



T.C.

NECMETTİN ERBAKAN ÜNİVERSİTESİ

Bilgisayar ve Öğretim Teknolojileri Eğitimi Anabilim Dalı

Bilgisayar ve Öğretim Teknolojileri Eğitimi Bilim Dalı

Yüksek Lisans Tezi

FARKLI GÖRSEL PROGRAMLARLA TASARLANAN KODLAMA  
EĞİTİMİNİN ORTAOKUL ÖĞRENCİLERİNİN AKADEMİK BAŞARI İLE  
KODLAMAYA KARŞI TUTUM VE ÖZ-YETERLİKLERİNE ETKİSİ

Teslime DENİZ

Danışman

Doç.Dr. Agah Tuğrul KORUCU

Konya 2021

## TEŐEKKÜR

Arařtırmanın belirlenmesinden itibaren süreçteki birçok problemin üstesinden gelmem de bana yardımcı olan, bitirme aşamasına kadar mesleki deneyimi ile rehber olan, engin hoşgörüsünü, sabrını ve zamanını hiçbir zaman esirgemeyen değerli tez danışmanım Doç. Dr. Agâh Tuğrul KORUCU' ya teşekkürlerimi sunarım.

Son olarak ayrıca bu günlere gelmemde büyük pay sahibi olan babam Mehmet KARAGÜLMEZ, annem Sare KARAGÜLMEZ ve kardeşim Ali KARAGÜLMEZ'e, yüksek lisans eğitimim boyunca benden desteklerini esirgemeyen sevgili eşim Fehmi DENİZ ve kızım Gökçe Zehra DENİZ'e sonsuz sevgilerimi ve teşekkürlerimi sunarım.

Teslime DENİZ

KONYA- 2021

## İÇİNDEKİLER

TEŞEKKÜR.....	ii
İÇİNDEKİLER .....	iii
TEZ ÇALIŞMASI ORJİNALLİK RAPORU.....	V
BİLİMSEL ETİK BEYANNAMESİ.....	Vi
SİMGELER VE KISALTMALAR .....	Vii
ÖZET .....	İx
ABSTRACT.....	X
1 GİRİŞ .....	1
1.1 Problem Durumu.....	1
1.2 Araştırmanın Amacı.....	3
1.3 Araştırmanın Önemi.....	4
1.4 Sayıtlar (Varsayımlar) .....	6
1.5 Sınırlılıklar .....	6
1.6 Tanımlar.....	7
2 ALAN YAZIN (İLGİLİ ARAŞTIRMALAR).....	9
2.1 Uzaktan Eğitim .....	9
2.2 Yapılandırıcı Yaklaşım .....	11
2.3 21. Yüzyıl Becerileri.....	12
2.4 Eğitimde Teknoloji Entegrasyonu .....	13
2.5 Kodlama.....	15
2.6 Blok Tabanlı Kodlama.....	18
2.6.1 Scratch.....	20
2.7 Unplugged (Bilgisayarsız) Kodlama.....	21
2.7.1 Unplugged (Bilgisayarsız) Kodlama Örnekleri .....	22
3 YÖNTEM .....	24
3.1 Araştırmanın Modeli.....	24
3.2 Araştırmanın Çalışma Grubu .....	25
3.3 Veri Toplama Araç ve/veya Teknikleri .....	26
3.3.1.Akademik Başarı Ölçeği.....	26
3.3.2. Bilgisayar Destekli Eğitime İlişkin Tutum Ölçeği.....	30
3.3.3. Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algı Ölçeği.....	31
3.3.4. Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeği.....	32
3.4 Verilerin Toplanması .....	35
3.5 Verilerin Çözümlemesi .....	41

4 BULGULAR.....	42
4.1 Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algı Ölçeği Güvenirlik Analizi Bulguları .....	42
4.2 Bilgisayar Destekli Eğitime İlişkin Tutum Ölçeği Güvenirlik Analizi Bulguları .....	42
4.3 Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeği Güvenirlik Analizi Bulguları.....	42
4.4 Deney Grubu Ön Test - Son Test Sonuçlarının İncelenmesi.....	43
4.5 Kontrol Grubu Ön Test - Son Test Sonuçlarının İncelenmesi.....	45
4.6 Deney – Kontrol Grubu Ön Test - Son Test Sonuçlarının İncelenmesi.....	47
5 TARTIŞMA, SONUÇ VE ÖNERİLER .....	50
5.1 Sonuç ve Tartışma.....	50
5.2 Öneriler .....	52
KAYNAKÇA.....	54
EKLER.....	60
EK 1 Bilgisayar Destekli Eğitim Yapmaya İlişkin Tutum Ölçeği.....	60
EK 2 Bilgisayar Destekli Eğitim Yapmaya İlişkin Öz Yeterlik ve Algı Ölçeği.....	61
EK 3 Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeği.....	62
EK 4 Scratch Eğitimi Akademik Başarı Testi .....	64
EK 5 Tospaa Eğitimi Akademik Başarı Testi.....	67
Çalışma Kağıdı 1.....	69
Çalışma Kağıdı 2.....	70
Çalışma Kağıdı 3.....	71
Çalışma Kağıdı 4.....	72
Çalışma Kağıdı 5.....	73
Çalışma Kağıdı 6.....	74
Çalışma Kağıdı 7.....	75
Scratch Nedir? İsimli Word Belgesi .....	76
Scratch Nedir 2? İsimli Word Belgesi .....	77
Scratch Nedir 3? İsimli Word Belgesi .....	80
Scratch Nedir 4? İsimli Word Belgesi .....	81
Scratch Nedir 5? İsimli Word Belgesi .....	82
Scratch Akvaryum Uygulaması .....	84
Scratch: Papağan Uygulaması.....	86
Scratch: Yön Tuşları Uygulaması.....	88

## TEZ ÇALIŞMASI ORJİNALLİK RAPORU

*Farklı Görsel Programlarla Tasarlanan Kodlama Eğitiminin Ortaokul Öğrencilerinin Akademik Başarı ile Kodlamaya Karşı Tutum ve Öz-Yeterliklerine Etkisi* başlıklı tez çalışmamın İç Kapak, Özetler, Ekler ve Ana Bölümlerden (Giriş, Alan Yazın, Yöntem, Bulgular, Tartışma, Sonuçlar ve Öneriler) oluşan toplam **98** sayfalık kısmına ilişkin, 2/07/2020 tarihinde tez danışmanım tarafından **Turnitin** adlı intihal tespit programından aşağıda belirtilen filtrelemeler uygulanarak alınmış olan orijinallik raporuna göre, tezimin benzerlik oranı **%16** olarak belirlenmiştir.

Uygulanan filtrelemeler:

1. Tez kabul sayfası hariç,
2. Tez çalışması orijinallik raporu sayfası hariç,
3. Bilimsel etik beyannamesi sayfası hariç,
4. Önsöz hariç,
5. İçindekiler hariç,
6. Simgeler ve kısaltmalar hariç,
7. Kaynakça hariç
8. Özgeçmiş hariç,
9. Alıntılar dâhil,
10. 7 kelimedenden daha az örtüşme içeren metin kısımları hariç

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

12/07/2021

Teslime DENİZ

Doç. Dr. Agah Tuğrul KORUCU

## **BİLİMSEL ETİK BEYANNAMESİ**

Bu tezin tamamının kendi çalışmam olduğunu, planlanmasından yazımına kadar tüm aşamalarında bilimsel etiğe ve akademik kurallara özenle riayet edildiğini, tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez hazırlama kurallarına uygun olarak hazırlanan bu çalışmada başkalarının eserlerinden yararlanılması durumunda bilimsel kurallara uygun olarak atıf yapıldığını ve bu kaynakların kaynakça listesine eklendiğini beyan ederim.

12/07/2021

Teslime DENİZ

## SİMGELER VE KISALTMALAR

### Simgeler

f: Frekans

$\bar{X}$  : Ortalama

% : Yüzde

N: Katılan Kişi Sayısı

P: Anlamlılık Deęeri

Sd: Serbestlik Derecesi

Ss: Standart Sapma

T: t-deęeri

## **Kısaltmalar**

MIT: Massachusetts Institute of Technology

MEB: Millî Eğitim Bakanlığı

STEM: Science, Technology, Engineering and Math

FeTeMM: Fen, Teknoloji, Mühendislik ve Matematik

SPSS: Statistical Package for the Social Sciences

BİT: Bilgi ve iletişim teknolojileri

EBA: Eğitimde Bilişim Ağı

FATİH: Fırsatları Artırma ve Teknolojiyi İyileştirme Hareketi

BÖTE: Bilgisayar ve Öğretim Teknolojileri Bölümü

BTÖYA: Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algı Ölçeği

## ÖZET

Bilgisayar ve Öğretim Teknolojileri Eğitimi Anabilim Dalı  
Bilgisayar ve Öğretim Teknolojileri Eğitimi Bilim Dalı  
Yüksek Lisans Tezi

### FARKLI GÖRSEL PROGRAMLARLA TASARLANAN KODLAMA EĞİTİMİNİN ORTAOKUL ÖĞRENCİLERİNİN AKADEMİK BAŞARI İLE KODLAMAYA KARŞI TUTUM VE ÖZ-YETERLİKLERİNE ETKİSİ

Teslime DENİZ

Bu araştırma öğrencilere kodlama eğitimi verilirken öğrencilerinin bilgisayar ve yazılımı dersine karşı olan tutumlarını ölçmek, öğrencilerin bilgisayar ve yazılımı dersine karşı öz yeterlik algılarını belirlemek, öğrencilerin kodlama eğitimine yönelik tutumlarını ölçmek, öğrencilerin akademik başarılarını ölçmek ve bu ölçümler sonucunda ortaya çıkan eğitim çıktılarının farklılıklarını gözlemlemek amacı ile gerçekleştirilmiştir. Farklı görsel programlarla tasarlanan kodlama eğitiminin ortaokul öğrencilerinin akademik başarı ile kodlamaya karşı tutum ve öz-yeterliklerine etkisi inceleyen bu araştırmada nicel araştırma yöntemlerinden yararlanılmıştır. Bu çalışmada araştırma modeli olarak nicel araştırma yöntemi benimsenmiş olup “Ön test – Son test Kontrol Gruplu Yarı Deneysel Desen Modeli” kullanılmıştır. Çalışma grubunu 2020-2021 eğitim öğretim yılında Konya Gödene TOKİ Ortaokulunda öğrenim gören 17 kız ve 28 erkek öğrenciden olan 6. sınıf öğrencisi oluşturmaktadır. Uygulamalar öğretim programında ders planı süresine uygun olarak 10 hafta boyunca yapılmıştır. Araştırmada derse yönelik akademik başarı testi geliştirilmiş Bilgisayar Destekli Eğitime İlişkin Tutum Ölçeği, Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algı Ölçeği, Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeği belirlemek amaçlı 3 farklı ölçekten yararlanılmıştır. Toplamda 10 hafta süren eğitimlerin sonucunda elde edilen veriler IBM SPSS Statistics 22 programında analiz edilmiştir. Kullanılan ölçeklerin Cronbach Alfa güvenirlik katsayıları ise Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algı Ölçeği .870, Bilgisayar Destekli Eğitime İlişkin Tutum Ölçeği .81, Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeği .92 olarak hesaplanmıştır. Kullanılan akademik başarı ölçeklerinin güçlük indeksleri ise Scratch Akademik Başarı Testi .758 Tospaa Akademik Başarı Testi .726 olarak hesaplanmıştır. Elde edilen verilere göre geliştirilen ölçek geçerli ve güvenilir bulunmuştur. Araştırmanın sonunda elde edilen bulgulara göre Scratch programının kodlama eğitiminde uygulamasının öğrencilerin programlama becerileri üzerinde anlamlı bir farklılık oluşturduğu ve kodlama eğitimine katkı sağladığı görülmüştür. Bilgisayarlı kodlama ortamının, Akademik Başarı, Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algı Ölçeği, Bilgisayar Destekli Eğitime İlişkin Tutum Ölçeği ve Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeğinin toplam üzerindeki etki büyüklüğünü belirlemek için eta kare değeri incelenmiştir. Akademik başarı, bilgisayar destekli eğitimde öz-yeterlilik algılarını, bilgisayar destekli eğitime tutumlarını ve eğitsel bilgisayar oyunları destekli kodlama eğitimi üzerinde “geniş” bir etki büyüklüğüne sahip olduğu söylenebilir.

**Anahtar Kelimeler:** bilgisayarlı kodlama, akademik başarı, bilgisayar destekli eğitimde öz-yeterlilik algıları, bilgisayar destekli eğitim tutumları, eğitsel bilgisayar oyunları destekli kodlama eğitimi

## ABSTRACT

Department of Computer Education and Instructional Technology  
Computer Education and Instruction Technology Program  
Master Thesis

### THE EFFECT OF CODING EDUCATION DESIGNED WITH DIFFERENT VISUAL PROGRAMS ON ACADEMIC SUCCESS AND ATTITUDES AND SELF-EFFICIENCIES OF SECONDARY SCHOOL STUDENTS

Teslime DENİZ

This research was conducted with the aim of measuring students' attitudes towards computer and software course while giving coding education to students, determining students' self-efficacy perceptions towards computer and software course, measuring students' attitudes towards coding education, measuring students' academic success and observing the differences in educational outcomes resulting from these measurements. has been carried out. Quantitative research methods were used in this study, which examines the effects of coding education designed with different visual programs on the academic achievement, attitudes and self-efficacy of secondary school students. In this study, the quantitative research method was adopted as the research model and the "Pretest - Posttest Quasi-experimental Design Model with Control Group" was used. The population of the research consists of 6th grade students, 17 girls and 28 boys, studying at Konya Gødene TOKİ Secondary School in the 2020-2021 academic year. The applications were made for 10 weeks in accordance with the lesson plan period in the curriculum. In the study, an academic achievement test for the course was developed, the Attitude Scale towards Computer-Aided Education, the Self-Efficacy Perception Scale of Computer-Aided Education, the Attitude Scale towards Computer-Aided Coding Learning 3 different scales were used to determine the . The data obtained as a result of the training, which lasted for 10 weeks in total, were analyzed in the IBM SPSS Statistics 22 program. The Cronbach Alpha reliability coefficients of the scales used were calculated as .870, for the Self-Efficacy Perception Scale for Computer Assisted Education, .81 for the Attitude Scale for Computer Assisted Education, and .92 for the Attitude Scale for Educational Computer Games Assisted Coding Learning. The difficulty indexes of the academic achievement scales used were calculated as .758 in the Scratch Academic Achievement Test and .726 in the Tospaa Academic Achievement Test. The scale developed according to the data obtained was found to be valid and reliable. According to the findings obtained at the end of the research, it was seen that the application of the Scratch program in coding education made a significant difference on the programming skills of the students and contributed to the coding education. The eta squared value was examined to determine the effect size of the computer coding environment on the total of Academic Achievement, Self-efficacy Perception Scale for Computer Aided Education, Attitude Scale towards Computer Assisted Education and Attitude Scale towards Educational Computer Games Supported Learning. It can be said that academic achievement has a "large" effect size on self-efficacy perceptions in computer-assisted education, attitudes towards computer-assisted education and educational computer-assisted coding education.

**Keywords:** computerized coding, academic success, self-efficacy perceptions in computer assisted education, computer assisted education attitudes, educational computer games supported coding training

# BÖLÜM 1

## 1 GİRİŞ

Bu bölümde araştırmanın problem durumu, araştırmanın amacı, araştırmanın önemi, varsayımlar, sınırlılıklar ve tanımlamalar hakkında bilgi sunulmuş ve araştırmada kullanılan terimlere ilişkin kavramsal açıklamalar yapılmıştır.

### 1.1 Problem Durumu

Günümüz bilgi ve teknoloji çağı olarak adlandırılmaktadır ve bizler de bilgi ve teknoloji çağında yaşamaktayız. Yaşadığımız dönemde öğrencilerin bilgileri ezberlemesinden ziyade mevcut bilgiye öğrencinin kendinin ulaşabilmesi ve karşılaştığı problemleri bilimsel süreç becerileriyle çözebilmeleri beklenir. Çağın olağan bir getirisi olarak günlük yaşantımızın her alanında teknoloji bulunmaktadır. Teknoloji sağlık, mühendislik, bankacılık, güvenlik ve bunun gibi birçok alanda kullanılmakta bunun sonucunda eğitim alanında da teknolojinin kullanımı bir ayrıcalık olmaktan çıkıp zorunluluk haline gelmiştir (Geçer ve Funda, 2010; Somyürek, 2014). Teknolojideki bu denli hızlı gelişimlerden ötürü yenilikler ve eğitim birbirini senkron bir şekilde takip etmektedirler (Batdal, 2005). Tüm alanlarda gelişen ve ilerleyen teknoloji eğitim ve öğretim sürecinde de aktif olarak kullanılmaya başlanılmıştır (Baş, 2011; Tsai ve Tsai, 2003). Kısa bir süre önce hayatımıza giren 21. yüzyıl becerileri kavramı, çocuklarımızı gelecekteki dünyaya hazırlar aynı zamanda bireylerin hangi becerilerle donatılması gerektiğini üzerinde çalışılmaktadır (Roy-Singh, 1991). Eğitim de meydana gelen değişimlerden dolayı, yaşadığımız yüzyılda bilgiyi tüketmeleri yerine üretmeleri beklenmektedir. 21. Yüzyılın ve çağın istediği birey, kendisine veriler bilgileri olduğu gibi kabul edip ezberleyen, yönlendirilmeyi ve yönetilmeyi bekleyen değil; aktarılan bilgiyi yorumlayan, anlamlandıran ve tüm bunlardan sürece aktif katılım sağlayan bireylerdir (Yıldırım ve Simsek, 2008). Öğrencilere bu becerilerin verilebilmesi için, bu becerilerin eğitim sistemlerinin içerisinde yer alması gerekmektedir. Özellikle bu becerilerin teknoloji ile entegre edilmesi hesaplamalı düşünme fikrini ortaya çıkarmış ve ülkeler eğitim sistemleri içerisinde STEM eğitim modelini entegre etmeye başlamışlardır (Demir ve Seferoğlu, 2017). STEM kısaltması Science, Technology, Engineering ve Mathematics terimlerinin baş harfleri ile oluşturulmuştur. STEM eğitimi fen ve matematik dersi konularının bütüncül bir yaklaşımla derslerin uygulamalı bir

olarak işlenebilmesini sağlamaktadır (Kırkıcı ve Aydın, 2018). STEM eğitimi günlük yaşamla bu disiplinlerin bütünleşmiş bir şekilde ilişkilendirilerek verilmesini sağlayan bir eğitim yöntemidir (Yıldırım ve Selvi, 2017) .STEM eğitiminin uygulanmasının artması ile başka yenilikçi yaklaşım modelleri de eğitimde uygulanmaya başlamıştır. Günümüzde yenilikçi yaklaşım yöntemlerinden biri olarak uygulanan eğitimlerden birisi de kodlama eğitimidir. Çağımızda gelişmiş ve gelişmekte olan ülkeler öğrencilere kodlama becerilerinin kazandırılabilmesi için mevcut eğitim sistemlerine kodlama eğitimini entegre etmeye yönelik çalışmalar yürütmektedirler. Zamanımızın ve geleceğin gerekli becerilerine sahip olan insanlar yetiştirebilmek için 21. Yüzyıl becerilerinden olan dijital okur-yazarlık becerisi için kodlama eğitimi oldukça önem kazanmıştır. Teknolojinin hızlı bir şekilde günlük hayatımıza etki etmesi ve ilerleyen teknolojilerin hem öğrencinin hem de öğretmenin eğitim hayatlarına dahil olmasıyla kodlama eğitiminin önemi artmıştır. Bunun sonucunda okullarda kodlama eğitimi yaygınlaşmaya başlamıştır. Ortaokullarda kademeli olarak bilişim teknolojileri ve yazılım dersi müfredatı içerisinde yer almaktadır (Ünsal, 2019).

Yapılan araştırmaların sonuçlarında ilköğretim ve ortaöğretim kademelerinde öğrenim gören öğrencilerin en başarısız oldukları alanlardan birisinin de problem çözme becerisinin olduğu belirlenmiştir. Öğrencilerin programlamayı öğrenmede için karşılaştığı zorluk çoğunlukla problem çözme becerilerinin olmamasıdır (Gomes ve Mendes, 2007). Scratch programının amacı çocuklarda ve gençlerde teknolojik kullanımı iyileştirmektir. Çocukların ve gençlerin programlamayı öğrenmeleri için ilk dil olarak Scratch programı uzmanlar tarafından tavsiye edilmektedir (Malan ve Leitner, 2007).

Türkiye’de, Bilişim Teknolojileri ve Yazılım dersi ortaokul ve imam hatip ortaokullarında 5. ve 6. sınıf seviyesinde haftada iki ders saati zorunlu olarak, 7. ve 8. sınıf seviyelerinde ise haftada iki ders saati olarak seçmeli bir ders olarak okutulmaktadır. Bilişim Teknolojileri ve Yazılım dersinin Millî Eğitim Bakanlığı tarafından açıklanan ve öğretim programında yayımlanan öğrenme alanlarının başlıkları şu şekilde belirlenmiştir;

1. Bilişim okur-yazarlığı,
  2. Bilişim teknolojileri aracılığı ile iletişim kurma, bilgi paylaşma ve kendini ifade edebilme,
  3. Araştırma yapabilme, bilgiyi yapılandırabilme ve iş birlikli çalışma,
  4. Problem çözebilme, programlama yapabilme ve özgün ürünler ortaya koyabilme,
- Bir problemi çözebilmek ve projeyi gerçekleştirebilmek amacıyla strateji geliştirebilir, çözüm yolu üretirken farklı bakış açıları ve yeni yaklaşımlar kullanabilir.
  - Yazarlık ve programlama dillerinin neler olduğunu tanıyabilir, minimum bir programlama dilini aktif bir biçimde kullanabilir.
  - Konuları ve sistemleri incelemek için model oluşturabilir, benzeşimler yapabilir ve canlandırmalar oluşturabilir (M.E.B., 2012).

Problem çözme, programlama ve özgün ürün geliştirme ünitesinde Scratch programının ve Tospaa bilgisayarsız kodlama programının kullanmanın uygun olacağı düşünülmektedir. Scratch programı sayesinde öğrenciler kendi projelerinin tasarımlarını yaparak yaşamları boyunca karşılaştıkları problemlere daha yaratıcı çözümler üretebilirken, Tospaa ile de problem çözme basamaklarını kavrayıp gerçek hayatta karşılaştıkları problemlere daha algoritmik çözüm yolları uygulayabilirler (Karabak ve Güneş, 2013; Lee, 2011).

Bu çalışmada, Scratch ve Tospaa programları kullanılarak mevcut problemlerin çözümü öğrenciler tarafından gerçekleştirilmiştir. Scratch ve Tospaa kolay, görsel olarak ilgi çekici ortamlar sağladıklarından ötürü öğrenciler problem çözme aşamasında tasarım ortamında aktif rol almışlardır.

## **1.2 Araştırmanın Amacı**

Bu araştırma öğrencilere kodlama eğitimi verilirken öğrencilerinin Bilişim Teknolojileri ve Yazılımı dersine karşı olan tutumlarını ölçmek, öğrencilerin Bilişim Teknolojileri ve Yazılımı dersine karşı öz yeterlik algılarını belirlemek, öğrencilerin kodlama eğitimine yönelik tutumlarını ölçmek, öğrencilerin akademik başarılarını ölçmek ve bu ölçümler sonucunda ortaya çıkan eğitim çıktılarının farklılıklarını

gözlemlemektir. Deneklerin bilgisayar destekli ve bilgisayar desteksiz eğitim ortamından hangisinde kodlama eğitimini daha verimli biçimde kavradığı belirlenmek istenilmektedir.

1. Bilgisayar kullanılmadan öğrencilere aktarılacak kodlama eğitimi öğrencilerde ön test ve son test sonrasında anlamlı bir farklılık sağlayacak mıdır?
2. Bilgisayar Destekli Eğitim ortamında öğrencilere aktarılacak olan kodlama eğitimi öğrencilerde ön test ve son test sonrasında anlamlı bir farklılık sağlayacak mıdır?
3. Öğrenciler bilgisayarlar kullanımına karşı hazır bulunurlukları ne seviyede?
4. Öğrencilerin Kodlama hakkında ön öğrenmeleri nelerdir?
5. Öğrencilerinin akademik başarı testinden aldıkları puanlar da dersin işleme stratejisi etkili olmuş mudur?

### **1.3 Araştırmanın Önemi**

Bilişim teknolojilerine eğitim ortamlarında daha çok ihtiyaç duyulmaya teknolojinin oldukça hızlı gelişmesi başlanmıştır ve bilişim teknolojilerinin önemi çağımızda giderek artmıştır. Bilişim teknolojileri alanı içinde yeni programların geliştirilmesini ve programların yazılması meydana gelen bu önem doğrultusunda önemli hale getirmiştir (Keskinsoy, 2010). Mevcut ihtiyacı karşılayabilmek için yaratıcı ve üretken bir nesil yetiştirilmesi gerekmekte bununla birlikte bireylere erken yaştan itibaren programlama eğitimi verilmelidir (Demirer ve Nurcan, 2016). Gelişen ve büyüyen dünyada yeni nesillerin mevcut programları tüketmelerini izlemektense onlara yeni programları nasıl üretebileceklerini konusunda yol gösterilmesi gerekmektedir (Kert ve Uğraş, 2009). Programın tanımını ise şu şekildedir; bilgisayar ortamında yazılıp bilgisayar komutları yardımı ile çalıştırılarak belli bir işlemlerin ve işlerin yapılmasını sağlayan yapıdır (Coşar, 2013).

Günümüzde geleneksel okuryazarlığa ek olarak teknolojik gelişmeler ışığında, e-okuryazarlık, bilgi okuryazarlığı, medya okuryazarlığı, görsel okuryazarlık ve bilgisayar okuryazarlığı gibi okuryazarlık türleri ortaya çıkmıştır (Kurudayıoğlu ve Tüzel, 2010). Bilgisayar yoluyla eğitim, çocukların oyun ile birlikte öğrenmesini sağlayan, çocuklar için oldukça ilgi çekicidir, çocuklar bilgisayarın tuşlarına basarak, ekranda oluşan görüntülerini oynayarak, farklı şekiller ve sonuçlar elde eder. Oyun sırasında çocuk fark

etmeden yeni bilgiler öğrenmektedir (Arıcı ve Demir, 2009). Bahsettiğimiz okuryazarlıkların gelişmesi ile çağımızda kodlama eğitimi çok dikkate alınan bir konuya başlamıştır. Türkiye’de de kodlama eğitimi konusu son zamanlarda en çok tartışılan bir konulardan olmaya başlamıştır. Kodlama mantıksal akıl yürütmenin bir parçası olarak akademik bir beceri olarak görülmekte ve günümüzde “21. yüzyıl becerileri” diye adlandırılan becerilerin bileşenlerinden biri olarak kabul edilmektedir (Sayın ve Seferoğlu, 2016). Genç neslin kodlama mantığını çocuk yaşlarda öğrenmeleri öğrencilerin diğer alanlardaki başarılarına da katkısı olacaktır. Çocuklar kodlamayı öğrenmeye başladıklarında tasarım sürecini kavrayacak; kodlamayı öğrenen öğrenci fikir bulacak, bulduğu fikrin uygulamasını yapacak, uygulamanın hatalar olursa bunları düzeltecek, son adımda da çevresi ile iş birliği yaparak çalışacaktır (Demirer ve Nurcan, 2016; Karabak ve Güneş, 2013). Başka bir deyişle kodlama yapmayı çocukların ve gençlerin karşısına çıkan problemlere daha hızlı çözümler üretebildikleri, yaptıkları hatalarını daha kolay çözebildikleri ve bu hataların sonuçlarını analiz edebildikleri gözlemlenmektedir (Baz, 2018).

Bilgisayarca düşünme kavramı son zamanlarda alan yazında kullanımını gün geçtikçe artan kavramlardan birisi haline olmuştur. Bilgisayarca düşünme, ortada olan bir dizi problemin çözümünün yapılması için tasarım, algoritmik düşünme ve uygulama geliştirme aşamaları olarak açıklanabilir. Problemin tanımlanması, problemin anlaşılması ve problemin çözümlenmesine yönelik olarak algoritmaları ortaya çıkarma adımlarını içermektedir (Thomas ve diğerleri, 2015). Bilgisayarca düşünme kavramında yer alan algoritmik düşünme ise programlamayla yakından ilişkilidir. Tüm programlama dilleri algoritmalar tarafından oluşturulmaktadır, programlama dilleri yalnızca kod satırlarından ibaret değildir bunun yanında mevcut problemin çözümü için mantıklı bir şekilde atılacak adımları sıralamaktadır (Yünkül ve diğerleri, 2017). Kısacası programlama, çocukların bilgisayar gibi düşünme becerilerini arttırmalarına olanak verir (Lawanto ve diğerleri, 2017).

Yakın zamanda yapılan araştırmalar ile kodlama eğitimi alan çocukların karşısına çıkan problemler karşısında çözüm geliştirme becerilerini arttırdıkları ortaya çıkarmıştır. Kodlama yeteneği sonucunda çocukların yaptıkları yanlışlara daha kolay çözüm yolu üretebildikleri, sonuçlarını değerlendirme becerilerinin de arttığı belirtilmektedir (Coravu ve diğerleri, 2015; Resnick ve Silverman, 2005). Zor ve sıkıcı

bir süreçten oluşan programlama eğitimi bu sebep nedeni ile erken yaştaki çocukların ilgisini yeterince çekmemektedir. Fakat çağımızda kodlama öğretiminde öğrencilerin eğlenceli ve kolay bir şekilde kod yazabilmeleri için kodlamayı öğrenciler için eğlenceli hale getirecek çeşitli kodlama araçları geliştirilmektedir (Demirer ve Nurcan, 2016).

Bu bağlamda yeni nesil öğrencilerin bilgisayar okuryazarlığı ve e-okuryazarlık kapsamında yetiştirilmesi gerekmektedir. Yapılan eğitimler ile öğrencilerin 21. yy becerilerini kazanmış olmaları öğrencilerin gelecekteki iş istihdamlarını da etkileyecektir bu bilgiler ışığında öğrencilere başarılı bir kodlama eğitimi vermek onları gelecek çağlara hazırlamakta doğru bir adım olacaktır. Kodlamanın çağımızın ve geleceğin en önemli bileşenlerinden birisi olması kaçınılmaz bir gerçektir. Bundan dolayı da öğrencilerimize kaliteli kodlama eğitimi ve problem çözme becerilerini kazandırmamız gerekmektedir.

#### **1.4 Sayıtlılar (Varsayımlar)**

1. Bu araştırma kapsamında katılımcılara ölçekler uygulanmıştır. Dolayısıyla, katılımcıların ölçeklerde yer alan soruları, düşüncelerini doğru yansıtacak şekilde içtenlikle yanıtladıkları varsayılmıştır.
2. Araştırma kapsamında bulunan öğrenciler aynı koşullarında değerlendirme yapmışlardır.

#### **1.5 Sınırlılıklar**

1. Bu araştırma 2020-2021 öğretim yılında Konya Meram Gödene TOKİ İmam Hatip Ortaokulunda öğrenim gören 45 öğrencinin görüşleriyle sınırlıdır. Ayrıca öğrencilerin düşünce ve görüşleri ölçeklerde yer alan soruların kapsamı ile sınırlıdır.
2. Bulgular 2020-2021 eğitim-öğretim yılı birinci dönem de elde edilen verilerdir.
3. Bu çalışma için öğrencilerle toplam 10 haftalık bir çalışma ile yapılmıştır.
4. Araştırma Covid-19 nedeniyle uzaktan eğitim yöntemleri kullanılarak yapılmıştır.
5. Çalışma uzaktan eğitim yöntemi ile yapıldığı için bilgisayar laboratuvarı kullanılamamıştır.
6. Araştırma sırasında yapılması gereken Scraeth uygulamalarında öğrencilerin evlerinde bilgisayar mevcut olmasına ve kullanımı konusunda da öğrencilerin öz yeterliliklerine bağlı kalmıştır.

## 1.6 Tanımlar

**Programlama:** Mevcut bir problemin çözümü için doğru karar vermek, en kısa çözüm yolu bulmak ve yaşam koşullarını en üst seviyeye çıkarmak olarak tanımlanabilir (Aytekin ve diğerleri, 2018).

**Unplugged Kodlama:** Bilgisayar ile ilgili zor olarak belirtilen kavramları kolay, eğlenceli ve seviyeye uygun bir biçimde öğretmek için alternatif bir yöntem olarak kullanılmaktadır. Bilgisayar bilimi ile ilgili temel kavramlar ve bilgilerin bilgisayarın olmadığı ortamlarda kazandırılması için daha çok etkinlik temelli aktif bir öğrenme yaklaşımıdır. Bilgisayar biliminde problemin çözümüne yönelik algoritmaların oluşturulması ve şifrelemelerin bilgisayarsız ortamda öğrencilere aktarılması şeklinde gerçekleştirilmektedir (Tim Bell, 2015).

**Scratch:** Massachusetts Teknoloji Enstitüsü (MIT - Massachusetts Institute of Technology) tarafından 10-16 yaşları arasındaki çocukların ve gençlerin daha kolay bir biçimde kodlama öğrenmesi için geliştirilmiş, programlama dili yazılımıdır (Scratch, 2020).

**Blok tabanlı programlama:** Kullanıcıların öğrenmek için daha zor bir süreç olan soyut kod ve komutların adım adım somutlaştırılarak öğretilmesini ve anlık dönütler verilmesini sağlayan programlamadır (Saygıner ve Tüzün, 2017).

**Bilgi ve iletişim teknolojileri (B.İ.T.):** Çalışmalarda gerekli bilgiyi ortaya çıkarabilmek için yararlı olan tüm bilgisayar ve teknolojiyi içinde barındıran gereçlerdir (Gibbons ve diğerleri, 2000).

**Algoritma:** Bir problemi veya işlemi çözebilmek için hazırlanan adımlar dizisidir (Choi ve diğerleri, 2017).

**Algoritmik düşünme:** Bir problemi veya işlemi çözebilmek için hazırlanan adım adım düşünme biçimidir (Choi ve diğerleri, 2017) .

**STEM (Science, Technology, Engineering, Mathematics):** Bilim, teknoloji, mühendislik ve matematik terimlerinin İngilizce anlamlarının baş harflerinin (Science, Technology, Engineering, Mathematics) birleşmesi ile ortaya çıkan ve bu alanlarla

yapılan eğitimin entegre edilmesi sonucunda yapılan öğrenme işlemleridir (Aydeniz ve Bilican, 2018).

**21. Yüzyıl Becerileri:** İnsanların zamanın beklentilerine cevap verebilmesi ve kendi ihtiyaçlarını karşılayabilmesi için problem çözme, eleştirel düşünme, analiz etme, yenilik ve yaratıcılık gibi sahip olması gereken becerilerdir (Çolak, 2018).

**Eğitimde Teknoloji Entegrasyonu:** Eğitimde teknoloji entegrasyonu öğretmenlerin pedagojik yeterlikleri, öğretim programları, yapılacak faaliyetlerin maddi olarak uygun olması ve kurumun tamamının bu entegrasyona hazır olması gibi basamakları içermektedir (Bolat ve diğerleri, 2020).

## BÖLÜM 2

### 2 ALAN YAZIN (İLGİLİ ARAŞTIRMALAR)

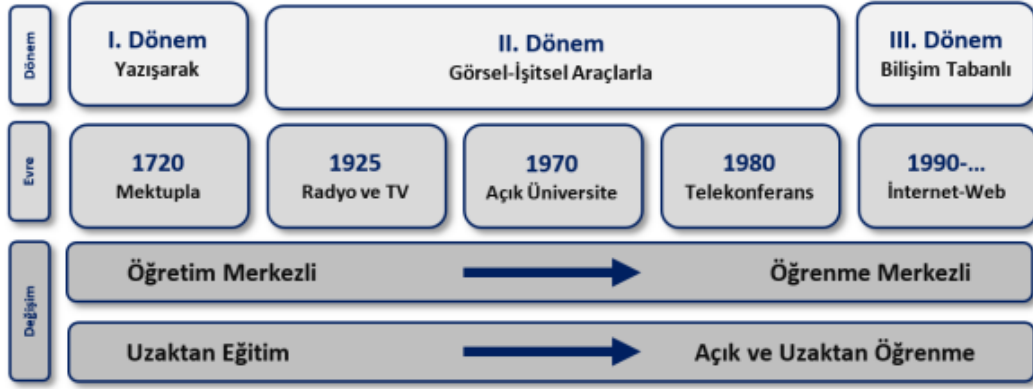
Bu bölümde çalışma kapsamında ele alınan çeşitli kaynaklarda bulunan bilgilerin taranması sonucu elde edilen literatür bilgileri yer almaktadır. Bu bağlamda uzaktan eğitim, yapılandırıcı yaklaşım, 21. yüzyıl becerileri, kodlama, blok tabanlı kodlama, scrath, unplugged (bilgisayarsız) kodlama ve tospaa ile ilgili yapılan çalışmalara yer verilmiştir.

#### 2.1 Uzaktan Eğitim

Geleneksel eğitim öğretim yöntemlerindeki sınırlılıklar sebebi ile sınıf içi etkinliklerin yürütülemediği durumlarda, eğitim çalışmalarını planlayan kişiler ve uygulayan kişiler ile öğrenenler arasında, etkileşim ve iletişimin özel olarak hazırlanmış öğretim programları ve çeşitli çevrim içi ortamlar yoluyla belli bir merkezden sağlandığı öğretim yöntemine uzaktan eğitim denir. Kısaca uzaktan eğitim, öğrenci ve öğretmenin birbirlerinden ayrı yerlerde, senkron veya asenkron zamanlarda iletişim kurması yoluyla oluşturulan eğitimidir (Odabaş, 2003).

Dünyada ilk uzaktan eğitim mektup yoluyla İngiltere’de yapılmıştır. Uzaktan eğitimde iletişimin üç temel elemanı vardır: öğrenen, öğretmen ve iletişim metodu. İletişim metodu öğrenen ile öğretene arasındaki bağıdır (Moore, 1973) . Kurulan bu bağ veya ortam teknolojinin herhangi bir şekli ile olmaktadır. Bunlar; TV yayını, posta, , bilgisayar ağı, radyo olabilir. Basılı materyal; görüntü, ses, ve grafik uzaktan eğitim için kullanılan ürünlerdir (Dinçer, 2016).

Uzaktan eğitim mevcut teknolojileri pragmatist yani faydacı bir yaklaşımla öğrenen, öğretene ve öğrenme kaynakları arasındaki sınırlılıkları ortadan kaldırmaya çalışan, bunu gerçekleştirebilmek için kullanan disiplinler arası bir yöntemdir. Uzaktan eğitimin evreleri ve dönemi araştırıldığında tüm süreçte yaygın olan bilgi ve iletişim teknolojilerinin öğrenme-öğretme ortamlarında kullanıldığı ve bu bilgi ve iletişim teknolojilerin uzaktan eğitimin dönem ve evrelerinin neler olduğunu belirlediği görülmektedir. Tüm bunlara ek olarak uzaktan eğitim aşamalarına bakıldığında açıklık, öğrenme, esneklik kavramlarına doğru bir yönelim olduğu dikkatten kaçmamaktadır (Bozkurt, 2017).



Şekil 2.1 Uzaktan eğitimin küresel bağlamda dönem ve evreleri (Bozkurt, 2016)

Türkiye’de 1923- 1960’lı yıllar arasında uzaktan eğitimin ilk olarak kavramsal olarak incelenmiştir. 1970’li yılların ardından ise farklı girişim yöntemleri ile orta eğitim seviyelerinde uzaktan eğitim çalışmaları yapılmış, bazı tecrübeler elde edilmiş ve azami miktarda ilerleme yaşanmıştır (Bozkurt, 2017). Uzaktan eğitimin gelişme sürecinde bahsettiğimiz radyo ve televizyon, kitaplar, bununla birlikte posta servisleri gibi araç gereçler kullanılmış, eğitimin kalitesini ve yöntemlerini tartışılabilir sistemler ortaya konulmuştur. Bu sistemlerin en güzel örneklerden birisi ise Anadolu Üniversitesinin ön lisans ve lisans eğitimi veren açık öğretim programlarıdır. Bununla birlikte açık lise olarak bilinen eğitim programında ülkemizde uzaktan eğitim ile ilgili örneklerdendir (Al ve Madran, 2004).

2019 yılı sonunda tüm dünya ülkelerinde başlayan COVID-19 salgını kısa zaman içerisinde ülkemizde de görülünce 2020 yılı Şubat ayı itibari ile bazı tedbirler alınmaya başlanılmıştır. Maske kullanımı, seyahat kısıtlamaları, hijyen tedbirleri ve karşımızdaki kişilerle aramıza koyacağımız sosyal mesafeye rağmen salgın kontrol altına alınamamış son olarak da tüm ülkede sokağa çıkma yasakları uygulanmıştır. Alınan bu tedbirlerden eğitim kurumları oldukça etkilenmiş ve yapılan toplantılar doğrultusunda eğitime ara verilmiştir. Salgının oldukça hızlı yayılmaya devam edince örgün eğitim öğretimden vazgeçilmiş ve uzaktan eğitim öğretim sistemine geçiş kararı alınmıştır ve alınan kararın tüm eğitim kurumlarında uygulanmasına karar verilmiştir (Kahraman, 2020). Alınan kararın ardından hemen sonrasında, Milli Eğitim Bakanlığına bağlı kurumlar EBA sistemini kullanarak derslerini uzaktan eğitim modeliyle işlemeye başlamıştır. Bu sayede Milli Eğitim Bakanlığına bağlı kurumlar 2019-2020 eğitim-öğretim yılı ikinci dönemini uzaktan eğitim yöntemiyle tamamlayabilmiştir. 2020-2021

eđitim d6neminin ilk haftalarında okullar kademeli olarak eđitime aılmış b6ylelikle hem y6z y6ze hem uzaktan eđitim y6r6t6lm6şt6r. Ancak salgının seyrinin artması 6zerine t6m kademeler Kasım ayında tekrar uzaktan eđitim y6ntemi ile eđitime geiř yapmıřtır.

## 2.2 Yapılandırıcı Yaklařım

Yapılandırmacı 6đrenme felsefesinin ilk temellerini 18. y6zyılda Giambattista Vico'nun "insanların kendilerinin yaptığı řeyleri daha iyi anladığı" ile ilgili alıřması ile atmıřtır. Bu alanda birok bilim insanının alıřma yapmıř olmasına rađmen yapılandırmacılıđın ne olduđunu tanımlayan ve bu konuda fikirler 6reten ilk 6nemli bilim adamları Jean Peaget ve John Dewey olmuřtur (Arslan, 2007). Piaget'in yaptığı alıřmalar ile 6đrenmenin yalnızca bilgi aktarmak olmadığını, 6đrenenin bilgiyi yapılandırabileceđini ortaya koymuřtur (6zmen, 2004). Dewey ise yapılandırmacılıđı, insanların kendi deneyimlerinin yapılandırılması olarak aıklamaktadır (Aslan ve Aydın, 2016).

ađımızda 6đrencilerin "bilgiyi depolamaktan ziyade bilgiye nasıl ulařacakları" konusunda bir eđitim y6ntemi benimsenmeye bařlanmıřtır. Bu sebeple eđitim ortamında verimli bir 6đretim alanı oluřturmak amacıyla 6đrenmenin niteliđini artırarak nasıl daha verimli gerekleřebileceđini 6zerine odaklanmak gerekmektedir (Yılmaz, 2019). Bunun ıřıđında 6đrenmeyi etkili hale getirebilmek amacıyla birok 6đrenme modeli oluřturulmuř ve bu bađlamda y6ntemler geliřtirilmiřtir (Tarım ve Akdeniz, 2003). Geliřtirilen bu y6ntemle 6đrenci bilgiyi kendi zihninde yapılandırdığı iin, 6đrenme iřlevine aktif olarak katılmıř olacaktır. Yapılandırmacı 6đrenme modeli, bilgiyi bireye y6kleme yerine bireyin bilgiye ulařabilmesini, 6đrencinin bilgiyi alması yerine anlayabilmesini ve anlam oluřturabilmesini, kısacası 6đrenciye 6đretme yerine bilgiyi yapılandırabilmesini amalamaktadır (Duru, 2014). Yapılandırmacı y6ntemde bilgi mutlak deđildir ve kiřinin dođru bilgiye ulařması iin kendi zihinsel haritasını yapılandırması gerekmektedir (Uđurlu, 2009). Mevcut bilginin dođruluđundan ok kullanıřlılıđı daha 6nemlidir (Tan, 2009). Kiři daha 6nceki deneyim, bilgi ve becerilerine g6re belirlediđi d6ř6nce yapısını alır ve kendine g6re bazı deđiřikliklerle řekillendirerek, ilkelere dayalı bir řekilde oluřturacağı yeni yapıyı kendi 6znel durumlarına uyarlayarak benimser (Duman, 2007).

### 2.3 21. Yüzyıl Becerileri

Yakın zamansa hayatımıza giren yeni bir kavram olan 21. yüzyıl becerileri kavramıdır. 21. yüzyıl becerileri kapsamında gelecekteki dünya için çocuklarımızı hazırlarken aynı zamanda çocuklarımızın hangi beceriler ile donatılması gerektiği ifade edilmeye çalışılmaktadır (Roy-Singh, 1991). Günümüzde hızla gelişen teknoloji ile birlikte toplumda yaşanan bu gelişmelere ayak uydurabilecek, sürece aktif olarak katılabilecek ve tüm bu değişimleri yapılandırıp, değişimlere uyum sağlayabilecek kişilere ihtiyaç duyulmaktadır (Korucu ve Çakır, 2015). Günümüzdeki eğitim sisteminin temelini okuma, yazma, matematik ve fen dersleri oluşturmaktadır. Ancak mevcut eğitim programları, öğrencileri gerçek yaşama hazırlamada yetersiz kalmakta ve 21. yüzyıl iş dünyasının beklentilerini karşılayan çalışanlar geliştirilmesinde istenilen başarıyı sağlayamamaktadır (Uluyol ve Eryılmaz, 2015). Yaşadığımız yüzyılda başarılı olmak için, öğrencilerden farklı alanlarda bilgi kazanmalarına ek olarak bilim, teknoloji ve kültür alanlarında yeterlilik kazanmaları beklenmektedir (Pink ve Pink, 2005).

Yaşadığımız yüzyılda öğrenciler, zamanlarının büyük bir kısmını akıllı telefonlar, tabletler, bilgisayarlar vb. dijital araçlarla oyun oynayarak, müzik dinleyerek, mesajlaşarak, sosyal medyayı kullanarak, internette gezinerek ve buna benzer şeylerle geçirmektedir. Gününün hiç azımsanmayacak bir kısmını teknolojiyle iç içe geçiren öğrencilerin bilgiye ulaşma, bilgiyi kullanma ve yapılandırmada süreçleri de eskiye göre tamamen farklılık göstermektedir. Bundan dolayı da teknolojik cihazların içine doğan ve interneti, dijital oyunları çok iyi kullanabilen bu yeni nesil dijital yerliler olarak adlandırılmıştır. Dijital yerliler olarak adlandırılan bu nesil, teknolojileri günlük yaşantılarının hemen her alanında çeşitli amaçlarla kullanabilmekte ve bu teknolojilerden faydalanabilmektedirler (Bozkurt, 2014; Prensky, 2001b, 2009).

Okul dışında çok fazla teknolojiyle iç içe olan dijital yerlilerin, okulda teknolojiden uzak kalmaları pek mümkün olmamaktadır. Kısacası bilgiye kısa sürede ulaşabilen, bilgisayar oyunlarını oynamayı ev ödevlerini yapmaktan daha çekici bulan, görsel ve videoları izlemeyi düz metinleri okumak yerine tercih eden, vaktinin oldukça büyük bir kısmını internette geçiren bir neslin; ders kitapları, tebeşir ve tahtaların olduğu bir eğitim sistemine uyum sağlaması ve onların da bu öğrenme yöntemine motive olması ve katılması oldukça zordur (Prensky, 2001a). Öğretme yöntemlerini değiştirip, hem çağa hem de dijital yerlilere ayak uydurması dijital yerlileri eğitim

sürecine adapte edebilmek için gerekmektedir ve bunun da derslerine teknolojiyi entegre ederek sağlanacağı vurgulanmaktadır (Kurt ve diğerleri, 2013).

## 2.4 Eğitimde Teknoloji Entegrasyonu

Durmaksızın kendini güncel duruma getiren bilim ve teknolojiyle beraber bütün dünya devletleri kendi aralarında, sanayi, teknolojik gelişmeler ve ekonomik başarı bakımından önderlik mücadelesine girişmiştir. Bu mücadele sonucunda bilgiyi aktarabilen 21. yy mahareti ile donanmış kalifiye kişilere olan ihtiyaç fazlasıyla artmıştır. Bu nedenle bahse konu özelliklerde ehil, üreten kişileri yetiştirmek amacıyla her devlet teknoloji ve bilimde ortaya çıkan yenilik ve gelişmeleri eğitim sistemlerine dahil etme hususunda çeşitli ve farklı yaklaşımları kullanmaktadır (Akgün ve Türel, 2021). Teknoloji ve bilimdeki bu değişimler, günümüz insanların, teknolojinin etkili kullanımı, problem çözme, eleştirel düşünme, yaratıcılık, bilgi yönetimi, işbirliği ve iletişim gibi değişik maharetlere sahip olmasını zorunlu kılmıştır (Çavaş ve diğerleri, 2020).

Eğitimdeki köklü değişikliklerin odak noktasını teknoloji ve bilim alanındaki gelişmeler meydana getirdiği için köklü değişikliklerin temelini de fen eğitimindeki çalışmalar oluşturmaktadır. Çünkü devletlerin dünya üzerindeki gelişmişlik seviyesini tespit etmede fen eğitimi önemli bir etkidir (Boyunsuz, 2021). Fen bilimleri dersi öğrencilerin hayata hazırlanmasında ve temel yaşam bilimleri olarak ele alınmasından dolayı eğitim-öğretim programlarının ve uygulamalı öğretimin mecburi olduğu zamanlardan biridir. Öğrencilerin yaratıcılıklarını, problem çözme becerilerini, sosyal becerilerini geliştirmeyi amaç edinen ve Fen, Teknoloji, Mühendislik ve Matematiği (FeTeMM) dahil eden STEM yaklaşımı ön plana çıkmaktadır (Gülgün ve diğerleri, 2017). STEM nosyonunun özgün hali Science, Technology, Engineering ve Mathematics nosyonlarının kısaltmasından meydana gelmektedir. STEM kısaltması, orijinalinde science (fen), technology (teknoloji), engineering (mühendislik) ve mathematics (matematik) terimleri çevrelerinin baş harflerinin birlikte kullanılmasından oluşmaktadır. Ülkemizde FeTeMM (Fen, Teknoloji, Mühendislik, Matematik) olarak da isimlendirilmesi yapılmaktadır. Fen, teknoloji, mühendislik ve matematik çalışma çevrelerinin içerdiği beceri ve bilgilerin bütünleştirilmesiyle, öğrencilere iletişim kurma, takım çalışması yapma, araştırma yapma, yaratıcı düşünebilme, sorgulama, üretme ve

günlük hayat problemlerini çözebilme gibi becerilerin kazandırılmasını amaçlanmaktadır (Tezel ve Yaman, 2017).

Eğitimde teknoloji entegrasyonu öğretim programları, öğretmenlerin pedagojik yeterlikleri, yapılacak olan çalışmaların ekonomik olması ve kurumun hepsinin hazır olması gibi alanları içermektedir (Tinio, 2003). Günümüzde yapılan araştırmaların genelinde öğretmenlerin derslerinde sunular yapmak, öğrenciler için hazırladıkları ev ödevlerini elektronik posta ile göndermek, alanlarına uygun konuları anlatmak için çeşitli görseller hazırlamak ve kolay çizimler yapmalarıyla birlikte, okullarda teknoloji kullanımı hususunda bilgisayar tabanlı teknolojinin daha çok ön plana çıktığı belirtilmektedir (Bolat ve diğerleri, 2020). Öğretmenler, öğrenmenin merkezinde öğrencinin olduğu yapılandırıcı yaklaşım konusunu özümlemeli, ayrıca okul müdürleri, öğretmenlerin bu yaklaşımı derslerine nasıl dahil edebilecekleri hakkında çeşitli çalışmalar yapmalarına destek olmalıdır (Samancıoğlu ve Summak, 2014). İnternet ortamında hazırlanan dersler, öğrencilerin öğrenmeye olan ilgi ve meraklarını artırıp öğrenme sırasında daha aktif hale gelmelerini sağlamaktadır (Koehler ve diğerleri, 2007). Var olan çalışmalar sonucunda teknoloji kullanımı öğretmenlerin sınıf içi öğretme yöntemlerini planlama ve uygulamada sık sık değişiklik yapmasını ve öğrencinin merkeze alındığı uygulamaları artırmasını sağlamıştır (Windschitl ve Sahl, 2002). Öğretmenlerin teknolojiyi derslerine entegre edebilmeleri için hayatlarının her alanında teknolojik araçları kullanmaları teşvik edilmeli ve kendi bilgisayarlarının olması veya İnternete bağlanabilecekleri bir cihaz (tablet, akıllı telefon, vb.) edinmeleri desteklenmelidir (Samancıoğlu ve Summak, 2014). Öğretmenlerin bilgisayar kullanımını yükseltmek ayrıca teknolojiyi derslerine entegre edebilme seviyelerini artırmak uzun süre alacak bir süreç olup donanım, yazılım, İnternet ve teknik alanlarında eğitilmeli; bilişim teknolojileri kullanımı hususunda gelişimleri sağlanmalıdır (Wachira ve Keengwe, 2011). Mesleğe henüz başlamış öğretmenlerin teknolojiyi kullanma yetkinliklerinin, tecrübeli öğretmenlerin teknolojiyi kullanma yetkinliklerine göre daha olumlu durumda olduğu belirtilmektedir (Ertmer ve Ottenbreit-Leftwich, 2010). Bu bağlamda sadece öğretmen yetiştirme sürecinde değil, hizmet içi eğitimlerin de göz önüne alınması gerektiği söylenebilmektedir. Öğretmenlerin günlük yaşamda bilgisayar kullanmaları ile derslerinde teknolojiyi kullanmaları arasında pozitif yönlü bir ilişki olduğu düşünülmektedir. Eğitim fakültelerinde yeteri kadar bilgisayar dersi almış, teknoloji ile ilgili hizmet içi eğitim

faaliyetlerine katılmış ayrıca kişisel gelişim çalışmalarına katılmış eğitimcilerin kendilerini derslerinde, teknoloji kullanımı konusunda daha iyi hissettiği görülmektedir (Bolat ve diğerleri, 2020). Öğretmenlerin hizmet içi eğitim için ihtiyaç duydukları konu başlıkları arasında “öğretim materyalini etkin kullanma”, “eğitimde teknoloji kullanımı” ve “İnternetin eğitim amaçlı kullanımı” olduğu görülmüştür (Saritepeci ve diğerleri, 2016). Eğitim öğretim süreçlerine bilişim teknolojilerinin etkili entegrasyonu, sadece öğretim programlarına bu teknolojilerin entegrasyonu ile oluşan bir durum değildir (Tezci, 2016). Uygulama esnasında tasarlanan öğretim programına göre sınıfta bilişim teknolojilerinin uygun şekilde kullanılması yetersiz kalıp, bu öğretimin sanal ortamlara da dahil edilmesi gerekir (Öztürk ve Tetik, 2015). Bilişim teknolojilerinin eğitimin her basamağında uygulanmasını yapacak kişiler programın uygulayıcısı olan öğretmenlerdir (Nezih ve Çakır, 2016). Bu sebeple öğretmenlerin bilişim teknolojileri kullanım seviyeleri ve bu teknolojik araçları hangi gaye ile kullandıkları ele alınması gereken önemli bir konudur (Tezci, 2016). Teknoloji entegrasyonunda öğretmenlerin teknolojiyi uygun bir şekilde kullanmaları, teknolojik araçları daha net tanımaları, özümsemeleri hangi derste hangi uygulamaları ne şekilde gerçekleştireceklerini bilmelerine bağlıdır (Bolat ve ark., 2020). Ayrıca öğretmenlerin bu teknolojileri derslerinde uyguladıktan sonra geri bildirim alıp bu uygulamaların öğrencilere ne kadar katkı yaptığını da tespit etmeleri fazlasıyla önemlidir (Bolat ve diğerleri, 2020).

## **2.5 Kodlama**

Kodlama terimi yakın zamanda adından sıklıkla bahsedilen bir konu olarak hayatımızın içerisinde yer almaya başlamıştır. Hızla gelişen teknolojiyle birlikte kodlama konusu ülkelerin öğretim programlarına entegre edilmekte ve bu amaç dahilinde çeşitli etkinlikler de yapılmaktadır. Yapılan bu etkinlikler sonucunda çocuklarda mantıksal problem çözme, akıl yürütme, iş birliği, iletişim, bilgi ve teknoloji okuryazarlığı gibi becerilerin geliştirilmesi amaçlanmaktadır (Sayın ve Seferoğlu, 2016).

Kodlama, bir işlemin bilgisayarlar aracılığıyla yerine getirilmesi için izlemesi gereken yolun adım adım yazılmasının komutlarla anlatılmasıdır (EuropeanSchoolnet). Kısacası bilgisayarın anlayacağı dili kullanarak mevcut problemin çözümünü üretme işidir (Van-Roy ve Haridi, 2004; Yiğit, 2016). Yani kodlama; adım adım verilen yönergeleri izlemeleri ve onların tam olarak ne yapmaları gerektiğinin adımlar halinde

bilgisayarlar aracılığı ile açıkça söylenmesidir (Eba, 2020). Kodlama eğitimi aracılığıyla öğrencilere henüz çok küçük yaşlarda araştırma, bilgi işlemsel düşünme, sorun çözme, buluş yapma, sorgulama, üretme, projeler geliştirme gibi beceriler kazandıracaktır. Bu sayede ilerleyen yaşlarda bireylerin yazılım işinde olmasalar bile, hayatlarının tüm alanlarında analitik düşünecek, sorgulayacak, araştırarak ve üretecektir (BtDersleri, 2020).

Kodlama eğitimi ve bilgisayar eğitimi giderek ilerleme kaydetmektedir. ilkokuldan itibaren ders müfredatlarında yer almaya başlayan programlama eğitimi çocuklara bilgisayar alanının yanında yaratıcılık ve analitik düşünme yeteneği kazandırırken aynı zamanda problem çözme becerilerini de geliştirmektedir (Baz, 2018). Günümüzde, gelecekteki zamanın zorunlulukları yörüngesinde yeni işler yapabilmek için kodlama eğitimi zorunluluk olmuştur (Sayın ve Seferoğlu, 2016). Kısa süre önce yapılan araştırmalar bize şunları göstermektedir; kodlama öğrenen kişilerin karşılarına çıkan problemlere karşı hızlı ve kolay çözüm üretme becerilerini geliştirdiklerini belirtmektedir. Kodlama yeteneği aracılığıyla çocukların yaptıkları yanlışları daha çabuk çözebildikleri ve ortaya çıkan sonuçları analiz etme yeteneklerinin de arttığı gözlenmektedir (Coravu ve diğerleri, 2015; Resnick ve Silverman, 2005). Yeni kodlama öğrenmeye başlayan öğrenciler kodlama eğitimi derslerinde oldukça zorlanmakta ve kodlamayı karmaşık yapıları olarak nitelendirmektedirler (Erol, 2015). Bu nedenlerden dolayı kodlama eğitimi küçük yaştaki çocuklara yönelik daha anlaşılır, kolay ve zevkli hale getirilmesi amacıyla birçok araç/ortam bulunmaktadır. Bu araçlar şunlardır; bilgisayarsız (unplugged) kodlama etkinlikleri, metin tabanlı kodlama araçları ve blok tabanlı kodlama araçları kullanılarak öğrencilerin yazdıkları kodların sonucunu fiziksel ortamda görmelerine olanak sağlayan robotik kodlama araçları şeklinde sınıflandırılabilir (Koçin, 2020).

Bilgisayarın anlamlandırabileceği makine diline komutların kurallı ve anlamlı bir sıralamaya sokmak zihinsel açıdan oldukça karmaşık olsa dahi kodlama öğrenen bireyler cephesinde olumlu bilişsel katkılar sağlamaktadır (Sırakaya, 2018). Kodlama eğitimi sayesinde kişiler, 21. Yüzyıla gerekliliklerine kendilerini hazırlanmakta, 21.yüzyıl için gerekli olan becerileri bir arada elde etmekte, günümüzün ihtiyaçlarını karşılayabilecek seviyede olmakta, çağın gerekliliği haline gelen yeterlilikleri edinmekte ve tüm bunlara ek olarak ülkenin gelişmişlik seviyesini, ülke ekonomisini ve

ülke refahına olumlu yönde katkıları olmaktadır. Günümüzde ülkelerin rekabet güçlerini etkileyen en önemli bileşen yazılım sektöründeki güçleri oranlarıdır ve tüm sektörlerde yazılımın gücünü hissetmekteyiz. Tüm bunların sonucu olarak da dünya genelinde yazılımın önemli yeterince anlaşılmiş ve programlama eğitimi okulöncesi döneme kadar indirilmiştir (Baz, 2018) .

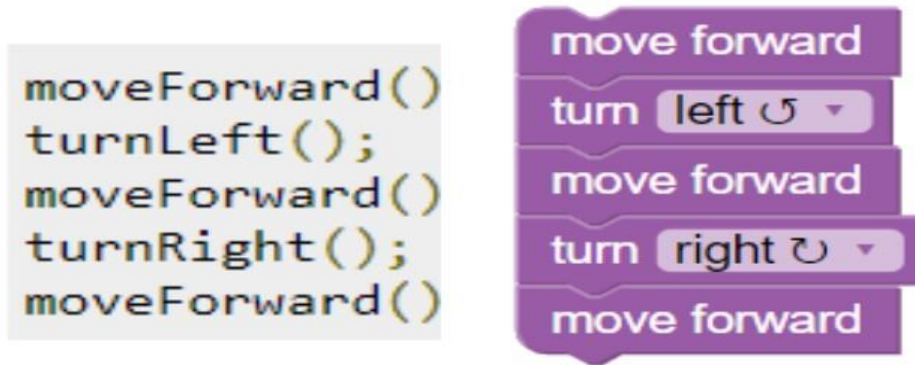
Kodlama veya programlama olarak adlandırılan dersler tüm dünyada olduğu gibi Türkiye’de de özellikle son dönemlerde daha çok gündeme gelmeye başlamıştır (Kodlamanisa, 2020). Programlama dersleri neticesinde öğrencilerin sadece kod yazmaları değil; eleştirel düşünme, işbirliği, yenilikçi düşünme, iletişim, problem çözme, bilgi okuryazarlığı gibi 21.yüzyıl becerilerinin gelişmesi beklenmektedir. Zamanın şartlarında insanların büyük bir kısmı teknoloji kullanımı konusunda yeterli bilgi ve deneyime sahip olmasına rağmen yeni bir şeyleri nasıl üretileceğini, hangi adımları takip etmesi gerektiğini bilmemektedir. Yeni ürünleri teknolojiyi kullanarak elde etmek bugün dünyadaki en önemli becerilerden birisi olarak görülmektedir (Microsoft, 2020). Matematiksel düşünme becerisi düşük olarak belirtilen öğrencilerin kodlama sayesinde karmaşık bilişimsel düşünme yetenekleri geliştirmekte ve karmaşık matematiksel düşünceleri kullanmalarına yardımcı olmaktadır (Taylor ve diğerleri, 2010).

Tüm dünyada kodlama öğretimi konusunda yaşanan gelişmeler doğrultusunda Türkiye’de de kodlama eğitimine verilen önem gün geçtikçe artmaktadır. Bu amaç dahilinde bilgisayar dersleri 2012 yılına kadar “Bilgisayar”, “Bilişim Teknolojileri” gibi çeşitli adlandırmaların sonucunda, 2012 yılında alınan bir karar ile “Bilişim Teknolojileri ve Yazılım” olarak revize edilmiştir. 2012 yılına kadar dersin içeriğinde bilişim teknolojileri temelleri ve temel office programlarına ait kazanımlar yer almaktaydı. Sonradan yapılan değişikliklerle dersin adında yazılım ifadesi ilk defa literatürde yer almış olup, müfredata algoritma ve programlama ile ilgili konular sa eklenmiştir. Bu sayede 5. Sınıf seviyesinde öğrencilere temel kodlama eğitimleri verilmeye başlanmıştır (TTKB, 2012a). Bilgisayar derslerinde yapılan bu güncellemeler sonucunda 5. ve 6. sınıflarda haftada iki ders saati zorunlu olmuş, 7 ve 8. Sınıf seviyelerinde ise seçmeli ders olarak okutulmaya başlanmıştır. Bilgisayar dersinin müfredatında programlama ve kodlama eğitimi içeriği genişletilmiştir. Bu çerçevede Problem Çözme, Programlama ve Özgün Ürün Geliştirme üniteleri müfredata

eklenmiştir (TTKB, 2012a). Ortaokul seviyesinde Milli Eğitim Bakanlığı tarafından yapılan bu değişiklikler, ortaöğretim seviyesinde de çeşitli değişikliklerin yapılmasının önünü açmıştır. Türkiye’de mevcut sistemde ortaöğretim meslek liselerinin bilişim teknolojileri alanında programlama eğitimi verilmekteydi. Yakın zamanda kodlamanın önem kazanmasıyla birlikte MEB, 2016 yılında yayınladığı öğretim programıyla Bilgisayar Bilimleri dersini Kur-1 ve Kur-2 olarak belirlemiş ve diğer liselere de bu dersi uygulamaya koymuştur. Kur-1’in içeriği incelendiğinde “Problem Çözme ve Algoritmalar” ve “Programlama” ünitesi kodlama kısmını oluşturmaktadır. Kur-2 incelendiğinde ise “Robot Programlama, Web Tabanlı Programlama ve Mobil Programlama” olarak tamamen kodlama üzerine kurulmuştur (TTKB, 2017). “Anadolu İmam Hatip Liselerinde, Anadolu Liselerinde, Güzel Sanatlar ve Spor Liselerinde” Bilgisayar Bilimleri dersi isimi ile seçmeli olarak haftada iki saat, hazırlık sınıfı bulunan Anadolu Liselerinde ve Sosyal Bilimler Liselerinde zorunlu olarak haftada dört saat, Fen Liselerinde zorunlu olarak haftada iki saat okutulması kararı alınmıştır. Aynı zamanda Türkiye’de 2011 yılında uygulamaya konulan FATİH projesi kapsamında 2017 yılında EBA içeriğine kodlama modülü eklenmiştir.

## 2.6 Blok Tabanlı Kodlama

Soyut yapıda olan geleneksel programlama dillerinin öğrenmesi zor ve karmaşık halde olması blok tabanlı kodlama araçlarına eğilimi arttırmıştır.



Şekil 2.2 Aynı Komutu Uygulayan Metin Tabanlı Kod ile Blok Tabanlı Kod Örneği (Blocklygames, 2020)

Öğrencilerin işlem basamaklarına ayırma, sıralama yapma, algoritma oluşturma, kendi oyununu ve uygulamasını yapma işlemlerini blok kodlama araçları ile daha kolay

daha basit bir şekilde yapması sağlanmaktadır (Sırakaya, 2018). Blok tabanlı kodlama araçlarında karmaşık ve uzun kodlardan ziyade sürükle bırak mantığına dayalı kod blokları kullanılmaktadır. Bu sayede herhangi bir program yazarken kod bloklarını algoritmaya uygun olarak yerleştirmek yeterli olmaktadır. Kod bloklar genellikle farklı renklerde tasarlanmıştır. Kullanıcıların renkli bloklar sayesinde bir programı oluştururken hata yapma olasılığı azalmaktadır. Blok temelli kodlama araçları, kullanımı basit bir ara yüzüne sahiptir. Bu kodlama araçları, her yaştan öğrencilerin kullanabileceği basitlikte tasarlandığından kodlama eğitiminde öğrenciler istediği bir projeyi tasarlama noktasında herhangi bir sıkıntı yaşamamaktadır. Blok tabanlı kodlama araçlarında, metin tabanlı kod yazma problemi ortadan kalktığı için kod yazma hatalarını ve kod ezberleme problemini ortadan kaldırmıştır. Aynı zamanda mantık hataları kolaylıkla görülebilmekte ve program çalışırken bloklar değiştirilerek müdahale edilebildiğinden hata ayıklama işlemi oldukça basittir. Hazırlanan uygulamalara ve projelere resim, ses, video gibi birçok çoklu ortam ögesi blok tabanlı kodlama araçlarında eklenebildiği için kullanıcıların farklı uygulamalar geliştirmelerini desteklemektedir. Bu sayede öğrenciler, programlama öğrenirken sadece kendi ilgi duyduğu uygulama, animasyon, dijital hikaye, oyun gibi bir yazılımla sınırlı kalmamakta; elektronik bileşenleri kullanarak robotlar gibi fiziksel araçları da programlayabilmektedir. Tüm bunlara ek olarak blok tabanlı kodlama araçlarında hazırlanan projeler kullanıcılar tarafından çevrimiçi ortamında paylaşılabilir. Web ortamda kayıtlı kullanıcılar tarafından görüşler alınabilmekte ve yaşanan sorunlara karşı çözüm önerileri için destek bulunabilmektedir (Gülbahar, 2020).

Scratch programının ilköğretim öğrencilerinin yaratıcılık düzeyleri üzerinde etkisinin araştırıldığı bir çalışmada scratch programının öğrencilerin yaratıcılık becerisi, akıcı düşünme ve girişkenlik üzerinde olumlu bir etki yarattığı belirlenmiştir (Kobsiripat, 2015). Scratch, blok tabanlı programlama aşağıda detaylı olarak incelenmiştir. Bunlar dışında birçok blok tabanlı kodlamayı öğreten platformlar da mevcuttur. Kodlamaya yeni başlayanlar daha çok olmak üzere kodlama becerilerini geliştirmek isteyenler içinde çok farklı uygulamalarda bulunmaktadır. Bunlar Alice, Blockly Games, Kodu Game Lab, Daisy Dinosaur, Code.org, Bitsbox, ve Code Monkey, Code Combat, Lightbot, Grok Learning, Kidsruby, Bomberbot, Touch Develop ve Tech rocket gibi birkaç farklı blok tabanlı programlama platformları da vardır (Ünsal, 2019).

### 2.6.1 Scratch

Scratch programı, MIT Üniversitesi Medya Laboratuvarı ve Yasmin Kafai'nin UCLA grubu işbirliğiyle hazırlanmış olan görsel bir programlama ortamıdır. Scratch programı kullanıcılara kendi hayal güçlerini kullanarak, koordinat, açı gibi çeşitli geometrik kavramlara ek olarak koşullu ifadeler, diziler, döngüler gibi programlamada temel olabilecek kavramları da içinde barındıran açık kaynak kodlu bir programdır. Program kullanıcıların farklı kod bloklarını birleştirerek anlamlı tasarımlar yaptıkları, yaptıkları bu tasarımları anında uygulama, kontrol etme imkânı buldukları bir ortamdır (Calder, 2010). MIT tarafından 2007 yılında geliştirilmiş küçük yaşta çocuklara özellikle ilköğretim ve ortaokul öğrencileri ve programlamayla ilk kez tanışan kişilere kodlama öğretmek amacıyla tasarlanmış ücretsiz blok tabanlı kodlama aracıdır. 150'den fazla ülkede ve 40'tan fazla dilde kullanılmakta olan scratch programı, 8 ile 16 yaş grubu için tasarlanmıştır (ScratchAbout, 2020). Program kullanıcıları; oyun, interaktif hikaye, simülasyon, animasyon gibi çalışmalar oluşturabilmekte ve oluşturdukları projeleri bireysel internet sitesinde yayınlama olanağına sahip olmaktadır (Brennan ve Resnick, 2013). Aynı zamanda scratch programında hazırlanan projeler scratch programının web sitesi olan "scratch.mit.edu" adresinde de paylaşılabilir. Dünyanın farklı bölgelerinde bulunan ve "scratch.mit.edu" sitesine üye olan bireylerin kendi projelerini paylaşma, başkalarının projelerini görme, aynı zamanda beğendikleri ve merak ettikleri projeleri bilgisayarlarına indirerek projenin nasıl yapıldığına ilişkin kodları görebilme imkanı sunmaktadır (Adams, 2010).

Scratch tarzında sürükle bırak mantığına dayalı programlar algoritmik düşünmenin gelişmesini sağlar ve bilgisayarda programlamanın temelini oluşturur. Bununla birlikte öğrenenler bilgisayarlı düşünme becerisine sahip olurlar (Brennan ve Resnick, 2013). Ayrıca problem çözme ve yaratıcı düşünme becerilerini de sağlar (Calder, 2010). Tüm bunlara ek olarak da bireylerin yönetim ve kontrol becerilerinin gelişimi için Scratch'ın en iyi program olduğu düşünülmektedir (Ferrer-Mico ve diğerleri, 2012).

Scratch fazla programlama bilgisine sahip olmadan yaratıcı, etkileşimli projeler geliştirmek ve projeleri sosyal paylaşım sitesinde paylaşma imkânı sağlayacak şekilde tasarlanmıştır (Meerbaum-Salant ve diğerleri, 2011). Scratch ile programlama lego parçalarını birleştirmek veya yapboz yapmak gibidir (Malan ve Leitner, 2007; Wilson

ve Moffat, 2010). Canlılık, tek pencereleli kullanıcı ara yüzü, hata mesajı olmaması, komut setinin en aza indirilmesi ve kod bloklarını canlandırma scratch programının özellikleridir (Maloney ve diğerleri, 2010).

Programlama; problem çözmeyi, sistematik düşünmeyi ve stratejileri tasarlamayı desteklemektedir ve bireye düşüncesini yansıtmaya imkânı sağlamaktadır. Scratch programının üç temel ilkesi ise şunlardır; daha fazla düşünebilme, daha fazla anlam katma ve diğer programlama dillerine göre daha fazla sosyallik sağlamaktır. Daha fazla anlam katma ilkesi ile çeşitlilik ve bireyselleştirme özellikleri üzerinde durulmaktadır. Daha fazla sosyallik ilkesi ile de bireylerin birbirlerini desteklemesi, işbirliği içinde çalışmaları; birbirlerinin yaptıkları projelerini eleştirmeleri ve tamamlamaları üzerinde durulmaktadır (Resnick ve diğerleri, 2009).

Scratch programı rahat ve görsel formatı ile programlamanın önündeki engelleri ortadan kaldırmış ve bireylere kolayca oyunlar ve animasyonlar yapma olanağı sağlamıştır. Scratch sahnesini gerçek hayattaki tiyatro sahnesine benzetebiliriz. Tıpkı tiyatrodaki olduğu gibi scratch de becerikli öğrencilere, hayal güçlerini sahneye yansıtmaya imkânı sağlamaktadır. Bunun yanında yazılım, öğrencilere koordinat sistemi gibi matematiksel kavramları öğretir (Lee, 2011). Öğrencilere işbirliği içinde çalışmayı da öğreten scratch sadece teknik becerileri öğretmekle kalmamaktadır.

Scratch ile öğrenciler problem çözme becerilerini geliştirebilir, yaratıcı etkinlikler tasarlayabilir ve problemlere birden çok çözüm yolu bulabilirler, scratch projelerini yeniden tasarlayabilir, çalışarak deneyimlerini arttırabilir ve işbirlikçi öğrenme aktiviteleri gerçekleştirebilirler (Kordaki, 2012; Resnick ve diğerleri, 2009).

## **2.7 Unplugged (Bilgisayarsız) Kodlama**

Yaşadığımız dünyada da kullandığımız üst düzey düşünme becerileri bir programlama dilini bilmek, tasarlama, planlama, yazma ve sınamayı bilmeyi gerektir (Robins ve diğerleri, 2003). Bilgisayarların olmadığı ortamlarda, üst düzey düşünme becerilerini harekete geçirebilecek etkinliklere dayalı ders planlarının üretilmesi, kodlama öğretiminde faydalı sonuçlar verebilir. Ayrıca teknolojik imkânların kısıtlı olduğu eğitim ortamlarında verilecek kodlama eğitiminin kazanımlarına erişmede faydalı olabilir.

Fatih Projesi ile birlikte özellikle taşradaki eğitim kurumlarının teknolojik alt yapılarında büyük oranda iyileştirilme gerçekleştirilse de kodlama eğitimin verilmesi için uygun teknolojik şartların tam olarak sağlanmadığı eğitim kurumları da halen bulunabilmektedir. Bilişim teknolojileri sınıfında yeterli sayıda bilgisayarın bulunmaması veya güncel yazılımları destekleyemeyecek derecede eski olması bilgisayarlı kodlama eğitiminin verilememesine neden olabilmektedir. Bu tarz durumlarda Bilişim Teknolojileri ve Yazılım dersi kapsamında kodlama eğitiminin verilmesi Bilgisayarsız Kodlama Eğitimi ders planına göre yapılabilir (Sade, 2020).

Bazı kart oyunları ve fiziksel etkinlikler aracılığı ile bireylere kodlama eğitimi vermek için kullanılan teknikler Bilgisayarsız (unplugged) kodlamanın bütünüdür. Yaşamımızın tüm alanında yer alan yazılımın temel seviyede okul öncesi ve ilköğretimde verilebilmesi için birçok yöntem kullanılabilir. Bunlardan birisi de bilgisayarın öğrenme ortamında olmadan yapıldığı kodlama etkinlikleridir. Bilgisayarsız ortamda verilen kodlama eğitiminin temel mantığında ise bir senaryo üzerinden öğrencilerin problem çözmesini sağlamak yatmaktadır. Bilgisayarsız ortamda kodlama özellikle erken yaşlardaki çocukların ihtiyacı olan işbirlikli öğrenme, fiziksel aktivite, problemi çözerken farklı çözüm yollarını da düşünme becerisini geliştirmesi, iletişim becerilerinin geliştirilmesi ve bir problemin olasılıklı yollarını keşfetmesinin yanında kodlamanın dışında birçok kazanıma da katkı sağlamaktadır (Kodlamanisa, 2020).

### **2.7.1 Unplugged (Bilgisayarsız) Kodlama Örnekleri**

Tospaa.org: Bilişim sınıfının ya da bilgisayarın bulunmadığı eğitim kurumlarında kodlama eğitimi verebilmek için geliştirilmiş bir masaüstü kart oyunudur Tospaa. Oyunu oynamak için ise gerekli olan belgelerin yazıcıdan renkli çıktısını almak yeterli olacaktır. Tospaa oyunun amacı tospaayı engellere takılmadan hedefine ulaştırmaktır (Tospaa, 2020).

csunplugged.org: CS Unplugged, renkli boya kalemleri, kartlar, ip ve birçok eğlenceli araç gereç ile kullanılan bulmacalarla ve ilgi çekici oyunlar sayesinde kodlama dersi veren ücretsiz öğrenme etkinlikleri koleksiyonudur (Computer, 2020).

kesfetprojesi.org: 2014 yılından itibaren Google aracılığıyla Milli Eğitim Bakanlığı iş birliği ile Keşf@ Projesi (www.kesfetprojesi.org) hayata geçirilmiş ve

uygulanmaya başlanmıştır. Bu proje dahilinde, 5.sınıf seviyesinde eğitim gören öğrencilerine yönelik olarak bilinçli ve güvenli internetin nasıl kullanılacağı konusunda eğitim, sınıf içi organizasyonların, videoların ve oyun kartlarının da dahil olduğu farklı yöntemlerle, öğretmenlerin, velilerin ve çocukların kullanımına sunulmuştur (Kodlamanisa, 2020).

### ***2.7.1.1 Tospaa***

Son zamanlarda küçük yaşlarda kodlama eğitimi tüm dünyada olduğu gibi ülkemizde de eğitimin temel bileşenlerinden biri haline almaya başlamıştır. Yapılan farklı eleştirilere rağmen kodlama eğitimi akımının önümüzdeki on yılı domine edeceği düşünülmektedir. Öyle ki kodlama eğitimini daha ilgi çekici hale getirmek adına çeşitli uygulama ve mobil ortamlarda birlikte elektronik cihazlar da birbiri ardına piyasaya sunulmaya başlandı. Fakat farklı bir yönden ele alındığında çocuklarımızın henüz çok küçük ve oyun oynayacağı çağlarda ellerine tabletler tutuşturmak yahut ekranın başına saatlerce oturtmak yararlı bir etkinlik için dahi de olsa aileleri oldukça endişelendirmektedir. Kodlama eğitim öğretim müfredatımıza girmiş olsa da ülkemiz de bulunan eğitim kurumlarının koşulları nedeniyle kodlama eğitiminin ihtiyaç duyduğu araç ve gereçlere sahip okul sayısı sınırlıdır. Bu nedenden ötürü de bilgisayarsız kodlama (unplugged) hem öğrencilerimize kazandırmamız gereken kazanımları onlara sunabileceğimiz aynı zamanda da teknolojiden bağımsız olunan bir öğrenme ortamında eğitim imkanı oluşturuyor. Bilgisayarsız (unplugged) kodlama farklı kart oyunları ve fiziksel etkinlikler aracılığıyla öğrencilere kodlama eğitimi verme için kullanılan tekniklerdir. Bir örnek ile açıklayacak olursak; tospaa oyunu gibi bilgisayara ihtiyaç duyulmayan faaliyetler unplugged kodlama olarak ifade edilmektedir. Bilgisayarsız kodlama özellikle de okul öncesi ve ilkokul çağında çocuklarımızın ihtiyacı olan işbirlikli öğrenme, fiziksel aktivite ve iletişim becerilerinin geliştirilmesi gibi kodlamanın dışında da pek çok kazanıma karşılık gelmektedir (Tospaa, 2020).

## BÖLÜM 3

### 3 YÖNTEM

Bu bölümde araştırma yöntemi ele alınmaktadır. Araştırmanın modeli, araştırmanın çalışma grubu, veri toplama araçları, araştırmada kullanılan ölçek ve veri çözümlene teknikleriyle ilgili ayrıntılı bilgiler bu bölümde yer alacaktır.

1. Bilgisayar kullanılmadan öğrencilere aktarılacak kodlama eğitimi öğrencilerde ön test ve son test sonrasında anlamlı bir farklılık sağlayacak mıdır?
2. Bilgisayar Destekli Eğitim ortamında öğrencilere aktarılacak olan kodlama eğitimi öğrencilerde ön test ve son test sonrasında anlamlı bir farklılık sağlayacak mıdır?
3. Öğrenciler bilgisayarlar kullanımına karşı hazır bulunurlukları ne seviyede?
4. Öğrencilerin Kodlama hakkında ön öğrenmeleri nelerdir?
5. Öğrencilerinin akademik başarı testinden aldıkları puanlar da dersin işleme stratejisi etkili olmuş mudur?

#### 3.1 Araştırmanın Modeli

Farklı görsel programlarla tasarlanan kodlama eğitiminin ortaokul öğrencilerinin akademik başarı ile kodlamaya karşı tutum ve öz-yeterliklerine etkisi inceleyen bu araştırmada nicel araştırma yöntemlerinden yararlanılmıştır. Bu çalışmada araştırma modeli olarak nicel araştırma yöntemi benimsenmiş olup “Ön test – Son test Kontrol Gruplu Yarı Deneysel Desen Modeli” kullanılmıştır.

Bu çalışma da ön test ve son test deney ve kontrol gruplu eşitlenmemiş yarı deneysel desen tercih edilmiştir (Büyüköztürk ve diğerleri, 2012). Gruplardan yansız atama ile deney ve kontrol grubu seçilir. Rastgele seçim yapmanın mümkün olmadığı zamanlarda başvurulan bir yöntemdir. Bu yöntem; deneklerin seçilmesinde rastgele dağılımın uygun olmadığı bir deney yaklaşımını içeren tasarımıdır. Önceden oluşturulmuş sınıfların kullanılmasıyla gerçekleştirilen bir yöntemdir.

Çalışmada öğrenciler deney ve kontrol grubu olmak üzere iki gruba ayrılmaktadır (Karasar, 1999). Kontrol grubunda dersler Tospaa bilgisayarsız kodlama programı kullanılarak işlenmiştir. Uygulamalar öğretim programında ders planı süresine uygun olarak 10 hafta boyunca yapılmıştır. Kontrol grubunda dersler bu şekilde

işlenirken; deney grubunda Scratch yazılımı desteğiyle dersler işlenmiştir. Her iki grupta da aynı şekilde deney öncesinde ve deney sonrasında ölçümler yapılmaktadır (Büyüköztürk ve diğerleri, 2012). Deney ve Kontrol gruplarında araştırmaya başlarken ve araştırma bitiminde öğrencilere kodlamaya yönelik akademik başarı testi, Bilgisayar Destekli Eğitime İlişkin Tutum Ölçeği, Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algı Ölçeği, Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeği yapılmıştır. Verilerin mümkün olduğunca ayrıntılı ve direkt olarak çalışma grubunun ifadeleri ile analiz edilmesi önem arz etmektedir, nicel araştırma ile de öne atılan teori için elde edilen veriler baz alınarak öğrenme çıktılarında etkili olup olmadığı incelenmiştir (Creswell ve Clark, 2017; Miles ve diğerleri, 1994; Strauss ve Corbin, 1998).

**Tablo 3. 1** Araştırmaya yönelik yarı deneysel desen tablosu

Grup	Ön Test	İşlem	Son Test
D	O <sub>1</sub>	X <sub>te</sub>	O <sub>2</sub>
K	O <sub>1</sub>	X <sub>se</sub>	O <sub>2</sub>

D = Deney grubu

K = Kontrol grubu

X<sub>te</sub> = Tospaa ile kodlama eğitimi

X<sub>se</sub> = Scratch ile kodlama eğitimi

O<sub>1</sub> = Deney ve kontrol grubu ön test uygulaması

O<sub>2</sub> = Deney ve kontrol grubu son test uygulaması

### 3.2 Araştırmanın Çalışma Grubu

Araştırmanın çalışma grubunu 2020–2021 eğitim ve öğretim yılında Konya ili Meram ilçesi Meram Gödene TOKİ İmam Hatip ortaokulda eğitim görmekte olan ilköğretim 6.sınıf öğrencileri oluşturmaktadır. Araştırma 10 hafta süren uzaktan eğitim ile uygulanmıştır. Çalışmada bağımsız değişkenlerin, bağımlı değişkenleri nasıl etkilediğini sınamak amacıyla bir deney ve bir kontrol grubu oluşturulmuştur. Deney grubunda N=21 öğrenci, kontrol grubunda N=24 öğrenci olmak üzere toplamda N= 45 öğrenci üzerinde çalışma gerçekleştirilmiştir.

**Tablo 3. 2** Çalışma grubu demografik verileri

		<b>N</b>	<b>%</b>
<b>Cinsiyet</b>	Kadın	17	37.7
	Erkek	28	62.3
<b>Deney grubu</b>	Kadın	7	29.1
	Erkek	14	66.9
<b>Kontrol grubu</b>	Kadın	10	41.6
	Erkek	14	58.4
<b>Toplam</b>		<b>45</b>	<b>100</b>

Tablo 3.2’den de görüldüğü gibi araştırmanın çalışma grubunda ki öğrencilerin % 37.7’sini kadın öğrenciler, % 62.3’ünü ise erkek öğrenciler oluşturmaktadır. Deney grubunu % 29.1’ini kadın öğrenciler, % 66.9’unu ise erkek öğrenci; Kontrol grubunu % 41.6’sını kadın öğrenciler, % 58.4’ünü ise erkek öğrenciler oluşturmaktadır.

### **3.3 Veri Toplama Araç ve/veya Teknikleri**

Çalışmamız kapsamında kullandığımız veri toplama araçları şu şekildedir:

#### **3.3.1.Akademik Başarı Ölçeği**

İki gruba da ön test ve son test olarak uygulanan öğrencilerin kodlamaya yönelik başarılarını ölçmek amacıyla Milli Eğitim Bakanlığı 2020/2021 yılı 6.sınıf Bilişim teknolojileri ve yazılım öğretim programında bulunan problem çözme ve programlama kazanımını kapsayan başarı testi kullanılmıştır.

Çalışmaya katılan öğrencilerin akademik başarıları ders alanında uzmanı öğretmenler tarafından Scratch Akademik Başarı Testi (EK 4) ve Tospaa Akademik Başarı (EK 5) testi olarak hazırlanmıştır. Scratch akademik başarı testinde 14 adet soru bulunmaktadır, Tospaa akademik başarı testinde ise 5 adet soru bulunmaktadır.

Çoktan seçmeli bir testin (akademik başarı testi) güvenilirlik katsayısının belirlenmesi gerekmektedir. Bunun için genel olarak KR-20 ve KR-21 formülleri kullanılır (Güven, 1990). Bu yöntem kullanılırken, uygulama sürecinde ölçme

aracındaki veri setleri oluşturulmak için maddelerden alınan cevaplarda beklenen özellik varsa “1”, beklenen özellik yoksa “0” puan verilerek oluşturulur. Testlerin iç güvenilirlik katsayısının tespiti için belirlenmiş özellikler dikkate alınır ve Kr-20 veya Kr-21 formüllerinden hangisi daha uygun ise o formül kullanılır (Ercan ve Kan, 2004) .

Kr-20 formülü (Ergin, 1995);

n : testteki madde sayısı

q : bir maddeyi doğru cevaplamayanların oranı = (1-p)

P : bir maddeyi doğru cevaplayanların oranı

$\sum pq$  her madde için hesaplanan ( p x q)'ların toplamı

$S^2$  : test toplam puanlarının varyansı olmak üzere,

#### **Kr-20 Formülü**

$$r_{i\zeta} = \left( \frac{n}{n-1} \right) \left( \frac{S^2 - \sum pq}{S^2} \right)$$

Maddeleri yanıtlanmayanlar ve yanlış cevap verenler değerlendirilmediği, doğru cevap verenlerin değerlendirildiği durumlarda Kr-20 formülü kullanılır. Testteki maddeler farklı puanlar ile değerlendirilecekse Kr-20 formül kullanılamamaktadır. Testte bulunan maddelerin güçlük değerleri birbirine yakınsa Kr-21 formülü kullanılabilir (Ergin, 1995).

Kr-21 formülü (Ergin, 1995);

n : testteki madde sayısı

X<sub>t</sub>: Toplam puanların ortalaması

$S^2$ : test toplam puanlarının varyansı olmak üzere;

#### **Kr-21 Formülü**

$$r_{i\zeta} = \left( \frac{n}{n-1} \right) \left( 1 - \frac{\bar{X}_t (n - \bar{X}_t)}{nS_t^2} \right)$$

Çalışmada kullanılan Scratch Akademik Başarı Testi (EK 4) ve Tospa Akademik Başarı (EK 5) testi için belirlenen belirtke tablosu Tablo 3.3 ve Tablo 3.4’de belirtilmektedir.

**Tablo 3. 3** Scratch akademik başarı testinin (EK 4) hazırlanmasına yönelik belirtke tablosu

Sorular	Kazanımlar
1	Algoritma kavramını açıklar.
2	Akış şeması bileşenlerini ve işlevlerini açıklar.
3	Problemin çözümü için bir algoritma geliştirir.
4	Hatalı bir algoritmayı doğru çalışacak biçimde düzenler
5,6,7,8,9	Blok tabanlı programlama aracının arayüzünü ve özelliklerini tanıır
10,11,12	Blok tabanlı programlama aracında sunulan bir programı verilen ölçütlere göre geliştirerek düzenler.

**Tablo 3. 4** Tospaa akademik başarı (EK 5) testinin hazırlanmasına yönelik belirtke tablosu

Sorular	Kazanımlar
1	Algoritma kavramını açıklar.
2	Akış şeması bileşenlerini ve işlevlerini açıklar.
3	Problemin çözümü için bir algoritma geliştirir.
4	Hatalı bir algoritmayı doğru çalışacak biçimde düzenler
5	Blok tabanlı programlama aracında sunulan bir programı verilen ölçütlere göre geliştirerek düzenler.

Çoktan seçmeli test olan her iki test için Kr-21 değerlerine bakıldığında Scratch Akademik Başarı Testinin güçlük indisi = .758 ve Tospaa Akademik Başarı testinin güçlük indisi = .726 olarak bulunmuştur.

Deney ve kontrol grubuna uygulama öncesi (ön testler) uygulanan akademik başarı testi, bilgisayar destekli eğitime ilişkin öz-yeterlik algı ölçeği, bilgisayar destekli eğitime ilişkin tutum ölçeği ve eğitsel bilgisayar oyunları destekli kodlama öğrenimine yönelik tutum ölçeği sonuçlarının karşılaştırma (ilişkisiz örneklem için bağımsız t-testi) sonucu Tablo 3.5'te, Tablo 3.6'da ve Tablo 3.7'de verilmiştir.

**Tablo 3. 5** Gruplar arası bilgisayar destekli eğitime ilişkin öz-yeterlik algı ölçeği ön test karşılaştırma sonuçları

Gruplar	N	$\bar{X}$	Ss	Sd	t	p	
Ön test	Deney grubu	21	66,95	4,82	43	-,972	.336*
	Kontrol	24	65,57	4,71			

grubu

\*p<0.05

Uygulama öncesi deney ve kontrol grupları belirlendikten sonra, deney ve kontrol grubuna yapılan ön testlerde (deney grubu ön test ortalaması  $\bar{X}=66,95$ ; kontrol grubu ön test ortalaması  $\bar{X}=65,57$ ) \*p<.05 anlamlılık düzeyi için .05<.336 büyüktür ve anlamlı değildir. Bir başka deyişle katılımcıların Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algı Ölçeği ön test puanları anlamlı bir farklılık göstermemektedir.

**Tablo 3. 6** Gruplar arası bilgisayar destekli eğitime ilişkin tutum ölçeği ön test karşılaştırma sonuçları

	Gruplar	N	$\bar{X}$	Ss	Sd	t	p
Ön test	Deney grubu	21	62,20	7,11			
	Kontrol grubu	24	59,47	5,41	43	-1,433	.159*

\*p<0.05

Uygulama öncesi deney ve kontrol grupları belirlendikten sonra, deney ve kontrol grubuna yapılan ön testlerde (deney grubu ön test ortalaması  $\bar{X}=62,20$ ; kontrol grubu ön test ortalaması  $\bar{X}=59,47$ ) \*p<.05 anlamlılık düzeyi için .05<.159 olduğu için anlamlı değildir. Bir başka deyişle katılımcıların Bilgisayar Destekli Eğitime İlişkin Tutum Ölçeği ön test puanları anlamlı bir farklılık göstermemektedir.

**Tablo 3. 7** Gruplar arası eğitsel bilgisayar oyunları destekli kodlama öğrenimine yönelik tutum ölçeği ön test karşılaştırma sonuçları

	Gruplar	N	$\bar{X}$	Ss	Sd	t	p
Ön test	Deney grubu	21	92,20	13,91			
	Kontrol grubu	24	87,71	16,14	43	- 1.003	.321*

\*p<0.05

Uygulama öncesi deney ve kontrol grupları belirlendikten sonra, deney ve kontrol grubuna yapılan ön testlerde (deney grubu ön test ortalaması  $\bar{X}=92,20$ ; kontrol grubu ön test ortalaması  $\bar{X}=87,71$ ) \*p<.05 anlamlılık düzeyi için .05<.321 olduğu için anlamlı değildir. Bir başka deyişle katılımcıların Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeği ön test puanları anlamlı bir farklılık göstermemektedir.

Bir başka deyişle çalışma grubu deney ve kontrol grubu sayılarının yaklaşık olarak benzer olmasından ve yapılan istatistik testlerinin ön test sonuçlarının karşılaştırmasından (Bilgisayar destekli eğitime ilişkin öz-yeterlik algı ölçeği, bilgisayar destekli eğitime ilişkin tutum ölçeği, eğitsel bilgisayar oyunları destekli kodlama öğrenimine yönelik tutum ölçeği) bulunan bu sonuçlarla her iki grup içinde (deney ve kontrol grubu) araştırma öncesi denk olduğu ve araştırmanın uygulaması için uygun olduğu belirlenmiştir.

### **3.3.2. Bilgisayar Destekli Eğitime İlişkin Tutum Ölçeği**

Bilgisayar destekli eğitime ilişkin tutum ölçeği (EK 1) aşağıda açıklanan aşamalardan geçerek hazırlanmıştır.

Ölçek araştırmacısı tarafından ilgili literatür taranarak ve öğretmen adaylarının da görüşüne başvurularak deneme amaçlı tutum maddeleri yazılmıştır. Yazılan maddelerin uygun uzunlukta olup olmadığı, anlaşılabilirlik seviyesinin ve yeterlik dereceleri hakkında değerlendirilmesi için 3 tane eğitim programları ve öğretim (EPÖ) uzmanı, 5 tane ölçme değerlendirme uzmanının görüşüne başvurulmuştur. Görüşmenin ardından toplam 41 maddeden oluşan bir deneme ölçeği hazırlanmıştır. Ölçekteki maddeler “kesinlikle katılıyorum”, “katılıyorum”, “kararsızım”, “katılmıyorum”, “kesinlikle katılmıyorum” şeklinde belirtilen 5’li dereceleme ölçeğinde düzenlenmiştir. Olumlu soru maddelerinde 1 “kesinlikle katılmıyorum”, 2 “katılmıyorum”, 3 “kararsızım”, 4 “katılıyorum”, 5 “kesinlikle katılıyorum” olarak alınmıştır. Olumsuz soru maddelerinde 1 “kesinlikle katılıyorum”, 2 “katılıyorum”, 3 “kararsızım”, 4 “katılmıyorum”, 5 “kesinlikle katılmıyorum” olarak alınmıştır.

Hazırlanan bu deneme ölçeği 2005–2006 eğitim öğretim yılı güz döneminde Hacettepe Üniversitesi Eğitim Fakültesi sınıf öğretmenliği bölümünde eğitim gören (N:151) 4. Sınıf öğretmen adayları üzerinde uygulanmıştır. Örneklem grubunu oluşturan öğretmen adayların 116’sı bayan öğrenci oluştururken 35’ini ise erkek öğrenciler oluşturmaktadır.

Hacettepe Üniversitesi Eğitim Fakültesinde öğrenim görmekte olan öğretmen adaylarının bilgisayar destekli eğitime ilişkin tutumlarını belirlemek amacıyla geliştirilen bu ölçeğin yapı geçerliğini ortaya koymak ve ölçekte yer alan maddelerin faktör yüklerinin belirlenerek boyutlandırılması amacıyla faktör analizi yapılmıştır.

Ölçekte yer alan maddelerin benzer davranışları ne ölçüde ölçtüğünü belirleme, alınan puanlar ile ölçeğin toplam puanı arasındaki ilişki (madde-toplam korelasyonu) hesaplanarak yapılmıştır. Ayrıca ölçekte yer alan her bir maddenin tutum düzeyi bakımından bireyleri ayırt etmede ne derece yeterli olduğunu belirlemek amacıyla ölçek puanlarına göre üst %27 ve alt %27'lik grubun madde puanları arasındaki farkın anlamlılığına bakılmıştır.

Ölçeğin güvenirlik çalışması için ise Cronbach Alpha Katsayısı formülü kullanılmıştır.

Faktör yükleri ve madde-toplam korelasyonları katsayısı düşük olan maddeler ölçekten çıkarılmıştır. Bütün bu işlemlerden sonra 20 maddelik nihai ölçek belirlenmiştir.

Belirlenen ölçeğin maddelerden 10 tanesi olumlu, 10 tanesi ise olumsuz özellik göstermektedir. Ölçeğin Kaiser-Meyer-Olkin (KMO) katsayısı 0.88, Barlett Testi anlamlılık değeri 0.000 bulunmuştur. Ölçeğin Cronbach- alpha güvenirlik katsayısı 0.93 olarak bulunmuştur (Arslan, 2006b).

### **3.3.3. Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algı Ölçeği**

Bilgisayar destekli eğitime ilişkin öz yeterlilik algısı ölçeğinin (EK 2) geliştirilmesinde şu aşamalar izlenmiştir:

2004-2005 eğitim öğretim yılında bahar döneminde Zonguldak Karaelmas Üniversitesi Eğitim Fakültesi sınıf öğretmenliği bölümü 3. sınıfında okuyan toplam 40 öğrenciye öğretmenlik mesleğine ilişkin öz yeterliliklerini içeren bir kompozisyon yazmaları istenmiştir. Yazdırılan bu kompozisyonlar, araştırmacı tarafından analiz edilip ölçek maddeleri 49 olarak belirlenmiştir. Ölçekte kullanılan maddelerin uygun uzunlukta olup olmadığı, anlaşılabilirlik seviyesinin nasıl olduğu ve yeterlilik dereceleri hakkında alanında uzman kişilerden görüş alınmıştır. Bu görüşmelerin ardından toplam 44 maddeden oluşan deneme ölçeği hazırlanmıştır. Ölçekte bulunan maddeler “kesinlikle katılıyorum”, “katılıyorum”, “kararsızım”, “katılmıyorum”, “kesinlikle katılmıyorum” şeklinde belirtilen 5’li dereceleme ölçeği olarak hazırlanmıştır. Olumlu soru maddelerinde 1 “kesinlikle katılmıyorum”, 2 “katılmıyorum”, 3 “kararsızım”, 4 “katılıyorum”, 5 “kesinlikle katılıyorum” olarak alınmıştır. Olumsuz soru maddelerinde

1 “kesinlikle katılıyorum”, 2 “katılıyorum”, 3 “karasızım”, 4 “katılmıyorum”, 5 “kesinlikle katılmıyorum” olarak alınmıştır.

Hazırlanan ölçek aynı bölümde okuyan (N:130) 3. ve 4. sınıf öğrencilerine uygulanmıştır. Bu öğrencilerden 47’si 3. sınıf, 83’ü 4. sınıf öğrencisidir. Öğrencilerden 74’ü kadın, 56’sı erkektir. 3. ve 4. sınıf öğrencilerinin beraber alınmasının sebebi: her iki grubun da bilgisayar destekli eğitime yönelik almaları gereken dersin tamamını almış olmalarıdır. Ölçek, kendilerine daha önce konu hakkında kompozisyon yazdırılan öğrencilere uygulanmamıştır.

Eğitim bilimleri fakültesinde eğitim öğretim görmekte olan öğretmen adaylarının bilgisayar destekli eğitime ilişkin öz yeterliliklerini belirlemek amacıyla geliştirilen ölçeğin yapı geçerliğini ortaya koymak ve ölçekte yer alan maddelerin faktör yüklerinin belirlenerek boyutlandırılması amacıyla faktör analizi yapılmıştır. Ölçekte yer alan maddelerin benzer davranışları ne ölçüde ölçtüğünü belirleme, alınan puanlar ile ölçeğin toplam puanı arasındaki ilişki (madde- toplam korelasyonu) hesaplanarak yapılmıştır. Aynı zamanda ölçekte yer almakta olan her bir maddenin öz yeterlilik düzeyi bakımından bireyleri ayırt etmede ne derece yeterli olduğunu belirlemek amacıyla ölçek puanlarına göre üst %27 ve alt %27’lik grubun madde puanları arasındaki farkın anlamlılığına bakılmıştır. Edilen verilerin örneklem grubuna uygunluğu 0.001 düzeyinde KMO değeri .86, Barlett Testi anlamlılık değeri 0.000 olarak bulunmuştur.

Yapılan uygulamanın ardından ölçek 20 maddeye indirilmiş bu maddelerin 11’i olumlu madde 9’u olumsuz madde özelliği göstermektedir. Ölçeğin Kaiser-Meyer-Olkin (KMO) katsayısı 0.86, , Barlett Testi anlamlılık değeri 0.000 bulunmuştur. Cronbach- alpha güvenirlik katsayısı 0.94 olarak bulunmuştur (Arslan, 2006a).

### **3.3.4. Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeği**

Nicel araştırma yöntemlerinden betimsel araştırma deseni kullanılarak Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeği (EK 3) hazırlanmıştır.

Araştırma 2015-2016 eğitim öğretim yılında Elazığ şehrinde bulunan Merkez Cumhuriyet Ortaokulu öğrencilerinin arasından rastgele örnekleme yöntemiyle seçilmiş olan 127 erkek öğrenci (%47), 113 kız öğrenci (%53) toplam 240 öğrenci aracılığı ile gerçekleştirilmiştir. Çalışmanın yapıldığı okulun konumu il merkezinde olmasına rağmen çevresindeki ailelerin gelir düzeyleri ve velilerin profilleri karma dağılım göstermektedir.

Literatür taraması yapılarak ölçeğin geliştirilmesi sürecine başlanmış, kodlama hakkında yapılan çalışmalarda hangi veri toplama araçlarının kullanıldığı incelenmiştir. Araştırmalarda gözlem formu (Taylor ve diğerleri, 2010), problem çözme becerileri testi (Kalelioglu ve Gülbahar, 2014; Nam ve diğerleri, 2010) ,öğrenci görüşlerinin (Genç ve Karakuş; Malan ve Leitner, 2007; Nam ve diğerleri, 2010) veri toplama aracı olarak kullanılması ile hazırlanmıştır. Fakat eğitsel bilgisayar oyunları destekli kodlama öğrenimine yönelik ortaokul seviyesinde öğrenim gören öğrencilerinin tutumlarının olduğu Türkçe bir ölçeğe rastlanmamıştır. Türkçe bir ölçek ihtiyacına cevap vermesi amacıyla ölçek 5’li likert tipte hazırlanmıştır. Ölçeğin maddeleri ise kesinlikle katılıyorum, katılıyorum, kararsızım, katılmıyorum ve kesinlikle katılmıyorum olarak belirlenmiştir. Hazırlanan ölçekte geliştirilen maddeler ortaokul öğrencilerinin seviyelerine uygun olacak biçimde oldukça sade bir dille, rahat anlaşılır biçimde yazılmıştır. 40 madde olarak hazırlanan ölçek, 28 olumlu madde ve 12 olumsuz maddeden oluşmaktadır. Anlam olarak aynı olmasına rağmen farklı ifadeler içeren benzer önermelerle ölçeğin güvenilirliği artırılmaya çalışılmıştır (madde 11- madde 18; madde 8- madde 22). Hazırlanan ölçeğin kapsam geçerliliğinin tespiti için ölçek, alanında uzman olan Bilgisayar ve Öğretim Teknolojileri Bölümünde (BÖTE) 3 öğretim üyesi, 1 Türkçe Eğitimi öğretim üyesi ile farklı okullarda görev alan 3 Bilgi İletişim Teknolojileri öğretmenine incelettirilmiştir. Yapılan incelemelerin ardından gelen dönütlere neticesinde 4 maddenin ölçeğin planlanan alt boyutlarını ölçmediği kanısına varıldığı için bu maddeler ölçekten çıkarılmıştır (madde 2, madde 10, madde 21, madde 19).

Anlaşılmayan noktaların belirlenmesi için ölçeğin 24 adet ortaokul öğrencisi (her kademedan 6’şar adet öğrenci) ile ön deneme aşaması gerçekleştirilmiştir. Ölçeğin 3. maddesinde yer alan “kendimi bilgisayar oyunundaki kahramanın yerine koyarım” cümlesinin yeterince açıklayıcı olmadığı kanısına varıldığı için madde “bilgisayar

oyunundaki kahraman başarılı olduğunda kendimi başarmış hissederim” ifadesi olarak değiştirilmiştir. Ölçeğin 7. maddesinde yer alan “bilgisayarda kodlama yapmayı biliyorum” cümlesi ise “bilgisayarda kodlama yapmayı bilen kişilerin kavrama yeteneği gelişir” ifadesine dönüştürülerek aradaki sebep sonuç ilişkisi vurgulanmıştır. Ölçekte yer alan bazı kelimelerin öğrencilerin büyük kısmı tarafından tam anlaşılmadığı için eşli oyun ikili oyun olarak, sosyal- aktif olarak ve pasifleştirme-tembelleştirme olarak değiştirilmiştir.

Ön pilot olarak 240 ortaokul öğrencisine uygulanan ölçeğin yapı geçerliliği ve güvenilirlik çalışması tamamlanmıştır. Ortaokulda öğrenim gören öğrencilerinin eğitsel bilgisayar oyunları destekli kodlama öğrenimine yönelik tutumlarını analiz etmek amacı doğrultusunda ölçek, 40 maddeden ve 3 teorik boyuttan oluşacak şekilde geliştirilmiştir. Teorik 3 boyut ise şunlardır; öğrencilerin kodlama öğrenimine karşı isteği, bilgisayar oyunlarının derslerde eğitim amaçlı kullanımına yönelik öğrencilerin ilgisi ve bilgisayarın asosyalleşmesine yönelik öğrencilerin endişeleri tutumlarıdır. Aracın faktör desenini belirlemek için açımlayıcı faktör analizi yapılmıştır. Ölçeğin güvenilirliğinin tespiti amacıyla Cronbach Alpha katsayısı hesaplanmıştır. Veriler SPSS 23 programı ile analiz edilmiştir.

Ortaokulda öğrenim gören öğrencilerinin eğitsel bilgisayar oyunları destekli kodlama öğrenimine yönelik geçerli ve güvenilir bir tutum ölçeği geliştirilmek için yapılan çalışma neticesinde hazırlanan ölçek toplam 28 maddeden ve üç faktörden oluşmuştur. Birinci faktör 12 maddeden, ikinci faktörde 11 maddeden ve üçüncü faktörde ise 5 maddeden oluşmaktadır. Bu faktörlerin, öğrencilerin kodlama öğrenimine karşı isteği, bilgisayar oyunlarının derslerde eğitim amaçlı kullanımına yönelik öğrencilerin ilgisi ve bilgisayarın asosyalleşmesine yönelik öğrencilerin endişesi ile ilgili tutumları yansıttığı söylenebilir. Yapılan istatistikler sonucunda geliştirilen ölçek geçerli ve güvenilir olarak bulunmuştur. Görsel kodlama dilleri ile yapılan araştırmalar son yıllarda giderek artış göstermektedir. Yapılan bu araştırmalarda ise bilgisayar kodlama ile erken yaşlarda tanışan öğrencilerin problem çözme ve eleştirel düşünme becerilerinde yükseliş olduğu kanısına varılmıştır (Keçeci ve diğerleri, 2016).

Hazırlanan ölçek, öğrencilerin kodlama öğrenimine karşı isteği, bilgisayar oyunlarının derslerde eğitim amaçlı kullanımına yönelik öğrencilerin ilgisi ve bilgisayarın asosyalleşmesine yönelik öğrencilerin endişesi temel alınarak

geliştirilmiştir. Açımlayıcı faktör analizi ölçeğin faktör desenini belirlemek için yapılmıştır. KMO değeri ise .840 olarak hesaplanmıştır. Ki-kare değerinin Barlett küresellik sonuçlarına göre anlamlı olduğu görülmüştür ( $\chi^2_{(630)}=3087.798$ ;  $p<.01$ ). Bileşenlerin varyansa yaptıkları toplam katkı % 41.39'dur. Ölçeğin Cronbach Alfa güvenilirlik katsayısı .833 olarak hesaplanmıştır. 22 olumlu 6 olumsuz madde içeren 28 maddelik ölçek, geçerlik güvenilirlik çalışmaları sonucunda düzenlenmiştir. Geliştirilen ölçeğin geçerli ve güvenilir olduğu yapılan istatistiksel analizler neticesinde görülmüştür (Keçeci ve diğerleri, 2016).

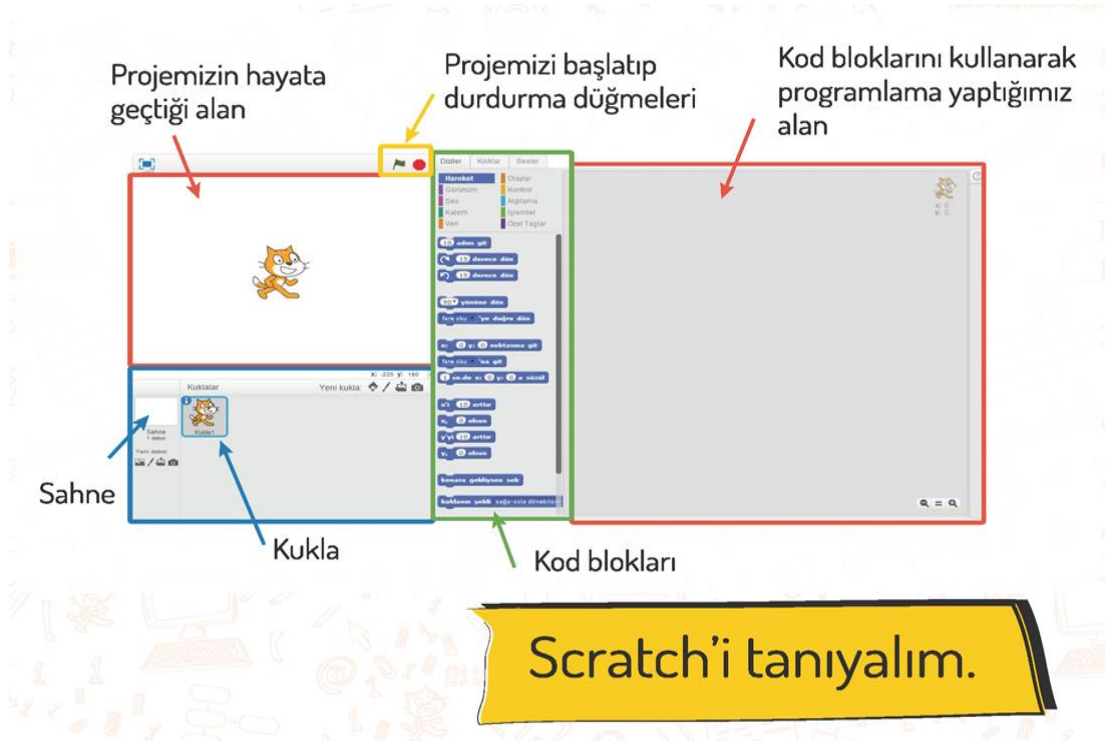
### 3.4 Verilerin Toplanması

Bu çalışma Milli Eğitim Bakanlığı'na bağlı okullardan Konya ilinde bulunan Meram Gödene TOKİ İmam Hatip Ortaokulu'nda 2020-2021 eğitim öğretim yılının birinci döneminde toplam 10 hafta süresince Bilişim Teknolojileri ve Yazılım dersi kapsamında ortaokul 6. sınıf öğrencileriyle yürütülmüştür. Problem çözme, programlama ve özgün ürün geliştirme Bilişim Teknolojileri ve Yazılım dersinin genel amaçlarından (TTKB, 2012b).

Nicel veriler toplanırken kullanılan problem çözme becerisi ölçeği ön test olarak 21-25 Eylül 2020 tarihleri arasında, son test ise 30 Kasım- 4 Aralık 2020 tarihleri arasında internet ortamındaki bir formlar aracılığıyla öğrencilere uygulanmıştır. Öğrenciler 10 haftalık sürede whatsapp gruplarından gönderilen çalışma ve aktivite kağıtları ile ilk kez kullandıkları Scratch ve Tospaa programları ile kodlamayı öğrenmişlerdir. Ön testler Scratch ve Tospaa programlama sürecine başlamadan önce, son testler ise Scratch ve Tospaa programlama sürecinin bitiminde uygulanmıştır.

Programlama öğretiminin ilk haftasında kontrol grubuyla Unplugged (Bilgisayarsız) Kodlama Nedir? (Tospaa) konusu işlenerek başlanmıştır. Öğrencilere Unplugged (Bilgisayarsız) Kodlama araçları hakkında ve bilgisayarsız sınıflarda kodlamanın nasıl yapılacağı ile ilgili bilgilerin olduğu bir power point sunusu paylaşılmıştır.

Programlama öğretiminin ilk haftasında deney grubuyla Blok Tabanlı Programlama Nedir? ve Scratch ile Tanışalım isimli power point sunumları paylaşılmıştır. Paylaşılan Scratch sunumundan bir örnek şekilde verilmiştir.



Şekil 3.3 Scrath Arayüzü (ScratchAbout, 2020)

Programlama öğretiminin ikinci haftasında kontrol grubuyla Tospaa bilgisayarsız kodlama oyunu tanıtımının yapıldığı sunu paylaşılmıştır. Paylaşılan Tospaa sunumundan bir örnekler şekillerde verilmiştir.



Şekil 3.4 Tospaa Tanıtım Kartları (Tospaa, 2020)

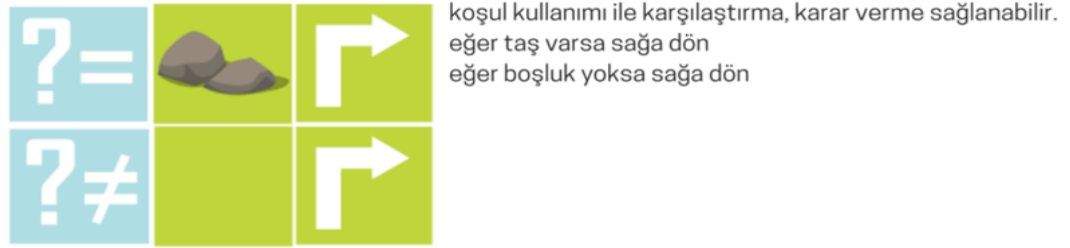


ileri , sola dön, sağa dön  
oyun sonunda yön kartlarını  
en az kullanan kazanmış sayılır.

Şekil 3.5 Tospaa Yön Kartlarını (Tospaa, 2020)



Şekil 3.6 Tospaa Döngü Kartları (Tospaa, 2020)



Şekil 3.7 Tospaa Koşul Kartları (Tospaa, 2020)

Programlama öğretiminin ikinci haftasında deney grubuyla Scratch nedir isimli Word belgesi paylaşılmış belgede Scratch programının kurulumu ve kullanımı hakkında bilgi verilmiş. Programının bilgisayarlara kurulumunun zorunlu olmadığı online olarak da programa <https://scratch.mit.edu> linkinden girilerek kullanılabilirliğinden bahsedilmiştir. Aynı zamanda programı bilgisayarına kurmak isteyenler için de kurulum aşamalarının yer aldığı bir power point sunumu paylaşılmıştır.

Programlama öğretiminin üçüncü haftasında kontrol grubuyla çalışma kağıdı 1 isimli Word belgesi paylaşılmış böylece öğrencilerin başlangıç seviyesinde programı kullanmaya başlamaları amaçlanmıştır.

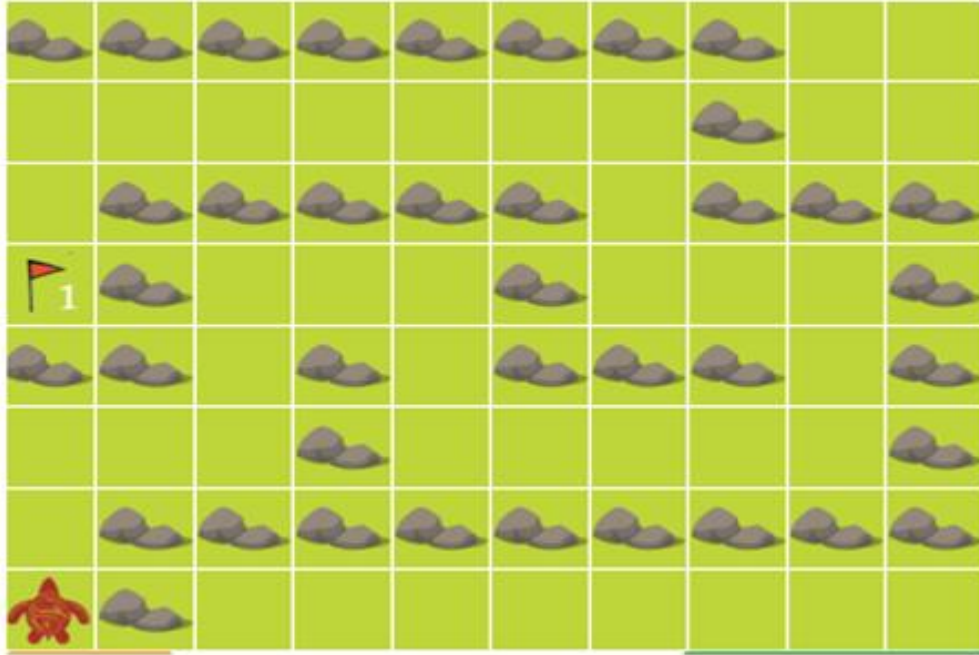
Programlama öğretiminin üçüncü haftasında deney grubuyla Scratch nedir 2 isimli Word belgesi paylaşılmış belgede kukla bilgisi, kılıklar, bloklar, hareket ve görünüm bloklarının nasıl kullanıldığı ve bu bloklarının anlamlarının neler olduğunun anlatıldığı word belgesi paylaşılmıştır. Paylaşılan word bir örnekler şekillerde verilmiştir.



Şekil 3.8 Scratch Kılıklar ve Bloklar (ScratchAbout, 2020)

Programlama öğretiminin dördüncü haftasında kontrol grubuyla çalışma kağıdı 2 isimli word belgesi paylaşılmıştır ve belgede iki adet uygulama bulunmaktadır. Uygulamalardan birisinde üç tospaanın tek bir kod ile hedeflere ulaştırılması istenilirken diğer uygulamada labirent isimli oyun tahtasında tospaaı engellere değmeden hedeflere ulaştırılması istenilmiştir. Paylaşılan çalışma kağıdından bir örnek şekilde verilmiştir.

## Labirent



Şekil 3.9 Labirent Uygulaması (Tospaa, 2020)

Programlama öğretiminin dördüncü haftasında deney grubuyla Scratch nedir 3 isimli Word belgesi paylaşılmış belgede ses ve kalem bloklarının nasıl kullanıldığı ve anlamlarının neler olduğunun açıklandığı Word belgesi paylaşılmıştır.

Programlama öğretiminin beşinci haftasında kontrol grubuyla çalışma kağıdı 3 isimli word belgesi paylaşılmış bu belgede iki adet oyun tahtası paylaşılmış birincisinde kırmızı tospaanın tek sayıları mavi tospaanın ise çift sayıları toplaması istenilmiş böylece matematik öğretimi ile kodlama öğretimi birlikte verilmek istenilmiştir, ikinci oyun tahtasında ise öğrencilerin tek bir kodla iki tospaayı da hedeflerine ulaştırmaları istenilmiştir.

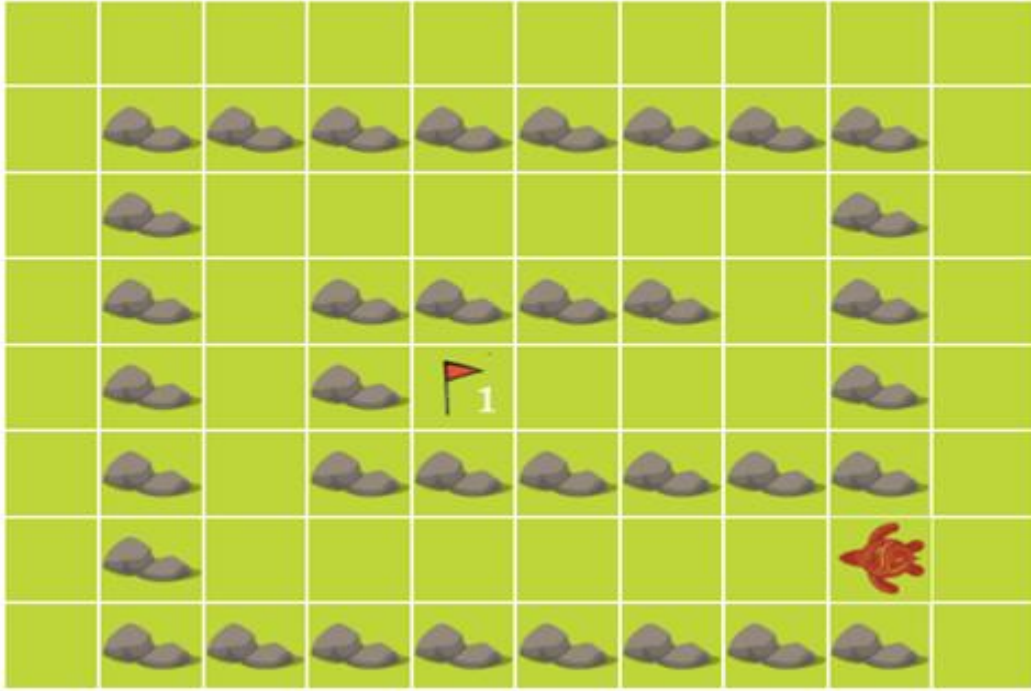
Programlama öğretiminin beşinci haftasında deney grubuyla Scratch nedir 4 isimli Word belgesi paylaşılmış belgede olaylar ve kontrol bloklarının nasıl kullanıldığı ve anlamlarının neler olduğunun açıklandığı word belgesi paylaşılmıştır.

Programlama öğretiminin altıncı haftasında kontrol grubuyla çalışma kağıdı 4 isimli word belgesi paylaşılmış bu belgede tospaaların engellere değmeden istenilen hedeflere ulaştırılması istenilmiştir.

Programlama öğretiminin altıncı haftasında deney grubuyla Scratch nedir 5 isimli Word belgesi paylaşılmış belgede algılama, işlemler ve veri bloklarının nasıl kullanıldığı ve anlamlarının neler olduğu açıklanmıştır.

Programlama öğretiminin yedinci haftasında kontrol grubuyla çalışma kağıdı 5 isimli Word belgesi paylaşılmış engellere değmeden istenilen hedeflere ulaştırılması istenilmiştir. Paylaşılan çalışma kağıdından bir örnek şekilde verilmiştir.

## Dön Dolaş



Şekil 3.10 Dön Dolaş Uygulaması (Tospaa, 2020)

Programlama öğretiminin yedinci haftasında deney grubuyla akvaryum isimli Word belgesi paylaşılmış belgede belirtilen komutların kullanılması ile içinde balıkların hareket ettiği bir akvaryum oluşturulması istenilmiştir.

Programlama öğretiminin sekizinci haftasında kontrol grubuyla çalışma kağıdı 6 isimli Word belgesi paylaşılmış tospaaların engellere değmeden istenilen hedeflere ulaştırılması istenilmiştir.

Programlama öğretiminin sekizinci haftasında deney grubuyla papağan isimli Word belgesi paylaşılmış belgede yer alan adımların uygulanması ile Scratch sahnesinde uçan bir papağan tasarımının oluşturulması istenilmiştir. Paylaşılan belgeden bir örnek şekilde verilmiştir.



Şekil 3. 11 Papağan Uygulaması (ScratchAbout, 2020)

Programlama öğretiminin dokuzuncu haftasında kontrol grubuyla çalışma kağıdı 7 isimli Word belgesi paylaşılmış birden fazla tospaanın aynı oyun tahtasında engellere değmeden istenilen hedeflere ulaştırılması istenilmiştir.

Programlama öğretiminin dokuzuncu haftasında deney grubuyla yön tuşları isimli Word belgesi paylaşılmış belgede yer alan komutlarla öğrencilerin Scratch ortamında tasarladıkları kedinin yön tuşları ile adım atarak yürütmesi sağlayacaklardır.

Programlama öğretiminin onuncu haftasında kontrol grubuyla Tospaa eğitimi hakkında genel tekrar yapılmış ve son testler paylaşılmıştır.

Programlama öğretiminin dokuzuncu haftasında deney grubuyla Scratch genel tekrar isimli Word belgesi ve son testler paylaşılmıştır.

### 3.5 Verilerin Çözümlemesi

Katılımcıların demografik bilgileri betimsel istatistiklerle açıklanmıştır. Bu istatistikleri frekans, yüzde, aritmetik ortalama ve standart sapmadır. SPSS 22 (Statistical Package for Social Sciences) versiyonlu program kullanılarak çalışmanın nicel boyutunda deneysel işlem öncesi ve sonrasında toplanan sayısal verilerin analiz edilmiştir. Araştırmada ilişkili örneklem için t-testi kullanılmıştır. Bu analizde amaç deneysel işlem sürecinden geçen öğrencilerin araştırma öncesinde uygulanan ön test ile araştırma sonrasında uygulanan son testten toplanan verilerin karşılaştırılmasıdır. İlişkisiz örneklemde, iki ilişkisiz örneklem ortalamaları arasındaki farkın anlamlı olup olmadığını test etmek için t-testi kullanılır (Büyüköztürk, 2011).

## BÖLÜM 4

### 4 BULGULAR

Araştırmanın bu bölümünde araştırma soruları, araştırmadan toplanan verilere göre yapılan istatistiksel analizlerinin sonuçları ve bu sonuçlara araştırma sorularına ait yorumlar ayrıntılı olarak sunulmuştur.

#### 4.1 Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algı Ölçeği Güvenirlik Analizi Bulguları

Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algı Ölçeği uygulama sonrasında çalışma grubundan toplanan veriler ile yapılan güvenirlilik testinde iç güvenirlilik katsayısı olan Cronbach's Alfa güvenirlilik katsayısı .870 olarak bulunmuştur.

**Tablo 4. 1** Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algı (BTÖYA) ölçeği güvenirlilik analizi sonuçları

Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algı Ölçeği	
Cronbach's Alfa	Ölçekteki Madde Sayısı
.870	20

#### 4.2 Bilgisayar Destekli Eğitime İlişkin Tutum Ölçeği Güvenirlilik Analizi Bulguları

Bilgisayar Destekli Eğitime İlişkin Tutum Ölçeğine yönelik uygulama sonrasında çalışma grubundan toplanan veriler ile yapılan güvenirlilik testinde iç güvenirlilik katsayısı olan Cronbach's Alfa güvenirlilik katsayısı .81 olarak bulunmuştur.

**Tablo 4. 2** Bilgisayar destekli eğitime ilişkin tutum ölçeği güvenirlilik analizi sonuçları

Bilgisayar Destekli Eğitime İlişkin Tutum Ölçeği	
Cronbach's Alfa	Ölçekteki Madde Sayısı
.81	20

#### 4.3 Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeği Güvenirlilik Analizi Bulguları

Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeğine yönelik uygulama sonrasında çalışma grubundan toplanan veriler ile yapılan

güvenirlik testinde iç güvenirlik katsayısı olan Cronbach's Alfa güvenirlik katsayısı .92 olarak bulunmuştur.

**Tablo 4. 3** Eğitsel bilgisayar oyunları destekli kodlama öğrenimine yönelik tutum ölçeği

Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeği	
Cronbach's Alfa	Ölçekteki Madde Sayısı
.92	28

#### 4.4 Deney Grubu Ön Test - Son Test Sonuçlarının İncelenmesi

##### **Deney grubu Akademik Başarı Testi Ön test – Son test karşılaştırması (bağımlı örneklem için t – testi / eşleştirilmiş t testi)**

Uygulama sonucunda deney grubu öğrencilerinin durumlarını belirlemek için yapılan ön test ve son testlerin karşılaştırmalarının sonucu Tablo 4. 4’de verilmiştir.

**Tablo 4. 4** Akademik başarı testi toplam için deney grubu ön test-son test karşılaştırma sonuçları

	Test	N	$\bar{X}$	Ss	Sd	t	p
<b>Deney Grubu</b>	Ön test	21	69,58	14,21	23	23,985	.000
	Son test	21	87,08	15,24			

\*P<0.05

Deney grubu ön test-son test puanları arasında (ön test ortalaması  $\bar{X}$  =69,58; son test ortalaması  $\bar{X}$  =87,08) istatistiksel olarak \*p<.05 anlamlılık düzeyi için farklı olduğu görülmüştür (p<0.05). Deney grubu öğrencilerinin katıldıkları bilgisayarlı kodlama uygulaması sonucunda akademik başarı testindeki başarılarını arttırdıkları belirlenmiştir (Tablo 4. 4).

##### **Deney grubu Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algı Ölçeği Ön test – Son test karşılaştırması (bağımlı örneklem için t – testi / eşleştirilmiş t testi)**

Deney grubu öğrencilerinin uygulama sonucunda durumlarını belirlemek için yapılan ön test ve son testlerin karşılaştırmalarının yapılmıştır. Sonuç Tablo 4. 5’de verilmiştir.

**Tablo 4. 5** Bilgisayar destekli eğitime ilişkin öz-yeterlik algı ölçeği için deney grubu ön test-son test karşılaştırma sonuçları

	Test	N	$\bar{X}$	Ss	Sd	t	p
<b>Deney</b>	Ön test	21	66,95	4,82	23	68,016	.000
<b>Grubu</b>	Son test	21	73,79	4,13			

\*P<0.05

Yapılan uygulamada deney grubunun ön test-son test puanları arasında (ön test ortalaması  $\bar{X}=66,95$ ; son test ortalaması  $\bar{X}=73,79$ ) \*p<.05 anlamlılık düzeyi bakımından istatistiksel olarak farklı olduğu bulunmuştur (p<0.05). Deney grubu öğrencilerinin katıldıkları bilgisayarlı kodlama uygulaması sonucunda Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algılarını arttırdıkları belirlenmiştir (Tablo 4. 5).

**Deney Grubu Bilgisayar Destekli Eğitime İlişkin Tutum Ölçeği Ön Test – Son Test Karşılaştırması (Eşleştirilmiş T Testi)**

Deney grubu öğrencilerinin uygulama sonucunda durumlarını belirlemek için yapılan ön test ve son testlerin karşılaştırılmıştır. Bu karşılaştırma sonucu Tablo 4. 6'da verilmiştir.

**Tablo 4. 6** Bilgisayar destekli eğitime ilişkin tutum ölçeği için deney grubu ön test-son test karşılaştırma sonuçları

	Test	N	$\bar{X}$	Ss	Sd	t	p
<b>Deney</b>	Ön test	21	62,20	7,11	23	42,840	.000
<b>Grubu</b>	Son test	21	74,45	6,92			

\*P<0.05

Yapılan uygulamada deney grubunun ön test-son test puanları arasında (ön test ortalaması  $\bar{X}=62,20$ ; son test ortalaması  $\bar{X}=74,45$ ) \*p<.05 anlamlılık düzeyi bakımından istatistiksel olarak farklı olduğu bulunmuştur (p<0.05). Deney grubu öğrencilerinin katıldıkları Scratch uygulaması sonucunda Bilgisayar Destekli Eğitime İlişkin Tutumlarını arttırdıkları belirlenmiştir (Tablo 4. 6).

**Deney Grubu Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeği Ön Test – Son Test Karşılaştırması (Eşleştirilmiş T Testi)**

Deney grubu öğrencilerinin uygulama sonucunda durumlarını belirlemek

için yapılan ön test ve son testlerin karşılaştırılmıştır. Bu karşılaştırma sonucu Tablo 4.7’de verilmiştir.

**Tablo 4.7** Eğitsel bilgisayar oyunları destekli kodlama öğrenimine yönelik tutum ölçeği için deney grubu ön test-son test karşılaştırma sonuçları

	Test	N	$\bar{X}$	Ss	Sd	t	p
<b>Deney Grubu</b>	Ön test	21	92,20	13,91	23	32,469	.000
	Son test	21	95,54	9,53			

\*P<0.05

Yapılan uygulamada deney grubunun ön test-son test puanları arasında (ön test ortalaması  $\bar{X}=92,20$ ; son test ortalaması  $\bar{X}=95,54$ ) \*p<.05 anlamlılık düzeyi bakımından istatistiksel olarak farklı olduğu bulunmuştur (p<0.05). Deney grubu öğrencilerinin katıldıkları Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutumları arttırdıkları belirlenmiştir (Tablo 4.7).

#### 4.5 Kontrol Grubu Ön Test - Son Test Sonuçlarının İncelenmesi

##### **Kontrol grubu Akademik Başarı Testi (Tospaa) Ön test – Son test karşılaştırması (eşleştirilmiş t testi)**

Kontrol grubu öğrencilerinin uygulama sonucunda durumlarını ortaya koymak için ön test ve son testlerin karşılaştırmaları yapılmıştır. Sonucu Tablo 4.8’de verilmiştir.

**Tablo 4.8** Akademik başarı testi kontrol grubu toplam için ön test-son test karşılaştırma sonuçları

	Test	N	$\bar{X}$	Ss	Sd	t	p
<b>Kontrol Grubu</b>	Ön test	24	45,71	23,62	20	8,867	,000
	Son test	24	73,80	27,60			

\*p<0.05

Yapılan uygulamada kontrol grubunun ön test-son test puanları arasında (ön test ortalaması  $\bar{X}=45,71$ ; son test ortalaması  $\bar{X}=73,80$ ) \*p<.05 anlamlılık düzeyi bakımından istatistiksel olarak farklı olduğu bulunmuştur (p<0.05). Kontrol grubu öğrencilerinin katıldıkları Tospaa uygulaması ile yapılan öğretim sonucunda Akademik Başarılarında anlamlı bir farklılık olduğu ve Akademik Başarılarını arttırdıkları görülmüştür (Tablo 4.8).

**Kontrol Grubu Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algı Ölçeği Ön Test – Son Test Karşılaştırması (Eşleştirilmiş t testi)**

Kontrol grubu öğrencilerinin uygulama sonucunda durumlarını ortaya koymak için ön test ve son testlerin karşılaştırmaları yapılmıştır. Sonucu Tablo 4.9’da verilmiştir.

**Tablo 4.9** Bilgisayar destekli eğitime ilişkin öz-yeterlik algı ölçeği için kontrol grubu ön test-son test karşılaştırma sonuçları

	Test	N	$\bar{X}$	Ss	Sd	t	p
<b>Kontrol</b>	Ön test	24	65,57	4,71	20	63,693	.000
<b>Grubu</b>	Son test	24	78,19	4,61			

\*p<0.05

Yapılan uygulamada kontrol grubunun ön test-son test puanları arasında (ön test ortalaması  $\bar{X}=65,57$ ; son test ortalaması  $\bar{X}=78,19$ ) \*p<.05 anlamlılık düzeyi bakımından istatistiksel olarak farklı olduğu bulunmuştur (p<0.05). Kontrol grubu öğrencilerinin katıldıkları Tospaa uygulaması ile yapılan öğretim sonucunda Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algılarında anlamlı bir farklılık olduğu görülmüştür (Tablo 4. 9).

**Kontrol grubu Bilgisayar Destekli Eğitime İlişkin Tutum Ön test – Son test karşılaştırması (eşleştirilmiş t testi)**

Kontrol grubu öğrencilerinin uygulama sonucunda durumlarını ortaya koymak için ön test ve son testlerin karşılaştırmaları yapılmıştır. Sonucu Tablo 4. 10’da verilmiştir.

**Tablo 4. 10** Bilgisayar destekli eğitime ilişkin tutum ölçeği için kontrol grubu ön test-son test karşılaştırma sonuçları

	Test	N	$\bar{X}$	Ss	Sd	t	p
<b>Kontrol</b>	Ön test	24	59,47	5,418	20	50,299	.000
<b>Grubu</b>	Son test	24	65,09	6,015			

\*p<0.05

Yapılan uygulamada kontrol grubunun ön test-son test puanları arasında (ön test ortalaması  $\bar{X}=59,47$ ; son test ortalaması  $\bar{X}=65,09$ ) \*p<.05 anlamlılık düzeyi bakımından istatistiksel olarak farklı olduğu bulunmuştur (p<0.05). Kontrol grubu öğrencilerinin katıldıkları Tospaa uygulaması ile yapılan öğretim sonucunda

Bilgisayar Destekli Eğitime İlişkin Tutumlarında anlamlı bir farklılık olduğu görülmüştür (Tablo 4. 10).

**Kontrol grubu Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeği Ön test – Son test karşılaştırması (eşleştirilmiş t testi)**

Kontrol grubu öğrencilerinin uygulama sonucunda durumlarını ortaya koymak için ön test ve son testlerin karşılaştırmaları yapılmıştır. Sonucu Tablo 4. 11’de verilmiştir.

**Tablo 4. 11** Eğitsel bilgisayar oyunları destekli kodlama öğrenimine yönelik tutum ölçeği için kontrol grubu ön test-son test karşılaştırma sonuçları

	Test	N	$\bar{X}$	Ss	Sd	t	p
<b>Kontrol Grubu</b>	Ön test	24	87,71	16,14	20	24,904	.000
	Son test	24	85,57	14,13			

\*p<0.05

Yapılan uygulamada kontrol grubunun ön test-son test puanları arasında (ön test ortalaması  $\bar{X}=87,71$ ; son test ortalaması  $\bar{X}=85,57$ ) \*p<.05 anlamlılık düzeyi bakımından istatistiksel olarak farklı olduğu bulunmuştur (p<0.05). Kontrol grubu öğrencilerinin katıldıkları Tospaa uygulaması ile yapılan öğretim sonucunda Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutumlarında anlamlı bir farklılık olduğu görülmüştür (Tablo 4. 11).

**4.6 Deney – Kontrol Grubu Ön Test - Son Test Sonuçlarının İncelenmesi**

**Deney-kontrol grubu Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algı Ölçeği son testler karşılaştırması (bağımsız t testi)**

Bilgisayarlı kodlamada Scratch uygulamasını kullanan öğrenciler (deney grubu) ile Tospaa uygulamasını kullanan öğrencilerin (kontrol grubu), “Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algı Ölçeği Sonuçları” karşılaştırıldığında ortaya çıkan sonuç Tablo 4. 12’de verilmiştir.

**Tablo 4. 12** Bilgisayar destekli eğitime ilişkin öz-yeterlik algı ölçeği için gruplar arası (deney- kontrol grubu) son test karşılaştırma (t- testi) sonuçları

Gruplar	N	$\bar{X}$	S	Sd	t	p	
<b>Son test</b>	Deney grubu	21	78,19	4,61	43	3,373	.002*
	Kontrol grubu	24	73,79	4,13			

\*P<0.05

Deney ve kontrol grubuna uygulama sonrası yapılan son testlerde  $*p < .05$  anlamlılık düzeyi için  $.00 < .05$  olduğu için anlamlıdır. Yapılan son testlerde (deney grubu son test ortalaması  $\bar{X} = 78,19$ ; kontrol grubu son test ortalaması  $\bar{X} = 73,79$ ) deney grubunun son test puanları kontrol grubunun son test puanlarından daha yüksek olduğu belirlenmiştir (Tablo 4. 12). Elde edilen sonuç uygulamanın Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algılarına yönelik deney grubunun lehine olduğunu göstermektedir. Ayrıca, bilgisayarlı kodlama ortamının, Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algı Ölçeği üzerindeki etki büyüklüğünü belirlemek için eta kare değeri incelenmiştir. Etki büyüklüğü değerleri  $\eta^2 = .133$  olarak hesaplanmıştır. Bu durumda, etki büyüklüğü değeri ( $\eta^2 = 0.133$ ) göz önünde bulundurulduğunda, bilgisayarlı kodlama ortamının, Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algıları üzerinde “geniş” bir etki büyüklüğüne sahip olduğu söylenebilir.

**Deney-kontrol grubu Bilgisayar Destekli Eğitime İlişkin Tutum Ölçeği son testler karşılaştırması (bağımsız t testi)**

Bilgisayarlı kodlama Scratch uygulamasını kullanan öğrenciler (deney grubu) ile Tospaa uygulamasını kullanan öğrencilerin (kontrol grubu), “Bilgisayar Destekli Eğitime İlişkin Tutum Ölçeği puanları” karşılaştırıldığında ortaya çıkan sonuç Tablo 4.13’de verilmiştir.

**Tablo 4. 13** Bilgisayar destekli eğitime ilişkin tutum ölçeği gruplar arası (deney - kontrol grubu) son test karşılaştırma (t - testi) sonuçları

Gruplar	N	$\bar{X}$	S	Sd	t	p
Son test	Deney grubu	21	74,45	6,92	43	-4,809 .000*
	Kontrol grubu	24	65,09	6,01		

\*P<0.05

Deney ve kontrol grubuna uygulama sonrası yapılan son testlerde  $*p < .05$  anlamlılık düzeyi için  $.00 < .05$  olduğu için anlamlıdır. Yapılan son testlerde (deney grubu son test ortalaması  $\bar{X} = 74,45$ ; kontrol grubu son test ortalaması  $\bar{X} = 65,09$ ) deney grubunun son test puanları kontrol grubunun son test puanlarından daha yüksek olduğu belirlenmiştir (Tablo 4. 13). Elde edilen bu sonuç uygulamanın deney grubu yararına olduğunu göstermektedir. Ayrıca, bilgisayarlı kodlama ortamının, Bilgisayar Destekli Eğitime İlişkin Tutumları üzerindeki etki büyüklüğünü

belirlemek için eta kare değeri incelenmiştir. Etki büyüklüğü değerleri  $\eta^2 = .129$  olarak hesaplanmıştır. Bu durumda, etki büyüklüğü değeri ( $\eta^2 = 0.129$ ) göz önünde bulundurulduğunda, bilgisayarlı kodlama ortamının, Bilgisayar Destekli Eğitime İlişkin Tutumları üzerinde “geniş” bir etki büyüklüğüne sahip olduğu söylenebilir.

**Deney-kontrol grubu Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeği son testler karşılaştırması (bağımsız t testi)**

Bilgisayarlı kodlamada Scratch uygulamasını kullanan öğrenciler (deney grubu) ile Tospaa uygulamasını kullanan öğrencilerin (kontrol grubu), “Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutumları” karşılaştırıldığında ortaya çıkan sonuç Tablo 4. 14’de verilmiştir.

**Tablo 4. 14** Eğitsel bilgisayar oyunları destekli kodlama öğrenimine yönelik tutum ölçeği gruplar arası (deney - kontrol grubu) son test karşılaştırma (t - testi) sonuçları

Gruplar	N	$\bar{X}$	S	Sd	t	p
Son test	Deney grubu	21	95,54	9,53	43	-2,805 .008*
	Kontrol grubu	24	85,57	14,13		

\*P<0.05

Deney ve kontrol grubuna uygulama sonrası yapılan son testlerde  $*p < .05$  anlamlılık düzeyi için  $.00 < .05$  olduğu için anlamlıdır. Yapılan son testlerde (deney grubu son test ortalaması  $\bar{X} = 95,54$ ; kontrol grubu son test ortalaması  $\bar{X} = 85,57$ ) deney grubunun son test puanları kontrol grubunun son test puanlarından daha yüksek olduğu belirlenmiştir (Tablo 4. 14). Elde edilen bu sonuç uygulamanın deney grubu yararına olduğunu göstermektedir. Ayrıca, Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeği toplam üzerindeki etki büyüklüğünü belirlemek için eta kare değeri incelenmiştir. Etki büyüklüğü değerleri  $\eta^2 = .134$  olarak hesaplanmıştır. Bu durumda, etki büyüklüğü değeri ( $\eta^2 = 0.134$ ) göz önünde bulundurulduğunda, bilgisayarlı kodlama ortamının, Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutumları üzerinde “geniş” bir etki büyüklüğüne sahip olduğu söylenebilir.

## BÖLÜM 5

### 5 TARTIŞMA, SONUÇ VE ÖNERİLER

#### 5.1 Sonuç ve Tartışma

Araştırma kapsamında çalışma grubunu oluşturan 45 adet ortaokul öğrencisine ait uygulama süreci öncesinde uygulanan demografik bilgi formları, Akademik başarı testi, Bilgisayar destekli eğitime ilişkin öz-yeterlik algı ölçeği, bilgisayar destekli eğitime ilişkin tutum ölçeği, eğitsel bilgisayar oyunları destekli kodlama öğrenimine yönelik tutum ölçeklerine çalışma grubunun vermiş olduğu cevaplar incelendiğinde elde edilen bilgilere göre sonuçlara ulaşılmıştır.

Araştırma kapsamında ele alınan 45 adet ortaokul 6. sınıf öğrencisine uygulanan demografik bilgiler incelendiğinde elde edilen bilgilere göre sonuçlara ulaşılmıştır. Araştırmanın çalışma grubunda ki öğrencilerden; kadın öğrenciler N=17 yani % 37.7'sini, erkek öğrenciler N= 28 yani % 62.3'ünü oluşturmaktadır. Araştırmanın deney grubunda ki öğrencilerden; kadın öğrenciler N= 7 yani %29.1'ini, erkek öğrenciler N= 14 yani %69.9'unu oluşturmaktadır. Araştırmanın kontrol grubunda ki öğrencilerden; kadın öğrenciler N= 10 yani %41.6'sını, erkek öğrenciler N= 14 yani %58.4'ünü oluşturmaktadır.

Çalışmada kullanılmış çoktan seçmeli test olan Scratch Akademik Başarı Testi (EK 4) ve Tospaa Akademik Başarı (EK 5) testi, her iki test için Kr-21 değerlerine bakıldığında Scratch Akademik Başarı Testinin güçlük indisi = .758 ve Tospaa Akademik Başarı testinin güçlük indisi = .726 olarak bulunmuştur.

Deney ve kontrol grubuna uygulama öncesi (ön testler) uygulanan akademik başarı testi, bilgisayar destekli eğitime ilişkin öz-yeterlik algı ölçeği, bilgisayar destekli eğitime ilişkin tutum ölçeği ve eğitsel bilgisayar oyunları destekli kodlama öğrenimine yönelik tutum ölçeği sonuçlarının karşılaştırma (ilişkisiz örneklemeler için bağımsız t-testi) sonuçlarında öğrencilerin puanları arasında anlamlı bir farklılık gözlemlenmemiştir. Yapılan ön testler ile çalışma grubu deney ve kontrol grubu sayılarının yaklaşık olarak benzer olmasından ve yapılan istatistik testlerinin ön test sonuçlarının karşılaştırmasından (Bilgisayar destekli eğitime ilişkin öz-yeterlik algı ölçeği, bilgisayar destekli eğitime ilişkin tutum ölçeği, eğitsel bilgisayar oyunları destekli kodlama öğrenimine yönelik tutum ölçeği) bulunan bu sonuçlarla her iki grup

içinde (deney ve kontrol grubu) araştırma öncesi denk olduğu ve araştırmanın uygulaması için uygun olduğu kanısına varılmıştır.

Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algı Ölçeği uygulama sonrasında çalışma grubundan toplanan veriler ile yapılan güvenilirlik testinde iç güvenilirlik katsayısı olan Cronbach's Alfa güvenilirlik katsayısı .870 olarak bulunmuştur. Bilgisayar Destekli Eğitime İlişkin Tutum Ölçeğine yönelik uygulama sonrasında çalışma grubundan toplanan veriler ile yapılan güvenilirlik testinde iç güvenilirlik katsayısı olan Cronbach's Alfa güvenilirlik katsayısı .81 olarak bulunmuştur. Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeğine yönelik uygulama sonrasında çalışma grubundan toplanan veriler ile yapılan güvenilirlik testinde iç güvenilirlik katsayısı olan Cronbach's Alfa güvenilirlik katsayısı .92 olarak bulunmuştur.

Deney ve kontrol grubuna uygulama sonrası yapılan son testlerde  $*p < .05$  anlamlılık düzeyi için  $.00 < .05$  olduğu için anlamlıdır. Yapılan son testlerde (deney grubu son test ortalaması  $\bar{X} = 78,19$ ; kontrol grubu son test ortalaması  $\bar{X} = 73,79$ ) deney grubunun son test puanları kontrol grubunun son test puanlarından daha yüksek olduğu belirlenmiştir. Elde edilen sonuç uygulamanın Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algılarına yönelik deney grubunun lehine olduğunu göstermektedir. Ayrıca, bilgisayarlı kodlama ortamının, Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algı Ölçeği üzerindeki etki büyüklüğünü belirlemek için eta kare değeri incelenmiştir. Etki büyüklüğü değerleri  $\eta^2 = .133$  olarak hesaplanmıştır. Bu durumda, etki büyüklüğü değeri ( $\eta^2 = 0.133$ ) göz önünde bulundurulduğunda, bilgisayarlı kodlama ortamının, Bilgisayar Destekli Eğitime İlişkin Öz-yeterlik Algıları üzerinde “geniş” bir etki büyüklüğüne sahip olduğu söylenebilir.

Deney ve kontrol grubuna uygulama sonrası yapılan son testlerde  $*p < .05$  anlamlılık düzeyi için  $.00 < .05$  olduğu için anlamlıdır. Yapılan son testlerde (deney grubu son test ortalaması  $\bar{X} = 74,45$ ; kontrol grubu son test ortalaması  $\bar{X} = 65,09$ ) deney grubunun son test puanları kontrol grubunun son test puanlarından daha yüksek olduğu belirlenmiştir. Elde edilen bu sonuç uygulamanın deney grubu yararına olduğunu göstermektedir. Ayrıca, bilgisayarlı kodlama ortamının, Bilgisayar Destekli Eğitime İlişkin Tutumları üzerindeki etki büyüklüğünü belirlemek için eta kare değeri incelenmiştir. Etki büyüklüğü değerleri  $\eta^2 = .129$  olarak

hesaplanmıştır. Bu durumda, etki büyüklüğü değeri ( $\eta^2= 0.129$ ) göz önünde bulundurulduğunda, bilgisayarlı kodlama ortamının, Bilgisayar Destekli Eğitime İlişkin Tutumları üzerinde “geniş” bir etki büyüklüğüne sahip olduğu söylenebilir.

Deney ve kontrol grubuna uygulama sonrası yapılan son testlerde  $*p<.05$  anlamlılık düzeyi için  $.00 <.05$  olduğu için anlamlıdır. Yapılan son testlerde (deney grubu son test ortalaması  $\bar{X}=95,54$ ; kontrol grubu son test ortalaması  $\bar{X}=85,57$ ) deney grubunun son test puanları kontrol grubunun son test puanlarından daha yüksek olduğu belirlenmiştir. Elde edilen bu sonuç uygulamanın deney grubu yararına olduğunu göstermektedir. Ayrıca, Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeği toplam üzerindeki etki büyüklüğünü belirlemek için eta kare değeri incelenmiştir. Etki büyüklüğü değerleri  $\eta^2= .134$  olarak hesaplanmıştır. Bu durumda, etki büyüklüğü değeri ( $\eta^2= 0.134$ ) göz önünde bulundurulduğunda, bilgisayarlı kodlama ortamının, Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutumları üzerinde “geniş” bir etki büyüklüğüne sahip olduğu söylenebilir.

Tüm bu sonuçlar bağlamında Scratch uygulamasının bilgisayarlı ortamda yapılmış olması öğrencileri öğrenmeye teşvik etmiştir. Öğrenciler blok tabanlı kodlama araçlarından birisi olan Scratch’i kullanarak hem bilgisayar uygulamalı bir kodlama eğitimi almışlar, öğrenciler çalışmaları daha aktif takip etmişlerdir. Yapılan araştırmalar ve elde edilen veriler biz göstermektedir ki bilişim teknolojileri ve yazılım dersinde bilgisayarlı kodlama için okullarda bilgisayar sınıfları gibi uygun ortamların oluşturulması gerekmektedir. Bilişim Teknolojileri ve Yazılım dersi öğretmenlerinin ise bilgisayarlı kodlama eğitimini öğrencileri için farklı portalları kullanarak vermelerinin öğrencilerin bilişsel ve algoritmik düşünme becerilerini geliştireceği öngörüsünde bulunabiliriz. Tüm bunlara ek olarak eğitim öğretimde bilişim teknolojileri ders saatinin arttırılması ve hafta sonu kursların istenildiği tüm okullarda öğrenciler için açılmasının öğrencilerin faydasına olacağı düşünülmektedir

## 5.2 Öneriler

Bu bölümde, çalışmanın sonucunda elde edilen sonuçlara dayanarak gelecek çalışmalara yardımcı olabilmesi açısından oluşturulmuştur.

1. Bu çalışmada elde edilen veriler 10 haftalık sınırlı süre ve sınırlı bir çalışma grubunda gerçekleştirilen uygulama sonucunda elde edilmiştir. Böyle bir çalışmanın katılımcı sayısı fazla bir çalışma grubu ile uzun süreli bir uygulama ile yapılmasının elde edilecek sonuçların daha da genellenebilir olmasını sağlayabilir.
2. Yapılan çalışma Covid-19 salgını nedeni ile uzaktan eğitim ortamında gerçekleştirilmiştir. Salgın süreci sona erdikten sonra çalışmanın yüz yüze eğitim ortamında da yapılarak çıkacak sonuçlar incelenebilir.
3. Araştırmaya 6. sınıf öğrencileri üzerinde yapılmıştır, benzer bir çalışmada farklı sınıf düzeylerinden öğrenciler ve farklı bir blok kodlama aracı üzerinde yapılabilir.
4. Bu araştırmadan elde edilen bulgular sonucunda bilgisayarlı ortamda yapılan eğitim öğretimin öğrenciler üzerinde olumlu bir etki yarattı kanısına varılmış, bilgisayar destekli öğretimlerin yaygınlaştırılması için okul içi ve okul dışında öğrencileri teşvik edici faaliyetler düzenlenebilir. Bunun için öğrencilerin kodlama haftasına (Codeweek) katılmaları öğrencilerin kodlama etkinliklerinde; analitik düşünme ve problem çözme becerilerinin geliştirilmesi, web 2.0 araçlarıyla oyun kurabilmeleri, motivasyon ve farkındalıklarını artırabilmeleri sağlanıp kodlamanın önemi konusunda farkındalık yaratılabilir.
5. Scratch yazılımıyla yapılan bu çalışma ortaokul 6.sınıf öğrencileriyle yürütülmüştür. Edinilen bulgular Scratch ile kodlama öğretiminin akademik başarıyı etkilediği yönünde olmuştur. Bundan dolayı her yaş için yaratıcılığı destekleyen ve geliştiren bir yazılım olan Scratch'in anaokulundan yükseköğretime kadar her kademedeki kullanılabileceği söylenebilir.
6. Kodlama eğitiminin amacına uygun bir şekilde yapılabilmesi için okullarda kodlama sınıfları kurulabilir. Öğrenciler de bu sınıf ortamlarında yaparak yaşayarak öğrenme deneyimleri yaşayabilirler.

## KAYNAKÇA

- Adams, J. C. (2010). *Scratching Middle Schoolers' Creative Itch*. Paper presented at the Proceedings of the 41st ACM technical symposium on Computer science education, 356-360.
- Akgün, K., & Türel, Y. K. (2021). Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümü Öğrencilerinin Stem Yaklaşımına Yönelik Farkındalıklarının Belirlenmesi. *Eğitim Teknolojisi Kuram ve Uygulama*, 11(1), 116-128.
- Al, U., & Madran, O. (2004). Web Tabanlı Uzaktan Eğitim Sistemleri: Sahip Olması Gereken Özellikler ve Standartlar.
- Arıcı, N., & Demir, C. (2009). Okul Öncesi Çocukları için İngilizce Kelime Eğitim Programı.
- Arslan. (2006a). Bilgisayar Destekli Eğitime İlişkin Öz Yeterlilik Algısı Ölçeği. *Abant İzzet Baysal Üniversitesi Eğitim Fakültesi Dergisi*, 3(2), 24-33.
- Arslan. (2006b). Bilgisayar Destekli Eğitim Yapmaya İlişkin Tutum Ölçeği. *Yüzüncü Yıl Üniversitesi Eğitim Fakültesi Dergisi*.
- Arslan. (2007). Eğitimde Yapılandırmacı Yaklaşımlar. . *Ankara Üniversitesi Eğitim Bilimleri Fakültesi Dergisi*, 40 (1), 41-61.
- Aslan, D., & Aydın, H. (2016). Yapılandırmacı Öğretim Kuramın Felsefi Paradigmaları: Bir Derleme Çalışması. *Uşak Üniversitesi Eğitim Araştırmaları Dergisi*, 2(2), 56-71.
- Aydeniz, M., & Bilican, K. (2018). STEM Eğitiminde Global Gelişmeler ve Türkiye için Çıkarımlar. *Pegem Atıf İndeksi*, 69-92.
- Aytekin, A., Çakır, F. S., Yücel, Y. B., & Kulaöz, İ. (2018). Geleceğe Yön Veren Kodlama Bilimi ve Kodlama Öğrenmede Kullanılabilecek Bazı Yöntemler. *Avrasya Sosyal ve Ekonomi Araştırmaları Dergisi*, 5(5), 24-41.
- Baş, G. (2011). İlköğretim Öğretmenlerinin Eğitsel İnternet Kullanımı Öz- Yeterlilik İnançlarının Farklı Değişkenler Açısından İncelenmesi. *Eğitim Teknolojisi Kuram ve Uygulama*, 1(2), 35-51.
- Batdal, G. (2005). Öğrenci Odaklı Bir Yaklaşımla İlköğretim Matematik Programlarının Değerlendirilmesi. *XIV. Ulusal Eğitim Bilimleri Kongresi Kitabı*, 343-346.
- Baz, F. Ç. (2018). Çocuklar için Kodlama Yazılımları Üzerine Karşılaştırmalı Bir İnceleme. *Current Research in Education*, 4(1), 36-47.
- Blocklygames. (2020). blocklygames. Retrieved from <https://blockly.games/>, Erişim Tarihi: 15.12.2020.
- Bolat, D., Korkmaz, Ö., & Çakır, R. (2020). Ortaokul Öğretmenlerinin Bilişim Teknolojilerini Kullanım ve Derslerine Entegre Edebilme Düzeylerinin Belirlenmesi. *Ahmet Keleşoğlu Eğitim Fakültesi Dergisi*, 2(2), 229-250.
- Boyunsuz, N. (2021). *Yenilenen Eğitim Fakültesi Öğretmenlik Programlarının Stem Okuryazarı Öğretmenleri Yetiştirmesi Açısından İncelenmesi*, Yüksek Lisans Tezi, Necmettin Erbakan Üniversitesi Eğitim Bilimleri Enstitüsü.
- Bozkurt, A. (2014). Homo ludens: Dijital Oyunlar ve Eğitim. *Eğitim Teknolojileri Araştırmaları Dergisi*, 5(1), 1-21.
- Bozkurt, A. (2016). *Bağlantıcı Kitlese Açık Çevrimiçi Derslerde Etkileşim Örüntüleri ve Öğreten Öğrenen Rollerinin Belirlenmesi*, Doktora Tezi, Anadolu Üniversitesi, Sosyal Bilimler Enstitüsü, Uzaktan Eğitim Anabilim Dalı, Eskişehir.
- Bozkurt, A. (2017). Türkiye'de Uzaktan Eğitimin Dünü, Bugünü ve Yarını. *Açıköğretim Uygulamaları ve Araştırmaları Dergisi*, 3(2), 85-124.
- Brennan, K., & Resnick, M. (2013). *Stories From the Scratch Community: Connecting With Ideas, Interests, and People*. Paper presented at the Proceeding of the 44th ACM technical symposium on Computer science education, 463-464.

- BtDersleri. (2020). BtDersleri. Retrieved from <https://www.btdersleri.com/ders/kodlama-e%C4%9Fitimi-nedir-kodlama-dersinin-amac%C4%B1>, Erişim Tarihi: 13.12.2020.
- Büyüköztürk, Ş. (2011). Sosyal Bilimler İçin Veri Analizi El Kitabı (17. basım). Ankara: Pegem Akademi.
- Büyüköztürk, Ş., Çakmak, E., Akgün, E., Karadeniz, Ş., & Demirel, F. (2012). Bilimsel Araştırma Yöntemleri.(12. Basım). Ankara: Pegem Akademi.
- Calder, N. (2010). Using Scratch: an Integrated Problem-Solving Approach to Mathematical Thinking. *Australian Primary Mathematics Classroom*, 15(4), 9-14.
- Choi, J., Lee, Y., & Lee, E. (2017). Puzzle Based Algorithm Learning for Cultivating Computational Thinking. *Wireless Personal Communications*, 93(1), 131-145.
- Computer, C. w. a. (2020). ComputerScience without a computer. Retrieved from <https://csunplugged.org/en/>, Erişim Tarihi: 15.12.2020.
- Coravu, L., Marian, M., & Ganea, E. (2015). *Scratch and recreational coding for kids*. Paper presented at the 2015 14th RoEduNet International Conference-Networking in Education and Research (RoEduNet NER), 85-89.
- Coşar, M. (2013). Problem Temelli Öğrenme Ortamında Bilgisayar Programlama Çalışmalarının Akademik Başarı, Eleştirel Düşünme Eğilimi ve Bilgisayara Yönelik Tutuma Etkileri. *Gazi Üniversitesi, Ankara*.
- Creswell, J. W., & Clark, V. L. P. (2017). *Designing and Conducting Mixed Methods Research*: Sage publications.
- Çavaş, P., Ayar, A., & Gürcan, G. (2020). Türkiye’de STEM Eğitimi Üzerine Yapılan Araştırmaların Durumu Üzerine Bir Çalışma. *Yüzüncü Yıl Üniversitesi Eğitim Fakültesi Dergisi*, 17(1), 823-854.
- Çolak, M. (2018). *Ortaokul Fen Bilimleri Dersinin 21. Yüzyıl Becerilerini Kazandırmadaki Etkililiğine İlişkin Öğretmen Görüşleri (Kayseri İli Örneği)*, Yüksek Lisans Tezi, Erciyes Üniversitesi, Kayseri.
- Demir, G., & Seferoğlu, S. S. (2017). Yeni Kavramlar, Farklı Kullanımlar: Bilgi-İşlemsel Düşünmeyle İlgili Bir Değerlendirme. *Eğitim teknolojileri okumaları*, 801-830.
- Demirer, V., & Nurcan, S. (2016). Programming Education and New Approaches Around The World and İn Turkey/Dünyada ve Türkiye’de Programlama Eğitimi ve Yeni Yaklaşımlar *Eğitimde Kuram ve Uygulama*, 12(3), 521-546.
- Dinçer, S. (2016). Bilgisayar Destekli Uzaktan Eğitim ve Uzaktan Eğitime Genel Bir Bakış. *Adana, Seyhan, Türkiye*.
- Duman, B. (2007). *Eğitimde Çağdaş Yaklaşımlar*. Ankara: Pegem Akademi, 283-391.
- Duru, S. (2014). Yapılandırmacı ve Geleneksel Öğrenme Ortamlarının Öğretmen Adaylarının Eğitim İnançları Üzerine Etkisi.
- Eba. (2020). Eba. Retrieved from [www.eba.gov.tr/kod](http://www.eba.gov.tr/kod), Erişim Tarihi: 13.12.2020.
- Ercan, İ., & Kan, İ. (2004). Ölçeklerde Güvenirlik ve Geçerlik.
- Ergin, Y. D. (1995). 1. Ölçeklerde Geçerlik ve Güvenirlik.
- Erol, O. (2015). *Scratch İle Programlama Öğretiminin Bilişim Teknolojileri Öğretmen Adaylarının Motivasyon ve Başarılarına Etkisi*, Doktora Tezi, Anadolu Üniversitesi.
- Ertmer, P. A., & Ottenbreit-Leftwich, A. T. (2010). Teacher Technology Change: How Knowledge, Confidence, Beliefs, and Culture Intersect. *Journal of research on Technology in Education*, 42(3), 255-284.
- EuropeanSchoolnet. (European Schoolnet). Retrieved from [http://fcl.eun.org/documents/10180/14689/Computing+our+future\\_final.pdf/746e36](http://fcl.eun.org/documents/10180/14689/Computing+our+future_final.pdf/746e36), Erişim Tarihi: 13.12.2020.
- Ferrer-Mico, T., Prats-Fernández, M. À., & Redo-Sanchez, A. (2012). Impact of Scratch Programming on Students’ Understanding of Their Own Learning Process. *Procedia-Social and Behavioral Sciences*, 46, 1219-1223.

- Geçer, A. K., & Funda, D. (2010). Bilgisayar Okur-Yazarlık Düzeylerinin Belirlenmesi: Kocaeli Üniversitesi Örneği (ss, 20-44). *Yüzüncü Yıl Üniversitesi Eğitim Fakültesi Dergisi*, 7(1), 20-44.
- Genç, Z., & Karakuş, S. Tasarımla Öğrenme: Eğitsel Bilgisayar Oyunları Tasarımında Scratch Kullanımı.
- Gibbons, A. S., Nelson, J., & Richards, R. (2000). The Architecture of Instructional Simulation: A Design for Tool Construction. *Center for Human-System Simulation Technical Report, Idaho Falls, ID: Idaho National Engineering and Environmental Laboratory*. Retrieved from [http://webpub.byu.net/asg33/ineel\\_simulation\\_paper.pdf](http://webpub.byu.net/asg33/ineel_simulation_paper.pdf).
- Gomes, A., & Mendes, A. J. (2007). *Learning to Program-Difficulties and Solutions*. Paper presented at the International Conference on Engineering Education–ICEE.
- Gülbahar, Y. (2020). *Bilgi İşlemsel Düşünmeden Programlamaya* (Y. Gülbahar Ed.). Pegem Akademi.
- Gülgün, C., Yılmaz, A., & Çağlar, A. (2017). Fen Bilimleri Dersinde Uygulanan STEM Etkinliklerinde Bulunması Gereken Nitelikler Hakkında Öğretmen Görüşleri. *Journal of Current Researches on Social Sciences*, 7(1), 459-478.
- Kahraman, M. E. (2020). COVID-19 Salgınının Uygulamalı Derslere Etkisi ve Bu Derslerin Uzaktan Eğitimle Yürütülmesi: Temel Tasarım Dersi Örneği. *Medeniyet Sanat Dergisi*, 6(1), 44-56.
- Kalelioglu, F., & 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.
- Karabak, D., & Güneş, A. (2013). Ortaokul Birinci Sınıf Öğrencileri için Yazılım Geliştirme Alanında Müfredat Önerisi. *Eğitim ve Öğretim Araştırmaları Dergisi*, 21(2-3), 163-169.
- Karasar, N. (1999). Bilimsel Araştırma Yöntemi. *Ankara: Nobel Yayın Dağıtım*.
- Keçeci, G., Alan, B., & Zengin, F. K. (2016). Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeği: Geçerlilik ve Güvenirlik Çalışması. *Education Sciences*, 11(3), 184-194.
- Kert, S. B., & Uğraş, T. (2009). *Programlama Eğitiminde Sadelik ve Eğlence: Scratch Örneği*. Paper presented at the The First International Congress of Educational Research, Çanakkale, Turkey.
- Keskinsoy, A. (2010). Mesleki Liselerde Görsel Programlama Başarısını Etkileyen Faktörler. *Yayımlanmamış Yüksek Lisans Tezi*. Marmara Üniversitesi, İstanbul.
- Kırkıç, K. A., & Aydın, E. (2018). *Merhaba STEM Yenilikçi Bir Öğretim Yaklaşımı: Eğitim Yayınevi*.
- Kobsiripat, W. (2015). Effects of The Media to Promote the Scratch Programming Capabilities Creativity of Elementary School Students. *Procedia-Social and Behavioral Sciences*, 174, 227-232.
- Koçin, E. (2020). *Algoritma ve Kodlama Eğitiminin Sınıf Öğretmeni Adaylarının Kodlama Başarısına ve Kodlamaya İlişkin Özyeterlik Algısına Etkisi Yüksek Lisans Tezi*. Çanakkale Onsekiz Mart Üniversitesi.
- Kodlamanisa. (2020). Bilgisayarsız Kodlama. Retrieved from <https://www.kodlamanisa.gov.tr/unplugged-kodlama/>, Erişim Tarihi: 15.12.2020.
- Koehler, M. J., Mishra, P., & Yahya, K. (2007). Tracing the Development of Teacher Knowledge in A Design Seminar: Integrating Content, Pedagogy and Technology. *Computers & Education*, 49(3), 740-762.
- Kordaki, M. (2012). Diverse Categories of Programming Learning Activities Could Be Performed Within Scratch. *Procedia-Social and Behavioral Sciences*, 46, 1162-1166.
- Korucu, A. T., & Çakır, H. (2015). Dinamik Web Teknolojileri ile Geliştirilen İşbirlikli Öğrenme Ortamını Kullanan Öğretmen Adaylarının Görüşleri. *Adıyaman Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, 2015(19), 221-254.

- Kurt, A. A., Günüş, S., & Ersoy, M. (2013). Dijitalleşmede Son Durum: Dijital Yerli, Dijital Göçmen ve Dijital Göçebe. *Ankara Üniversitesi Eğitim Bilimleri Fakültesi Dergisi*, 46(1), 1-22.
- Kurudayıođlu, M., & Tüzeli, A. G. M. S. (2010). 21. Yüzyıl Okuryazarlık Türleri, Deđişen Metin Algısı ve Türkçe Eğitimi. *Türklük Bilimi Araştırmaları*(28).
- Lawanto, K., Close, K., Ames, C., & Brasiel, S. (2017). Exploring Strengths and Weaknesses in Middle School Students' Computational Thinking in Scratch *Emerging Research, Practice, and Policy on Computational Thinking* (pp. 307-326): Springer.
- Lee, Y.-J. (2011). Scratch: Multimedia Programming Environment for Young Gifted Learners. *Gifted Child Today*, 34(2), 26-31.
- M.E.B. (2012). Bilişim Teknolojileri ve Yazılım Dersi (5,6.7 ve 8. Sınıflar ) Öğretim Programı. *Milli Eğitim Bakanlığı*.
- Malan, D. J., & Leitner, H. H. (2007). Scratch for Budding Computer Scientists. *ACM Sigcse Bulletin*, 39(1), 223-227.
- 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), 1-15.
- Meerbaum-Salant, O., Armoni, M., & Ben-Ari, M. (2011). *Habits of Programming in Scratch*. Paper presented at the Proceedings of the 16th annual joint conference on Innovation and technology in computer science education, 168-172.
- Microsoft. (2020). How does computer science (CS) relate to STEM. Retrieved from <https://www.microsoft.com/en-us/corporate-responsibility/skills-employability/computer-science>, Erişim Tarihi: 14.12.2020.
- Miles, M. B., Huberman, A. M., Huberman, M. A., & Huberman, M. (1994). *Qualitative Data Analysis: An Expanded Sourcebook*: sage.
- Moore, M. G. (1973). Toward a Theory of Independent Learning and Teaching. *The Journal of Higher Education*, 44(9), 661-679.
- Nam, D., Kim, Y., & Lee, T. (2010). *The Effects of Scaffolding-Based Courseware for the Scratch Programming Learning on Student Problem Solving Skill*. Paper presented at the Proceedings of the 18th International Conference on Computers in Education, 723-727.
- Nezih, Ö., & Çakır, H. (2016). Ortaokul Matematik Öğretmenlerinin Matematik Öğretiminde Bilişim Teknolojileri Kullanımına İlişkin Görüşleri. *Mersin Üniversitesi Eğitim Fakültesi Dergisi*, 12(1).
- Odabaş, H. (2003). İnternet Tabanlı Uzaktan Eğitim ve Bilgi ve Belge Yönetimi. *Türk Kütüphaneciliđi*, 17(1), 22-36.
- Özmen, H. (2004). Fen Öğretiminde Öğrenme Teorileri ve Teknoloji Destekli Yapılandırmacı (Constructivist) Öğrenme. *The Turkish Online Journal of Educational Technology*, 3(1), 100-111.
- Öztürk, Ö. K., & Tetik, E. (2015). Sosyal Ağ Destekli Bilişim Teknolojileri Eğitiminin Öğrencilerin Akademik Başarılarına Etkisi *Education Sciences*, 10(3), 151-168.
- Pink, D. H., & Pink, D. H. (2005). *A Whole New Mind: Moving From the Information Age to the Conceptual Age* (Vol. 50): Riverhead Books New York.
- Prensky, M. (2001a). Fun, Play And Games: What Makes Games Engaging. *Digital game-based learning*, 5(1), 5-31.
- Prensky, M. (2001b). Nativos Digitais, İmigrantes Digitais. *On the horizon*, 9(5), 1-6.
- Prensky, M. (2009). H. Sapiens Digital: From Digital İmmigrants and Digital Natives to Digital Wisdom. *Innovate: journal of online education*, 5(3).
- Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., . . . Silverman, B. (2009). Scratch: Programming for All. *Communications of the ACM*, 52(11), 60-67.

- Resnick, M., & Silverman, B. (2005). *Some Reflections on Designing Construction Kits for Kids*. Paper presented at the Proceedings of the 2005 conference on Interaction design and children, 117-122.
- Robins, A., Rountree, J., & Rountree, N. (2003). Learning and Teaching Programming: A Review and Discussion. *Computer science education*, 13(2), 137-172.
- Roy-Singh, R. (1991). *Education for the Twenty-First Century: Asia-Pacific Perspectives*: Unesco Principal Regional Office for Asia and the Pacific.
- Sade, A. (2020). *Kodlama Öğretiminin 6. Sınıf Öğrencilerinin Bilgisayarca Düşünme Becerilerine, Matematik Kaygı Algılarına ve Problem Çözme Algılarına Etkisi, Yüksek Lisans Tezi*, Mersin Üniversitesi Eğitim Bilimleri Enstitüsü.
- Samancıoğlu, M., & Summak, M. S. (2014). Öğretmenlerin Derslerde Teknoloji Kullanımlarını Etkileyen Faktörler: Kişisel Bilgisayar Kullanımı ve Öğretim Yaklaşımları. *Atatürk Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, 18(2), 195-207.
- Sarıtepeci, M., Durak, H., & Seferoğlu, S. S. (2016). Öğretmenlerin Öğretim Teknolojileri Alanında Hizmet-içi Eğitim Gereksinimlerinin FATİH Projesi Kapsamında İncelenmesi 1. *Turkish Journal of Computer and Mathematics Education*, 7(3), 601.
- Saygıner, Ş., & Tüzün, H. (2017). Programlama eğitiminde yaşanan zorluklar ve çözüm önerileri.
- Sayın, Z., & Seferoğlu, S. S. (2016). Yeni Bir 21. Yüzyıl Becerisi Olarak Kodlama Eğitimi ve Kodlamanın Eğitim Politikalarına Etkisi. *Akademik Bilişim Konferansı*, 3-5.
- Scratch, A. (2020). About Scratch. Retrieved from <https://scratch.mit.edu/about>, Erişim Tarihi: 06.12.2020.
- Sırakaya, M. (2018). Kodlama Eğitimine Yönelik Öğrenci Görüşleri. *Ondokuz Mayıs University Journal of Education*, 37(2).
- Somyürek, S. (2014). Öğretim Sürecinde Z Kuşağının Dikkatini Çekme: Artırılmış Gerçeklik. *Eğitim Teknolojisi Kuram ve Uygulama*, 4(1), 63-80.
- Strauss, A., & Corbin, J. (1998). *Basics of Qualitative Research Techniques*: Sage publications Thousand Oaks, CA.
- Tan, Ş. (2009). *Öğretim İlke ve Yöntemleri*. Pegem Yayıncılık 4. Baskı Ankara.
- Tarım, K., & Akdeniz, F. (2003). İlköğretim Matematik Derslerinde Kubaşık Öğrenme Yönteminin Kullanılması. *Hacettepe Üniversitesi Eğitim Fakültesi Dergisi*, 2003(24), 215-223.
- Taylor, M., Harlow, A., & Forret, M. (2010). Using a Computer Programming Environment and an Interactive Whiteboard to Investigate Some Mathematical Thinking. *Procedia-Social and Behavioral Sciences*, 8, 561-570.
- Tezci, E. (2016). Öğretmenlerin BİT Entegrasyon Yaklaşımlarının Ölçülmesine Yönelik Ölçek Geliştirme. *Kastamonu Eğitim Dergisi*, 24(2), 975-992.
- Tezel, Ö., & Yaman, H. (2017). FeTeMM Eğitimine Yönelik Türkiye’de Yapılan Çalışmalardan Bir Derleme.
- Thomas, J., Odenwingie, O., Saunders, Q., & Watlerd, M. (2015). While Enacting Computational Algorithmic Thinking in the Context of Game Design. *Journal of Computer Science and Information Technology*, 3(1).
- Tim Bell, I. H. W. a. M. F. (2015). Unplugged, [https://classic.csunplugged.org/wp-content/uploads/2015/03/CSUnplugged\\_OS\\_2015\\_v3.1.pdf](https://classic.csunplugged.org/wp-content/uploads/2015/03/CSUnplugged_OS_2015_v3.1.pdf), Erişim Tarihi: 06.12.2020.
- Tinio, V. L. (2003). ICT in Education: e-ASEAN Task Force.
- Tospaa. (2020). Tospaa Bilgisayarsız Kodlama Oyunu. Retrieved from <http://tospaa.org/>, Erişim tarihi: 03.09.2019.
- Tsai, M.-J., & Tsai, C.-C. (2003). Information Searching Strategies in Web-Based Science Learning: The Role Of Internet Self-Efficacy. *Innovations in education and Teaching International*, 40(1), 43-50.
- TTKB. (2012a). *Bilişim Teknolojileri ve Yazılım Dersi Öğretim Programı*. Milli Eğitim Bakanlığı. Ankara.

- TTKB. (2012b). Bilişim Teknolojileri Ve Yazılım Dersi Öğretim Programı. Retrieved from [https://www.bilgisayarbilisim.net/attachments/bilisimteknolojileri\\_ortaokul-rar.29815/](https://www.bilgisayarbilisim.net/attachments/bilisimteknolojileri_ortaokul-rar.29815/)
- TTKB. (2017). *Bilişim Teknolojileri ve Yazılım Dersi Öğretim Programı*. Milli Eğitim Bakanlığı. Ankara.
- Uğurlu, C. T. (2009). İlköğretim Birinci Sınıf Öğretmenlerinin Yapılandırmacı Öğrenme Yaklaşımı ile İlk Okuma Yazma Öğretimine İlişkin Görüşleri. *Elektronik Sosyal Bilimler Dergisi*, 8(30), 103-114.
- Uluyol, Ç., & Eryılmaz, S. (2015). 21. Yüzyıl Becerileri Işığında FATİH Projesi Değerlendirmesi. *Gazi Üniversitesi Gazi Eğitim Fakültesi Dergisi*, 35(2), 209-229.
- Ünsal, K. (2019). *Ortaokul ve Lise Yöneticilerinin Kodlama Eğitimine Yönelik Görüşlerinin İncelenmesi (Bağcılar İlçesi Örneği)*, Yüksek Lisans Tezi, Marmara Üniversitesi.
- Van-Roy, P., & Haridi, S. (2004). *Concepts, Techniques, and Models of Computer Programming*: MIT press.
- Wachira, P., & Keengwe, J. (2011). Technology İntegration Barriers: Urban School Mathematics Teachers Perspectives. *Journal of science education and technology*, 20(1), 17-25.
- Wilson, A., & Moffat, D. C. (2010). *Evaluating Scratch to Introduce Younger Schoolchildren to Programming*. Paper presented at the PPIG.
- Windschitl, M., & Sahl, K. (2002). Tracing Teachers' Use of Technology in a Laptop Computer School: the İnterplay of Teacher Beliefs, Social Dynamics, and İstitutional Culture. *American educational research journal*, 39(1), 165-205.
- Yiğit, M. F. (2016). *Görsel Programlama Ortamı ile Öğretimin Öğrencilerin Bilgisayar Programlamayı Öğrenmesine ve Programlamaya Karşı Tutumlarına Etkisinin İncelenmesi*, Yüksek Lisans Tezi. Ondokuz Mayıs Üniversitesi.
- Yıldırım, A., & Simsek, H. (2008). Sosyal Bilimlerde Nitel Araştırma Yöntemleri. *Ankara: Seckin*.
- Yıldırım, B., & Selvi, M. (2017). STEM uygulamaları ve tam öğrenmenin etkileri üzerine deneysel bir çalışma.
- Yılmaz, Ş. (2019). *Scratch Programı Öğretiminde bBrikte Öğrenme Tekniği Kullanımının Öğrencilerin Akademik Başarısına ve Öz Yeterlik Algısına Etkisi*, yüksek Lisans Tezi, Afyon Kocatepe Üniversitesi Fen Bilimleri Enstitüsü.
- Yünkül, E., Durak, G., Çankaya, S., & 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.

## EKLER

### EK 1 Bilgisayar Destekli Eğitim Yapmaya İlişkin Tutum Ölçeği

		Kesahilde Katılmıyor um	Katılmıyor um	Katılmıyor um	Katılmıyor um	Kesahilde Katılmıyor um
1.	Bilgisayar eğitimde etkili kullanılmaz.					
2.	Bilgisayarı derste severek ve isteyerek kullanırım.					
3.	Mecbur kalmadıkça bilgisayar derste desteklemek amacıyla kullanmam.					
4.	Bilgisayar Destekli Eğitim benim için önemli bir konudur.					
5.	Bilgisayar Destekli Eğitim ile yapılan derslerde öğrenciler yaratıcılıklarını geliştiremez.					
6.	Bilgisayarı derslerimde daha etkili kullanmamın yollarını araştırırım.					
7.	Bilgisayarla eğitimi bir türlü bağdaştırıramıyorum.					
8.	Bilgisayarı kullandığı derslerde öğrenciler daha iyi öğrenir.					
9.	Bilgisayar Destekli Eğitim yapmak yerine konuyu kendim anlatırım.					
10.	Öğretmenler bilgisayar kullanmaya teşvik edilmelidir.					
11.	BDE ile ders yapmak zaman kaybıdır.					
12.	Bilgisayar öğrencilerin dikkatini çekmede etkili araçtır.					
13.	BDE ile öğrenciler diğer yöntem ve tekniklere göre daha az öğrenir.					
14.	Bilgisayar yardımıyla yapılan dersler eğlenceli geçer.					
15.	Bilgisayar desteği ile yapılan eğitimin katkısına harcanan emeği karşılamaz.					
16.	Her sınıfta bilgisayar aktif bir şekilde kullanılmaktadır.					
17.	Dersleri yaparken bilgisayarı öğretim amaçlı kullanmaya düşünmem.					
18.	Bilgisayarı etkili bir öğretim aracı olduğunu düşünüyorum.					
19.	Bilgisayarı başından bir an önce kalkmak isterim.					
20.	Derslerimde bilgisayar kullanmaya çalışırım.					



**EK 3 Eğitsel Bilgisayar Oyunları Destekli Kodlama Öğrenimine Yönelik Tutum Ölçeği**

		Kesinlikle Katılmıyorum	Katılmıyorum	Kararsızım	Katılıyorum	Kesinlikle Katılıyorum
1.	Bilgisayarla (tablet, telefon vb.) oyun oynamak okul başarıma katkı sağlar.					
2.	Bilgisayar oyunlarından yararlanılarak ders çalışmak faydalıdır.					
3.	Dersleri bilgisayarla oyun oynayarak işlemeyi isterim.					
4.	Bilgisayar oyunlarındaki kahramanların dış görünüşümü kendim ayarlamak isterim.					
5.	Bilgisayarda kendi oyunumu tasarlamak isterim.					
6.	Bilgisayarda kodlama yapmayı öğrenmek isterim.					
7.	Bilgisayar ile ilgili meslek sahibi olanlar aktif değildir.					
8.	Oyunların seviyeli olması ilgimi çeker.					
9.	Oyunlardaki seviyelerin çokluğu oyuna olan ilgimi artırır.					
10.	Oyun hazırlarken kahramanın başına neler gelebileceğini kendim belirlemek isterim.					
11.	Kendi hazırladığım bilgisayar oyunumu arkadaşlarımla da oynamasını isterim.					
12.	Bilgisayarda kendi oyunumu hazırlamak ilgimi çekmez.					
13.	Arkadaşlarımla tasarladığım oyunları oynamak eğlenceli olabilir.					

14.	Bilgisayarda kod yazmak benim için zordur.					
15.	Bilgisayarda kod yazmayı öğrenmek istemem.					
16.	Sokakta oynanan oyunlar benim için tehlikelidir.					
17.	Kodlama öğreniminin benim için faydalı olacağımı düşünüyorum.					
18.	Bilgisayar oyunları ile eğitim öğretim olmaz.					
19.	Kodlama öğrenimi problem çözme becerimi geliştireceği için sınavlarda başarımla artar.					
20.	Kendi oyunumu tasarlamak yaratıcılığımı geliştirecektir.					
21.	Bilgisayarla oyun oynamak boş zamanları değerlendirmek için idealdir.					
22.	Dışarıda (futbol, basketbol, voleybol vb.) oynanan oyunlardan daha bilgisayarla oynanan oyunlar daha iyidir.					
23.	Arkadaşlarımla oyun oynamaktan daha bilgisayar oyunu oynamayı tercih ederim.					
24.	Bilgisayar oyunlarında ikili oyunları daha çok severim.					
25.	Kodlama öğrenimi zeka geliştirir.					
26.	Bilgisayarda oyun oynamak bireyleri tembelleştirir.					
27.	Derslerin bilgisayar oyunları ile işlenmesi dersle olan ilgimi artırır.					
28.	Bilgisayarda oyun kodlamayı öğrenirsem derslerim de başarımla artar.					

## EK 4 Scratch Eğitimi Akademik Başarı Testi

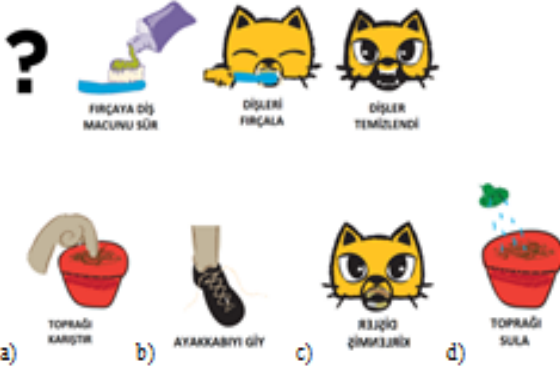
1) "Bir sorunu çözmek veya belirlenmiş bir amaca ulaşmak için tasarlanan yola, takip edilen işlem basamaklarına ..... denir." Tanımda boş bırakılan aşağıdaki kavramlardan hangisi gelmelidir?(10 p)

- a)Algoritma      b)Problem      c)Yazılım      d)Programlama Dili

2) Algoritmayı başlatan ve bitiren akış şeması şekli aşağıdakilerden hangisidir?(10 p)

- a)Elips      b)Paralelkenar      c)Dikdörtgen      d)Eşkenar dörtgen

3) Resimde diş fırçalama algoritması görmektesiniz. Algoritmada soru işaretiyle ifade edilen adım aşağıdaki seçeneklerden hangisi olmalıdır?(15 p)



4) Ekmek alma algoritmasının adımları karışık olarak verilmiştir. Aşağıdaki seçeneklerden hangisinde bu adımlar doğru olarak sıralanmıştır?(15 p)

- a Ekmekçi al  
b. Bitir  
c. Sokağa çık  
d. Markete git  
e. Eve dön  
f. Kasaya götür  
g. Başla  
h.Parayı öde

- a) g, c, d, a, f, h, e, b      b) c, d, h, a, b, e, f, g      c) g, d, e, a, h, f, c, b      d) a, h, c, d, e, f, g, b

[Bu bölümde Scratch programı ile ilgili sorular bulunmaktadır.]

5) Scratch karakteri olan kedi, hangi (x,y) noktasına geldiği zaman sahneden kaybolur?(5p)

- a)(400,400)      b)(158,0)      c)(240,180)      d)(ScratchAbout, 2020)





[12-13-14. soruları aşağıdaki resme göre cevaplayınız.]



12) Köpekbalığının ve balığın kenara gelince geri dönmesini sağlayan kod nedir?(5p)

a) 

b) 

c) 

d) 

13) Köpekbalığın balığa değince balık kayboluyorsa kullanılan kod hangisidir?(5p)

a) 

b) 

c) 

d) 

14) Köpekbalığın sürekli ağını açıp kapatıyormuş gibi görünmesini istiyorsak kullanmamız gereken kod bloğu hangisidir?(5p)

a) 

b) 

c) 

d) 

## EK 5 Tospa Eğitim Akademi Başarı Testi

1) "Bir sorunu çözmek veya belirlenmiş bir amaca ulaşmak için tasarlanan yola, takip edilen işlem basamaklarına ..... denir." Tanımda boş bırakılan aşağıdaki kavramlardan hangisi gelmelidir?(10 p)

- a)Algoritma
- b)Problem
- c)Yazılım
- d)Programlama Dili

2) Algoritmayı başlatan ve bitiren akış şeması şekli aşağıdakilerden hangisidir?(10 p)

- a)Elips
- b)Paralelkenar
- c>Dikdörtgen
- d>Eşkenar dörtgen

3) Resimde diş fırçalama algoritması görmekteyiz. Algoritmada soru işaretiyle ifade edilen adım aşağıdaki seçeneklerden hangisi olmalıdır?(15 p)



4)Ekmek alma algoritmasının adımları karışık olarak verilmiştir. Aşağıdaki seçeneklerden hangisinde bu adımlar doğru olarak sıralanmıştır?(15 p)

- a. Ekmeği al
- b. Bitir
- c. Sokağa çık
- d. Markete git
- e. Eve dön
- f. Kasaya götür
- g. Başla
- h. Parayı öde

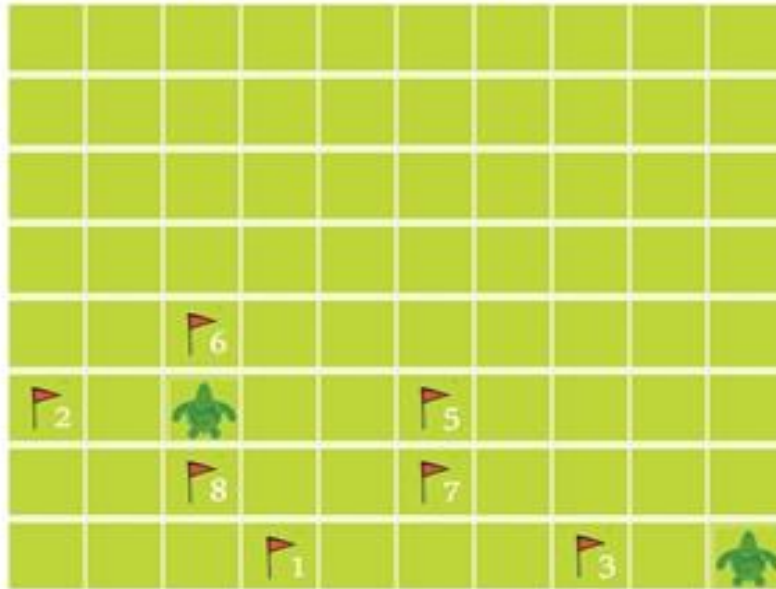
a) g, c, d, a, f, h, e, b

b) c, d, h, a, b, e, f, g

c) g, d, e, a, h, f, c, b

d) a, h, c, d, e, f, g, b

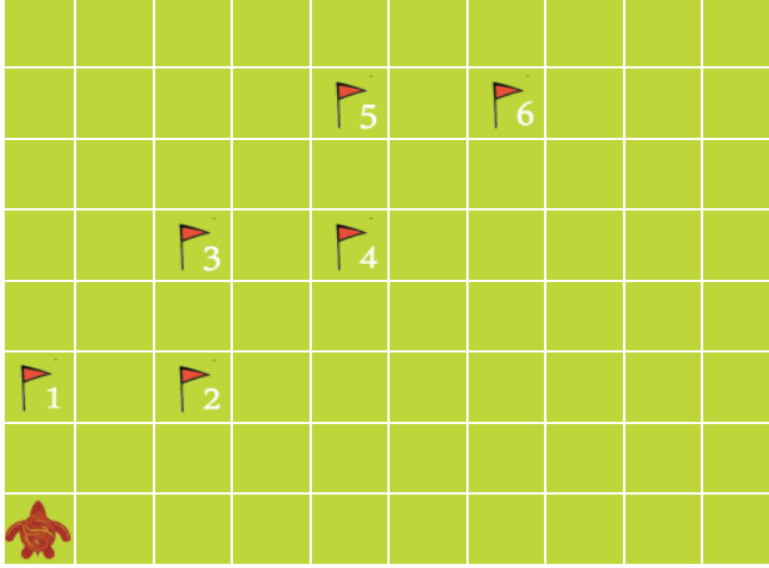
5. Tosmaları hedef noktalarına ulaştırın (60 p)



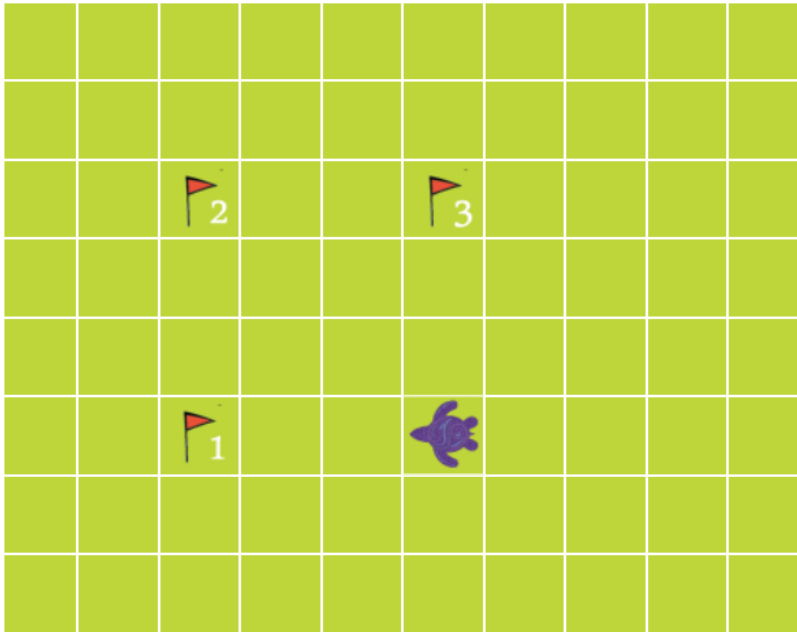
## Çalışma Kağıdı 1

Tospaları hedeflere ulaştırın

### Zigzag Tospaa



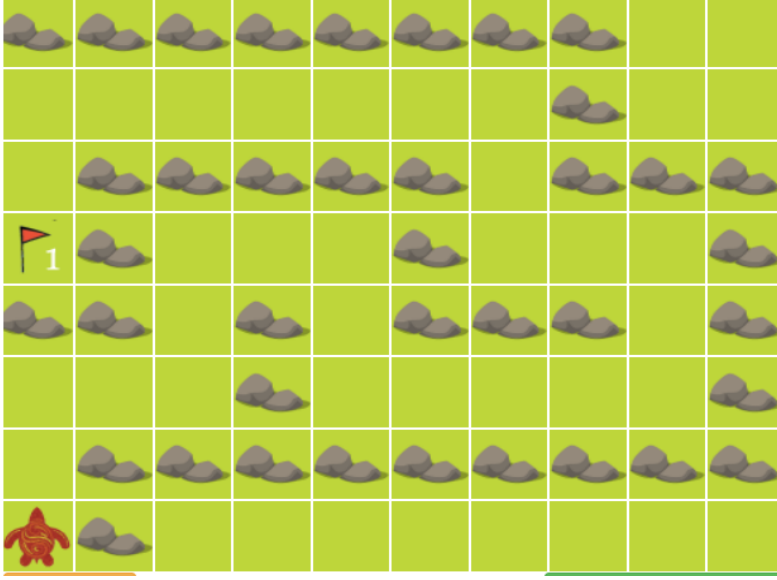
### Köşe Kapmaca



## Çalışma Kağıdı 2

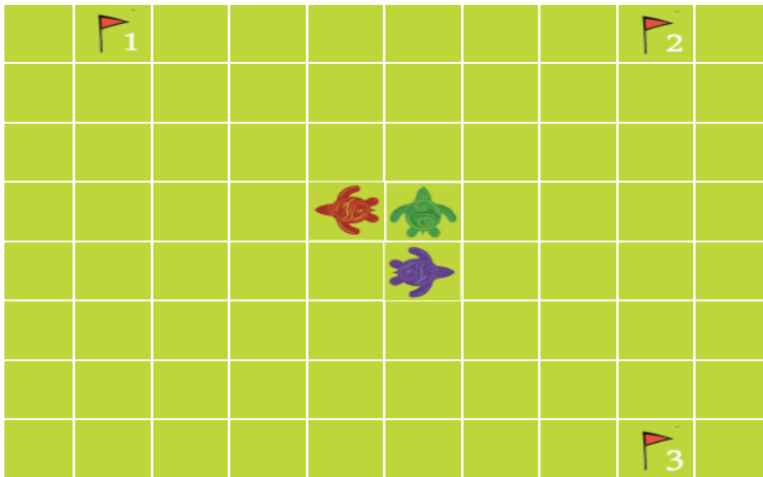
Tospaayı engellere değmeden hedeflere ulaştırın

### Labirent



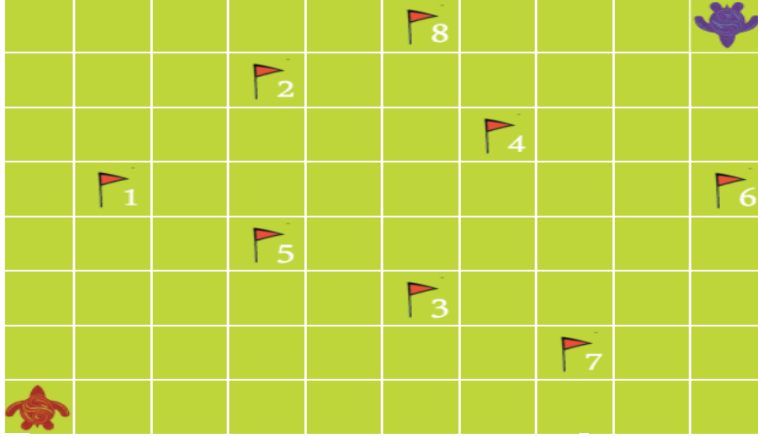
Tospaaları aynı yönerge ile hedeflerine ulaştırın

### 3 tospaa tek kod



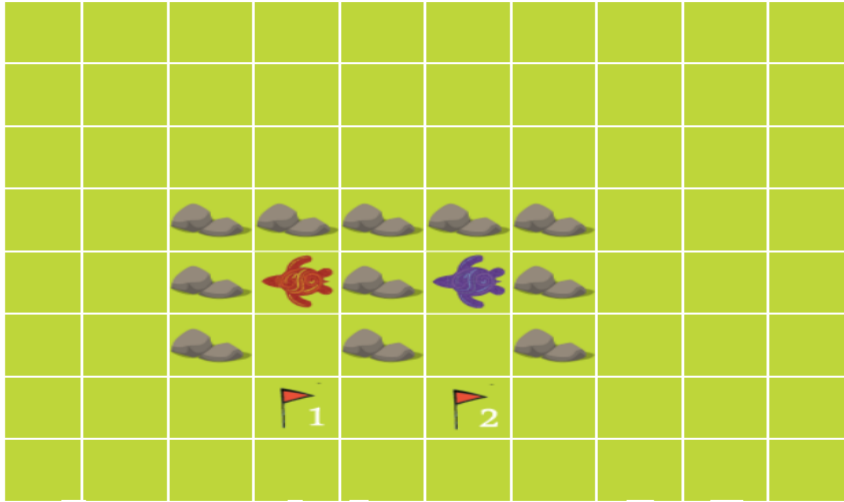
## Çalışma Kağıdı 3

### Tek Sayılar Çift Sayılar



Tospaa ile matematik öğretilemez mi? Kırmızı Tospaa tek sayıları, Mavi Tospaa çift sayıları toplasin

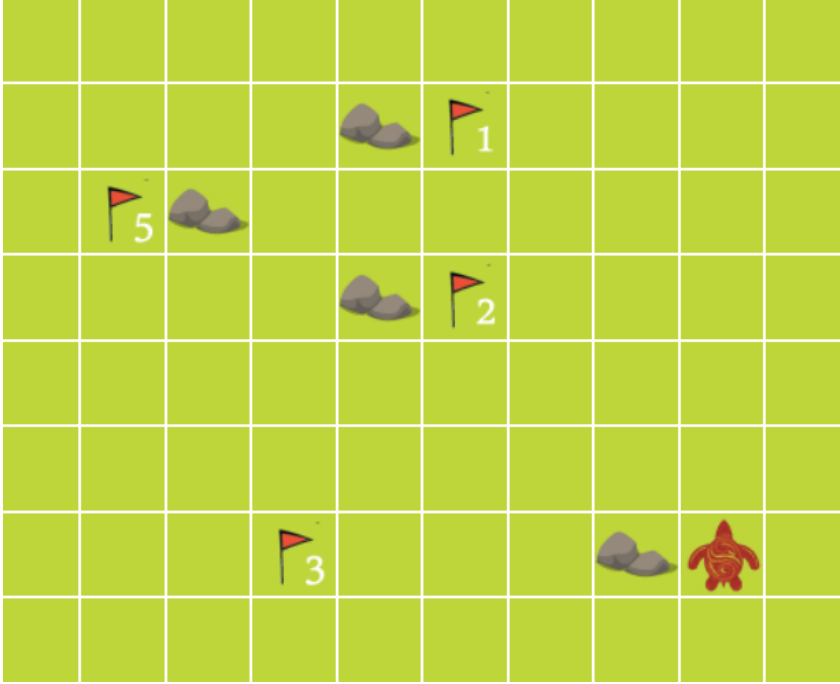
### Koşullar



Tek bir kodla iki tospaa'yı da hedeflerine ulaştıralım

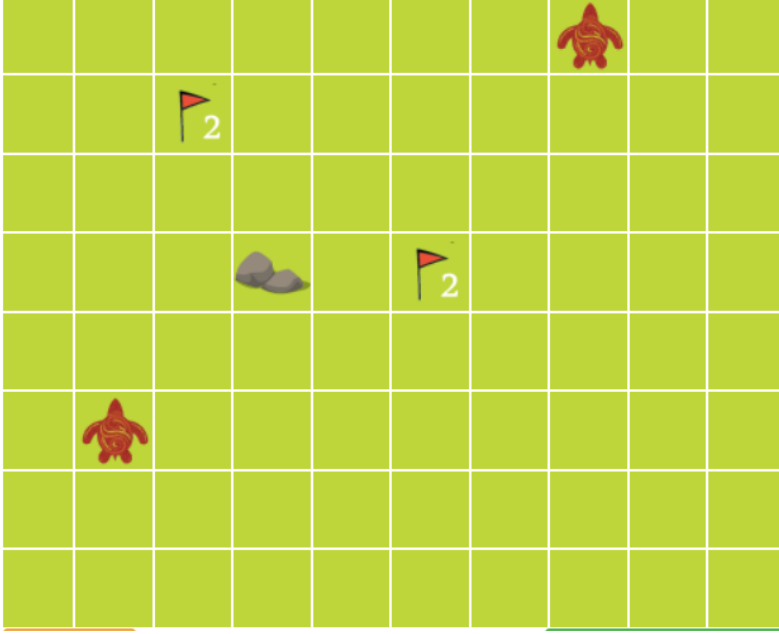
## Çalışma Kağıdı 4

Tospaayı engellere değmeden hedeflere ulaştırın



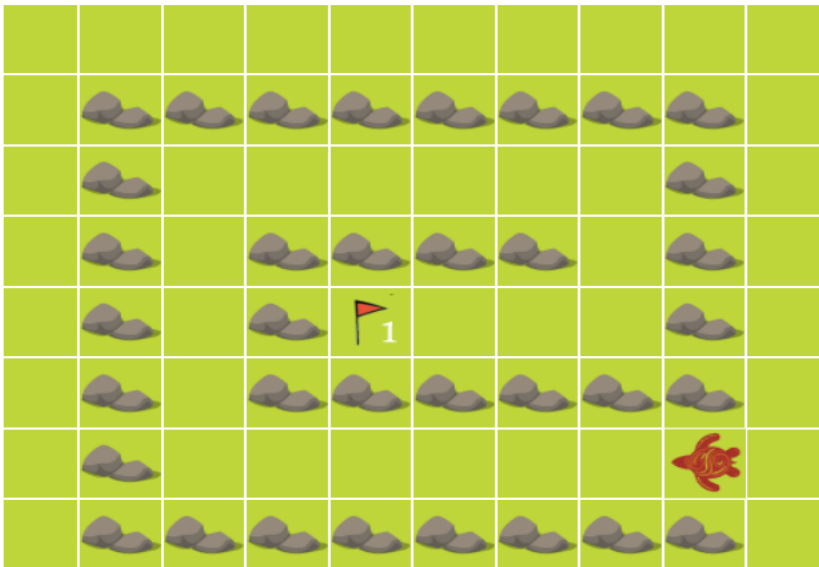
## Çalışma Kağıdı 5

Tospaaları engelle değmeden hedeflerine ulaştırın



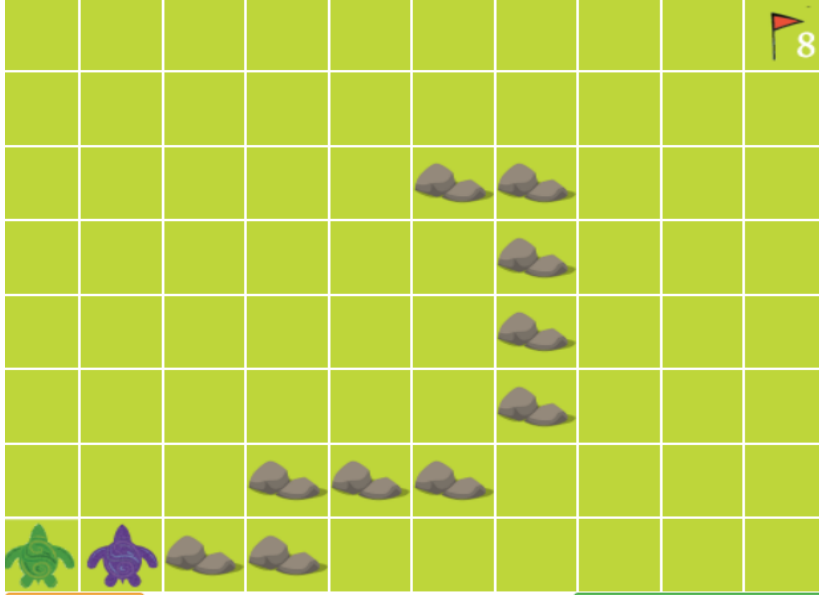
Tospaaları engelle değmeden hedeflerine ulaştırın

## Dön Dolaş

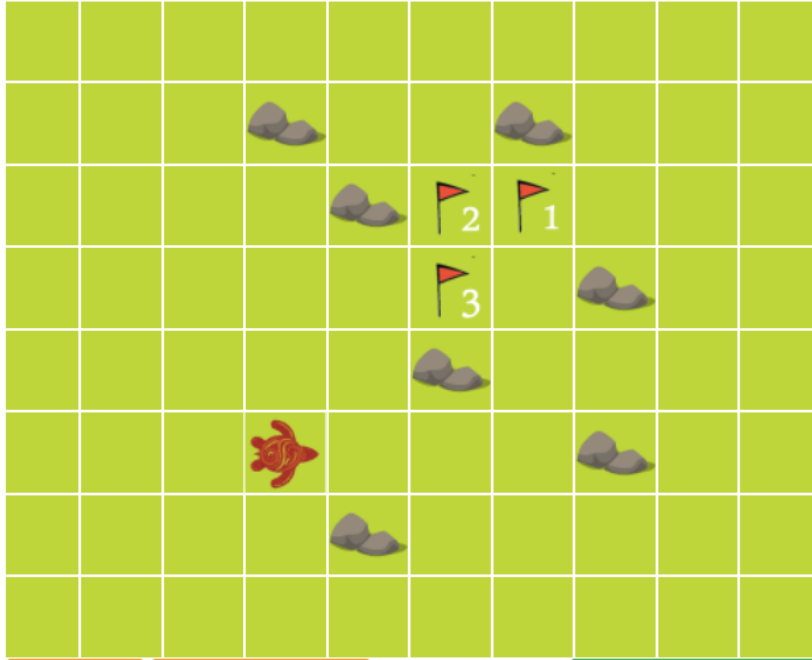


## Çalışma Kağıdı 6

Tospaaları engele değmeden hedefe ulaştırın

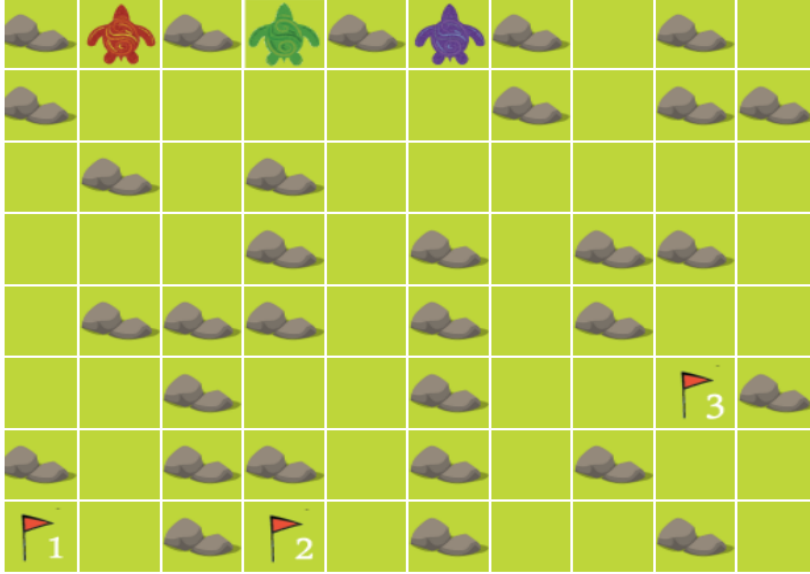


Tospaayı engele değmeden hedeflere ulaştırın

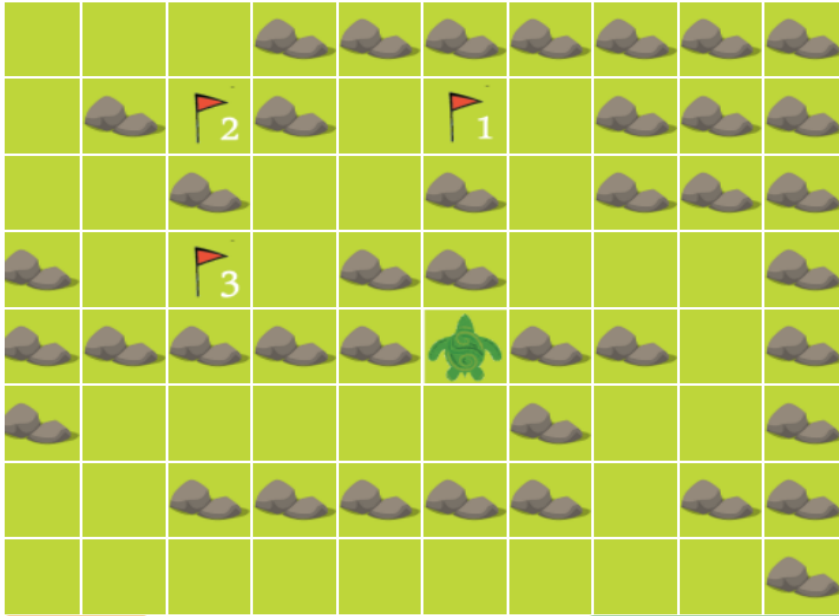


## Çalışma Kağıdı 7

Tospaaları engele değmeden hedeflere ulaştırın



Tospaayı engele değmeden hedeflere ulaştırın



## Scratch Nedir? İsimli Word Belgesi

Scratch programı eğlenceli bir ortamda resim, ses, müzik gibi çeşitli medya araçlarını bir araya getirebileceğimiz, kendi animasyonlarımızı, bilgisayar oyunlarımızı tasarlayabileceğiniz ya da interaktif hikayeler anlatabileceğimiz ve paylaşabileceğimiz bir grafik programlama dilidir. Scratch web sayfası, Scratch' i öğrenebilmemiz için birçok kaynak içermektedir. Web sayfasını ziyaret etmek isterseniz 'http://scratch.mit.edu' adresinden Scratch ana sayfasına ulaşabilirsiniz. Yukarıdaki linkte scratch.edu uzantılı bir site olduğunu tespit ediyoruz. Daha önce öğrendiğimiz bilgileri hatırladığımızda edu uzantılı sitelerin üniversitelerin siteleri olduğunu söylemiştik. Scratch.edu olmasının sebebi bu projenin MIT üniversitesi tarafından hazırlanmış olup sürekli gelişiyor olmasındandır. İnternet üzerinden Scratch için hazırlanmış kaynaklara baktığımızda birbirinden farklı ara yüzlerin olduğunu görebilirsiniz. Kodlama programı olan Scratch iki şekilde kullanılabilir. 1. Seçenek web sitesine üye olarak online yani çevrimiçi şekilde projeler hazırlayabilir. 2. seçenek ise Scratch imagine Program'ı bilgisayarınıza indirip kurarak internet bağlantısı olmadan da projeler hazırlayabilmektir. Bununla ilgili ayrıntılı yazıya <https://ozgurseremet.com/scratch-kurulumu-ve-kullanimi/> linkinden ulaşabilirsiniz. Bu yazıda Scratch Windows, Scratch Mac Os veya Scratch Linux seçeneklerini görebilirsiniz. Tablet ve telefonlarda henüz Scratch programını kullanamıyoruz. Scratch Android veya Scratch IOS sürüm şuan bulunmamaktadır. Ama gelen talepler doğrultusunda en kısa sürede Android ve IOS sistemlerde kullanılabilmesi için gerekli çalışmaların yapılacağına inanıyorum. Tablet ve telefonlar için ufak bir versiyonu bulunmaktadır. Bu versiyon Scratch jr olarak adlandırılıp mini versiyonudur.

### Üye Olma ve Sitede Çalışma

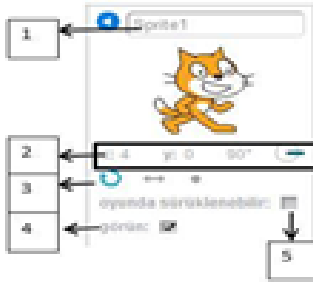


Scratch.mit.edu adresi açıldığında sağ üstteki Scratch'a Katıl bağlantısından siteye üye olabilirsiniz. Daha sonraki kullanımlarımızda aynı sayfadaki Giriş bağlantısıyla da sisteme kullanıcı adı ve şifrenizle girebilirsiniz. Bu sayfa üzerindeki Oluştur bağlantısıyla doğrudan site üzerinde çalışarak programlar geliştirebilirsiniz. Keşfet bağlantısını kullanarak diğer üyeler tarafından hazırlanan programları hazırlanma aşamalarını ve kullanılan komutları görebilirsiniz.

## Scratch Nedir 2? İsimli Word Belgesi

### Kukla Bilgisi

Kuklamanın sol üst köşesindeki tuşuna tıklanarak açılan bölümde kuklayla ilgili ayarlar yapmak için kullanılır. Bu ayarlar;



1. **Kukla adı:** Kuklamanın adını değiştirmek için kullanılır.

2. **Konum ve Yön bilgisi:** Kuklamanın koordinatlarını(konum) ve yönünü belirtir bu bölümden kukla yönü değiştirilebilir.

3. **Dönme Özellikleri:** Karakterin nasıl döneceğini belirlediğimiz bölümdür. Bunlar;



1. Tıklanırsa karakter kostümü 360° dönebilir özelliği kazanır.

2. Tıklanırsa karakter kostümü sağa-sola dönebilir

3. Tıklanırsa karakter kostümü hiç bir şekilde dönemez.

4. **Görün:** Proje çalıştırıldığında kuklamanın ekranda görünüp görünmeyeceğini ayarlar.

5. **Oyunda Sürüldenebilir:** Proje çalıştırıldıktan sonra kuklamanın fare ile tutularak sürüklenip sürüklenemeyeceğini ayarlar.

### Kılıklar (Kostümler)



Karakterimizin başka görünümlerinin olmasını istiyorsak kütüphaneden ekleyebilir, kendimiz çizebilir, bilgisayarımızda kayıtlı bir resmi içe aktarabiliriz ya da kamera ile çekebiliriz. Var olan kostümleri kostümün sağ üst köşesindeki 'x' butonuna basarak silebiliriz, seçili kostümü sağdaki düzenleme alanında

düzenleyebiliriz. Kostüm üzerinde sağ tık yaparak kopyalayabilir ve silebiliriz.



### Sesler

Karakter için var olan sesleri görebilmek için bu bölüme tıklarız. Bu bölümden yeni sesler

ekleyebilir ( sırasıyla kütüphaneden ses ekler, ses kaydeder, bilgisayarda kayıtlı bir sesi ekler), var olan sesi dinleyebilir, düzenle ve etkiler altındaki komutlarla düzenleyebilir veya silebilirsiniz.

## Blok Paketi (Diziler)



Blok Paletinde karakterleri programlayabilmek için kullanılan bloklar vardır. 10 ana kategoride toplanmış blok grubu bulunmaktadır.

## Hareket

- 10 adım git
- 15 derece dön
- 15 derece dön
- 90 yönüne dön
- 'ye doğru dön
- x: 142 y: -37 noktasına git
- fare oku 'na git
- sn.de x: 142 y: -37 a süzül
- x'i 10 arttır
- x, 0 olsun
- y'yi 10 arttır
- y, 0 olsun
- kenara geldiysen sek
- kuklanın şekli sağa-sola dönebilzin
- x konumu
- y konumu
- yönü

Hareket bloğunda yer alan blokların işlevleri aşağıdaki gibidir.

1. Karakteri öne ya da arkaya doğru hareket ettirir.
2. Karakterin istenilen derece kadar saat yönünde dönmesini sağlar.
3. Karakterin istenilen derece kadar saat yönü tersine dönmesini sağlar.
4. Karakterin hangi yöne döneceğini belirler. (0=yukarı, 90=sağ,180=aşağı, -90=sol)
5. Karakteri 'mause işaretçisi' veya başka karakterler varsa istenilen 'karakter' e doğru döndürülür.
6. Karakteri istenilen x ve y konumuna getirir.
7. Karakter 'mause işaretçisi' veya başka karakterler varsa istenilen 'karakter' ile aynı konuma gönderilir.
8. Karakter belli bir süre içerisinde belirlenen x ve y konumuna hareket eder.
9. Karakterin bulunduğu konumun x değeri istenilen değer kadar arttırılır veya azaltılır.
10. Karakterin x konumunu istenilen değere çeker.
11. Karakterin bulunduğu konumun y değeri istenilen değer kadar arttırılır veya azaltılır.
12. Karakterin y konumunu istenilen değere çeker.
13. Karakter ekranın kenarına değdiği an karakteri ters yöne döndürür.
14. Karakterin sağa-sola dönme, etrafında dönebilme ve hiç dönmeme ayarlarını yapar
15. Karakterin x pozisyonu bilgisini verir. Bu seçenek işaretlendiğinde x değerini ekranda görebiliriz.
16. Karakterin y pozisyonu bilgisini verir. Bu seçenek işaretlendiğinde y değerini ekranda görebiliriz.
17. Karakterin yön bilgisini verir. Bu seçenek işaretlendiğinde yön bilgisini ekranda görebiliriz.

## Görünüm



Görünüm bloğunda yer alan blokların işlevleri tablodaki gibidir.

1. Karakter istenilen süre boyunca 'Hello!' yazan kutudaki değeri ekranda konuşma balonu içerisinde gösterir.
2. Karakter 'Merhaba' yazan kutudaki değeri ekranda balon içerisinde gösterir. Kutu içerisi boş bırakılırsa konuşma balonu oluşmaz.
3. Karakter istenilen süre boyunca 'Hmm...' yazan kutudaki değeri ekranda düşünme balonu içerisinde gösterir.
4. Karakter 'Hmm...' yazan kutudaki değeri ekranda balon içerisinde gösterir.
5. Karakteri ekranda görünür hale getirir.
6. Karakteri Ekranda görünmez yapar.
7. Karakteri istenilen kostüme geçmesini sağlar.
8. Karakterin o anki kostümünden bir sonraki kostümüne geçmesini sağlar.
9. Mevcut sahne dekorunu istenilen dekorla değiştirir.
10. Karakterin rengi, balıkgözü, Hızlı dön, Piksellere böl, mozaik, parlaklık, hayalet efekti sayı yazan yerdeki değer kadar değişir.
11. Karakterin rengi, balıkgözü, Hızlı dön, Piksellere böl, mozaik, parlaklık, hayalet efekti sayı yazan yerdeki değer olur.
12. Karakter üzerinde uygulanmış bütün efektleri geri alır.
13. Karakterin boyunu istenilen değer kadar değiştirir.
14. Karakterin boyu istenilen %' ye getirilir.
15. Karakterin diğer bir üst katmana çıkmasını sağlar. Böylece karakter diğer karakterlerin önünde görünebilir.
16. Karakterin istenilen katman değeri kadar geri gitmesini sağlar. Böylece karakter diğer karakterlerin arkasında gizlenebilir.
17. Karakterin o anki kostüm bilgisini verir. Yan tarafındaki kutu

işaretlendiğinde ekranda kostüm bilgisi görülür.

18. Sahnede o anda gösterilen dekorun adını verir.

19. Karakterin ebat değerini % olarak verir. Yan tarafındaki kutu işaretlendiğinde ebat değeri ekranda görünür.

## Scratch Nedir 3? İsimli Word Belgesi

### Ses

- 1 miyav sesini çal
- 2 miyav sesini bitene kadar çal
- 3 tüm sesleri durdur
- 4 18 davulunu 0.25 vuruş çal
- 5 0.25 vuruş sus
- 6 60 notasını 0.5 vuruş çal
- 7 çalgıyı 1 yap
- 8 sesi 10 birim yükselt
- 9 ses şiddetini % 100 yap
- 10 ses şiddeti
- 11 tempoyu 20 arttır
- 12 tempoyu 60 vuruş/dk yap
- 13 tempo

Ses bloğunda yer alan blokların işlevleri tablodaki gibidir.

1. Seçili olan müzik çalmaya başlar. Program müziğin bitmesini beklemeden diğer blokları çalıştırmaya devam eder.
2. Seçili olan müzik çalmaya başlar ve bitene kadar diğer bloğa geçilmez.
3. Çalmaya devam eden müzikler bu blok çalıştığında durur.
4. 48 (davul) sesini istenilen istenen sürede bir çalar. (18 sesinin bulunduğu yer açılır menüdür ve ses buradan değiştirilebilir.)
5. İstenilen vuruşun bitmesini bekler.
6. İstenilen notayı istenilen sürede bir çalar.
7. Enstrümanı(çalgıyı) değiştirmeyi sağlar.
8. Sesi azaltır ya da arttırır.
9. Ses yüksekliğini belli bir % ye getirir.
10. O anki ses seviyesi bilgisini verir. Yan taraftaki kutu işaretlendiğinde ekranda ses bilgisi % olarak görünür.
11. Tempo değeri istenilen değere göre değiştirir.
12. Tempo değeri saniyede istenilen vuruş değeri yapılır.
13. O anki tempo değeri bilgisini verir. Yan taraftaki kutu işaretlendiğinde ekranda tempo değeri görünür. (20-500)

### Kalem

- 1 temizle
- 2 iz bırak
- 3 kalemi bastır
- 4 kalemi kaldır
- 5 kalem rengini yap
- 6 kalem rengini 10 arttır
- 7 kalem rengini 0 yap
- 8 kalem tonunu 10 arttır
- 9 kalem tonunu 50 yap
- 10 kalem kalınlığını 1 arttır
- 11 kalem kalınlığını 1 yap

Kalem bloğunda yer alan blokların işlevleri tablodaki gibidir.

1. Ekranı kalem ile çizilenlerin hepsini siler.
2. Karakterin o anki konumunda ekrana kendi imajını (görüntüsünü) yapıştırır.
3. Karakterin hareket ettiği doğrultuda kalem ile ekranı çizer.
4. Karakterin hareket ettiği yerler çizilmez.
5. Kalem rengi istenilen renk değerine geçer.
6. Kalem rengi bu blok çalıştığında istenilen değer kadar değişir.
7. Kalem renginin istenilen değer olması sağlanır.
8. Kalem tonu istenilen değer kadar değiştirilir. (kalem tonu 0' dan 100' e doğru açılır, 100' den 200'e doğru koyulaşır.)
9. Kalem tonu istenilen değer olur.
10. Kalem kalınlığı istenilen değer kadar değişir.
11. Kalem kalınlığı istenilen değer olur.

## Scratch Nedir 4? İsimli Word Belgesi

### Olaylar



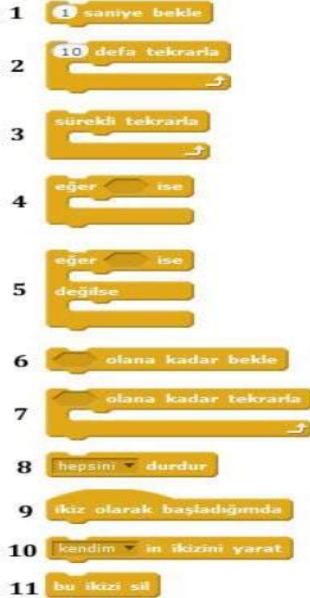
Olaylar bloğunda yer alan blokların işlevleri tablodaki gibidir.

1. Proje içerisinde yeşil bayrağa tıkladığında bu blok altında sıralanmış kod kümesini çalıştırır.
2. Seçilmiş olan tuşa (burada boşluk tuşu seçilmiş) basıldığında bu blok altında sıralanmış kod kümesini çalıştırır.
3. Karaktere tıkladığında bu blok altında sıralanmış kod kümesini çalıştırır.
4. Sahne dekoru burada seçilen dekor olunca bu blok altında sıralanmış kod kümesini çalıştırır.
5. Ses şiddeti belirtilen rakamdan yüksek olunca bu blok altında sıralanmış kod kümesini çalıştırır.
6. Yapılan duyurunun ardından bu blok altındaki kod kümesi çalışır.
7. Bu blok ile yeni haber yazarak komut kümelerini çalıştırmak için yeni bir blok oluşturmuş oluruz. Bu blok çalıştığında yapılan

duyurunun ardından bloktaki diğer komutlar çalışmaya devam ederken bu duyuruya ait kod kümesi de çalışmaya başlar.

8. Yapılan duyurunun ardından bu bloğun altında kod kümesi var ise bu duyuruya ait kod kümesinin çalışmadan bekler, bitiminden sonra kendi çalışmasına devam eder.

### Kontrol



Kontrol bloğunda yer alan blokların işlevleri tablodaki gibidir.

1. Projeyi istenilen süre kadar bekletir, süre tamamlanınca sıradaki bloktan çalışmaya devam eder.
2. Bu blok içerisinde bulunan bloklar sırasıyla istenilen değer kadar tekrar tekrar çalıştırılır.
3. Bu blok içerisinde bulunan bloklar sırasıyla program durdurulana kadar tekrar tekrar çalıştırılır (sonsuz döngü).
4. Eğer' den sonra gelen koşul doğru ise program bu blok içerisine girer ve bu blokları çalıştırır, ardından bu blok içerisinden çıkarak sonraki bloktan çalışmaya devam eder.
5. Eğer koşul doğru ise hemen altında yer alan kod blokları icra edilir ve bu kod bloğundan çıkılarak program icraya devam eder. Eğer koşul yanlış ise 'öyle değil ise' kısmında yer alan bloklar icra edilir, icradan sonra bu bloktan çıkılarak program akışı kaldığı yerden devam eder.
6. Koşul doğru olana kadar beklenir, koşul doğru olunca aşağıdaki bloklar çalışmaya başlar.
7. Koşul doğru olunca blok içerisine girilerek kodlar icra edilir. Koşulun her döngüde doğru olması halinde icra devam eder. Koşul doğrulanmadığı halde altındaki kodlar çalıştırılır.
8. 'Hepsini', 'bu diziyi', ' kuklanın kendi dizilerini' seçenekleri ile istenen kod dizilerini (bloklarının) çalışması durdurulur.
9. Bir kuklanın ikizi program akışı esnasında komutla oluşturulduğunda bu blok altındaki komutlar çalışır
10. Program akışı esnasında kendinin yada seçilecek başka bir kuklanın ikizi oluşturulur.
11. Oluşturulan kukla ikizin silinmesini sağlar.

## Scratch Nedir 5? İsimli Word Belgesi

### Algılama



Algılama bloğunda yer alan blokların işlevleri tablodaki gibidir.

1. Karakterimizin karaktere / Mouse işaretçisine/ kenara değip değmediği kontrolünü yapar. (seçim için açılır menüye tıklayın.)
2. Karakterin seçilen renge değip değmediği kontrolünü yapar. (Rengi değiştirmek için renk üzerine tıklayıp ekrandan renk seçin.)
3. Karakterin üzerindeki seçtiğiniz rengin seçtiğiniz bir başka renge değip değmediği kontrolünü yapar.
4. Karakterin açılır menüden seçtiğimiz başka bir karakter ya da mouse işaretçisiyle arasında olan mesafe bilgisini verir.
5. Diye Sor ve bekle komutu ardından verdiğiniz yanıt bu 'yanıt' bloğunda tutulur.
6. Blok yanındaki kutuya tıkladığında vermiş olduğun yanıt proje ekranında görünür.
7. Seçili olan tuşa basılı olma durumunu kontrol eder.
8. 'Fare basılı' mı kontrolü yapar.
9. Farenin x koordinatını verir.
10. Farenin y koordinatını verir.
11. Ses yüksekliği bilgisini barındırır.
12. Web kamerasındaki görüntünün hareket oranını ve yönünü tespit eder.
13. Web kamerasının görüntüsünü alarak sahnede görünmesini sağlar.
14. Web kamerasın görüntüsünü istenilen oranda

saydamlaştırır.

15. Saniye cinsinden geçen süre bilgisini tutar.

16. Süre ölçeri sıfırlar

17. Seçili olan karakterin/sahnenin seçenekler dahilindeki (x koordinatı/ y koordinatı/ yön/ şu anda görünen kostüm/ ebat/ ses seviyesi) bilgisini kullanabilmemizi sağlar

18. Bilgisayarın tarih ve saatine göre yapılan seçim doğrultusunda şimdiki yıl, ay, gün, haftanın kaçınıcı günü olduğu(Pazardan başlayarak), saat, dakika ve saniye bilgilerini verir.

19. 01.01.2000 tarih ve 00:00 saatten şu anki tarih ve saate kadar geçen gün sayısını verir.

20. Scratch kullanıcı adınızı içerir. Çalışması için siteme kullanıcı adınız ve şifrenizle giriş yapmış olmanız gerekir.

### İşlemler



İşlemler bloğunda yer alan blokların işlevleri tablodaki gibidir.

1. Boş kutulara yazılan iki değeri toplar.
2. İlk kutudaki değerden ikinci kutudaki değeri çıkarır.
3. İki değeri çarpar.
4. İlk kutudaki değeri ikinci kutudaki değere böler.
5. Belirtilen iki değer arasında rastgele bir sayı üretir.
6. İlk kutudaki değerden ikinci kutudaki değere küçük durumu kontrolü yapılır.
7. İki değerden bir ileri eşitliği durumu kontrolü yapılır.
8. İlk kutudaki değerden ikinci kutudaki değere büyük durumu kontrolü yapılır.
9. Belirtilen iki koşulunda doğruluk kontrolünü yapar.
10. Belirtilen iki koşuldaki en az birinin doğru olması durumunu arar.
11. Koşulda verilen durumun olmaması istendiği durumlarda bu blok kullanılır.
12. İstenilen iki değeri metin olarak birleştirilir.
13. İstenilen değerden hangi sırasındaki harfi isteniyor ise bu blok ile bu bilgi elde edilir.
14. Değerin uzunluk bilgisini bu blok ile elde edilir.

15. Bu blok ilk kutuya yazılan değerden ikinci kutuya yazılan değere göre modunu bulurken elde edilen kalanı verir.

16. İstenilen değerden en yakın tam sayıya yuvarlanmasını sağlar.

17. Matematiksel terimleri bulabileceğimiz (mutlak değer, karekök, sin, cos, tan, asin, acos, atan, ln, log, e^, 10^A) bu blok ile kutuya yazılan değerin karşılığını bilebilir ve bu değeri sayısal işlemlerde kullanabiliriz.

## Veri



Veri bloğunda yer alan komutların ve blokların işlevleri tablodaki gibidir.

1. Bu komut ile yeni bir değişken oluştururuz.
2. Oluşturulan 'Puan' isimli değişken verisi. (Kutu işaretlendiğinde ekranda değişken değerinin ne olduğu görünür.)
3. Var olan değişkenlerden istenilen seçilerek istenilen değer yapılır.
4. Var olan değişkenlerden istenilen seçilerek istenilen değer kadar artırılabilir ya da azaltılabilir.
5. Var olan değişkenlerden istenilen değişkeni ekranda gösterir.
6. Var olan değişkenlerden istenilen değişkeni ekranda gizler.
7. Bu komut ile yeni bir liste oluştururuz.
8. Sınıf adlı liste bilgisini barındırır. Kutu işaretlendiğinde liste içeriği ekranda görünür.
9. İstenilen değeri seçili olan listeye ekler.
10. Listenin istenen sırasındaki kaydı, son sırasındaki kaydı ya da tüm kayıtları siler.
11. İstene listenin ilk/son veya herhangi bir sıradaki kayıt bilgisini barındırır.
12. İstene listenin toplam kayıt sayısını verir.
13. İstene listeyi ekranda gösterir.
14. İstene listeyi ekranda gizler.
15. İstenilen değerin seçilen listede olup olmadığını kontrol eder.
16. İstene listenin ilk/son veya herhangi bir sırasına yeni bir nesne eklenir. (Eski bilgi silinmez, bu bilgi bir sonraki eleman olarak kaydırılır.)
17. İstene listenin ilk/son veya herhangi bir sırasındaki elemanı yerine yeni nesnemiz eklenir.

## Scratch Akvaryum Uygulaması

Akvaryum uygulaması için öncelikle sahnemize 3 tane balık ekliyoruz. Balıkların boyutlarını ve sahnede bulunacakları konumları isteğimize göre düzenliyoruz.

Balıklarımızdan biri seçili iken kontrol kod bloğundan “yeşil bayrak tıkladığında” kodunu alıyoruz. Balıkların sürekli tekrar eden bir şekilde hareket etmesini istediğimiz için altına yine kontrol kod bloğundan sürekli kodunu ekliyoruz.



Sürekli kodunun içine hareket kod bloğundan “10 adım gidin” kodunu ekliyoruz. Bu kod balığın hareket etmesini sağlayacaktır. 10 sayısı tıkladığında değiştirilebilir.



Çalışmamızı bu haliyle çalıştırdığımızda balığımız sahne kenarına kadar gidip duracaktır. Balığımızın sahne için sağa sola doğru hareket etmesini istediğimiz için sürekli kodunun içine hareket kod bloğundan “kenarda ise, zıplatın” kodunu ekleriz. Bu kod balığımızın ters yöne hareket etmesini sağlayacaktır.

Çalışmayı şimdi çalıştırdığımızda balığımız kenar geldiğinde ters yöne doğru hareket ettiğini göreceğiz yalnız balığımız başını çevirmeden hareket etmektedir. Balığımızın yönünü de hareket yönüne doğru dönmesi için karakter özelliklerinin olduğu yerde “sadece sola ya da sağa dönük” butonunun aktif ederek balığımız dönecektir.

Sadece sola ya da



Bütün balıklarımızın hareket etmesini istediğimiz için bu kod kümesini diğer balıklar için kopyalamamız gerekmektedir. Yeşil bayrak tıkladığında kod başlığında sağ tıklayıp kopyala ile kopyalama işlemi yaptıktan sonra bu kod kümesini diğer balıklardan birinin üzerine sürükleyerek kopyalamış oluruz. Ya da her balık için aynı kod kümesini tekrardan oluşturmak gerekmektedir.

Kod kümesini kopyaladıktan sonra her balığın yüzme hızı aynı olmadığından “10 adım gidin” kodundaki sayıyı her balık için farklı yapalım.

Balıklarımız artık hareket halinde olacaktır. Sahnenin arka planını akvaryum görünümünü yapıp sahneye başka balıklarda ekleyebilirsiniz.

Akvaryumumuza ses efekti ekleyelim şimdide. Bunun için sahnemiz seçili iken kontrol kod bloğundan “yeşil bayrak tıkladığında” kodunu alıyoruz. Ve sesin sahne içinde sürekli çalmasını istediğimizden altına yine kontrol kod bloğundan “sürekli” kodunu ekliyoruz.



tıkliyoruz.

Sürekli kodunun içine ses kod bloğundan sesimizin sürekli çalması için “pop sesini bitene kadar çal” kodunu ekliyoruz. Dikkat ederseniz pop yazısının yanında bir ok işareti vardır bu ok işaretinden sesi değiştirebiliriz. Bu sesi değiştirmek için öncelikle kodlarımızı eklediğimiz yazı bölümünde sesler sekmesine



Sesler sekmesinde içeri aktar butonuyla efektlerden “bubbles” sesini ekliyoruz ve kodda bu pop yerine bubbles sesini seçiyoruz.



Sahnemize bir tane ağzı kapalı bir köpek balığı ekleyelim. Köpek balığının kodları farklı olacağından kodlarını biz yazacağız. “yeşil bayrak tıkladığında” komutunu köpek balığımız seçili iken alıyoruz. Altında sürekli kodunu ekliyoruz. sürekli kodunun içine bu sefer hareket kod bloğundan “... doğru dönün” kodunu ekliyoruz.



“... doğru dönün” kodunda boş olan bölüme tıkladığımızda açılan seçeneklerden “Mouse işaretçisi”ni seçiyoruz. Yapmaya çalıştığımız şey şu, bu kodla köpek balığımız faremizin yönüne dönmesini sağlar. Faremizi takip etmesi içinde hareket kod bloğundan “10 adım gidin” kodunu sürekli kodunun içine ekleyerek köpek balığımızı hareket ettirmiş oluyoruz.

## Scratch: Papağan Uygulaması

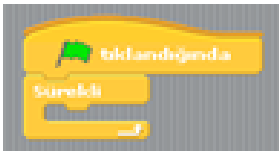
Yeni karakter ekle butonundan Animals klasöründen papağan “papağan1-a ” karakterini sahnemize ekliyoruz.

Sahnemizde olan kedi karakterini sağ tıklayıp sahneden siliyoruz.

Papağan karakterimize uçma görünümünü vermek için papağanımızın 2. bir resmi olan “papağan1-b” yi papağanımız seçili iken kostüm bölümünde “içeri aktar” butonundan ekliyoruz.



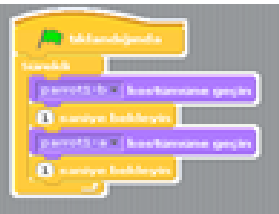
Kodlarımızı papağana vereceğimiz için papağanımız seçili iken kod bloklarından “Kontrol” bloğunu açıyoruz ve kod başlangıcı için “yeşil bayrak tıkladığından” kodunu alıyoruz.(bu kodun altına yeşil bayrak butonuna tıkladığından ne olacağını gösteren kodları ekleyeceğiz.)



Papağanımızın sürekli uçmasını istediğimiz için süreklilik içeren bir hareketlilik için kontrol kod bloğundan “sürekli” kodunu aşağıdaki gibi ekliyoruz.



Resimde de görüldüğü üzere “sürekli” kodunun içi boş, sürekli ne olacağını bu kısma ekleyeceğiz. Hatırlarsanız 2 papağan resmimiz vardı kanat çırpın. Bu iki görünüm arasında geçiş yapacağımız için “Görünüm” kod bloğundan “..... kostümünü seçin” kodunu sürekli kodunun içine alıyoruz. Boşluk kısmında gördüğümüz gibi papağan1-a yazıyor. İlk görünümümüz bu sonraki görünümün papağan1-b olmasını istiyoruz. Bunun için son eklediğimiz kodundan bir tane daha altına ekliyoruz.



Şimdi sahnemizi yeşil bayrak butonuna basarak çalıştırıp baktığımızda papağanımızın çok hızlı olarak kanat çırpıtığını göreceğiz. Bunu belli bir süre aralığında yapabilmesi için papağanın iki görünümünden sonrası için süre ekleyeceğimiz bunun için kontrol kod bloğundan “1 saniye bekleyin” kodunu resimdeki gibi ekliyoruz. Dilerssek beyaz alanlarda yazan 1 sayısını tıklayarak değiştirebiliriz.



Yeşil bayrak butonuna basıp sahnemizi izlediğimizde görüyoruz ki papağanımız durduğu yerde kanat çırpmakta, papağanımıza hareket halinde kanat çırpabiliriz. Sahnemizde papağanımızın uçuşa nerden başlamasını istiyorsak oraya taşıyoruz. Bunun için papağanımız seçili iken kontrol bloğundan yeni bir “yeşil bayrak tıkladığında” komutu ekliyoruz. Papağanımızı sahnede hareket ettirirken sahnede bulunduğu yeri belirten x ve y koordinatları değişmektedir. Papağanımız uçarak nereye

gitmesini istiyorsak o konumun x ve y koordinatlarını bilmeliyiz. Bu koordinatlar faremiz sahnede hareket ederken sahnenin sağ alt köşesinde gösterilmektedir.

Yeni eklediğimiz “yeşil bayrak tıkladığında” kodunun altına hareket kod bloğundan başlangıç



konumunu belirten “x ..... y ..... konumuna gidin” kodunu altına ekliyoruz.

Bundan sonrasında kuşun gideceğini konumlara kaç saniyede gideceğini belirten kodlar



ekliyoruz. Bu hareket kod bloğunun altındaki “1 saniyede x .... y ..... konumuna gidin” kodu eklenerek yapılır.

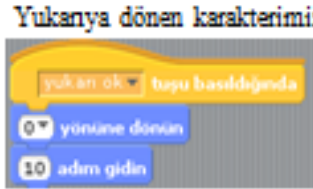
Son olarak isterseniz sahne arka planınızı değiştirip konuya uygun bir manzara yapabilirsiniz.

## Scratch: Yön Tuşları Uygulaması

Sahnemizde kedi karakterimiz seçili iken kontrol kod bloğundan “boşluk tuşuna basıldığında” kodunu alıyoruz. Bu kodun boşluk kısmına tıklayıp “yukarı ok” olarak değiştiriyoruz. Bu yukarı klavyede yön tuşu olan yukarı okuna basıldığında anlamına gelmektedir.



Bu kodun altına yukarı yön tuşuna basıldığında ne olmasını istediğimizi belirten kodları ekleyeceğiz. Karakterimize hareket vereceğimiz için hareket kod bloğundan “90 yönüne dönün” kodunu ekliyoruz ve beyaz bölümde yazan 90 sayısını tıklayarak yukarıyı ifade eden (0) ı seçiyoruz karakterimiz böylece yukarıya döner.



Yukarıya dönen karakterimizi hareket ettirmek için yine hareket kod bloğundan “10 adım gidin” kodunu altına ekliyoruz. Aynı kod kümesinden 4 tane olacak şekilde çoğaltıyoruz. Her bir kontrol kümesi yön tuşlarından(aşağı, sağa, sola) kullanılacak şekilde ayarlamalar yapılıyor.



Şimdi kedimize yürüyormuş gibi bir görünüm vermek için kedimizin kostümlerini kullanacağız. Hatırlarsak karakter özelliklerinde kostümler bölümü vardı burada karakterimize ait iki tane yürüyüş resmi var. Şimdi kod bölümümüze kontrol kod bloğundan “yeşil bayrak tıklandığında” kodunu alıyoruz, bunun altına yine kontrol kod bloğundan “sürekli” kodunu ekliyoruz. Sürekli kodunun içine kostümler arası geçiş için görünüm kod bloğundan “kostüm2 kostümüne geçin” kodunu iki defa ekliyoruz ve birinin kostüm2 bölümünü kostüm1 olarak değiştiriyoruz. Kostümler arası geçişi görebilmemiz için aralarına kontrol kod bloğundan “1 saniye bekleyin” kodunu ekliyoruz. Artık kedimiz yön tuşları ile adım atarak yürüyecektir.