Adı Soyadı	Rukiye Berna BALA
Numarası	148301031005
Ana Bilim / Bilim Dalı	Eğitim Bilimleri Anabilim Dalı Eğitim Programı ve Öğretim Bilim Dalı
Program	Tezli Yüksek Lisans <input checked="" type="checkbox"/> Doktora <input type="checkbox"/>
Tez Danışmanı	Prof. Dr. Füsün Gülderen ALACAPINAR
Tezin Adı	6.Sınıf Öğrencilerine Programlama Dili Öğretilirken Kullanılan Scratch Programının Öğrencilerin Problem Çözme Becerilerine ve Tutumlarına Etkisi

### ÖZET

Bu çalışmanın amacı, Scratch programı ile verilen programlama eğitiminin öğrencilerin; başarı testinden aldıkları puanlara, problem çözme becerisi ölçeğinden aldıkları puanlara ve derse yönelik tutum ölçeğinden aldıkları puanlara etkisinin olup olmadığını incelemektir. Ayrıca gerçekleştirilen uygulamayla ilgili öğrencilerin görüşleri de alınmıştır. Çalışmaya Konya ilinde bulunan bir devlet okulunda 6.sınıfta öğrenim görmekte olan 22 öğrenci katılmıştır. Çalışmada ön test-son test tek gruplu deneysel desen kullanılmıştır. Çalışma 8 hafta sürmüştür. Veri toplama aracı olarak; başarı testi, problem çözme ölçeği, bilişim teknolojileri dersine yönelik tutum ölçeği ve öğrencilerin görüşlerini almak için görüşme formu kullanılmıştır.

Araştırmadan elde edilen bulgulara göre, öğrencilerin başarı testinden aldıkları ön test ve son test puanları arasında, son test lehine anlamlı bir fark bulunmuştur.

Problem çözüme ölçeğinden alınan ön test ve son test puanları arasında anlamlı bir fark olmadığı belirlenmiştir. Bu sonuca göre Scratch programının öğrencilerin problem çözüme becerisi üzerinde istatistiksel olarak anlamlı bir etkisi olmadığı söylenebilir. Bilişim Teknolojileri dersine yönelik tutum ölçeğinden alınan ön test ve son test puanları arasında, son test puanları lehine anlamlı bir fark olduğu bulunmuştur. Öğrencilerin Scratch ile programlama öğretimine ilişkin görüşlerinin genel olarak olumlu olduğu ifade edilebilir. Öğrenciler programı kullanarak güzel ve eğlenceli vakit geçirdiklerini belirtmişlerdir. Scratch programı ile yaptıkları çalışmaların ilgilerini çektiklerini ve yapılan çalışmalarda zorlandıkları bir durum olmadığını belirtmişlerdir. Öğrenciler genel olarak dersi daha çok sevdiklerini, yeni şeyler öğrendiklerini, bilgisayarı daha etkin bir şekilde kullanabileceklerini belirtmişlerdir. Araştırmanın sonunda araştırmacı tarafından önerilerde bulunulmuştur.

**Anahtar Kelimeler:** Programlama, Scratch programı, Problem çözüme becerisi, Tutum



**T. C.**  
**NECMETTİN ERBAKAN ÜNİVERSİTESİ**  
**Eğitim Bilimleri Enstitüsü Müdürlüğü**



<b>Öğrencinin</b>	Adı Soyadı	Rukiye Berna BALA		
	Numarası	148301031005		
	Ana Bilim / Bilim Dalı	Eğitim Bilimleri Anabilim Dalı Eğitim Programı ve Öğretim Bilim Dalı		
	Program	Tezli Yüksek Lisans	<input checked="" type="checkbox"/> Doktora	<input type="checkbox"/>
	Tez Danışmanı	Prof. Dr. Füsun Gülderen ALACAPINAR		
	Tezin Adı	Effects of Scratch Programme Used in Teaching Programming Language to 6th Grade Students on Their Problem Solving Ability and Attitudes		

### SUMMARY

The aim of this study is to teach the programming given by Scratch software; from the achievement test, problem-solving skill scores and to examine whether they have an impact on the scores they get from the attitude scale. In addition, opinions of the students on the subject were also taken. The study consists of 22 students attending 6th grade in the state school in Konya. In the study, pre-test, post-test single-group experimental design was used. In this study, 8-week course was taught with the software called Scratch. As a data collection tool; achievement test, problem solving scale, attitude scale for computer course and semi-structured form were used to get their opinions.

According to the findings of the study, a significant difference was found between the pre-test and post-test scores of the students. There was increasing on the past-test scores. There was no significant difference in the scores obtained from the problem solving scale. It is observed that the effect of Scratch program used in programming teaching on problem solving skills has no statistically significant effect. There was a significant difference between the pre-test and post-test scores obtained from the attitude scale for the computer course. Additionally, student have had positive opinions and attitudes about Scratch programming teaching. They stated that they had a nice and enjoyable time using the program. They stated that they are interested in the work they do with the Scratch program in general. They stated that there is no difficulty in the studies done with Scratch program. The students stated that they generally liked the lesson more, learned new things and could use the computer more effectively. Recommendations were given at the end of the study.

**Keywords:** Programming, Scratch, Problem solving skill, Attitude

## İÇİNDEKİLER

<b>BİLİMSEL ETİK SAYFASI</b> .....	ii
<b>YÜKSEK LİSANS TEZİ KABUL FORMU</b> .....	iii
<b>TEŞEKKÜR</b> .....	iv
<b>ÖZET</b> .....	v
<b>SUMMARY</b> .....	vii
<b>İÇİNDEKİLER</b> .....	ix
<b>KISALTMALAR</b> .....	xii
<b>TABLolar LİSTESİ</b> .....	xiii
<b>BÖLÜM I</b> .....	1
<b>GİRİŞ</b> .....	1
1.1. Problem Durumu .....	1
1.2. Araştırmanın Amacı .....	4
1.3. Araştırmanın Önemi .....	4
1.4. Problem Cümlesi .....	5
1.5. Sayıtlar .....	5
1.6. Sınırlılıklar .....	6
1.7. Tanımlar .....	6
<b>BÖLÜM II</b> .....	7
<b>KURAMSAL AÇIKLAMALAR VE İLGİLİ ARAŞTIRMALAR</b> .....	7
2.1. Programlama Dili ve Eğitimi .....	7
2.1.1. Programlama ve Programlama Dili .....	7
2.1.2. Programlamada Temel Kavramlar .....	9
2.1.3. Programlama Eğitimi.....	11
2.2. Problem Çözme Becerisi.....	19

2.2.1. Problem.....	20
2.2.2. Problem Çözme .....	21
2.2.3. Problem Çözme Basamakları .....	22
2.2.4. Problem Çözme Becerisi .....	23
2.2.5. Problem Çözme Becerisini Kazanmış Bireyin Özellikleri .....	23
2.2.6. Problem Çözme Becerisinin Yararları ve Sınırlıkları .....	24
2.2.7. Problem Çözme Becerisini Sınama .....	24
2.2.8. Problem Çözme Becerisi ve Bilişim Teknolojileri ve Yazılım Dersi .....	25
2.3. Tutum .....	28
2.4. Yapılan Çalışmalar.....	30
2.4.1. Programlama ile İlgili Yapılan Çalışmalar .....	30
2.4.2. Problem Çözme Becerisi ile İlgili Yapılan Çalışmalar .....	36
<b>BÖLÜM III</b> .....	38
<b>YÖNTEM</b> .....	38
3.1. Araştırmanın Modeli .....	38
3.2. Çalışma Grubu .....	39
3.3. Veri Toplama Araçları .....	39
3.3.1. Başarı Testi.....	39
3.3.2. Problem Çözme Ölçeği.....	40
3.3.3. Bilişim Teknolojileri Dersine Yönelik Tutum Ölçeği.....	41
3.3.4. Yarı Yapılandırılmış Görüşme Formu.....	41
3.4. Verilerin Toplanması .....	42
3.5. Verilerin Analizi.....	43
<b>BÖLÜM IV</b> .....	44
<b>BULGULAR VE YORUM</b> .....	44
4.1. Birinci Denenceye İlişkin Bulgular.....	44

4.2. İkinci Denenceye İlişkin Bulgular .....	44
4.3. Üçüncü Denenceye İlişkin Bulgular .....	45
4.4. Alt Probleme İlişkin Bulgular .....	46
<b>BÖLÜM V</b> .....	49
<b>TARTIŞMA, SONUÇ VE ÖNERİLER</b> .....	49
5.1. Tartışma.....	49
5.1.1. Birinci Denenceye İlişkin Sonuçların Tartışılması.....	49
5.1.2.İkinci Denenceye İlişkin Sonuçların Tartışılması .....	50
5.1.3.Üçüncü Denenceye İlişkin Sonuçların Tartışılması .....	50
5.2. Sonuç.....	51
5.3. Öneriler .....	51
<b>KAYNAKÇA</b> .....	52
<b>EKLER</b> .....	59
Ek 1. Başarı Testi .....	59
Ek 2. Problem Çözme Ölçeği.....	69
Ek 3. Bilişim Teknolojileri Dersine Yönelik Tutum Ölçeği .....	71
Ek 4. Yarı Yapılandırılmış Görüşme Formu.....	73
Ek 5. Araştırma İzni .....	74
Ek 6. Örnek Ders Planı.....	75
Ek 7. Çalışma Kâğıdı 1 .....	78
Ek 8. Hedef-Davranışlar.....	79
<b>ÖZGEÇMİŞ</b> .....	81

**KISALTMALAR**

**AR-GE:** Arařtırma Geliřtirme

**MEB:** Milli Eđitim Bakanlıđı

**MIT:** Massachusetts Teknoloji Enstitüsü

**MATLAB:** Matrix Laboratory

**ISTE:** Uluslararası Eđitim Teknolojileri Topluluđu

**BTY:** Biliřim Teknolojileri ve Yazılım

**PHP:** Hypertext Preprocessor

**PISA:** Uluslararası Öğrenci Deđerlendirme Programı

**TDK:** Türk Dil Kurumu

**TABLolar LİSTESİ**

Tablo 1. Programlama Eğitiminin Ülkelerin Okul Seviyelerine Göre Durumu .....	11
Tablo 2. Çocuklara Yönelik Geliştirilen Programlama Araçları .....	14
Tablo 3. Çalışmada Kullanılan Modelin Simgesel Gösterimi .....	38
Tablo 4. Belirtke Tablosu .....	40
Tablo 5. Bilişim Teknolojileri Dersine Yönelik Tutum Ölçeği Maddelerinin Puanlanması .....	41
Tablo 6. Başarı Testi Ön Test ve SonTest İlişkili Örneklem T-Testi Sonuçları ...	44
Tablo 7. Problem Çözme Ölçeği Ön Test ve SonTest İlişkili Örneklem T-Testi Sonuçları .....	45
Tablo 8. Bilişim Teknolojileri Dersine Yönelik Tutum Ölçeği Ön Test ve SonTest İlişkili Örneklem T-Testi Sonuçları.....	46

## BÖLÜM I

### GİRİŞ

Bu bölümde araştırmanın problem durumu, amacı, önemi, problem cümlesi, sayıtları, sınırlılıkları ve tanımları üzerinde durulmuştur.

#### 1.1. Problem Durumu

Teknolojinin etkilerini sosyal yaşamdan ekonomiye, turizmden sağlık sektörüne kadar pek çok alanda görmekteyiz. Özellikle son yıllara bakıldığında daha önce akla bile gelmeyecek teknolojik gelişmeler konuşulmakta ve bu teknolojik ürünler kullanılmaktadır. Endüstri 4.0, drone, elektrikli arabalar, üç boyutlu yazıcılarda geliştirilen protez organlar, arttırılmış gerçeklikle hazırlanan dergi ve kitaplar, robot teknolojileri... Örnekleri çoğaltmamız mümkün fakat dikkat edilmesi gereken husus, hızla değişen ve gelişen bu çağda üretim ve tüketim dengesinin sağlanabilmesidir.

Gelişmiş olarak nitelendirilen ülkelere bakıldığında; bilgi ve iletişim teknolojileri sektöründe yetişmiş insan gücüne sahip oldukları görülmektedir. Gelişmiş ülkeler sahip oldukları nitelikli insan kaynağı ile bilgi üretip bunu dünyaya satmaktadırlar. Bilgi üreten bir toplum olmak ve katma değeri yüksek ürünleri üretebilmek için bireyleri; tüketimden üretime yöneltmek gerekmektedir. Gençlerimizi ve çocuklarımızı üretim yapabilecekleri alanlara yönlendirmeliyiz. Türkiye'nin de katma değeri yüksek ürünler üretebilen ve teknolojiyi ihraç eden ülke konumuna gelebilmesi için bireylere programlama becerisi kazandırmak önemli hale gelmiştir. Günümüzde yazılım alanında yetişmiş insanlara duyulan ihtiyaç artmaktadır. Bu konuda üretken ve yaratıcı bireylerin yetiştirilmesi ülkemizin geleceği için önemli hale gelmektedir (Demirer ve Sak, 2016: 4).

Bu hedefe ulaşabilmek için öğrencilere verilen bilişim eğitimi okul öncesi, ilkokul ve ortaokul kademelerinde yaygınlaştırılmalıdır. Ortaokul seviyesinde Bilişim Teknolojileri dersinde anlatılmakta olan ofis uygulamaları, etik ve güvenlik,

donanımsal bilgilerin yanı sıra öğrencilere programlamaya yönelik konularda öğretilmelidir. 2012 yılında ismi 'Bilişim Teknolojileri ve Yazılım' olarak güncellenen dersin aynı zamanda öğretim programı da güncellenerek programlama konusuyla ilgili bir ünite eklenmiştir (MEB, 2012). 2018 yılında dersin öğretim programı tekrar güncellenerek Problem Çözme ve Programlama ünitesine yer verilmiştir. Bu üniteye; algoritma tasarımına ilişkin (arama, sıralama vb.) anlayış geliştirme; sözel ve görsel olarak ifade etme, problem çözmek için değişken, atama, sıralı mantık, karar yapısı, döngü ve fonksiyon yapılarını kullanma, problemleri çözmek için uygun programlama yaklaşımını seçme ve uygulama konusunda beceriler kazandırılması amaçlanmıştır (MEB, 2018).

İlgili çalışmalara bakıldığında çocuklara programlama konusunda eğitim verildiğinde; öğrencilerin okula, derslere olan motivasyonlarının arttırılabileceği ve öğrencilerin problem çözme ile analitik düşünme becerilerinin geliştirilebileceği görülmektedir (Akpınar ve Altun, 2014: 3). Başka bir çalışmada araştırmacılar; programlama öğretiminin matematiksel düşünmeyi geliştireceğini, günlük hayattaki birçok problemin programlama ile kazanılan pratik ve yaratıcı fikirlerle çözüme kavuşturulabileceğini, alternatif düşünme yolları geliştirerek çözümlere eleştirel bakabileceklerini ifade etmişlerdir (Çakıroğlu, Sarı, Akkan, 2011: 1096). Bir diğer çalışmada, kullanılan programlama dilinden bağımsız olarak programlama becerisinin; mantıksal düşünme, algoritma oluşturma gibi farklı alanlarda problem çözme ve analitik düşünme becerisini kazandırabileceği ifade edilmiştir (Ersoy, Madran, Gülbahar, 2011: 1). Başka bir araştırma da ise programlama eğitiminin çocukların problem çözme becerilerine olumlu yönde katkı sağladığı sonucuna ulaşılmıştır (Çetin, 2012: 5).

Dikkat edilmesi gereken konulardan birisi bu eğitimin hangi dönemde verileceğidir. Utting, Cooper, Kölling, Maloney, Resnick (2010: 2), çalışmalarında farklı yazılımlar için farklı yaş aralıkları sunarak, Scratch için 8-16 yaş aralığını önermişlerdir. Ayrıca çocukların bilişsel gelişimlerinin, programlama eğitimi verilirken kullanılacak yazılımların seçiminde etkili olacağını ifade etmişlerdir.

Özoran, Çağıltay ve Topallı (2012: 125), çalışmalarında öğretmen ve öğrencilerin, programlama öğretiminde farklı sorunlarla karşılaşabileceklerini belirtmişlerdir. Bu sorunlar; değişkenler, döngüler, diziler, fonksiyonlar ve programlama dillerinin sözdizimi gibi programlama kavramlarının karmaşıklığından kaynaklanmaktadır. Bu kavramların karmaşık yapısının, öğrencilerin motivasyonlarına etki edebileceğini belirtmişlerdir.

Programlama dili derslerinde başarıyı etkileyen pek çok durum vardır. Bunlardan bazıları istek, programlamaya yönelik tutum, programlama dil yapısının komplike oluşu, öğretim yöntem ve tasarımı olarak sıralanabilir. Programlama eğitimi sırasında öğrenci yeteri kadar istekli değilse, dersler ilerledikçe konudan soğuyabilir ve bu durum daha az çalışmasına yol açabilir. Bir programın üretilmesi aşamasında, bilgisayar karşısında çok çalışmak gerektiğini düşünen öğrenciler, bu süreçleri sıkıcı bulabilirler (Ersoy ve diğerleri, 2011: 3).

Programlamayı yeni öğrenen bireyleri zorlayan durumların başında, kod yazarken yapılan yazım hataları gelmektedir. Tıpkı yeni bir dil öğrenmeye benzeyen bu süreçte yanlış yazılan kelimeler, unutulan işaretler öğrencileri oldukça zorlamaktadır. Programlama eğitiminde temel kavramların anlaşılması çok önemlidir. Aksi halde yazılan programlar ezbere yapılan uygulamalara dönüşmektedir. İlk olarak işin temelini oluşturan algoritma mantığı öğretilmeli ardından temel kavramlar verilmelidir. Programlama sürecini kolaylaştıracak, daha eğlenceli hale getirebilecek araçlar ile dersi işlemek gerekmektedir.

İlgili araştırmalara bakıldığında, programlamaya yeni başlayanlar için süreci daha eğlenceli hale getirecek ve temel kavramların daha kolay öğrenilebilir olmasını sağlayacak çeşitli yazılımlar kullanıldığı görülmektedir (Coşar, 2013; Genç ve Karakuş, 2011; Yükseltürk ve Altıok, 2016). Süreci daha etkileşimli ve kolay hale getiren programlara örnek olarak; Scratch, Microsoft Small Basic, Alice, MIT AppInventor verilebilir. Scratch ile programlama eğitimi alan Bilişim Teknolojileri öğretmen adaylarının görüşlerinin incelendiği bir çalışmada Yükseltürk ve Altıok

(2016), adayların Scratch programına ilişkin algılarının olumlu olduğunu belirtmişlerdir.

### **1.2. Araştırmanın Amacı**

Bu araştırmada, altıncı sınıf öğrencilerine programlama dili öğretilirken kullanılan Scratch programının, öğrencilerin başarılarına, problem çözme becerilerine ve bilişim teknolojileri dersine yönelik tutumlarına etkisini tespit etmek, uygulamaya ilişkin öğrenci görüşlerini belirlemek amaçlanmıştır.

### **1.3. Araştırmanın Önemi**

Yaşadığımız yüzyılda bilgisayar teknolojisi her yerde karşımıza çıkmaktadır. Yaşanan bu değişimle birlikte ülkelerin nitelikli insan gücüne olan ihtiyacı artmıştır. Her vatandaşın teknolojiyi anlayabilmesi, kullanabilmesi, üretim yapabilmesi için bilgisayar bilimi eğitimi görmesi gerekmektedir. Verilen bu eğitimle insanların meslek olarak; bilgisayar bilimcisi, yazılım uzmanı gibi unvanlara sahip olmasını beklemek doğru değildir. Burada istenilen, özellikle küçük yaş grupları olmak üzere herkese bilgisayarı kullanarak; yeni fikirler oluşturma ve üretim yapma fırsatı verilmesidir. Bu hedefe ulaşmak için pek çok ülke, eğitim sistemlerine ders olarak bilgisayar bilimini eklemeye başlamıştır (Sayın, 2017: 137-145).

Son yıllarda programlama becerisinin birçok alanda ayırt edici ve aranan bir yeterlilik olduğundan bahsedilmektedir (Kalelioğlu ve Gülbahar, 2014: 33-34; Yükseltürk ve Altıok, 2015: 51). Özellikle küçük yaş gruplarını programlama ile tanıştırmak ve onlara bu beceriyi edindirmek için çeşitli yazılımlar geliştirilmekte, etkinlikler yapılmaktadır. Programlama eğitimi zor ve uzun bir süreçtir. Özellikle ilk ve orta seviyeli okullarda bu beceriyi kazandırmaya yönelik çeşitli araştırmalar yürütülmektedir. Yapılan bu çalışmada ortaokul seviyesinde programlama öğretimini daha verimli kılabilme amaçlanmıştır.

Yapılan araştırmalar incelendiğinde ortaokul seviyesinde yeterli sayıda çalışma olmadığı gözlenmiştir. Bu araştırma, programlamayı öğretmek için Scratch

yazılımının ortaokul seviyesinde kullanımını araştıracaktır. Çalışma ile öğrencilerin bu alana yönelik farkındalık kazandırılması da amaçlanmaktadır. Erken yaş gruplarında çocukların programlamaya yönelik ilgi ya da becerilerinin belirlenmesi ile öğrenciler bu alana yönlendirilebilir. İsteyen öğrencilere kendilerini geliştirmeleri için yol açılması sağlanabilir. Bilgi ve iletişim teknolojileri alanında yetişecek yazılımcıların ülke ekonomisine katkı sağlayacağı söylenebilir.

#### **1.4. Problem Cümlesi**

Ortaokul 6.sınıf öğrencilerine programlama dili öğretilirken kullanılan Scratch programının öğrencilerin başarı testi, problem çözme becerisi ölçeği ve derse yönelik tutum ölçeğinden aldıkları puanları üzerinde anlamlı bir etkisi var mıdır? Scratch ile programlama öğretimine ilişkin öğrencilerin görüşleri nelerdir?

#### **Denenceler**

1. Scratch programı uygulanan grubun başarı testinden aldıkları ön test- son test puanları arasında anlamlı bir fark vardır.
2. Scratch programı uygulanan grubun problem çözme beceri ölçeğinden aldıkları ön test- son test puanları arasında anlamlı bir fark vardır.
3. Scratch programı uygulanan grubun derse yönelik tutum ölçeğinden aldıkları ön test- son test puanları arasında anlamlı bir fark vardır.

#### **Alt Problem**

Scratch ile programlama öğretimine ilişkin öğrencilerin görüşleri nelerdir?

#### **1.5. Sayıtlar**

- Kontrol altına alınamayan değişkenler, sonucu anlamlı derecede etkilememiştir.
- Kapsam geçerliliği için başvuru uzman kanısı yeterlidir.

## 1.6. Sınırlılıklar

- Araştırma 2017-2018 eğitim- öğretim yılı
- Çalışmaya katılan 6-A sınıfı öğrencileri
- Çalışma süresi olan 8 hafta
- Çalışmada kullanılan Scratch yazılımı
- Başarı puanı, başarı testinden aldıkları puanlar
- Problem çözme becerileri, problem çözme ölçeğinden aldıkları puanlar
- Tutum puanları, bilişim teknolojileri dersine yönelik tutum ölçeğinden alınan puanlar ile sınırlıdır.

## 1.7. Tanımlar

**Scratch Programı:** Uygulama esnasında öğrencilere programlamanın temel kavramlarının anlatımında kullanılacak uygulamadır. Kod blokların sürükle-bırak mantığıyla bir araya getirilmesi ile çalışır.

**Başarı Testi Puanı:**30 sorudan oluşan, Başarı Testi' nden her bir öğrencinin aldığı puandır.

**Problem Çözme Becerisi Puanı:**20 sorudan oluşan, Problem Çözme Ölçeği' nden her bir öğrencinin aldığı puandır.

**Tutum Puanı:** 36 sorudan oluşan, Bilişim Teknolojileri Dersine Yönelik Tutum Ölçeği' nden her bir öğrencinin aldığı puandır.

## BÖLÜM II

### KURAMSAL AÇIKLAMALAR VE İLGİLİ ARAŞTIRMALAR

Bu bölümde, Türkiye ve dünyada yapılan çalışmalar incelenerek araştırma problemi içerisinde bulunan programlama dili ve eğitimi, problem çözme becerisi, tutum, yapılan çalışmalar açıklanmıştır.

#### 2.1. Programlama Dili ve Eğitimi

##### 2.1.1. Programlama ve Programlama Dili

Bilgisayara verilen komutların bütününe program denir. Programlama ise, bilgisayarın donanıma nasıl davranacağını anlatan, bilgisayara yön veren komutlar, kelimeler, aritmetik işlemlerdir. Kesici ve Kocabaş'a (2007) göre, bir bilgisayar programının hazırlanması beş aşamadan meydana gelir (aktaran Keskinsoy, 2010: 11):

1. Problemin Tanımı
2. Çözüm Yolunun Belirlenmesi (Algoritmanın Hazırlanması)
3. Programın Kodlanması
4. Programın Yorumlanması ve Derlenmesi
5. Programdaki Hataların Belirlenmesi ve Giderilmesi

Bu aşamalarla ilgili açıklamalar aşağıda verilmiştir.

**1. Problemin Tanımı:** Problemin çok iyi anlaşılmasıdır. Problem konusunda yeterli bilginiz yoksa o konuda derinlemesine araştırma yaparak kafanızdaki bütün soru işaretlerini ortadan kaldırmanız gerekmektedir.

**2. Çözüm Yolunun Belirlenmesi (Algoritmanın Hazırlanması):** Problem açık bir şekilde tanımlandıktan sonra problemin çözümü için yapılacak işlemler, ilgili açıklamalarla beraber maddelendirilir (algoritma). Bu işlemden sonra programın genel yapısı belirli semboller vasıtasıyla mantıklı şekilde ayrıntılı olarak gösterilir

(akış şeması). Kullanacağınız çözüm yolu daha önce kullanılan ya da size ait bir yol olabilir.

**3. Programın Kodlanması:** Algoritma ve akış şeması hazırlandıktan sonra ilgili programlama dili kurallarına göre programın yazılmasıdır.

**4. Programın Yorumlanması ve Derlenmesi:** Yazılan bilgisayar programının bilgisayarın anlayabileceği dile (makine dili) çevrilmesidir. Derlenen program çalıştırılmaya hazırlanmış demektir.

**5. Programdaki Hataların Belirlenmesi ve Giderilmesi:** Programdaki yazım (kodlama) ve mantık hatalarının belirlenmesi ve giderilmesidir. Mümkünse bu aşamada programın başka bir gözle kontrol edilmesi daha uygundur. Daha sonra program tekrar makine diline çevrilir ve çalışması denir.

Programcının, bir bilgisayara ne yapmak istediğini anlatmasının standartlaştırılmış yoluna programlama dili denir (Başçiftçi, 2010: 2). Programlama diliyle ilgili başka bir tanımda şu şekildedir: Bir problemin algoritmik çözümünün bilgisayara anlatılmasını sağlayan, son derece sıkı sıkıya kuralları bulunan kurallar dizisidir. Programlama dillerini beş gruba ayırarak incelenebilir (Çamoğlu, 2010: 12-13):

**Birinci Nesil Programlama Dilleri:** Makine dilleri olarak isimlendirilir. Bu diller, ilk bilgisayarların programlanabilmesi için oluşturulmuştur. Geliştirildikleri bilgisayara özgü olan bu kodların başka bir makinede çalışma özelliği yoktur. Kodlar özel bir derleyiciye ya da çeviriciye gereksinim olmadan direk bilgisayara yazılmaktadır.

**İkinci Nesil Programlama Dilleri:** Çevirmen (Assembly) diller olarak adlandırılırlar. Makine dillerine göre daha anlaşılır yapıları vardır. Kodların işlemciye direk yazılması yerine kısaltmalar ve komutlar kullanılmıştır. Bu dillerle yazılan programların makinelerin anlayacağı şekilde 1 ve 0'lara dönüştürülmesi için çevirmen programlar (assembler) kullanılmaktadır.

**Üçüncü Nesil Programlama Dilleri:** İkinci nesil programlama dillerinden daha kolay kullanılır ve daha yaygındır. Bu dilde yazılan programların donanımdan bağımsız olup farklı cihazlarda da kullanımı sağlanmıştır. Bu programlama dilleri daha çok İngilizce ağırlıklı olarak insanların kullandığı dillere yakınlaşmıştır. Geliştirilen programların istenilen bilgisayarlarda çalışmaları için derleyici (compiler) ya da yorumlayıcı (interpreter) adı verilen programlara ihtiyaç duyulur. Fortran, Cobol, C, C++ en bilinen üçüncü nesil programlama dilleridir.

**Dördüncü Nesil Programlama Dilleri:** Ticari amaçlar için tasarlanan bu dillerin kullanımı diğer dillere göre daha kolaydır. Programcıya gereksinim olmadan kullanılması ve iş dünyasına özgü ihtiyaçları karşılaması için geliştirilmiştir. Örnek olarak MATLAB, Windows Forms, Oracle Forms-Reports, SPSS gösterilebilir.

**Beşinci Nesil Programlama Dilleri:** Programcının algoritma geliştirerek çözüm üretmesinin ötesinde, koşulları ve kısıtlamaları bilgisayara verdiğinizde, bilgisayarın çözümü kendisinin bulmasına yönelik olarak tasarlanmaktadır. Kodlamanın yerine bildirimsel yöntemle çalışan ve özellikle yapay zekâ alanında araştırmalarda kullanılan bu yeni programlama dillerine örnek olarak Prolog, Lisp, OPS5 ve Mercury verilebilir.

Bu sınıflandırma ile programlama dillerinin gelişimi hakkında bilgi sahibi olabiliriz. Zaman içerisinde ihtiyaçlara bağlı olarak gelişim gösteren dillerin ortak bir noktası vardır. Bu nokta hepsinde ortak olan temel kavramlardır.

### **2.1.2. Programlamada Temel Kavramlar**

Yazılıma yeni başlayanların, bu alanda kendini geliştirmek isteyenlerin yapması gereken ilk iş temel kavramları öğrenmektir. Eryılmaz (2003: 17-44) temel kavramları şöyle sıralamıştır:

**Genel Programlama Bilgisi:** Programlama, aşamaları olan ve ciddi çalışma gerektiren bir süreçtir. İlk olarak problem çözme, bu problemin aşamalarını tasarlama ve algoritma oluşturabilme davranışlarının oluşması gerekmektedir.

**Problem Çözümü:** Bir problemin çözümü için öncelikle programcı tarafından sorunun anlaşılması gerekmektedir. Bilgisayarda bir problemi çözerken izlenmesi gereken adımlar: Problemin tanımlanması, çözüm yollarının belirlenmesi ve en uygun çözüm yolunun seçilmesi, seçilen çözüm yolunun algoritmasının oluşturulması, akış çizelgesinin hazırlanmasını, programın kodlanmasını ve programın çalıştırılması adımlarından oluşur.

**Algoritma:** Bir problemin çözümünde izlenecek olan işlem adımlarıdır. Algoritmanın başlangıcı olmalıdır. Çözümler adım adım yazılmalıdır. Bitişi belirtilmelidir.

**Değişkenler ve Sabitler:** Programlama dillerinde sabit ve değişken olmak üzere iki farklı şekilde değer gösterimi vardır. Sabit değer, programda hep aynı değere almaktadır. Değişken değer, programın her çalışmasında farklı değer alabilir.

**Atama İşlemi:** Bir değişkene değer aktarmak için kullanılır.

**Karar Verme ve Kontrol İşlemleri:** Programda bulunan değişkenlerin değerlerini kontrol etmek amacıyla kullanılır.

**Döngü Yapıları:** Programın belli bir bölümünün birden çok tekrarlanması işlemidir. Değişkenin belirli bir değerden başlayarak son değeri alıncaya kadar ilgili işlemler tekrarlanır.

**Diziler:** Aynı tip değişkenlerden oluşurlar. Bir değişken adı altında birden çok değer tutulmasını sağlayan yapılardır.

Bayman ve Mayer (1988) programcılığı yeni öğrenen bireyler üzerinde yaptığı çalışmalarında kavramsal bilgi ile öğrenenlerin problem çözme performansları arasında anlamlı bir ilişki bulmuşlardır. Buna göre öğrenenlerin programlamada başarılı olabilmeleri için temel programlama kavramlarını ve programlama mantığını iyi kavramaları gerekmektedir. Üst düzey bilişsel beceri gerektiren programlama becerisinin kazandırılması için temel programlama kavramlarının ve programlama mantığının kazandırılması önemlidir (aktaran Erol, 2015: 6).

### 2.1.3. Programlama Eğitimi

Dünyada programlama öğretimi hakkında yapılan çalışmalara bakıldığında, ülkelerin son yıllarda bu konuya önem verdikleri gözlenmektedir. Programlama eğitiminin çoğunlukla ortaokul öğretim programlarında yer aldığı, birkaç yıldır da ilkokul öğretim programlarına eklendiği görülmektedir. European Schoolnet araştırma raporuna göre, Avrupa Birliği ülkelerinde programlamayı öğretim programlarına ekleyen ülkelerin olduğu görülmektedir. Ayrıca bazı ülkelerin öğrencilerini, okul dışı faaliyetlerle programlama ile tanıştırdıkları anlaşılmaktadır (Şimşek, İ., 2018: 38). Tablo 1’de ülkelerin seviyelere göre programlama öğretimi durumları verilmiştir.

**Tablo1. Programlama Eğitiminin Ülkelerin Okul Seviyelerine Göre Durumu**

	İlkokul	Ortaokul	Ortaokul(Meslek)	Lise	Meslek Liseleri
Avusturya			X	X	X
Belçika	X	X	X		
Bulgaristan				O	O
Çek Cumhuriyeti					O
Danimarka		O		X	X
Estonya	X	X	X	X	X
Finlandiya	O	O			
Fransa	X	O		O	
Macaristan				O	
İngiltere	O	O		O	
İrlanda		X			
İspanya	X	X		X	
İsrail	X	X	X	X	X
Litvanya				X	
Malta				X	
Polonya		X		X	X
Portekiz		O		O	
Slovakya	O	O	O	O	O

Zorunlu O Seçmeli X

Son yıllarda dünyada, programlama eğitimini daha küçük yaşlarda çocuklarla tanıştırmak için çeşitli etkinlikler düzenlenmektedir. Yapılan araştırmalarda 21.yüzyıl bireylerinde bulunması gereken; problem çözme, yaratıcılık, algoritmik ve bilgi-işlemsel düşünme gibi temel becerilerin programlama ve bilgisayar biliminin öğretilmesi ile kazandırılabilmesine yönelik bulgulara ulaşılmıştır (Çatlak, Tekdal ve Baz, 2015: 13).

Gönüllü bir kuruluş olan code.org sitesi, "Anybody can learn" anlayışıyla faaliyet göstermektedir. Sitede Kodlama Saati adıyla yapılan etkinliklerle öğrencilerin kodlama ile tanıştırılması amaçlanmıştır. Çevrim içi olarak erişilebilen etkinliklerde dil desteği mevcut olup, çocukların ve gençlerin ilgi gösterdikleri çeşitli film, animasyon ve oyun karakterlerinden yararlanılmıştır. Code.org sitesinde bilgisayar alanında uzmanlardan sanatçılara kadar farklı kişilerin programlama ile ilgili görüşlerine yer verilmiştir.

EU CodeWeek (Avrupa Kod Haftası) organizasyonu ise Milli Eğitim Bakanlığı'nın da üyesi olduğu Avrupa Okul Ağı (European Schoolnet) tarafından yürütülmektedir. Etkinlik, Ekim ayı içinde Avrupa'da bulunan ve katılmak isteyen okullar ile gerçekleştirilmektedir. Avrupa Kod Haftası'nın sitesinde yapılan etkinliklerin amaçları şu şekilde açıklamıştır: Programlama ve diğer teknolojik aktiviteler yoluyla yaratıcılığı, problem çözmeyi ve iş birliğini teşvik eden bir taban hareketidir. Buradaki amaç, programlamayı daha görünür kılmak, gençlere, yetişkinlere ve yaşlılara fikirlerin kodlamayla nasıl hayata geçirildiğini göstermektir. Ayrıca bu becerileri ortaya çıkarmak ve motive olmuş kişileri öğrenme amacıyla bir araya getirmektir (<https://codeweek.eu/about/>).

Farklı ülkelerin, ilk ve orta seviyedeki okullarının öğretim programlarına bilgisayar ve kodlama alanında dersler eklediklerini görmekteyiz. Örneğin, İngiltere Bilgi ve İletişim Teknolojileri adındaki dersi 2013 yılında Bilgi İşleme (computing) olarak değiştirmiştir. Bu dersi yaş gruplarına göre üç aşamalı olarak gruplandırmıştır. Bilgi İşleme dersi; bilgisayar bilimi, bilgi teknolojileri ve dijital okuryazarlık olmak

üzere üç esas üzerine inşa edilmiştir. İngiltere Bilgisayar Bilimi derslerini ilköğretimde zorunlu hale getiren ilk G20 ülkesi olmuştur (Sayın, 2017: 140-141).

Litvanya sadece mantıksal düşünme ve kodlama becerisini A-12 düzeyinde kazandırmak istemiştir. Çek Cumhuriyeti, Türkiye ve İsrail gibi ülkeler mantıksal düşünme becerisini, problem çözme becerisini, kodlama becerilerini ve diğer temel yeterlilikleri, öğrencilerin bilişim teknolojileri alanındaki kariyer planlarını desteklemek amacıyla öğretim programlarına eklemiştir (Sayın, 2017; 139).

Türkiye’de programlama eğitimi; ortaokul kademesinde, meslek liselerinin bilişim teknolojileri alanında ve üniversitelerin çeşitli mühendislik bölümlerinde verilmektedir. Ayrıca alanda kendini geliştirmek isteyenler için özel kurslarda bulunmaktadır. Genellikle laboratuvarında işlenen dersler, gösterip-yaptırma tekniği kullanılarak anlatılmaktadır. Değerlendirme kısmında da yazılı sınavlar, uygulama sınavları ya da bitirme projeleri olmaktadır. Programlama eğitiminin ilköğretim düzeyinde amacı profesyonel yazılım geliştirmekten ziyade öğrencilerin algoritmik düşünme becerilerini somut bir şekilde uygulayarak öğrenmelerini sağlamaktır. Öğrencilerin programlamaya yönelik tutumlarını olumlu biçimde geliştirmelerini sağlayan Scratch, Alice ve Kodu gibi birçok araç mevcuttur (Avcı ve Ersoy, 2018: 74).

Gomes ve Mendes (2007) yaptıkları çalışmada, bir programlama dilini öğrenmenin öğrenciler tarafından, uzmanlık gerektiren ve ileri seviyede eğitim almış kişilerin uğraşabileceği bir iş olarak gördüklerini belirtmişlerdir. Öğrenciler geleneksel programlama dillerini öğrenmesi zordur. Bu yüzden programlamaya başlamak için tercih edilen programlama dili çok önemlidir. Özellikle yeni başlayacak gruplarda süreci kolaylaştıracak blok tabanlı programla araçlarının tercih edildiğini görmekteyiz.

### 2.1.3.1. Çocuklara Yönelik Programlama Araçları

Programlama dillerinin kendine özgü sözcük yapısı ve yazım kuralları, soyut olan kavramsal kuralları çocukların ilk zamanlarda zorlanmalarına sebep olabilmektedir. Yeni başlayanlara süreci biraz daha kolaylaştırmak için çok sayıda blok tabanlı programlama aracı geliştirilmiştir.

Blok tabanlı programlama araçlarında kullanılan sözdizimi yerine, görsellerle ifade edilen, sürükle- bırak mantığıyla çalışan bloklar bulunmaktadır. Böylece metin tabanlı dillerde yaşanan yazım hatalarını ve kuralların ezberlenmesinin engellemesi amaçlanmıştır. Blokların yap-boz şeklinde tasarlanması sadece doğru parçaların bir araya gelmesine izin vermektedir. Yapılan araştırmalara bakıldığında blok tabanlı araçların öğrencilerin programlama öğrenimine katkısını görmekteyiz. Erol (2015), Scratch ile yapılan programlama öğretiminin öğrencilerin istek ve programlama başarısına etkisini incelediği araştırmasında, deney grubunda yer alan katılımcıların programlama başarı puanlarında deney grubu lehine anlamlı bir fark olduğunu bulmuştur. Scratch ile çalışmalar yapan öğrenciler; etkinliklerinin eğlenceli ve kolay olduğunu, ders boyunca yapılan faaliyetlerin programlama mantığını kazandırmada ve motivasyonu arttırmada etkili olduğunu belirtmişlerdir. Programlama eğitiminde, çocuklara yönelik çeşitli programlar geliştirilmiştir. Tablo 2' de programlara örnekler verilmiştir.

**Tablo 2. Çocuklara Yönelik Geliştirilen Programlama Araçları**

Seviye	Program adı
Anaokulu	Scratch JR, Code.org,
İlkokul	Monkey, Scratch, Kodable,
Ortaokul	Scratch, AppInventor, Kodu Game Lab, Code.org, Alice, Small Basic
Lise Seviyesi	Kodu Game Lab, AppInventor, mobil uygulamalar,

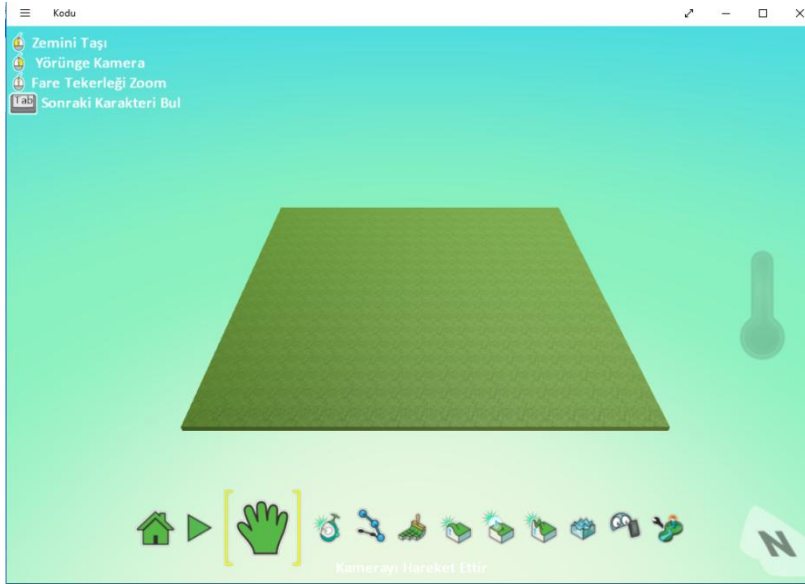
### Microsoft Small Basic

Microsoft şirketi tarafından ücretsiz olarak hazırlanan Small Basic yazılımı, ilköğretim çağındaki çocuklar içindir. Aynı zamanda programlamaya yeni başlayan bireylerde kullanılabilir. Türkçe dil desteği sağlayan uygulamaya, kendi web sitesinden (<http://smallbasic.com/>) ulaşılabilir. Ayrıca sitede yeni başlayanlar için Small Basic kitapçığı yer almaktadır. Basit ve kolay bir arayüze sahip olması programın avantajlarından biridir.

### KoduLab

Kodu Game Lab Microsoft tarafından oyun tasarlamak için geliştirilmiş bir uygulamadır. Görsel tabanlı bir programlama aracı olan Kodu problem çözme, hikâye anlatımı ve programlama öğretimi için kullanılabilir. Bu programı, programlama becerisine sahip olmayanlar da kullanabilir (<https://www.kodugamelab.com/about/>). Şekil 1'de programdan görüntü yer almaktadır.

**Şekil 1. Kodu Game Lab Programı Ekran Görüntüsü**

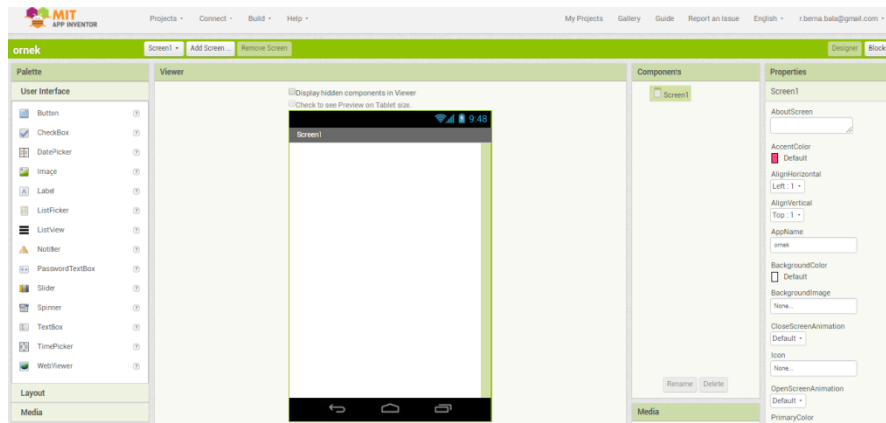


### MIT AppInventor

Google ve MIT Üniversitesinin işbirliği ile geliştirilen uygulama Android mobil işletim sistemli telefonlar için uygulama geliştirebilen ücretsiz bir araçtır.

Uygulama oluşturmak için sürükle-bırak arayüzü kullanılır. Kullanıcılar, grafik arayüzünü tasarlar, kodlarını temsil etmek için birbirine uyan bulmaca parçalarını sürükleyip bırakır ve kodlarını bir emülatör veya bir Android cihazda test eder. Bu ortam, değişkenler, koşullar, döngüler, prosedürler, mantık, olaylar ve listeler kavramlarını öğretmeye yardımcı olur. Kullanıcıların Android için gerekli incelikleri öğrenmeden Java dilinde kodlama yapmalarını sağlar (Yıldırım, 2017: 11). Şekil 2’de programdan görüntü yer almaktadır.

## Şekil 2. AppInventor Programı Ekran Görüntüsü



### Alice

Üç boyutlu bir ortamda programlama öğretilen, hikâye oluşturabilen, oyun geliştirebilen ya da animasyon hazırlanabilen ücretsiz bir yazılımdır. Üç boyutlu nesnelere oluşturarak, sanal bir dünya tasarlamaya ve nesnelere hareket ettirmeye izin verir. Kullanıcılar Alice programı ile bir program oluşturmak için ilgili kodları sürükle-bırak mantığına göre çalışma alanına ekleyebilmektedir. Böylece sözdiziminden kaynaklanan hatalarla uğraşmak zorunda kalmazlar. Alice programındaki talimatlar nesne tabanlı programlama dilindeki standart kodlamalara karşılık gelmektedir. Kullanım kolaylığı, etkileşimli ve görsel olması gibi avantajlarından dolayı farklı çalışmalarda kullanılmıştır (Baştemur Kaya ve Çakır, 2018: 189).

### 2.1.3.2. Scratch Programı

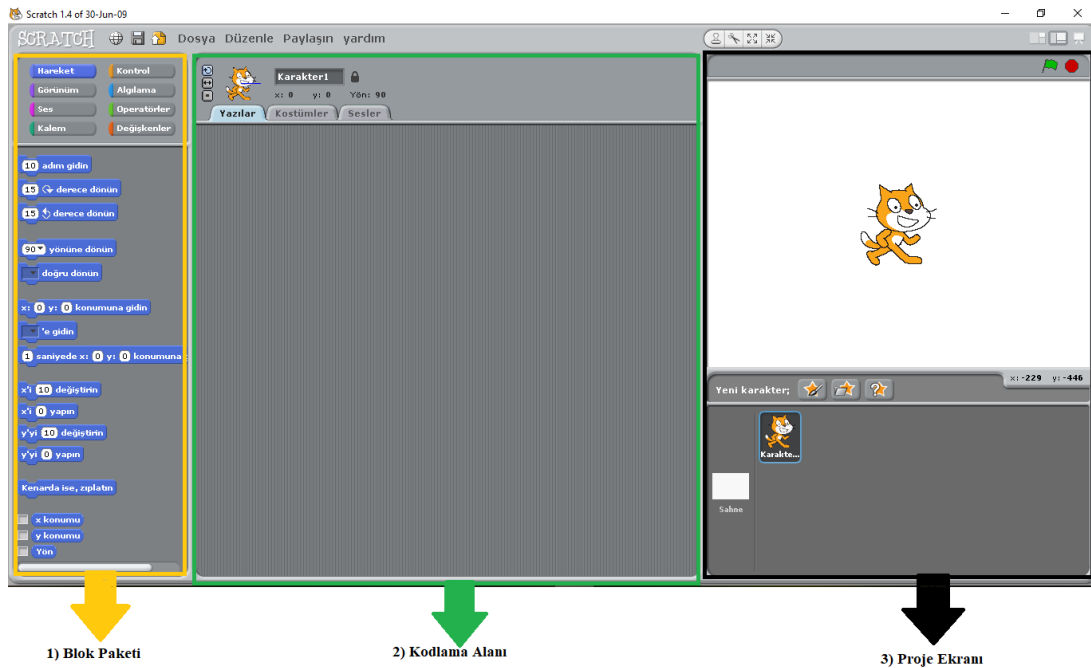
“Hayal et, kodla, paylaş” sloganını ile duyurulan Scratch programı MIT (Massachusetts Institute of Technology) Medya laboratuvarlarında geliştirilmiştir. Program 150'den fazla ülkede 40'tan fazla dilde kullanılmaktadır. Scratch ile öğrenciler kod yazmak yerine kod bloklarını sürükleyip bırak yöntemiyle birleştirip programlarını oluştururlar. Sadece çocuklar için değil kodlamaya yeni başlayan herkesin kullanabileceği bu programla, kendi hikâyeyi, oyunlarınızı ya da animasyonlarınızı oluşturabilir ve bunları uygulamanın sitesinden başkalarıyla paylaşabilirsiniz (<https://scratch.mit.edu/about>). Scratch, (öncelikle 8 ila 16 yaş arasında) kullanıcılarına animasyonlu hikâyeler, oyunlar gibi projeler üzerinde çalışırken programlamayı öğrenmelerini sağlayan görsel bir programlama ortamıdır (Maloney ve diğerleri, 2010: 7). Ücretsiz olan bu program her yaşta insanın kullanabileceği blok tabanlı bir programlama aracıdır.

Scratch programı ile kod bloklarını bir araya getirerek projeler üretmek metin tabanlı programlama dillerinden çok daha kolaydır (Genç ve Karakuş, 2011: 982). Program yazarken yapılabilecek yazım hatalarını en aza indirgeyen Scratch programı bu özelliğiyle çocuklar için daha uygun bir programlama aracıdır. Bu programla çocuklara farklı alanlarda projeler üretme imkânı da verilmiştir. Scratch, öğretmenlere programlama kavramlarını her düzeye öğretebilmeleri ve ürün oluşturabilmeyi kolaylaştırarak, öğrencilerin isteklerini arttırmaları için olanak sağlamıştır (Yükseltürk ve Altıok, 2016: 41).

Scratch' i kullananların, etkileşimli projeleri programlarken ve paylaşırken, 21.yüzyılın gerekli becerileri olarak nitelendirilen; yaratıcı düşüncelerini, sistematik olarak nasıl düşüneceklerini ve işbirlikli bir şekilde nasıl çalışabileceklerini öğrendikleri ifade edilmektedir (Resnick ve diğerleri, 2009: 60). Ayrıca Gezgin ve diğerleri (2017: 187) çalışmalarının sonucunda şunları belirtmişlerdir; Scratch programının özünde oyun olması sebebi ile dersleri eğlenceli bir formata dönüştürebileceği, motivasyonu arttırabileceği, öğrencileri yaratıcı düşünme becerisi kazandırarak üreten ve problem çözebilen bireyler olmaya teşvik edebileceği

düşünülmektedir. Bu çalışmada Scratch programı kullanılmasının nedeni; Türkçe dil desteğinin bulunması, çok sayıda kaynak dokümanın bulunması, ücretsiz olması ve sürükle-bırak yönteminin programlama konusuna yeni başlayanların seviyesine daha uygun olmasıdır. Scratch programın arayüzü; blok paketi, kodlama alanı ve sahne olmak üzere üç ana kısımdan oluşur. Şekil 3'te programın arayüzü gösterilmiştir.

### Şekil 3. Scratch Arayüzü



1) **Blok Paketi:** Yapılması istenilen projede kullanacağımız kodlara ait menüler burada bulunur. Kullanmak istediğimiz komutu sürükle-bırak yöntemiyle kodlama alanına taşıyoruz. Bu bölüm sekiz (hareket, görünüm, ses, kalem, kontrol, algılama, operatörler ve değişkenler) menüden oluşmaktadır. Şekil 4'te menülerin ekran görüntüsü yer almaktadır.

### Şekil 4. Blok Paketi



2) **Kodlama Alanı:** Bu alanda yapılacak çalışmaya uygun kodlamalar yapılır. Kod bloklarını sürükle-bırak yöntemiyle getirdiğimiz alandır. Üç kısımdan oluşur: yazılar, kostümler, sesler. Şekil 5’te kodlama alanı gösterilmiştir.

**Şekil 5. Kodlama Alanı**



3) **Proje Ekranı:** Oluşturduğumuz etkileşimli animasyonların, oyunların, projelerin uygulamaya geçtiği, yapılanların görüldüğü sahnedir. Bu bölümün sağ üst köşesinde yeşil bayrak ve kırmızı düğme bulunmaktadır. Yeşil bayrak projenin başlamasını sağlarken kırmızı düğmede projeyi durdurur. Şekil6’da proje ekranı gösterilmiştir.

**Şekil 6. Proje Ekranı**



## 2.2. Problem Çözme Becerisi

Bilişim Teknolojileri ve Yazılım dersinin güncellenen öğretim programında, özel amaçlar başlığında vurgulanan noktalardan bazıları şunlardır (MEB, 2018):

- Bilgisayar bilimine ilişkin genel bir anlayış ve teknik birikim oluşturmalarını sağlamak,
- Problem çözme ve bilgi-işlemsel düşünme becerileri edinmelerini ve geliştirmelerini sağlamak,
- Akıl yürütme sürecini takip edebilmelerini ve değerlendirmelerini sağlamak,

Öğrencilere kazandırılmak istenen bu hedeflerle bilgiyi ezberlemeleri değil bilgiyi sorgulayarak araştırmaları, etik kurallara uyarak bilişim teknolojilerini kullanmaları, süreçte karşılaştıkları problemleri çözmeleri ve etkili bir iletişim kurabilmeleri beklenmektedir. Araştırmanın bu bölümünde problem, problem çözme becerisi ve özellikleri üzerinde durulmuştur.

### **2.2.1. Problem**

İnsanı rahatsız eden, çeşitli durumlarda var olan zorlukları gidermek için çözme isteği uyandıran birtakım belirsizliklere problem denir. Kişi için sorun taşıyan, çözülmesi gereken durum olarak ifade edilebilir.

Tertemiz (2000: 11-12) problemi şu şekilde tanımlamıştır: Sonucu bilinmeyen ya da zor olan durumdur. Problemin önemi, keşfedilecek, tartışılacak ya da düşünülecek bir soru olmasıdır (aktaran Altun, 2013: 29). Problemin başka bir tanımı ise Türk Dil Kurumu (TDK, 2010) güncel sözlükte şu şekilde ifade edilmiştir: Teoremler veya kurallar yardımıyla çözülmesi istenen soru.

Problemler ile sadece derslerde çözülen sorularda değil günlük hayatta da karşılaşırız. Örneğin bir çocuğun bozulan arabası, annenin akşama yetiştirmesi gereken yemeği ya da bir gencin arkadaşlarıyla buluşmak için bindiği otobüsün arızalanması. Bu durumların her biri ilgili kişiler için sorun teşkil etmektedir ve çözülmesi gerekmektedir. Kişiler sorunları ortadan kaldırmak için problem çözmeyi bilmeli ve problem çözme becerisine sahip olmalıdırlar.

### 2.2.2. Problem Çözme

Problem çözme, hedefe ulaşmak için karşılaşılan güçlüklerin uygun bir yolla çözümlenmesi sürecidir. Problemi çözebilmek için öncelikle bir sorun olarak algılanması ve tanımlanması gerekir. Ardından problem çözme aşamasına geçilebilir. Yaşamın her alanında karşımıza çıkan problem çözmeyi öğrencilere sistemli bir şekilde verecek olan okullara büyük iş düşmektedir. Okullarda problem ve problem çözme kavramlarının öneminden bahsedilerek öğrencilerin problem çözmeye yönelik motivasyonlarının artırılması gerekir.

Literatür incelediğinde problem çözmenin ne olduğuna dair farklı tanımların olduğu görülecektir. Öğülmüş (2006), problem çözmeyi mevcut durumla erişilmek istenen amaç arasındaki boşluğun (olanla olması gereken durum arasındaki farkın) algılandığı ve bunun yol açtığı gerginliği ortadan kaldırmaya yönelik çabaları içeren bilişsel ve davranışsal bir süreç olarak ifade etmiştir. Morgan (1999), problem çözmeyi karşılaşılan engeli aşmanın en iyi yolunu bulmak olarak ifade etmektedir (aktaran Sezgin, 2011: 18).

Problemi çözerken bireyi etkileyen farklı faktörler bulunmaktadır. Bunları şu şekilde sıralayabiliriz: Zekâ, önceki yaşantılar, ön bilgi, değer, tutum, ilgi, istek, kaygı düzeyi, ailenin sosyal yapısı, eğitim düzeyi, sağlık durumu.

Problem çözme sürecinde önemli olan noktalardan biri de, sürecin kendisidir. “Davidson, Deuser ve Sternberg (1994), problem çözme sürecini yaşanan durumdan istenilen duruma ulaşmak için gösterilen çaba olarak tanımlamakta ve problem çözmenin üst bilişsel süreçleri içerdiğini söylemektedirler. Bu bilişsel süreçler kişiye, problemi tanımlamasında, gerçek problemin ne olduğunun belirlenmesinde ve çözüme nasıl ulaşılabileceğinin belirlenmesinde yardımcı olmaktadır” (aktaran Yıldız ve Ekşisu, 2011: 191). Problemi çözerken amaç sadece sonuca ulaşmak olmamalıdır. Bir problemin çözümünde adım adım ilerlenmesi, belli bir yolun izlenmesi gerekmektedir.

### 2.2.3. Problem Çözme Basamakları

Problemi çözmeye başlamadan önce yapılması gereken ilk iş problemin farkında olmak ve onu anlamaktır. Ardından çözüm için bilgi toplamak gerekir. Çeşitli çözüm yolları ürettikten sonra en uygun olanı seçilmelidir. Problemin çözümü için seçilen yol uygulanır sonrasında ise çözüm değerlendirilerek süreç tamamlanır.

“Dewey’ in problem çözme yaklaşımına dayalı olarak Bingham tarafından belirlenmiş olan problem çözme aşamaları şunlardır:

- Problemin farkında olmak ve onunla uğraşma isteği duymak.
- Problemi açıklamak, ilgili olduğu alanı tanımak ve ilgili olduğu problemler grubunu anlamaya çalışmak.
- Problemle ilgili bilgileri toplamak, problemin çözümüne uygun düşecek bilgileri seçmek ve düzenlemek.
- Toplanan bilgiler ışığında muhtemel çözüm yolları belirlemek.
- Çözüm yollarını değerlendirerek en iyisini seçmek.
- Seçilen çözüm yolunu uygulamak.
- Kullanılan çözüm yolunu değerlendirmek.” (aktaran Saracaloğlu ve diğerleri, 2001: 123)

Stevens (1998), problem çözme sürecini şu şekilde ifade etmiştir: Problemin anlaşılması, gerekli bilgilerin toplanması, problemin köküne inilmesi, çözüm yollarının ortaya konulması, en iyi çözüm yolunun seçilmesi, problemin çözülmesi (aktaran Ünsal ve Ergin, 2011: 81).

Gür ve Hangül (2015: 97) ise çalışmalarında problem çözme basamaklarına şu şekilde yer vermişlerdir: Problemi anlama, bir plan hazırlama (uygun stratejiyi belirleme), hazırlanan planı uygulama ve çözümü değerlendirme.

Örneklere de görüldüğü gibi bir problemin çözümünde farklı aşamalar önerilmiştir. Problemin çözümünde ilk olarak problemin tanımlanması

gerekmektedir. Ardından çözüm yollarının bulunması, en uygun yola karar verilmesi ve uygulanması aşamaları takip edilmelidir. Unutulmaması gereken problemi çözerken bir formül ya da ezberlenmesi gereken bir yol bulunmamasıdır. Öğrenciler probleme çözüm getirirken çeşitli aşamaları kullanmaları ve problem çözme becerisi edinmeleri sağlanmalıdır.

#### **2.2.4. Problem Çözme Becerisi**

Kişilerin yaşamlarında karşılarına çıkacak problemleri çözmek için gerekli olan beceriye problem çözme becerisi denir. Öğrencilere bu beceri kazandırılırken onların ilgisini ve merakını çekecek, öğrenme düzeylerine uygun problemler verilmelidir.

Erkaper (2007: 13), çalışmasında şunlara yer vermiştir: Problem çözme becerisi, bireylerin farklı koşullara uyum sağlamalarına yardım eder. İnsanların buldukları ortama uyum sağlamaları için problem çözme becerilerini geliştirmeleri gereklidir.

Problem çözme becerisini öğrencilere kazandırabilmek için; onların aktif olabilecekleri öğrenme durumları oluşturup, yaşayarak öğrenmeleri sağlanmalıdır. Yeni durumlarla baş edebilecek beceri ve tutum geliştirmelerini sağlamak gerekmektedir. Öğrenci karşılaştığı problemi çözmek için uğraş verecek, bu süreçte yeri geldiğinde ön öğrenmelerini de işin içine katacaktır. Yeni durumla baş etme yollarını ararken farklı yollar deneyip, farklı disiplinlerden yararlanarak problemi çözmeye çalışacaktır. Öğrenci bu süreci etkin bir biçimde gerçekleştirirse problem çözme becerisine yönelik önemli kazanımlar elde edecektir.

#### **2.2.5. Problem Çözme Becerisini Kazanmış Bireyin Özellikleri**

Problem çözme becerisini kazanmış bireyin taşıması gereken özellikler nelerdir diye sorulduğunda, farklı cevaplar alınmaktadır. Çalışmalar incelendiğinde ise ortak noktalar şunlardır: Kendine olan özgüveni yüksektir, karşılaştığı zorlukları çözmeye yönelik olarak kendini yeterli hisseder, karar verme, eleştirel düşünme gibi zihinsel

becerileri yüksektir, düşüncelerini ifade eder, farklı yönlerden olaylara bakabilir, probleme karşı birden fazla çözüm getirebilir.

### **2.2.6. Problem Çözme Becerisinin Yararları ve Sınırlıkları**

Problem çözme becerisini edinmiş olan öğrenciler, süreçte etkin oldukları ve yaparak yaşayarak öğrendikleri için öğrenmeleri daha kalıcıdır. Öğrencilerde öğrenme sorumluluğu artar. Bir problemin çözümünde hangi adımları uygulayacaklarını, nasıl bir yol izleyeceklerini öğrenirler. Problem çözümü için düşünme, karar verme ve değerlendirme gibi becerilerini de geliştirirler. Problem durumları iyi belirlendiğinde, öğrencide ilgi ve istek uyandırabilir. Farklı kaynaklardan araştırma yaparak bilgiye ulaşırlar. Hata yaparak, hatalarından ders çıkararak doğruya ulaşabileceklerini öğrenirler.

Bu beceriyi kazandırmanın bir takım zorlukları elbette bulunmaktadır. Öğrencinin süreçte etkin olacağını düşünürsek içeriğin bu yönde hazırlanması ve uygulanması gerekecektir. Problem çözme becerisinin sınırlılıkları arasında şunları sayabiliriz: Öğretmen için; etkinlikleri ve ders planını hazırlama uzun ve yorucu bir süreçtir. Sürecin değerlendirmesi güç olabilir. Her ders için uygun problemler bulunması ve plan yapılması güçtür. Öğrenciler için; bazı durumlarda ek maliyet getirebilir. Problem durumu öğrencinin ilgisini çekmezse, onu çözülecek bir problem olarak algılamazsa herhangi bir ilerleme kaydedilemez.

### **2.2.7. Problem Çözme Becerisini Sınama**

Öğrencide gözlemlemeye karar verdiğimiz doğrudan ve dolaylı davranışları kazanıp kazanmadığını; kazandıysa ne ölçüde kazandığını, kazanmadıysa neden kazanmadığını, kazanabilmesi için eğitim sisteminde neler yapılması gerektiğini belirleme, sınama durumunun kapsamındadır (Sönmez, 2012). Öğrencilere kazandırılmak istenen problem çözme becerisinin ne kadar etkili olduğunu belirlemek için sürecin sonunda bir değerlendirme yapılmalıdır. Şahin (2004: 168), çalışmasında problem çözme becerisinin değerlendirilmesinin uzun süre

alabileceğinden bahsetmiştir. Değerlendirme için öğrencinin bir problemi çözme sırasında problemlere nasıl yaklaştığına bakmak gerekmektedir.

Değerlendirme aşaması sonuç değil süreç odaklı olmalıdır. Öğrenciye verilen bir problemi çözmesi onun bu beceriyi kazandığını göstermeye yetmeyebilir. Aynı zamanda süreçte istenilen basamakları ne kadar yerine getirdiğine, süreçteki gelişimine bakılmalıdır. Bir ders saati yerine daha uzun bir zamanda değerlendirme yapılmalıdır.

### **2.2.8. Problem Çözme Becerisi ve Bilişim Teknolojileri ve Yazılım Dersi**

Bilişim teknolojilerini sanattan sağlığa, güvenlikten eğitime pek çok farklı alanda kullanmaktayız. İlk, orta ve lise düzeyinde verilecek Bilişim Teknolojileri veya Bilgisayar Bilimleri dersleriyle öğrencilerin; yenilikleri takip eden, dünyayı anlayan, sorgulayan bireyler olarak yetiştirilmesi istenmektedir. Bu kapsamda farklı kuruluşlar öğrencilerin sahip olması gereken özellikleri belirlemeye yönelik çeşitli standartlar oluşturmuşlardır. Eğitimde Teknoloji için Uluslararası Topluluk (ISTE) tarafından öğrenciler, öğretmenler, Bilişim Teknolojileri öğretmenleri ve yöneticiler için eğitim teknolojilerinin kullanılmasında bazı standartlar belirlenmiştir. Öğrenciler için belirlenen standartlar aşağıdaki gibidir (ISTE, 2016):

1. Yetkin Öğrenen
2. Dijital Vatandaş
3. Bilgiyi Düzenleyen
4. Yaratıcı Tasarımcı
5. Bilişimsel Düşünen: Öğrenciler teknoloji yöntemlerinin çözüm geliştirme ve çözümleri test etme gücünden yararlanarak sorunları anlayacak ve çözecek stratejiler geliştirirler ve kullanırlar. Öğrenciler:
  - a) Çözüm arayışı sırasında veri analizi, soyut model ve algoritmik düşünce gibi teknoloji destekli yöntemlere uygun şekilde sorunların tanımını yaparlar.

- b) Veri toplarlar, konuyla ilgili veri setlerini belirlerler, dijital araçlarla verileri analiz ederler, verileri problem çözmeyi ve karar almayı kolaylaştıracak şekilde kullanırlar.
  - c) Sorunları bileşenlerine ayırırlar, kilit bilgilere erişirler, karmaşık sistemleri anlamak veya sorun çözmek için tanımlayıcı modeller geliştirirler.
  - d) Otomasyonun nasıl işlediğini kavrarlar, otomatik çözümler oluşturmak ve test etmek için gereken aşamaları geliştirmek üzere algoritmik yaklaşımdan yararlanırlar.
6. Yaratıcı İletişimci
7. Global İşbirlikçi

ISTE' nin belirlediği bilişimsel düşünme başlığında; öğrencilerden problemleri anlamak ve çözmek için teknolojik yöntemlerin gücünü kullanması, problemleri anlaması ve problem çözme stratejileri geliştirmesi beklenmektedir.

Türkiye'de öğrencilere liselerde Bilgisayar Bilimi dersleri, ortaokullarda ise Bilişim Teknolojileri ve Yazılım dersi verilmektedir. Öğrencilerin; bilgi ve iletişim teknolojilerini etik değerlere uygun, etkili ve üretken bir biçimde kullanabilmesi istenmektedir. Ortaokullarda dersin beş üniteden oluştuğunu görmekteyiz (MEB, 2018). Ünitelerin isimleri şu şekildedir:

- Bilişim Teknolojileri,
- Etik ve Güvenlik,
- İletişim Araştırma ve İş Birliği,
- Ürün Oluşturma,
- Problem Çözme ve Programlama

Dersin yenilenen öğretim programında da belirtildiği gibi teknolojiyi etkin bir biçimde kullanabilen, problemlere çözüm önerebilen, kendi yazılımlarını geliştirerek bilgiyi üretebilen bireyler olmaları amaçlanmaktadır (MEB, 2018). Özellikle

problem çözüme becerilerinin oluşması ve gelişmesi için; öğretim programında yer alan; Problem Çözme ve Programlama ünitesinden yararlanılmalıdır.

Durak (2009: 4), çalışmasında problem çözme ve programlama arasındaki ilişkiyi şu şekilde ifade etmiştir: “Polya’ nın problem çözme basamakları programlama konusunda şu şekilde özelleştirilebilir: Problemi tanımlama, algoritma geliştirme, kodlama ve değerlendirme, programın sınanması. Önce problemin ne olduğu tanımlanır, sonra işleyiş sırası belirlenir (algoritması hazırlanır), daha sonra uygun bir programlama dilinde yazılır ve son basamakta da kullanıma hazır hale gelen program sınanır.”

Problem çözümedeki farklı yolların belirlenmesi ve uygun olanının seçilmesi; algoritma geliştirmeye karşılık gelir. Algoritmayı şu şekilde tanımlayabiliriz: Problemin çözümüne ulaşmak için uygulanması gerekli adımların açık, düzenli ve sıralı bir şekilde, adım adım ifade edilmesidir. Çözüme yönelik birden fazla algoritma geliştirilebilir. İçinden uygun olanını seçerek çözümün uygulanması basamağına yani kodlama kısmına geçilir. Belirlenen algoritmaya uygun olarak bir programlama dili yardımıyla kod yazılır. Sonrasında ise yazılan kodların değerlendirilmesi aşamasına geçilir. Özelleştirilen adımlar incelendiğinde aslında problem çözme basamaklarıyla programlamanın aşamalarının birbirine benzediği görülür. Bilişim teknolojileri ve yazılım dersinde öğretilecek olan programlama konusu ile öğrencilerin problem çözme becerisine katkı sağlayacağını ifade edebiliriz.

Öğrencilerin dersler ile zihinsel becerilerini geliştirmek için uygun kazanımlar yazılmalıdır. Bazı problemlerin çözümünde birden fazla yol olabileceği gibi tek bir yol olabileceği de gözden kaçırılmamalıdır. Üzerinde durulması gereken diğer nokta problem çözerken belirlenmiş bir kuralın olmadığı, çözüm için belli strateji ve çeşitli adımların uygulanması gerektiğidir.

### 2.3. Tutum

Bilişim teknolojileri ve yazılım dersiyle öğrencilere kazandırılmak istenilen davranışları etkileyen faktörlerden biriside derse karşı tutumlarıdır. Öğrencilerin derse karşı olumlu tutum geliştirmeleri sağlanırsa, kazandırılmak istenen davranışlara ulaşmanın daha kolay olacağı söylenebilir. Bu yüzden tutum kavramını daha iyi anlamak ve bilmek büyük önem taşımaktadır.

Tutum, insanın etrafındaki bir duruma, bir olaya ya da bir nesneye yönelik olarak gösterdiği davranış, tavidir. Hem somut hem de soyut kavramlar tutumla ilişkilendirilebilir. İnsan, bindiği arabadan tuttuğu takıma, kapı komşusundan okulda gördüğü derse kadar çeşitli durumlarda farklı davranışlar sergiler. Doğrudan gözlemleyemediğimiz tutumu ancak davranışa dönüştüğü zaman fark edebiliriz. Bireyin davranışlarında herhangi bir değişiklik gözlemleniyorsa, bireyin bir tutuma sahip olduğunu söyleyebiliriz. Araştırmalarda tutum kavramı için farklı tanımlar bulunmaktadır.

Hoog ve Vaughan (2005) tutumu, doğrudan gözlenebilir olmasa da, bir davranışı belirleyen eyleme ilişkin seçim ve kararlarımıza yön veren yapı olarak tanımlanmaktadır (aktaran Özgürol, 2010).

Senemoğlu (2013: 417) tutumu, bireyin herhangi bir grup şeye, bireylere, olaylara ve çok çeşitli durumlara karşı bireysel etkinliklerindeki seçimini etkileyen kazanılmış içsel bir durum olarak tanımlamıştır.

İnceoğlu (2010: 13), bireyin kendine ya da çevresindeki herhangi bir nesne, toplumsal konu, ya da olaya karşı deneyim, bilgi, duygu ve güdülerine dayanarak örgütlediği zihinsel, duygusal ve davranışsal bir tepki ön eğilimi olarak tanımlamıştır.

Tutumla ilgili olarak şunları söyleyebiliriz:

- Tutum, doğrudan gözlenemez.
- Tutum sonradan oluşur, kişinin sahip olduğu bilgi değişirse sahip olduğu tutumu da değişebilir,

- Tutum, olumlu ya da olumsuz olabilir.
- Tutum ile davranış arasında uyum olmalıdır.

Tutum; zihinsel, duygusal ve davranışsal olmak üzere üç faktörden meydana gelir (İnceoğlu, 2010). Tutumun oluşabilmesi için üç ögenin uyumlu bir şekilde çalışması gerekir.

- Zihinsel (bilişsel) Öge; bir olgu ya da durum hakkında sahip olunan bilgidir. Kişinin bilgisi değiştiğinde tutumu da değişecektir. Örneğin; sigaranın sağlığa zararlı olduğunu bilmesi
- Duygusal (duyuşsal) Öge; bir olgu ya da duruma yönelik hisler, duygulardır. Bu öge diğer ikisinden bağımsız değildir. Bireyin herhangi bir durum karşısında olumlu ya da olumsuz duygulara sahip olması önceki deneyimleriyle ilgilidir. Yani birey önceden bu uyarıcılarla karşılaşmıştır. Örneğin; sigaranın kokusundan rahatsız olması ve hoşlanmaması.
- Davranışsal Öge; bireyin belli bir uyarıcı grubundaki tutum konusuna karşı davranış eğilimini yansıtır. Bu davranış eğilimleri sözler ya da diğer hareketlerden gözlemlenebilir. Örneğin; sigarayı hiçbir zaman kullanmaması.

Yapılan açıklamalara göre tutum, bireyin önceki yaşantılarına bağlı olarak oluşmakta ve değişmektedir. Tutumun oluşmasında bireyin sahip olduğu özellikler, önceki yaşantıları, sosyal çevresi gibi farklı etkenler bulunmaktadır. İnsanların farklı yollarla tutum geliştirdiklerini, bu tutumların değişken olduğunu söyleyebiliriz. Sahip olunan tutumlar değişime direnç göstermektedirler. Değişime gösterilen dirence rağmen, tutumlar yavaş olarak, yeni bilgi ve deneyimler edindikçe değişirler (Tavşancıl, 2014: 81).

Bireyin herhangi bir şeye karşı olan tutumunu ancak davranışa dönüştüğünde gözlemleyebiliriz. Tutumları belirleyebilmek için davranışı gözlemleyebilir, kişi ile görüşebilir veya tutumun ölçülmesi için geliştirilmiş olan bazı ölçekleri kullanabiliriz. Bu ölçeklerden bazıları şunlardır: Bogardus Ölçeği, Thurstone Ölçeği, Gutman Ölçeği, Likert Ölçeği. Tavşancıl (2005: 115-167) çalışmasında ölçeklerle ilgili şu bilgilere yer vermiştir:

**Bogardus Ölçeği:** Sosyal uzaklık ölçeği olarak geliştirilmiştir. Farklı topluluklardan olan kişilerin birbirleri ile ilişkilerinin kabul veya red derecelerini kıyaslamak amacıyla geliştirilmiştir. Herhangi bir grubun toplumsal bakımdan benimsenme derecesini ölçmek üzere seçilmiş maddelerden oluşur.

**Thurstone Ölçeği:** Eşit görünen aralıklar tekniğine göre düzenlenmiş tutum ölçeğidir. Ölçekte yer alan ifadeler olumludan olumsuzu doğru sıralanmıştır. Bireyler ifadelerden olumlu bulduklarını işaretlemektedirler. Bireylerin işaretledikleri değerlerin ortalama değeri hesaplanarak bulunan değer, ölçekten alınan puanı vermektedir.

**Gutman Ölçeği:** Tek boyutlu bir ölçektir. Bireylere ölçekte yer alan her ifadeyi kabul edip etmedikleri sorulur. Bireyin ölçekten aldığı puan, kabul ettiği ifadelerin sayısını vermektedir. Bu bakımdan ölçek, birikimli ölçekler sınıfında yer almaktadır.

**Likert Ölçeği:** Bu ölçekte, tutumları belirlenecek bireylerin tepkide bulunacakları çeşitli ifadeler bulunur. Birey verilen ifadelere katılıp katılmadığını belirtmek için maddelere önceden belirlenmiş puanı verir. Ölçekten alınan puan, bireyin ifadelere verdiği puanların toplamından oluşur.

## 2.4. Yapılan Çalışmalar

### 2.4.1. Programlama ile İlgili Yapılan Çalışmalar

Genç ve Tınmaz (2010), üniversite öğrencileriyle yaptığı çalışmada öğrencilerin programlama dilleri dersindeki problem çözme becerilerini ve bilgisayara yönelik tutumlarını incelemişlerdir. Araştırma sonucunda alınan programlama dilleri dersinin öğrencilerin problem çözme becerilerini ve bilgisayara yönelik tutumlarını arttırmada etkisinin olmadığı bulunmuştur. Bunun sebebinin seçilen programlama dilinin yapısından kaynaklandığı ve programlama dili dersinde çözülen problemlerin genelde matematiksel problemler üzerine kurulduğu, dolayısıyla öğrencilerin günlük problemlerle ilişkilendirememesi olarak belirtilmiştir.

Nam, Kim ve Lee (2010), çalışmalarında Scratch ile programlama öğrenmeyi kolaylaştırmaya yönelik, yapı-iskelesi yaklaşımına uygun dersler hazırlayarak

sürecin öğrencilerin problem çözme becerilerine etkisini incelemiştir. Altıncı sınıf öğrencileriyle gerçekleştirilen çalışmanın sonuçları şöyledir: Her iki grubunda problem çözme becerilerinde önemli bir artış gözlenmiştir. Yapı iskelesi yaklaşımına uygun ders işlenen grubun daha iyi ilerleme gösterdikleri tespit edilmiştir.

Genç ve Karakuş (2011), çalışmalarında Eğitimde Bilgisayar Oyunları Tasarımı dersinde öğrencilerin tasarım sürecine aktif katılımlarını sağlamak, matematiksel ve kompütasyonel becerilerinin gelişimini sağlamak amacıyla grafiksel bir programlama dili olan Scratch' i kullanmışlardır. Üniversite öğrencileriyle yapılan çalışmada, öğrencilerin Scratch hakkında olumlu görüşlere sahip oldukları, tasarımla öğrenmenin kalıcı bir öğrenme sağladığı ortaya çıkmıştır.

Özoran ve diğerleri (2012), çalışmalarında mühendislik öğrencilerine Programlamaya Giriş dersini Scratch programını kullanarak vermişlerdir. Dersin uygulama aşamasında program tanıtılmış ve projeler geliştirilerek ders işlenmiştir. İlgili dersin bir önceki yılda alınan final sınav sonuçlarını karşılaştırmak için, final soruları benzer hazırlanmıştır. Başarısız olan öğrenci sayısında azalma belirlenmiştir. Öğrenciler; Scratch ile programlamanın daha kolay ve anlaşılır olduğunu, ikinci dönemde dersi bu program ile almak istediklerini, programlamaya yeni başlayacaklara da bu yazılımı önereceklerini belirtmişlerdir.

Çetin (2012), programlama eğitiminin öğrencilerin problem çözme becerileri üzerinde etkisinin olup olmadığını incelediği yüksek lisans tez çalışmasını 5. sınıfta öğrenim gören öğrencilerle gerçekleştirmiştir. Araştırmanın sonucunda çocuklar için bilgisayar programlama eğitiminin uygulanabilir olduğu, programlama eğitiminin çocukların problem çözme becerilerine olumlu yönde katkı sağladığı belirlenmiştir. Öğrenciler programlama eğitiminden memnun kaldıklarını, bilgisayarla neler yapabileceklerine ilişkin düşüncelerinde olumlu yönde değişiklikler olduğunu ve bu tür eğitimlere devam etmek istediklerini belirtmişlerdir. Veliler ise bilgisayar programlama eğitiminin çocuklarının bilgi ve becerilerine katkı sağladığını, çocuklarının bilgisayar kullanımında olumlu yönde değişiklikler olduğunu ve böyle eğitimlere ihtiyaçlarının olduğunu belirtmişlerdir.

Coşar (2013), doktora tez çalışmasında problem temelli bir öğrenme ortamında ilköğretim yedinci sınıf öğrencilerinin programlama çalışmalarının akademik başarılarına, eleştirel düşünme eğilimlerine ve bilgisayara yönelik tutumlarına etkisini incelemiştir. Araştırma yedinci sınıfta eğitim gören öğrencilerle yapılmıştır. Öğrencilerin probleme dayalı öğrenmelerini gerçekleştirmek için web tabanlı hazırlanan bilgisayar programlama dersi kullanılmıştır. Araştırma sonucunda problem temelli öğrenme ortamında 7. sınıf öğrencilerine verilen programlama eğitiminin, onların programlama akademik başarılarına, eleştirel düşünme eğilimlerine ve bilgisayara yönelik tutumlarına olumlu katkılar sağladığı belirtilmiştir.

Kalelioğlu ve Gülbahar (2014), çalışmalarında Scratch ile programlamanın 5. sınıf öğrencilerinin problem çözme becerilerine etkisini araştırmışlardır. Ayrıca, çalışmaya katılan öğrencilerin programlama hakkındaki düşüncelerini de almışlardır. Araştırma sonucuna göre Scratch ile programlama, öğrencilerinin problem çözme becerilerinde anlamlı bir farklılık yaratmamıştır. Öğrencilerin programlamayı sevdiklerini ve programlarını geliştirmek istediklerini belirtmişlerdir.

Nikou ve Economides (2014), lise öğrencileriyle yürüttükleri çalışmada Scratch ve AppInventor araçlarını kullanmışlardır. Bu programlama araçlarının öğrencilerin programlamayı öğrenmeye yönelik motivasyonlarına etkisi incelenmiştir. Çalışmanın sonucunda öğrencilerin programlamaya yönelik; içsel amaç yönelimi, öğrenme inançlarının kontrolü ve öz yeterliklerini arttırdığı fakat dışsal motivasyon üzerinde bir etkisinin olmadığı bulunmuştur.

Solmaz (2014), doktora tezinde programlama dili öğretiminde Alice yazılımının ders başarısı, eleştirel düşünme, problem çözme becerileri ile üst bilişsel farkındalık düzeyine etkisini araştırmıştır. Üniversite öğrencileri ile gerçekleştirilen çalışmada deney grubuna 7 hafta boyunca Alice yazılımı kullanılarak PHP programlama dilinde temel programlama kavramları öğretilmiş, kontrol grubuna ise PHP programlama dilinde temel kavramlar herhangi bir yardımcı araç kullanılmadan öğretilmiştir. Çalışmanın sonucunda Alice programının PHP programlama dili

öğretiminde ders başarısı üzerinde çok fazla etkili olmadığı ifade edilmiştir. Alice programı kullanımının öğrencilerin eleştirel düşünme eğilimi, problem çözme becerisi ve üst bilişsel farkındalık değişkenleri üzerinde anlamlı bir etki yaratmadığı bulunmuştur. Alice programını kullanan öğrenciler ile yapılan görüşmede; Alice yazılımının programlama veya algoritma mantığını kavramaya, programlamanın temel kavramlarını öğrenmeye yardımcı olduğu ve programlamayı eğlenceli hale getirdiği belirtilmiştir.

Tekerek ve Altan (2014), çalışmalarında ilköğretim 6. sınıf Bilgi ve İletişim Teknolojileri dersinde Scratch ortamının algoritma öğretimindeki etkisini incelemiştir. Çalışmada deney ve kontrol gruplarının son test başarı puanları arasında istatistiksel olarak anlamlı bir fark bulunmamıştır. Benzer şekilde, cinsiyet değişkeni açısından deney ve kontrol gruplarının son test puanları arasında da istatistiksel olarak anlamlı bir fark tespit edilmemiştir.

Akçay(2015), yüksek lisans tez çalışmasında Bilişim Teknolojileri öğretmen adaylarının programlamaya ilişkin öz yeterlik düzeylerini, problem çözme beceri düzeylerini ve sorgulama beceri düzeylerini incelemiştir. Araştırma sonucunda öğrencilerin programlamaya ilişkin öz yeterliklerinin orta düzeyde olduğu bulunmuştur. Öğrencilerin sorgulama becerilerine sahip olduğu ve problem çözme becerilerinin yüksek olduğu tespit edilmiştir. Çalışmadan elde edilen bulgulara göre öğrencilerin programlamaya ilişkin öz yeterlik algıları ile problem çözme becerileri arasında bir farklılık tespit edilememiştir. Çalışma sonucunda elde edilen bir diğer sonuç ise öğrencilerin sorgulama becerileri ile programlamaya ilişkin öz yeterlik düzeyleri arasında anlamlı bir farklılığın olmasıdır.

Calao ve diğerleri (2015), çalışmalarında ilköğretim 6. sınıf öğrencilerinin matematiksel becerileri geliştirilirken kullanılan Scratch yazılımının öğrencilerin sayısal düşünme becerisine etkisini incelemiştir. Üç ay süren çalışmada sonuçlar şu şekilde ifade edilmiştir: Scratch ile eğitim alan deney grubunun matematik bilgisinde istatistiksel olarak anlamlı bir fark bulunmuştur. Bu nedenle, Scratch görsel programlama ortamı kullanılarak hesaplamalı düşünmenin geliştirilmesinin,

ilköğretim öğrencilerinin matematiksel modelleme, akıl yürütme, problem çözme ve egzersiz süreçleri açısından performanslarını geliştirmelerine olanak sağladığı sonucuna varılmıştır.

Ouahbi ve diğerleri (2015), araştırmalarında öğrencilere programlama temellerini öğretmek için Scratch programını kullanmışlar ve öğrencilerin programlamaya yönelik motivasyonunu araştırmışlardır. Lise öğrencileri ile yapılan çalışmanın öncesinde ve sonrasında, öğrencilerin ilgi düzeylerini belirlemek için anket uygulanmıştır. Çalışmanın sonucunda, Scratch programını kullanılan gruptaki öğrencilerin diğer gruptaki öğrencilere göre başarı ve motivasyonlarında anlamlı bir artış olduğu bulunmuştur. Ayrıca öğrencilere programlama çalışmalarına devam etme istekleri sorulduğunda, Scratch programı ile ders gören öğrencilerin % 65'i, programlamaya devam etmeyi düşünürken, diğer gruplarda öğrenim gören öğrencilerin sadece % 10,3' ü programlamaya devam etmek istediklerini belirtmişlerdir.

Korkmaz (2016), çalışmasında Scratch tabanlı oyun aktivitelerinin öğrencilerin bilgisayar programlamayı öğrenmeye yönelik tutumları, öz-yeterlik inançları ve akademik başarı düzeylerine olan etkilerini araştırmıştır. Çalışma grubu mühendislik fakültesinde öğrenim gören öğrencilerden oluşmuştur. Deney grubuna Scratch tabanlı oyun aktiviteleri ile etkinlikler yapılırken, kontrol grubunda geleneksel yöntemle C ++ programlama dili konuları anlatılmıştır. Çalışmanın sonucunda ise Scratch tabanlı etkinliklerin öğrencilerin tutumları ve öz-yeterlik algıları üzerinde herhangi bir anlamlı etki yaratmadığı bulunmuştur. Scratch ile C++ dersleri verilen grubun akademik başarılarının, geleneksel yöntemle işlenen grubun akademik başarılarına göre daha yüksek olduğu belirlenmiştir.

Yünkül ve diğerleri (2017), Scratch yazılımının öğrencilerin bilgisayarca düşünme becerilerine etkisini araştırdıkları çalışmalarında deney grubuna bir dönem boyunca Scratch eğitimi verilmiştir. Dönem sonunda her iki grupta yer alan öğrencilere BDBÖ (Bilgisayarca Düşünme Becerileri Ölçeği ) uygulanmış ve elde edilen verilerden Scratch eğitimi alan öğrencilerin daha yüksek BDBÖ puanlarına

ulaştıkları görülmüştür. Scratch programının kullanımının problem çözme, algoritmik düşünce ve yaratıcı düşünme becerileri üzerinde anlamlı bir etkisi olduğu sonucuna ulaşılmıştır.

Korkmaz (2018), çalışmasında Scratch ve Lego Mindstorms Ev3 ile yapılan programlama etkinliklerinin öğrencilerin; problem çözme becerilerine, mantıksal-matematiksel düşünme becerilerine ve akademik başarıya yönelik etkisini araştırmıştır. Çalışmada birinci grupta, Scratch tabanlı aktiviteler uygulanmıştır. İkinci grupta ise Lego Mindstorms Ev3 tabanlı tasarım etkinlikleri uygulanmıştır. Kontrol grubunda ise C++ tabanlı etkinlikler uygulanmıştır. Uygulama sonucunda Scratch ile ilgili oyun etkinliklerine dayanan eğitim programının, ikinci ve kontrol gruplarına göre mantıksal-matematiksel düşünme becerisine daha olumlu bir katkı yaptığı bulunmuştur. Ayrıca Lego Mindstorms Ev3 ile tasarım etkinliklerine dayanan eğitim programı öğrencilerin problem çözme becerilerine, Scratch ile ilgili oyun etkinliklerine ve geleneksel öğretim etkinliklerine dayanan eğitim programlarından daha olumlu katkı sağladığı bulunmuştur.

Dinçer (2018), yüksek lisans tezinde Scratch ve Kodu Game Lab ile programlama öğretiminin 6. sınıf öğrencilerinde programlama akademik başarısı, tutum ve öz-yeterlik algıları üzerine olan etkisini incelemiştir. Scratch ile eğitim alanların, Kodu Game Lab ile eğitim görenlere göre başarılarının anlamlı derecede arttığı belirlenmiştir. Scratch ve Kodu Game Lab ile öğrenim gören öğrencilerin, tutum ve öz-yeterlik algıları arasında anlamlı bir fark olmadığı saptanmıştır. Her iki grupta da öğrencilerin programlama dillerine karşı olan öz-yeterlik algılarının arttığı ifade edilmiştir. Ayrıca Kodu Game Lab ile programlama eğitiminin, tutumu da arttığı ortaya çıkmıştır.

Şimşek, E. (2018), çalışmasında görsel programlama ve robotik programlama etkinliklerinin, programlama öğretimi sürecinde öğrencilerin bilgi işlemsel düşünme becerilerine ve akademik başarılarına etkisini araştırmıştır. Araştırmada öğrencilere bir ay boyunca görsel programlama ve robotik eğitimi verilmiştir. Daha sonra gruplar yer değiştirilerek 1. gruba mBlock, 2. gruba Scratch eğitimi verilmiştir. Eğitim

sonrası ölçüm tekrarlanmıştır. Araştırma sonunda hem akademik başarıda hem de bilgi işlemsel düşünme pratiklerinde iki grubun da eşdeğer puanlar aldığı gözlemlenmiştir. Sonuçlar programlama temel eğitimi için iki araçta kullanılabilir olduğu şeklinde yorumlanmıştır.

#### **2.4.2. Problem Çözme Becerisi ile İlgili Yapılan Çalışmalar**

Özsoy (2005), çalışmasında ilköğretim 5. sınıfta problem çözme becerisi ile matematik dersi başarısı arasındaki ilişkiyi incelemiştir. Araştırma sonunda; ilköğretim 5. sınıf matematik başarısı ile problem çözme becerisi arasında anlamlı ve pozitif yönde bir ilişki bulunduğu bulunmuştur.

Altun ve Emir (2008), çalışmalarında geleneksel öğretim ve problem çözme yöntemlerinin öğrencilerin erişim düzeyine, kalıcılığa ve derse karşı tutumlarına etkisini araştırmışlardır. Çalışma 6. sınıfta öğrenim gören öğrenciler ile yapılmıştır. Grupların toplam erişim ve kalıcılık ortalama puanları arasında deney grubunun lehine anlamlı bir fark olduğu tespit edilmiştir. Araştırmada problem çözme yönteminin öğrencilerin sosyal bilgiler dersine yönelik tutumlarını artırdığı bulunmuştur.

Kaş (2010), çalışma yaprakları kullanılarak yapılan öğretimin 8. sınıf öğrencilerinin cebir problemlerini çözme ve cebirsel düşünme becerilerine etkisini araştırmıştır. Öğretim çalışmaları sonrasında, çalışma yaprakları ile yapılan öğretimin, öğrencilerin cebirsel problem çözme ve cebirsel düşünme becerilerine olumlu etki yaptığı görülmüştür. Bu etki cebirsel problem çözme becerisinde geleneksel öğretim yöntemine göre daha anlamlı bulunmuştur.

Çetin (2012), araştırmasında programlama eğitiminin öğrencilerin problem çözme becerileri üzerine etkisinin olup olmadığını incelemiştir. Araştırmanın sonucunda programlama eğitiminin çocukların problem çözme becerilerine olumlu yönde katkı sağladığı belirlenmiştir. Öğrenciler ise programlama eğitiminden memnun kaldıklarını, bilgisayarla neler yapabileceklerine ilişkin düşüncelerinde

olumlu yönde deęişiklikler olduğunu ve bu tip eęitimlere devam etmek istediklerini belirtmişlerdir.

Saracaloęlu ve Kanmaz (2012), üniversite öğrencileriyle yaptıkları çalışmada öğrencilerin problem çözme becerilerini belirleyerek ve bu beceriyi farklı deęişkenler açısından incelemiştir. Araştırma sonucunda; öğrencilerin genel olarak problem çözme becerilerinin ve problem çözme yaklaşımlarının orta düzeyde olduğu belirlenmiştir. Öğrencilerin problem çözme becerisinin cinsiyet, sosyo-ekonomik düzey deęişkenlerine göre anlamlı düzeyde farklılaşmadığı tespit edilmiştir

Kuloęlu ve Arı (2014), çalışmada Fen ve Teknoloji bölümündeki öğretmen adaylarının problem çözme becerilerini; cinsiyet, sınıf düzeyi, öğretim türü, anne-baba eğitim durumları, yaşadıkları ve kaldıkları yere göre incelemiştir. Araştırmadan elde edilen bulgulara göre öğretmen adaylarının problem çözme becerilerinin; cinsiyete, anne-baba eğitim durumuna, kaldıkları ve yaşadıkları yere göre anlamlı farklılık göstermediği bulunmuş ancak erkeklerin problem çözme becerilerinin bayanlara göre yüksek düzeyde olduğu görülmüştür.

Solmaz (2014), doktora tezinde programlama dili öğretiminde Alice yazılımının ders başarısı, eleştirel düşünme ve problem çözme becerileri ile üst bilişsel farkındalık düzeyine etkisini araştırmıştır. Araştırma sonucunda eleştirel düşünme eğilimi, problem çözme becerisi ve üst bilişsel farkındalık deęişkenlerine ait sonuçlara göre Alice programı kullanımının bu deęişkenler üzerinde anlamlı bir etki yaratmadığı bulunmuştur.

## BÖLÜM III

### YÖNTEM

Bu bölümde araştırmanın modeli, çalışma grubu, veri toplama araçları, verilerin toplanması ve verilerin analizi açıklanmıştır.

#### 3.1. Araştırmanın Modeli

Çalışmada, Scratch programı ile işlenen derslerin, öğrencilerin başarısı, problem çözme becerisi ve Bilişim Teknolojileri dersine yönelik tutum puanları üzerindeki etkisi araştırılırken; ön test, son test tek grup deneysel desen ile nitel araştırma yöntemi kullanılmıştır. Bu bakımdan açıklayıcı yaklaşım işe koşulmuştur. Scratch ile programlama öğretimine ilişkin öğrencilerin görüşlerine de başvurulmuştur. Araştırmanın deneysel modelinin simgesel görünümü aşağıdaki tabloda gösterilmiştir.

**Tablo 3. Çalışmada Kullanılan Modelin Simgesel Gösterimi**

Grup	Ön test	Yapılacak İşlemler	Son Test
G	T1	Scratch programı	T1
	T2	ile derslerin	T2
	T3	işlenmesi	T3

G simgesi, çalışma grubunu ifade etmektedir. Gruba deneysel çalışmadan önce bazı testler uygulanmıştır. Bunlar;

T1: Başarı Testi

T2: Problem Çözme Ölçeği

T3: Bilişim Teknolojileri Dersine Yönelik Tutum Ölçeği

Deneysel çalışmanın bitiminde çalışma grubuna yukarıda belirtilen testler tekrar uygulanmıştır.

### **3.2. Çalışma Grubu**

Araştırmada evren ve örneklem tayinine gidilmeyip uygun çalışma grubu kullanılmıştır. Araştırmanın çalışma grubu 2017-2018 eğitim öğretim yılında Konya ili Seydişehir ilçesinde bir devlet okulunda Bilişim Teknolojileri dersini alan 6. sınıf öğrencilerinden oluşmuştur. Uygulamaya 22 öğrenci katılmıştır.

### **3.3 Veri Toplama Araçları**

Çalışmanın nicel verileri başarı testi, problem çözme ölçeği, Bilişim Teknolojileri dersine yönelik tutum ölçeği ile toplanmıştır. Çalışmanın nitel verileri ise görüşme formu ile toplanmıştır.

#### **3.3.1. Başarı Testi**

Çalışmaya katılan öğrencilerin erişim düzeylerini belirlemek için kullanılan başarı testi, araştırmacı tarafından hazırlanmıştır (Ek 1). Bu testte farklı tipte sorulara yer verilmiş olup, sorular ders kitabı, ilgili siteler ve farklı akademik kitapların araştırılması sonucunda elde edilmiştir. Başarı testi hazırlanırken önce hedef-davranışlar belirlenmiştir. Ardından davranışlara uygun olarak belirtke tablosu hazırlanmıştır. Belirtke tablosu Tablo 4’ te verilmiştir. Davranışlara uygun olarak 30 soru hazırlanmıştır. Bu soruların uygunluğunun belirlenmesi için üç uzmanın görüşü alınmıştır. Çalışmanın güvenilirlik hesaplaması Miles ve Huberman (1994), uyum yüzdesi formülü kullanılarak % 79 bulunmuştur. Hazırlanan testin güvenilirliği hesaplanmış ve güvenilirlik katsayısı 0.814 olarak hesaplanmıştır.

**Tablo 4. Belirtke Tablosu**

Kazanım İçerik	Kavram Bilgisi	İlkeler Bilgisi	İlkeleri Açıklayabilme	Problemlerin Çözümünde İlkeleri Kullanabilme	Toplam
Programlama ilgili temel kavramlar	2	2		3	7
Doğrusal mantık yapısı		2	1	3	6
Döngü yapısı		4	1	3	8
Karar yapısı		3	1	3	7
Farklı durumlara uygun kod bloklarını kullanarak problemleri çözebilme				2	2
					30

### 3.3.2. Problem Çözme Ölçeği

Yapılan sekiz haftalık çalışmanın başlangıcında ve bitiminde öğrencilerin problem çözme becerilerinde anlamlı bir değişim olup olmadığını belirlemek için “Problem Çözme Ölçeği” kullanılmıştır (Ek 2). Kullanılan bu ölçek, Kardeş ve diğerlerinin (2014), “Problem Çözme Envanterini İlköğretim Öğrencilerine Uyarlama Çalışması: Doğrulayıcı Faktör Analizi Sonuçları” adlı çalışmalarından alınmıştır.

İlgili çalışma, Şahin, Sahin ve Heppner (1993) tarafından Türkçeleştirilen ve yetişkinlerin problem çözme becerilerini ölçmeye yönelik psikometrik bir araç olan PÇE’ nin beşinci sınıf öğrencilerine uyarlanmasıdır. Ölçek toplam 20 maddeden oluşmaktadır. Ölçek 4’lü likert tipinde olup, maddeler “Tamamen Katılıyorum = 4”, “Çoğunlukla Katılıyorum = 3”, “Ara Sıra Katılıyorum = 2” ve “Hiç Katılmıyorum = 1” şeklindedir. Ölçeğin Cronbach Alfa güvenilirliği hesaplanmış ve güvenilirlik katsayısı 0,608 olarak bulunmuştur.

### 3.3.3. Bilişim Teknolojileri Dersine Yönelik Tutum Ölçeği

Öğrencilerin derse yönelik tutumlarında anlamlı bir değişiklik olup olmadığını belirlemek için, uygulamanın başlangıcında ve bitişinde “Bilişim Teknolojileri Dersine Yönelik Tutum Ölçeği” kullanılmıştır (Ek 3). Araştırmada kullanılan bu ölçek Işık (2010)’ın doktora çalışması için geliştirmiş olduğu tutum ölçeğidir.

Araştırmada kullanılan tutum ölçeğinde bulunan 36 maddeden 19’u olumsuz, 17’si ise olumlu madde şeklinde oluşturulmuştur. Ölçek, 3’lü Likert tipinde, “Katılmıyorum = 1”, “Kararsızım = 2” ve “Katılıyorum = 3” olarak düzenlenmiştir. Maddelerin puanlandırılması ise Tablo 5’te gösterilmiştir. Ölçeğin Cronbach Alfa güvenilirliği hesaplanmış ve güvenilirlik katsayısı 0,945 olarak belirlenmiştir.

**Tablo 5. Bilişim Teknolojileri Dersine Yönelik Tutum Ölçeği Maddelerinin Puanlanması**

	Puan	
	Olumlu Madde	Olumsuz Madde
Katılıyorum	3	1
Kararsızım	2	2
Katılmıyorum	1	3

### 3.3.4. Yarı Yapılandırılmış Görüşme Formu

Araştırmanın nitel boyutunda çalışma grubundaki öğrencilerin, uygulamaya yönelik görüşleri alınmıştır. Elde edilen görüşlere betimsel analiz yapılmış ve açıklanıp yorumlanmıştır. Betimsel analiz, nitel verilerin analizinde kullanılan en basit analiz şeklidir. Araştırmacı soruları rapor haline getirirken aynen yazmamalıdır. Soruları başlık, kategori altında toplamalıdır (Sönmez ve Alacapınar, 2014). Yarı yapılandırılmış görüşme formu araştırmacı tarafından hazırlanmış açık uçlu dört sorudan oluşmaktadır.

### 3.4. Verilerin Toplanması

Sekiz hafta süren uygulama sürecinin başlangıcında öğrencilere başarı testi, problem çözme ölçeği ve Bilişim Teknolojileri dersine yönelik tutum ölçeği uygulanmıştır. Uygulama için yöneticilere haber verilmiş, Konya İl Milli Eğitim Müdürlüğü'nden izin alınmıştır (Ek 5). Uygulama, bir ders 40 dakika olmak üzere haftada iki ders olarak toplam sekiz hafta sürmüştür. Hazırlanan hedef-davranış listesi Ek 8' de verilmiştir. Araştırma için belirlenen hedef-davranışlara uygun olarak ders planları hazırlanmıştır. Planlardan birisi Ek 6' da verilmiştir. Planda uygulanan çalışma kâğıdı Ek 7' de verilmiştir.

İlk beş hafta boyunca bilgi ve kavrama düzeyindeki davranışlara yönelik olarak çalışmalar yapılmıştır. Öğrencilere programlamadaki temel kavramlar, doğrusal mantık, döngü mantık ve karar mantık yapıları ile ilgili konular anlatılmıştır. Konular anlatıldıktan sonra öğrencilere örnek çalışmalar gösterilmiştir. Ardından konuyla ilgili örnek bir uygulama yapmaları istenmiştir. Öğrenciler uygulamalarını yaparken onlara rehberlik edilmiştir. Yapılan etkinlik isimleri şu şekildedir:

- Yön Tuşları
- Akvaryum
- Avcı Köpekbalığı
- Top Oyunu

Son üç hafta ise uygulama düzeyindeki davranışlara yönelik çalışmalara yer verilmiştir. Öğrencilerden anlatılan hedefleri kapsayan, ilgilerini çeken bir konu hakkında proje hazırlamaları istenmiştir. Uygulamanın bitiminde öğrencilerin görüşlerini almak için hazırlanan yarı yapılandırılmış görüşme formu ise Ek 4' te verilmiştir. Uygulamanın bitiminde çalışma grubuna başarı testi, problem çözme ölçeği ve tutum ölçeği son test olarak tekrar uygulanmıştır.

### 3.5. Verilerin Analizi

Çalışma grubuna uygulanan başarı testinin puanlaması aşamasında doğru cevaplara 1, yanlış cevaplara 0 değeri verilmiştir. Testte 1. sorunun sekiz, 2. sorunun yedi alt maddesi bulunmaktadır. Testten alınacak en yüksek puan 43' tür. Öğrencilerin başarı testi, problem çözme ölçeği ve Bilişim Teknolojileri dersine yönelik tutum ölçeğinden aldıkları ön test ve son test puanları arasında anlamlı bir fark olup olmadığını belirlemek için İlişkili Örneklem t Testi uygulanmıştır. Öğrencilerin görüşlerini belirlemek için ise yarı yapılandırılmış görüşme formu kullanılarak elde edilen veriler, betimsel analiz tekniği kullanılarak analiz edilmiştir.

## BÖLÜM IV

### BULGULAR VE YORUM

Bu bölümde araştırmanın denencelerine ve alt problemine ilişkin yanıtlarını bulmak için ölçme araçlarıyla toplanan verilere ait bulgular verilmiş; bu bulgulara göre yorumlamalar yapılmıştır.

#### 4.1. Birinci Denenceye İlişkin Bulgular

Araştırmanın birinci denencesi “**Scratch programı uygulanan grubun başarı testinden aldıkları ön test, son test puanları arasında anlamlı bir fark vardır.**” denencesi ile ilgili bulgular incelenerek, Tablo 6’ da gösterilmiştir.

**Tablo 6. Başarı Testi Ön Test ve Son Test İlişkili Örneklem t-Testi Sonuçları**

	n	$\bar{X}$	SS	t	p
Ön-Test	22	11,14	4,357	-9,893	0,00*
Son-Test	22	23,82	7,248		

\*(p<0.05)

Öğrencilere, uygulamanın başlangıcında ve bitiminde yapılan başarı testinden aldıkları puanlar arasında anlamlı bir fark olup olmadığını belirlemek için ilişkili örneklem t-testi yapılmıştır. Uygulama öncesi alınan sınav puanları ( $\bar{X}_{\text{öntest}} = 11,14$ ) ile uygulama sonrası alınan sınav puanları ( $\bar{X}_{\text{sontest}} = 23,82$ ) arasında, t değeri 21 serbestlik derecesinde  $-9,893$  [ $t_{(22)} = -9,893$ ,  $p = 0,00 < 0,05$ ] olarak hesaplanmıştır. p değerine bakıldığında 0,05 manidarlık düzeyinde anlamlı bir fark görülmüştür. Bu durum, yapılan çalışmanın öğrencilerin deney öncesi ve sonrası başarıları arasında anlamlı bir fark olduğunu göstermiştir.

#### 4.2. İkinci Denenceye İlişkin Bulgular

Araştırmanın ikinci denencesi “**Scratch programı uygulanan grubun problem çözme beceri ölçeğinden aldıkları ön test, son test puanları arasında anlamlı bir fark vardır.**” denencesi ile ilgili bulgular incelenmiştir. Programlama öğretiminde kullanılan Scratch programının öğrencilerin, problem çözme becerisine

etkisini belirlemek için uygulama öncesinde ve sonrasında problem çözme ölçeği dağıtılmıştır. Ölçekten aldıkları puanlar arasında anlamlı bir fark olup olmadığını belirlemek için yapılan ilişkili örneklem t-testi sonuçları Tablo 7’de verilmiştir.

**Tablo 7. Problem Çözme Ölçeği Ön Test ve Son Test İlişkili Örneklem t-Testi Sonuçları**

	n	$\bar{X}$	SS	t	p
Ön-Test	22	54,5	6,54	-,840	0,410*
Son-Test	22	55,9	6,60		

\*(p<0.05)

Tablo 7’ de çalışma grubuna ait problem çözme ölçeği ön test ve son test puanlarının ortalamaları incelendiğinde, son-test puanlarının ortalaması ( $\bar{X}= 55,9$ , SS=6,60) ön-test puanlarının ortalamasından ( $\bar{X}= 54,5$ , SS=6,54) daha yüksek olduğu bulunmuştur. Ayrıca ilişkili örneklem t-test sonucuna göre problem çözme ölçeğinden alınan puanlarda, t değeri 21 serbestlik derecesinde -,840 olarak ( $t_{(22)} = -,840$ , p= 0,410 <0,05) hesaplanmıştır. Bu bulguya göre p değerine bakıldığında 0,05 manidarlık düzeyinde anlamlı bir fark olmadığı belirlenmiştir. Programlama öğretiminde kullanılan Scratch programının öğrencilerin, problem çözme becerisine etkisinin istatistiksel olarak anlamlı bir etki etmediği gözlemlenmiştir.

### 4.3. Üçüncü Denenceye İlişkin Bulgular

Araştırmanın üçüncü denencesi **“Scratch programı uygulanan grubun derse yönelik tutum ölçeğinden aldıkları ön test, son test puanları arasında anlamlı bir fark vardır.”**denencesi ile ilgili bulgular incelenmiştir. Programlama öğretiminde kullanılan Scratch programının öğrencilerin, Bilişim Teknolojileri dersine yönelik tutumlarını belirlemek için uygulamanın öncesinde ve sonrasında bilişim teknolojileri dersine yönelik tutum ölçeği dağıtılmıştır. Ölçekten aldıkları puanlar arasında anlamlı bir fark olup olmadığını belirlemek için yapılan ilişkili örneklem t-testi sonuçları Tablo 8’ de verilmiştir.

**Tablo 8. Bilişim Teknolojileri Dersine Yönelik Tutum Ölçeği Ön Test ve Son Test İlişkili Örneklem t-Testi Sonuçları**

	n	$\bar{X}$	SS	t	p
Ön-Test	22	70,0	3,28	-5,002	0,000*
Son-Test	22	89,5	18,5		

\*(p<0.05)

Tablo 8’de çalışma grubuna ait Bilişim Teknolojileri dersine yönelik tutum ölçeğinden alınan ön test, son test puanlarının ortalamaları incelendiğinde son test puanlarının ortalamasının ( $\bar{X}= 89,5$ ,  $SS=18,5$ ) ön-test puanlarının ortalamasından ( $\bar{X}= 70$ ,  $SS=3,28$ ) daha yüksek olduğu görülmektedir. Ayrıca ilişkili örneklem t-testi sonucuna göre Bilişim Teknolojileri dersine yönelik tutum ölçeğinden alınan ön test ve son test puanlarında,  $t$  değeri 21 serbestlik derecesinde  $-5,002$  ( $t_{(22)} = -5,002$ ,  $p= 0,00 < 0,05$ ) olarak hesaplanmıştır. Bu bulguya göre p değerine bakıldığında 0,05 manidarlık düzeyinde anlamlı bir fark olduğu görülmektedir. Programlama öğretiminde kullanılan Scratch programının öğrencilerin, Bilişim Teknolojileri dersine yönelik olumlu tutumlarını arttırdığı görülmüştür.

#### 4.4. Alt Probleme İlişkin Bulgular

Araştırmanın alt problemi olarak belirlenen “Scratch ile programlama öğretimine ilişkin öğrencilerin görüşleri nelerdir?” sorusuna öğrencilerin verdikleri ortak cevaplar birlikte sunulmuş, farklı görüşler ise alıntı olarak ifade edilmiştir.

#### Öğrencilerin Scratch Programına İlişkin Görüşleri

Öğrencilerin Scratch programıyla ilgili düşüncelerinin olumlu olduğu ifade edilebilir. Öğrenciler Scratch programını kullanarak güzel ve eğlenceli vakit geçirdiklerini belirtmişlerdir. Örneğin bir öğrenci; “Güzel oyunlar yapabiliyoruz” demiştir. Başka bir öğrenci ise; “Sevdim, her sınıfta görmeliyiz” demiştir. Diğer öğrenci ise; “Problem çözmemi geliştirdi, derse katılımım arttı.” şeklinde görüş belirtmiştir.

### **Öğrencilerin Problem Çözme ve Programlama Ünitesinde Scratch Programı ile Yaptıkları Çalışmalarda İlgilerini Çeken Durumlara İlişkin Görüşleri**

Öğrenciler Scratch programı ile yaptıkları çalışmaların ilgilerini çektiğini belirtmişlerdir. Bir öğrenci; “Sahne ve karakterleri istediğim şekilde değiştirebiliyorum, istediğimi yapabiliyorum.” demiştir. Bir diğer öğrenci; “İlgimi çeken, animasyon ve çizgi film yapabilmemiz.” demiştir. Bir başka öğrenci ise; “Benim istediklerimi yapması.” görüşünü belirtmiştir. Farklı olarak iki öğrenci ise ilgilerini özellikle çeken bir şeyin olmadığını belirtmişlerdir.

### **Öğrencilerin Problem Çözme ve Programlama Ünitesinde Scratch Programı ile Yaptıkları Çalışmalarda Beğenmedikleri Durumlara İlişkin Görüşleri**

Öğrenciler Scratch programı ile yapılan çalışmalarda genel olarak beğenmedikleri ya da zorlandıkları bir durum olmadığını belirtmişlerdir. Bir öğrenci; “Zorlandığım bir yer olmadı” demiştir. Bir başka öğrenci; “Beğenmediğim bir yanı yok.” demiştir. Bazı öğrenciler yaptıkları hatayı bulmakta zorlandıkları için bu durumu beğenmediklerini belirtmişlerdir. Bir öğrenci; “Bazen kafamın karıştığı oldu.” şeklinde görüş bildirmiştir. Bir başka öğrenci ise; “Yaptıklarımın yanlış çıkması, bazen yanlış bulamam beni zorladı.” şeklinde ifade etmiştir.

### **Öğrencilerin Problem Çözme ve Programlama Ünitesinin Sağladığı Yararlara İlişkin Görüşleri**

Öğrenciler genel olarak dersi daha çok sevdiklerini, yeni şeyler öğrendiklerini, bilgisayarı daha etkili bir şekilde kullanabileceklerini belirtmişlerdir. Bir öğrenci; “Bilgisayarı daha yararlı şekilde kullanabileceğimi fark ettim.” demiştir. Bir öğrenci yapılan çalışmayla ilgili; “Problem çözmeye yardımcı, oyun yapmayı öğrendim.” şeklinde görüş belirtmiştir. Bir öğrenci; “Derse yararı oldu, bilgisayarı daha etkili kullanıyoruz.” demiştir. Bir başka öğrenci; “Kendi yazılımımızı yaparak para kazanabiliriz.” demiştir. Bir diğer öğrenci ise; “Bilgimizi arttırdı, eğlenceli vakit geçirmemizi sağladı.” demiştir. Bir başka öğrenci ise; “Bilgisayarın nasıl

kullanılacağını öğrendim.” demiştir. Bir diğere öğrenci ise; “Programlamayı öğrendik.” demiştir. Başka öğrenci; “Dersten zevk almamı sağladı.” demiştir.

## BÖLÜM V

### TARTIŞMA, SONUÇ VE ÖNERİLER

Bu bölümde, araştırmadan elde edilen bulgulara dayanılarak ulaşılan sonuçlar tartışılmış ve önerilere yer verilmiştir.

#### 5.1. Tartışma

##### 5.1.1. Birinci Denenceye İlişkin Sonuçların Tartışılması

Araştırmanın birinci denencesi **“Scratch programı uygulanan grubun başarı testinden aldıkları ön test- son test puanları arasında anlamlı bir fark vardır.”** ile ilgili sonuçlara bakıldığında yapılan çalışmanın öğrencilerin akademik başarısını olumlu yönde etkilediği gözlemlenmiştir. Bu sonuçlar, alan yazındaki Özoran, Çağıltay ve Topallı’ nın (2012) çalışmasıyla benzerlik göstermiştir. Programlamaya Giriş Dersi’ nde Scratch programlama ortamı kullanılarak yapılan çalışmada, bir önceki yıla göre başarısız olan öğrenci sayısında düşüş olduğu belirlenmiştir. Korkmaz (2016), çalışmasında Scratch tabanlı oyun aktivitelerinin öğrencilerin bilgisayar programlamayı öğrenmeye yönelik tutumları, öz-yeterlik inançları ve akademik başarı düzeylerine olan etkilerini araştırmıştır. Scratch ile C++ dersleri verilen grubun akademik başarısının, geleneksel yöntemle işlenen grubun akademik başarısına göre daha yüksek olduğu belirtilmiştir. Dinçer (2018), Scratch ve Kodu Game Lab ile programlama öğretiminin 6. sınıf öğrencilerin programlama akademik başarısı, tutumu ve öz-yeterlik algıları üzerine olan etkisini incelediği çalışmasında Scratch ile programlama öğrenen öğrencilerin, Kodu Game Lab ile öğrenim görenlere göre başarıları anlamlı derecede arttığı görülmüştür. Programı uygulayarak öğrendikleri için uygulamalı eğitimin öğrenci başarısını anlamlı derecede etkilediği yapılan araştırmalarda ortaya konulmuştur. Ayrıca araştırmanın nitel bulguları incelendiğinde öğrenciler dersten zevk aldıklarını belirtmişlerdir. Öğrencinin dersten zevk alması da başarıyı etkileyen faktörlerden biridir. Öğrenciler öğrendikleriyle para kazanabileceklerini ifade etmişlerdir. Öğrenciler öğrenilenlerin yaşamda bir iş yaradığını hissedince, onu öğrenmeye daha çok istekli olabilir. Bu da başarıyı etkileyen diğer bir etken olarak düşünülebilir.

### 5.1.2.İkinci Denenceye İlişkin Sonuçların Tartışılması

Araştırmanın ikinci denencesi **“Scratch programı uygulanan grubun problem çözme beceri ölçeğinden aldıkları ön test- son test puanları arasında anlamlı bir fark vardır.”** ile ilgili bulgular incelendiğinde anlamlı bir fark olmadığı belirlenmiştir. Bu sonuçlar alan yazındaki Genç ve Tınmaz’ ın (2010) ile Kalelioğlu ve Gülbahar’ ın (2014) çalışma sonuçları ile benzerlik göstermiştir. Korkmaz(2018), çalışmasında öğrencilerin Scratch ile ilgili yapılan oyun etkinliklerinin problem çözme becerilerine anlamlı bir şekilde katkı sağlamadığını bulmuştur. Yapılan bazı çalışmalarda ise aksi yönde sonuçlarda bulunmuştur; Nam, Kim ve Lee (2010), Scratch ile programlama öğrenmesini kolaylaştırmaya yönelik, yapı-iskelesi yaklaşımına uygun bir süreç hazırladıkları çalışmalarında, öğrencilerin problem çözme becerilerine etkisini incelemişlerdir. Sonunda her iki grubun problem çözme becerilerinde önemli bir artış bulmuşlardır. Çetin de (2012)yaptığı çalışmada, programlama eğitiminin çocukların problem çözme becerilerini arttırdığını belirtmiştir. Öğrenciler uygulama düzeyinde öğrendikleri becerileri yeni durumlarda kullanamamışlardır. Bunun nedeni eğitim durumlarında uygulama düzeyindeki kazanımlara yeterli zamanın ve ağırlığın verilmemesi olabilir. Ayrıca uygulama düzeyindeki soruların yeterli sayıda çözülmemesi de bu sonucu doğurabilir.

### 5.1.3.Üçüncü Denenceye İlişkin Sonuçların Tartışılması

Araştırmanın üçüncü denencesi **“Scratch programı uygulanan grubun derse yönelik tutum ölçeğinden aldıkları ön test- son test puanları arasında anlamlı bir fark vardır.”** ile ilgili bulgular incelenmiştir. Bulgulara göre anlamlı bir fark olduğu görülmektedir. Coşar (2013) çalışmasında öğrencilerin bilgisayara yönelik tutumlarında olumlu etkiler yaptığını bulmuştur.Ouahbi ve diğerleri (2015),araştırmalarında programlama temellerini öğretmek için kullandıkları Scratch programının,öğrencilerin programlamaya yönelik motivasyonunu araştırmışlardır. Çalışmanın sonucunda, Scratch programı kullanılan gruptaki öğrencilerin diğer gruptaki öğrencilere göre başarılarında ve motivasyonlarında anlamlı bir artış olduğu bulunmuştur. Araştırmanın nitel verilerinden elde edilen sonuçlarında da öğrencilerin

bu sonucu destekleyen ifadeleri vardır. Öğrenciler dersten zevk aldıklarını, çalışmaların işe yaradığını ve para kazanabileceklerini belirterek, duyuşsal alan boyutunu desteklemişlerdir. Bu desteğin tutumu da etkilemiş olabileceği düşünülmektedir.

## 5.2. Sonuç

- Scratch ile programlama eğitimi alan öğrencilere uygulanan başarı testi ön test ve son test puanlarının arasındaki farka bakıldığında akademik başarının arttığı gözlemlenmiştir.
- Programlama öğretiminde kullanılan Scratch programının öğrencilerin, problem çözme becerisine anlamlı bir etkisinin olmadığı sonucuna ulaşılmıştır.
- Programlama öğretiminde kullanılan Scratch programının öğrencilerin, Bilişim Teknolojileri dersine yönelik tutumlarını olumlu bir şekilde arttırdığı görülmüştür.

## 5.3.Öneriler

Araştırmada elde edilen bulgulara göre şu önerilerde bulunulmuştur:

- Programlama öğretiminde farklı programlama araçları kullanılarak öğrencilerin başarı, problem çözme becerisi ve tutumları üzerinde ki etkileri incelenebilir.
- Scratch programını farklı yaş gruplarında deneyerek programlama öğretimindeki etkililiği araştırılabilir.
- Scratch programının farklı düşünme becerilerini (bilgi işlemsel düşünme) kazandırmada etkisinin olup olmadığı incelenebilir.
- Çalışma süresinin artırılması ile tekrar edilen çalışmanın, problem çözme becerisi kazandırmada istatistiksel olarak anlamlı bir fark oluşturabileceği düşünülmektedir.

## KAYNAKÇA

- Akçay, A. (2015). *Programlama becerisi öz yeterliliğinin problem çözme ve sorgulama becerileri bağlamında incelenmesi*. Yüksek lisans tezi. Necmettin Erbakan Üniversitesi Eğitim Bilimleri Enstitüsü, Konya.
- Akpınar, Y., ve Altun, A. (2014). Bilgi toplumu okullarında programlama eğitimi gereksinimi. *İlköğretim Online*, 13 (1), 3. Erişim Adresi: <http://ilkogretim-online.org.tr>
- Altun, A., ve Emir, S. (2008). Sosyal bilgiler dersinde problem çözme yönteminin erişimi, kalıcılığa ve tutuma etkisi. *Hasan Âli Yücel Eğitim Fakültesi Dergisi*, 10, 79-100. Erişim Adresi: <http://dergipark.gov.tr/iuhayefd/issue/8792/109891>
- Altun, M. (2013). *Düzenli eğitsel oyun oynayan 11-12 yaş grubu çocuklarda problem çözme becerisinin incelenmesi*. Yayınlanmamış yüksek lisans tezi. Gazi Üniversitesi Eğitim Bilimleri Enstitüsü, Ankara.
- Avcı, Ü., ve Ersoy, H. (2018). Bilgisayar programlama derslerinde öğrenme motivasyonu ölçeğinin Türkçe uyarlaması: geçerlilik ve güvenilirlik çalışması. *Journal of Higher Education & Science / Yükseköğretim ve Bilim Dergisi*, 8 (1), 73-81. doi: 10.5961/jhes.2018.249
- Başçiftçi, F. (2010). *C programlama dili*. (2. Baskı). Konya: Kampüs Atlas Kitapevi.
- Baştemur Kaya, C., ve Çakır, H (2018). Programlama dili öğreniminde Alice programının kullanım sürecinin incelenmesi. *Eğitimde Nitel Araştırmalar Dergisi- Journal of Qualitative Research in Education*, 6 (2), 187-206. doi:10.14689/issn.2148 - 2624.1.6c2s9m
- Calao, L. A., Moreno-León, J., Correa, H. E., & Robles, G. (2015). Developing mathematical thinking with Scratch. *In Design for teaching and learning in a networked world*. 17-27.
- Codeweek. <https://codeweek.eu/about/> Erişim Tarihi: 02.12.2018
- Coşar, M. (2013). *Problem temelli öğrenme ortamında bilgisayar programlama çalışmalarının akademik başarı, eleştirel düşünme eğilimi ve bilgisayara yönelik tutuma etkileri*. Yayınlanmamış Doktora tezi. Gazi Üniversitesi Eğitim Bilimleri Enstitüsü, Ankara.

- Çakıroğlu, Ü., Sarı, E., ve Akkan, Y. (2011). The view of the teacher about the contribution of teaching programming to the gifted students in the problem solving. *5th International Computer & Instructional Technologies Symposium*, 22-24 September 2011, Fırat University.
- Çamoğlu, K. (2010). *Programlama ve veri tabanı mantığı* (2. Baskı). İstanbul: Kodlab.
- Çatlak, Ş., Tekdal, M., ve Baz, F. Ç. (2015). Scratch yazılımı ile programlama öğretiminin durumu: bir doküman inceleme çalışması. *Journal of Instructional Technologies & Teacher Education*, 4(3),13-25.
- Çetin, E. (2012). *Bilgisayar programlama eğitiminin çocukların problem çözme becerileri üzerine etkisi*. Yayımlanmamış yüksek lisans tezi. Gazi Üniversitesi Eğitim Bilimleri Enstitüsü, Ankara.
- Demirer, V., ve Sak, N. (2016). Programming education and new approaches around the world and in Turkey / Dünyada ve Türkiye'de programlama eğitimi ve yeni yaklaşımlar. *Eğitimde Kuram ve Uygulama*, 12 (3), 521-546. Erişim Adresi: <http://dergipark.gov.tr/download/article-file/262355>
- Diñer, A. (2018). *6.sınıf öğrencilerine Scratch ve Kodu game lab programlama dillerinin öğretiminde öğrencilerin tutum, öz yeterlilik ve akademik başarılarının karşılaştırılması*. Yayımlanmamış yüksek lisans tezi. Dokuz Eylül Üniversitesi Eğitim Bilimleri Enstitüsü, İzmir.
- Durak, G. (2009). *Algoritma konusunda geliştirilen "programlama mantığı öğretici-p.m.ö" yazılımının öğrenci başarısına etkisi*. Yayımlanmamış yüksek lisans tezi. Balıkesir Üniversitesi Fen Bilimleri Enstitüsü, Balıkesir.
- Erkaper, Ş. (2007). *İlköğretim u kademe fen bilgisi derslerinde problem çözme becerisi ile kısa süreli bellek kapasitesi arasındaki ilişkinin belirlenmesi*. Yayımlanmamış yüksek lisans tezi, Balıkesir Üniversitesi Fen Bilimleri Enstitüsü, Balıkesir.
- Erol, O. (2015). *Scratch ile programlama öğretiminin bilişim teknolojileri öğretmen adaylarının motivasyon ve başarılarına etkisi*. Doktora tezi. Anadolu Üniversitesi Eğitim Bilimleri Enstitüsü, Eskişehir.
- Ersoy, H., Madran, R. O., ve Gülbahar, Y. (2011). Programlama dilleri öğretimine bir model önerisi: robot programlama. *Akademik Bilişim '11 Konferansı*, Malatya.

- Eryılmaz, S. (2003). *Algoritma tasarlama ve programlamaya giriş* (1. Baskı). Ankara: Detay Kitap ve Yayıncılık.
- Genç, Z., ve Tınmaz, H. (2010). Programlama dilleri dersi içerisinde öğrencilerin problem çözme becerileri ve bilgisayar tutumları değerlendirmesi: Bir Fırat Üniversitesi örneği. *Uluslararası Eğitim Teknolojileri Konferansı (IETC 2010)*, İstanbul.
- Genç, Z., ve Karakuş, S. (2011).Tasarımla öğrenme: eğitsel bilgisayar oyunları tasarımında Scratch kullanımı. *In 5th International Computer & Instructional Technologies Symposium (ICITS)*, Elazığ.
- Gezgin, D. M., Özcan, S. N., Ergün, K., Köse, Ö., ve Emir, N. (2017) Bilgisayar programlama eğitiminde Scratch programı kullanımına ilişkin lise öğrencilerinin görüşleri. *Uluslararası Bilimsel Araştırmalar Dergisi*, 182-188. doi: 10.21733/ibad.367073
- Gomes, A., & Mendes, A. J. (2007). Learning to program-difficulties and solutions. *In international conference on engineering education–ICEE (Vol. 2007)*. Erişim Adresi: <http://icee2007.dei.uc.pt/proceedings/papers/411.pdf>
- Gür, H., ve Hangül, T. (2015). Ortaokul öğrencilerinin problem çözme stratejileri üzerine bir çalışma. *Pegem Eğitim ve Öğretim Dergisi*, 5 (1), 95-112. doi:10.14527/pegegog.2015.005
- ISTE (International Society for Technology in Education) Standarts For Students, 2016 (<http://www.iste.org/standards/for-students#startstandards>). Erişim Tarihi:
- Işık, A. D. (2010).*Bilişim teknolojileri dersi için oluşturmacı yaklaşım doğrultusunda hazırlanan öğrenme paketinin etkileri*. Doktora tezi, Dokuz Eylül Üniversitesi Eğitim Bilimleri Enstitüsü, İzmir.
- İnceoğlu, M. (2010). *Tutum algı iletişim* (5. Baskı). İstanbul: Beykent Üniversitesi Yayınevi.
- Kalelioğlu, F., ve Gülbahar, Y. (2014). The effects of teaching programming via Scratch on problem solving skills: a discussion from learners' perspective. *Informatics in Education*, 13 (1), 33–50. Erişim Adresi: [https://www.mii.lt/informatics\\_in\\_education/pdf/infe232.pdf](https://www.mii.lt/informatics_in_education/pdf/infe232.pdf)
- Kardaş, N., Anagün, Ş. S., ve Yalçınoğlu, P. (2014). Problem çözme envanterini ilköğretim öğrencilerine uyarlama çalışması: doğrulayıcı faktör analizi sonuçları.

- Elektronik Sosyal Bilimler Dergisi*, 13 (51), 182-194. Erişim Adresi: <http://dergipark.gov.tr/download/article-file/70583>
- Kaş, S. (2010).*Sekizinci sınıflarda çalışma yaprakları ile öğretimin cebirsel düşünme ve problem çözme becerisine etkisi*. Yayınlanmamış yüksek lisans tezi. Marmara Üniversitesi Eğitim Bilimleri Enstitüsü, İstanbul.
- Keskinsoy, A. (2010).*Mesleki liselerde görsel programlama başarısını etkileyen faktörler*. Yayınlanmamış yüksek lisans tezi. Marmara Üniversitesi Eğitim Bilimleri Enstitüsü, İstanbul.
- Kodu Game Lab. <https://www.kodugamelab.com/about/>Erişim Tarihi: 05.12.2018
- Korkmaz, Ö. (2016). The effects of Scratch-based game activities on students' attitudes, self-efficacy and academic achievement. *International Journal of Modern Education & Computer Science*, 8 (1), 16-23. doi: 10.5815/ijmecs.2016.01.03
- Korkmaz, Ö. (2018). The effect of Scratch- and lego mindstorms Ev3-Based programming activities on academic achievement, problem-solving skills and logical-mathematical thinking skills of students. *MOJES: Malaysian Online Journal of Educational Sciences*, 4 (3), 73-88. Erişim Adresi: <https://mojes.um.edu.my/article/view/12658>
- Kuloğlu, A., ve Arı, Ü. (2014). Fen ve teknoloji öğretmen adaylarının problem çözme becerilerinin çeşitli değişkenler açısından incelenmesi. *Sosyal Bilimler Dergisi*, 4 (8),94-109.
- Maloney, J., Resnick, M., Rusk, N., Silverman, B., & Eastmond, E. (2010). The Scratch programming language and environment. *ACM Transactions on Computing Education (TOCE)*, 10 (4), 16. <http://doi.acm.org/10.1145/1868358.1868363>.
- Miles, M. B., & Huberman, A. M. (1984). *Qualitative Data Analysis: An Expanded Sourcebook*. Sage.
- Milli Eğitim Bakanlığı. (2012).*Ortaokul ve imam hatip ortaokulu bilişim teknolojileri ve yazılım dersi öğretim programı*. Talim Terbiye Kurulu Başkanlığı, Ankara.
- Milli Eğitim Bakanlığı. (2018). *Bilişim teknolojileri ve yazılım dersi öğretim programı*. Talim Terbiye Kurulu Başkanlığı, Ankara.

- Nam, D., Kim, Y., & Lee, T. (2010). The effects of scaffolding-based courseware for the Scratch programming learning on student problem solving skill. *Proceedings of the 18th International Conference on Computers in Education*. Putrajaya, Malaysia. 723-727.
- Nikou, S. A., & Economides, A. A. (2014). Transition in student motivation during a Scratch and an App inventor course. In 2014 *IEEE Global Engineering Education Conference (EDUCON)*, 1042-1045. DOI > 10.1109/EDUCON.2014.6826234
- Ouahbi, I., Kaddari, F., Darhmaoui, H., Elachqar, A., & Lahmine, S. (2015). Learning basic programming concepts by creating games with Scratch programming environment. *Procedia-Social and Behavioral Sciences*, 191, 1479-1482. doi: 10.1016/j.sbspro.2015.04.224
- Ozoran, D., Cagiltay, N., & Topalli, D. (2012). Using Scratch in introduction to programming course for engineering students. In *2nd International Engineering Education Conference*, 125-132.
- Özgürol, M. B. (2010). *Mesleki ve teknik lise öğrencilerinin bilgisayarına yönelik tutumlarının çeşitli değişkenlere göre incelenmesi*. Yayımlanmamış yüksek lisans tezi. Ankara Üniversitesi Eğitim Bilimleri Enstitüsü, Ankara.
- Özsoy, G. (2005). Problem çözme becerisi ile matematik başarısı arasındaki ilişki. *Gazi Üniversitesi Gazi Eğitim Fakültesi Dergisi*, 25(3), 179-190.
- Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., & Kafai, Y. (2009). Scratch: programming for all. *Communications of the ACM*, 52 (11), 60-67. doi:10.1145/1592761.1592779
- Saracaloğlu, A. S., Serin, O., ve Bozkurt, N. (2001). Dokuz eylül üniversitesi eğitim bilimleri enstitüsü öğrencilerinin problem çözme becerileri ile başarıları arasındaki ilişki. *M. Ü. Atatürk Eğitim Fakültesi Eğitim Bilimleri Dergisi*, 14, 121-134.
- Saracaloğlu, A. S., ve Kanmaz, A. (2012). Eğitim fakültesi birinci sınıf öğrencilerinin problem çözme becerilerine genel bir bakış. *e-Journal of New World Sciences Academy*, 7 (2), 683-699.
- Sayın, Z. (2017). Bilgisayar bilimi eğitimi kapsamı. Y. Gülbahar (Ed.), *Bilgi işlemsel düşünmeden programlamaya içinde* (s. 13-141). Ankara: Pegem Akademi.
- Scratch. <https://scratch.mit.edu/about/> Erişim Tarihi: 26.12.2018

- Senemođlu, N. (2013). *Gelişim, öğrenme ve öğretim* (23. Baskı). Ankara: Yargı Yayınevi.
- Sezgin, E. (2011). *Problem çözme becerisi ölçeğinin geliştirilmesi*. Yayımlanmamış yüksek lisans tezi. Ankara Üniversitesi Eğitim Bilimleri Enstitüsü, Ankara.
- Small Basic. <http://smallbasic.com/> Erişim Tarihi: 05.12.2018
- Solmaz, E. (2014). *Programlama dili öğretiminde Alice yazılımının ders başarısı, eleştirel düşünme ve problem çözme becerileri ile üst bilişsel farkındalık düzeyine etkisi*. Yayımlanmamış doktora tezi. Gazi Üniversitesi Eğitim Bilimleri Enstitüsü, Ankara.
- Sönmez, V. (2012). *Öğretmen el kitabı* (17. Baskı). Ankara: Anı Yayıncılık.
- Sönmez, V., ve Alacapınar, F. G. (2014). *Örneklendirilmiş bilimsel araştırma yöntemleri* (3. Baskı) Ankara: Anı Yayıncılık.
- Şahin, Ç. (2004). Problem çözme becerisinin temel felsefesi. *Kazım Karabekir Eğitim Fakültesi Dergisi*, 10.
- Şimşek, E. (2018). *Programlama öğretiminde robotik ve Scratch uygulamalarının öğrencilerin bilgi işlemsel düşünme becerileri ve akademik başarısına etkisi*. Yayımlanmamış yüksek lisans tezi. On dokuz Mayıs Üniversitesi Eğitim Bilimleri Enstitüsü, Samsun.
- Şimşek, İ. (2018). Dünyada programlama öğretimi. Y. Gülbahar ve H. Karal (Ed.), *Kuramdan uygulamaya programlama öğretimi içinde* (s. 38-65). Ankara: Pegem Akademi.
- Tavşancıl, E. (2005). *Tutumların ölçülmesi ve spss ile veri analizi*. (2. Baskı). Ankara: Nobel Kitap.
- Tavşancıl, E. (2014). *Tutumların ölçülmesi ve spss ile veri analizi*. (5. Baskı). Ankara: Nobel Kitap.
- TDK. <http://www.tdk.gov.tr> Erişim Tarihi: 22.11.2018.
- Tekerek, M., ve Altan, T. (2014). The effect of Scratch environment on student's achievement in teaching algorithm. *World Journal on Educational Technology*, 6 (2), 132-138.
- Utting, I., Cooper, S., Kölling, M., Maloney, J., & Resnick, M. (2010). Alice, Greenfoot, and Scratch--a discussion. *ACM Transactions on Computing Education*, 10 (4), 17. <https://doi.org/10.1145/1868358.1868364>

- Ünsal, Y., ve Ergin, İ. (2011). Fen eğitiminde problem çözme sürecinde kullanılan problem çözme stratejileri ve örnek bir uygulama. *Savunma Bilimleri Dergisi*, 10(1), 72-91.
- Yıldırım, E. (2017). *Scratch programlama dili eğitime yönelik bir mobil uygulamanın geliştirilmesi*. Yüksek lisans tezi. Çanakkale On Sekiz Mart Üniversitesi Eğitim Bilimleri Enstitüsü, Çanakkale.
- Yıldız, S. A., ve Ekşisu, M. (2011). Problem çözme becerisini geliştirme programının 9. Sınıf öğrencilerinin problem çözme becerisi üzerindeki etkisi. *Erzincan Üniversitesi Eğitim Fakültesi Dergisi*, 13(1), 189-206.
- Yükseltürk, E., ve Altıok, S. (2015). Bilişim teknolojileri öğretmen adaylarının bilgisayar programlama öğretimine yönelik görüşleri. *Amasya Üniversitesi Eğitim Fakültesi Dergisi*, 4 (1), 50-65.
- Yükseltürk, E., ve Altıok, S. (2016). Bilişim teknolojileri öğretmen adaylarının programlama öğretiminde Scratch aracının kullanımına ilişkin algıları. *Mersin Üniversitesi Eğitim Fakültesi Dergisi*, 12 (1), 39-52.
- Yünkül, E., Durak, G., Çankaya, S., ve Abidin, Z. (2017). The effects of Scratch software on students' computational thinking skills. *Necatibey Eğitim Fakültesi Elektronik Fen ve Matematik Eğitimi Dergisi (EFMED)*, 11 (2), 502-517. doi: 10.17522/balikesirnef.373424

## EKLER

### Ek 1. Başarı Testi

1- Tabloda dokuz tane kavramın tanımları verilmiştir. Kavramların uygun tanımlarını, kavramların önünde bulunan noktalı yere yazınız

..... Algoritma	<b>1-</b> Bir problemi çözebilmek için uygulanması gereken adımların açık, düzenli ve sıralı bir şekilde ifade edilmesidir.
..... Sahne	<b>2-</b> Tasarladığımız karakterin hareketlerini gördüğümüz bölümdür.
.....Karakter	<b>3-</b> Scratch programında kullanılan bir nesnedir. Bu objeye farklı kostümlerde giydirek görünümünü değiştirebiliriz.
.....Programlama Dili	<b>4-</b> Bir Problemin algoritmik çözümünün bilgisayara aktarmak için kullanılan, özel olarak geliştirilmiş komut yapılarıdır.
..... Scratch	<b>5-</b> resim, ses, müzik gibi çeşitli medya araçlarını bir araya getirebileceğimiz, kendi animasyonlarımızı, bilgisayar oyunlarımızı tasarlayabileceğimiz ya da etkileşimli hikayeler yazabileceğimiz bir grafik programlama dilidir.
.....Doğrusal Mantık Yapısı	<b>7-</b> İşlemleri sırasıyla çalıştırdığını ifade eder.
.....Karar Yapısı	<b>8-</b> İşlemleri iki ya da daha fazla seçenek arasından seçim yapmak hakkı tanıyan ve ona göre devam ettirmek için kullanılır.
.....Döngü Yapısı	<b>9-</b> Yapılan bazı işlemleri tekrarlamak için kullanılır.

2 - A kutusundaki numaralarla verilmiş kodlar bulunmaktadır. Bu kodları kutucukta verilen kod bloklarıyla eşleştiriniz.



..... - Algılama
..... - Görünüm
..... - Kontrol
..... - Hareket
..... - Ses
..... - Kalem
..... - Operatörler

3- Serpil eve gelen misafirleri için çay demlemek istemektedir. Serpil'in çay demlemesi için gerekli adımlar numaralı olarak verilmiştir. Ama bu adımlardan bazıları yanlış olarak verilmiştir?

1	Başla
2	Demliğe yeteri kadar çayı ekle
3	Kaynayınca kadar bekle
4	Kaynamış suyu demliğe dök
5	Bitir
6	Çaydanlığın altına suyu koy
7	Çay demlenince bardaklara doldur
8	Çaydanlığın altına tekrar su dök

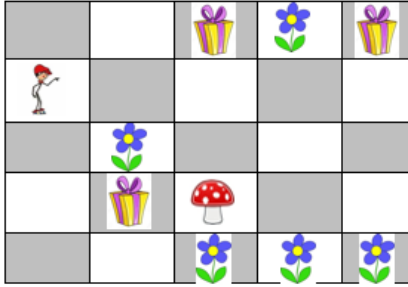
A) 1-2-7-8

B) 3-5-7-8

C) 2-5-6-8

D) 4-5-6-7

4- Cin Ali'nin mantara gidebilmesi için gerekli adımlar seçeneklerde verilmiştir. Hangisi algoritma kurallarına uygu şekilde yazılmıştır?



A) Başla- İlerle- İlerle- Sağa Dön- İlerle- İlerle- Sağa Dön- İlerle- İlerle- Bitir

B) Başla- İlerle- İlerle – İlerle- İlerle- Sağa Dön- İlerle- İlerle- sağa dön- İlerle- İlerle- Bitir

C) Başla- İlerle- İlerle- Sağa Dön- İlerle- İlerle- Bitir

D) Başla- İlerle- İlerle – İlerle- İlerle- Sola Dön- İlerle- Bitir

5 – Cin Ali'nin elma ağacına ulaşması için seçeneklerde takip etmesi gereken yollar verilmiştir. Seçeneklerden hangisi doğrusal mantık yapısına uygundur?



A)

İlerle  
İlerle  
İlerle  
Sağa dön  
İlerle  
Sola dön  
İlerle

B)

İlerle  
İlerle  
İlerle  
Sağa dön  
İlerle  
İlerle

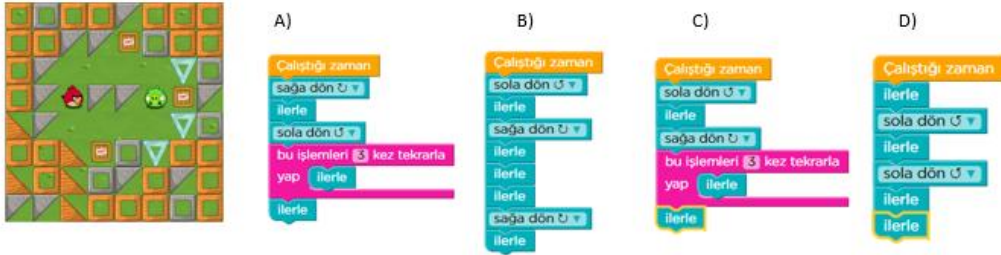
C)

İlerle  
İlerle  
Sağa dön  
İlerle  
Sola dön  
İlerle

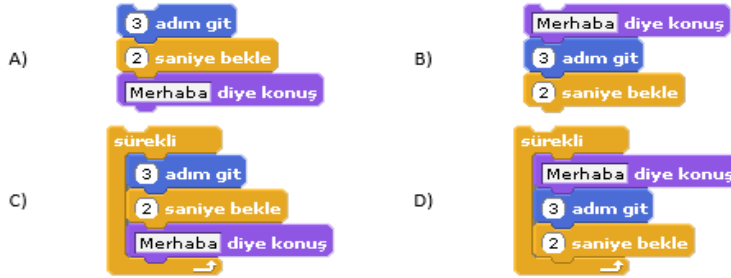
D)

İlerle  
İlerle  
İlerle  
Sağa doğru yol varsa  
Sağa dön  
İlerle

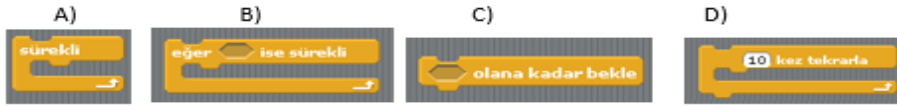
6- Sınırlı kuşun yeşil hayvanı yakalaması için seçeneklerde verilen yollardan hangisi doğrusal mantık yapısına uygundur?



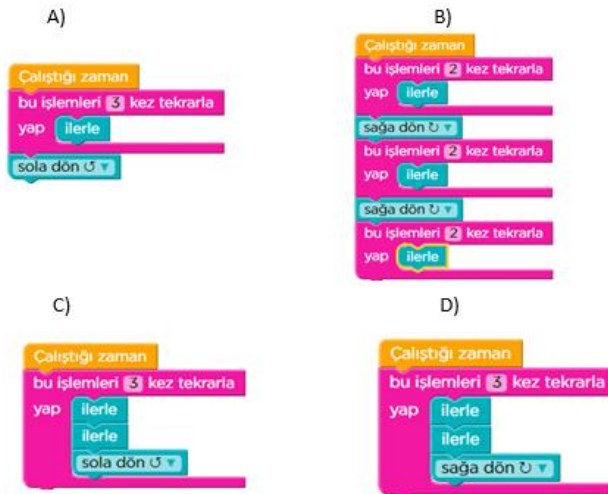
7- Önce 3 adım git. Sonra 2 saniye bekle. Daha sonra da “Merhaba” söyle ve bu yapıların hepsi sürekli olsun. Yukarıdaki açıklamaya uygun kod bloğu aşağıdakilerden hangisidir?



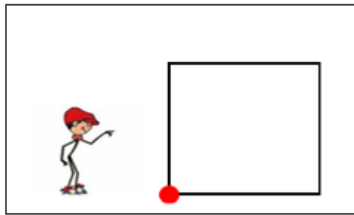
8- Verilen bir işlemin belirli sayıda tekrarlanmasını istiyorsak hangisini kullanmalıyız?



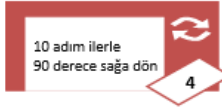
9- Zombi' nin ayçiçeğine gitmesi için takip etmesi gereken yollar aşağıda verilmiştir. Seçeneklerden hangisi döngü yapısına uygundur?

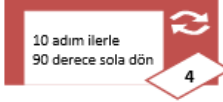


10- Cin Ali'ye kare çizmesi için seçeneklerde yollar verilmiştir. Aşağıdaki seçeneklerden hangisi döngü yapısına uygundur? (Karenin her kenarı 10 adım uzunluğundadır)

A) 

A) 10 adım ilerle  
90 derece sola dön  
10 adım ilerle  
90 derece sola dön  
10 adım ilerle  
90 derece sola dön  
10 adım ilerle  
90 derece sola dön

B) 

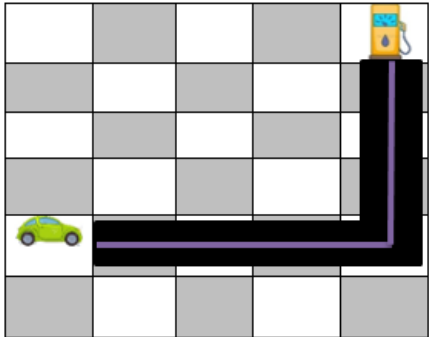
C) 


D) 10 adım ilerle  
90 derece sağa dön  
10 adım ilerle  
90 derece sağa dön  
10 adım ilerle  
90 derece sağa dön  
10 adım ilerle  
90 derece sağa dön

11- Hazırlayacağımız bir oyunda, sahnedeki karakter bir şarta bağlı olarak hareket ediyorsa aşağıdaki kodların hangisi kesinlikle kullanılmıştır?

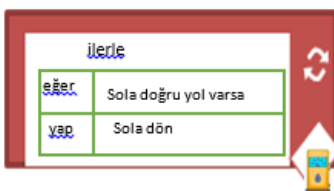
A)  B)  C)  D) 

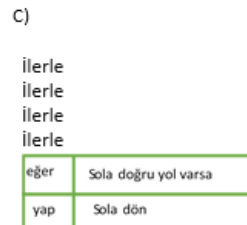
12- Zeynep arabasına benzin alması için yolun sonundaki benzin istasyonuna gitmesi için gitmesi gereken yollar verilmiştir. Seçeneklerden hangisi karar yapısına uygundur?

A) 


A) 

Sola dön  
ilerle  
ilerle  
ilerle

B) 

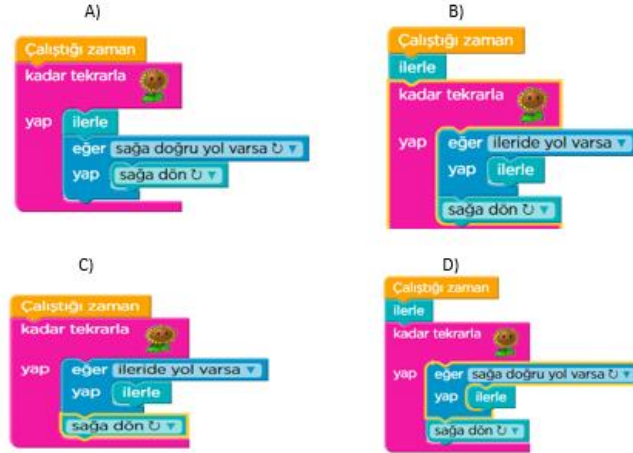
C) 

ilerle  
ilerle  
ilerle  
ilerle

D) 

ilerle  
ilerle  
ilerle

13- Zombi'nin ayçiçeğine gitmesi için takip etmesi gereken yollar aşağıda verilmiştir. Seçeneklerden hangisi karar yapısına uygundur?



14- Bir problemin çözümünde doğrusal yapısı kullanılması için problemde hangi ilke aranmalıdır? Açıklayarak yazınız.

15- Bir problemin çözümünde döngü yapısı kullanılması için problemde hangi ilke aranmalıdır? Açıklayarak yazınız.

16- Bir problemin çözümünde karar yapısı kullanılması için problemde hangi ilke aranmalıdır? Açıklayarak yazınız.

17- Resimde kod blokları verilmiştir. Bu kod bloklarına ait algoritmayı yazınız.

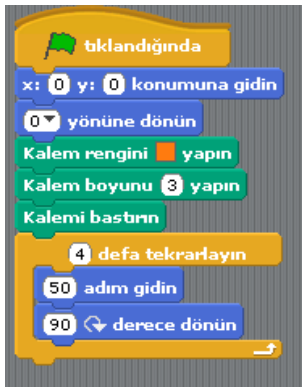


18- Aşağıda verilen problem durumu verilmiştir. Problemin çözümü için gerekli algoritmayı yazınız.



Mustafa amca tilkisini, kazını ve bir mısır çuvalını nehrin karşısına geçirmesi gerekmektedir. Mustafa amca karşıya geçebileceği tek araç ufacık bir kayıkmiş ve hepsinin beraber karşıya geçmesi mümkün değildir. Kayığa her defasına birini alabiliyormuş; ya tilkiyi, ya mısır çuvalını ya da kazı yanına alabilecekmış. Ancak bir sorunu daha varmış, tilkiyle kazı yalnız bırakırsa tilki kazı yermiş, kazla da mısır çuvalını yalnız bıraksa bu sefer kaz mısırları yermiş. Mustafa amca hepsini nasıl karşıya geçirebilir?

19- Resimde kod blokları verilmiştir. Bu kod bloklarına ait algoritmayı yazınız.



20-Zombi'nin ayçiçeğine gitmesi için kullanman gereken bloklar aşağıda verilmiştir. Kutucuğa uygun blokları kullanarak çözümü yazınız



21- Sınırlı kuşun yeşil hayvanı yakalaması için kullanman gereken bloklar aşağıda verilmiştir. Kutucuğa uygun blokları kullanarak çözümü yazınız.



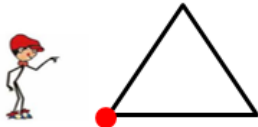
22- Sınırlı kuşun yeşil hayvanı yakalaması için kullanman gereken bloklar aşağıda verilmiştir. Kutucuğa uygun blokları kullanarak çözümü yazınız.



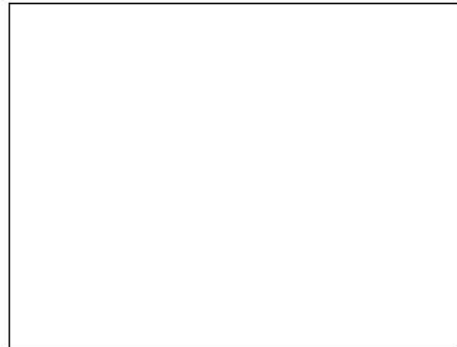
23- “Seçtiğimiz karakterin Klavyede ki boşluk tuşuna basıldığında kostüm değıştirsın ve bu işlem sürekli devam etsin.” Aşğıda karışık olarak verilmiş olan komutları kullanarak, verilen durumu gerçekleştiren komut bloğunu boş alana çizerek oluşturun.



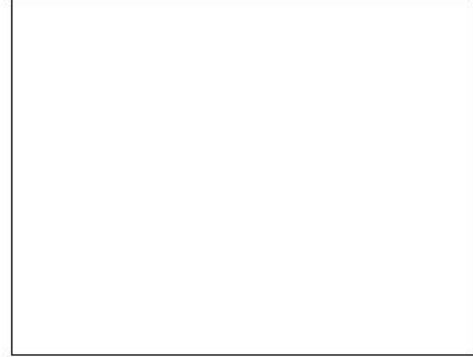
24- Cin Ali'nin kırmızı noktalı yerden başlayarak üçgen çizmesi için gerekli kodları yazınız.



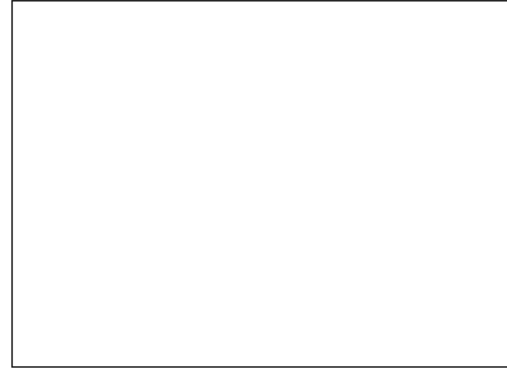
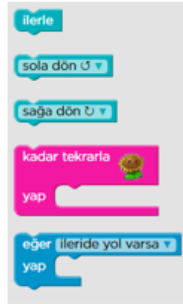
25-Zombi'nin ayçiçeğine gitmesi için kullanman gereken bloklar aşağıda verilmiştir. Kutucuğa uygun blokları kullanarak çözümü yazınız.



26-Zombi'nin ayçiçeğine gitmesi için kullanman gereken bloklar aşağıda verilmiştir. Kutucuğa uygun blokları kullanarak çözümü yazınız.



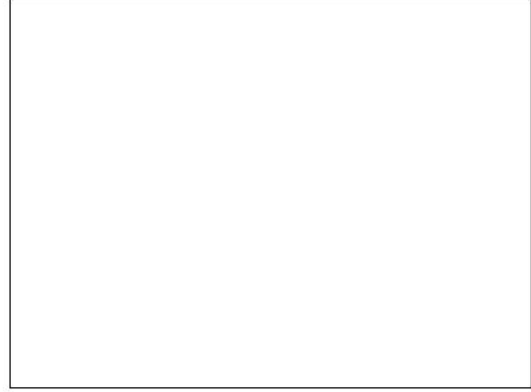
27-Zombi'nin ayçiçeğine gitmesi için kullanman gereken bloklar aşağıda verilmiştir. Kutucuğa uygun blokları kullanarak çözümü yazınız.



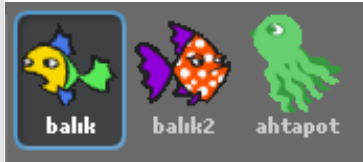
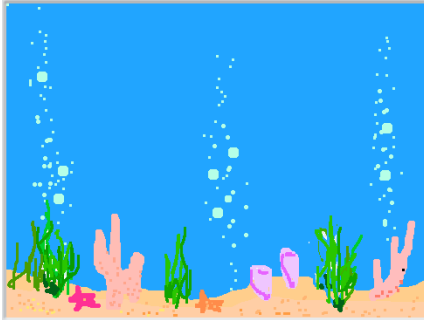
28- Arı'nın çiçeklere giderek nektar toplaması için kullanman gereken bloklar aşağıda verilmiştir. Kutucuğa uygun blokları kullanarak çözümü yazınız.



29-Zombi'nin ayçiçeğine gitmesi için kullanman gereken bloklar aşağıda verilmiştir. Kutucuğa uygun blokları kullanarak çözümü yazınız.



30- Akvaryumda sürekli sağa ve sola yüzen 2 tane balık ve 1 tane ahtapot olsun. Balıklardan birisi diğerinden daha hızlı yüzsün. Ahtapotun yüzerken kostüm değiştirmesini sağlayın. Aşağıdaki kutucuğa gerekli kodları yazınız.



## Ek 2. Problem Çözme Ölçeği

	Tamamen Katılıyorum (4)	Çoğunlukla Katılıyorum (3)	Ara Sıra Katılıyorum (2)	Hiç Katılmıyorum (1)
1. Bir sorunu çözemediğimde, neden çözemediğimi düşünmem.				
2. Bir sorunu ilk denememde çözemezsem, bu sorunu bir daha çözemeyeceğimi düşünürüm.				
3. Bir sorunu çözebilmek için belirli bir yol izledikten sonra beklediğim sonuçla ortaya çıkan sonucu karşılaştırırım.				
4. Bir sorunum olduğunda bildiğim bütün çözüm yollarını düşünürüm.				
5. Başlangıçta çözülemez gibi görünse de, birçok sorunu çözebilirim.				
6. Bir problemi çözerken kararlar alırım ve sonunda bunlardan mutlu olurum.				
7. Küçük ya da büyük olsun sorunlarımı çözmek için zaman ayırmam, her şeyi oluruna bırakırım.				
8. Bir sorunla karşılaştığımda ne yapacağıma karar vermeden önce, sorun üzerinde düşünürüm.				
9. Bir karar verirken değişik seçenekleri karşılaştırırım ve her bir seçeneğin bir diğerine göre sonuçları üzerinde düşünürüm.				
10. Bir sorunu çözmek için plan yaptığımda bu planın işe yarayacağından oldukça emin olurum.				
11. Benim ya da bir başka kişinin yaptığı bir davranışın sonucunu tahmin etmeye çalışırım.				
12. Yeterli çaba gösterdiğimde ve zamanım olduğunda, karşılaştığım bütün sorunları çözebileceğime inanırım.				
13. Her zaman karşılaştığım problemlerden farklı olan problemleri de çözebileceğimden eminim.				
14. Bir sorunu çözmeye çalışırken boşa emek harcadığımı, gerçek konuya bir türlü ulaşamadığımı hissederim.				
15. Bir sorunla karşılaştığımda, ani kararlar veririm ve sonra yaptığımdan pişman olurum.				
16. Yeni ve zor sorunları çözme yeteneğime güvenirim.				

17.Farklı seçenekleri karşılaştırmak ve bu seçeneklerden birine karar vermek için düzenli bir şekilde çalışırım.				
18.Bazen o kadar duygusal olurum ki, sorunu çözmeme yarayacak seçenekleri fark edemem.				
19.Bir karar verdikten sonra, beklediğim sonuçla gerçekleşen sonuç genellikle aynıdır.				
20.Bir sorun olduğunu fark ettiğimde, yaptığım ilk şeylerden birisi, sorunun ne olduğunu tam olarak anlamaya çalışmaktır.				

Ölçekteki süresiz değişkenler araştırmacı tarafından izin alınarak sürekli değişkenlere dönüştürülüp aritmetik ortalama, standart sapma kullanılarak t istatistiği kullanılmıştır.

### Ek 3. Bilişim Teknolojileri Dersine Yönelik Tutum Ölçeği

Sevgili Öğrencim;

Bu tutum ölçeği bilgisayar dersine yönelik tutumunuzu ölçmek amacıyla hazırlanmıştır. Yapmanız gereken her ifadeyi okuduktan sonra size uygun olan seçeneği işaretlemektir. Soruların herhangi bir doğru cevabı bulunmamaktadır. Sadece belirtilen ifade ile ilgili sizin düşünceniz belirlenmeye çalışılacaktır. Bu araştırma sonuçları gizli kalacak, sonuçlar kimseye gösterilmeyecektir. Bu bakımdan cevaplarınızı samimi olarak verebilirsiniz. Lütfen her ifade için düşüncenizi belirtiniz.

	Katılmıyorum (1)	Kararsızım (2)	Katılıyorum (3)
1. Bilgisayar dersini severim.			
2. Bilgisayar dersinin konularıyla ilgili daha çok şey öğrenmek isterim.			
3. Bilgisayar dersine girerken huzursuz olurum.			
4. Bilgisayar dersine çalışırken canım sıkılır.			
5. Çalışma zamanımın önemli bir kısmını bilgisayar dersine ayırırım.			
6. Bilgisayar dersi olmasa öğrenciliğin daha zevkli olacağına inanırım.			
7. Bilgisayar dersinde zaman geçmek bilmez.			
8. Bilgisayar dersi ilgimi çeker.			
9. Bilgisayar dersinde uykum gelir.			
10. Bilgisayar dersindeki uygulamaları yaparken canım sıkılır.			
11. Okul dışında bilgisayar dersine çalışmaktan nefret ederim.			
12. Bilgisayar dersinde mutlu olurum.			
13. Bilgisayar dersindeki etkinliklere gönüllü olarak katılırım.			
14. Bilgisayar dersine kendi isteğimle çalışırım.			
15. Bilgisayar dersinin ileriki yıllarda bir daha karsıma çıkmasını istemem			
16. Bilgisayar dersine karşı bir ilgim yoktur.			

17. Bilgisayar dersi denilince aklıma sıkıntı verici şeyler gelir.			
18. Bilgisayar dersini düşününce strese girerim.			
19. Bilgisayar dersi başarılı olduğum bir derstir.			
20. Bilgisayar dersinde başarılı olmak benim için önemlidir.			
21. Zorunlu olmasa bilgisayar derslerine girmezdim.			
22. Bilgisayar dersinden yüksek notlar alabilirim.			
23. Bilgisayar derslerinde sadece oyun oynayarak vakit geçirmek isterim.			
24. Bilgisayar dersini ileriki yıllarda da almak isterim.			
25. Bilgisayar öğretmenlerini sevmem			
26. Bilgisayar derslerini iple çekerim.			
27. Her okulda bilgisayar dersinin olması gerektiğini düşünürüm.			
28. Birinci sınıftan itibaren her yıl bilgisayar dersinin olması gerektiğini düşünürüm.			
29. Okula başladığım günden bu yana bilgisayar dersine girmek isterim.			
30. Bilgisayar derslerinde endişeli olurum.			
31. Bilgisayar dersinde öğrendiğim konular ilgimi çeker.			
32. Bilgisayar dersinde islenen konuları öğrenmek benim için zordur.			
33. Bilgisayar dersinde etkinlik yapmak sıkıcıdır.			
34. Bilgisayar dersinde yaptığım yeni bir etkinlik ilgimi çeker.			
35. Bilgisayar dersinde yaptığım etkinliklerin yanlış kısımlarını düzeltmekten sıkılırım.			
36. Bilgisayar derslerinin olduğu günler okula gitmek içimden gelmez.			

Ölçekteki süresiz değişkenler araştırmacı tarafından izin alınarak sürekli değişkenlere dönüştürülüp aritmetik ortalama, standart sapma kullanılarak t istatistiği kullanılmıştır.

#### **Ek 4. Yarı Yapılandırılmış Görüşme Formu**

#### **Programlama Ünitesinde Kullanılan Scratch Programı Hakkında Alınan Öğrenci Görüşleri**

1. Scratch programı hakkında düşüncelerin nelerdir?
2. Problem Çözme Ve Programlama ünitesinde Scratch programı ile yaptığın çalışmalarda ilgini çeken durumlar neler oldu?
3. Problem çözme ve programlama ünitesinde Scratch programı ile yaptığın çalışmalarda beğenmediğin durumlar nelerdir?
4. Problem çözme ve programlama ünitesinin sana sağladığı yararlar nelerdir?

## Ek 5. Araştırma İzni



T.C.  
KONYA VALİLİĞİ  
İl Millî Eğitim Müdürlüğü

Sayı : 83688308-605.99-E.4666877

05.03.2018

Konu: Araştırma İzni (Rukiye Berna BALA)

NECMETTİN ERBAKAN ÜNİVERSİTESİ REKTÖRLÜĞÜNE  
(Öğrenci İşleri Daire Başkanlığı)

İlgi : 26/02/2018 tarihli ve 48178250-300-E.3031 sayılı yazınız.

Üniversiteniz Eğitim Bilimleri Enstitüsü Eğitim Bilimleri Anabilim Dalı Programı ve Öğretimi Bilim Dalı Tezli Yüksek Lisans Programı Öğrencisi Rukiye Berna BALA'nın "6. Sınıf Öğrencilerine Programlama Dili Öğretilirken Kullanılan Scratch Programının Öğrencilerin Problem Çözme Becerilerine ve Tutumlarına Etkisi" konulu araştırmasını uygulama talebi incelenmiştir.

Araştırmanın, Seydişehir Ketenli Ortaokulunda öğrenim gören 6. sınıf öğrencilerine eğitim öğretimi aksatmamak kaydıyla uygulanmasında sakınca görülmemektedir. Araştırmacı, Müdürlüğümüze bağlı eğitim kurumlarındaki çalışmalarını 2017-2018 eğitim öğretim yılı içerisinde tamamlamak zorundadır. Araştırma kapsamında yürütülecek çalışmalar 2017-2018 eğitim öğretim yılında tamamlanmaması durumunda Müdürlüğümüzden tekrar izin alınması gerekmektedir.

Araştırmada Müdürlüğümüz tarafından onaylanarak gönderilen veri toplama araçları kullanılacak olup, araştırma sonucunun CD ortamında iki nüsha olarak Müdürlüğümüze gönderilmesi gerekmektedir.

Bilgilerinizi ve adı geçene tebliğini arz ederim.

Mukadder GÖRSOY  
İl Millî Eğitim Müdürü

Ek:

- 1-Erişi Testi (12 Sayfa)
- 2-Bilgisayar Dersine Yönelik Tutum Ölçeği (1 Sayfa)
- 3-Problem Çözme Ölçeği(1 Sayfa)

Görevli Elektronik İmza  
Aşağı İmza  
05.03.2018

## Ek 6. Örnek Ders Planı

### GÜNLÜK PLAN

#### BİÇİMSEL BÖLÜM

**Ders:** Bilişim Teknolojileri ve Yazılım

**Sınıf:**6. Sınıflar

**Ünite Adı:**Problem Çözme ve Programlama

**Hedef:**Scratch programı ile ilgili temel kavramların anlam bilgisi

**Davranış:** “Scratch programı, sahne, karakter, kod blokları” kavramlarının tanımlarını; ne anlama geldiğini yazma, söyleme, seçip işaretleme, eşleştirme

**Süre:** 40’ + 40’

#### GİRİŞ BÖLÜMÜ

**1.Dikkati Çekme:** Öğretmenin “Günlük hayatta hangi yazılımları kullanıyoruz?” ve “Kullandığımız programlar, yazılımlar nasıl yapılır, geliştirilmesi için neye ihtiyaç vardır?” sorularının sorması ve öğrencilerden yanıt alması. “Öğrencilere yanıtlarınızı unutmayın, dersin sonunda tekrar tartışacağız.” demesi.

**2.Güdüleme:** Bu derste öğreneceğiniz kavramlarla Scratch programını tanıyıp; kendi oyunlarınızı kendiniz geliştirebileceksiniz.

**3.Gözden Geçirme:** Kendinize ait ufak uygulamalar ve oyunlar geliştirebileceksiniz.

**4.Derse Geçiş:** Öğretmenin “Uygulama geliştirmeye başlamadan önce açıklamam gereken temel kavramları öğrenmeniz gerek.” demesi ve ilgili açıklamaların yapılması.

#### GELİŞTİRME BÖLÜMÜ

##### Etkinlikler

- Öğrencilere ”oyun oynamayı seviyoruz peki hiç bu oyunları yapmayı düşündünüz mü? ” sorusu yöneltilir. “12 yaşında bir çocuk sizce oyun yapabilir mi? siz böyle birini tanıyor musunuz?” soruları tartışıldıktan sonra 12 yaşında uygulama geliştiren çocuğun videosu izletilir ve üzerinde konuşulur.

2. Sınıfa “Bill Gates, Mark Zuckerberg isimlerini duyan var mı? Bilgisayarlar ilk ne zaman tanışmışlar? İlk yazdıkları program neydi?” soruları sorularak üzerinde tartışılır. Bilgisayar bilimleri alanında çalışan önemli isimlerin konuştuğu bir video daha izletilir. Burada amaç, kod yazmanın, bir uygulama geliştirmenin sanıldığı kadar zor olmadığını, dahi olmak gerekmediğinin, ufak adımlar atarak bu işe başlanacağını çocuklara sezdirilmesidir.
3. Öğretmenin “kendi oyunlarımızı yapmadan önce kullanacağımız yazılımı tanımamız gerek, Scratch programını daha önce duyan var mı?” der ve programla yapılan çeşitli uygulamalar gösterilir.
4. Daha sonra “Scratch” kelime anlamından bahsedilir, program ile neler yapılacağından bahsedilir. Programı kullanarak yapılan çeşitli örnekler gösterilir.
5. Programın arayüzü anlatılır. Scratch arayüzü 3 ana bölmeden oluşur. Blok Paleti(Kod Blokları): Karakterimizi programlamak için kullanabileceğimiz blokların bulunduğu kısımdır. Burada her bir bloğun ne işe yaradığı kısaca anlatılır. Kodlama Alanı: Blokları sürükleyerek komut dizileri oluşturacağımız alandır. Sahne: Bizim sahnemizdir. Projemizi çalıştırdığımızda tasarladığımız her şey burada hayat bulur. Yazılımın diğer simgelerinden, ne işe yaradıklarından bahsedilir. Karakter kavramından bahsedilir, sahneye karakter ekleme gösterilir.
6. Şimdi eklediğimiz bu karakteri hareket ettirelim ve ilk kodumuzu yazmaya başlayalım denilerek, önceden hazırlanan bir uygulama gösterilir. Sizlerden de dersin sonunda bir karakteri hareket ettirmenizi isteyeceğim denilerek, öğrencilerin hedeften haberdar olunması sağlanır.
7. Hareket bloğunda olan yön tuşlarından bahsedilir, her bir yön için tek tek kod yazılır. Öğrencilerden sahneye başka bir karakter eklemeleri istenir ve hareket ettirilmesi sağlanır.
8. Öğretmenin, “Anlatamadığım veya tekrar anlatılmasını istediğiniz bir yer var mı?” diye sorması. Varsa tekrar açıklayıp yeni bir örnek vermesi.

## SORULAR

**Ara Özet:** Scratch programının genel yapısı gösterilir. Yeni karakter ekleme ve yön tuşları ile ilgili yapılan etkinlik tekrar edilir.

**Ara Geçiş:** Öğrencilere“ yeni karakter ekleme ve yön tuşları ile ilgili size çalışma kâğıdı dağıtacağım. Öğrendiklerinizi uygulamanızı istiyorum.” denmesi. Öğretmen Çalışma Kâğıdı 1’i öğrencilere dağıtması, yaptırması ve toplaması.

## KAPANIŞ

**Son Özet:** Artık istediğiniz bir uygulama ya da oyunu geliştirmeniz için gereken komutları öğrendiniz. Sizlerde istediğiniz bir uygulama yazabilirsiniz.

**Tekrar Güdüleme:** Bu derste öğreneceğiniz kavramlarla Scratch programını tanıyıp; kendi oyunlarınızı kendiniz geliştirebileceksiniz.

**Kapanış:** Dikkati çekmedeki problem tekrar sorulur ve üzerinde tartışılır.

## DEĞERLENDİRME

Aşağıdaki soruların her birini öğrenciye sorarak doğru yanıt veren öğrenciye pekiştireç vermesi. Yanlış ya da eksik yanıt veren öğrenciye doğru yanıtı buldurulması ve tekrar ettirilmesi.

- ✓ “Scratch programı, sahne, karakter, kod blokları” kavramlarının ne olduğunun sorulması

## Ek 7. Çalışma Kâğıdı 1

### YÖN TUŞLARI ETKİNLİĞİ

Sahneye eklediğimiz dalgıçın klavyeden yön tuşlarını kullanarak (yukarı, sağa, sola, aşağı) hareket etmesini istiyoruz.



Aşağıdaki resimde yukarı yön tuşuna basıldığında hareket etmesi için gereken kod blokları verilmiştir. Resmin altına blokların ne işe yaradığını yazınız.



Aşağıdaki resimde karışık olarak verilen bloklar vardır. Bu blokları kullanarak; dalgıçın aşağı, sağa ve sola hareket etmesi için gereken kodları yazınız.



## **Ek 8. Hedef-Davranışlar**

**Hedef 1.** Programlamayla ilgili temel kavramların *anlam bilgisi*

**Davranış 1.**“algoritma, program, programlama dili, doğrusal mantık yapısı, karar yapısı, döngü yapısı, Scratch programı” kavramlarının tanımlarını; ne anlama geldiğini yazma, söyleme, seçip işaretleme, eşleştirme

**Hedef 2.** Algoritmayla ilgili *ilkeler bilgisi*

**Davranışlar**

**2.1.** Algoritma yazılırken bir başlangıcı olduğunu yazma, söyleme, seçip işaretleme, eşleştirme

**2.2** Algoritma yazılırken her adımda bir işin gerçekleşmesi gerektiğini yazma, söyleme, seçip işaretleme, eşleştirme

**2.3.**Algoritma yazımının en kısa şekilde tamamlanması gerektiğini yazma, söyleme, seçip işaretleme, eşleştirme

**2.4.**Algoritma yazılırken bir bitişi olduğunu yazma, söyleme, seçip işaretleme, eşleştirme

**Hedef 3.** Doğrusal mantık yapısıyla ilgili *ilkeler bilgisi*

**Davranış 3.**Doğrusal mantık yapısının; işlemleri sırasıyla çalıştırdığını yazma, söyleme, seçip işaretleme, eşleştirme

**Hedef 4.** Döngü yapısıyla ilgili *ilkeler bilgisi*

**Davranış 4.** Döngü yapısının işlemleri; bazı işlemi tekrarlamak için kullanıldığını yazma, söyleme, seçip işaretleme, eşleştirme

**Hedef 5.** Karar yapısıyla ilgili *ilkeler bilgisi*

**Davranış 5.** Karar yapısının işlemleri; iki ya da daha fazla seçenek arasından seçim yapmak hakkı tanıyan ve ona göre devam ettirmek için kullanıldığını yazma, söyleme, seçip işaretleme, eşleştirme

**Hedef 6.**Doğrusal mantık yapısını kullanarak oluşturulmuş problemlerle ilgili temel *ilkeleri açıklayabilme*

**Davranış 6.** Doğrusal mantık yapısının işlemleri neden sırasıyla çalıştırdığını açıklayarak yazma, söyleme, seçip işaretleme, eşleştirme

**Hedef 7.** Döngü yapısını kullanarak oluşturulmuş problemlerle ilgili temel *ilkeleri açıklayabilme*

**Davranış 7.** Döngü yapısının işlemleri neden bazı işlemi tekrarlamak için kullanıldığını **açıklayarak** yazma, söyleme, seçip işaretleme, eşleştirme

**Hedef 8.** Karar yapısını kullanarak oluşturulmuş problemlerle ilgili temel *ilkeleri açıklayabilme*

**Davranış 8.** Karar yapısının işlemleri neden iki ya da daha fazla seçenek arasından seçim yapmak hakkı tanıyan ve ona göre devam ettirmek için kullanıldığını açıklayarak yazma, söyleme, seçip işaretleme, eşleştirme

**Hedef 9.** Verilen farklı örnekler için algoritma yazarak problemleri çözebilme

**Davranış 9.** Verilen farklı örnekler için algoritma yazarak problemleri çözebilme

**Hedef 10.** Doğrusal mantık yapısı kullanılarak oluşturulan *problemleri çözebilme*

**Davranış 10.** Doğrusal mantık yapısını kullanarak verilen problemlerin çözümünü yapma, seçme işaretleme

**Hedef 11.** Döngü yapısını kullanarak algoritma oluştururken *problemleri çözebilme*

**Davranış11.** Döngü yapısını kullanarak verilen problemlerin çözümünü yapma, seçme işaretleme

**Hedef 12.** Karar yapısı kullanılarak oluşturulan *problemleri çözebilme*

**Davranış 12.** Karar yapısını kullanarak verilen problemlerin çözümünü yapma, seçme işaretleme

**Hedef 13.** Farklı durumlara uygun kod bloklarını kullanarak *problemleri çözebilme*

**Davranış 13.** Durumlara uygun olarak kod yazabilmek için verilen problemlerin çözümünü yapma, seçme işaretleme



**T. C.**  
**NECMETTİN ERBAKAN ÜNİVERSİTESİ**  
**Eğitim Bilimleri Enstitüsü Müdürlüğü**



### ÖZGEÇMİŞ

**Adı ve Soyadı** : Rukiye Berna BALA  
**Doğum Yeri** : Konya  
**Doğum Tarihi** : 16.08.1991  
**Medeni Durumu** : Bekâr

#### Öğrenim Durumu

**İlköğretim** : Cumhuriyet Mahallesi Ahmet Haşhaş İlköğretim Okulu  
**Lise** : Cumhuriyet Lisesi  
**Lisans** : Gazi Üniversitesi Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümü  
**Çalıştığı Kurum** : MEB  
**Adres** : r.berna.bala@gmail.com