



T.C.
NECMETTİN ERBAKAN ÜNİVERSİTESİ
EĞİTİM BİLİMLERİ ENSTİTÜSÜ

Bilgisayar ve Öğretim Teknolojileri Eğitimi Anabilim Dalı
Bilgisayar ve Öğretim Teknolojileri Eğitimi Bilim Dalı

Yüksek Lisans Tezi

BLOK TABANLI KODLAMA EĞİTİMİNİN ORTAOKUL ÖĞRENCİLERİNİN
BİLGİ İŞLEMSEL DÜŞÜNME BECERİLERİ VE KODLAMA ÖĞRENİMİNE
YÖNELİK TUTUMLARINA ETKİSİ: BLOCKY ÖRNEĞİ

Havva Nur TOTAN

Danışman
Doç. Dr. Agâh Tuğrul KORUCU

Konya 2021

ÖN SÖZ (TEŞEKKÜR)

Çağımızda kodlama eğitiminin önemi günden güne artmakta ve bu eğitimle birlikte bireylerde bilgi işlemsel düşünme, verilerin analizi ve sentezi, problem çözme, çözümleri farklı problemlere aktarabilme gibi 21.yüzyıl becerilerinin kazandırılabilceğı düşünölmektedir. Bunun farkında olan pek çok ölkede kodlama eğitimini okul öncesinden itibaren eğitim müfredatlarına dahil etmektedir. Ölkemizde de son yıllarda yapılan müfredat deęişikliğiyle birlikte kodlama eğitimi, ilkököl kademesinde farklı disiplinlerin içerisine entegre edilerek ortaoköl kademesinde ise Bilişim Teknolojileri ve Yazılım dersinin içerisinde verilmeye başlanmıştır. Yazılım oluşturma ve kodlama becerisiyle ilk kez tanışan öğrencilere bu eğitim blok tabanlı kodlama araçları ile verilmektedir. Konu hakkında alanyazın incelendiğinde çok fazla deneysel çalışmaya yer verilmediğı görölmüştür. Bu araştırmada blok tabanlı kodlama eğitiminin ortaoköl öğrencilerinin, bilgi işlemsel düşünme becerileri ve kodlama öğrenimine yönelik tutumlarına etkisi: blocky örneğı belirlenmesi hedeflenmiştir. Yapılan araştırmanın sonuçları ile alanyazına ve ölkemiz eğitim politikalarına az da olsa katkı sağlanması ümit edilmektedir.

Çalışmamın her aşamasında desteğini ve rehberliğini esirgemeyen değerli fikirleriyle bana yol gösteren danışmanım Sayın Doç. Dr. Ağâh Tuğrul KORUCU' ya teşekkür ederim.

Ayrıca bu sürece birlikte başladığımız, manevi olarak desteğini hiçbir zaman esirgemeyen değerli arkadaşım Şerife GÜRKEZ' e, çalışmama gönüllü olarak katılarak destek olan Eşmekaya Ortaokulu 5.sınıf öğrencilerime teşekkür ederim.

Son olarak bu süreçte beni yüreklendiren ve varlığını hep hissettiren aileme sonsuz teşekkürler.

Havva Nur TOTAN

KONYA- 2021

İÇİNDEKİLER

ÖN SÖZ (TEŞEKKÜR).....	ii
İÇİNDEKİLER	iii
TABLolar LİSTESİ.....	v
ŞEKİLLER LİSTESİ	vi
TEZ ÇALIŞMASI ORJİNALLİK RAPORU	vii
BİLİMSEL ETİK BEYANNAMESİ.....	viii
SİMGELER VE KISALTMALAR	ix
ÖZET	x
ABSTRACT.....	xi
1. GİRİŞ.....	1
1.1 Problem Durumu.....	5
1.2 Araştırmanın Amacı ve Önemi	6
1.3 Sayıtlar (Varsayımlar)	8
1.4 Sınırlılıklar.....	8
1.5 Tanımlar.....	9
2. KURAMSAL ÇERÇEVE VE ALAN YAZIN (İLGİLİ ARAŞTIRMALAR).....	10
2.1.Kodlama Kavramı ve Kodlama Eğitiminin Önemi	10
2.2.Kodlama Eğitimi Üzerine Yapılmış Çalışmalar	13
2.2.1.Yurt İçinde Yapılmış Çalışmalar	14
2.2.2.Yurt Dışında Yapılmış Çalışmalar.....	18
2.3.Blok Tabanlı Görsel Kodlama	23
2.3.1.Blok Tabanlı Kodlama Araçları.....	25
2.3.2.Blocky	28
2.4.Bilişim Teknolojileri ve Yazılım Dersi.....	32
2.5.Bilgi İşlemsel Düşünme	33
2.6.Bilgi işlemsel Düşünme Becerisi ve Kodlama İlişkisi.....	36
2.7.Bilgi İşlemsel Düşünme Becerisi Üzerine Yapılmış Araştırmalar	38
3. YÖNTEM	43
3.1 Araştırmanın Modeli.....	43
3.2. Çalışma Grubu	45
Öğrencilerin Demografik Özellikleri.....	46
3.3. Veri Toplama Araçları	48
3.3.1. Demografik Bilgi Formu.....	48

3.3.2. Öğrenme Sürecine İlişkin Görüş Bildirme Formu.....	48
3.3.3. Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algısı Ölçeği (BİDBÖA).....	49
3.3.4. Kodlama Eğitimine Yönelik Tutum Ölçeği (KEYTÖ).....	49
3.4. Verilerin Toplanması	50
3.5. Araştırma ve Uygulama Süreci.....	50
3.5 Verilerin Analizi	55
4. BULGULAR.....	57
4.1 Araştırmaya Dair Nicel Bulgular	57
4.1 Araştırmaya Dair Nitel Bulgular.....	61
5. TARTIŞMA, SONUÇ VE ÖNERİLER	63
5.1 Sonuç ve Tartışma.....	63
5.2 Öneriler	68
KAYNAKÇA.....	70

TABLolar LİSTESİ

Tablo- 1 Araştırmaya Yönelik Yarı Deneysel Desen	44
Tablo- 2 Gruplar Arası Bilgi İşlemsel Düşünme Becerisine Yönelik Öz yeterlik Ölçeği Ön Test Karşılaştırma Sonuçları	45
Tablo- 3 Gruplar Arası Kodlama Eğitimine Yönelik Tutum Ölçeği Ön Test Karşılaştırma Sonuçları.....	45
Tablo- 4 Çalışma Grupları Yüzde-Frekans Dağılımı.....	46
Tablo- 5 Çalışma Grupları Cinsiyet Dağılımı Yüzde Frekans Tablosu.....	46
Tablo- 6 Çalışma Grupları İnternet Bağlantısına Sahip Olma Durumları Yüzde Frekans Tablosu	47
Tablo- 7 Çalışma Grupları Tablet/Telefon/Bilgisayar Sahiplik Durumları Yüzde Frekans Tablosu	47
Tablo- 8 Çalışma Grupları Tablet/Telefon/Bilgisayar Günlük Kullanım Süreleri Yüzde Frekans Tablosu	47
Tablo- 9 Uygulama Evreleri	50
Tablo- 10 Uygulama Aşamasına İlişkin Ders Planı Özet Tablo.....	51
Tablo- 11 Bilgi işlemsel düşünme becerisine yönelik öz yeterlik ölçeği güvenilirlik analizi	57
Tablo- 12 Kodlama eğitimine yönelik tutum ölçeği güvenilirlik analizi.....	57
Tablo- 13 Bilgi işlemsel düşünme becerisine yönelik öz yeterlik ölçeği için deney grubu ön test-son test Karşılaştırma Sonuçları.....	57
Tablo- 14 Kodlama eğitimine yönelik tutum ölçeğine için deney grubu ön test-son test karşılaştırma sonuçları	58
Tablo- 15 Bilgi işlemsel düşünme becerisine yönelik öz yeterlik ölçeği için kontrol grubu ön test-son test karşılaştırma sonuçları	58
Tablo- 16 Kodlama eğitimine yönelik tutum ölçeği için kontrol grubu ön test-son test karşılaştırma sonuçları.....	59
Tablo- 17 Bilgi işlemsel düşünme becerisine yönelik öz yeterlik ölçeği için gruplar arası (deney- kontrol grubu) son test karşılaştırma (t- testi) sonuçları.....	59
Tablo- 18 Kodlama eğitimine yönelik tutum ölçeğine için gruplar arası (deney- kontrol grubu) son test karşılaştırma (t- testi) sonuçları	60
Tablo- 19 Deney grubu öğrencilerinin öğrenme sürecine ilişkin görüşleri	61

ŞEKİLLER LİSTESİ

Şekil- 1 Blocky Games Ekran Görüntüsü	29
Şekil- 2 Bulmaca Kategorisi Ekran Görüntüsü	30
Şekil- 3 Labirent Kategorisi Ekran Görüntüsü	30
Şekil- 4 Bloklara Ait Javascript Komut Ekranı	31
Şekil- 5 Kuş Kategorisine Ait Ekran Görüntüsü.....	31
Şekil- 6 Bilgisayarsız Kodlama Materyali	53
Şekil- 7 Bilgisayarsız Kodlama Materyali- Komutlar	53
Şekil- 8 Koşul Bloğu ve Döngü Bloğu	54
Şekil- 9 Koşul Bloğu.....	55

TEZ ÇALIŞMASI ORJİNALLİK RAPORU

Blok Tabanlı Kodlama Eğitiminin Ortaokul Öğrencilerinin, Bilgi İşlemsel Düşünme Becerileri ve Kodlama Öğrenimine Yönelik Tutumlarına Etkisi: Blocky Örneği başlıklı tez çalışmamın İç Kapak, Özetler, Ekler ve Ana Bölümlerden (Giriş, Alan Yazın, Yöntem, Bulgular, Tartışma, Sonuçlar ve Öneriler) oluşan toplam **108** sayfalık kısmına ilişkin, 6/07/2021 tarihinde tez danışmanım tarafından **iThenticate: Plagiarism Detection Software** adlı intihal tespit programından aşağıda belirtilen filtrelemeler uygulanarak alınmış olan orijinallik raporuna göre, tezimin benzerlik oranı **%19** 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.

6/07/2021

Havva Nur TOTAN

Doç. Dr. Agâh 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.

6/07/2021

Havva Nur TOTAN

SİMGELER VE KISALTMALAR

Kısaltmalar

BİT: Bilgi İletişim Teknolojileri

KEYTÖ: Kodlama Eğitimine Yönelik Tutum Ölçeği

BİDBÖA: Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algısı Ölçeği

BT: Bilişim Teknolojileri

BTY: Bilişim Teknolojileri ve Yazılım

MEB: Millî Eğitim Bakanlığı

TTKB: Talim ve Terbiye Kurulu Başkanlığı

MIT: Massachusetts Institute of Technology (Massachusetts Teknoloji Enstitüsü)

BİD: Bilgi İşlemsel Düşünce

BTP: Blok Tabanlı Programlama

BBÖD: Bilgisayar Bilimi Öğretmenleri Derneği (Computer Science Teacher Association – CSTA)

EBA: Eğitim Bilişim Ağı

TBD: Türkiye Bilişim Derneği

STEM: Science Technology Engineering and Mathematics

ITEST: Innovative Technology Experiences for Students and Teachers

ÖZET

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

BLOK TABANLI KODLAMA EĞİTİMİNİN ORTAOKUL ÖĞRENCİLERİNİN BİLGİ İŞLEMSEL DÜŞÜNME BECERİLERİ VE KODLAMA ÖĞRENİMİNE YÖNELİK TUTUMLARINA ETKİSİ: BLOCKY ÖRNEĞİ

Havva Nur TOTAN

Bu araştırmanın amacı blok tabanlı kodlama eğitiminin 5.sınıf kademesindeki öğrencilerinin bilgi işlemsel düşünme becerileri ve kodlamaya yönelik tutumlarına etkisinin Blocky kodlama ortamı kullanılarak incelenmesidir. Uygulama sürecin eğitim aşaması 4 hafta ve haftada 2 saat olacak şekilde planlanmıştır. Kodlama dersinde öğrencilerin bilgi işlemsel düşünme becerisi öz yeterlikleri ve kodlama öğrenmeye yönelik tutumları üzerine etkisinin araştırıldığı bu çalışmada araştırma modeli olarak karma (mixed) yöntem benimsenmiştir. Ayrıca ön test- son test kontrol gruplu tekrarlanan ölçümlerden oluşan yarı deneysel bir yöntem ile desenlenmiştir. Araştırmada nicel veriler bilgi işlemsel düşünme becerisi öz yeterlik algısı ölçeği (Gülbahar, Kert ve Kalelioğlu, 2018) ve kodlama öğrenmeye yönelik tutum ölçeği (Karaman ve Büyükalın Filiz, 2019) ile nitel veriler ise araştırmacı tarafından geliştirilen öğrenme sürecine ilişkin görüş bildirme formu ile toplanmıştır. Araştırmanın çalışma grubunu 2019-2020 eğitim öğretim yılında Aksaray ilinde bir ortaokulda 5.sınıf kademesinde öğrenim görmekte olan 38 öğrenci oluşturmaktadır. Çalışma grubu belirlenirken öğrencilerin ilk kez kodlama eğitimi alıyor olmasına dikkat edilmiştir. Çalışma grubunu oluşturan öğrencilerden 20'si kız, 18'i erkektir. Toplamda 4 haftalık eğitim sürecinin sonucunda elde edilen nicel veriler IBM SPSS 20 programında ve paket programlar kullanılarak analiz edilmiştir. Nitel verilerin analizinde içerik analizi yönteminden faydalanılmıştır. Nitel veriler yüzde ve frekans değerleri ile açıklanmıştır. Yapılan analizler sonucunda öğrencilerin bilgi işlemsel düşünme becerisi öz yeterliklerinin ve kodlama öğrenmeye yönelik tutumlarının uygulama öncesi ve uygulama sonrası değerleri karşılaştırıldığında pozitif yönde bir artış gözlenmiştir. Bu artışın, bilgi işlemsel düşünme becerisi öz yeterlik ve kodlamaya öğrenmeye yönelik tutum için istatistiksel olarak anlamlı olduğu görülmüştür. Bunun yanında yorumlanan nitel veriler de bu bulguyu destekler niteliktedir. Bu çalışma, bu alan ile ilgili farklı araştırmalar üretilmesine yardımcı olabilir. Blok tabanlı kodlama eğitiminde Blocky ortamının, Bilgi İşlemsel Düşünme Becerileri Öz Yeterlik Algısı Ölçeği (BİDBÖA) ve Kodlama Eğitimine Yönelik Tutum Ölçeği (KEYTÖ) Ölçeği üzerindeki toplam etki büyüklüğünü belirlemek için eta kare değeri incelenmiştir. Bulgulara göre Blok tabanlı kodlama eğitiminde Blocky ortamının, öğrencilerin bilgi işlemsel düşünme becerileri öz yeterlik algıları ve kodlama öğrenmeye yönelik tutumları üzerinde “geniş” bir etki büyüklüğüne sahip olduğu söylenebilir. Bu çalışma, blok tabanlı kodlama ve bilgi işlemsel düşünme alanında farklı faktörler de sürece dahil edilerek yapılacak çalışmalar için yol gösterici bir çalışma olabilir.

Anahtar Kelimeler: Bilgi işlemsel düşünme, blocky, kodlama, tutum

ABSTRACT

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

THE EFFECT OF BLOCK BASED CODING EDUCATION ON THE STUDENTS' ATTITUDES ABOUT THE SECONDARY SCHOOL STUDENTS' COMPUTATIONAL LEARNING SKILLS AND CODING LEARNING: BLOCKY SAMPLE

Havva Nur TOTAN

The aim of this study is to examine the effect of block-based coding education on 5th grade students' computational thinking skills and attitudes towards coding using the Blocky coding environment. The training phase of the implementation process is planned as 4 weeks and 2 hours a week. In this study, in which the effect of students' computational thinking skills on their self-efficacy and attitudes towards learning coding was investigated in the coding course, mixed method was adopted as a research model. In addition, the pretest-posttest was designed with a quasi-experimental method consisting of repeated measurements with a control group. In the study, quantitative data were collected through the computational thinking skill self-efficacy perception scale (Gülbahar, Kert, & Kalelioğlu, 2018) and the attitude towards learning coding scale (Karaman & Büyükalan Filiz, 2019), and the qualitative data were collected with the opinion form about the learning process developed by the researcher. The study group of the research consists of 38 5th grade students studying at a secondary school in Aksaray in the 2019-2020 academic year. While determining the study group, attention was paid to the fact that the students were taking coding training for the first time. Among the students that make up the study group, the number of female students is 20 and the number of male students is 18. The quantitative data obtained as a result of the 4-week training process in total were analyzed using IBM SPSS 20 program and package programs. Content analysis method was used in the analysis of qualitative data. Qualitative data are explained in terms of percentage and frequency. As a result of the analysis, a positive increase was observed when the pre-application and post-application values of the students' computational thinking skills self-efficacy and attitudes towards learning coding were compared. This increase was found to be statistically significant for computational thinking skill self-efficacy and attitude towards learning coding. In addition, the qualitative data interpreted also support this finding. This study can help produce different researches in this field. In the block-based coding education, the eta squared value was examined to determine the total effect size of the Blocky environment on the Information Computational Thinking Skills Self-Efficacy Scale and the Attitude Scale for Coding Education. According to the findings, it can be said that the Blocky environment in block-based coding education has a "large" effect size on students' computational thinking skills, self-efficacy perceptions and attitudes towards learning coding. This study can be a guiding study for studies to be done in the field of block-based coding and computational thinking by including different factors in the process.

Keywords: computational thinking, blocky games, coding, attitude

BÖLÜM 1

1. GİRİŞ

Teknolojik olarak hızla gelişen ve bu gelişmeler bağlamında değişen, içinde bulunduğumuz bilgi toplumunda öğrenenlerden beklenen temel beceriler de değişmektedir. Yaşamımızda neredeyse her alana etkisi bulunan teknolojilerin bireyler üzerindeki en önemli etkilerinden biri de bireylerde bazı becerileri zorunlu kılmasıdır (Ramazanoğlu, 2021). Bilim ve teknoloji alanında yaşanan gelişmeler birtakım ihtiyaçları beraberinde getirerek 21.yüzyıl becerileri kavramının oluşmasına zemin hazırlamıştır (Bülbül Şoltan, 2018). Öğrenenler, çağın gerektirdiği bu ihtiyaçlara yönelik bilgi ve beceriler kazandırılarak yetiştirilmelidir. Sürekli gelişmekte olan dünyamızda artık bireylerden eleştirel ve algoritmik düşünme, problem çözme ve çözüm yollarını benzer durumlara uyarlayabilme gibi bilişsel gelişim için önemli olan bu becerilere sahip olmaları beklenmektedir (Scot, 2018). 21.yüzyıl becerileri kazanmış bir birey, problem çözebilen, çözümleri farklı problemlere uyarlayabilen, üretken, yaşam boyu öğrenen ve sınıfta öğrendiğinin ötesine gidebilen bireydir (Gülbahar, Kert & Kalelioğlu, 2019). Ayrıca bireylere kazandırılan bu beceriler ile tüketen bir nesilden daha çok üreten bir neslin yetiştirilmesine imkân sağlanır (Saygılı Yıldırım, 2020). Yeni bir kavram olan bilgi işlemsel düşünme becerisi de 21.yüzyıl becerileri arasında gösterilmektedir. Bilgi işlemsel düşünme becerisi, bireylerin karşılaştıkları sorunları daha küçük alt sorunlara bölerek onları çözebilmeleri ile ilgilidir. BBÖD ve Uluslararası Eğitim Teknolojisi Derneği (2014) (The International Society for Technology in Education- ISTE) bilgi işlemsel düşünme becerisinin eleştirel düşünme, yaratıcı düşünme, algoritmik düşünme, problem çözme ve işbirlikçi öğrenmenin dışavurumu olduğunu ifade etmektedir. Curzon (2015) bilgi işlemsel düşünmeyi bireylerin herhangi bir problemi çözme becerisi olarak tanımlamıştır. Bilgi işlemsel düşünmeyi sistem tasarlamının, problem çözmenin ve bilgisayar bilimi yoluyla insan davranışlarını anlamının bir yolu olarak tanımlamak da mümkündür (Yağcı, 2018). Li vd. (2020) ise bilgi işlemsel düşüncenin bilgisayarlara odaklanmaktan öte bilginin işlenmesi ile alakalı bir süreç olarak incelenmesi gerektiğini ifade etmiştir. Ayrıca bilgisayarlar aracılığı ile yönetilecek olan problem çözme süreçleri, veri analizi ve veri işleme gibi pek çok durumun öncesinde bilgi işlemsel düşünme becerisinin kazanılmış olması gerekliliği ortaya çıkmıştır (Özyol, 2019). Üst düzey düşünme becerilerinden biri olarak kabul

edilmiş olan bu becerinin üstün yetenekli öğrencilerin eğitimine de zenginleştirilmiş etkinlikler ile birlikte katkı sağlayabileceği belirtilmiştir (Taş, 2018). Bunun yanında günlük hayatımızda oldukça sık kullandığımız problem çözme yeteneği de bilgi işlemsel düşünme becerisinin temelini oluşturmaktadır (Özyol, 2019).

Bireylere büyük sorunları daha küçük ve çözümlenebilir parçalara bölerek çözüm yollarını öğreten bilgi işlemsel düşüncenin tanımlarına bakıldığında alt becerileri ile bilgi çağında öğrencilerinden beklenen becerilerinin birbiriyle örtüştüğü görülmektedir (Oluk, Korkmaz & Oluk, 2018). Bilgi işlemsel düşüncenin günlük yaşam becerilerindeki etkililiği ve öğrenenlere kazandırdığı 21.yüzyıl becerileri göz önünde bulundurulduğunda eğitim müfredatlarına entegrasi oldukça önemli hale gelmektedir. Park, Song ve Kim (2015)'e göre veri elde etme, bu verilerin gösterimi ve analizi, problem çözme süreçlerinde algoritmalar oluşturma, soyutlama ve modelleme, gibi birtakım becerilerin sistematik şekilde geliştirilmesinde bilgi işlemsel düşünme becerisi oldukça etkilidir. Ayrıca bilgi işlemsel düşünce matematik, fen ve bilgisayar alanları başta olmak üzere disiplinler arası yaklaşımların kullanıldığı proje tabanlı öğretim süreçlerini de destekleyen bir beceridir (Gülbahar, Kalelioğlu, Doğan & Karataş, 2020). Bilgi işlemsel düşünme becerisinin günden güne artan önemi beraberinde kodlama eğitimini de ön plana çıkarmaktadır (Akçay, Karahan & Türk, 2019). Öğrencilerde bilgi işlemsel düşünme ve 21.yüzyıl becerilerini kazandırmaya yönelik oyun tasarımı, robotik ve kodlama gibi alanlarda teknoloji destekli pek çok çalışma yapılmaktadır (Tutulmaz, 2019). Tüm bunların yanında bilgi işlemsel düşüncenin öğretiminde dijital ortamlar veya fiziksel araçlar kullanılabileceği gibi, düşünmeye odaklı bilgisayarsız etkinlikler de kullanılabilmektedir (Gülbahar, Kalelioğlu, Doğan & Karataş, 2020). 21.yüzyıl becerilerin öğrencilerde geliştirilebilmesi adına kodlama eğitiminin oldukça önemli bir yeri olduğu düşünülmektedir (Akpınar & Altun, 2014). Kodlama kavram olarak; genel anlamda istenilen işlerin, birtakım komutlar kullanılarak bilgisayara yaptırılması olarak tanımlanabilir (Akkaş Baysal, Ocak & Ocak, 2020).

Lau ve Yuen (2011), kodlama öğrenmenin; bireylere iletişim, sistemli düşünme, işbirlikli çalışabilme, problemin analizi ve çözüm yöntemleri geliştirebilme şeklinde kazanımları olduğunu ifade etmişlerdir. Akpınar ve Altun (2014), kodlama ile öğrencilerin uzamsal düşünme becerilerinin yanında dijital okuryazarlıklarının ve bilgisayar ile öğrenme becerilerinin de gelişebileceğini belirtmiştir.

Bilgi işlemsel düşünme becerisi, herhangi bir problemin adımlar halinde çözümünü oluşturma(algoritma), çözümün benzer problemlerle ilişkilendirilmesi (örüntü oluşturma), karmaşık problemleri çözülebilir alt parçalara bölme (problemi ayrıştırma) ve problemlerin çözümünde bilgisayarların da desteğinin olması durumu(otomasyon) olarak da tanımlanmıştır (Yadav, Hong & Stephenson, 2016). Bilgi işlemsel düşünce ile birlikte öğrenenlerin kazandığı tüm bu beceriler kodlama eğitimi ile doğrudan ilişkilidir. Çünkü bilgi işlemsel düşünme becerisinin tanımlarında yer alan algoritma kavramı, kodlamanın da temelini oluşturmaktadır. Algoritma kavramı bir görev veya bir problemin çözüm adımlarının detaylı bir şekilde sıralanması olarak tanımlanabilir (Saygıner, 2017). Kodlama sürecinde, bir problemin çözümü için geliştirilen algoritmalar programlama dilleri aracılığı ile komutlara dönüştürülür (Çamoğlu, 2009). BİD becerisinin temelinde de benzer şekilde bir problemin çözümünü mantıklı bir sıralama ile adımlar halinde belirlenmesi durumu vardır. Bu sebeple bireylere BİD becerisinin kazandırılması noktasında kodlama eğitimi öne çıkmaktadır. Resnick (2013), BİD becerisini kazanmanın yanında kodlamayı öğrenmenin, bireylere problem çözme, proje oluşturma ve iletişim stratejileri gibi yeterlilikler de kazandırdığını ve bireylerin yaşı, meslekleri ve ilgi alanları fark etmeksizin bu becerilerin herkes için önemli olduğunu vurgulamaktadır. Ayrıca 21. Yüzyıl öğrenenlerine yaratıcılıklarını geliştirip özgün ürünler ortaya koyabilecekleri aynı zamanda içerik düzenleyebilecekleri, paylaşımcı dijital ortamlar sunmak oldukça önemlidir. Bununla birlikte Akçay ve Çoklar (2016), araştırmalarında kodlamanın analitik ve eleştirel düşünce, çok yönlü düşünerek problem çözebilme becerilerini de geliştirdiğini ifade etmiştir. Tüm bu faydalara ek olarak erken yaşta alınan kodlama eğitiminin bireyin zihinsel gelişimine de destek olabileceği düşünülmektedir (Demir & Demir 2021). Bu kapsamda kodlamanın bilgi işlemsel düşünme becerisini geliştirmeye etkisinin ve öğrenenlere sağladığı faydaların farkına varan ülkeler kodlama eğitimini müfredatlarına entegre etmişlerdir. Özellikle Avrupa ülkelerinde kodlama eğitiminin erken yaşta verilmesi ile problem çözme ve mantıksal düşünme becerilerinin geliştirilmesi hedeflenmektedir (European SchoolNet, 2015). Avrupa birliğinin, kodlama alanında farkındalık yaratmak ve bireylerde kodlama yeteneğinin geliştirmesini sağlamak amacı ile dünya çapında düzenlediği Avrupa Kod Haftası'na (EU CodeWeek) 2019 yılında 4,2 milyon kişi ve 80' den fazla ülke katılmıştır (CodeWeek, 2020). Ülkemizde de son zamanlarda ilköğretim ve ortaöğretim müfredatına programlama dersleri dâhil edilmektedir. Ülkemizde ilköğretimde

verilmekte olan Bilişim Teknolojileri ve Yazılım dersi müfredatı ile öğrencilerin ders kapsamında kodlama yapmalarına olanak tanınmaktadır. Eryılmaz (2003) kodlamayı, analizi ve tasarımı yapılmış bir problemin çözümü için adımlar oluşturularak programlama dili aracılığı ile bilgisayara aktarılması olarak tanımlamıştır. Kodlama kavramı en genel tanımı ile bilgisayara karmaşık problemlerin çözümüne ilişkin komutlar verme sürecidir.

Kodlama kalıplarında yer alan sıralama ve algoritmalar, öğretilen pek çok beceriyi güçlendirmekte ve dersin kazanımları ile bireylerin 21.yüzyıl için gerekli olan zihin alışkanlıklarını kazanmaları hedeflenmektedir. Kodlama eğitimi, hedef kitleye uygun kodlama ortamı, öğretim yöntemi ve programlama dili gibi bazı sorunları da içermektedir (Erdem, 2018).

Kodlama eğitimini ilk kez alan bireylerde metin tabanlı programlama dillerinin karmaşıklığından ötürü öğrenme güçlüğü olabilmektedir. Programlama dillerinin kendine ait yazım kuralları ve soyut kavramsal kalıpları öğrenenlerin ilk zamanlarda zorlanmalarına neden olabilmektedir (Bala, 2019). Bu durum öğrencilerin derse yönelik motivasyonları da etkilemekte ve kodlamaya karşı olumsuz tutum geliştirmelerine sebep olmaktadır. Öğrenenlerdeki bu sorunu ve somutlaştırma problemini ortadan kaldırmak aynı zamanda öğrencilerin kodlamaya karşı olumsuz tutum geliştirmesini önlemek amacı ile görselliği ön planda tutan ve öğrenilmesi daha kolay blok tabanlı kodlama ortamları/araçları geliştirilmiştir (Çatlak, Tekdal & Baz, 2015; Yılmaz İnce, 2020). Bu araçların amacı kodlama sürecini görselleştirerek kod yazmayı kolaylaştırmak ve öğrencilerin bu süreci sevmesini sağlamaktır (Bergin & Martinez, 1996). Ayrıca ilköğretim kademesindeki öğrencilerin yaşları göz önüne alındığında blok tabanlı görsel kodlama araçları öğrenciler için daha ilgi çekici bir öğrenme ortamı oluşturmaktadır (Uzunboylar, 2017). Bununla birlikte Garneli, Giannakos ve Chorianopoulos (2015) metin tabanlı dilleri daha iyi öğrenebilmek ve kodlama için eğlenceli aynı zamanda etkin bir öğrenme sağlanabilmesi için görsel tabanlı programlama dillerinin kullanılmasının daha yararlı olacağını belirtmişlerdir. Blok tabanlı kodlama ortamları öğrencilerin kod bloklarını sürükleyip bırak yöntemi ile tıpkı bir yapboz parçası gibi birleştirip animasyon ve oyunlar oluşturduğu ve eğlenerek öğrendikleri derleyicilerdir (Haymana & Özalp, 2020). Blok tabanlı araçlar sayesinde kodlama eğitimi çok daha zevkli ve daha basit bir hale gelmektedir (Ünsal, 2019). Blok

tabanlı kodlama araçları, küçük yaş grubundaki öğrencilerin seviyesinin üzerinde karmaşık bir yapıya sahip geleneksel programlama dillerini öğrenmelerine gerek kalmadan, kendi uygulamalarını geliştirebilmelerine olanak sağlamaktadır. Bu araçlar ile öğrenciler kod bloklarını sürükleyerek birbirine bağlayıp deneme yanılma ile kod yazarken aynı zamanda sözdizimi kuralları ile de uğraşmak zorunda kalmazlar (Göncü, Çetin & Şendurur, 2020). Kodlama kavramı son yıllarda daha çok “Scratch, Alice, kodu, code.org, blocky,” vb. blok tabanlı programlama araçları ile gündeme gelmiştir. Araştırmada kullanılacak olan blok tabanlı kodlama aracı Blocky programlamayı JavaScript kullanarak blok temelli olarak anlatmaktadır. Öğrenme ortamında bulmacalar ve bulmacaları tamamlamak için kod blokları bulunmaktadır. Tamamen ücretsiz olması ve öğrencilerin seviyelerine uygulduğu bakımından blocky tercih edilmiştir.

Çalışmanın bu bölümünde araştırmanın problem durumu, araştırmanın amacı, araştırmanın önemi, varsayımlar, sınırlılıklar ve tanımlamalar hakkında bilgi verilmiştir.

1.1 Problem Durumu

Teknolojinin içine doğan bir nesil olan z kuşağı, hayatlarının hemen hemen her alanında geleneksel yöntemler dışında farklı beklentilere girmektedirler. Bu anlamda kodlama eğitimi z kuşağının hayal güçlerini beklentileri ile harmanlayıp tasarım yeteneklerini de geliştirebilecekleri bir fırsat olarak görülmektedir (Yünkül, Durak ve Çankaya, 2018). Bunun yanında kodlama; bireylere yaratıcılık imkânı sunarken aynı zamanda yenilikleri de harekete geçirerek; matematik, fen bilgisi gibi branşlarda da çocukların başarılarını desteklediği için özgüvenlerini de artırmaktadır (Bilge Adam Koleji, 2017). Küçük yaşlarda verilen kodlama eğitimi ile bireylerde analitik düşünme, problem çözme ve bilgi işlemsel düşünme gibi 21. Yüzyıl becerileri geliştirilirken aynı zamanda bilgi teknolojilerine olan yatkınlığın da arttığı görülmektedir (Şahin, 2019). BİD ile bireyler öğrenme süreçlerindeki problemlerini çözebilme, çözüm için algoritmalar oluşturarak bu algoritmaları benzer problemlere uyarlama becerileri kazanırlar. Bu becerinin öğrencilere kazandırılmasında önemli bir rol oynayan kodlama eğitiminin erken yaşlardan başlayarak verilmesinde öğrenilenlerin transferini sağlamaları bakımından ilköğretim kurumları oldukça önem arz etmektedir (Kert ve Uğraş, 2009). İlköğretim kademesindeki öğrenciler ilk kez karşılaştıkları bir disiplin olmasından kaynaklı kodlama mantığını kavramakta zorlanabilmektedirler. Bu sebeple verilecek eğitimin niteliği kadar kullanılacak araçların ve yöntemin hedef kitleye

uygunluđu da son derece önemlidir. Bu bakımdan blok tabanlı görsel programlama ortamları kodlama ile yeni tanışan bireylere, problemleri küçük parçalara ayırarak karmaşık ilişkileri analiz edip mantıksal çözüm yolları oluşturma ve bir programlama dilinin temel yapılarını kullanabilme becerisi kazandırmaktadır (Çağıltay ve Fal, 2014). Blok tabanlı kodlama ortamlarında küçük yaşlardaki çocukların kodlama kavramlarını öğrenmesinin, aynı zamanda BİD becerisinin kazandırılmasında da destek sağlayacağı düşünülmektedir (Tağci, 2019).

Bu araştırmada bilgi işlemsel düşünmenin tüm boyutları göz önünde bulundurularak, blok tabanlı kodlama öğretiminin bu düşünme becerisine ve kodlama öğrenmeye yönelik tutuma etkisinin ölçülmesi amacıyla deneysel bir araştırma yapılacaktır.

Tezin konusu Bilişim Teknolojileri ve Yazılım dersinde blok tabanlı kodlama eğitiminin öğrencilerin bilgi işlemsel düşünme becerisi ve kodlama öğrenmeye yönelik tutumları üzerindeki etkisinin incelenmesidir. Bu kapsamda ilköğretim 5. sınıf öğrencilerine Bilişim Teknolojileri ve Yazılım dersinde 8 hafta boyunca blok tabanlı kodlama öğretimi Blocky yardımıyla verilecektir. Uygulanacak ön test ve son testlerden sonra toplanan veriler ışığında blok tabanlı kodlama öğretiminde blocky uygulamasının, öğrencilerin bilgi işlemsel düşünme becerisi öz yeterlikleri ve kodlama öğrenmeye yönelik tutumları üzerine etkisi incelenecektir.

1.2 Araştırmanın Amacı ve Önemi

Bu araştırmanın amacı Bilişim Teknolojileri ve Yazılım dersinde blok tabanlı kodlama eğitiminin, ortaokul öğrencilerinin bilgi işlemsel düşünme becerisi öz yeterlikleri ve kodlama öğrenmeye yönelik tutumlarına etkisini incelemektir. Bu amaç doğrultusunda aşağıdaki alt amaçlara yanıtlar aranacaktır.

Blok tabanlı kodlama eğitimi alan;

- 1- Öğrencilerin uygulanan bilgi işlemsel düşünme becerisi öz yeterlik algısı ölçeğinden aldıkları puanlar ön testlere kıyasla son testlerde anlamlı bir farklılık göstermiş midir?
- 2- Öğrencilerin uygulanan kodlama eğitimine yönelik tutum ölçeğinden aldıkları puanlar ön testlere kıyasla son testlerde anlamlı bir farklılık göstermiş midir?

Bilgi işlemsel düşünme ile öğrenciler bilgisayarlarla çözümlerini otomatik hale getirip problemleri daha etkili çözümlerle düşünmenin sınırlarını genişletebilmektedir (Gülbahar, Kert ve Kalelioğlu, 2019). Bilgi işlemsel düşünme, sadece bilgisayar bilimcileri, mühendisleri veya matematikçileri değil, hemen hemen herkesi ilgilendiren ve çocukluğun ilk dönemlerinden itibaren kazanılması gereken bir beceri olarak kabul edilmektedir (Wing, 2006, 2008).

Oldukça hızlı gelişen bilgi iletişim teknolojileri ile birlikte çocuklar teknoloji ile daha erken yaşlarda tanışmakta ve telefon, tablet vb. araçların kullanımını okul öncesi kademesine kadar yayılmaktadır. Yaşamımız teknoloji odaklı hale geldikçe, çocuklara teknolojik cihazları nasıl kullanacaklarını öğretmekten öte onlara bilgisayar programları ve oynadıkları oyunları çeşitli yönlendirmeler ile kendilerinin nasıl geliştirebileceklerini öğretme yolunda ilerleme gerekliliği ortaya çıkmaktadır. Kodlama eğitimi ile öğrenciler tüketen konumdan üreten konuma geçebilir. Çağımızda kodlama eğitimine büyük önem verilmekte ve bilgi işlemsel düşünme becerilerinin kodlama eğitimi ile öğrencilere kazandırılması konusunda birçok ülke çalışmalar başlatmaktadır. Ayrıca bireylere yazılım alanında verilen eğitimin ülkelerin teknolojik anlamda gelişmesini sağladığı düşünülmektedir (Yılmaz İnce, 2020). PISA sonuçları incelendiğinde Finlandiya, Yeni Zelanda, İngiltere, ABD gibi ülkeler kodlama eğitimlerine müfredatlarında yer vererek çağın becerilerini öğrencilere kazandırma konusunda hızlı ilerlemeler göstermektedirler (Akpınar ve Altun, 2014). İngiltere 2013 yılında, BİT dersinin içeriğinde güncelleme yaparak kodlama eğitimine daha fazla yer ayırmış, sınıf öğretmenlerinin de bu konuda eğitilmesini sağlamıştır (Şenol, 2019). Ülkemizde de MEB bu konuda çalışmalar yaparak müfredatta bazı güncellemelere gerek duymuştur. Öğrencilere 21.yüzyıl becerilerin kazandırılması için programlama eğitimi, 5. ve 6. sınıflarda zorunlu 7. ve 8. sınıflarda seçmeli olarak okutulan Bilişim Teknolojileri ve Yazılım dersi müfredatına dâhil edilmiştir. BTY dersinin genel amaç ve kazanımları incelendiğinde bilişim teknolojilerini kullanarak iletişim kurma, bilişim okuryazarlığı, bilgi paylaşma, araştırma ve bilgiyi yapılandırma, işbirlikli çalışma, problem çözme, programlama ve özgün ürün geliştirme becerilerinin kazandırılmasının amaçlandığı görülmektedir (MEB, 2012).

Öğrencilerin karmaşık kavramları doğrudan kodlama etkinlikleri kapsamında öğrenmesi pek çok sorunu da beraberinde getirebilir ve kodlamaya karşı olumsuz tutum

geliştirmelerine sebep olabilir. Öğrencilerdeki kavramsal bilgi eksikliği kullanılan programlama dilinin kuralları ile birleştiğinde, kodlama ve problem çözme süreçlerinde daha çok zorlanmalarına neden olabilir (Kalelioğlu, 2015). Özellikle işlem öncesi ve somut işlemler döneminde olan çocukların kodlama öğretimi sürecinde birçok eylem ve kavram öğrenenler için soyut kalmaktadır (Ersoy, Madran ve Gülbahar, 2006).

Bu sebeple çocuklara ilk kodlama deneyimlerinin farklı yöntemler ile desteklenerek eğlenceli bir şekilde öğretilmesi, onların ileride karşılaşacakları karmaşık programlama dillerini ve mantıksal akıl yürütmeyi daha kolay öğrenebilmelerini sağlayacaktır (Scharf, Winkler & Herczeg, 2008). Kodlama eğitimini daha zevkli, eğlenceli ve kolay öğrenilebilir hale getirmek amaçlı geliştirilen blok tabanlı kodlama (BTP) araçları giderek yaygınlaşmaktadır. BTP, karmaşık metinsel ifadelerin yerine onları temsil eden bloklar ve grafiklerle kodlama eğitimi için daha eğlenceli bir yol olarak görülmektedir (Çavdar, 2018). Kodlama öğretimini her yaşta bireye ulaştırmayı hedefleyen çevrimiçi görsel tabanlı platformlardan bazıları şunlardır: ScratchJr, Code.org, Mblock, Alice, Blockly, CodeMonkey ve Snap.

Yapılan bu çalışmanın kodlama öğretiminde blok tabanlı kodlama araçlarının, bilgi işlemsel düşünme becerileri ve öğrencilerin kodlama tutumları üzerine etkisi ve blocky konuları bakımından alana katkı sağlayacağı ve bu konuda yapılacak yeni araştırmalar için de bir yol gösterici olacağı öngörülmektedir.

1.3 Sayıtlar (Varsayımlar)

- 1- Seçilen örneklemin araştırmanın evrenini temsil edecek büyüklükte olduğu ve kullanılan ölçme araçları ve yöntemlerinin araştırmanın amaçlarına ulaşabilmesi için uygun olduğu kabul edilmiştir.
- 2- Araştırmanın verilerinin toplandığı bütün aşamalarda güvenilir ölçümler gerçekleştirilmiştir.
- 3- Uygulama süresince fiziksel ortam değişkenlerinin öğrencileri aynı oranda etkilediği kabul edilmiştir.

1.4 Sınırlılıklar

Bilişim Teknolojileri ve Yazılım dersinde blok tabanlı kodlama eğitiminin ortaokul öğrencilerinin bilgi işlemsel düşünme becerileri ve kodlama öğrenmeye yönelik

tutumlarına etkisinin blocky örneđi ile araştırıldıđı bu araştırma; 2019-2020 öğretim yılı, Aksaray ilinde öğrenim gören 5. Sınıf öğrencileri ile 6 haftalık uygulama süreci ve Blocky isimli blok tabanlı programlama aracının kullanılması ile sınırlıdır.

1.5 Tanımlar

Blok Tabanlı Programlama: Metin tabanlı klasik programlama dilinden farklı olarak komutları görsel blokların temsil ettiđi programlama yaklaşımı

Kodlama: Bir amaca ulaşmak için geliştirilen algoritmayı belirli bir programlama dili ile yazma

Algoritma: Bir problemin çözümünde izlenecek yolun mantıksal sıralama dahilinde adım adım yazılması.

Bilgi İşlemsel Düşünme: Bir problemin çözümünde bilgi iletişim teknolojileri araçlarından da faydalanarak en etkili çözüme ulaşmak için muhtemel çözümlerin tanımı, analizi ve gerekli adımların belirlenerek uygulanması.

Programlama Dili: Bilgisayar dili olarak isimlendirilen özel birtakım semboller ile kelimelerin bir araya gelerek oluşturdukları komutlar bütünü.

BÖLÜM 2

2. KURAMSAL ÇERÇEVE VE ALAN YAZIN (İLGİLİ ARAŞTIRMALAR)

2.1.Kodlama Kavramı ve Kodlama Eğitiminin Önemi

İçinde bulunduğumuz 21. yüzyılda oldukça hızlı gelişim gösteren bilgi iletişim teknolojileri, bu gelişmeler ile birlikte pek çok yeni kavramı da beraberinde getirmiştir. 21. Yüzyıl becerileri ve teknolojinin odağında hayatımıza giren kavramlardan biri de kodlama kavramıdır. Kodlama terimi, eğitsel olarak ilk kez 60'lı yıllarda Seymour Papert'in geliştirdiği ve çocukların bilgisayarda çizimler yapabildiği, müzik besteleyip, oyun yazabildikleri Logo programlama dili ile kullanılmıştır (Garc a-Pe alvo, Reimann, Tuul, Rees & Jormanainen, 2016). İlgili pek çok araştırmada programlama ve kodlama terimlerinin aynı kavramı karşılamak için kullanıldığı görülmektedir. European Schoolnet' in yayınlamış olduğu "Computing Our Future" adlı raporda programlama ve kodlama kavramlarının birbirlerinin yerine kullanılabildiği gibi kodlama kavramının daha çok kullanıldığı belirtilmiştir (Balanskat & Engelhardt, 2015). Programlama, en genel tanımı ile bilgisayara gönderilen komutlar toplamıdır (Demirkol, 2016). Bir başka ifadeyle programlama, analiz edilerek tasarlanmış bir problemin çözümüne ait adımların bir programlama dili kullanılarak bilgisayar ortamına aktarılmasıdır (Eryılmaz, 2003). Bir amaç dahilinde programlama dillerinin kendilerine özgü yazım kuralları çerçevesinde komutların yazılıp çalıştırılması süreci kodlama olarak tanımlanmaktadır (Ersoy, Madran ve Gülbahar, 2011). Ülkar (2016) kodlamayı, dijital ortamda uygulamalar oluşturmak, web siteleri tasarlamak ve programları oluşturmak için bilgisayarda ilgili dilde komutlar yazmak şeklinde tanımlamıştır. Akyüz (2018), programlamayı karmaşık problemlerin çözümünde küçük parçaları birleştirerek uygulanan yararlı bir zihinsel etkinlik olarak tanımlamış ve programlamanın çocuklarda düşünceleri organize edebilme, parça bütün ilişkisini fark etme ve ifade edebilme gibi yeteneklerin gelişiminde katkı sağlayabileceği ifade etmiştir. Literatürde yer alan tanımlar incelendiğinde kodlama kavramı, problemlerin çözümü için birtakım kurallar dahilinde yazılan kod satırları ile bilgisayar uygulamaları oluşturma süreci olarak tanımlanabilir. Bayman ve Mayer' e göre (1988), kod yazım sürecinde gerekli olan üç temel bilgi türü vardır: Söz dizimsel, kavramsal ve stratejik bilgi.

1- Söz dizimsel (Syntactic) Bilgi: Programlama dilinin yazım biçimine ait bilgidir.

2- Kavramsal (Conceptual) Bilgi: Programlama dilleri arasında çok fazla farklılık göstermeyen, programlama öğretiminde yer alan kavramlara ait bilgidir.

3- Stratejik (Strategic) Bilgi: Bir problemin çözümü için gerekli olan problem çözme becerisidir. Söz dizimsel ve kavramsal bilgi, bu bilgi türü için ön koşul niteliğindedir. Stratejik bilgi, bu bilgiler de kullanılarak problemin çözümüne yönelik etkili bir algoritma geliştirme sürecinde kullanılan bilgi türüdür.

Kodlama becerisi, matematik ve okuma yazma gibi günümüz dijital dünyasında, temel bir gereksinim halini almıştır (Uzunboylar, 2017). Kodlama öğrenmek, bireylere kod yazarak problem çözmenin yanı sıra yaratıcı düşünmeyi, işbirliği içinde çalışabilmeyi, analiz yeteneği ve eleştirel düşünebilmeyi de kazandırmaktadır. Ayrıca kodlama becerisi, teknolojinin hızlı gelişimi içinde yetişen bireylerin var olan yazılımları kullanmak yerine, karşılaştıkları problem durumlarına çözümler üretecek yeni yazılımlar geliştirebilmeleri açısından da önem taşımaktadır (Tağci, 2019). Kodlama, problem çözme süreçlerini de içerisinde barındıran bir beceridir (Somuncu, 2021). Kodlama becerisi ile bireyler, problem çözme süreçlerinde akıl yürütme becerisini de kullanarak sıfırdan bir ürün ortaya koyabilirler. Kodlama öğretimi çocuklara teknolojik becerileri geliştirirken aynı zamanda; yaratıcılık ve kritik düşünme, problem çözerken planlama, karar verme ve değerlendirme, takım çalışması ile iletişim becerileri gibi hayatın her alanında gerekli olabilecek becerileri geliştirme imkânı sunmaktadır (Uzunboylar, 2017). Bu beceriler ile aynı zamanda çocukların ders başarıları da olumlu yönde etkilenmektedir. Tüm bu getirileri ve bu konuda yapılan çalışmalar göz önünde bulundurulduğunda kodlama öğretiminin erken yaşlarda verilmesi gerekliliği ortaya çıkmaktadır. Erken yaşlarda kod yazma becerisi kazanan çocuklar, ilerleyen dönemlerde kendi yazılımlarını geliştirebilecektir (Demirer ve Sak, 2016). Çocukluk döneminden başlayarak verilen kodlama eğitimini, süreç ve ürün olarak ayrı şekilde değerlendirdiğimizde keşfetmeye dayalı yapılandırmacı öğrenme ve işbirlikli öğrenme sürecine paralel bir altyapı oluşturduğunu ve düşünme becerilerine de olumlu katkılar sağlayabileceğini ifade edebiliriz (Kert ve Uğraş, 2009). Çocukların mantıksal akıl yürütme becerisini kazanabilmesi ve ileride karşılaştıkları karmaşık programlama dillerini daha kolay öğrenebilmeleri için ilk kodlama deneyiminin erken yaşlarda eğlenceli yollarla verilmesinin faydalı olacağı düşünülmektedir (Scharf, Winkler & Herczeg, 2008). Erken yaşlarda kodlama eğitimi almış olan öğrenciler,

gelecekte yazılım alanında eğitimlerini devam ettirmeseler dahi, bu eğitim onlara hayatta pek çok alandaki başarılarına katkı sağlayacaktır (Karabak ve Güneş, 2013). Portelance (2015) yaptığı araştırmada erken çocukluk döneminde verilen kodlama eğitiminin çocukların bilgi işlemsel düşünme becerilerini geliştirebilmeleri için fırsatlar yarattığını belirtmiştir. Kalelioğlu (2015) ise tüketen bir toplumdaki üreten bir toplum yapısına geçebilmek için çocuklara erken yaşlarda kodlama eğitimi verilmesi gerektiğini, bu eğitimin aynı zamanda öğrencilerin akademik başarısına katkı sağlayacağını ifade etmiştir. Bu görüşlere ek olarak erken yaşta verilen kodlama eğitimi çocukların 21.yüzyılda toplumun gelişmesinde önemli rol oynamalarına olanak tanıyan bir şeyler üretebileceklerinin de farkına varmalarını sağlayacaktır (Durak, Karaoğlu, Yılmaz, Yılmaz ve Seferoğlu, 2017). Öğrenciler farklı disiplinlerde öğrendikleri konuları eğitici animasyon şeklinde kodlayarak hem bir ürün ortaya koyup hem de öğrendiklerini pekiştirebilirler (Haymana ve Özalp, 2020).

Kodlama becerisi kazanmış bireylerin, donanımları ile ülkelerinin vizyonuna ve gelişimine önemli ölçüde katkı sağlayacağı düşünülmektedir (Meccawy, 2017). Erken yaşlardan başlayarak verilen kodlama eğitiminin, dijital ekonomi konusunda kalkınmaya da yardımcı olacağı düşüncesi ile birçok ülke bu alanda girişimlere başlamıştır (Bülbül Şoltan, 2018). Avrupa’da genç ve yetişkin bireyleri kodlama ile bir araya getirmek ve kodlamayı yaygınlaştırmak amaçlı 2013 yılından bu yana Avrupa Kod Haftası ismi ile etkinlikler düzenlenmektedir (Atiker, 2019). Çin de kodlama eğitimi konusunda gerekli girişimlerde bulunarak Pekin’de “Tarena Öğrenim Merkezi” ismi ile ilk genç kodlama sınıfını kurmuştur. Başlangıçta yaklaşık 50 öğrenci ile kurulan bu sınıfın 4000 öğrenciye ulaşması hedeflenmiştir. Bu girişimle birlikte çocuklar erken yaşlarda rehberlerin yönlendirmesiyle kodlamanın temellerini öğrenerek, ülkelerinin teknolojik olarak kalkınmalarına katkı sağlama yönünde adım atmışlardır (Fraye, 2016). İngiltere’de eğitim müfredatı, öğrencilerde algoritmik düşünce ve problem çözme becerilerini geliştirmeyi hedefleyen kazanımlar ile 5-16 yaş aralığını kapsamaktadır (Atiker, 2019). Bireylerin teknolojiyi tüketen konumdan teknoloji üreten konuma geçmelerine fırsat sunan kodlama eğitimi aynı zamanda sayısal düşünme becerilerini de önemli ölçüde destekleyerek farklı disiplinlere de katkı sağlamaktadır. Yapılan araştırmaların da bu bakış açılarını desteklemesi ve birçok ülkenin eğitim gündeminde kodlamanın önemli yer tutmasıyla birlikte ülkemizde de eğitim müfredatının kodlamayı kapsayacak şekilde güncellenmesi gerekliliği ortaya çıkmıştır.

Bununla birlikte öğrencilerin ulaşması hedeflenen kazanımlar da 21. yüzyıl becerileri doğrultusunda değişmiştir. Değişen kazanımlara ulaşabilmek adına günümüz eğitim sistemlerinde çeşitli çalışmalar yapılmaktadır (Pakman, 2018). Bu kapsamda önceki yıllarda ilköğretim kademesinde seçmeli ders şeklinde verilen “Bilişim Teknolojileri” dersi, ismi “Bilişim Teknolojileri ve Yazılım” olarak değiştirilerek 5.ve 6. sınıflarda zorunlu ders kapsamına alınmıştır. Ders müfredatına eklenen “Problem çözme ve algoritma” ünitesinin alt kazanımları incelendiğinde gelişen teknoloji beraberinde öğrencilerin kazanması hedeflenen yeni becerilerin eklendiği görülmektedir (Pakman, 2018). Ülkemizdeki ve yurt dışında kodlama ile ilgili yürütülen çalışmalar ve “Bilişim Teknolojileri ve Yazılım” dersi detaylı olarak araştırmanın ilerleyen başlıklarında verilmiştir.

Kodlama eğitiminde metin tabanlı yapılar ve soyut kavramlar öğrenimi güçlendirebilmektedir (Haymana ve Özalp, 2020). Bu durum kodlamanın öğrenciler tarafından karmaşık ve zor bir süreç olarak görülmesine sebep olmaktadır. Özellikle kodlama eğitimine yeni başlayan bireyler kodlama derslerini zor olarak nitelendirmekte ve karmaşık bulmaktadır (Bennedsen & Caspersen, 2008; akt Erol, O. 2015). Öğrencilerin kodlamayı zor bir beceri olarak görmesinin nedenlerinden biri ilk kez karşılaştıkları bir ders olması olarak düşünülebilir (Jenkins, 2002). Kodlamanın zor olarak algılanmasının önemli sebeplerinden biri de soyut ve matematiksel bir beceri olmasından dolayı öğrencilerin önyargılı yaklaşmasıdır. Bu nedenler beraberinde hayal kırıklığı, başarısızlık ve motivasyon eksikliği gibi durumları ortaya çıkarmaktadır. Öğrenenlerin kodlamayı karmaşık bir süreç olarak görmesi sebebiyle, kodlamanın temelini ve algoritma mantığını kavratmaya yönelik görsel blok tabanlı ortamlar geliştirilmiştir (Akyol Altun, 2018). Görsel blok tabanlı kodlama ortamları araştırmanın ilerleyen başlıklarında detaylı olarak incelenmiştir.

2.2.Kodlama Eğitimi Üzerine Yapılmış Çalışmalar

21.yüzyıl becerileri ekseninde gündeme gelen ve birçok ülkenin farklı kademelerde eğitim müfredatlarına da entegre ettiği kodlama üzerine pek çok araştırma yapılmıştır. Bu araştırmalar yurt içi ve yurt dışı araştırmalar olarak aşağıdaki başlıklarda incelenmiştir.

problem çözüme ve temel programlama becerilerini kazandırmada etkili olduğu sonucuna varılmıştır.

Yükseltürk ve Altıok (2016), yaptıkları çalışmada bilişim teknolojileri öğretmen adaylarının kodlama öğretiminde Scratch aracının kullanılması üzerine algılarını incelemeyi amaçlamışlardır. 159 öğretmen adayı ile yürütülen çalışmada anketler ve grup görüşmeleri ile veri toplanmıştır. Toplanan veriler neticesinde öğretmen adaylarının, Scratch aracı ile kodlama öğretme konusunda kullanışlılık ve motivasyon konusunda olumlu fikirleri olduğu görülmüştür.

Yükseltürk, Altıok ve Üçgül (2016), yaptıkları çalışmada “Kendi Oyunumu Programlıyorum” temalı yaz kampındaki etkinlikler ile problem çözme becerisi arasındaki ilişkiyi incelemişlerdir. Bu betimsel çalışmanın örneklemini 6. ve 7. sınıfa devam eden 25 öğrenci oluşturmuştur. Kampın başında ve sonunda öntest ve sontest olarak Çocuklar İçin Problem Çözme Envanteri (Serin, Bulut Serin ve Saygılı, 2010) uygulanmıştır. Öntest ve sontestlerden elde edilen veriler incelendiğinde öğrencilerin problem çözme becerilerinde anlamlı bir farklılaşma olmadığı görülmüştür.

Saygıner ve Tüzün (2017), yurtiçi ve yurtdışı ilköğretim kademesinde kodlama eğitimin durumunu tespit etmek için yaptıkları alan yazın taramasında kodlama eğitimin önemini farkında olduğu sonucuna ulaşılmıştır. Bu araştırmada yurt içinde ve dışında farklı öğretim programlarında hedeflenen kazanımlarda ve kullanılan programlama dillerinde farklılıklar olduğu görülmüştür.

Yecan, Özçınar ve Tanyeri (2017), BTY dersinde görsel tabanlı programlama öğretimi yapan öğretmenlerin ilköğretim kademesinde programlama öğretimindeki görüşlerini, deneyimlerini ve önerilerini belirlemeyi amaçlamıştır. Araştırmada nitel veriler yarı yapılandırılmış görüşme formu ile altı BT öğretmeninden, nicel veriler ise 14 soruluk bir anket ile çeşitli illerde görev yapan 107 BT öğretmeninden elde edilmiştir. Görüşme formundan elde edilen veriler neticesinde öğretmenler; sırası ile son yıllardaki değişimi takip edebilmek, ofis programları öğreniminin öğrenciler açısından sıkıcı bulunması ve giderek değişen öğrenci profiline uyum sağlayabilmek amacıyla programlama öğretimine başladıklarını ifade etmişlerdir. Elde edilen nicel bulgular incelendiğinde; öğretmenlerin programlama öğretimi verme nedenleri sırası ile, öğrencileri yaratıcılığa ve üretmeye teşvik ettiğini düşünmeleri, çağın gerekliliği olarak

bireylerin en az bir programlama dili öğrenmeleri gerektiği, günlük hayatta da problem çözme konusunda programlamanın öğrencilere katkı sağlayacağı, 21.yüzyıl da değişime ayak uydurma gerekliliği, programlamanın ilgi çekici olması ve BTY dersi öğretim programında yer alması şeklinde olduğu görülmüştür.

Akyol Altun (2018) okul öncesi kademesinde beş yaş grubundaki öğrencilere OSMO coding ile temel kodlama eğitimi vererek, bu eğitimin problem çözme becerileri üzerindeki etkisini görmeyi amaçlamıştır. Uygulama öncesi ve sonrası “Problem çözme becerisi ölçeği” ön test ve son test olarak öğrencilere uygulanmış, uygulama esnasında öğrencilere ait gözlemler ise gözlem formu ile toplanmıştır. Araştırma sonucunda elde edilen bulgulara göre temel algoritma ve kodlama eğitimin öğrencilerin problem çözme becerilerinde anlamlı farklılıklar oluşturduğu görülmüştür.

Eryılmaz ve Deniz (2019) yaptıkları çalışma ile Türkiye’de programlama eğitimi üzerine yapılan çalışmalarını incelemeyi amaçlamışlardır. Bu kapsamda 2008-2018 yılları arasında yayımlanan 68 tez ve 78 makale olmak üzere toplamda 146 çalışma betimsel analiz yöntemi ile incelenmiştir. Analiz sonuçlarında göre en çok çalışmanın 2018 yılında yapıldığı ve çalışmaların birçoğunun programlama konusunda BTY dersinde Scratch yazılımı kullanılarak yapıldığı görülmüştür.

Anılan ve Gezer (2020), kodlama etkinliklerine ve analitik düşünme becerisine ilişkin sınıf öğretmenlerinin görüşlerinin incelenmeyi amaçladıkları çalışmalarında 15 sınıf öğretmeninden görüş almışlardır. Çalışmadan elde edilen sonuçlar doğrultusunda, öğretmenlere göre kodlama etkinlikleri ile yaratıcı ve analitik düşünme ve problem çözme becerilerinin geliştirebileceği aynı zamanda analitik düşünce ile kodlama etkinlikleri arasında uyumlu bir ilişki olduğu ifade edilmiştir.

Yapılan araştırmalarda, kodlama eğitiminde öğretmen ve öğrenci görüşleri, öğretimde karşılaşılan zorluklar, kullanılan programlama ortamları, kodlamanın 21.yüzyıl düşünme becerilerine etkisi gibi farklı boyutlar incelenmiştir. Araştırmaların son yıllarda yoğunlaşmış olması ülkemizde bu konunun gün geçtikçe daha fazla önemsendiğinin de göstergesidir. Bu bağlamda milli eğitim politikalarında da bu yönde geliştirmelere gidilmiştir. 2012 yılına kadar “Bilişim Teknolojileri” adında verilen dersin ismi alınan kararla “Bilişim Teknolojileri ve Yazılım” dersi olarak değiştirilmiştir (BTE Derneği, 2013). Bu değişiklikle birlikte güncellenen ders

müfredatına “Problem Çözme ve Programlama” ünitesi altında algoritma ve blok tabanlı kodlama konularının alt kazanımları eklenmiştir (TTKB, 2013). 2016 yılında alınan kararlar ile iki kurdan oluşan “Bilgisayar Bilimleri” dersinin ortaöğretimde kademeli olarak verilmesi kararlaştırılmıştır (MEB, 2016). Dersin Kur 1 ve Kur 2 olarak yayınlanan müfredatı incelendiğinde “problem çözme ve algoritmalar”, “robot programlama”, “mobil programlama” ve “web programlama” gibi konu başlıklarından lise kademesinde de programlama eğitime verilen önem anlaşılmaktadır (MEB, 2016). EBA portalı üzerinden de öğrenciler ve öğretmenler kendi programlarını oluşturabilmekte, oluşturdukları yazılımı paylaşabilmektedirler. Özel okullarda da oldukça önemsenen kodlama eğitimi okul öncesinden itibaren yön kavramları ile basit algoritmalar oluşturarak başlanmıştır (Bülbül Şoltan, 2018).

Milli eğitim dışında özel kurumlar ve sivil toplum kuruluşları da kodlama eğitimine yönelik farkındalık oluşturmak amaçlı birtakım faaliyetler başlatmışlardır. Bu faaliyetlerden biri Bilişim Garaj Akademisi’nin başlattığı 7-8, 9-12 ve 13-16 yaş gruplarına, problem çözme becerileriyle programlama eğitimine farkındalık oluşturmak amacıyla web tasarımı, robot tasarım ve programlama gibi alanlarda verilen eğitimlerdir (Demirer ve Sak, 2016). 2014 yılında Türkiye Bilişim Derneği (TBD) ve bazı üniversitelerin işbirliği ile öğrencilerde oluşan programlamanın zor olduğu algısını yıkmak amacıyla “Bilgisayar Programlama Çocuk Oyuncağı” isimli bir etkinlik düzenlenmiştir (TBD, 2014). Ortak amaçlarla oluşturulan bir diğer proje ise bir mobil operatör şirketi tarafından yürütülmektedir. 2016-2017 eğitim öğretim yılı içerisinde Millî Eğitim Bakanlığının da desteğiyle 7-13 aralığındaki öğrenciler için başlatılmıştır (Hatisaru, 2016). Bu projenin hedeflerinden biri de programlama bilgisine sahip kız öğrencilerin sayısını artırmaktır (Yüzak, 2016). Habitat derneği, özel bir banka ve Microsoft Türkiye işbirliği ile 2015 yılında başlatılan ve ülkemizin farklı illerinden 8-12 yaş aralığındaki çocukları kapsayan “Minik Parmaklar Geleceği Programlıyor” projesi halen devam etmektedir (Habitat, 2017). Proje ile çocukların teknoloji konusunda tüketici konumundan çıkıp üretici konuma geçmelerinin sağlanması, analitik düşünme becerilerini geliştirmek ve kod okuryazarlığı bilincini yaymak hedeflenmiştir (Habitat, 2017). Bu gibi projeler ve eğitimler zaman zaman öğretmenler için de açılmaktadır.

2.2.2.Yurt Dışında Yapılmış Çalışmalar

Son yıllarda teknolojinin eğitime entegrasyonu birlikte öğrenenlere kazandırılması gereken birtakım beceriler de gündeme gelmiş ve ülkelerin pek çoğu bu alanda araştırmalara yoğunlaşmışlardır. 21.yüzyıl becerilerinin kazandırılması konusunda kodlama eğitiminin önemi üzerine de birçok araştırma yapılmıştır.

Bishop- Clark, Courte ve Howard (2007) araştırmalarında programlama dili olarak Alice uygulaması kullanarak gerçekleştirilen bilgisayar dersinin nitel ve nicel veriler kapsamında incelemiştir. Katılımcılara Alice programlama ortamında iki buçuk hafta boyunca eğitim verilmiştir. Uygulanan öntest ve sontestler araştırmanın nicel verilerini, öğrencilerin süreçte kazandıkları deneyimler ise nitel verileri oluşturmuştur. Elde edilen nicel verilere göre katılımcıların özgüvenlerinde, programlama kavramlarını anlama düzeylerinde ve programlama eğitiminden aldıkları keyifte anlamlı bir artış olduğu görülmüştür. Bu bulgular elde edilen nitel veriler ile de desteklenmiştir.

Bir diğer araştırma da Maloney, Peppler, Kafai, Resnick ve Rusk (2008) tarafından, programlama aracı olarak Scratch kullanılarak yapılmıştır. 8-18 yaş grubundaki katılımcılar ile bir yıl boyunca gerçekleştirilen araştırmada Scratch üzerinden oyunlar tasarlanmıştır. “Bilgisayar Kulübü” adı altında okul dışı etkinlikler olarak tasarlanan oyunlar döngü, koşullu ifadeler, değişkenler, boolean mantığı vb. yapıları içermeleri yönünden incelenmiştir. Ayrıca araştırma kapsamında katılımcı görüşlerine de başvurulmuştur. Araştırma sonuçlarında, programlama mantığının kavranması açısından tasarlanan oyunlar içerisinde programlama yapılarının kullanılmasının oldukça önemli olduğu ve Scratch’ in programlama öğretiminde yardımcı bir araç olabileceği görülmüştür. Bunun yanında katılımcılar Scratch’ i matematik ve programlama ile bağdaştırdıklarını ifade etmişlerdir.

Calder (2010) Scratch programlama dilini kullanan öğrenciler ile görüşmeler ve gözlemler yapmış, öğrencilerin blog yazılarındaki görüşlerini incelemiştir. Çalışma neticesinde, Scratch’ in ilgi çekici ve kullanımının kolay olduğu, matematiksel kavramların verilmesinde ve problem çözme konusunda motive edici bir ortam sunduğu görülmüştür.

Meerbaum- Salant, Armoni ve Ben-Ari (2013) çalışmalarında Scratch uygulamasının bilgisayar bilimi kavramlarını öğretip öğretemeyeceğini irdelemişlerdir. Bloom Taksonomisi ve SOLO (Structure of Observed Learning Outcomes) taksonomisine göre hazırlanan anket ve test soruları araştırmanın veri toplama araçlarını oluştururken Scratch' e uygun materyaller de geliştirilmiştir. İki farklı okulda uygulanan anketler ve materyallerin yanı sıra gözlemler de yapılmıştır. Elde edilen bulgulara göre öğrenciler belli kavramlarda zorlansa da Scratch' in bilgisayar bilimi kavramlarını öğretmede etkili bir araç olabileceği bulgusuna varılmıştır. Ayrıca Öğrencilerin zorlandığı kavramların, öğretim yöntemi değiştirilerek tekrar verilebileceği ifade edilmiştir.

Briggs (2013) yaptığı çalışmada Scratch uygulamalarının öğrenenlere sağladığı faydaları incelemeyi amaçlamıştır. Üç ayrı okulda yürütülen çalışmada veriler gözlemler, görüşmeler ve video kayıtları yoluyla toplanmıştır. Araştırma bulgularına göre Scratch ile yapılan etkinliklerin öğrenenlere sağladığı faydalar temelde 12 kategori altında listelenmiştir. Bu kategorilerde bağımsız öğrenmeyi destekleme, keşfetmeye teşvik etme, mantıksal düşünceyi geliştirme gibi başlıkların önde olduğu görülmüştür. Öğretmenin süreçte rehber olarak rol alması öğrencilerin kendi öğrenmelerinin sorumluluğunu almasını ve tasarladıkları oyunlar ile ortaya bir ürün koyabilmelerini de sağlamıştır. Araştırma sonucunda Scratch ortamının programlama eğitiminde etkili bir araç olduğu görülmüştür.

Wyffles, Martens ve Lemmens (2014), çalışmalarında üç ayrı programlama ortamını öğrenci görüşleri ile birlikte incelemişlerdir. Sekizinci sınıf öğrencileri ile yürütülen çalışmada öğrenciler üç gruba ayrılmıştır. İlk gruba Scratch, ikinci gruba Greenfoot ve üçüncü gruba Dwengo programı ile altı haftalık eğitim verilmiş ve bu süreçte öğrencilerin görüşleri alınarak araştırma verileri elde edilmiştir. Elde edilen bu verilere göre Scratch kullanan grupta öğrencilerin programlama sürecini kolay ve eğlenceli buldukları görülmüştür. İkinci ve üçüncü gruptaki öğrenciler ise programlama sürecinin belediklerinden zor olduğunu ifade etmişlerdir.

Chaudhary, Agrawal ve Sureka (2016), yaz kampındaki ilköğretim kademesinden 9 öğrencinin katılımıyla Lego Mindstorms EV3 robot programlama kitini kullanarak bir araştırma yürütmüşlerdir. Araştırmada katılımcıların bir sorun karşısında geliştirdikleri problem çözme yaklaşımlarını, soruna nasıl alternatif çözümler

üretebildiklerini ve iş birliği içerisinde bir ürün ortaya koyabilme süreçlerini belirlemek hedeflenmiştir. Öğrencilere, koşullu ifadeler, döngüler gibi programlama kavramları ve sensörler, motorlar, konnektörler vb. Lego bileşenleri ile robot tasarlayıp görsel programlama ortamında tasarladıkları robotları programlayabilmeleri için eğitimler verilmiştir. Ardından öğrencilere birtakım görevler sunularak görevleri yerine getirdikleri süreç boyunca gözlemler yapılmıştır. Gözlemler ve süreç sonunda öğrencilere sorulan sorulardan kaydedilen cevaplar araştırmanın verilerini oluşturmuştur. Gözlemler neticesinde metin tabanlı programlama ortamları yerine blok tabanlı programlama ortamlarının hedeflenen becerilerin kazandırılmasında daha etkili olduğu sonucuna varılmıştır. Araştırma sonucunda; Lego Mindstorms EV3 robot programlama kitinin çocukların problem çözme, bilgi işlemsel düşünme, işbirlikçi öğrenme, mühendislik ve robot programlama gibi becerilerini geliştirmek adına uygun bir ortam sunduğu görülmüştür.

Bers, González-González ve Armas–Torres (2019), yaşları üç ile beş arasında değişen çocuklar ile robotik uygulamalarla kodlama ve bilgi işlemsel düşünmeyi öğretmeyi amaçlayan bir çalışma gerçekleştirmişlerdir. Çalışmada, KIBO isimli ekransız kodlama yapılabilen programlanabilir oyuncak robot kullanılmıştır. Çalışma bulgularından yola çıkılarak kodlamanın farklı müfredat alanlarına entegrasyonu ile okul öncesi sınıflarında uygun kodlama deneyimleri geliştirmenin mümkün olduğunu ifade edilmiştir.

Kodlama eğitimi konusunda yurt dışında yapılan çalışmalar incelendiğinde kodlama kavramının pek çok farklı boyutta araştırıldığı görülse de çalışmaların birçoğu erken yaşta programlama eğitimi üzerine yoğunlaşmaktadır. Bunun yanında 21.yüzyıl becerilerinin kazandırılmasında kodlama eğitimin önemli bir rolü olduğu da düşünülmektedir (Shin, Park & Bae, 2013). Bu bağlamda gelişmiş ve gelişmekte olan pek çok ülke kodlama eğitimini, küçük yaşlardan itibaren vermeye üzere eğitim müfredatlarına farklı şekillerde entegre etmeye başlamışlardır. Uygulamalarda farklılıklar olsa da temelde hedeflenen kodlama eğitimi ve bilişim teknolojileri becerilerinin kazandırılmasıdır (Tağci, 2019).

Avrupa Okul Ağı (European Schoolnet) 2014 yılının Ekim ayında yayınladığı ve 2015 yılında yeniden güncellenen raporda Avrupa’da ülkelerin kodlamayı müfredatlarına nasıl entegre ettiklerine dair verilere yer verilmiştir. Bu rapora göre 12

ülke kodlamayı BİT dersi içerisinde farklı başlıklar ile verirken; Fransa, Danimarka, İspanya, Slovakya, Finlandiya ve Estonya kodlamayı matematik, fen vb. disiplinlere entegre etmişlerdir (European Schoolnet, 2015). Danimarka’da basit kodlama bilgisine sahip olmak Matematik, Fizik ve Kimya dersleri için bir zorunluluk iken Slovakya’da da eğitimin tüm kademelerinde kodlama becerileri zorunlu olarak müfredata entegre edilmiştir (European Schoolnet, 2015). 2015 verilerine göre Çek Cumhuriyeti, Polonya, Malta ve Litvanya bu yıldan itibaren daha somut adımlarla kodlama eğitimine müfredatlarında daha çok yer verirken Finlandiya da kodlamayı çekirdek müfredatı içerisine yerleştirmiştir (European Schoolnet, 2015). Bulgaristan’da “Algoritmik Problem Çözme ve Programlama”, Belçika’da ise “Bilişimsel Düşünce ve Programlama” başlıkları ile BİT dersi içerisinde öğrencilere kodlama becerileri verilmektedir (European Schoolnet, 2015).

Yazılım konusunda oldukça ilerlemiş olan Hindistan’da eğitim programı incelendiğinde neredeyse bütün kademelerde bilgisayar dersinin bulunduğu görülmektedir. Programlama eğitimi ilköğretim kademesinde “Algoritmanın Temelleri ve LOGO Programlama Eğitimi” olarak, Ortaokul kademesinde ise BASIC programlama dili ile giriş yapılarak değişkenler, döngüler gibi temel programlama kavramları şeklinde verilmektedir. Lise düzeyinde ise ilköğretimde aldıkları temel programlama eğitiminin üzerine katarak ileri seviye programlama yapabilecek şekilde müfredat düzenlenmiştir (SSRVM, 2007).

Macaristan’da yazılım geliştirmenin yalnızca erkeklerin becerisi olmadığını kanıtlamak amacıyla teknoloji firması Prezi’ nin de girişimiyle “Programcı Kızlar” isminde bir proje başlatılmıştır. Sivil toplum örgütlerinin de destekleri ile gün geçtikçe gelişen proje kapsamında ortaokul kademesindeki kız öğrenciler “Processing” programlama dilinde eğitim almaktadırlar (Euronews, 2015).

Güney Kore’de Bilim ve Gelecek Planlama Bakanlığının açıklamasıyla 2014 yılından itibaren 72 okul pilot bölge olarak seçilmiş ve ilköğretimden itibaren programlama dersi zorunlu olmuştur (Erümit, 2018). Aynı zamanda açıklamada, ilköğretim tamamladıkları 2017, liselerde ise 2018 yılından itibaren kademeli olarak programlama derslerinin zorunlu olacağı belirtilmiştir (Özçakmak, 2014).

Estonya’da, 2012 yılında pilot uygulama olarak başlatılan “Progetiiger” adlı proje ile 7-19 yaş arası çocuklar kodlama ile tanışmış ve ardından kodlama eğitiminin 1.sınıftan itibaren verilmesi kararlaştırılmıştır (Olson, 2012).

Kodlama eğitimini küçük yaşlardan itibaren vermeye başlayan bir diğer ülke de Avustralya’dır. Millî Eğitim Bakanlığı eğitim programını, ilkokulda 5 yaşından itibaren iki yıl süreyle temel kodlama eğitiminin ardından sonraki yıllarda ileri düzeyde eğitimler verilerek 7 yaşında bir çocuğun programlama mantığının temeli kavrayabileceği şekilde düzenlemiştir (Kahraman, 2015; Saygıner, 2017).

Fransa’da 2015 yılında kodlama eğitimi üzerine yapılmış bir çalışma sonucunda bu eğitimin okul öncesinden itibaren verilebileceği belirtilmiştir. Eğitim politikalarında bu yaş grubunda verilen eğitimlerde öğrencilerin görsel düşünme becerisini geliştirmek amaçlanmıştır (Euronews, 2015).

İngiltere’de 2013 yılı itibariyle ilkokul kademesinden başlanarak tüm kademelerde kodlama eğitimine başlanmıştır. Bunun devamında 2014 yılı ülke çapında “Kodlama Yılı” olarak kabul edilmiştir (Özdemir, 2015). Eğitim programındaki BİT dersinin müfredatı da çağın gereklerine uygun bir şekilde güncellenerek kapsamlı bir kodlama bölümü eklenmiştir (Saygıner, 2017). BİT dersinin içerisine entegre edilen kodlama eğitimi öğrencilerin yaş gruplarına göre kademelere ayrılmıştır. 5-6 yaş grubu için oluşturulan birinci basamakta algoritma kavramı verilirken, 7-11 yaş grubu için oluşturulan ikinci basamakta karmaşık programlama yapıları oluşturabilme ve hata ayıklama becerileri kazandırılması hedeflenmiştir (Öndeş, 2016). 12-14 yaş grubu için oluşturulan üçüncü basamakta ise artık öğrencilerin birden fazla programlama diline temel anlamda hâkim olması amaçlanmıştır (Öndeş, 2016).

Eğitimde kodlama alanında öne çıkan ülkelerden biri de Amerika Birleşik Devletleridir. Ülkedeki pek çok sivil toplum örgütü, Google ve Microsoft gibi yazılım firmaları da okullarda kodlama eğitimi üzerine yapılan çalışmalara önemli ölçüde destek sağlamaktadır (Saygıner, 2017). Bu çalışmalardan biri olan “code.org” platformunu ABD’de yaklaşık 6 milyon öğrenci kullanırken pek çok farklı ülkeden öğrenciler de bu platforma erişim sağlayıp kendi programlarını yazabilmektedirler (Öndeş, 2016). Ayrıca ülkede Bilgisayar Bilimi Öğretmenleri Derneği – BBÖD

(Computer Science Teacher Association– CSTA) gibi kuruluşlar ile de verilen eğitimlerin yeterlilikleri belirlenmektedir (Tağci, 2019).

Ülkelerin eğitim sistemleri incelendiğinde programlama eğitimlerinin farklı başlıklar altında farklı disiplinler içerisinde verildiği görülmektedir. Yaş seviyelerinden ve programlama eğitiminin verildiği ortamdaki (blok tabanlı, basic, metin tabanlı vb.) farklılıklar oluşmuş olsa da pek çok ülke erken yaşta programlama öğretimini eğitim politikalarına dahil etmişlerdir.

2.3.Blok Tabanlı Görsel Kodlama

Son yıllarda erken yaşta programlama veya kodlama öğrenmeye yönelik eğilimler oldukça artmıştır (Al Jarrah, 2016). Birçok eğitim felsefesi tarafından da desteklenen erken yaşlarda kodlama eğitiminin doğru yollarla verilmesi gerektiği fikri göz ardı edilemez bir gerçektir (Çavdar, 2018). Kodlama eğitiminde bireye sunulan programlama aracının yanlış olması, bireylerin önyargı geliştirmesi ve kodlamaya karşı geliştirebilecekleri olumsuz tutum ile ilginin azalmasına da sebep olabilmektedir (Ünsal, 2020). Bu bağlamda programlama eğitiminde daha geniş bir yaş aralığına hitap edebilmek ve programlama becerisini anlaşılır ve kolay yollardan kazandırabilmek amaçlı birtakım ortamlar ve araçların kullanılma gerekliliği doğmuştur. Genel anlamda bu ortam ve araçlar, metin tabanlı (text-based) ve blok tabanlı (block-based) araç ve ortamlar olarak sınıflandırılabilir (Durak, Karaoğlan Yılmaz, Yılmaz ve Seferoğlu, 2017). Bilgisayar programlama dilleri tıpkı bir yabancı dil öğrenimi gibi zaman alan ve ilk deneyimde kullanıcıya karmaşık gelen dillerdir (Ctrlbizde.com, 2020). Bu sebeple araştırmalarda anlamayı ve de uygulamayı kolaylaştırmak için görsel programlama araçlarının kullanılması önerilmektedir (Erümit ve diğerleri, 2017). Programlama öğrenmeye yeni başlayan bireylerin öğrenimini kolaylaştırmak, daha zevkli hale getirerek programlamayı sevmelerini sağlamak için günümüzde oyun tabanlı öğrenme vb. çeşitli yöntemler kullanılmaktadır (Saygıner, 2017). Blok tabanlı görsel programlama (BTP) bu yöntemlerden biridir. Literatürde Blok tabanlı kodlama olarak da geçmekte ve araştırmada çoğunlukla bu tanımlama ile kullanılmıştır. Blok tabanlı kodlama ortamları ile daha fazla kullanıcıya erişilebilirlik hedeflenmiştir. Blok tabanlı kodlama eğitimindeki farklı öğrenme ortamlarında yer alan resim, müzik gibi çoklu ortam araçları ile çocukların ilgisini çekmekte ve derse katılım için de motivasyonlarını artırmaktadır (Pakman, 2018).

Genel olarak iki amaç için geliştirilen blok tabanlı programlama ortamları için ilk amaç “Programlamadaki sözdizimini basite indirgeyerek temel düzeyde bilgiye sahip herkesin programlama yapabilmesini sağlamak” ve ikinci amaç ise “hedeflenen kitlenin ilgisini çekerek programlamayla ilgili birey sayısını artırmak” tır (Medlock-Walton vd., 2014). Bu ortamlarda resim ve ses gibi birtakım görsel ve işitsel araçlar bir arada kullanılarak soyut yapıdaki değişken, döngü vb. öğrenilmesi zor programlama kavramları animasyonlar ile desteklenerek somutlaştırabilmektedir (Erol, 2015). Böylelikle öğrenme ortamına daha fazla duyu organı dahil edilmekte ve kalıcı bir öğrenmenin sağlanması kolaylaşmaktadır. Grafiksel arayüzü ile çoğunlukla kod yazmayı gerektirmeyen blok tabanlı kodlama ortamlarında kullanıcılar, lego parçalarını anımsatan kod bloklarını sürükleyip bırak yöntemi ile yerleştirerek projeler oluşturabilmektedir. Metinsel ifadelerin yerine kullanılan bu kod blokları ile de daha eğlenceli bir öğrenme sunmaktadır. Çünkü blok tabanlı ortamların, metin tabanlı ortamlara göre daha anlaşılır ve basit bir arayüzü bulunmaktadır. (Aytekin, Sönmez Çakır, Yücel ve Kulaözü, 2018). Bununla birlikte blok tabanlı kodlama ortamları çocukların derleme (compiling) ya da sözdizimi hatalarına (syntax errors) takılmadan karmaşık görsel programlar oluşturabilmesine de olanak sağlamıştır (Durak, Karaoğlan Yılmaz, Yılmaz ve Seferoğlu, 2017). Aynı zamanda blok tabanlı kodlama ortamları sahip olduğu bu özellikler ile çocukların programlama konusunda oluşan bilişsel engellerini de azaltmaktadır (Hill, 2015 akt. Durak, Karaoğlan Yılmaz, Yılmaz ve Seferoğlu, 2017). Blok tabanlı kodlama ile öğrenciler karmaşık kod satırları içinde kaybolmadan kodlamaya ilişkin temel kavramları daha kolay öğrenebilmektedir (EğİN, 2019). Böylece öğrencilerin programlama eğitimine yönelik tutumları da olumlu yönde değişebilmektedir. Bununla birlikte öğrenciler planlama, problem çözme ve benzeri üst düzey kodlama becerileri ile ilgilenmek için daha fazla vakit bulabilirler (Göncü, Çetin ve Şendurur, 2020). Bütün bunlar göz önünde bulundurulduğunda bu ortamlarının özellikle çocuklar için uygun olduğu ifade edilebilir. Weintrop ve Wilensky (2015), yaptıkları araştırmada öğrencilerin BTP ortamlarının programlamaya yeni başlayanlar için tasarlanmış bir öğrenme ortamı olduğunu ve BTP’ nin normal programlama ile aynı fakat daha basit olduğunu söylediklerini belirtmektedir. Bir başka araştırmada da blok tabanlı kodlama ortamlarından oldukça yaygın olan Scratch kullanımının çocukların programlama eğitimlerine katılımları ve programlamayı öğrenme konusunda olumlu yönde etki ettiği sonucuna varılmıştır (Zuckerman, Blau & Monroy-Hernández, 2009). Erol (2015) blok tabanlı kodlama araçlarının motivasyon ve başarıya etkisini incelediği

araştırmasında ise katılımcılar için, yapılan oyun tasarlama etkinliklerin programlamanın mantığını kazandırma ve motivasyonu artırmada etkili olduğunu ifade etmiştir. Çatlak, Tekdal ve Baz (2015), çalışmalarında Scratch ile gerçekleştirilen bilimsel araştırmaları incelemiştir. Araştırmalarda kodlama eğitimine başlangıç aşamasında Scratch programının verilebileceği, soyut kavramların öğreniminde ve kavramlar arası ilişkiyel bağların kurulmasında çoklu ortam öğeleri ile desteklenerek öğrenme sürecinin daha kolay olabileceğine dair bulgulara ulaşıldığı belirtilmiştir. Numanoğlu ve Keser (2017) çalışmalarında blok tabanlı kodlama platformlarının programlama öğretiminde kullanılabilirliğini ve bu platformların yapısını araştırmayı hedeflemişlerdir. Araştırmaları sonucunda blok tabanlı kodlama yazılımlarının programlama öğretiminde soyut kavramları somutlaştırabildiği ve öğrenenlerin bilgi işlemsel düşünme ve problem çözme gibi becerilerini hızlı bir şekilde geliştirebildiği sonucuna varmışlardır. Aydođdu (2020) blok tabanlı programlama etkinliklerinin öğretmen adaylarının programlamaya ilişkin öz yeterlik algıları ve hesaplamalı düşünme becerileri üzerine etkisini incelemeyi amaçladığı çalışmasında 4 hafta süresince blok tabanlı etkinlikler yapılmıştır. Çalışma sonucunda BTP etkinliklerinin öğretmen adaylarında programlamaya yönelik öz yeterlik algısı üzerinde olumlu etkisi olduğu ancak hesaplamalı düşünme becerileri üzerinde herhangi bir etkisi olmadığı görülmüştür. Ünsal (2020) da çalışmasında blok tabanlı kodlama etkinlikleri ile ilkökul 2.sınıf öğrencilerinin bilgi işlemsel düşünme becerilerini incelemiştir. Deneysel desenin kullanıldığı araştırmada, blok tabanlı kodlama ortamı olarak code.org platformunu kullanmıştır. Araştırma sonucunda code.org gibi blok tabanlı ortamlarda verilen kodlama eğitiminin öğrencilerin BİD becerilerini artırdığı görülmüş ve bu ortamların ilkökul kademesinde kullanımı önerilmiştir.

Yapılan araştırmalar neticesinde kodlama konusunda yeterince bilgi sahibi olmayan ve erken yaştaki bireylerin kodlama öğreniminde blok tabanlı kodlama ortam ve araçları yararlı olarak görülmektedir. Bu ortam ve araçlardan bir kısmı sonraki başlıkta detaylı olarak incelenmiştir.

2.3.1.Blok Tabanlı Kodlama Araçları

MIT ve Papert araştırmacılarının 1969 yılında öğrencilere matematiksel ve mantıksal düşünme becerileri kazandırma amacıyla geliştirmiş oldukları Logo programlama diliyle başlayan, günümüzde de yazılım firmaları ile dünyanın önde gelen

üniversitelerinin iş birliği ile geliştirdiği çocuklara 21.yüzyıl becerilerini kazandırmayı hedefleyen ve aynı zamanda programlama öğrenmeyi kolay ve eğlenceli hale getiren çeşitli programlama araçları/ortamları vardır (Erdem, 2018). Bunlara örnek olarak bazı blok tabanlı kodlama araçları/ortamları aşağıda başlıklar halinde detaylı olarak verilmiştir.

2.3.1.1.Scratch

Ücretsiz olarak sunulan ve 8-16 yaş grubu için tasarlanmış bir uygulama olan Scratch yazılımı, MIT tarafından 2003 yılında bir proje olarak başlatılmıştır (Genç ve Karakuş, 2011). Blok tabanlı görsel bir programlama dili olan Scratch ile gençlere ve ilk defa programlama öğrenecek olan bireylere programlama öğretmek amaçlanmıştır (Yükseltürk ve Altıok, 2016; Scratch, 2020). Renkli bir arayüze sahip olan Scratch ortamında; hazır fonksiyonlar kategorilere ayrılmış, böylece programlamaya yeni başlayan bireylerin de karmaşık yapılara takılmadan programlamayı kavrayabilmelerini sağlamaktadır (İzgöl, 2015). Çevrimiçi ve çevrimdışı olarak da kullanılabilen Scratch ile kullanıcılar oyunlar, etkileşimli hikayeler ve animasyonlar oluşturabilmektedir (CodeWizardshq, 2016). Scratch' te kod yazmak yerine kategorilenmiş kod bloklarının sürükle bırak yöntemi ile taşınması, kullanıcıdan kaynaklı yazım hatalarını ortadan kaldırmaktadır. Bu sayede programlamanın temel mantığı, bireyler tarafından daha kolay öğrenilebilmektedir (Atiker, 2019). Ayrıca Scratch kullanıcılarına Türkçe de dahil olmak üzere çok sayıda dil desteği de sunmaktadır.

2.3.1.2.MIT App Inventor

MIT App Inventor, MIT ve Google iş birliği ile Android işletim sistemini kullanan mobil cihazlara yönelik uygulamalar geliştirebilmek için tasarlanmış blok tabanlı görsel bir yazılımdır (Cuiumju, 2013). MIT App Inventor projesi ile tüm bireyleri özellikle de gençleri teknolojiyi tüketen kuşaklar olmaktan çıkararak üreten konuma geçirmek amaçlanmıştır (MIT App Inventor, 2020). Hemen hemen her yaşta kullanıcıya hitap eden App Inventor ücretsiz ve web tabanlı olup, kullanabilmek için g mail hesabına ihtiyaç duyulmaktadır. App Inventor kullanıcıları yazılımı kurmadan da İnternet bağlantısının bulunduğu herhangi bir yerden g mail hesabı ile oturum açarak projelerine devam edebilmektedir (Saygıner, 2017). MIT App Inventor, basit ve görsel olarak sunulan grafik arayüzü ile metin tabanlı kodlamanın karmaşıklığını ortadan kaldırarak, sürükle-bırak mantığı ile uygulama oluşturma konusunda deneyimsiz kişiler

için de kısa sürede uygulama oluşturabilmeye olanak sağlamaktadır (MIT App Inventor, 2020).

2.3.1.3.Code.org

Kodlama ortamları arasında en çok bilinenlerden biri olan code.org 2013 yılında Hadi ve Ali Patrovi tarafından bilgisayar programcılığını teşvik etmek ve daha erişilebilir hale getirmek amacıyla kurulmuş, kâr amacı gütmeyen bir organizasyondur (Code.org, 2020). Microsoft, Facebook, Google gibi birtakım teknoloji firmalarının da desteklediği code.org platformu web tabanlı çalışmaktadır ve <https://code.org/> adresinden erişilebilmektedir. Platform ücretsiz olup 4-14 yaş arası her birey için çeşitli seviyelerde dersler sunmaktadır. Bu seviyeler 4 yaş ve üstü, 6 yaş ve üstü, 8 yaş ve üstü, 10 yaş ve üstü gibi sınıflandırılmıştır. Seviyelerin içerisinde bulunan ISTE standartlarına göre hazırlanmış birçok oyunda bireyler verilen görevleri birçok blok tabanlı kodlama ortamında olduğu gibi sürükle-bırak yöntemi ile yerine getirmektedir. Aynı zamanda kodladıkları oyunun metin tabanlı yazılmış kodlarını da görebilme imkanları vardır. Seviyelerine göre seçtikleri oyunun tüm basamaklarını tamamlayan kullanıcılara sertifika verilmektedir. Code.org çevrimiçi bir platform olduğu için kullanıcılar yaptıkları çalışmalarını da platform üzerinde paylaşabilmektedirler (CodeWizardshq, 2016). Platform, pek çok dil desteğinin yanında Türkçe dil desteği de sunmasıyla geniş kitlelere hitap edebilmektedir.

2.3.1.4.Kodu Game Lab

Microsoft tarafından geliştiren Kodu Game Lab yazılımı ile programlama becerisi olmayan çocuklar ve gençler basit görsel programlama dili ile oyun geliştirebilir, oynayabilir ve geliştirdikleri oyunları arkadaşları ile paylaşabilir (Candar, 2016). Yazılımın içerisinde bulunan hazır özellikler sayesinde herhangi programlama becerisine sahip olmayan çocukların ve gençlerin yanı sıra bir oyun yapmak için herkes kolaylıkla kullanabilir. Ücretsiz olan Kodu Game Lab yazılımının Türkçe dil desteğinin de bulunması bir diğer olumlu yanındır (Kodugamelab, 2018). Bireylerin programlama konusunda basit seviyede eğitimini amaçlayan kodu game lab yayınlandığı günden bu yana birçok eğitim kurumunun odağı olmuş ve yayınlandığı ilk yılda Amerika'da 60' a yakın kurumda eğitim aracı olarak kullanılmıştır (Programlar, 2010).

2.3.1.5. Tynker

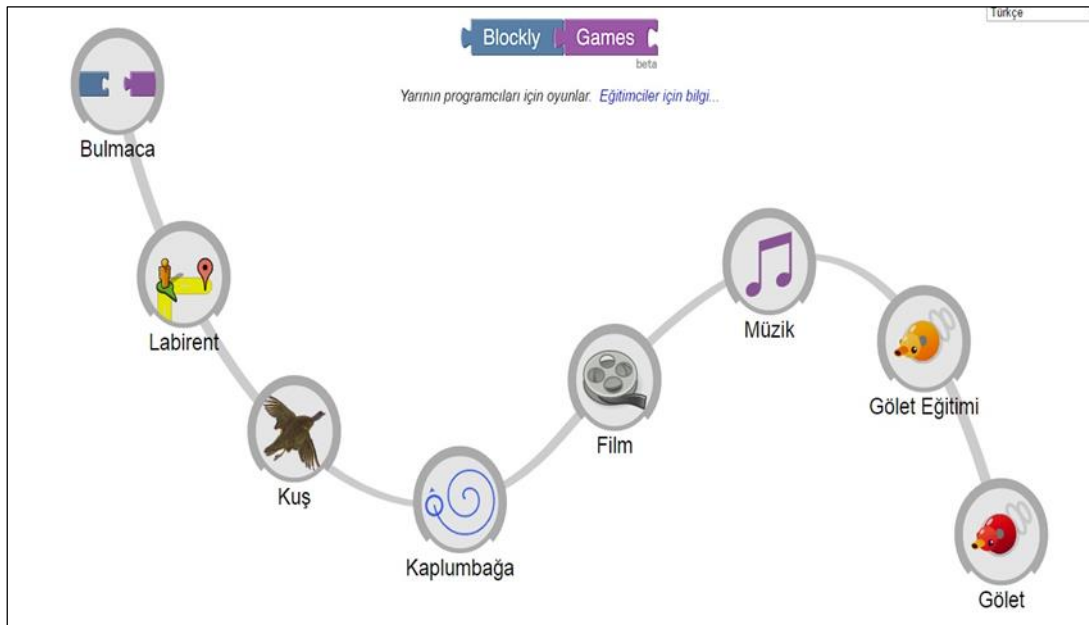
Tynker, HTML 5 ve Javascript tabanlı herhangi bir eklentiye ihtiyaç duyulmadan tablet ve akıllı telefonlarda da kullanılabilen, 4-14 yaş grubundaki çocuklar için kendi kendine öğrenme temasında çevrimiçi bir programlama eğitim ortamıdır. Ayrıca Tynker, kendi kendine öğrenme sağlayan çevrimiçi kursların yanı sıra okullar ve öğretmenler için de ilgi çekici bir programlama müfredatı sunmaktadır (Wikipedia, 2019). Yayımlanan kurslar, uzay temalı oyunlar şeklinde tasarlanmıştır. Karakterlerin uzay gemileri ve roketlerden oluşması çocukların ilgisini çekmekte, çocuklar yaşları ile eşleşen seviyelere göre ilerleme yapabilmektedirler (CodeWizardhq, 2016). Kaynak kodu yazmak yerine, kod bloklarının sürüklenip bırakılmasıyla daha görsel bir ortamda yazılımın oluşturulması bir takım yazım hatalarının da önüne geçmektedir. Tynker kısmen ücretsiz olan bir platformdur.

2.3.2. Blockly

Blockly games, Google tarafında geliştirilmiş erken yaştaki çocuklara programlama öğretmeyi amaçlayan bir dizi eğitsel oyundan oluşmaktadır. Blockly ile daha öncesinde programlama konusunda deneyimi olmayan öğrencilere programlamayı sevdirmek ve verilen görevlerin tamamlanması ile kodlama mantığını geliştirmek amaçlanmıştır. Aynı zamanda ücretsiz ve açık kaynak kodlu olması sebebiyle bağımsız kodlama uygulamaları da geliştirilebilir. Her kullanıcı için kendi hızında bir öğrenme sunan Blockly Games, web tabanlıdır ve aynı zamanda çevrimdışı kullanılabilmesi için indirilebiliyor olması da tüm öğrenciler için erişilebilirlik sağlamaktadır (Blockly, 2020). 8 yaş ve üzeri için tasarlanan Blockly programlamayı JavaScript kullanarak blok temelli olarak anlatmaktadır. Blockly ortamı öğrencilere sağladığı sade arayüzü ve kod yazmayı gerektirmeyen yapısı ile öğrencilerin olumsuz tutumlarını ortadan kaldırıp, derse karşı motivasyonlarını da artırabilir.

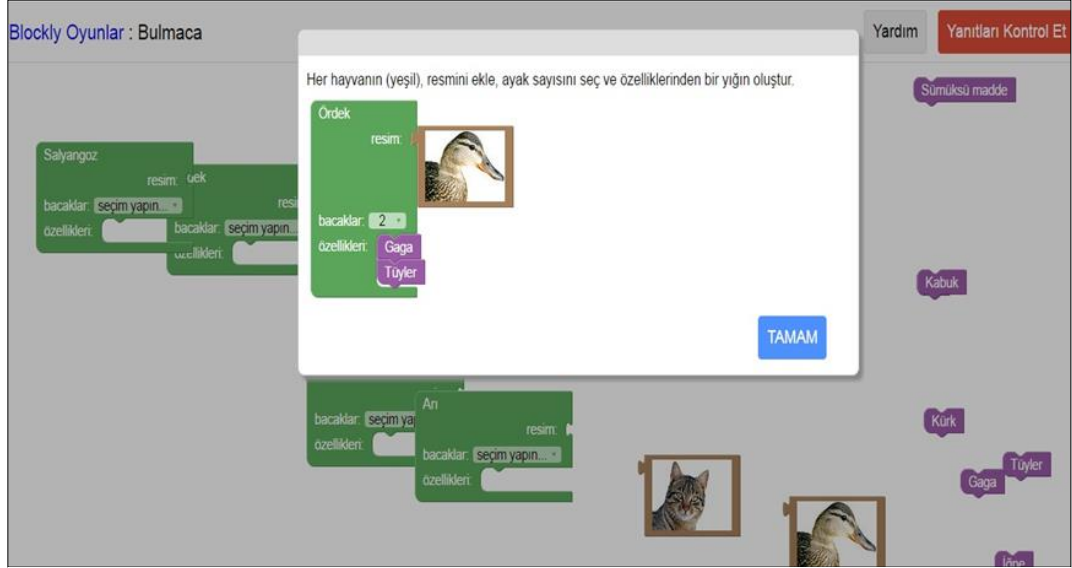
Öğrenme ortamında çeşitli bulmacalar ve bu bulmacaları tamamlayabilmek için kod blokları bulunmaktadır (CodeWizardhq, 2016). Öğrenciler kod bloklarını sürüklenip bırak yöntemi ile komut ekranına tıpkı bir yap-boz gibi yerleştirilerek ekrandaki karakter ile görevleri yerine getirirler. Bu bulmacaları tamamlayan kullanıcılar metin tabanlı programlama dillerini kullanmaya hazır olarak görülmektedir. Bulmacalar yedi kategoriye ayrılmış ve her biri programlama ile ilgili farklı bir kazanımı öğretmek için tasarlanmıştır. Kategoriler şu şekildedir (Blockly, 2020) (Şekil- 1):

- **Puzzle**, Blockly' nin şekillerine ve parçaların nasıl birleştiğine hızlı bir giriş niteliğinde olan bulmacadır.
- **Labirent**, döngülere ve koşullara başlangıç seviyesinde bir giriştir. Seviyeler basit başlar kademeli olarak zorlaşarak ilerler.
- **Kuş**, giderek karmaşıklaşan koşullu ifadeleri kavranmasını amaçlayan bulmacadır.
- **Kaplumbağa** kategorisinde döngülere giriş yapılarak çeşitli örnekler ile iç içe döngüler kavratılır.
- **Film**, matematiksel denklemlere ve fonksiyonlara giriş niteliğindedir.
- **Pond Tutor**, ilk iki görevden sonra javascript' e geçiş yaparak metin tabanlı programlama sunar.
- **Gölet** kategorisi programlama ile ilgili döngüler, koşullu yapılar, fonksiyonlar vb. tüm kavramların bir arada kullanılarak bir oyun tasarlandığı kategoridir. Bu kategoride javascript ile metin tabanlı kodlama ve blokları kullanarak blok tabanlı kodlama olmak üzere iki seçenek mevcuttur.



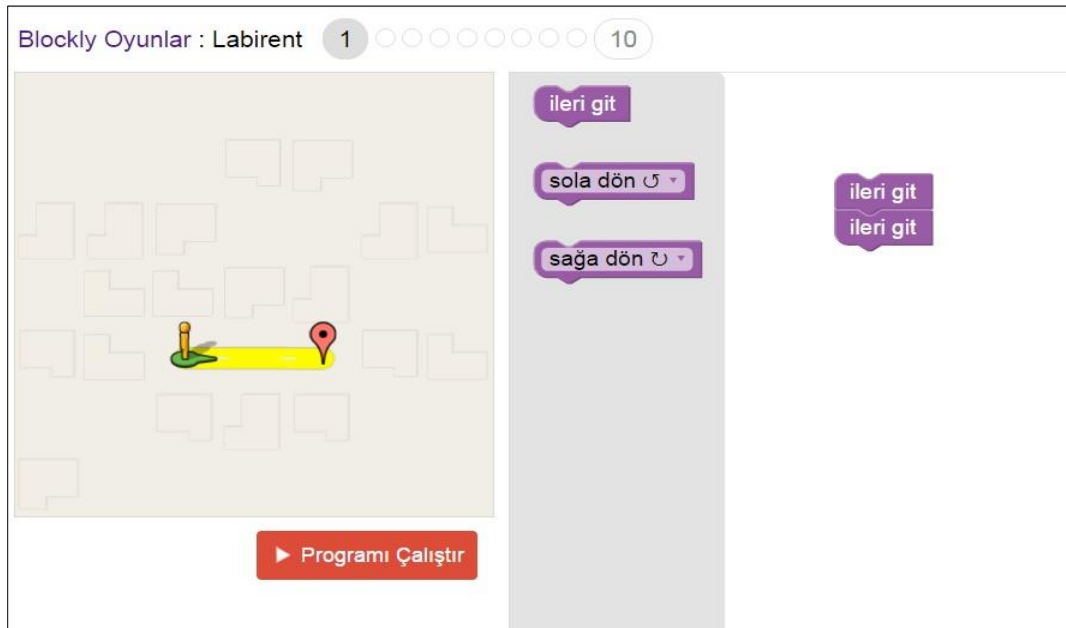
Şekil- 1 Blockly Games Ekran Görüntüsü

Şekil-2’ de bulmaca kategorisine ait bir ekran görüntüsü verilmiştir. Bu kategoride öğrenciler basit düzeydeki bulmacaları sürükle bırak yöntemi ile çözerek Blockly kodlama ortamına giriş yapmaktadırlar. Ayrıca bu kategoride algoritmalarındaki sabit ve değişken kavramı da öğrencilere kazandırılmaktadır.



Şekil- 2 Bulmaca Kategorisi Ekran Görüntüsü

Şekil 3 ve şekil 4’ de labirent kategorisinden bir ekran görüntüsü verilmiştir. Bu kategoride toplam 10 bulmaca bulunmaktadır. Bulmacalar doğrusal mantık yapısı, karar yapıları ve döngüleri içerecek şekilde ilerlemektedir.



Şekil- 3 Labirent Kategorisi Ekran Görüntüsü

Şekil 4’de görüldüğü gibi bloklar doğru bir şekilde yerleştirilip seviye tamamlandığında, öğrenciler seviyenin JavaScript komutlarını görebilmektedirler.



Şekil- 4 Bloklara Ait Javascript Komut Ekranı

Şekil 5’ de kuş kategorisinden bir ekran görüntüsü verilmiştir. Kuş kategorisinde giderek karmaşıklaşan toplam 10 bulmaca bulunmaktadır.



Şekil- 5 Kuş Kategorisine Ait Ekran Görüntüsü

Bu arařtırmada da blok tabanlı kodlama ortamı olarak Blockly' de yer alan bulmaca, labirent ve kuř kategorileri kullanılacaktır. Blockly seilmesinin sebebi ise literatürde üzerine yapılmıř çok fazla arařtırma bulunmuyor olmasıdır.

2.4.Biliřim Teknolojileri ve Yazılım Dersi

Programlama eđitimi gemiř yıllardan bu yana pek çok ülkenin eđitim gündeminde olmuř ve ilköđretimden liseye eđitim programlarında da bu yönde deđiřikliklere gidilmiřtir. Ülkemizde ise bu eđitim meslek liselerindeki belli bölümlerde ve lisans eđitimindeki alakalı bilim dallarında verilmekte idi (Erdem, 2018). Son yıllarda ülkelerin eđitim müfredatlarını erken yařlardan itibaren kodlama eđitimi vermeye yönelik güncellemesi ile birlikte ülkemizde de bu konuyla ilgili kazanımlar Biliřim Teknolojileri ve Yazılım (BTY) dersinin ierisine yerleřtirilmiřtir.

Semeli Bilgisayar dersi temel bilgisayar okuryazarlıđının kazandırılması amacıyla Türkiye'de ilk kez 1998 yılında ilköđretim okullarının müfredatına eklenmiř ve öđretim programının ieriđinde deđiřiklikler yapılırsa da 2012 yılına kadar zorunlu olmayan eđitim sürecine dahil edilmiřtir (Bülbül řoltan, 2018). 2012 yılında Talim-Terbiye Kurulu Bařkanlıđı, Yenilik ve Eđitim Teknolojileri Genel Müdürlüđünün 03.09.2012 tarihinde 14975 sayılı yazısı ile dersin ismi Biliřim Teknolojileri ve Yazılım dersi olarak güncellenmiř ve görüřülen yeni müfredatın, 5.6.7. ve 8. sınıflarda semeli olarak okutulmasına karar verilmiřtir (Uzgun ve Ayka, 2016). Ülkemizde 2012 yılı itibariyle deđiřen eđitim sistemiyle birlikte Biliřim Teknolojileri ve Yazılım dersi kazanımları ierisinde yer alan "problem çözmeye, programlama ve özgün ürün geliřtirme" kazanımı için çeřitli blok tabanlı araçlar ile çocukların eđlenerek uygulama geliřtirebilecekleri araçlar da eđitime dahil edilmeye bařlanmıřtır (Berksoy, Sözcü, Armađan ve Arslan, 2016). 2012 yılında yayınlanan öđretim programında uluslararası standartlar incelenerek belirlenen yeterlilikler biliřim okuryazarlıđı, biliřim teknolojilerini kullanarak iletiřim kurma, bilgi paylařma ve kendini ifade etme, arařtırma yapma, bilgiyi yapılandırma ve iřbirlikli çalıřma, problem çözmeye, programlama ve özgün ürün geliřtirme olarak kategorize edilmiřtir ancak bu program öncelikle kapsamlı bir kodlama ve algoritma öđretimine yönelik hazırlanmamıřtır (Uzgun ve Ayka, 2016). Temel Eđitim Genel Müdürlüđünün 08.05.2013 tarihli ve 886487 sayılı teklif yazısı üzerine ilköđretim kurumları haftalık ders çizelgesi ortaokul kısmında 5. ve 6. sınıflarda semeli ders statüsünden zorunlu ders statüsüne alınan BTY

dersi 7. ve 8. sınıflarda ise yine haftada 2 saat olmak üzere seçmeli dersler kapsamında olduğu görülmektedir (TTKB, 2013). Dersin öğretim programında en güncel hali ile Talim ve Terbiye Kurulu Başkanlığı'nın 24.01.2018 tarihinde yayımlanmış olduğu kurul kararına göre, 5. ve 6. sınıf düzeyi için belirlenmiş çeşitli alt konu ve kazanımlarıyla birlikte beş temel ünite (Bilişim Teknolojileri, Etik ve Güvenlik, İletişim, Araştırma ve İş Birliği, Ürün Oluşturma, Problem Çözme ve Algoritma) yer almaktadır (TTKB, 2018). Öğretim programındaki 5.ünite olan Problem Çözme ve Algoritma ünitesi ile öğrencilerin problem çözme ve bununla birlikte bilgi işlemsel düşünme becerilerini kazanmaları hedeflenmiştir. Bu ünitenin alt amaçları incelendiğinde; öğrencilerde temel algoritma mantığını geliştirerek en az bir programlama dili hakkında belli düzeyde beceriye sahip olmaları ve günlük hayatta karşılaşılan sorunlara çözüm getirmeye yönelik özgün ürünler oluşturabilmeleri gibi kazanımlar olduğu görülmektedir.

2.5.Bilgi İşlemsel Düşünme

21.yüzyıl becerilerinin alt boyutlarından ve bir çeşit analitik düşünme becerisi olan bilgi işlemsel düşünme ulusal literatüre farklı Türkçe karşılıklar ile geçmiştir. Bunların bir kısmı; “Bilgisayarca Düşünme” (Korkmaz, Çakır ve Özden, 2015), “Komputasyonel Düşünme” (Şahiner ve Kert, 2016), “Hesaplamalı Düşünme” (Özçınar, Yecan ve Tanyeri, 2016), “Bilişimsel Düşünme” (Sayın ve Seferoğlu, 2016) “Bilgisayımsal Düşünme” (Çınar ve Tüzün, 2017) şeklindedir. Bilgi işlemsel düşünme, Wing (2006) tarafından bir problem çözme yaklaşımı olarak tanımlanmanın yanında, bilgisayar bilimi kavramları ile birtakım sistemler tasarlayarak insan davranışlarını anlayabilmek aynı zamanda problem çözme süreçleri için gerekli bir beceri olarak ifade edilmiştir. Özellikle son yıllarda oldukça gündemde olan bilgi işlemsel düşünme kavramı farklı zamanlarda farklı araştırmacılar tarafından tanımlanmıştır. Denning (2009), bilgi işlemsel düşünmenin geçmişinin 1950’li yıllara kadar uzandığını, o yıllarda girdileri çıktıya dönüştürerek problemlerin analizinde kullanılan algoritmik düşünce şeklinde bilindiğini belirtmektedir. Ayrıca Denning (2003), bilgi işlemsel düşünmeye farklı açılardan değinerek, bunları 7 kategori altında toplamıştır. Bu kategorileri de şu şekilde açıklamıştır:

1) *Hesaplama*: Bilgi girişi, algoritmanın oluşturulması ve uygulanmasını içine alan süreç.

- 2) *İletişim*: Herhangi bir nesneden bir diğerine bilgi aktarılması süreci.
- 3) *Koordinasyon*: Amaca ulaşılabilmesi için gerçekleştirilecek işlemlerin zamanlamasının yapılması
- 4) *Anımsama*: İşlemlerin etkin bir şekilde gerçekleştirilebilmesi için bilginin düzenlenmesi ve kodlanması
- 5) *Otomasyon*: Hesaplamanın fiziksel sistemlerle birleştirilmesi
- 6) *Değerlendirme*: Verilerin istatistiksel ve deneysel analizinin yapılması
- 7) *Tasarım*: Sürecin, nesnelere ve sistemlerin organizasyonu, tasarlanma süreci.

Futschek (2006) bilgi işlemsel düşünmenin, problemi analiz etme, anlama ve ifade etme, çözüm önerileri geliştirme ve çözümlere uygun algoritma oluşturabilme gibi alt becerilerden oluşan bir kavram olduğunu ifade etmiştir. Wing'e (2006) göre bilgi işlemsel düşünme becerisi, bilgisayarların değil, bireylerin problem çözümünde kullandıkları kendileri özgü bir düşünme becerisidir. Hu (2011) bilgi işlemsel düşünmeyi; bilgisayarların çalışma prensibine benzeterek, bir problemi bilgi işleme araçları ile çözmek, çözümü farklı problemlere aktarabilmek ve analiz sentez metotlarıyla yeni ürünler geliştirebilmek olarak tanımlamıştır. Voskoglou ve Buckley (2012) ise bilgi işlemsel düşünmenin bilgisayar bilimlerinde eleştirel düşünceyle birlikte uygulanan ve problemleri sonuca ulaştırmada kullanılan yeni bir yöntem olduğunu vurgulamıştır. Aho (2012)' a göre bilgi işlemsel düşünme, problemleri formüle ederken problem çözme süreçlerini de algoritmik olarak ifade etmektir. Shailaja ve Sridaran (2015) göre ise matematik, mühendislik gibi bilimlerde kullanılan düşünme süreçlerinin birleşimidir. Mannila ve diğerleri (2014) bilgi işlemsel düşünmeyi, problemlere getirilen çözümleri formülleştirmeye yarayan bilgisayar bilimlerinden birtakım kavramlardan ve düşünce sürecinden oluşan bir terim olarak tanımlamıştır. Patan (2016), ise çözümünde teknoloji kullanılabilecek bir problemi tanımlayıp, çözümü algoritmik düşünce yoluyla belirleyerek, en verimli çözümü seçip, benzer problem durumlarında da bu çözümü transfer edebilmeyi içeren bir beceri şeklinde tanımlamıştır.

ISTE ve CSTA (2011) bilgi işlemsel düşünceyi, bilgisayar ve benzeri teknolojik cihazlar ile problemleri formüle ederek, verilerin analiz edilerek düzenlenmesi, modellenmesi ve bu çözüm sürecinin farklı problemlere de transfer edilebilmesi olarak tanımlamıştır. ISTE (2016), 21.yüzyıl öğrenen standartlarını; yenilikçi ve bilgiyi oluşturan, dijital vatandaş, yaratıcı ve bilgi işlemsel düşünebilen bireyler olarak

belirlemiştir. ISTE (2016) standartları bilgi işlemsel düşünme becerileri ile birlikte öne çıkan becerileri şöyle tanımlanmıştır:

- Öğrenenler; problem tanımlarını, soyut modeller ve veri analizleri ile çözüm bulurken algoritmik düşünce gibi teknoloji destekli yöntemler ile formüle eder.
- Öğrenenler, problem çözme süreçlerinde karar vermeyi kolaylaştırmak için çeşitli veri analiz yollarını kullanırlar.
- Öğrenenler, problemleri parçalarına ayırarak, önemli verileri seçer, problem çözme süreçlerini basitleştirmek, sistemleri çözebilmek için modeller geliştirirler.
- Öğrenenler sistemlerin nasıl işlediğini anlar, çözümler üretmek ve bu çözümleri denemek amacıyla adımlar geliştirmek için algoritmik düşünceyi kullanırlar.

İlk bakışta karmaşık görünen bir problemi daha anlaşılır ve yönetilir şekilde küçük parçalara bölmeyi (ayırıştırma) içerir. Problemin daha küçük parçalarından her biri, benzer problemlerin nasıl çözüldüğü dikkate alınarak (benzerlik arama) önemsiz bilgileri göz ardı edip (soyutlama), önemli ayrıntılara odaklanılabilir. Problem parçalarının her biri için basit kurallar ve adımlar (algoritma) tasarlanabilir (BBC, 2018).

Bunların dışında Kong ve Abelson (2019), bilgi işlemsel düşünmeyi aşağıda belirtilen özellikleri de kapsayan bir problem çözme süreci olarak tanımlamıştır:

- 1) Bilgisayar ve dijital araçları problem çözme süreçlerinde kullanmamıza olanak sağlayacak şekilde problem oluşturabilme.
- 2) Bir problemin verilerini mantıksal sıralama dahilinde organize ederek analizini yapabilme.
- 3) Verileri, modeller ya da simülasyonlar ile soyutlayabilme.
- 4) Algoritmik düşünmeyi kullanarak çözümleri otomatikleştirme.
- 5) En etkili adım ve kaynağa ulaşma hedefi ile olası çözümleri belirleyerek, analiz etme ve uygulayabilme.
- 6) Uyguladığı problem çözme adımlarını ve süreci genelleyebilme ve farklı problemlere aktarabilme.

Alanyazındaki bu tanımlamalardan yola çıkarak bilgi işlemsel düşünmenin bilişimsel fikirler üreterek teknoloji destekli bir çözüm süreci olduğu söylenebilir (Ramazanoğlu, 2021). Tanımlamaların yanı sıra bir tür analitik düşünce becerisi olan

BİD, öğretim müfredatları ile bütünleştirildiğinde öğrencilerde matematiksel düşünme becerilerini de geliştirebileceği görülmüştür (Oluk, 2017). Uluslararası bir etkinlik olan “Bilge Kunduz-Uluslararası Enformatik ve Bilgi İşlemsel Düşünme Etkinliği” bütün öğrencilere bilgisayar bilimleri konusundaki farkındalığını artırırken aynı zamanda eğlendirerek BİD becerisini kazandırmayı hedeflemektedir (Bilge Kunduz, 2019). Bilgi işlemsel düşüncenin bireye günlük yaşam becerilerinde ve aynı zamanda akademik anlamda kazandırdıkları düşünüldüğünde bu alanda geliştirilen stratejiler ve yapılan araştırmalar da oldukça önemli hale gelmiştir.

2.6.Bilgi İşlemsel Düşünme Becerisi ve Kodlama İlişkisi

Genel bir tanımlama ile bilgi işlemsel düşünme becerisi, bireylerin karşılaştıkları problemlerin çözüm süreçlerinde bilgi iletişim teknolojileri araçlarını da kullanarak oluşturdukları bilişsel stratejiler olarak nitelendirilebilir. Bilgi işlemsel düşünme kavramı yalnızca bilgisayar bilimi ile ilgilenenlerin değil, tüm bireylerin benimsemesi gereken bir düşünme becerisidir. Kodlama eğitimi, bu kavramı benimsetmek için kullanılacak yollardan birisidir. Dünya genelinde birçok eğitimci ve eğitim kuruluşu, bilgi işlemsel düşünmenin de temel unsurlarından olan kodlamanın erken çocukluk döneminde öğretilmesini savunmaktadır. Kodlama becerisinin temelini oluşturan algoritmalar, birçok kaynakta bilgi işlemsel düşünmenin bir boyutu olarak yer almaktadır. Bu durum kodlama eğitimi ve bilgi işlemsel düşünme becerisinin aynı dönemlerde yaygınlaşmasının gerekçesi olarak da gösterilebilir (Üzümcü ve Bay, 2018). Kodlama, öğrenenlerin karmaşık durumları ve verileri daha kolay anlamlandırabilmelerini sağlarken aynı zamanda bilgi işlemsel düşünme becerilerinin gelişmesine yardımcı olmaktadır (Resnick ve diğerleri, 2009). Kodlama ve bilgi işlemsel düşünme becerisi öğrenme ortamlarında birbirlerini destekler nitelikteki becerilerdir. Çünkü bilgi işlemsel düşünme problemlerin detaylı analizini yaparken kodlama becerisinin de geliştirilmesine yardımcı olmakta ve problemlerin daha derinlemesine anlaşılıp alternatif çözümler üretilmesine katkı sağlamaktadır (Sanz, 2015).

Bilgi işlemsel düşünmenin de içinde bulunduğu 21. yüzyıl becerilerinin öğrencilere kazandırılması ile alakalı farklı yaklaşımlar üzerine araştırmalar yapılmıştır. Dijital oyunlar, robotik kodlama etkinlikleri ve elektronik logolar bilgi işlemsel düşünmenin geliştirilmesi için başvurulan yaklaşımlardan bazılarıdır (Çınar ve Tüzün,

2017). Kodlama herhangi bir problemin çözümü için oluşturulan işlem basamaklarının kodlara çevrilerek bilgisayar ortamında işlenmesi ve derlenmesi olarak tanımlanabilir (Arabacıoğlu, Bülbül ve Filiz, 2007). Tanıma göre aslında bir problem çözme süreci olarak nitelendirilen kodlama gibi benzeri tasarım temelli öğrenme faaliyetleri bilgi işlemsel düşünme becerisinin geliştirilmesine katkı sağlayabilmektedir (Alsancak-Sırakaya, 2019). Kodlama eğitimi ve bilgi işlemsel düşünme ilişkisi üzerine de yapılmış pek çok çalışma mevcuttur. Yapılan birçok araştırma bilgi işlemsel düşünme becerisinin bireylere kazandırılması sürecinde kodlama eğitiminin oldukça önemli bir yeri olduğunu vurgular niteliktedir. Weinberg (2013) yaptığı çalışmada bilgi işlemsel düşünmenin bireylere kazandırılmasında dört ayrı method kullanılabileceğini ifade etmiştir. Bunları disiplinler arası uygulamalar, bilgisayarsız kodlama ortamları, BTP ortamları ve robot programlama uygulamaları olarak sınıflandırmıştır. Lye ve Koh (2014) tarafından yapılmış olan çalışmada da bilgi işlemsel düşünme becerisinin kazandırılmasında kodlama öğretiminin en etkili yol olduğu ifade edilmiştir. Fessakis, Gouli ve Mavroudi (2013) erken yaşta kodlama öğrenmenin, üst düzey düşünme becerilerinden olan algoritmik düşünme, problem çözme ve bilgi işlemsel düşünme becerilerinin gelişmesine önemli katkı sağladığını belirtmiştir. Benzer bir ifadeyle Kalelioğlu (2015), erken yaşta kodlama öğretimi ile eleştirel, yaratıcı ve algoritmik düşünme gibi üst düzey düşünme becerilerine katkısının olduğunu ifade etmiştir. Lockwood ve Mooney (2017) araştırmalarında bütün öğrencilere bilgi işlemsel düşünme becerisinin kazandırılması gerekliliğinden bahsetmiştir. Carnegie Üniversitesinde yapılan bir araştırmaya göre hesaplamalı düşünme (BİD), problemleri çözerek sistem tasarlamak ve insan davranışlarını anlamak da dahil bu beceriler ile birlikte bilgisayar programlamadan çok daha fazlası olarak nitelendirilmiştir. Yine bu araştırmaya göre öğrencilerde bilgi işlemsel düşünmeyi geliştirmek için bilgisayar programlama eğitiminin kullanılabileceği belirtilmiştir (Calao, Moreno-Leon, Correa & Robles, 2015). Oluk vd. (2018) tarafından yapılan çalışmada Scratch' in algoritma geliştirme ve bilgi işlemsel düşünme becerilerine etkisini incelemek amaçlı deney ve kontrol grubu oluşturulmuştur. Gruplardan birinde görsel kodlama ortamları kullanılarak eğitim verilirken diğerinde mevcut eğitim müfredatı ile çalışma yürütülmüştür. Araştırmanın sonucunda görsel kodlama ortamları ile yapılan derste, diğer gruba göre öğrencilerin algoritma geliştirme ve bilgi işlemsel düşünme becerisinin anlamlı derecede farklılaştığı görülmüştür. Ramazanoğlu (2021)' da blok tabanlı kodlama üzerine yapmış olduğu çalışmada kodlamanın öğrencilerde; yaratıcı ve

algoritmik düşünme, problem çözme, eleştirel ve bilgi işlemsel düşünme, derse katılım ve mesleki alan becerilerinin geliştirilmesinde olumlu katkılarının olduğunu ifade etmiştir

Crow (2014)' nin makalesinde belirttiği üzere oldukça hızlı gelişen dünyamızda karşılaşılan problemlere nasıl çözümler getirileceği, yaratıcılığın nasıl kullanılacağı küçük yaşlardan itibaren kazandırılması gereklidir. Bu da gelecek nesilleri bilgi işlemsel düşünmeye teşvik etmekle mümkün olabilir. Yapılan araştırmalar doğrultusunda görülmektedir ki bilgi işlemsel düşünme üzerinde kodlama öğretimin etkisi oldukça büyüktür.

2.7.Bilgi İşlemsel Düşünme Becerisi Üzerine Yapılmış Araştırmalar

Lee vd. (2011), “Computational Thinking for Youth in Practice” adlı araştırmasına, bilgi işlemsel düşünmenin pratikte nasıl olduğu ve öğrencilerde geliştirmek için neler yapılabileceğini anlamayı amaçlamıştır. STEM (Bilim, Teknoloji, Mühendislik ve Matematik) ve Bilişim eğitimcileri işbirliği ile yürütülen bu çalışmada kullanılan örnekler ITEST (Öğrenciler ve Öğretmenler İçin Yenilikçi Teknoloji Deneyimleri) tarafından toplanmıştır. Bilgi işlemsel düşünme becerisinin geliştirilebilmesi için “Kullan-Değiştir-Yarat” olarak isimlendirilen bir yapı önerilmiştir. Bir bilgisayar oyunu veya bir programın kullanılmasıyla ortaya çıkan yapı, programın arayüzünün ve diğer bileşenlerinin değiştirilmesiyle ilerlemektedir. Son aşama olarak öğrenciler, kendi uygulamalarını üretip programın içyapısını öğrenmiş olurlar.

Barcelos ve Silveira (2012), “Teaching Computational Thinking in Initial Series” adlı araştırmalarında Matematik ve bilgi işlemsel düşünme arasındaki ilişkiyi incelemeyi amaçlamışlardır. Bu amacın belirlenmesinde etkin olan hipotez, Latin Amerika’da Matematik bilgisinin zayıf olmasının, üniversitede Bilgisayar bilimine duyulan ilginin az olmasında ve Bilgisayar derslerini bırakma oranlarının fazla olmasında etkili olabileceğidir. Temel eğitim noktasında bilgi işlemsel düşünme becerilerinin işe koşulmasının matematiksel becerilerin gelişmesini sağlayabileceği ve böylece öğrencilerin meslek seçimlerinde bilgisayar bilimlerine yönelimlerinin artabileceği belirtilmiştir.

Czerkowski ve Lyman (2015), “Exploring Issues About Computational Thinking in Higher Education” adlı araştırmasında lisans düzeyinde, bilgi işlemsel düşünme becerilerinin durumunu incelemeyi hedeflemiştir. Bu doğrultuda Bilgisayar bilimleri ve diğer bilimlerde bilgi işlemsel düşünmenin süreçlere dahil edilmesiyle ilgili yayınlara ulaşılmış ve bu yayınlar incelenmiştir. Araştırmanın sonucunda, Bilgisayar bilimiyle birlikte farklı disiplinlerin de iş birliğiyle ortak modeller oluşturarak bilgi işlemsel düşünme kavramının yükseköğretimde uygulanması önerilmiştir.

Shailaja ve Sridaran (2015), “Computational Thinking the Intellectual Thinking for the 21st Century” adıyla yürütmüş oldukları araştırmalarında, K12 düzeyindeki öğrencilerde bilgi işlemsel düşünme becerisinin nasıl kazandırılacağı konusu üzerinde durmuşlardır. Araştırmada, K2 seviyesinde tanıma, görselleştirme ve genelleme, K3-K5 seviyesinde algoritmik düşünme, ayrıştırabilme ve değerlendirme yapma, K6-K8 seviyesinde soyutlama ve eleştirel düşünme becerilerinin kazandırılması birtakım örnekler sunularak açıklanmıştır.

Howland ve Good (2015) 12-13 yaş grubundaki ortaokul öğrencileri ile yapmış oldukları çalışmada oyun tasarım süreçlerinin bilgi işlemsel becerilerine etkisini incelemeyi amaçlamışlardır. Araştırmada öğrencilere bilgi işlemsel düşünce kavramlarını ölçmeyi hedefleyen ön test ve son testler uygulanmıştır. Ardından geliştirilen uygulamalar incelenmiştir. Araştırma sonucunda elde edilen bulgulara göre oyun tasarım süreçleri öğrencilerin bilgi işlemsel düşünme ve programlama becerilerini geliştirmektedir.

Korkmaz, Çakır ve Özden (2015), “Bilgisayarca Düşünme Beceri Düzeyleri Ölçeğini” ortaokul kademesine uyarlamak için bir çalışma yürütmüşlerdir. Bu çalışmada “Computational Thinking” kavramı “bilgisayarca düşünme” şeklinde Türkçe’ye çevrilerek kullanılmıştır. Ortaokul öğrencilerinde bilgisayarca düşünme becerilerinin ölçülmesinin hedeflendiği çalışmada, önceden üniversite öğrencilerine uygulanmış ölçek ortaokul düzeyine uyarlanarak uygulanmıştır. Çalışma grubunu 7. ve 8. sınıf kademesinden öğrenciler oluşturmuştur. Araştırma sonucunda elde edilen bulgulara göre ortalamanın en yüksek bulunduğu bilgisayarca düşünme becerisi işbirliklilik iken en düşük olanı ise problem çözme becerisi olmuştur.

Şahiner ve Kert 2016 yılında “Komputasyonel Düşünme Kavramı ile ilgili 2006-2015 Yılları Arasındaki Çalışmaların İncelenmesi” adıyla bir çalışma yürütmüşlerdir. Bu çalışmada 2006 ile 2015 yılları arasında Komputasyonel düşünme ile ilgili yapılmış araştırmaları içerik analizi yöntemi ile incelemek hedeflenmiştir. Örneklemini İngilizce yazılmış 22 makalenin oluşturduğu araştırmanın bulgularına göre çalışmalar genelde ortaokul öğrencileri ile yapılmış ve birçoğunda nitel araştırma yöntemi kullanılmış olduğu görülmüştür. Ayrıca Komputasyonel düşünme ile alakalı yazılmış makaleler 2012 yılından sonra önemli bir artış göstermiştir. Araştırmanın sonunda, Komputasyonel düşünme kavramıyla alakalı yeni çalışmaların yapılması ve eğitim-öğretim süreçlerine entegre edilmesi önerilmiştir.

Yapılan bir diğer araştırma Yolcu (2016)’ nun, “Programlama Eğitiminde Robotik Kullanımının Akademik Başarı, Bilgi İşlemsel Düşünme Becerisi ve Öğrenme Transferine Etkisi” adlı çalışmasıdır. Karma yöntem ile yürütülen araştırmada çalışma grubunu 6.sınıfa devam eden 47 öğrenci oluşturmuştur. Araştırmanın nicel boyutunda deneysel desen kullanılırken, nitel veriler öğrencilerin robotik etkinlikler ile alakalı görüşleri ile belirlenmiştir. Araştırmada deney grubunda öğretim boyunca robotik etkinliklerden faydalanılmıştır. Kontrol grubunda kullanılan öğretim yönteminde herhangi bir değişiklik yapılmamış olup halihazırdaki yöntem kullanılmıştır. Araştırma sonucunda deney grubundaki başarı ve öğrenme transferi puan ortalamasının, kontrol grubundaki başarı ve öğrenme transferi puan ortalamasından anlamlı düzeyde yüksek olduğu görülmüştür. Ayrıca öğrenciler robotik setlerle ve etkinliklerle alakalı olumlu geri dönüşlerde bulunmuşlardır.

Korucu, Gençtürk ve Gündoğdu (2017), “Examination of the Computational Thinking Skills of Students” adıyla yaptıkları araştırmalarında ortaokul kademesindeki öğrencilerde bilgi işlemsel düşünme becerilerini birtakım değişkenlere göre incelemeyi amaçlamışlardır. Nicel yöntemin kullanıldığı araştırmada, veriler Korkmaz, Çakır ve Özden (2015) tarafından uyarlanan “Bilgisayarca Düşünme Beceri Düzeyleri Ölçeği” ile toplanmıştır. Araştırma bulgularına göre öğrencilerde bilgi işlemsel düşünme becerisi, mobil teknolojilere sahiplik ve sınıf düzeyine göre anlamlı bir şekilde farklılaşırken; haftalık internet kullanımı, cinsiyet ve mobil teknolojileri kullanım yeterliği değişkenleri açısından farklılaşmanın olmadığı görülmüştür.

Rose, Habgood ve Jay (2017) yaptıkları çalışmada programlama yaklaşımlarının bilgi işlemsel düşünme becerileri üzerindeki etkisini incelemişlerdir. Araştırma, öğrencilerin ScratchJr ve Lightbot programlama araçları ile programlamaya olan yaklaşımlarını ve aynı zamanda bu yaklaşımlarının bilgi işlemsel düşünme becerilerinin gelişimine etkisini incelemeyi amaçlamıştır. Araştırmanın örneklemini Kuzey İngiltere’de ilköğretim kademesinde öğrenimine devam eden, 6 ve 7 yaşlarında 20 erkek ve 20 kız öğrenci oluşturmuştur. Araştırma sonunda, öğrencilere uygulanan muhakeme testine göre uygulamanın her iki versiyonunda da performansların benzer olduğu görülmüştür. İleride yapılacak araştırmalar için programlama araçlarının algoritmik düşünme, ayırıştırma ve soyutlama gibi becerileri geliştirmedeki katkısının yanında bilgi işlemsel düşünme üzerine katkısına da odaklanılması önerilmiştir.

Taş (2018), araştırmasında üstün yetenekli öğrencilerde, bilgisayar destekli matematik öğretiminin öğrencilerin Matematik dersine ilişkin tutumuna ve bilgi işlemsel düşünme becerilerine etkisini incelemiştir. Karma yöntemin kullanıldığı çalışmanın örneklemini Bilim ve Sanat Merkezindeki 7.sınıf kademesi oluşturmuştur. Nitel boyutta yarı yapılandırılmış görüşme ile katılımcılara sorular sorulmuş, nicel boyutta ise deneysel desen kullanılmıştır. Kontrol grubu Matematik dersini geleneksel yöntemler ile alırken deney grubu bilgisayar destekli eğitim ile almıştır. Çalışmanın sonucunda deney grubundaki öğrencilerin, bilgi işlemsel düşünmenin yaratıcılık, algoritmik düşünme ve İşbirlikli çalışma alt boyutlarında kontrol grubuna göre son test puanlarında anlamlı derecede farklılaşma olduğu görülmüştür. Bununla birlikte çalışmanın nitel bulguları incelendiğinde deney grubu öğrencileri aldıkları eğitimle ilgili olumlu görüşler bildirmişlerdir.

Şimşek (2018), araştırmasında programlama eğitiminde Scratch ve robotik uygulamalarının bilgi işlemsel düşünme becerisi ve akademik başarı üzerine etkisini incelemeyi amaçlamıştır. 5. Sınıf öğrencileri ile yürütülen araştırmada nicel yöntem kullanılmıştır. Araştırma sonunda öğrencilerin akademik başarı ve bilgi işlemsel düşünme becerileri karşılaştırıldığında her iki öğretim yöntemi için de puanlarda anlamlı bir farklılaşma olmadığı görülmüştür.

Berikan (2018), “Bilgi İşlemsel Düşünme Becerisine Yönelik Tasarlanan “Veri Setleriyle Problem Çözme” Öğrenme Deneyiminin Biçimlendirici Değerlendirmesi” adlı araştırmasında 7. sınıf öğrencileri ile 12 kişiden oluşan bir çalışma grubu

oluşturmuştur. Nitel yöntem kullanılan araştırmada öğrencilere yönelik “veri setleriyle problem çözmeye” etkinlikler planlanarak uygulanmıştır. Araştırmanın sonunda, öğrencilerin bilgi işlemsel düşünme becerilerini, problem çözmeye etkinliklerine aktardıkları görülmüştür.

Tutulmaz (2019), araştırmasında öğrencilerde bilgi işlemsel düşünme becerisinin gelişmesine katkı sağlayabilecek bir veri görselleştirme uygulamasının tasarlanması, geliştirilmesi ve değerlendirilmesini amaçlamıştır. Araştırma bu amaçla ortaöğretim kademesinde 11.sınıf öğrencileri ile gerçekleştirilmiştir. Uygulama aşaması için döngüler tasarlanmış ve döngüler kendi içinde “problem analizi”, “çözüm tasarlama”, “çözüm geliştirme” “uygulamada değerlendirme” ve “yansıma” evrelerini içerecek biçimde iki döngüden oluşmuştur. Araştırmanın sonucunda döngüler şeklinde geliştirilen veri görselleştirme uygulama sürecinin bilgi işlemsel düşünme becerileri üzerinde etkili olduğu ortaya konmuştur.

Kaya, Korkmaz ve Çakır (2020) çalışmaları ile oyunlaştırılmış eğitsel robot etkinlikleri ile öğrencilerde problem çözmeye yansıtıcı düşünce ve bilgi işlemsel düşünce becerilerini gözlemlemeyi hedeflemiştir. Araştırma BTY dersi kapsamında 6.sınıf öğrencileri ile yürütülmüştür. Araştırma sonucunda oyunlaştırılmış robot etkinliklerinin, öğrencilerin bilgi işlemsel düşünce ve problem çözmeye yönelik yansıtıcı düşünce becerilerine olumlu anlamda katkı sağladığı görülmüştür.

BÖLÜM 3

3. YÖNTEM

Çalışmanın bu bölümünde araştırmada kullanılan yöntem ayrıntılı bir şekilde anlatılmaktadır. Araştırmanın çalışma grubu, araştırma sorularının cevaplanması, araştırma modeli, veri toplama araçları, araştırma sorularının cevaplanması için gerekli olan ve kullanılan istatistiksel testler, araştırmanın bölümleri ve araştırma uygulaması hakkında bilgilere yer verilmiştir.

3.1 Araştırmanın Modeli

Blok tabanlı kodlama eğitiminin ortaokul öğrencilerin bilgi işlemsel düşünme becerisine ve kodlama öğrenimine yönelik tutumlarına etkisinin araştırıldığı bu araştırmada karma(mixed) yöntem benimsenmiştir. Karma(mixed) yöntem, tek bir araştırmada nicel ve nitel yöntemlerin birlikte kullanılarak yaklaşımlarının birleştirildiği bir araştırma yöntemi olarak nitelendirilmektedir (Johnson & Onwuegbuzie, 2004). Bir başka ifade ile karma yöntem, tamamıyla farklı yaklaşımlar ile yapılandırılmış nitel ve nicel desene bağlı yöntem ve tekniklerin bütünleşmesi olarak ifade edilmektedir (Alkan, Şimşek ve Armağan Erbil, 2019). Yalnızca nicel yöntemden faydalanılarak yapılan çalışma sonuçlarının, durumları betimlemede yetersiz kaldığından hareketle nitel yöntemin de araştırmada kullanılmasının sonuçların anlamlandırılmasında daha etkili olacağı kanaatine varılmıştır.

Araştırmada nicel veriler bilgi işlemsel düşünme becerisi öz yeterlik algısı ölçeği ve kodlama öğrenmeye yönelik tutum ölçeği ile nitel veriler ise araştırmacı tarafından geliştirilen öğrenme sürecine ilişkin görüş bildirme formu ile toplanmıştır. Araştırma farklı açılardan öğrencilere etkisinin araştırılması nedeniyle deneme modelindedir. Deneme modelleri, neden-sonuç ilişkilerini belirlemeye çalışmak amacıyla, doğrudan araştırmanın kontrolü altında gözlenmek istenen verilerin üretildiği araştırma modelleridir (Karasar, 2009). Ön test- son test kontrol gruplu tekrarlanan ölçümlerden oluşan yarı deneysel bir yöntem uygulanmıştır. Ön test-son test kontrol gruplu deneysel desenin kullanıldığı çalışmalarda; deneklerin araştırma uygulamasının hem öncesinde hem de sonrasında, bağımlı değişken ile ilgili ölçüme tabi tutulmaları ile akademik çalışmalarda uygulanmaktadır. Ayrıca, tüm değişkenlerin kontrol altına alınmasının mümkün olmadığı hallerde (Cohen vd., 2013) ve özellikle eğitim teknolojisi alanında

yapılan çalışmalarda araştırmacılar tarafından en sık kullanılan desendir (Kılıç-Çakmak vd., 2013). Ön test-son test kontrol gruplu deneysel desendeki çalışmalarda katılımcılar deney ve kontrol grubu olarak iki gruba ayrılmaktadır. Bu gruplar rastlantısal olarak birisi deney birisi kontrol grubu olarak belirlenmektedir. Denekler araştırma uygulamasının hem öncesinde hem de sonrasında, bağımlı değişken ile ilgili ölçümlere tabi tutulur (Karasar, 2009). Ön test ve son test farklı zamanlarda uygulanan aynı içerikten oluşan testlerden meydana gelmektedir. Bir başka değişle her iki grupta da aynı şekilde deney öncesinde ve deney sonrasında ölçümler yapılmaktadır (Büyüköztürk vd., 2012). Süreç içerisinde deney grubu öğrencilerine blok tabanlı kodlama kapsamında 4 haftalık blocky eğitimi verilmesi planlanmıştır. Kontrol grubu ise yıllık plan dâhilinde öğretime devam etmiştir. Bu kapsamda öğrencilerin bilgi-ışlemsel düşünme becerileri öz yeterlik algıları ve kodlama öğrenmeye ilişkin tutumlarına yönelik olarak ön test ve son testler uygulanarak nicel veriler toplanmıştır.

Yapılan bu çalışmada bağımsız değişken blok tabanlı kodlama eğitiminde kullanılan “blocky” uygulaması olarak belirlenmiştir. Araştırmadaki bağımlı değişkenler ise bilgi işlemsel düşünme becerisi ve kodlama öğrenmeye yönelik tutumlarıdır. Araştırmaya yönelik yarı deneysel desen Tablo 1 de gösterilmiştir.

Tablo- 1 Araştırmaya Yönelik Yarı Deneysel Desen

Grup	Ön Test	İşlem	Son Test
D	O ₁	X _{BLC}	O _{2-NG}
K	O ₁	X _G	O ₂

D = Deney grubu

K = Kontrol grubu

X_{BLC} = Blocky uygulaması ile verilmiş kodlama eğitimi

X_G = Geleneksel öğretim yöntemleri ile verilmiş kodlama eğitimi

O₁ = Deney ve Kontrol grubu ön test uygulaması

O₂ = Deney ve Kontrol grubu son test uygulaması

NG = Deney grubu öğrencilerinden nitel verilerin toplanması

Deney ve kontrol grubuna uygulama öncesi (ön testler) uygulanan bilgi işlemsel düşünme becerisine yönelik öz yeterlik ölçeği ve kodlama eğitimine yönelik tutum ölçeği sonuçlarının karşılaştırma (ilişkisiz örneklem için bağımsız t-testi) sonucu Tablo 2 ve Tablo 3’ de verilmiştir.

Tablo- 2 Gruplar Arası Bilgi İşlemsel Düşünme Becerisine Yönelik Öz yeterlik Ölçeği Ön Test Karşılaştırma Sonuçları

	Gruplar	N	\bar{X}	Ss	Sd	T	p
Ön test	Deney grubu	19	73,78	11,83			
	Kontrol grubu	19	67,94	10,94	36	1,579	,123*

*p<0.05

Uygulama öncesi deney ve kontrol grupları belirlendikten sonra, deney ve kontrol grubuna yapılan ön testlerde (deney grubu ön test ortalaması =73,78; kontrol grubu ön test ortalaması =67,94) *p<.05 anlamlılık düzeyi için .05<p olduğu için anlamlı değildir.

Tablo- 3 Gruplar Arası Kodlama Eğitimine Yönelik Tutum Ölçeği Ön Test Karşılaştırma Sonuçları

	Gruplar	N	\bar{X}	Ss	Sd	t	p
Ön test	Deney grubu	19	135,47	16,53			
	Kontrol grubu	19	133,89	19,68	36	,268	,790*

*p<0.05

Uygulama öncesi deney ve kontrol grupları belirlendikten sonra, deney ve kontrol grubuna yapılan ön testlerde (deney grubu ön test ortalaması =135,47; kontrol grubu ön test ortalaması =133,89) *p<.05 anlamlılık düzeyi için .05<p olduğu için anlamlı değildir (Tablo 3). Bir başka deyişle çalışma grubu sayılarının eşit olmasından ve yapılan istatistik testlerinden (Bilgi İşlemsel Düşünme Becerisine Yönelik Öz yeterlik Ölçeği ve Kodlama Eğitimine Yönelik Tutum Ölçeği) bulunan bu sonuçlarla her iki grubun araştırma öncesi eşit olduğu belirlenmiştir.

3.2. Çalışma Grubu

Araştırmanın çalışma grubunu 2019-2020 eğitim öğretim yılı 2.dönem içerisinde Aksaray'da bir ortaokulda 5. sınıfta öğrenim görmekte olan 38 öğrenci oluşturmaktadır. 19 öğrenci deney grubu, 19 öğrenci ise kontrol grubu olarak belirlenmiştir (Tablo 4).

Araştırmada deney ve kontrol grubu yansız olarak belirlenmiştir. Araştırma, yapısı sebebi ile deney ve kontrol gruplarının öncesinde algoritma, kodlama vb. eğitimleri almamış olmasını gerektirmektedir. Hali hazırda uygulanmakta olan müfredata incelendiğinde 6. 7. ve 8. sınıfların Bilişim Teknolojileri ve Yazılım dersi kapsamında daha önce kodlama eğitimi almış olabilecekleri düşünülerek çalışma grubunun BTY dersi kapsamında kodlama eğitimini ilk kez alacak olan 5.sınıflardan oluşturulmasına karar verilmiştir. Öğrencilerin araştırmaya katılabilmesi için velilerden gerekli izin alınmıştır. Veli izin dilekçesi EK-4’ de sunulmuştur.

Tablo- 4 Çalışma Grupları Yüzde-Frekans Dağılımı

Gruplar	N	%
Deney Grubu	19	50,0
Kontrol Grubu	19	50,0
Toplam	38	100,0

Öğrencilerin Demografik Özellikleri

Araştırmaya katılan 38 öğrencinin cinsiyet, internet bağlantısına sahip olma durumu, BİT araçlarından herhangi birine sahip olma durumu ve bu araçları günlük ne kadar süre kullandığı verileri göz önünde bulundurularak dağılımları incelenmiştir. Bu dağılımlar aşağıda tablolar halinde verilmiştir.

Tablo- 5 Çalışma Grupları Cinsiyet Dağılımı Yüzde Frekans Tablosu

Çalışma Grupları	Kız		Erkek		Toplam	
	f	%	f	%	f	%
Deney Grubu	10	52,6	9	47,3	19	50,0
Kontrol Grubu	10	52,6	9	47,3	19	50,0
TOPLAM	20	52,6	18	100,0	38	100,0

Öğrencilerin cinsiyet değişkenine göre dağılımı tablo 5’ de gösterilmiştir. Çalışma grubunun %52,6’ sı kız, %47,3’ ü erkek olmakla birlikte tüm öğrenciler ilköğretim 5.sınıf seviyesi 10-11 yaş aralığındadır. (Tablo 5).

Tablo- 6 Çalışma Grupları İnternet Bağlantısına Sahip Olma Durumları Yüzde Frekans Tablosu

İnternet Bağlantısına Sahip Olma Durumu	N	%
Var	15	39,5
Yok	23	60,5
TOPLAM	38	100,0

Tablo 6 ‘da çalışma gruplarının internet bağlantısına sahip olma durumlarının dağılımı verilmiştir. Bu dağılıma göre katılımcıların %39,5 ‘i internet erişimine sahipken, %60,5’i ise internet erişimine sahip değildir.

Tablo- 7 Çalışma Grupları Tablet/Telefon/Bilgisayar Sahiplik Durumları Yüzde Frekans Tablosu

Tablet/telefon/bilgisayar Sahiplik Durumu	N	%
Var	21	55,3
Yok	17	44,7
TOPLAM	38	100,0

Tablo 7 ‘de çalışma gruplarının BİT araçlarına sahip olma durumlarının dağılımı verilmiştir. Bu dağılıma göre katılımcıların %55,3’ü tablet/ telefon/ bilgisayar gibi araçlardan herhangi birine sahipken, %44,7’si bu araçlardan herhangi birine sahip değildir.

Tablo- 8 Çalışma Grupları Tablet/Telefon/Bilgisayar Günlük Kullanım Süreleri Yüzde Frekans Tablosu

Günlük BİT kullanım süresi	N	%
0-3 saat	22	57,9
3-6 saat	14	36,8
6 saat ve üzeri	2	5,3
TOPLAM	38	100,0

Tablo 8 'de çalışma gruplarının BİT araçlarını günlük kullanım sürelerine ilişkin dağılım verilmiştir. Bu dağılıma göre katılımcıların %57,9'u günde 0-3 saat arası, %36,8'i 3-6 saat arası, %5,3 'ü ise 6 saat ve üzeri BİT araçlarını kullanmaktadır.

3.3. Veri Toplama Araçları

Bu araştırmada üç farklı veri toplama aracından faydalanılmıştır. Öğretim sürecinin etkililiğini ölçmek adına araştırmacı tarafından geliştirilen öğrenme sürecine yönelik görüş bildirme formu kullanılmıştır. Bununla birlikte Gülbahar, Kert ve Kalelioğlu (2018) tarafından geliştirilen bilgi işlemsel düşünme becerisine yönelik öz yeterlik algısı ölçeği ve Karaman ve Büyükalan Filiz (2019) tarafından geliştirilmiş olan kodlama eğitimine yönelik tutum ölçeği kullanılmıştır. Ölçeklere ve görüş bildirme formuna demografik bilgilerin toplanacağı ve araştırma bilgisinin verileceği bir bölüm eklenmiştir. Veri toplama amacıyla kullanılacak ölçme araçları hakkında detaylı bilgi başlıklar halinde bu bölümde verilmiştir.

3.3.1. Demografik Bilgi Formu

Demografik bilgi formu ölçeklerin başına eklenerek öğrencilere ait cinsiyet bilgisi, evde internet bağlantısına sahip olma durumu, kendisine ait tablet/telefon/bilgisayara sahip olma durumu, BİT araçlarını günlük kullanım süresi ve daha önce tablet veya bilgisayar ile kodlama yapıp yapmadığı bilgisi alınmıştır.

3.3.2. Öğrenme Sürecine İlişkin Görüş Bildirme Formu

Çalışmada araştırmacı tarafından geliştirilen bu ölçme aracı öğrencilerin öğrenme sürecine ilişkin duygu ve tutumları hakkında veri toplamak amacıyla kullanılmıştır. "Öğrenme sürecine ilişkin görüş bildirme formu" hazırlık sürecinde iki Türkçe öğretmeni ve üç BT öğretmeninin görüşleri alınmıştır. Uzman görüşlerinin de doğrultusunda gerekli düzenlemeler yapılarak 13 maddeden oluşan nihai form oluşturulmuştur. Hazırlanan görüş bildirme formu, "Ders senin için eğlenceli geçti mi?", "Ders esnasında kafa karışıklığı yaşadın mı?", "Bundan sonraki süreçlerde kodlama konusunda kendini geliştirip bir oyun tasarlamak ister misin?" gibi sorular içermektedir. Öğrencilere evet-hayır olarak iki cevap seçeneği sunulmuştur. Hazırlanan görüş formu EK- 3' de verilmiştir.

3.3.3. Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algısı Ölçeği (BİDBÖA)

Çalışmada öğrencilerin bilgi işlemsel düşünme becerileri öz yeterliklerini ölçmek amacıyla Gülbahar, Kert ve Kalelioğlu (2019) tarafından geliştirilen BİDBÖA ölçeği (EK- 1) kullanılmıştır. Ölçek “evet”, “kısmen”, “hayır” seçenekleri ile üçlü likert tipinde olup 36 maddeden oluşmaktadır. BİDBÖA ölçeği Ankara ve İstanbul illerinde bulunan 17 farklı ortaokuldaki 916 5. ve 6.sınıf öğrencisine uygulanarak analizleri yapılmıştır. Ölçek üzerinde yapılan açımlayıcı faktör analizi sonuçlarına göre BİDBÖA beş alt boyuttan oluşmaktadır. Bu boyutlar; 9 maddeden oluşan “Algoritma Tasarlama Yeterliği”, 11 maddeden oluşan “Problem Çözme Yeterliği”, 7 maddeden oluşan “Veri İşleme Yeterliği”, 6 maddeden oluşan “Temel Programlama Yeterliği” ve 6 maddeden oluşan “Özgüven Yeterliği” olarak belirlenmiştir. Analizler sonucunda ölçek toplam güvenilirlik katsayısı (Cronbach alfa) .94 olarak bulunmuştur. Bu değer ölçeğin oldukça güvenilir olduğunu göstermektedir. EK- 5’ de gösterildiği üzere ölçeğin kullanımı için gerekli izin alınmıştır.

3.3.4. Kodlama Eğitimine Yönelik Tutum Ölçeği (KEYTÖ)

Çalışmada öğrencilerin kodlama eğitimine yönelik tutumlarını ölçmek için Karaman ve Büyükalın Filiz (2019) tarafından geliştirilmiş olan kodlama eğitimine yönelik tutum ölçeği (EK- 2) kullanılmıştır. Beşli likert tipinde “kesinlikle katılmıyorum”, “katılmıyorum”, “kısmen katılıyorum”, “katılıyorum”, “kesinlikle katılmıyorum” şeklinde hazırlanmış olan tutum ölçeği 41 maddeden oluşmaktadır. 2018-2019 eğitim-öğretim yılında Ankara ili Keçiören ilçesindeki 5 farklı ortaokulda okuyan 503 öğrenciye uygulanarak analizleri yapılmıştır. Analizler sonucunda Kaiser-Meyer-Olkin (KMO) değeri .96 bulunmuş ve bu değer kabul edilebilir düzeyde olduğu görülmüştür. AFA sonucunda toplam 41 madde için 2 faktörlü yapı için açıklanan varyansın %47,89 olduğu saptanmıştır. Yapılan analizler sonucunda elde edilen veriler, KEYTÖ’ nin Ortaokulda Yer Alan Kodlama Eğitimine Yönelik Öğrenci Tutumlarını ölçmede geçerli ve güvenilir olduğunu göstermektedir. KEYTÖ 28 maddeden oluşan “Kodlama Eğitimine Yönelik Genel Olumlu Tutum” ve 13 maddeden oluşan “Kodlama Eğitimine Yönelik Genel Olumsuz Tutum” olmak üzere iki boyuttan oluşmaktadır. Ölçekte ters kodlanan madde bulunmamaktadır. EK- 5’ de gösterildiği üzere ölçeğin kullanımı için gerekli izin alınmıştır.

3.4. Verilerin Toplanması

Çalışma kapsamındaki veriler 2019-2020 eğitim öğretim yılının 2.döneminde toplanmıştır. Araştırma izni için gerekli formlar uygulamanın yapılacağı okul aracılığı ile il milli eğitim müdürlüğüne ve valilik makamına gönderilmiştir. Gerekli izinler EK-6 ve EK- 7 de sunulmuştur. EK- 4 belirtilen veli izin dilekçesi ile çalışmaya katılacak öğrenci velilerinden izin alındıktan sonra veri toplama araçları uygulanmıştır. Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algısı ve Kodlama Eğitimine Yönelik Tutuma ilişkin veriler blok tabanlı kodlama eğitiminin öncesinde ve sonrasında toplanmıştır. Öğrenme sürecine ilişkin görüş bildirme formu ise blok tabanlı kodlama eğitimi verildikten sonra deney grubu öğrencilerine uygulanmıştır.

3.5. Araştırma ve Uygulama Süreci

Yapılan çalışmanın araştırma süreci 3 ayrı evreden meydana gelmektedir. Uygulaması öncesi hazırlık evresi, uygulama evresi, uygulama sonrası evre olmak üzere üç ayrı evreyi kapsamaktadır. İlgili evreler aşağıdaki çizelgede verilmiştir (Tablo- 9).

Tablo- 9 Uygulama Evreleri

HAZIRLIK (1. ve 2. HAFTA)	UYGULAMA AŞAMASI (2. – 6. HAFTA)		SÜREÇ DEĞERLENDİRME
	DENEY GRUBU	KONTROL GRUBU	
<ul style="list-style-type: none">• Öğrencilerin süreç hakkında bilgilendirilmesi.• Araştırma için gerekli izin başvurularının yapılması.	<ul style="list-style-type: none">• Ön testlerin uygulanması.• Blocky uygulaması ile öğrencilere blok tabanlı kodlama eğitiminin verilmesi.• Son testlerin uygulanması.• Etkileşimli tahta.• Bilgisayar	<ul style="list-style-type: none">• Ön testlerin uygulanması.• Bilgisayarsız kodlama etkinlikleri ile blok tabanlı kodlama eğitiminin verilmesi.• Son testlerin uygulanması.• Etkileşimli tahta.	<ul style="list-style-type: none">• Öğrenme sürecine ilişkin görüş bildirme formunun deney grubu öğrencilerine uygulanması.

Araştırma sürecinin hazırlık aşaması iki hafta olarak belirlenmiştir. Bu aşamada öğrenciler süreçle ilgili bilgilendirilip, oryantasyonları sağlanmıştır. Araştırma izni için başvurular yapıp, gerekli izinler alınmıştır.

Dört hafta olarak belirlenen uygulama aşamasında, ilk hafta uygulama öncesi deney ve kontrol grubu olmak üzere her iki gruba da “Bilgi İşlemsel Düşünme Becerisi Öz Yeterlik Ölçeği” ve “Kodlama Eğitimine Yönelik Tutum Ölçeği” ön test olarak uygulanmıştır. Planlanan ders anlatımları deney ve kontrol grubu için haftada 2 ders saati olmak üzere 4 hafta boyunca devam etmiştir. Eğitim sürecinde anlatılacak konuların ve kullanılacak etkinliklerin belirlenmesinde yıllık planlardan ve ders kitabından faydalanılmıştır. Ayrıca kullanılan içerikler düzenlenirken; yakından uzağa, kolaydan zora, bilinenden bilinmeyene gibi öğrenme ilkeleri göz önünde bulundurulmuştur (Şimşek, 2014). Derslerde kontrol grubunda çoğunlukla anlatım ve gösterip yaptırma yöntemleri kullanılırken deney grubunda bu yöntemlerin yanında problem çözme ve soru cevap yöntemleri de kullanılmıştır. Ders anlatımlarının hemen sonrasında iki gruba ayrı ayrı son testler uygulanmıştır. Uygulama aşamasına ilişkin hazırlanan 4 haftalık plan ayrıntıları ile tablo- 10’ da verilmiştir.

Tablo- 10 Uygulama Aşamasına İlişkin Ders Planı Özet Tablo

Hafta	Konu	Kazanım	Süre	Etkinlik	
				Deney	Kontrol
1	Problem Çözme Kavramları ve Yaklaşımları	<ul style="list-style-type: none"> Algoritma kavramını açıklar. Bir problemin çözümü için algoritma geliştirir. Bir algoritma için akış şeması çizer 	40+40 dakika	<ul style="list-style-type: none"> Kek Yapıyorum Robotun Rotası Eyvah Akış Şemaları Karışmış 	<ul style="list-style-type: none"> Kek Yapıyorum Robotun Rotası Eyvah Akış Şemaları Karışmış
2	Programlama	<ul style="list-style-type: none"> Blok tabanlı programlama aracının arayüzünü tanır. Blok tabanlı programlama ortamında sunulan hedeflere ulaşmak için algoritma oluşturur. Doğrusal mantık yapısı açıklar ve 	40+40 dakika	<ul style="list-style-type: none"> Blocky Oyunlarını Tanıyorum Şimdi Bulmaca Çözelim Labirente Giriş 	<ul style="list-style-type: none"> Bilgisayarsız Kodlama Etkinlikleri

		doğrusal mantık yapısını içeren algoritmalar geliştirir.			
3	Programlama	<ul style="list-style-type: none"> • Karar yapısını ve işlevlerini açıklar. • Döngü yapısını ve işlevlerini açıklar. 	40+40 dakika	<ul style="list-style-type: none"> • Labirente Giriş • Bu Labirent Beni Durduramaz 	<ul style="list-style-type: none"> • Bilgisayarsız Kodlama Etkinlikleri
4	Programlama	<ul style="list-style-type: none"> • Döngü yapısını içeren algoritmalar geliştirir. • Karar yapılarını içeren algoritmalar geliştirir. 	40+40 dakika	<ul style="list-style-type: none"> • Kısa Yolu Buluyorum • Haydi Kuş Yuvaya 	<ul style="list-style-type: none"> • Bilgisayarsız Kodlama Etkinlikleri

İlk hafta deney ve kontrol gruplarının her ikisine de ön testler uygulandıktan sonra algoritma ve akış şemaları konusunda eğitim verilmiştir. 40 dakikalık 2 ders saatinden oluşan eğitimde, Tablo- 10' da yer alan etkinliklerden yararlanılmıştır. Algoritma eğitimine giriş aşamasında öncelikle öğrencilere günlük hayattan örnekler verilmiş ve öğrencilerden benzer örnekler istenmiştir. Ardından “Kek Yapıyorum” (EK- 8) etkinliği ile öğrencilerin konuyu pekiştirmesi sağlanmıştır. “Robotun Rotası” (EK- 8) etkinliği ile öğrencilerin yön kavramlarını da kullanarak algoritmalar oluşturması sağlanmıştır. Ayrıca sınıfın kareli zemininden faydalanarak çeşitli objelerin hedef olarak belirlenmesi ile öğrenciler yön komutlarını bire bir uygulayarak derse etkin katılmışlardır. İkinci ders akış şemaları kavramının tanımı yapılmış ve akış şemalarındaki şekillerin ne anlama geldiği anlatılmıştır. Öğrencilerden kek yapıyorum algoritmasını akış şemaları ile yazmaları istenmiştir. Ardından öğrencilere “Eyvah Akış Şemaları Karışmış” (EK- 8) etkinliğinin çalışma kağıtları dağıtılmıştır. Çalışma kağıdında verilen 3 farklı senaryoya ait karışık olarak yer alan akış şemalarını, sıra arkadaşları ile birlikte senaryolara uygun şekilde oluşturmaları istenmiştir.

Sonraki haftalarda kontrol grubu öğrencileri ile bilgisayarsız kodlama etkinlikleri için hazırlanan materyal (Şekil- 6) ile eğitime devam edilmiştir. Haftalık kazanımlara göre, materyal üzerine yerleştirilen karakter ve engeller ile çeşitli senaryolar oluşturulmuştur. Öğrenciler gruplandırılmış ve oluşturulan senaryolar, materyalde yer alan blok parçaları (Şekil- 7) öğretmen rehberliğinde bir araya getirilerek çözülmüştür. Süreçte öğrencilere gerekli dönüt ve düzeltmeler sağlanmış ve

öğrencilerden alınan geribildirimler ile uygulama sonlandırılmıştır. Uygulamanın son haftasında dersin son 10 dakikası “Bilgi İşlemsel Düşünme Becerisi Öz Yeterlik Ölçeği” ve “Kodlama Eğitime Yönelik Tutum Ölçeği” son test uygulaması için ayrılmıştır.



Şekil- 6 Bilgisayarsız Kodlama Materyali



Şekil- 7 Bilgisayarsız Kodlama Materyali- Komutlar

İkinci hafta deney grubundaki öğrencilere öncelikle Blockly ortamının arayüzü tanıtılmıştır. Öğrencilere bireysel bilgisayarlarından ortamın arayüzünü inceleyebilmeleri için süre verilmiştir. Ardından öğrencilerin temel blokları ve bu blokların birbiri ile bağlantısını anlayabilmeleri için ilk aşama olan bulmaca kategorisine geçilir. Hedefin; her hayvanın resmini, bacak sayısını ve özelliklerini yeşil bloklar üzerinde birleştirmek olduğu “şimdi bulmaca çözelim” etkinliği ile blokların kullanımı pekiştirilmiştir. Ardından öğrencilerin doğrusal mantık yapısını kavrayabilmesi için labirent kategorisinin ilk iki seviyesi ile “labirente giriş” etkinliğine

geçilerek ders tamamlanmıştır. Ders boyunca öğrencilere gerekli seviyeleri tamamlayabilmeleri için gerekli süre verilmiş, öğrencilere rehberlik yapılmış, dönüt ve düzeltmeler sağlanmıştır.

Üçüncü hafta deney grubu öğrencileri ile labirent kategorisi ile “labirente giriş” etkinliğine devam edilmiştir. 3.seviyeden itibaren birlikte döngü bloğu (Şekil- 8) sürece dahil edilmektedir. 3.seviyeden itibaren 6.seviyeye kadar olan bulmacalar ile öğrencilerin döngü yapısını ve işlevlerini kavramaları amaçlanmıştır. Ayrıca bu seviyelerde yer alan belirli blok sayısı ile seviyeyi tamamlama şartı öğrencilerin döngü bloğunun kullanım amacını daha kolay kavramaları sağlamıştır. Birlikte yapılan 3.seviye ile öğrenciler tekrar eden komutlarda döngü bloğunu kullanarak labirenti daha kolay çözebileceklerinin farkına varmışlardır. 4. ve 5. Seviyeler için öğrencilere 15 dakika süre verilmiştir. Bu süre içinde öğrenci bilgisayarları kontrol edilerek gerekli rehberlik yapılmıştır. Tüm öğrencilerin seviyeleri eksiksiz tamamladığından emin olduktan sonra 6. Seviyeden 10. Seviyeye kadar olan bulmacaları kapsayan “bu labirent beni durduramaz” etkinliğine geçilmiştir. Bu etkinlikle birlikte 6.seviyeden itibaren koşul ifadeleri (Şekil- 8) sürece dahil edilmektedir. Koşul ifadelerinin yapıları ve hangi durumlarda kullanıldıkları öğrencilere 6.seviyedeki labirentin çözümü ile etkileşimli tahtadan anlatılmıştır. Öğrencilerin koşul ifadeleri ile ilgili soruları cevaplandırıldıktan sonra 7. ve 8. seviyeleri bireysel bilgisayarlarından çözmeleri için öğrencilere 15 dakika süre verilmiştir. Öğrencilerin seviyeleri tamamlaması beklenirken çalışmalarını öğretmen bilgisayarından takip edilerek gerekli dönüt ve düzeltmeler yapılmıştır. Öğrencilere tanınan sürenin ardından 7. ve 8. seviyelerin çözümleri, etkileşimli tahta üzerinden öğrencilere de söz hakkı verilerek tamamlanmıştır. 9. ve 10. seviye ev ödevi olduğu belirtilerek ders tamamlanmıştır.



Şekil- 8 Koşul Bloğu ve Döngü Bloğu

Dördüncü hafta deney grubu öğrencilerinden dersin ilk dakikalarında labirent kategorisindeki aşamaları nasıl geçtikleri ve bu süreçte kullanılan kavramları hatırlamaları istenmiştir. Kısa bir tekrarın ardından kuş kategorisi ile birlikte “kısa yolu buluyorum” ve “haydi kuş yuvaya” etkinliğine geçilmiştir. Kuş kategorisindeki 1.seviyeyi öğrencilerin çözmeleri için 5 dakika süre verilmiştir. Bu sürede öğrencilere gerekli yönlendirmeler ile rehberlik yapılmıştır. Verilen sürenin ardından etkileşimli tahtadan seviyenin çözümü için gerekli kodlar incelenmiştir. Ardından 2. seviyeye geçilmiştir. 2. seviye ile birlikte “Eğer yap- Değilse” (Şekil 9) bloğu sürece dahil olmaktadır. Bu bloğun işlevi ve hangi durumlarda kullanıldığı etkileşimli tahta üzerinden 2. seviyenin çözümleri ile anlatılmıştır. Ardından öğrencilere 10 dakika süre verilerek 2. ve 3. seviyeyi bireysel bilgisayarlarından çözmeleri istenmiştir. Bu süre içerisinde öğrenci bilgisayarları kontrol edilerek gerekli dönüt ve düzeltmeler sağlanmıştır. Öğrencilere verilen sürenin ardından 3. seviye de etkileşimli tahtadan yansıtılarak çözülmüştür. Öğrencilerden de konunun anlaşılıp anlaşılmadığı ile ilgili geri dönütler alınarak ders tamamlanmıştır. Dersin kalan 15 dakikasında öğrencilere “Bilgi İşlemsel Düşünme Becerisi Öz Yeterlik Ölçeği” (EK-1) ve “Kodlama Eğitimine Yönelik Tutum Ölçeği” (EK-2) son test olarak uygulanmıştır. Son olarak “Öğrenme Sürecine İlişkin Görüş Bildirme Formu” (EK-3) nicel verileri desteklemek ve süreç değerlendirmeyi gerçekleştirmek amacı ile öğrencilere uygulanarak süreç tamamlanmıştır.



Şekil- 9 Koşul Bloğu

3.5 Verilerin Analizi

Verilerin analizi sürecinde öncelikle araştırmaya katılan öğrencilerin demografik bilgileri ile birlikte ölçme araçlarına verdikleri yanıtlar geçerliliği kontrol edilerek bilgisayar ortamına aktarılmıştır. Kodlama Eğitimine Yönelik Tutum ölçeği beşli likert tipinde bir ölçektir ve “1-Kesinlikle Katılmıyorum”, “2-Katılmıyorum”, “3-Kısmen”, “4-Katılıyorum”, “5-Kesinlikle Katılıyorum” şeklinde puanlanmıştır. BİDBÖA ölçeği ise üçlü likert tipinde bir ölçektir ve “1-Evet”, “2-Kısmen”, “3-Hayır” şeklinde puanlanmıştır. Ölçeklerden toplanan veriler bu puanlama üzerinden Microsoft Excel programına aktarılmıştır.

Çalışma grubuna ait demografik veriler yüzde ve frekans, aritmetik ortalama, standart sapma gibi betimsel istatistikler ile sunulmuştur. Araştırmada üzerinde çalışılan durumları çözümlmek amacıyla nicel ve nitel verilerden faydalanıldığından verilerin analizinde farklı analiz yöntemleri kullanılmıştır. Elde edilen nicel verilerin çözümlenmesinde SPSS (Statistical Package for the Social Sciences) paket programından faydalanılmıştır ve bulgular 0,95 güven ($p= 0.05$) düzeyinde test edilmiştir. Deneysel işlem sürecinden geçen öğrencilerin araştırma öncesinde uygulanan ön test ile araştırma sonrasında uygulanan son testten toplanan verilerin karşılaştırılmasında ilişkili örneklem için t-testi kullanılmıştır. İlişkisiz örneklem için t-testi, iki ilişkisiz örneklem ortalamaları arasındaki farkın anlamlı olup olmadığını test etmek için kullanılır (Büyüköztürk, 2011).

Nitel verilerin çözümlenmesinde ise içerik analizi yönteminden yararlanılmıştır. İçerik analizi yönteminde nitel araştırmadaki sorular ve cevaplar birbirleriyle ilişkili olacak şekilde bir araya getirilerek gruplandırılır (Özdemir, 2010). Bu yöntem ile verilerin ayrıştırılması, ana öğelerin belirlenmesi, verilerin belli temalar altında düzenlenmesi ve ortaya çıkan sayısal verilerin yorumlanması şeklinde dört bölümden oluşmaktadır (Yıldırım ve Şimşek, 2006). Bu çalışmanın nitel araştırması kısmında; içerik analizi çeşitlerinden frekans analizi kullanılmıştır. Bu analiz türü en basit şekilde birimleri yüzde ve frekans olarak ortaya koyarken herhangi bir ögenin yoğunluğunu anlamaya yardımcı olmaktadır (Tavsancıl ve Aslan, 2001).

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 Araştırmaya Dair Nicel Bulgular

Bilgi İşlemsel Düşünme Becerisi Öz Yeterlik Ölçeğine ilişkin uygulama sonrasında çalışma grubundan toplanan veriler ile yapılmış olan güvenilirlik testinde iç güvenilirlik katsayısı olan Cronbach's Alfa güvenilirlik katsayısı .870 olarak bulunmuştur (Tablo-11).

Tablo- 11 Bilgi işlemsel düşünme becerisine yönelik öz yeterlik ölçeği güvenilirlik analizi

Cronbach's Alfa	Ölçekteki Madde Sayısı
.870	36

Kodlama Eğitimine Yönelik Tutum Ölçeğine ilişkin uygulama sonrasında çalışma grubundan toplanan veriler ile yapılmış olan güvenilirlik testinde iç güvenilirlik katsayısı olan Cronbach's Alfa güvenilirlik katsayısı .847 olarak bulunmuştur (Tablo- 12).

Tablo- 12 Kodlama eğitime yönelik tutum ölçeği güvenilirlik analizi

Cronbach's Alfa	Ölçekteki Madde Sayısı
.847	41

Deney grubu Ön test – Son test karşılaştırması (eşleştirilmiş t testi)

Uygulama sonucunda deney grubu öğrencilerinin bilgi işlemsel düşünme becerisine yönelik durumlarını belirlemek amacı ile yapılan ön test ve son testlerin karşılaştırmalarının sonucu Tablo 13’de verilmiştir.

Tablo- 13 Bilgi işlemsel düşünme becerisine yönelik öz yeterlik ölçeği için deney grubu ön test-son test karşılaştırma sonuçları

Test	N	\bar{X}	Ss	Sd	t	p
Ön test	19	73,78	11,83			
Son test	19	135,47	16,53	18	27,16	.000

*P<0.05

Deney grubu ön test-son test puanları arasında (ön test ortalaması =73,78; son test ortalaması =135,47) istatistiksel olarak $p<.05$ anlamlılık düzeyi için farklı olduğu görülmüştür ($p<.05$). Deney grubu öğrencilerinin katılmış olduğu Blocky ortamında blok tabanlı kodlama eğitimi sonrasında bilgi işlemsel düşünme becerisine yönelik öz yeterliklerini arttırdıkları belirlenmiştir (Tablo- 13).

Deney grubu Ön test – Son test karşılaştırması (eşleştirilmiş t testi)

Uygulama sonucunda deney grubu öğrencilerinin kodlama eğitimine yönelik tutumlarını belirlemek amacı ile yapılan ön test ve son testlerin karşılaştırılmasının sonucu Tablo 14’de verilmiştir.

Tablo- 14 Kodlama eğitimine yönelik tutum ölçeğine için deney grubu ön test-son test karşılaştırma sonuçları

	Test	N	\bar{X}	Ss	Sd	t	p
Deney Grubu	Ön test	19	96,84	7,25	18	58,15	.000
	Son test	19	163,10	16,69			

* $P<.05$

Deney grubu ön test-son test puanları arasında (ön test ortalaması =96,84; son test ortalaması =163,10) istatistiksel olarak $*p<.05$ anlamlılık düzeyi için farklı olduğu görülmüştür ($p<.05$). Deney grubu öğrencilerinin katılmış olduğu Blocky ortamında blok tabanlı kodlama eğitimi sonrasında öğrencilerin kodlama eğitimine yönelik tutum puanlarının arttığı belirlenmiştir (Tablo- 14).

Kontrol grubu Ön test – Son test karşılaştırması (eşleştirilmiş t testi)

Uygulama sonucunda kontrol grubu öğrencilerinin bilgi işlemsel düşünme becerisine yönelik durumlarını belirlemek amacı ile yapılan ön test ve son testlerin karşılaştırmalarının sonucu Tablo 15’de verilmiştir.

Tablo- 15 Bilgi işlemsel düşünme becerisine yönelik öz yeterlik ölçeği için kontrol grubu ön test-son test karşılaştırma sonuçları

	Test	N	\bar{X}	Ss	Sd	t	p
Kontrol Grubu	Ön test	19	67,94	10,94	18	27,05	,000
	Son test	19	133,89	19,68			

* $p<.05$

Kontrol grubu ön test son test puanları arasında (ön test ortalaması =67,94; son test ortalaması =133,89) istatistiksel olarak * $p<.05$ anlamlılık düzeyi için farklı olduğu görülmüştür ($p<.05$). Kontrol grubu öğrencilerinin katıldıkları bilgisayarsız blok tabanlı kodlama eğitimi uygulaması sonucunda bilgi işlemsel düşünme becerisine yönelik öz yeterlik algılarında anlamlı bir farklılık olduğu belirlenmiştir (Tablo- 15).

Kontrol grubu Ön test – Son test karşılaştırması (eşleştirilmiş t testi)

Uygulama sonucunda kontrol grubu öğrencilerinin kodlama eğitimine yönelik tutumlarını belirlemek amacı ile yapılan ön test ve son testlerin karşılaştırmalarının sonucu Tablo 16’da verilmiştir.

Tablo- 16 Kodlama eğitimine yönelik tutum ölçeği için kontrol grubu ön test-son test karşılaştırma sonuçları

	Test	N	\bar{X}	Ss	Sd	t	p
Kontrol Grubu	Ön test	19	81,68	10,00	18	35,60	.000
	Son test	19	143,57	19,00			

* $p<.05$

Kontrol grubu ön test son test puanları arasında (ön test ortalaması =81,68; son test ortalaması =143,57) istatistiksel olarak * $p<.05$ anlamlılık düzeyi için farklı olduğu görülmüştür ($p<.05$). Kontrol grubu öğrencilerinin katıldıkları bilgisayarsız blok tabanlı kodlama eğitimi uygulaması sonucunda öğrencilerin kodlama eğitimine yönelik tutumları arasında anlamlı bir farklılık olduğu belirlenmiştir (Tablo- 16).

Deney-kontrol grubu son testler karşılaştırması (bağımsız t testi)

Blok tabanlı kodlama eğitiminde Blocky ortamını kullanan öğrenciler (deney grubu) ile kullanmayan öğrencilerin (kontrol grubu), bilgi işlemsel düşünme becerisine yönelik öz yeterlik ölçeği son test puanları karşılaştırıldığında ortaya çıkan sonuç Tablo 17’de verilmiştir.

Tablo- 17 Bilgi işlemsel düşünme becerisine yönelik öz yeterlik ölçeği için gruplar arası (deney- kontrol grubu) son test karşılaştırma (t- testi) sonuçları

	Gruplar	N	\bar{X}	S	Sd	t	p
Son test	Deney grubu	19	96,84	7,25	36	5,34	.000*
	Kontrol grubu	19	81,68	10,00			

Uygulama sonrası deney ve kontrol grubuna yapılmış olan son testlerde $*p<.05$ anlamlılık düzeyi için $.00 <.05$ olduğundan dolayı anlamlıdır. Yapılan son testlerde (deney grubu son test ortalaması =96,84; kontrol grubu son test ortalaması =81,68) deney grubunun son test puanlarının, kontrol grubunun son test puanlarından daha yüksek olduğu görülmüştür (Tablo- 17). Bu sonuca göre gerçekleştirilen uygulamanın deney grubu lehine olduğu söylenebilir. Ayrıca Blocky ortamının, bilgi işlemsel düşünme becerisi üzerindeki etki büyüklüğünü belirlemek amacı ile eta kare değeri incelenmiştir. Etki büyüklüğü değeri $\eta^2= .126$ olarak hesaplanmıştır. Bu durumda, etki büyüklüğü değeri ($\eta^2= 0.126$) göz önünde bulundurularak, Blocky ortamının, bilgi işlemsel düşünme becerisine yönelik öz yeterlik algısı üzerinde “geniş” bir etki büyüklüğüne sahip olduğu çıkarımı yapılabilir.

Deney-kontrol grubu son testler karşılaştırması (bağımsız t testi)

Blok tabanlı kodlama eğitiminde Blocky ortamını kullanan öğrenciler (deney grubu) ile kullanmayan öğrencilerin (kontrol grubu), kodlama eğitimine yönelik tutum ölçeği son test puanları karşılaştırıldığında çıkan sonuç Tablo 18’de verilmiştir.

Tablo- 18 Kodlama eğitimine yönelik tutum ölçeğine için gruplar arası (deney-kontrol grubu) son test karşılaştırma (t- testi) sonuçları

	Gruplar	N	\bar{X}	S	Sd	t	p
Son test	Deney grubu	19	163,10	16,69	36	3,36	.002*
	Kontrol grubu	19	143,57	19,00			

*P<0.05

Uygulama sonrası deney ve kontrol grubuna yapılan son testlerde $*p<.05$ anlamlılık düzeyi için $.00 <.05$ olduğu için anlamlıdır. Yapılan son testlerde (deney grubu son test ortalaması =163,10; kontrol grubu son test ortalaması =143,57) deney grubunun son test puanlarının, kontrol grubunun son test puanlarından daha yüksek olduğu görülmüştür (Tablo- 18). Bu sonuca göre gerçekleştirilen uygulamanın deney grubunun lehine olduğu söylenebilir. Ayrıca Blocky ortamının, öğrencilerin kodlama eğitimine yönelik tutumu üzerindeki etki büyüklüğünü belirlemek amacı ile eta kare değeri incelenmiştir. Etki büyüklüğü değeri $\eta^2= .121$ olarak hesaplanmıştır. Bu durumda, etki büyüklüğü değeri ($\eta^2= 0.121$) göz önünde bulundurularak Blocky

ortamının, öğrencilerin kodlama eğitimine yönelik tutumları üzerinde “geniş” bir etki büyüklüğüne sahip olduğu söylenebilir.

4.1 Araştırmaya Dair Nitel Bulgular

Araştırma sonucunda ulaşılan nicel verileri desteklemek ve öğrenim sürecinin verimliliği hakkında bilgi sahibi olmak amacı ile son testlerin ardından deney grubu öğrencilerine görüş bildirme formu uygulanmıştır. Forma ait veriler tablo- 19’ da yüzde ve frekans değerleri ile belirtilmiştir.

Tablo- 19 Deney grubu öğrencilerinin öğrenme sürecine ilişkin görüşleri

		N	%
1- Derste problem çözme sürecinde zorlukla karşılaştın mı?	<i>Evet</i>	5	26,4
	<i>Hayır</i>	14	73,6
2- Ders senin için eğlenceli geçti mi?	<i>Evet</i>	17	89,4
	<i>Hayır</i>	2	10,6
3- Ders esnasında kendini mutlu hissettin mi?	<i>Evet</i>	16	84,2
	<i>Hayır</i>	3	15,8
4- Ders ilgi çekici miydi?	<i>Evet</i>	18	94,7
	<i>Hayır</i>	1	5,3
5- Bu dersi tekrar almak ister misin?	<i>Evet</i>	19	100
	<i>Hayır</i>	0	0
6- Ders esnasında kafa karışıklığı yaşadın mı?	<i>Evet</i>	4	21,1
	<i>Hayır</i>	15	78,9
7- Ders esnasında sormak isteyip soramadığın şeyler oldu mu?	<i>Evet</i>	1	5,3
	<i>Hayır</i>	18	94,7
8- Ders esnasında kendini yetersiz hissettin mi?	<i>Evet</i>	3	15,8
	<i>Hayır</i>	16	84,2
9- Ders esnasında seni rahatsız eden durumlar oldu mu?	<i>Evet</i>	0	0
	<i>Hayır</i>	19	100
10- Dersten sıkıldığın anlar oldu mu?	<i>Evet</i>	0	0
	<i>Hayır</i>	19	100
11- Bu dersten öğrendiklerinle diğer derslerdeki başarımı olumlu etkileyeceğini düşünüyor musun?	<i>Evet</i>	18	94,7
	<i>Hayır</i>	1	5,3
12- Bundan sonraki süreçte kodlama konusunda kendini geliştirip bir oyun tasarlamak ister misin?	<i>Evet</i>	17	89,4
	<i>Hayır</i>	2	10,6

Tablo 19 incelendiğinde görülmektedir ki formda öğrencilere yöneltilen derse yönelik 12 tutum sorusundan pozitif tutumların ölçüldüğü 6 soruya öğrencilerin tamamına yakını “Evet” yanıtını vermiştir. 5. Maddede yer alan “Bu dersi tekrar almak ister misin?” sorusuna öğrencilerin tamamı evet yanıtını vermiştir. Negatif tutumların

ölçüldüğü 6 soruya ise yine öğrencilerin tamamına yakını “Hayır” yanıtını vermiştir. 9. maddede yer alan “Ders esnasında seni rahatsız eden durumlar oldu mu?” ve 10. maddede yer alan “Dersten sıkıldığın anlar oldu mu?” sorusuna öğrencilerin tamamı “Hayır” yanıtını vermiştir. Dersin ilgi çekici olup olmadığı ile alakalı olan maddeye öğrencilerin %94,7’ si “Evet” cevabını vermiştir. Ayrıca öğrenciler 11.maddeye %94,7 oranında “Evet” cevabını vererek bu derste öğrendiklerinin diğer derslerde başarılarını olumlu anlamda etkileyeceğini düşündüklerini belirtmişlerdir.

Negatif tutum içeren sorulara verilen yanıtlar incelendiğinde öğrencilerin ders esnasında anlaşılmayan noktaları sormakta çekingenlik yaşadıkları bir durum olmadığı görülmüştür. 6. Madde incelendiğinde %78,9 oranında öğrencinin ders esnasında kafa karışıklığı yaşamadığı görülmektedir. Olumlu tutum içeren sorulara hayır cevabının en fazla verildiği durum, %15,8 oranla “Ders esnasında kendini mutlu hissettin mi?” sorusu olmuştur. Olumsuz tutum içeren sorulara evet cevabının en fazla verildiği durum ise, %26,4 oranla “Derste problem çözme sürecinde zorlukla karşılaştın mı?” sorusu olmuştur.

BÖLÜM 5

5. TARTIŞMA, SONUÇ VE ÖNERİLER

5.1 Sonuç ve Tartışma

Araştırmanın bu bölümünde çalışma sonucunda elde edilen bulgular doğrultusunda ulaşılan sonuçlar alan yazındaki benzer sonuçlar ile birlikte tartışılmıştır. Araştırmanın çalışma grubunu oluşturan 38 öğrenciye ait demografik veriler şu şekildedir:

- Çalışma grubunu oluşturan öğrencilerden 5.sınıf kademesinden oluşmakta ve 20' si kız, 18' i erkektir.
- Çalışma grubunu oluşturan öğrencilerin 15'i evlerinde internet bağlantısına sahipken, 23 öğrencinin evlerinde internet bağlantısı yoktur.
- Çalışma grubunu oluşturan öğrencilerin 21'i tablet/telefon/bilgisayar gibi BİT araçlarından herhangi birine sahipken, 17 öğrenci bu araçlardan herhangi birine sahip değildir.
- Çalışma grubunu oluşturan öğrencilerin günlük BİT araçlarını kullanım süreleri 0-3 saat, 3-6 saat ve 6 saat ve üzeri olarak gruplandırılmıştır. Öğrencilerin 22' si BİT araçlarını günlük 0-3 saat, 14'ü 3-6 saat, 4' ü ise 6 saat ve üzeri kullanmaktadır.

Araştırma kapsamında deney grubuna verilen Blockly ile blok tabanlı kodlama öğretimi sonrasında sürecin etkililiğini değerlendirmek amacı ile deney grubu öğrencilerine uygulanan görüş bildirme formundan elde edilen bilgilere göre;

- “*Derste problem çözme sürecinde zorlukla karşılaştın mı?*” sorusuna deney grubunu oluşturan 5 öğrenci evet yanıtını verirken 14 öğrencinin hayır yanıtını verdiği gözlemlenmiştir.
- “*Ders senin için eğlenceli geçti mi?*” sorusuna deney grubunu oluşturan 17 öğrenci evet yanıtını verirken 2 öğrencinin hayır yanıtını verdiği gözlemlenmiştir. Fidan (2016) araştırmasında kodlama ile öğrencilerin derse motivasyonlarının ve katılımlarının arttığını, böylece sürecinin daha eğlenceli geçtiğini belirtmektedir.

- “*Ders esnasında kendini mutlu hissettin mi?*” sorusuna deney grubunu oluşturan 16 öğrenci evet yanıtını verirken 3 öğrencinin hayır yanıtını verdiği gözlemlenmiştir.
- “*Ders ilgi çekici miydi?*” sorusuna deney grubunu oluşturan 18 öğrenci evet yanıtını verirken 1 öğrencinin hayır yanıtını verdiğini gözlemlenmiştir.
- “*Bu dersi tekrar almak ister misin?*” sorusuna deney grubunu oluşturan öğrencilerin tamamının evet cevabını verdiği gözlemlenmiştir. Öğrencilerin kodlama dersine yönelik algılarının incelendiği araştırmalarda, öğrencilerin kodlama etkinliklerine karşı istekli ve ilgili oldukları görülmüştür (Kasalak, 2017).
- “*Ders esnasında kafa karışıklığı yaşadın mı?*” sorusuna deney grubunu oluşturan öğrencilerin 4’ü evet yanıtını verirken 15’inin hayır yanıtını verdiği gözlemlenmiştir.
- “*Ders esnasında sormak isteyip sormadığın şeyler oldu mu?*” sorusuna deney grubunu oluşturan öğrencilerin 1’i evet yanıtını verirken 18’inin hayır yanıtını verdiği gözlemlenmiştir.
- “*Ders esnasında kendini yetersiz hissettin mi?*” sorusuna deney grubunu oluşturan öğrencilerin 3’ü evet yanıtını verirken 16’ sının hayır yanıtını verdiği gözlemlenmiştir. Çavdar (2018) kodlama öğretiminde kullanılan çevrimiçi ortamları değerlendirdiği çalışmasında da öğrenciler kodlama dersinde kendilerini büyük ölçüde yeterli hissettikleri ve ders esnasında rahatsızlık veren herhangi bir durum olmadığını ifade etmişlerdir.
- “*Ders esnasında seni rahatsız eden durumlar oldu mu?*” sorusuna deney grubunu oluşturan öğrencilerin tamamının hayır yanıtını verdiği gözlemlenmiştir.
- “*Dersten sıkıldığın anlar oldu mu?*” sorusuna deney grubunu oluşturan öğrencilerin tamamının hayır yanıtını verdiği gözlemlenmiştir.
- “*Bu dersten öğrendiklerin ile diğer derslerdeki başarını olumlu etkileyeceğini düşünüyor musun?*” sorusuna deney grubunu oluşturan öğrencilerin 1’i hayır yanıtını verirken 18’inin evet yanıtını verdiği gözlemlenmiştir.
- “*Bundan sonraki süreçte kodlama konusunda kendini geliştirip bir oyun tasarlamak ister misin?*” sorusuna deney grubunu oluşturan öğrencilerin 17’si evet yanıtını

verirken 2 öğrencinin hayır yanıtını verdiği gözlemlenmiştir. Tağci (2019) kodlama eğitiminin ilkökul öğrencileri üzerindeki etkisini gözlemlediği araştırmasında öğrencilerin eğitimden sonraki süreçlerde de bu alanda becerilerini geliştirerek, kendi oyunlarını yapmak istediklerini belirtmiştir.

Araştırmanın nicel boyutunda deneysel uygulama sonrası toplanan sayısal verilerin analizi için istatistik paket programı olan SPSS 22 (Statistical Package for Social Sciences) programından yararlanılmıştır. Deneysel uygulama sürecine dahil olan öğrencilere araştırma öncesinde uygulanan ön test ile araştırma sonrasında uygulanan son testten toplanan verilerin karşılaştırılmasında ilişkili örneklem için t-testi kullanılmıştır. Sonuçlar $p=0,05$ anlamlılık düzeyinde analiz edilmiştir. Bu analize göre;

Uygulama sonucunda deney grubu öğrencilerinin katıldıkları blocky ile blok tabanlı kodlama eğitimi sonrasında bilgi işlemsel düşünme becerisine yönelik öz yeterlik algılarının arttığı belirlenmiştir (Tablo- 13). Deney grubu ön test-son test puanları arasında (ön test ortalaması =73,78; son test ortalaması =135,47) $p<.05$ anlamlılık düzeyi için istatistiksel olarak farklılık olduğu görülmüştür ($p<0.05$). Benzer bir sonuç olarak Lye & Koh (2014) yaptıkları çalışmada bilgi işlemsel düşünme becerisinin kazandırılmasında kodlama öğretiminin en etkili yol olduğu belirtmiştir. Ancak Şimşek (2018) tarafından yapılan ve programlama eğitiminde robotik uygulamalar ve Scratch' in bilgi işlemsel düşünme becerisi ve akademik başarı üzerindeki etkisini incelemeyi hedefleyen çalışmada tam tersi sonuçlara ulaşılmıştır. 5. Sınıf öğrencileri ile yürütülmüş olan çalışma sonuçlarına göre öğrencilerin akademik başarı ve bilgi işlemsel düşünme puanlarında anlamlı bir farklılaşma olmadığı görülmüştür. Uslu, Mumcu ve Eğin (2018) yapmış oldukları çalışmada görsel tabanlı kodlama etkinliklerinin bilgi işlemsel düşünme becerisi üzerine etkisini incelemiş, bilgi işlemsel düşünme becerileri üzerinde istatistiksel açıdan anlamlı bir farklılık bulunmadığını ancak kodlama eğitimi ile öğrencilerin hayal gücünün geliştirilebileceği sonucuna ulaşılmıştır. Alsancak Sırakaya (2019) programlamanın öğrencilerin BİD becerisi üzerine olan etkisini incelemeyi amaçladığı araştırmasında BİD düşünme puanı açısından çalışma grubunun öntest- sontest puanları arasında anlamlı farklılık olduğu bulgusuna ulaşmıştır.

Uygulama sonucunda deney grubu öğrencilerinin katıldıkları blocky ile blok tabanlı kodlama eğitimi sonrasında öğrencilerinin kodlama eğitimine yönelik

tutumlarında anlamlı bir artışın olduğu belirlenmiştir (Tablo- 14). Deney grubu ön test-son test puanları arasında (ön test ortalaması =96,84; son test ortalaması =163,10) $p<.05$ anlamlılık düzeyi için istatistiksel farklılık olduğu görülmüştür ($p<0.05$). Alan yazında da bu sonucu destekler nitelikte araştırmalar mevcuttur. Genç ve Karakuş (2011) araştırmalarında BTP ortamlarının kodlamaya yeni başlayan gençlerin sistematik ve mantıksal düşünme becerilerini geliştirirken, merak duygularını artırarak onları araştırmaya sevk ettiğini ve özgüvenlerini de artırdığını belirtmiştir. Weintrop ve Wilensky (2015), araştırmalarında öğrencilerin BTP ortamlarının kodlamaya yeni başlayanlar için tasarlanmış bir öğrenme ortamı olduğunu ve BTP' nin normal kodlama süreçleri ile aynı fakat daha basit olduğunu söylediklerini ifade etmektedir. Böylece öğrencilerin kodlamaya yönelik tutumları da olumlu anlamda değişmektedir.

Blok tabanlı kodlama eğitiminde Blocky ortamını kullanan öğrenciler (deney grubu) ile kullanmayan öğrencilerin (kontrol grubu), bilgi işlemsel düşünme becerisine yönelik öz yeterlik ölçeği son test puanları karşılaştırıldığında, deney grubu öğrencilerinin puanlarını artırdığı belirlenmiştir (Tablo- 17). Uygulama sonrası deney ve kontrol grubuna yapılmış olan son testlerde $p<.05$ anlamlılık düzeyi için $.00 <.05$ olduğundan dolayı anlamlıdır. Yapılan son testlerde (deney grubu son test ortalaması =96,84; kontrol grubu son test ortalaması =81,68) deney grubunun son test puanlarının, kontrol grubunun son test puanlarından daha yüksek olduğu görülmüştür (Tablo- 17). Bu sonuçtan yola çıkarak, gerçekleştirilen uygulamanın deney grubu lehine olduğu söylenebilir. Ayrıca Blocky ortamının, bilgi işlemsel düşünme becerisi üzerindeki etki büyüklüğünü belirlemek amacı ile incelenen eta kare değeri ile etki büyüklüğü değeri $\eta^2 = .126$ olarak bulunmuştur. Bu durumda, etki büyüklüğü değeri ($\eta^2 = 0.126$) göz önünde bulundurularak, Blocky ortamının, bilgi işlemsel düşünme becerisine yönelik öz yeterlik algısı üzerinde “geniş” bir etki büyüklüğüne sahip olduğu çıkarımı yapılabilir. Numanoğlu ve Keser (2017) blok tabanlı kodlama platformlarının programlama öğretiminde kullanılabilirliğini ve bu platformların yapısını araştırmayı hedefledikleri çalışmalarında da benzer sonuçlara ulaşmışlardır. Çalışmanın sonucunda blok tabanlı kodlama yazılımları ile kodlama öğretiminde soyut kavramların somutlaştırabildiği ve öğrenenlerin problem çözme ve bilgi işlemsel düşünme gibi becerilerini hızlı bir şekilde geliştirebildiği görülmüştür. Rose, Habgood ve Jay (2017) da yaptıkları çalışmada benzer sonuçlara ulaşmışlardır. Çalışmada ScratchJr ve Lightbot blok tabanlı programlama araçları ile öğrencilerin programlama olan yaklaşımlarını ve bilgi işlemsel

düşünme becerileri üzerindeki etkisi incelenmiştir. Araştırma sonunda, uygulamanın her iki versiyonunda da performanslarının benzer olduğu sonucuna varılmıştır. İleride yapılacak araştırmalarda bu araçların algoritmik düşünme, ayrıştırma ve soyutlama gibi becerileri geliştirmedeki katkısının yanında bilgi işlemsel düşünme üzerine katkısının da ayrıntılı olarak ele alınması önerilmiştir. Oluk vd. (2018) tarafından yapılan çalışmada bir başka blok tabanlı kodlama aracı olan Scratch' in algoritma geliştirme ve BİD becerilerine etkisini incelemek amaçlanmıştır. Deney ve kontrol grubu yürütülen bu araştırmada gruplardan birinde mevcut müfredat ile eğitim yapılırken diğerinde Scratch ortamı kullanılmıştır. Araştırma sonucunda Scratch ile yapılan derste, diğer gruba göre öğrencilerin BİD becerisinin anlamlı derecede farklılaştığı görülmüştür. Ünsal (2020), blok tabanlı kodlama ortamlarının, ilkokul 2.sınıf düzeyinde kullanımının BİD becerisine etkisini Code.org örneği ile incelendiği çalışmasında, kodlama derslerinin code.org gibi uygulama ağırlıklı BTP ortamları ile işlenmesinin bilgi işlemsel düşünme becerisinin geliştirilmesinde etkili olduğu sonucuna varmıştır. Benzer bir araştırma da Minecraft görsel tabanlı kodlama dersleri ile yapılmış olup bu etkinliklerin öğrencilerde BİD becerilerini artırdığı sonucuna ulaşılmıştır (Kutay, 2020).

Blok tabanlı kodlama eğitiminde Blocky ortamını kullanan öğrenciler (deney grubu) ile kullanmayan öğrencilerin (kontrol grubu), kodlama eğitime yönelik tutum ölçeği son test puanları karşılaştırıldığında, deney grubu öğrencilerinin puanlarını artırdığı belirlenmiştir (Tablo 18). Uygulama sonrası deney ve kontrol grubuna yapılan son testlerde $*p < .05$ anlamlılık düzeyi için $.00 < .05$ olduğu için anlamlıdır. Yapılan son testlerde (deney grubu son test ortalaması =163,10; kontrol grubu son test ortalaması =143,57) deney grubunun son test puanlarının, kontrol grubunun son test puanlarından daha yüksek olduğu görülmüştür (Tablo 18). Bu sonuçtan yola çıkarak, gerçekleştirilen uygulamanın deney grubunun lehine olduğu söylenebilir. Ayrıca Blocky ortamının, öğrencilerin kodlama eğitime yönelik tutumu üzerindeki etki büyüklüğünü belirlemek amacı ile incelenen eta kare değeri ile etki büyüklüğü değeri $\eta^2 = .121$ olarak bulunmuştur. Bu durumda, etki büyüklüğü değeri ($\eta^2 = 0.121$) göz önünde bulundurularak Blocky ortamının, öğrencilerin kodlama eğitime yönelik tutumları üzerinde “geniş” bir etki büyüklüğüne sahip olduğu çıkarımı yapılabilir. Bishop- Clark, Courte ve Howard (2007) Alice kodlama uygulamasını kullanarak yaptıkları araştırmada benzer sonuçlar elde etmişlerdir. İki buçuk hafta boyunca verilen eğitimin ardından uygulanan son testlere göre öğrencilerin programlama kavramlarını anlama

düzeyleri, özgüvenleri ve programlama eğitiminden aldıkları keyifte anlamlı bir artış olduğu belirlenmiştir. Bir başka çalışmada da BTP ortamlarından yaygın olarak kullanılan Scratch' in kodlama eğitimlerine ve çocukların kodlamayı öğrenmeye ilişkin algıları konusunda olumlu yönde etki ettiği sonucuna ulaşılmıştır (Zuckerman, Blau & Monroy-Hernández, 2009). Tağci (2019) yapmış olduğu çalışmada görsel kodlama araçlarının öğrencilerin ilgi ve motivasyonlarını artırdığı, öğrencilerin sürece aktif bir şekilde dahil olmasını sağladığı ve bu duruma paralel olarak öğrencilerin başarılarını artırmada etkili olduğunu belirtmiştir. Tüm bunlarla birlikte öğrencilerin kodlamaya karşı olumlu tutum geliştirmelerinde de görsel kodlama araçlarının önemli bir rol oynadığını vurgulamıştır. Otu (2020)' nun BTP ortamlarından Scratch ve Mblock' u; metin tabanlı programlama ortamlarından Python' u kullanarak öğrencilerin kodlamaya yönelik tutumlarını incelediği çalışmasında blok tabanlı ortamların öğrencilerin kodlamaya yönelik tutumlarını olumlu etkilediği görülmüştür.

Araştırma sonucunda Blockly ortamında yapılan blok tabanlı kodlama eğitiminin, öğrencilerin BİD becerileri ve kodlama eğitime yönelik tutumları noktasında olumlu anlamda etki sağladığı görülmüştür. Bunun yanında öğrenci görüşleri doğrultusunda blocky ortamında verilen eğitimin öğrencilerin derse karşı ilgi ve motivasyonlarını artırdığı sonucuna ulaşılmıştır. Alan yazındaki çalışmalar incelendiğinde blok tabanlı kodlama eğitiminin bilgi işlemsel düşünme becerisi ve öğrencilerin kodlama eğitime yönelik tutumlarına etkisinin farklı faktörler, farklı yaş grupları ve farklı öğrenme ortamları ile incelendiği pek çok ulusal ve uluslararası çalışma olduğu belirlenmiştir. Yapılan bu araştırmalar da göstermektedir ki kodlama öğretimi çağın becerileri özellikle bilgi işlemsel düşünme becerisini kazandırma noktasında oldukça önemlidir. Çağın gerektirdiği becerilerden geride kalmamak adına BTY dersi için okullarda gerekli ortamlar oluşturulması ve öğrencilere kodlama eğitimi ile çeşitli ürünler ortaya koyabilecekleri yeterli zaman tanınabilmesi açısından ders saatlerinin de artırılmasının yararlı olacağı düşünülmektedir. Böylece bireyler kodlama eğitimi ile birlikte üst düzey düşünme becerilerini kazanarak tüketen konumdan üreten konuma geçebilirler.

5.2 Öneriler

Araştırma sonucundan elde edilen bulgulara göre aşağıdaki önerilerde bulunulmuştur:

- Bu çalışma blok tabanlı kodlama eğitiminin bilgi işlemsel düşünme becerisi üzerine etkisini araştırmak amaçlı yapılmıştır. Blok tabanlı kodlama eğitiminin eleştirel

düşünme ve problem çözme, işbirlikli çalışma ve yaratıcılık gibi farklı 21. Yüzyıl becerileri üzerine etkisi incelenebilir.

- Alanyazında kodlama öğretimi için bilgisayarsız etkinlikler, görsel ve metinsel kodlama ortamları vb. araçlar kullanıldığı görülmektedir. Bu araçların kodlama öğretiminde tutum, motivasyon akademik başarı ve düşünme becerileri noktasında olumlu etkileri olurken bazı araştırmalarda herhangi bir etkisinin olmadığı da görülmektedir. Buradan yola çıkarak kodlama eğitiminde kullanılan araçlar kadar kullanılan yöntemin de oldukça önemli olduğu söylenebilir. Bu kapsamda farklı öğretim yöntemleri ile harmanlanan kodlama öğretiminin etkililiği araştırılabilir.
- Bu çalışma 5. Sınıf öğrencileri ile yürütülmüştür. Blok tabanlı kodlama öğretiminin farklı yaş gruplarındaki öğrencilerin düşünme becerileri ve akademik başarıları üzerine etkisi incelenebilir.
- Çalışma 6 hafta gibi kısıtlı bir sürede 38 öğrenci ile yürütülmüştür. Araştırma sonuçlarının daha fazla genelleylebilmemiz için daha geniş bir örneklem ile uzun süreli bir araştırma yapılabilir.
- Bu çalışmada blok tabanlı kodlama eğitiminde Blocky kullanımının öğrencilerin akademik başarısı üzerine etkisi incelenmemiştir. İleride bu konuda yapılacak araştırmalarda akademik başarı da dikkate alınabilir.
- Kodlama eğitimini ilk defa alan öğrenciler için Blocky ve benzeri blok tabanlı araçların ve ortamların kullanılmasının daha faydalı olacağı önerilmektedir.
- İlköğretim kademesinde 5. ve 6. Sınıflarda kodlama öğretimi BTY dersi kapsamında haftada 2 saat olarak verilmektedir. Ders saatinin artırılması ile kodlama sürecinin daha verimli geçeceği ve öğrencilerin daha fazla öğrenme sürecine dahil olacağı düşünülmektedir.
- Kodlama eğitimi üzerine basılı materyaller artırılıp okullarda öğrencilerin erişimine sunulabilir.

KAYNAKÇA

- Aho, A. (2012). Computation and Computational Thinking. *The Computer Journal*, 55(7), 832- 835.
- Akçay, A. & Çoklar, A.N. (2016). Bilişsel Becerilerin Gelişimine Yönelik Bir Öneri: Programlama Eğitimi. Erişim Adresi: https://www.researchgate.net/publication/320517601_Bilissel_Becerilerin_Gelisimine_Yonelik_Bir_Oneri_Programlama_Egitimi
- Akçay, A. O., Karahan, E & Türk, S. (2019). Bilgi işlemsel düşünme becerileri odaklı okul sonrası kodlama sürecinde ilkokul öğrencilerinin deneyimlerinin incelenmesi. *Eskişehir Osmangazi Üniversitesi Türk Dünyası Uygulama ve Araştırma Merkezi (ESTÜDAM) Eğitim Dergisi (ESTUDAM Journal of Education)*, 4 (2), 38-50.
- Akkaş Baysal, E., Ocak, G. & Ocak, İ. (2020). Kodlama ve Arduino Eğitimleri ile İlgili Lise Öğrencilerinin Görüşleri. *Elektronik Sosyal Bilimler Dergisi*, 19 (74), 777-796.
- Akpınar, Y. & Altun, A. (2014). Bilgi Toplumu Okullarında Programlama Eğitimi Gereksinimi. *İlköğretim Online*, 13(1), 1-4.
- Akyol Altun, C. (2018). *Okul Öncesi Öğretim Programına Algoritma ve Kodlama Eğitimi Entegrasyonunun Öğrencilerin Problem Çözme Becerisine Etkisi*. Yüksek Lisans Tezi, Ankara Üniversitesi, Eğitim Bilimleri Enstitüsü, Ankara.
- Akyüz, A.O. (2018). *Çocuklara Programlama Nasıl Öğretilir? "Prof. Dr. Emine Akyüz'e Armağan Akademisyenlikte 50.Yıl*, Pegem Akademi, Nisan 2018, Ankara.
- Al-Jarrah, A.A.M. (2016). *Collaborative virtual environments for introductory programming (cveip)*. Doktora tezi, New Mexico State Üniversitesi, New Mexico.
- Alkan, V., Şimşek, S., & Armağan Erbil, B. (2019). Karma yöntem: Öyküleyici alanyazın incelemesi. *Eğitimde Nitel Araştırmalar Dergisi – Journal of Qualitative Research in Education*, 7(2), 559-582.
- Alsancak Sırakaya, D. (2019). Programlama Öğretiminin Bilgi İşlemsel Düşünme Becerisine Etkisi. *Türkiye Sosyal Araştırmalar Dergisi*, 23(2), 575-590.
- Anılan, H., & Gezer, B. (2020). Kodlama Etkinliklerine ve Analitik Düşünme Becerisine Yönelik Sınıf Öğretmenlerinin Görüşlerinin İncelenmesi. *Anadolu University Journal of Education Faculty*, 4(4), 307-324.

- Arabacıođlu, T., Bülbül, H. İ., & Filiz, A. (2007). Bilgisayar programlama öğretiminde yeni bir yaklaşım. *Akademik Bilişim Konferansı*, Dumlupınar Üniversitesi, Kütahya.
- Atiker, B. (2019). *Programlama Öğretiminde Ortaokul Öğrencilerinin Bilgi İşlemsel Düşünme Becerilerinin Başarıya Etkisi*, Doktora Tezi, İstanbul Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- Aydođdu, Ş. (2020). Blok Tabanlı Programlama Etkinliklerinin Öğretmen Adaylarının Programlamaya İlişkin Öz Yeterlilik Algılarına ve Hesaplamalı Düşünme Becerilerine Etkisi. *Eğitim Teknolojisi Kuram ve Uygulama*, 10(1), 303-320.
- Aytekin, A., Sönmez Çakır, F., Yücel, Y. B. ve Kulaözü, İ. (2018). Geleceğe yön veren kodlama bilimi ve kodlama öğrenmede kullanılabilir bazı yöntemler. *Avrasya Sosyal ve Ekonomi Araştırmaları Dergisi*, 5(5), 24-41.
- Bala, R.B. (2019). *6.Sınıf Öğrencilerine Programlama Dili Öğretilirken Kullanılan Scratch Programının Öğrencilerin Problem Çözme Becerilerine Ve Tutumlarına Etkisi*, Yüksek Lisans Tezi, Necmettin Erbakan Üniversitesi, Eğitim Bilimleri Enstitüsü, Konya.
- Balanskat, A. & Engelhardt, K. (2015). Computing our future -computer programming and coding- priorities, school curricula and initiatives across Europe. Erişim Adresi: <http://www.eun.org/documents/411753/817341/Coding+initiative+report+Oct2014/2f9b35e7-c1f0-46e2-bf72-6315ccbaa754>
- Barcelos, T., & Silveira, I. (2012). Teaching computational thinking in initial series an analysis of the confluence among mathematics and computer sciences in elementary education and its implications for higher education. *2012 XXXVIII Conferencia Latinoamericana En Infermatica (CLEI) içinde* (s. 1-8). Medellin, Colombia.
- Bayman, P. & Mayer, R. (1988). Using conceptual models to teach BASIC computer Programming. *Journal of Educational Psychology*, 80(3), 291-298.
- BBC. (2018). *Introduction to Computational Thinking*. <https://www.bbc.com/education/guides/zp92mp3/revision/1>, Erişim Tarihi: 23.11.2020.
- Bennedsen, J. & Caspersen, M. E. (2008). Exposing the programming process. Bennedsen, J., Carpersen, M. E. ve Kolling, M. (Eds.). *Reflection on the theory of programming: Methods and implementation* 6-16. New York: Springer Berlin Heidelberg.
- Bergin J. & Martinez M. P. (1996), An overview of visualization: its use and design,

Report of the Working Group on Visualization. *Integrating Tech. into C.S.E.*, 6(96). Erişim Adresi: https://www.researchgate.net/publication/220808003_An_overview_of_visualization_its_use_and_design_report_of_the_working_group_in_visualization

- Berikan, B. (2018). *Bilgi işlemsel düşünme becerisine yönelik tasarlanan “veri setleriyle problem çözme” öğrenme deneyiminin biçimlendirici değerlendirmesi* (Yayımlanmamış doktora tezi). Gazi Üniversitesi, Eğitim Bilimleri Enstitüsü, Ankara.
- Berksoy, İ., Sözcü, Ö., Armağan, E., & Arslan, A. (2016). *Algoritma ve Programlama Eğitiminde Scratch Programı Üzerine Yapılan Çalışmaların Analizi*. <http://ab.org.tr/ab16/bildiri/74.docx>, Erişim Tarihi: 27.04.2020
- Bers, M. U., González-González, C., & Armas-Torres, M. B. (2019). Coding as a playground: Promoting positive learning experiences in childhood classrooms. *Computers & Education*, 138, 130-145.
- Bilge Adam Koleji (2017). *"Kodlama Eğitimi"*, <http://www.bilgeadamkoleji.com/Shared%20Documents/Kodlama.pdf>, Erişim Tarihi: 4.01.2020.
- Bilge Kunduz (2019). *Bilge kunduz- uluslararası enformatik ve bilgi işlemsel düşünme etkinliği*. <http://www.bilgekunduz.org/>, Erişim Tarihi: 22.04.2020.
- Bishop-Clark, C., Courte, J. & Howard, E. V. (2007). A quantitative and qualitative investigation of using Alice programming to improve confidence, enjoyment and achievement among non-majors. *Journal of Educational Computing Research*, 37(2) 193-207.
- Blocky (2020). *Blockly Games : About*. <https://blockly.games/about?lang=tr>, Erişim Tarihi: 27.03.2020
- Briggs, J. D. (2013). *Programming with Scratch software: The benefits for year six learners*. Yayımlanmamış Yüksek lisans tezi, Bath Spa University, İngiltere.
- BTE Derneği (2013). *Türkiye’de ilk ve ortaokullarda (ilköğretim) okutulan bilişim teknolojileri derslerinin tarihi*. <http://bte.org.tr/bte-derneği/bt-derslerinin-tarihcesi/>, Erişim Tarihi: 06.11.2020.
- Bülbül Şoltan, E. (2018). *Blok Tabanlı Programlama Dili ile Bağlam Temelli Öğrenme Ortamında Oyun Geliştiren Öğrencilerin Kodlama Becerilerindeki Gelişme Düzeylerinin İncelenmesi*, Yüksek Lisans Tezi, Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul

- Büyüköztürk, Ş. (2011). *Sosyal Bilimler İçin Veri Analizi El Kitabı*. Ankara: Pegem Akademi.
- Büyüköztürk, Ş., Kılıç-Çakmak, E., Akgün, Ö. E., Karadeniz, Ş., ve Demirel, F. (2012). *Bilimsel araştırma yöntemleri*. (11. Baskı). Ankara: Pegem Akademi.
- Calao, L. A., Moreno-León, J., Correa, H. E. & Robles, G. (2015). Developing Mathematical Thinking with Scratch: an Experiment with 6th grade students. *Design for Teaching and Learning in a Networked World* (pp.17-27). Springer International Publishing.
- Calder, N. (2010). Using Scratch: an integrated problem-solving approach to mathematical thinking, *Australian Primary Mathematics Classroom*, 15(4), 9-14.
- Candar, M. K. (2016). *Microsoft Kodu Game Lab Programı Giriş- Ders 1*. <http://www.eba.gov.tr/video/izle/32796791ed506c1ef439faf5c93370999537eee560003>, Erişim Tarihi: 26.03.2020
- Chaudhary, V., Agrawal, V. & Sureka, A. (2016). An Experimental Study on the Learning Outcome of Teaching Elementary Level Children using Lego Mindstorms EV3 Robotics Education Kit. *T4E 2016 The IEEE International Conference on Technology for Education*.
- CodeWeek, (2020). *CodeWeek hakkında*, <https://codeweek.eu/about> Erişim Tarihi: 26.03.2020.

- Code.org (2020). "About Us", <https://code.org/international/about>, Erişim Tarihi: 26.03.2020
- CodeWizardshq (2016). *19 Free & Paid Coding Websites for Kids*. <https://codewizardshq.com/coding-websites-for-kids/>, Erişim Tarihi: 25.03.2020
- Cohen, L., Manion, L., & Morrison, K. (2013). *Research Methods In Education*. London: Routledge.
- Crow, D. (2014). *Why every child should learn to code*. *The Guardian*. <https://www.theguardian.com/technology/2014/feb/07/year-of-code-dan-crowsongkick>, Erişim Tarihi: 25.12.2020.
- Ctrlbizde.com (2020). *Blok Tabanlı Programlama Nedir?* <https://www.ctrlbizde.com/index.php/dersler/bty-bilisim-teknolojileri-ve-yazilim-dersi/item/493-blok-tabanlı-programlama-nedir-sunu>, Erişim Tarihi: 28.03.2020
- Cuiumju, V. (2013). *App Inventor ile programlamaya giriş*. <https://www.technopat.net/2013/05/20/app-inventor-programlamaya-giris/>, Erişim Tarihi: 25.03.2020
- Curzon, P. (2015). *Computational thinking: Searching to speak*. <http://teachinglondoncomputing.org/free-workshops/computational-thinking-searching-to-speak/>, Erişim Tarihi: 03.09.2019.
- Czerkawski, B. C. & Lyman, E. (2015). Exploring Issues About Computational Thinking in Higher Education, *TechTrends*, 59(2), 57-65.
- Çağiltay, N. E. ve Fal, M. (2014). *Scratch ile programlama öğreniyorum*, ODTÜ Yayıncılık, Ankara
- Çamoğlu, K. (2009). *Programlama & Veritabanı Mantığı*, Kodlab Yayıncılık, İstanbul.
- Çatlak, Ş., Tekdal, M. ve Baz, F. Ç., (2015). Scratch Yazılımı ile Programlama Öğretiminin Durumu: Bir Doküman İnceleme Çalışması. *Journal of Instructional Technologies & Teacher Education*, 4(3), 13-25.
- Çavdar, L. (2018). *Kodlama Öğretiminde Kullanılan Çevrimiçi Platformların Değerlendirilmesi: Code.Org Örneği*, Yüksek Lisans Tezi, Gazi Osmanpaşa Üniversitesi, Eğitim Bilimleri Enstitüsü, Tokat.
- Çınar, M., ve Tüzün, H. (2017). Eğitimde bilgisayarlı düşünme uygulamalarına ilişkin bir alanyazın incelemesi. *11.Uluslararası Bilgisayar ve Öğretim Teknolojileri Eğitimi Sempozyumu*. 24-26 Mayıs 2017, İnönü Üniversitesi, Malatya.

- Demir, B. & Demir, F. (2021). Coding, Robotics and Computational Thinking in Preschool Education: The Design of Magne-Board. *Avrupa Bilim ve Teknoloji Dergisi*, (23), 52-61.
- Demirer, V. ve Sak, N. (2016). Dünyada ve Türkiye'de programlama eğitimi ve yeni yaklaşımlar. *Eğitimde Kuram ve Uygulama*, 12(3), 521-546.
- Demirkol, Z. (2016). *Çocuklar İçin Kodlama*, Pusula Yayıncılık, İstanbul.
- Denning, P. J., (2003). Great principles of computing. *Communications of the ACM*, 46(11), 15-20.
- Denning, P. J. (2009). The Profession of IT-Beyond Computational Thinking. *Communications of the ACM*, 52 (6), 28-30.
- Durak, H., Karaoğlan-Yılmaz, F. G., Yılmaz, R., ve Seferoğlu, S. S. (2017). Erken yaşta programlama eğitimi: Araştırmalardaki güncel eğilimlerle ilgili bir inceleme. *The Turkish Online Journal of Educational Techonogy*, 119-137.
- Eğın, F. (2019). *Bilişim teknolojileri öğretmenlerinin kodlama öğretimine ilişkin görüşlerinin incelenmesi*, Yüksek lisans tezi, Ege Üniversitesi, İzmir.
- Erdem, E. (2018). *Blok Tabanlı Ortamlarda Programlama Öğretimi Sürecinde Farklı Öğretim Stratejilerinin Çeşitli Değişkenler Açısından İncelenmesi*, Yüksek Lisans Tezi, Başkent Üniversitesi Eğitim Bilimleri Enstitüsü, Ankara.
- Erol, O. (2015). *Scratch ile programlama öğretiminin bilişim teknolojileri öğretmen adaylarının motivasyon ve başarılarına etkisi*. Yayımlanmamış Doktora Tezi. Anadolu Üniversitesi, Eğitim Bilimleri Enstitüsü, Eskişehir.
- Ersoy, H. ve Aydın, S. (2015). Ortaokul Öğrencilerine Programlama Becerileri Kazandırmada Scratch' in Etkililiği. *Akademik Bilişim Konferansı*.
- Ersoy, H., Madran, R.O. ve Gülbahar, Y. (2006). Programlama Dilleri Öğretimine Bir Model Önerisi: Robot Programlama. *Akademik Bilişim 07 Konferansı*, Kütahya.
- Ersoy, H., Madran, R. O. ve Gülbahar, Y., (2011). "Programlama dilleri öğretimine bir model önerisi: robot programlama", *Akademik Bilişim 2011 Konferansı*.
- Erümit, A. K. (2018). *Okul Öncesinde Programlama Eğitimi*, http://okuloncesistem.mu.edu.tr/Icerik/okuloncesistem.mu.edu.tr/Sayfa/9_%20Okul%20%C3%96ncesi%20D%C3%B6nemde%20Kodlama.pdf Erişim Tarihi: 04.11.2020
- Erümit, A.K., Benzer, A.İ., Aksoy, D. A., Aksoy, A. ve Şahin, G. (2017) Algoritmik Düşünme için Programlama Öğretimi Adımları. *Eğitim Teknolojileri Okumaları*, 2017(1-15), Ankara: TOJET.

- Eryılmaz, S. (2003). *Algoritma tasarlama ve programlamaya giriş*, Detay Yayıncılık, Ankara.
- Eryılmaz, S., & Deniz, G. (2019) Türkiye’de Programlama Eğitimi İle İlgili Yapılan Çalışmaların İncelenmesi: Bir Betimsel Analiz Çalışması. *Eğitimde Kuram ve Uygulama*, 15(4), 319-338.
- Euronews (2015). *Avrupa'da bilgisayar programlama dersleri ilköğretim müfredatına giriyor*. <http://tr.euronews.com/2015/09/03/avrupa-da-bilgisayar-programlama-dersleri-ilkogretim-mufredatina-giriyor>, Erişim Tarihi: 04.11.2020.
- European Schoolnet (2015). *Computing our future Computer programming and coding - Priorities, school curricula and initiatives across Europe*, European Schoolnet, https://www.researchgate.net/publication/284139559_Computing_our_future_Computer_programming_and_coding_-_Priorities_school_curricula_and_initiatives_across_Europe Erişim Tarihi: 04.11.2020
- Fessakis, G., Gouli, E., & Mavroudi, E. (2013). Problem solving by 5–6 years old kindergarten children in a computer programming environment: A case study. *Computers & Education*, 63, 87-97.
- Fidan, A., (2016). *Scratch ile Programlama Öğretiminde Oyunlaştırmanın Öğrenci Katılımına Etkisi*, Yüksek Lisans Tezi, Uludağ Üniversitesi, Bursa.
- Frayner, J. M. (2016). *China Pushes Coding for Kids in Effort to Tackle Innovation Gap*, NBC News, <http://www.nbcnews.com/news/china/chinapushes-coding-kids-effort-tackle-innovation-gap-n641966>, Erişim Tarihi: 25.08.2020
- Futschek, G. (2006). Algorithmic thinking: the key for understanding computer science. In *International Conference on Informatics in Secondary Schools-Evolution and Perspectives* (pp.159-168).
- Garc a-Pe alvo F. J., Reimann D., Tuul M., Rees A., ve Jormanainen I., (2016), An overview of the most relevant literature on coding and computational thinking with emphasis on the relevant issues for teachers, *TACCLE3 Consortium*, Belgium, 25 Mayıs 2018.
- Garneli, V., Giannakos, M. N., & Chorianopoulos, K. (2015). Computing education in K-12 schools: A review of the literature. *Global Engineering Education Conference (EDUCON), 2015 IEEE* (pp. 543–551). IEEE.
- Genç, Z., & Karakuş, S. (2011). Tasarımla Öğrenme: Eğitsel Bilgisayar Oyunları Tasarımında Scratch Kullanımı, 5. *Uluslararası Bilgisayar ve Öğretim Teknolojileri Sempozyumu*, 22-24 Eylül 2011, Elâzığ, 981-987.

- Göncü, A., Çetin, İ., & Şendurur, P. (2020). Bilişim Teknolojileri ve Yazılım Dersi Öğretmenlerinin Kodlama Eğitimine Yönelik Görüşleri. *Mersin Üniversitesi Eğitim Fakültesi Dergisi*, 16 (2), 301-321.
- Gülbahar, Y., Kalelioğlu, F., Doğan, D. & Karataş, E. (2020). Bebras: A Social Approach for Concept Based Learning of Informatics and Computational Thinking. *Ankara University Journal of Faculty of Educational Sciences (JFES)*, 53 (1), 241-272.
- Gülbahar, Y., Kert, S. B., & Kalelioğlu F. (2019). Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algısı Ölçeği: Geçerlik ve Güvenirlik Çalışması. *Türk Bilgisayar ve Matematik Eğitimi Dergisi*, 10(1), 1-29.
- Habitat (2017). *Minik Parmaklar Geleceği Programlıyor*. <https://habitatderneği.org/dijital-donusum/minik-parmaklar-gelecegi-programliyor/>, Erişim Tarihi: 01.11.2020.
- Hatisaru, S. (2016). *Vodafone' dan çocuklara kod yazma dersi*. <https://www.milliyet.com.tr/yazarlar/songul-hatisaru/vodafone-dan-cocuklara-kod-yazma-dersi-2241732>, Erişim Tarihi: 07.11.2020.
- Haymana, İ. & Özalp, D. (2020). Robotik ve Kodlama Eğitiminin İlkokul 4. Sınıf Öğrencilerinin Yaratıcı Düşünme Becerilerine Etkisi. *İstanbul Aydın Üniversitesi Eğitim Fakültesi Dergisi*, 6 (2), 247-274
- Hill, C. (2015). *Programming environments for children: Creating a language that grows with you*. Yayınlanmamış Yüksek lisans tezi, Kaliforniya Üniversitesi, Kaliforniya.
- Howland, K. & Good, J. (2015). Learning to communicate computationally with Flip: A bi-modal programming language for game creation. *Computers & Education*, 80, 224-240.
- Hu, C. (2011). Computational thinking: what it might mean and what we might do About it. *Proceedings of the 16th Annual Joint Conference on Innovation and Technology in Computer Science Education- ITiCSE '11*, 223-227.
- ISTE & CSTA (2011). *Operational Definition of Computational Thinking for K-12 Education*. <http://www.iste.org/docs/ct-documents/computational-thinking-operational-definitionflyer.pdf>, Erişim Tarihi: 13.11.2020.
- ISTE (2016.). The ISTE National Educational Technology Standards (NETS•S) and Performance Indicators for Students. <http://www.iste.org/standards/standards/for-students-2016>, Erişim Tarihi: 14.11.2020.

- İzgöl, K. (2015). *Scratch Nedir?* <http://maker.robotistan.com/scratch-nedir>, Erişim Tarihi: 25.03.2020
- Jenkins, T. (2002). On the difficulty of learning to program. *Proceedings of 3rd annual conference of the LTSN-ICS*, 53-58, Loughborough, United Kingdom, Erişim Adresi: <https://www.psy.gla.ac.uk/~steve/loaled/jenkins.html>
- Johnson, R. B., & Onwuegbuzie, A. J. (2004). Mixed methods research: a research paradigm whose time has come. *Educational Researcher*, 33(7), 14–26.
- Kahraman, B. (2015). *Avustralya’da ilkokullarda programcılık eğitimi veriliyor.* <http://www.webtekno.com/sektorel/avustralya-da-ilkokullarda-programcilik-egitimi-veriliyor-h10859.html>, Erişim Tarihi: 04.11.2020.
- Kalelioğlu, F. (2015). A New Way of Teaching programming Skills to K-12 students: Code.org. *Computers in Human Behavior*, 52, 200-210.
- Kalelioğlu, F. & Gülbahar, Y. (2014). The Effects of Teaching Programming via Scratch on Problem Solving Skills. *Informatics in Education*, 13(1), 33–50.
- Karabak, D. & Güneş, A. (2013). Ortaokul birinci sınıf öğrencileri için yazılım geliştirme alanında müfredat önerisi. *Journal of Research in Education and Teaching*, 2(3), 163-169.
- Karaman, U., & Büyükalan Filiz S. (2019). Kodlama Eğitimine Yönelik Tutum Ölçeği ’nin (KEYTÖ) Geliştirilmesi. *Gelecek Vizyonlar Dergisi (fvj: Future Visions Journal)*, 3(2), 36-47.
- Karasar, N. (2009). *Bilimsel Araştırma Yöntemi*. Ankara: Nobel Yayınları.
- Kasalak, İ., (2017). *Robotik Kodlama Etkinliklerinin Ortaokul Öğrencilerinin Kodlamaya İlişkin Öz-yeterlik Algılarına Etkisi ve Etkinliklere İlişkin Öğrenci Yaşantıları*, Yüksek Lisans Tezi, Hacettepe Üniversitesi, Ankara.
- Kaya, M., Korkmaz, Ö. ve Çakır, R. (2020). Oyunlaştırılmış robot etkinliklerinin ortaokul öğrencilerinin problem çözme ve bilgi işlemsel düşünme becerilerine etkisi. *Ege Eğitim Dergisi*, 21(1), 54-70.
- Kert, S. B. & Uğraş, T. (2009). *Programlama Eğitiminde Sadelik ve Eğlence: Scratch Örneği*. 1st International Congress of Educational Research. 1st International Congress of Educational Research.
- Kılıç-Çakmak, E., Çebi, A., Mihçi, P., Günbatar, M., S., ve Akçayır, M. (2013). A content analysis of educational technology research in 2011. *In 4th International Conference on New Horizons in Education. INTE 2013*, Roma.

- Kodugamelab (2018). *About*. <https://www.kodugamelab.com/about/>, Eriřim Tarihi: 26.03.2020
- Kong, S. C., & Abelson, H. (2019). *Computational Thinking Education*. <https://library.oapen.org/handle/20.500.12657/23182> Eriřim Tarihi:05.06.2020.
- Korkmaz, Ö., akır, R. & Özden, M. Y. (2015). Bilgisayarca Düşünme Beceri Düzeyleri Ölçeğinin (BDBD) Ortaokul Düzeyine Uyarlanması, *Gazi Eğitim Bilimleri Dergisi*, 1(2), 143-162.
- Korucu, A. T., Gençtürk, A. T. & Gündoğdu, M. M. (2017). Examination of the Computational Thinking Skills of Students. *Journal of Learning and Teaching in Digital Age*, 2(1), 11-19.
- Kukul, V. (2018). *Programlama Öğretiminde Farklı Yapılandırılan Süreçlerin Öğrencilerin Bilgi İşlemsel Düşünme Becerilerine, Öz yeterliliklerine ve Programlama Başarılarına Etkisi*, Doktora Tezi, Gazi Üniversitesi, Ankara.
- Kutay, E. (2020). *Coding With Minecraft: The Development Of Middle School Students' Computational Thinking*, Yüksek Lisans Tezi, Boğaziçi Üniversitesi, İstanbul.
- Lau, W. W., & Yuen, A. H. (2011). Modelling programming performance: Beyond the influence of learner characteristics. *Computers & Education*, 57(1), 1202-1213.
- Lee, I., Martin, F., Denner, J., Coulter, B., Allan, W., Erickson, J., Malyn-Smith, J. & Werner, L. (2011). Computational Thinking for Youth in Practice, *ACM Inroads*, 2 (1), 32–37.
- Li, Y., Schoenfeld, A. H., diSessa, A. A., Graesser, A. C., Benson, L. C., English, L. D. & Duschl, R. A. (2020). Computational thinking is more about thinking than computing. *Journal for STEM Education Research* 3,1-18.
- Lockwood, J. & Mooney, A., (2017). Computational thinking in education: where does it fit? a systematic literary review. Maynooth University, Ireland.
- Lye, S. & Koh, J. (2014). Review on teaching and learning of computational thinking through programing: What is next for K-12? *Computers in Human Behavior*, 41, 51-61.
- Maloney, J. H., Peppler, K., Kafai, Y., Resnick, M.& Rusk, N. (2008). Programming by choice: urban youth learning programming with scratch. *ACM SIGCSE Bulletin*, 40(1), 367-371.
- Mannila, L., Dagiene, V., Demo, B., Grgurina, N., Mirolo, C., Rolandsson, L. & Settle, A. (2014). Computational thinking in K-9 education. In: Proceedings of the Working Group Reports of the 2014 on Innovation & Technology in *Computer Science Education Conference*, ITiCSE-WGR 2014, 1–29. ACM, New York.

- MEB. (2012). *Ortaokul ve imam hatip ortaokulu bilişim teknolojileri ve yazılım dersi (5, 6, 7 ve 8. sınıflar) öğretim programı*. <http://mufredat.meb.gov.tr/Programlar.aspx>, Erişim Tarihi: 15.09.2019
- MEB, (2016). *Bilgisayar bilimi dersi öğretim programı kur 1- kur 2*. <http://mufredat.meb.gov.tr/Dosyalar/2018120203611364-BILGISAYAR%20BILIMI%20DERSI%20OGRETIM%20PROGRAMI.pdf>, Erişim Tarihi: 06.11.2020
- Meccawy, M. (2017). “Raising a programmer: Teaching Saudi children how to Code”, *International Journal of Educational Technology*, 4(26), 56-65.
- Medlock-Walton, P., Harms, K. J., Kraemer, E. T., Brennan, K., & Wendel, D. (2014). Blocks-based programming languages: Simplifying programming for different audiences with different goals. In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education* (pp. 545-546), ACM 2014.
- Meerbaum-Salant, O., Armoni, M. ve Ben-Ari, M. (2013). Learning computer science concepts with Scratch. *Computer Science Education*, 233, 239–264.
- MIT App Inventor (2020). *About Us*, <http://appinventor.mit.edu/explore/about-us.html>, Erişim Tarihi: 25.03.2020
- Numanoğlu, M. & Keser, H. (2017). Programlama Öğretiminde Robot Kullanımı – Mbot Örneği. *Bartın Üniversitesi Eğitim Fakültesi Dergisi*, 6 (2), 497-515.
- Olson, P. (2012). *Why Estonia has started teaching its first-graders to code*. <http://www.forbes.com/sites/parmyolson/2012/09/06/why-estonia-has-started-teaching-its-first-graders-to-code/#125f6b675790>, Erişim Tarihi: 04.11.2020.
- Oluk A., (2017). *Öğrencilerin bilgisayarca düşünme becerilerinin mantıksal matematiksel zekâ ve matematik akademik başarıları açısından incelenmesi*. Unpublished master’s thesis, Amasya Üniversitesi, Fen Bilimleri Enstitüsü, Amasya.
- Oluk, A., Korkmaz, Ö. & Oluk, H.A. (2018). Scratch’ın 5. Sınıf Öğrencilerinin Algoritma Geliştirme ve Bilgi-İşlemsel Düşünme Becerilerine Etkisi. *Turkish Journal of Computer and Mathematics Education*, 9(1), 54-71.
- Otu, T. (2020). *Kodlama Ortamlarının Ortaokul Öğrencilerinin Başarı, Tutum ve Bilgi İşlemsel Düşünme Becerisine Etkisi, Yüksek Lisans Tezi*, Bolu Abant İzzet Baysal Üniversitesi.
- Öndeş, Ö. (2016). *İngiltere ve ABD’de kodlama eğitimi*. <http://www.hurriyet.com.tr/ingiltere-ve-abd-de-kodlama-egitimi-40061515>, Erişim Tarihi: 04.11.2020.

- Özçakmak, Ş. (2014). *Bilgisayar kullanımı çocuklukta bağımlılık yapar mı?* <https://www.haberturk.com/polemik/haber/973204-bilgisayar-kullanimi-cocukta-bagimlilik-yapar-mi>, Erişim Tarihi: 04.11.2020.
- Özçınar, H., Yecan, E., & Tanyeri, T. (2016). *Öğretmen gözüyle görsel programlama öğretimi*. *Proceeding book*, 71-79.
- Özdemir, A. (2015). *Kodlama, okuma yazma bilmek kadar önemli!* <http://www.sozcu.com.tr/egitim/kodlama-okuma-yazma-bilmek-kadar-onemli.html>, Erişim Tarihi: 04.11.2020
- Özdemir, M. (2010). Nitel Veri Analizi: Sosyal Bilimlerde Yöntembilim Sorunsalı Üzerine Bir Çalışma. *Eskişehir Osmangazi Üniversitesi Sosyal Bilimler Dergisi*, 11(1), 323-343.
- Özyol, B. (2019). *Bilgi-işlemsel düşünme becerisinin kazandırılmasına yönelik bir ortam tasarımı ve geliştirilmesi- Design and development of an online application for acquisition of computational thinking skills*. Yüksek Lisans Tezi, Afyon Kocatepe Üniversitesi, Fen Bilimleri Enstitüsü, Afyonkarahisar.
- Pakman, N. (2018) *8-10 Yaş Grubu Öğrencilerine Uygulanan Temel Düzey Kodlama, Robotik, 3d Tasarım ve Oyun Tasarımı Eğitiminin Problem Çözme ve Yansıtıcı Düşünme Becerilerine Etkisi*, Yayınlanmış Yüksek Lisans Tezi, Bahçeşehir Üniversitesi, Eğitim Bilimleri Enstitüsü, İstanbul.
- Park, S. Y., Song K. S., & Kim, S. H., (2015). Cognitive Load Changes in Pre-Service Teachers with Computational Thinking Education. *International Journal of Software Engineering and Its Applications*, 9, 169-178
- Patan, B. (2016). *Okul Öncesi Kodlama Öğretim Programının Geliştirilmesi*. Yüksek Lisans Tezi, Bahçeşehir Üniversitesi, Fen Bilimleri Enstitüsü İstanbul.
- Portelance, D.J. (2015). *Code and tell: An exploration of peer interviews and computational thinking with ScratchJr in the early childhood classroom*. Yayınlanmamış yüksek lisans tezi, Tufts Üniversitesi, Massachusetts.
- Programlar (2010). *Microsoft Bir Sonraki Neslin Oyun Yapımcılarını Yetiştiriyor: KODU Game Lab*. <http://www.programlar.com/makale/microsoft-bir-sonraki-neslin-oyun-yapimcilarini-yetistiriyor-kodu-game-lab.3a.5.1.html>, Erişim Tarihi: 26.03.2020
- Ramazanoğlu, M. (2021). Robotik Kodlama Uygulamalarının Ortaokul Öğrencilerinin Bilgisayara Yönelik Tutumlarına ve Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlilik Algılarına Etkisi. *Türkiye Sosyal Araştırmalar Dergisi*, 25 (1), 163-174.

- Resnick, M. (2013). *Learn to Code, Code to Learn*. <https://web.media.mit.edu/~mres/papers/L2CC2L-handout.pdf>, Eriřim Tarihi: 15.02.2020.
- Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B., & Kafai, Y., (2009). Scratch: Programming for all. *Communications of the ACM*, 52(11), 60-67, Eriřim Adresi: <https://web.media.mit.edu/~mres/papers/Scratch-CACM-final.pdf>
- Rose, S. P., Habgood, M. P. J., & Jay, T. (2017). An exploration of the role of visual programming tools in the development of young children's computational thinking. *Electronic Journal of e-Learning*, 15(4), 297–309.
- Sanz, A., (2015). *Why Teaching and Learning How to Code in Schools*. *EdTech Review*, <http://edtechreview.in/trends-insights/insights/1934-whyteaching-and-learning-how-to-code-in-schools> Eriřim Tarihi: 14.12.2020.
- Saygılı Yıldırım, T. (2020). *Robotik Kodlama Öğretiminde Probleme Dayalı Öğrenme Yaklaşımının Başarı, Pozitif Duygu ve Bilgi İşlemsel Düşünmeye Etkisi*, Yüksek Lisans Tezi, Onsekiz Mart Üniversitesi, Eğitim Bilimleri Enstitüsü, Çanakkale.
- Saygıner, Ş. (2017). *Blok Tabanlı Görsel ve Metin Tabanlı Programlama Öğretimlerinin Eriři, Mantıksal Düşünme ve Motivasyona Etkileri*, Yüksek Lisans Tezi, Hacettepe Üniversitesi, Eğitim Bilimleri Enstitüsü, Ankara
- Saygıner, Ş. ve Tüzün, H. (2017). *İlköğretim düzeyinde programlama eğitimi: yurt dışı ve yurt içi perspektifinden bir bakış*. 19. Akademik Biliřim Konferansı, 8-10 Şubat, Aksaray.
- Sayın, Z. ve Seferođlu, S. S. (2016). Yeni bir 21. yüzyıl becerisi olarak kodlama eğitimi ve kodlamanın eğitim politikalarına etkisi. *Aydın: XVIII. Akademik Biliřim Konferansı 'nda Sunulmuş Bildiri*, 1-5.
- Scharf, F., Winkler, T., & Herczeg, M. (2008). “Tangicons: algorithmic reasoning in a collaborative game for children in kindergarten and first class”, *In Proceedings of the 7th international conference on Interaction design and children*, 242-249.
- Scot, A. L. (2018). *The 21st Century skills early learning framework*. <http://www.battelleforkids.org/networks/p21>, Eriřim Tarihi: 03.01.2021
- Scratch (2020). *Scratch about*. <https://scratch.mit.edu/about>, Eriřim Tarihi: 25.03.2020.
- Shailaja, J. & Sridaran, R. (2015). Computational Thinking the Intellectual Thinking for the 21st Century. *International Journal of Advanced Networking Applications (IJANA)*, 39-46.

- Shin, S., Park, P., & Bae, Y. (2013). The Effects of an Information- Technology Gifted Program on Friendship Using Scratch Programming Language and Clutter, *International Journal of Computer and Communication Engineering*, 2(3), 246-249.
- SSRVM (2007). *Model curriculum and teaching material for K-12 Indian schools*. <https://www.it.iitb.ac.in/~sri/papers/SSRVM-CS-March07.pdf>, Erişim Tarihi: 04.11.2020
- Şahin, E. (2019). *6-12 Yaş Gruplarında Robotik Araç ve Gereçleri Kullanarak Kodlama Öğretiminin Uygulanması ve Analizi*. Yüksek Lisans Tezi, Marmara Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- Şahiner, A. & Kert, S. B. (2016). Komputasyonel Düşünme Kavramı ile ilgili 2006-2015 Yılları Arasındaki Çalışmaların İncelenmesi, *European Journal of Science and Technology*, 5(9), 38-43.
- Şenol, Ş. (2019). İlkokulda kodlama eğitimi: Sınıf öğretmenleri örneği, Yüksek lisans tezi, Süleyman Demirel Üniversitesi, Isparta.
- Şimşek, A., 2014, *Öğretim Tasarımı*, Nobel Yayıncılık, Ankara.
- Şimşek, E. (2018). *Programlama Öğretiminde Robotik ve Scratch Uygulamalarının Öğrencilerin Bilgi İşlemsel Düşünme Becerileri ve Akademik Başarılarına Etkisi*, Yüksek Lisans Tezi, Ondokuz Mayıs Üniversitesi, Eğitim Bilimleri Enstitüsü, Samsun.
- Somuncu, B. (2021). *Okul Öncesi Eğitimi Anabilim Dalı Okul Öncesi Dönemdeki Çocukların Matematiksel Akıl Yürütme Becerilerine Kodlama Eğitiminin Etkisi*, Yüksek Lisans Tezi, Çukurova Üniversitesi, Sosyal Bilimler Enstitüsü, Adana.
- Tağci, Ç. (2019). *Kodlama Eğitiminin İlkokul Öğrencileri Üzerindeki Etkisinin İncelenmesi*, Yüksek Lisans Tezi, Afyon Kocatepe Üniversitesi, Fen Bilimleri Enstitüsü, Afyon.
- Taş, N. (2018). *Farklılaştırılmış Bilgisayar Destekli Matematik Etkinliklerinin Üstün Yeteneklilerin Bilgi İşlemsel Düşünme Öz yeterlikleri ve Matematiğe Yönelik Tutumlarına Etkisi*, Doktora Tezi, Atatürk Üniversitesi, Erzurum.
- Tavsancıl, E. & Aslan, E.A. (2001). *Sözel, Yazılı ve Diğer Materyaller için içerik Analizi ve Uygulama Örnekleri*, Ankara: Epsilon Yayınları.
- TBD (2014). *Programlama Çocuk Oyunağı*. http://egitimplatformu.aydin.edu.tr/gundem/haber_detay.asp?haberID=32, Erişim Tarihi: 01.11.2020.

- Tekerek, M. ve Altan, T. (2014). The effect of Scratch environment on student's achievement in teaching algorithm. *World Journal on Educational Technology*, 6(2), 132-138.
- TTKB (2013). *Ortaokul yeni ders programı*. <https://www.memurlar.net/common/news/documents/572077/ek-1-28052013-22.pdf>, Erişim Tarihi: 27.04.2020
- TTKB (2018). *Bilişim teknolojileri ve yazılım dersi öğretim programı. 2018*. <http://mufredat.meb.gov.tr/Dosyalar/2018124103559587-Bili%C5%9Fim%20Teknolojileri%20ve%20Yaz%C4%B1%C4%B1m%205-6.%20S%C4%B1n%C4%B1flar.pdf>, Erişim Tarihi: 27.04.2020
- Tutulmaz, M. (2019). *Bilgi-İşlemsel Düşünme Becerisinin Geliştirilmesine Yönelik Veri Görselleştirmenin Tasarlanması, Uygulanması ve Değerlendirilmesi*, Yüksek Lisans Tezi, Hacettepe Üniversitesi, Eğitim Bilimleri Enstitüsü, Ankara.
- Uslu, N.A., Mumcu, F. ve Eğin, F. (2018). Görsel programlama etkinliklerinin ortaokul öğrencilerinin bilgi-işlemsel düşünme becerilerine etkisi. *Ege Eğitim Teknolojileri Dergisi*, 2(1), 19-31.
- Uzgun, B. Ç., ve Aykaç, N. (2016). “Bilişim Teknolojileri ve Yazılım Dersi Öğretim Programının Öğretmen Görüşlerine Göre Değerlendirilmesi (Ege Bölgesi Örneği)”, The Evaluation of Information Technologies and Software Course's Curriculum According to the Teacher's Ideas, *Mustafa Kemal Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, 13(34), 273-297.
- Uzunboylar, O. (2017). *Ortaokul Düzeyinde Kodlama Öğretimine İlişkin Öğretmen ve Öğrenci Görüşleri*, Yüksek Lisans Tezi, Ege Üniversitesi Fen Bilimleri Enstitüsü, İzmir.
- Ülkar, E. (2016). *10 soruda kodlama eğitimi*. <http://www.hurriyet.com.tr/10-soruda-kodlama-egitimi-40076661>, Erişim Tarihi: 28.06.2020.
- Ünsal, K. (2019). *Ortaokul ve lise okul yöneticilerinin kodlama eğitimine yönelik görüşlerinin incelenmesi (Bağcılar ilçesi örneği)*, Yüksek lisans tezi, Marmara Üniversitesi- İstanbul Sabahattin Zaim Üniversitesi, İstanbul.
- Ünsal, İ. (2020). Blok Tabanlı Programlama Etkinliklerinin İlkokul 2.Sınıf Öğrencilerinin Bilgi İşlemsel Düşünme Becerisi ve Etkinlik Algısı Üzerine Etkisi, Yüksek Lisans Tezi, Sakarya Üniversitesi, Eğitim Bilimleri Enstitüsü, Sakarya.
- Üzümcü, Ö. & Bay, E. (2018). Eğitimde Yeni 21. Yüzyıl Becerisi: Bilgi İşlemsel Düşünme. *Uluslararası Türk Kültür Coğrafyasında Sosyal Bilimler Dergisi (TURKSOSBİLDER)*, 3(2), 1-16.

- Voskoglou, M. Gr. & Buckley, S. (2012). Problem Solving and Computers in a Learning Environment, *Egyptian Computer Science Journal*, 36 (4), 28-45.
- Weinberg, A. E. (2013). *Computational thinking: An investigation of the existing scholarship and research*. (Unpublished Doctoral Thesis), Colorado State University, School of Education, Colorado.
- Weintrop, D. ve Wilensky, U. (2015). *To block or not to block, that is the question: students' perceptions of blocks-based programming*. In Proceedings of the 14th International Conference on Interaction Design and Children- IDC (pp. 199–208), Newyork.
- Wikipedia (2019). *Tynkr*. <https://en.wikipedia.org/wiki/Tynker>, Erişim Tarihi: 26.03.2020
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-35.
- Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society*, 366, 3717-3725.
- Wyffels, F., Martens, B. ve Lemmens, S. (2014). Starting from Scratch: Experimenting with Computer Science in Flemish Secondary Education. *9 th Workshop in Primary and Secondary Computing Education*, 12-15.
- Yadav, A., Hong, H., & Stephenson, C. (2016). Computational thinking for All: Pedagogical approaches to embedding 21st century problem solving in K-12 classrooms. *TechTrends*. 60(6), 565–568.
- Yağcı, M. (2018). A Study on Computational Thinking and High School Students' Computational Thinking Skill Levels, *International Online Journal of Educational Sciences*, 10(2), 81-96.
- Yecan, E. Özçınar, H. ve Tanyeri, T. (2017). Bilişim Teknolojileri Öğretmenlerinin Görsel Programlama Öğretimi Deneyimleri. *İlköğretim Online*, 16(1), 377-393.
- Yıldırım, A. ve Şimşek, H. (2013). *Sosyal Bilimlerde Nitel Araştırma Yöntemleri*. Ankara: Seçkin Yayıncılık.
- Yılmaz İnce, E. (2020). Kodlama Eğitiminde Oyun Geliştirme Platformlarının Kullanımı. *Mehmet Akif Ersoy Üniversitesi Eğitim Fakültesi Dergisi*, (53), 1-17.
- Yolcu, V. (2016). *Programlama Eğitiminde Robotik Kullanımının Akademik Başarı, Bilgi İşlemsel Düşünme Becerisi ve Öğrenme Transferine Etkisi*, Doktora Tezi, Süleyman Demirel Üniversitesi, Isparta.

- Yükseltürk, E. ve Altıok, S. (2016). Bilişim teknolojileri öğretmen adaylarının programlama öğretiminde scratch aracının kullanımına ilişkin algıları. *Mersin Üniversitesi Eğitim Fakültesi Dergisi*, 12(1), 39–52.
- Yükseltürk, E., Altıok, S. ve Üçgöl, M. (2016). Oyun Programlamanın İlköğretim Öğrencilerinin Problem Çözme Becerilerine Etkileri: Bir Yaz Kampı Deneyimleri 2. *4th International Instructional Technologies & Teacher Education Symposium*, 06-08 Ekim 2016.
- Yünkül, E., Durak, G. & Çankaya S. (2018) *Blok Tabanlı Yazılımların Kodlama Öğretiminde Kullanımı*. Uluslararası Necatibey Eğitim ve Sosyal Bilimler Araştırmaları Kongresi (UNESAK), 26-28 Ekim 2018, (533-547) Balıkesir.
- Yüzak, Ö. (2016). *Kodlama yapacak çocuk yetiştirecek*. http://www.cumhuriyet.com.tr/haber/ekonomi/529954/Kodlama_yapacak_cocuk_yetistirecek.html, Erişim Tarihi: 07.11.2020
- Zuckerman, O., Blau, I., & Monroy-Hernández, A. (2009). Children's participation patterns in online communities: An analysis of Israeli learners in the Scratch online community. *Interdisciplinary Journal of E-Learning and Learning Objects*, 5(1), 263-274.