

# 12. B¼L¼M

## ERKEN YAřTA PROGRAMLAMA EĐİTİMİ: ARAřTIRMALARDAKİ G¼NCEL EĐİLİMLERLE İLGİLİ BİR İNCELEME

*Yrd. Doç. Dr. Hatice DURAK  
Bartın niversitesi*

*Yrd. Doç. Dr. Fatma Gizem KARAOĐLAN YILMAZ  
Bartın niversitesi*

*Yrd. Doç. Dr. Ramazan YILMAZ  
Bartın niversitesi*

*Prof. Dr. S¼leyman Sadi SEFEROĐLU  
Hacettepe niversitesi*

### zet

G¼n¼m¼zde erken yařta programlama đretimi alıřmaları ok sayıda lkede pop¼lerlik kazanmıř durumdadır. Programlama hem bir 21. y¼zyıl yeterliliđi olarak hem de eđitimde 21. y¼zyıl yeterliliklerinin kazandırılması iin nemli g¼r¼lmeye bařlanmıřtır. Bu alıřmada erken yařta programlama eđitimi zerine yapılan alıřmalardaki eđilimlerin incelenmesi amalanmaktadır. Bu amala; alanla ilgili olan ve SSCI'de dizinlenen dergilerde, ulusal ve uluslararası tez veri tabanlarında “coding+kids; coding in primary education; programming in elementary school; programming for K12/ at K12 schools/ for K12 education/in the K12 classroom” anahtar kelimelerini ieren 6 makale ve 19 tez amalı rnekleme y¼ntemi ile seilerek analiz edilmiřtir. Bu kapsamda belirlenen makalelerin/tezlerin analizi iin arařtırmacılar tarafından hazırlanan bir “Sınıflama Formu” kullanılmıřtır. Makaleler ve tezler, betimsel tarama yapılarak, makalenin/tezin konusu, y¼ntemi, veri

toplama araçları, örnekleme, veri analiz yöntemleri, tartışma ve öneriler bölümlerini kapsayacak şekilde ayrıntılı olarak incelenmiştir. Yapılan taramalar sonucu elde edilen veriler yüzde ve frekansa dayalı bir şekilde tablo ve grafiklerle sunulmuştur. Elde edilen verilere dayalı olarak erken yaşta programlama eğitimiyle ilgili çalışmalarda en çok nitel yöntemin kullanıldığı, örneklemin ortaokul düzeyinden seçildiği, çok geniş kitleleri kapsayacak şekilde uygulama yapılmadığı, en çok incelenen değişkenlerin akademik başarı ve düşünme becerileri olduğu sonuçlarına ulaşılmıştır. Bu bölümde sunulan verilerin ve sonuçların erken yaşta programlama eğitimiyle ilgili yapılanları ve mevcut araştırma boşluğunu görmek adına yol gösterici nitelikte olacağı düşünülmektedir.

**Anahtar Sözcükler:** erken yaş, çocuklar, programlama eğitimi, K12, betimsel tarama.

### **Hazırlık Soruları**

1. Programlama eğitiminin öğrenme kuramları açısından eğitimdeki yeri nedir?
2. Programlama öğretimi öğrencilere hangi becerileri kazandırmaktadır?
3. Programlama öğretiminde kullanılan araç ve ortamların (Scratch, Alice, Small Basic...) önemi nedir?
4. Türkiye’de ve çeşitli diğer ülkelerde programlama eğitiminin durumu nedir? Karşılaştırınız.
5. Programlama ile FeTeMM (Özgün adı: STEM, “Science, Technology, Engineering, Mathematics”) arasında nasıl bir ilişkinin varlığı söz konusu olabilir?

## Giriş

Programlama eğitiminin hedef kitlesi olarak genellikle yükseköğretimde okuyan öğrenciler düşünülmektedir. Fakat eğitimde 21. yüzyıl yeterliliklerinin kazandırılması çalışmalarında, programlama eğitimi ortaokul, hatta ilkokul ve ana sınıfı öğrencileri için önemli görülmeye başlanmıştır (Fessakis, Gouli, & Mavroudi, 2013; Kalelioğlu & Gülbahar, 2014). Programlama eğitimi özellikle; yaratıcılık, eleştirel düşünme ve problem çözme, iletişim ve işbirliği, sosyal ve kültürlerarası beceriler, üretkenlik, liderlik ve sorumluluk gibi 21. yüzyıl becerileri olarak adlandırılan becerilerin kazandırılmasında işlevsel görülmektedir (Einhorn, 2011; Grover, & Pea, 2013; Lau, & Yuen, 2011; Yen, Wu, & Lin, 2012). Çünkü programlama süreçleri, farklı düşünme becerileri ve bilgi alanlarını içermesi nedeniyle programlama sürecinin doğasında yer alan; iletişim becerileri, yaratıcılık, entelektüel merak, eleştirel ve sistematik düşünme, kişilerarası ve işbirliği becerileri, problemi tanımlama, formüle etme ve çözme, öz-yönelim (Partnership for 21<sup>st</sup> Century Skills, 2007) gibi becerilerin çocuklarda geliştirilmesini sağlamaktadır (İsmal, Ngah, & Umar, 2010; Lau, & Yuen, 2011). Öte yandan programlama becerisinin 21. yüzyılın temel becerilerinden biri haline geldiği de söylenebilir. Nitekim programlama sürecinde çocuklar; problemlere farklı yönleriyle bakıp çözümler üretebilmeyi, sistematik, üretken, yaratıcı düşünebilmeyi ve en makul çözümü işbirliği yaparak önerebilmeyi, kısaca mantıksal akıl yürütmenin yollarını keşfederler. Dolayısıyla programlama becerisinin içerdiği ve öğrenene kazandırdığı beceriler göz önüne alındığında bu becerinin 21. yüzyıl becerilerinden biri haline dönüşmüş olduğu görülmektedir.

Bu görüşlerden hareketle bu çalışmada öncelikle programlama süreci, programlamanın eğitimde önemi ve programlama çevrelerinin eğitsel bağlamda değerlendirilmeleriyle ilgili kuramsal çerçeve sunulacaktır. İkinci bölümde SSCI tarafından dizinlenen dergilerde ve ulusal ve uluslararası tez veri tabanlarında yayımlanan erken yaşta programlama eğitimi üzerine yapılan çalışmalarda eğilimler incelenecektir. Yapılan incelemelerden hareketle araştırmanın sonuçları ve öneriler sunulacaktır.

## 1. Araştırmanın Amacı

Bu araştırmanın amacı, erken yaşta programlama eğitimi üzerine yapılan çalışmalarda eğilimlerin incelenmesidir. Bu amaçla; eğitim alanındaki SSCI tarafından taranan dergilerde yayımlanan makalelerden ve ulusal ve uluslararası tez veri tabanlarından “coding+kids; coding in primary education; programming in elementary school; programming for K12/ at K12 schools/ for K12 education/in

the K12 classroom ” anahtar kelimelerini içeren toplam 6 adet makale ve yurtiçi ile yurtdışında yapılan 19 tez ayrıntılı olarak analiz edilmiştir.

Bu araştırmanın “erken yaşta programlama eğitimi üzerine yapılan çalışmalardaki eğilimlerin incelenmesi” şeklinde belirlenen amacı doğrultusunda aşağıdaki sorulara yanıt aranmıştır:

Erken yaşta programlama eğitimi ile ilgili çalışmalarda;

- a. Hangi araştırma yöntemleri kullanılmıştır?
- b. Öğrenme süreç ve sonuçları açısından hangi değişkenler incelenmiş ve hangi sonuçlar elde edilmiştir?
- c. Çalışmaların hedef kitlesini oluşturan öğrencilerin demografik özellikleri nelerdir?
- d. Hangi veri toplama araçları kullanılmıştır?
- e. incelenen çalışmalarda erken yaşta programlama eğitimi ile ilgili hangi sorunlar gözlenmiştir?
- f. Erken yaşta programlama eğitiminin etkili olması için incelenen çalışmalarda ne gibi önerilerde bulunulmuştur?
- g. Erken yaşta programlama eğitimi ile ilgili çalışmalarda yer verilen öğretim sürecinde;
- h. Hangi ortamlar/araçlar kullanılmaktadır?
- i. Hangi öğrenme alanları (Matematik/ Fen/ Bilgisayar vb..) tercih edilmektedir?

## 2. Araştırmanın Önemi

Bu çalışma; Türkiye’de erken yaşta programlama eğitimiyle ilgili çalışmalara yol gösterici olacağı, bu konudaki eğitsel araştırmalarının incelenip düzenlenmesi, söz konusu alandaki eğilimlerin belirlenmesi, bu alanda ileride çalışacak araştırmacılara ışık tutacağı düşünüldüğü için önem taşımaktadır.

Ayrıca çok eski bir konu olmayan çocuklara programlama eğitimi konusunda birbirinden bağımsız çalışmaların yapılması kadar sınıflanmalarının, eğilimlerinin ve araştırmaların sonuçlarına dayalı değerlendirmelerin yapılmasının alana ve araştırmacılara yapacağı katkının önemli olduğu düşünülmektedir. Bu noktada bu çalışmanın sonuçlarının araştırmacılara ve uygulayıcılara önemli ipuçları sağlayacağı söylenebilir.

### 3. Erken Yaşta Programlama Öğretiminin Önemi

Dijital yerli olarak adlandırılan günümüz öğrenenlerinin eski kuşaklardan farklı olarak çeşitli yeni yeterliklere sahip olmaları beklenmektedir. Bu yeterliklerden birisi programlama eğitimidir (Burke, 2012; Hill, 2015; Kazakoff, 2014). Programlama eğitiminin hedef kitlesi olarak çoğunlukla yükseköğretimde okuyan öğrenciler görülmektedir. Bununla birlikte eğitimde 21. yüzyıl bilgi ve iletişim teknolojileri (BİT) yeterliklerinin kazandırılması için çeşitli uygulamalarla birlikte, programlama eğitimi ortaokul, hatta ilkokul ve ana sınıfı öğrencileri için önemli görülmeye başlanmıştır (Fessakis, Gouli & Mavroudi, 2013; Kalelioğlu & Gülbahar, 2014). Özellikle son dönemlerde ABD’de “Hour of Code” (Microsoft, 2014) ve Avrupada “Code Week EU” gibi etkinliklere politik açıdan gösterilen önem ve çocukların yoğun ilgisi erken yaşta programlama eğitiminin önemli görüldüğünün göstergesi sayılabilir (Durak, 2015).

Araştırmalar öğrenenlerin üst düzey düşünme becerilerinin geliştirilmesini sağlamak için erken yaşta programlama eğitiminin verilmesi gerektiğini göstermektedir (Computer Science Teachers Association [CSTA], 2010; DiSessa, 2001; Fessakis, Gouli & Mavroudi, 2013; Lau & Yuen, 2011; Lee, 2011). Erken yaşta programlama eğitimi verilmesinin neden önemli olduğu konusu alanyazında genel olarak pedagojik (CSTA, 2010; DiSessa, 2001; Fessakis, Gouli & Mavroudi, 2013; Lau & Yuen, 2011; Lee, 2011) ve ekonomik boyutta (CSTA, 2010) ele alınmıştır. Pedagojik açıdan erken yaşta programlama eğitiminin önemine, yazılım geliştirme süreçlerinin çeşitli düşünme becerileri ve öğrenme yetkinliği üzerindeki etkilerine odaklanılmıştır (DiSessa, 2001). Yazılım geliştirme süreçleri çocuklarda algoritmik düşünme, problem çözme becerilerinin gelişiminin yanı sıra üst düzey düşünme becerilerinin gelişimi için önemli kabul edilmektedir (Fessakis, Gouli & Mavroudi, 2013).

Erken yaşta programlama eğitiminin önemi ekonomik boyutta değerlendirildiğinde ise, 21. yüzyılda toplumların gelişimi için teknoloji alanında gelişimin önemli olduğu söylenebilir. Teknoloji biliminin temeli sayılan bilgisayar programlamada yaşanan gelişimler çeşitli ekonomik sektörlerin de gelişimini sağlayacaktır (CSTA, 2010). Erken yaşta bilgisayar programlama eğitimi ise çocukların üretim yapabileceklerinin farkına varmalarına ve 21. yüzyıl toplumlarının gelişiminde etkin rol alabilmelerine olanak tanıyacaktır.

Erken yaşta programlama öğretiminde ortaya çıkan en büyük engellerden biri olarak, geleneksel programlama dillerinin karmaşık yapılarından ötürü öğrencilerin programlama öğrenimine yönelik olumsuz duygu ve düşüncelere sahip olabilmeleri gösterilmektedir (Hill, 2015). Bununla birlikte günümüzde çocuklara

programlama öğretiminde kullanılmaya başlanan blok tabanlı programlama ortamlarının çocukların derleme (compiling) veya sözdizimi hatalarını (syntax errors) önemsemeden karmaşık görsel programlar oluşturabilmesini sağlamaktadır. Böylece blok tabanlı programlama ortamları ile öğrencilerin programlama eğitimine yönelik duygu ve düşüncelerinin olumlu yönde gelişeceği düşünülmektedir. Örneğin blok tabanlı programlama ortamlarından olan Scratch kullanımının çocukların programlamayı öğrenmelerine ve programlama eğitimine katılımlarına olumlu yönde etki ettiği sonucuna ulaşılmıştır (Zuckerman, Blau & Monroy-Hernández, 2009).

### **3.1. Erken Yaşta Programlama Öğretimiyle İlgili Araştırmalar**

Alanyazın incelendiğinde erken yaşta programlama öğretimine ilişkin çeşitli araştırmaların gerçekleştirilmiş olduğu görülmektedir (Akçay, 2009; Burke & Kafai, 2010; Çetin, 2012; Gülbahar & Kalelioğlu, 2014; Kazakoff, 2015; Maloney, Pepler, Kafai, Resnick & Rusk, 2008; Portelance, 2015; Shin, Park & Bae, 2013). Bu araştırmaların sonuçları aşağıda şekilde özetlenebilir.

Maloney, Pepler, Kafai, Resnick ve Rusk (2008) tarafından yapılan çalışmada Scratch ile 8-18 yaş arasındaki çocuklara programlama öğretmek amaçlanmıştır. Okul sonrası düzenlenen etkinlikler sonucunda çocukların programlamaya ilişkin anahtar kavramları öğrendiği görülmüştür. Programlamayı tanıtmak ve öğretmek için Scratch'ın uygun bir program olduğu sonucuna ulaşılmıştır. Ayrıca Scratch'ın çocukların programlama öğrenimine yönelik güdülenmelerini arttırdığı da çalışmada ulaşılan bir diğer sonuçtur.

Akçay (2009) tarafından gerçekleştirilen çalışmada Small Basic programının ilköğretim düzeyinde bilgisayar derslerine kaynaştırılması (entegrasyonu) irdelenmiştir. Çalışmada; öğrenci ve öğretmenlerin programlamaya yönelik algıları, programlamanın öğrencilerin güdülenmeleri üzerindeki etkisi, Small Basic ortamının avantajları, dezavantajları ve kullanım kolaylığı gibi durumlar araştırılmıştır. Çalışma sonucunda Small Basic programının tüm öğrenci ve öğretmenlerce kabul gördüğü sonucuna ulaşılmıştır. Öğrenciler Small Basic kullanımının kendi güdülenmelerini olumlu yönde etkilediğini belirtmişlerdir. Öğrenci ve öğretmenler ayrıca Small Basic'in faydalı, kolay kullanılabilir ve avantajlı olduğunu ifade etmişlerdir.

Çetin (2012) tarafından yapılan çalışmada çocuklar için bilgisayar programlama eğitiminin problem çözme becerileri üzerine etkisi incelenmiştir. Bu çalışmada ayrıca öğrencilerin ve velilerin konuyla ilgili görüşleri de incelenmiştir. Çalışma sonucunda çocuklar için programlama eğitiminin çocukların problem

çözme becerilerine olumlu katkı sağladığı ortaya çıkmıştır. Veliler sunulan eğitimin, çocuklarının akademik bilgi ve becerilerine katkı sağladığını ve ayrıca çocuklarının bilgisayar kullanımına da olumlu etkileri olduğunu belirtmişlerdir. Ayrıca veliler bu tür eğitimlerin çocukları için ihtiyaç olarak gördüklerini söylemişlerdir.

Shin, Park ve Bae (2013) tarafından yapılan çalışmada üstün yetenekli öğrencilerin sınıflarında araştırmacılarca geliştirilen Clutter web sitesi aracılığıyla Scratch programlama ortamının kullanılmasının işbirlikli çalışma ve arkadaşlık üzerine etkisi araştırılmıştır. Araştırma sonucunda Scratch programının sınırlı işbirliğine izin verdiği, bireysel kullanılan bir ortam olduğu ve grup olarak eşzamanlı bir ürün geliştirmeye uygun olmadığı sonucuna ulaşılmıştır.

Gülbahar ve Kalelioğlu (2014) tarafından gerçekleştirilen çalışmada ilköğretim 5. sınıf öğrencilerinin problem çözme becerileri üzerinde Scratch ile programlamanın etkisi araştırılmıştır. Araştırma sonucunda Scratch platformunda programlama öğretiminin öğrencilerin problem çözme becerileri üzerinde anlamlı bir farklılığa yol açmadığı sonucuna ulaşılmıştır. Diğer taraftan öğrencilerin büyük bir çoğunluğu Scratch platformunun kullanımını kolay bulduklarını belirtmişlerdir.

Kazakoff (2014) tarafından yapılan çalışmada anaokulu öğrencilerinin ScratchJr öğrenme süreci ile öz-düzenleme becerileri arasındaki ilişki incelenmiştir. Araştırma sonucunda ScratchJr öğrenimi ile öz-düzenleme becerileri arasında anlamlı bir ilişki olmadığı sonucuna ulaşılmıştır.

Kazakoff (2015) tarafından yapılan çalışmada erken yaşta robot programlama öğrenmenin, çocukların teknoloji okuryazarlık becerileri üzerine etkisine odaklanılmıştır. Anasınıfı öğrencilerine teknoloji okuryazarlık becerilerinin yanı sıra eleştirel düşünme ve problem çözme becerilerini kazandırmayı amaçlayan bu çalışmada teknolojik araçlar kullanılarak yapılandırmacı yaklaşımın öğrenme ilkeleri çerçevesinde öğrenme ortamı tasarlanmıştır. Araştırma sonucunda öğrenenlerin hem dijital araçları kullanmayı hem de kazandırılması amaçlanan becerileri kazandırmada robot programlama öğretiminin etkili olduğu sonucuna varılmıştır.

Portelance (2015) tarafından yapılan çalışmada erken çocukluk döneminde ScratchJr ile çocuklara programlama öğretiminin bilgi işlemsel düşünme becerilerine olan etkisi incelenmiştir. Araştırma sonucunda ScratchJr ile çocuklara programlama öğretiminin bilgi işlemsel düşünme becerilerini geliştirmede çeşitli fırsatlar sunduğu sonucuna ulaşılmıştır.

Programlama öğretiminde farklı üst düzey düşünme yöntemleri ve farklı stratejiler kullanılabileceği için bu süreçte etkili olan çeşitli faktörler olduğu söyle-

nebilir (Lamb & Johnson, 2011). Alanyazında 8 yaş ile başlayıp yetişkin yaşlara kadar uzanan bir çalışma grubu ölçeğinde, programlama öğretimi sürecinde incelenen faktörler; demografik özellikler (Lau & Yuen, 2011), yaş, cinsiyet (Kiss, 2010), kişilik, zekâ, başarı, özyeterlilik algısı, bilgisayara karşı tutum, deneyim, düşünme ve problem çözme yetenekleri gibi bireysel farklılıklar (Kiss, 2013; Kurland, Pea, Clement & Mawby, 1986; Lau & Yuen, 2011), 21. yüzyıl becerileri (Lamb & Johnson, 2011), yaratıcı ve bağımsız düşünme, işbirlikli çalışma, sistematik düşünme, eleştirel yansıtma ve matematik becerileri (Ersoy, 2005) şeklinde sıralanabilir.

#### **4. Programlama Öğretiminde Kullanılan Ortamlar/Araçlar**

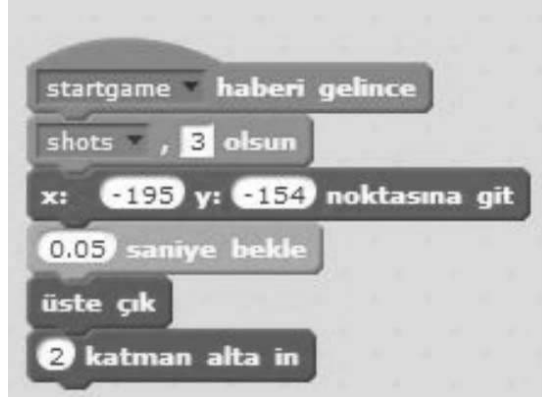
Çocuklar için programlama öğrenme becerilerini geliştirmeye yönelik eğitimler son birkaç yıl içerisinde büyük bir talep haline gelmiş durumdadır (Al-Jarrah, 2016). Bu nedenle günümüzde çocukların programlamayı anlayabilmeleri, programlama çalışmaları yapabilmeleri için çeşitli ortamlar ve araçlar kullanılmaya başlanmıştır. Genel bir bakış açısıyla değerlendirildiğinde bu ortam ve araçları, metin tabanlı (text-based) ve blok tabanlı (block-based) araç ve ortamlar olarak sınıflayabilmek mümkündür.

Metin tabanlı programlama dillerinde kodlama yaparken, programlama dilinin sözdizim kurallarına uygun olarak metin tabanlı karakterlerin kullanılması gerekir. Bazı metin tabanlı diller “C++, Java, ActionScript (Flash), JavaScript, Python ve Scheme” şeklinde listelenebilir. Diğer taraftan blok tabanlı programlama ortamları ise çoğunlukla grafiksel programlama ortamlarıdır. Bu tür ortamlarda kodlama işlemi, programlama dilinin sözdizim kurallarına göre bir kod yazılmasına gerek kalmadan, görsel nesnelerin ekran üzerinde sürükle-bırak mantığıyla yapılabilmektedir. Araştırmacılar blok tabanlı programlama ortamlarının sahip olduğu grafiksel arayüz ve genellikle kodlama yaparken sözdizim kurallarına uygun olarak kod yazımını gerektirmemesinden dolayı özellikle çocuklar için uygun olduğunu ifade etmektedir. Blok tabanlı programlama ortamlarının sahip olduğu bu özelliklerin aynı zamanda çocukların bilgisayar programlama konusunda oluşan bilişsel engellerin azaltılmasına da katkı sağladığı belirtilmektedir (Hill, 2015).

İlk blok tabanlı programlama ortamı 1986 yılında bir araştırma kapsamında oluşturulmuştur. Aradan geçen sürede blok tabanlı programlama ortamlarının sayısı artmıştır. Alice, Scratch, App Inventor, ToolBox ve Looking Glass gibi ortamlar son on yıl içerisinde oluşturulmuş olan blok tabanlı programlama ortamlarındandır. Alice ve Scratch bu alandaki en bilinen programlama dillerinden ikisidir (Medlock-Walton, Harms, Kraemer, Brennan & Wendel, 2014). Blok tabanlı programlama yaklaşımı, bileşen tabanlı programlama ve son kullanıcı prog-



ramlama paradigmasını birleştirmektedir (Mohamad, 2011; Almatary, Bakar & Zin, 2014). Blok tabanlı programlamanın temel amacı, uygulamada kullanımı kolay programlama yapabilmeyi sağlamaktır (Mohamad, 2011). Şekil 1'de Scratch programlama ortamı kullanılarak oluşturulmuş blok tabanlı programlamadan bir örnek verilmiştir.



Şekil 1. Scratch Programlama Ortamı Kullanılarak Oluşturulmuş Blok Tabanlı Programlamadan Bir Örnek

Blok tabanlı programlama ortamlarının tasarımı hedef kitlenin özelliklerine göre belirlenmektedir. Blok tabanlı programlama ortamları genel olarak iki ana amaç için oluşturulmaktadır (Medlock-Walton vd., 2014). Buradaki ilk amaç "Programlama sözdizimlerini basitleştirmek ve böylece temel programlama bilgisine sahip herkesin programlama yapabilmesini sağlamak"tır. İkinci amaç ise "Hedef kitlenin ilgisini çekerek ve programlama ile ilgilenen birey sayısını artırmak"tır. Çünkü programlamanın hedef kitlesi, yalnızca programlama bilgi ve becerisi belirli seviyede olan bireylerle sınırlı değildir (Al-Jarrah, 2016). Bu nedenle programlama bilgi ve becerisi düşük ya da hiç olmayan bireylerin programlama öğrenmelerinde blok tabanlı programlama ortamları yararlı araçlar olarak görülmektedir.

Kullanılan her programlama ortamının (App Inventor, Scratch, Alice, Looking Glass gibi) belirli hedef kitleler için ve belirli amaçlarla hazırlandıkları söylenebilir (Al-Jarrah, 2016; Medlock-Walton vd., 2014). Örneğin App Inventor, Massachusetts Institute of Technology (MIT) tarafından geliştirilen Android tabanlı bir platform olup, programlama deneyimleri belirli bir seviyede olan kullanıcıların uygulamalar geliştirebileceği bir ortam sunmaktadır. Benzer şekilde Scratch de MIT tarafından geliştirilen bir ortam olup, genellikle 8-16 yaş aralığındaki çocukların programlama deneyimi yaşayabileceği bir ortam sunmaktadır. Looking

Glass ise çocukların örnekler üzerinden programlamayı öğrenebileceği bir ortam olarak tasarlanmıştır (Al-Jarrah, 2016).

Alanyazında blok tabanlı programlama yaklaşımının bir takım avantajlarının olduğu belirtilmektedir (Mohamad, 2011; Almatary, Bakar & Zin, 2014; akt. Al-Jarrah, 2016). Bu avantajlar aşağıdaki şekilde özetlenebilir:

- Uygulama geliştirme maliyeti düşüktür.
- Blok tabanlı programlama ile uygulama geliştirme profesyonel beceriler gerektirmemektedir.
- Uygulama geliştirme hızlı bir şekilde yapılabilir.

Bütün bu açıklamalardan sonra çocuklara programlama öğretiminde yaygın bir şekilde tercih edilen bazı araç ve ortamlarla ilgili ayrıntılı bilgi sunulmasının yararlı olduğu düşünülmektedir. Bu bağlamda “Small Basic, Python, Scratch, Alice, App Inventor, Lego Mindstorms, Game Maker ve Kodu” isimli araç ve ortamlar incelenecektir (Bkz. Şekil 2).

#### 4.1. Small Basic

Small Basic, çocuklara bilgisayar programlama öğretmek amacıyla Microsoft tarafından geliştirilen bir programlama dilidir. Microsoft’un programlama öğrenmeye yeni başlayanlar için geliştirmiş olduğu ve temelde BASIC programlama dilinin basitleştirilmiş bir şeklidir. Blok tabanlı programlama ortamlarının aksine Small Basic ortamında program oluşturma programlama dilinin sözdizim kurallarına göre olmaktadır. Diğer taraftan Small Basic programının sahip olduğu; basitleştirilmiş arayüz ve editör, sözdizimini renklendirme, akıllı kod tamamlama gibi özellikler çocukların programlama sürecini kavramalarını kolaylaştıran özellikler olarak değerlendirilebilir. Small Basic’de ekrana “Merhaba Dünya!” yazdırabilmek için editöre yazılması gereken kod satırı `TextWindow.WriteLine(“Merhaba Dünya!”)` şeklindedir. Buradan da anlaşılacağı üzere çocuklar bu ortamda kodlama yaparken blok tabanlı ortamlardaki sürükle-bırak mantığı yerine Small Basic programlama dilinin sözdizim kurallarını takip etmek durumundadırlar.



Şekil 2. Çocuklara Programlama Öğretiminde Yaygın Bir Şekilde Tercih Edilen Bazı Araç ve Ortamlar

#### 4.2. Python

Python da Small Basic gibi metin tabanlı programlama dillerindedir. Python'un sahip olduğu küme parantezleri yerine girintilere dayalı basit sözdizimi, programlama öğrenmeye yeni başlayanlar için tercih edilen ortamlardandır. Bu aynı zamanda Python ile yazılmış programların diğer dillere göre kolaylıkla okunabilmesine de olanak sağlamaktadır. Python'un sahip olduğu önemli özelliklerden biri de "C, C++" gibi diğer metin tabanlı programlama dillerinin aksine derlemeye gereksinim duyulmadan yazılan programları çalıştırabiliyor olmasıdır. Bu özellik hızlı bir şekilde program geliştirmeye olanak tanımaktadır. Python çocuklara programlama öğretiminde kullanılan bir programlama dili olmakla beraber, Python kullanılarak web programlama, oyun programlama gibi gelişmiş programlar da yazılabilmektedir. Örneğin, Python'da ekrana "Merhaba Dünya!" yazdırabilmek için editöre yazılması gereken kod satırı `print("Merhaba Dünya!")` şeklindedir.

#### 4.3. Scratch

Scratch, küçük yaşta çocukların bilgisayar programlama becerilerini geliştirmek amacıyla kullanılan blok tabanlı programlama ortamlarındandır. Çocuklar bu ortamda etkileşimli projeler geliştirebilmektedirler. Scratch ortamı küçük çocuklar tarafından kullanılmak üzere özel olarak tasarlanmış bir programlama dili

ve etkileşimli animasyon yapma arayüzü içermektedir. Kullanıcılar Scratch ortamında nesnelere (örneğin, insanlar, hayvanlar, araçlar) hareket ettirebilmekte, görünümünü değiştirebilmekte, nesnelere ses ekleyebilmekte ve bu şekilde animasyonlu hikâyeler ve oyunlar geliştirebilmektedirler. Scratch ortamında kullanıcılar projelerinde birden çok sayfa açıp bu sayfalar arasında bağlantı kurabilme, kütüphanelerden karakterler ve arka plan ekleyebilme, boyama editörlerini kullanarak kendi karakterlerini ve arka planlarını oluşturabilme, sayfaya yazı ekleyebilme gibi programın özelliklerinden yararlanabilmektedirler. Bu arada Scratch özellikle 8 yaş ve üzeri çocuklar için geliştirilmiş bir programlama ortamı iken, ScratchJr'un özellikle 5-7 yaş dönemi küçük çocukların gelişimsel dönemlerine uygun olarak tasarlandığı ifade edilmektedir (Portelance, 2015).

#### **4.4. Alice**

Alice çocuklara programlama öğretiminde kullanılan üç boyutlu etkileşimli bir ortamdır. Alice başlangıçta Carnegie Mellon Üniversitesi'nde bir grup araştırmacı tarafından geliştirilmiştir (Al-Jarrah, 2016). Öğrenciler Alice'i kullanarak; üç boyutlu bir dünyada hikâye oluşturarak, etkileşimli bir oyun ya da üç boyutlu animasyon geliştirerek programlama öğrenebilirler. Alice de diğer blok tabanlı programlama ortamlarında olduğu gibi geleneksel programlama dillerinin sözdizimsel (syntactic) karmaşıklıklarından kaçınarak, öğrencilere yaratıcılık becerilerini geliştirmeleri için fırsatlar sunmaktadır. Alice Java tabanlı bir ortamdır. Alice'de geliştirilen programlar Java kodlarına dönüştürülebilmektedir. Alice programcılarının "Java ve C++" gibi geleneksel programlama dillerinde olduğu gibi kod yazmaları gerekmez. Alice ortamında, programlamaya yeni başlamış kullanıcılar bile ortamdaki hazır nesnelere (örneğin, insanlar, hayvanlar, araçlar) sürükle-bırak yaparak üç boyutlu ortam geliştirebilmektedirler.

#### **4.5. App Inventor**

App Inventor MIT ve Google tarafından bilgisayar programlamayı öğretmek amacıyla geliştirilen blok tabanlı bir programlama ortamıdır. Görsel nesnelere kullanarak sürükle bırak mantığı çerçevesinde programlama yapılmasına olanak tanıyan bu ortam, uygulama geliştirme sürecinde metin tabanlı programlama dillerinin karmaşık yapısını basitleştirmektedir. Karmaşık olmayan bu arayüz yapısı sayesinde çocuklara programlama öğretiminde kullanılacak ortamlardan biri olarak değerlendirilmektedir. Kullanıcılar App Inventor'i kullanarak Android tabanlı mobil uygulamalar geliştirebilmektedirler.

#### 4.6. Lego Mindstorms

Lego Mindstorms, MIT ve Lego tarafından geliştirilen, kullanıcıların robot programlama yapabilmelerine olanak tanıyan yazılım ve donanımdan oluşan bir teknolojidir. Lego Mindstorms sistemi kontrol etmek amacıyla kullanılan akıllı tuğla (intelligent brick), modüler sensörler ve motorlar ile lego parçalarının bir araya getirilmesi ile programlanabilir robotlar oluşturulmasına olanak sağlamaktadır. Lego Mindstorms'un bu donanımlarının programlanabilmesi için de blok tabanlı sürükle bırak mantığı ile çalışan bir program kullanılmaktadır.

#### 4.7. Game Maker

Game Maker, oyun programlamak amacıyla kullanılan Delphi temelli bir programlama dilidir. Game Maker ile basit oyunlar tasarlanabileceği gibi, üç boyutlu çok kullanıcılı gelişmiş oyun tasarımlarının yapılması da mümkündür. Programın kütüphanesinde bulunan hazır aksiyon efektlerini kullanarak yağmur efekti, duman efekti gibi efektlerin verebilmek mümkündür. Ayrıca kullanıcılar kendi efektlerini de geliştirebilmektedirler. Programın aksiyon kütüphanesi blok tabanlı programlama ortamlarında olduğu gibi sürükle bırak mantığı ile çalışmaktadır. Ayrıca programın sahip olduğu metin tabanlı programlama dili olan Game Maker Language ile kod satırları yazarak tasarlanan oyunları geliştirebilmek mümkündür. Game Maker'ın grafik düzenlemek ve ses eklemek için de program içerisinde editörleri bulunmaktadır.

#### 4.8. Kodu

Microsoft tarafından çocuklara programlama öğretmek amacıyla geliştirilen blok tabanlı programlama ortamlarından olan Kodu ile çocukların üç boyutlu oyun programlayabilmeleri mümkündür. Kodu ile tasarlanan oyunlar bilgisayar ve Xbox'ta çalıştırılabilmektedir. Kodu, basit bir program arayüzüne sahip olduğu için çocukların eğlenerek programlama sürecini kavramalarına yardımcı olabilmektedir. Kodu ile çocukların problem çözme, algoritma geliştirme, dijital hikâye hazırlama süreçlerini kavrayabilmeleri mümkündür. Yapısı Alice programına benzerdir.

## 5.Yöntem

### 5.1. Araştırmanın Modeli

Bu araştırma betimsel tarama modelinde bir araştırmadır. Betimsel tarama modeli ile gerçekleştirilen çalışmalar, ele alınan konu ile ilgili mevcut durumu var olan haliyle betimlemeye yöneliktir (Şimşek vd., 2009). Bu çalışmada erken yaşta programlama eğitimi üzerine yapılan çalışmalardaki eğilimler incelenerek var olan durum ortaya çıkarılmaya çalışıldığı için betimsel tarama modeli kullanılmıştır. Bu yöntem bağlamında yararlanılan veri toplama yöntemi belge incelemesidir. Bu veri toplama yöntemi, araştırılması planlanan olay veya durumlar hakkında yazılı bilgi içeren kaynakların ayrıntılı biçimde analiz edilmesine dayanır (Şimşek, & Yıldırım, 2011). Bu çalışma kapsamında Eğitim Bilimleri alanında SSCI tarafından taranan eğitim dergileri ile ulusal ve uluslararası tez veri tabanlarında “erken yaşta programlama eğitimi” üzerine yayınlanan 6 adet makale ve 19 tez, “*araştırma yöntemleri, ele alınan değişkenler, hedef kitle, örneklem seçme yöntemi, veri toplama araçları, veri analizi yöntemi, sonuçlar, ele alınan sorunlar ve öneriler, kullanılan ortamlar/araçlar ve öğrenme alanları*” açılarından incelenmiştir.

### 5.2. Veri Toplama Süreci

Veri toplama sürecinde ilk aşamada araştırmacılar tarafından “Erken Yaşta Programlama Eğitimi Yayınlarını Sınıflama Formu” isimli bir form geliştirilmiştir. Bu form geliştirilirken Göktaş vd. (2012) tarafından geliştirilen makale inceleme formundan yararlanılmış ve uzman görüşleri alındıktan sonra son hali verilmiştir. Bu süreçte çeşitli derleme çalışmalarındaki değerlendirme ölçütleri de incelenerek bir araştırma çerçevesi ortaya çıkarılmıştır (Bkz. Ek-1). Bu çalışma kapsamında ele alınan tüm çalışmalar yöntemsel, konunun doğası ve özellikleri açısından değerlendirilmiştir.

Bu çalışmada, erken yaşta programlama eğitimi konusunda gerçekleştirilen çeşitli araştırmalara ISI Web of Knowledge veritabanı üzerinden ulaşılmıştır. Bu kapsamda Social Sciences Citation Index (SSCI) atıf indekslerinde taranan 6 makale ve yurtiçi ile yurtdışında yapılan 19 tez ayrıntılı olarak incelenmiştir (Bkz. Ek-2). Taramalarda problem durumu için uygun görülen “*coding+kids; coding in primary education; programming in elementary school; programming for K12/ at K12 schools/ for K12 education/in the K12 classroom*” anahtar kelimeleri kullanılmıştır (Bkz. Tablo 1). Bu anahtar kelimelere alanyazında ele alınan başlıca proje, uygulama ve çalışmalarda öne çıkan değişkenlerin incelenmesi sonucunda karar verilmiştir.

Tablo 1. İlgili Çalışmalar İçin Taranan Kaynaklar ve Kullanılan Anahtar Kelimeler

Veri Tabanları, Arama Motorları ve Diğer Kaynaklar	Anahtar Kelimeler
SciVerse ScienceDirect	coding+kids
SciVerse Scopus	coding in primary education
JSTOR	programming in elementary school
ProQuest (thesis and dissertations)	programming for K12/ at K12 schools/ for
SAGE Publications	K12 education/in the K12 classroom
Taylor and Francis Online	
Wiley Online	
scholar.google.com	
tez2.yok.gov.tr	
ULAKBİM (ulusal veri tabanları)	

Araştırma, 2016 yılı Kasım ayı içerisinde yapılan alanyazın taraması sonucunda erişilen çalışmalarla sınırlıdır. Öte yandan çalışma konusunun son yıllarda popüler hale gelmesi ve nispeten yeni bir konu olması nedeniyle bu araştırmanın problem durumuna uygun biçimde belirlenen anahtar kelimeler yardımıyla, uluslararası veritabanlarında yer alan dergilerden ve tezlerin yer aldığı veri tabanlarından herhangi bir yıl kısıtlaması olmaksızın tarama yapılmıştır.

### 5.3. Verilerin Analizi

Makalelerin incelemesinde, uzman görüşleriyle şekillendirilen “Erken Yaşta Programlama Eğitimi Yayınlarını Sınıflama Formu” kullanılmıştır. Başlangıçta araştırmacılardan oluşan kodlamacılar arasında tutarlılığın sağlanması için kodlamalara birlikte başlanmıştır. Fikir ayrılığı oluşan çalışmalara tekrar geri dönmüştür. Bu çalışmalar üzerine tartışılarak fikir birliğine varılmış ve yeniden kodlama yoluna gidilmiştir. Daha sonra rastgele bölünen makaleleri kodlamacılar kodlamış ve güvenilirliği artırmak için başka bir kodlamacı tarafından da yapılan kodlamalar incelenmiştir. Yapılan kodlamalar makale inceleme formuna yansıtıldıktan sonra frekans ve yüzdeler hesaplanmıştır. Sonuçta elde edilen veriler tablo ve grafikler kullanılarak sunulmuştur.

## 6. Bulgular

Bu bölümde çalışmanın alt problemleri doğrultusunda elde edilen bulgular alt başlıklar halinde sunulmaktadır.

### 6.1. Erken Yaşta Programlama Eğitimi ile İlgili Çalışmalarda Yöntemsel Ayrıntılar ve Çalışmalarda Elde Edilen Sonuçlar

Araştırmanın ilk alt problemi “Erken yaşta programlama eğitimi ile ilgili çalışmalarda kullanılan araştırma yöntemleri, öğrenme süreç ve sonuçları açısından incelenen değişkenler, çalışmaların hedef kitlesinin özellikleri, örneklem seçme yöntemi, veri toplama araçları, veri analizi yöntemi, elde edilen sonuçlar, gözlenen sorunlar ve önerilerde nelerdir?” şeklinde belirlenmiştir. Bu alt probleme yanıt olarak bulgular alt başlıklar halinde sunulmuştur.

#### 6.1.1. Programlama Eğitimiyle İlgili Çalışmalarda Kullanılan Yöntemler

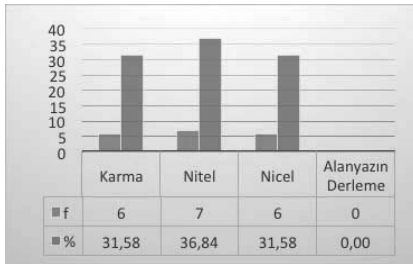
Erken yaşta programlama eğitimiyle ilgili ulusal ve uluslararası yapılan çalışmalar bir araya getirilerek yöntemsel açıdan eğilimleri incelenmiştir. İncelenen araştırmalardaki yöntemsel eğilimler Şekil 3’te genel olarak sunulmuş, Şekil 4’te tez ve makalelerde kullanılan yöntemler ayrı ayrı ele alınmıştır.



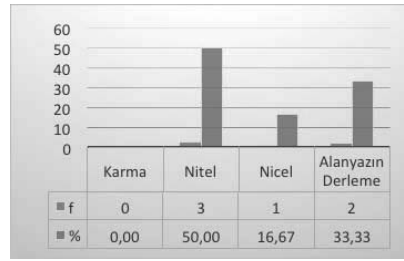
Şekil 3. İncelenen Çalışmalarda Kullanılan Yöntemlerin Dağılımı



Şekil 3'teki veriler incelendiğinde, erken yaşta programlama eğitimiyle ilgili çalışmaların çoğunda nitel araştırma yöntemlerinin tercih edildiği görülmektedir. Bu yöntem haricinde nicel ve karma yöntemlerin de sıklıkla kullanıldığı söylenebilir. Yöntemle ilgili dağılımlar incelenen tez ve makaleler açısından bazı farklılıklar göstermektedir. İncelenen tezlerde karma, nitel ve nicel yöntemler hemen hemen eşit dağılım gösterirken makalelerde nitel yöntemin ağır bastığı görülmektedir (Bkz. Şekil 4). Dehmenoğlu (2015), Doğan (2015), Patan (2016) ve Tilki (2016) bir tasarım geliştirerek öğrenen, öğretmen vb. tarafların görüşlerini almışlardır. Noble (2013), Bers, Flannery, Kazakoff ve Sullivan (2014) ile Sáez-López, Román-González ve Vázquez-Cano (2016) ise tasarım tabanlı araştırma yöntemi ile çalışmalarını gerçekleştirmiştir. Tasarım tabanlı çalışmalar bu çalışmada nitel yöntem altına alınmıştır. Nitel araştırma yöntemlerinden olan tasarım tabanlı araştırma yönteminde, bir öğretimi tasarlama ve uygulama sürecini geliştirmek amacıyla tasarımların olumlu ve olumsuz yönleri belirlenerek, yapılması gereken değişikliklerin ortaya konulması ve tasarımların düzenlenmesi gerekir (Merriam, 1998; Reigeluth & Frick, 1999). Öte yandan Benitti (2012) ile Lye ve Koh'un (2014) çalışmalarının alan yazın derleme çalışması olduğu belirlenmiştir.



**İncelenen Tezlerde Kullanılan Yöntemler**



**İncelenen Makalelerde Kullanılan Yöntemler**

Şekil 4. İncelenen Tezlerde ve Makalelerde Kullanılan Yöntemlerin Dağılımı

### 6.1.2. Programlama Eğitimiyle İlgili Çalışmalarda İncelenen Değişkenler

Bağımsız değişkenlerin bağımlı değişken üzerine etkisini belirlemeye dair deneysel yöntemle desenlenen çalışmaların incelenmesiyle elde edilen bulgular Tablo 2'de sunulmuştur.

Tablo 2. İncelenen Makalelerde Üzerinde Çalışılan Değişkenler

Ele Alınan Makaleler	Bağımlı Değişken	Bağımsız Değişken	Sonuç
Gülmez (2009)	Öğrenci başarı ve motivasyonu	Programlama öğretiminde görselleştirme araçları kullanma	Çalışma sonucunda döngü ve koşul kullanımı konusunda algoritmayı hikâyeleştiren araç lehine anlamlı fark gözlenmiştir. Çalışma grupları arasında motivasyon açısından anlamlı fark olmadığı belirlenmiştir. Öğrencilerin algoritma geliştirme başarıları ile Türkçe, matematik, İngilizce ve bilişim teknolojileri dersleri arasında anlamlı ilişki tespit edilmiştir.
Çetin (2012).	Problem çözme becerileri	Bilgisayar programlama eğitimi	Çocuklar için bilgisayar programlama eğitiminin problem çözme becerilerini geliştirmede uygulanabilir olduğu, programlama eğitiminin çocukların problem çözme becerilerine olumlu yönde katkı sağladığı belirlenmiştir.
Aslan (2014)	Problem çözmeye yönelik yansıtıcı düşünme, matematik (Olasılık) başarıları	Video geliştirmede Scratch kullanımı	Scratch çalışmasının öğrencilerin olasılık öğrenimi üzerinde istatistiksel olarak anlamlı bir etkisi olduğu bulunmuştur. Öte yandan, problem çözmeye yönelik yansıtıcı düşünme becerisi üzerinde kayda değer bir etki bulunamamıştır.
Olgun (2014)	Düşünme stilleri	Programlama öğretimi, cinsiyet	13 düşünme stili içerisinde özel düşünme, aşamalı düşünme, kuralsız düşünme, ayrıntıcı düşünme ve dışa dönük düşünme stillerinde programlama eğitiminin bir etkisi olduğu bulunmuştur.

Dehmenoğlu (2015)	Akademik başarı	Mobil destekli programlama öğretimi	Mobil öğrenmenin öğrencilerin akademik başarıları üzerinde istatistiksel olarak anlamlı bir etkisinin olduğu, mobil öğrenme ile desteklenen öğrencilerin sadece yüz yüze eğitim alan öğrencilerden daha yüksek başarı sağladığı görülmüştür.
Ceylan (2015)	Akademik başarı	Harmanlanmış öğrenme yöntemi	Programlama öğretiminin gerçekleştirildiği harmanlanmış öğrenme ortamının öğrencilerin akademik başarılarında olumlu bir fark oluşturduğu sonucuna ulaşılmıştır.
Doğan (2015)	Akademik başarı, eleştirel düşünme becerileri	Programlama eğitimi	Eleştirel düşünme becerileri ve algoritma başarılarında anlamlı bir farklılık bulunmuştur. Ayrıca ortaokulda bilgisayar programlama öğretimi oyun geliştirme süreci olarak gerçekleştirmenin daha faydalı bir yöntem olduğu söylenebilir.
Öztürk (2016)	Akademik başarı, bilgisayara yönelik tutum, kendi kendine öğrenme düzeyleri	Programlama dili öğretiminde ters yüz öğretim yöntemi	Ters yüz öğrenme yöntemi ile öğrenim gören öğrencilerin akademik başarıları, teknolojiyle kendi kendine öğrenme düzeyleri ve bilgisayara yönelik tutumları Geleneksel Öğretim Yöntemi ile öğrenim gören öğrencilere göre daha yüksek çıkmış ve gruplar arasındaki fark anlamlı bulunmuştur.
Tilki (2016)	İlgi ve motivasyon	Programlamadaki temel kavramlarını web tabanlı oyunla öğretim	Programlamadaki temel kavramları web tabanlı oyunla öğretim ön test sonuçları arasında ilgi ve motivasyon puanları açısından anlamlı fark bulunmuştur.
Gregg (2014)	İşbirlikçi çalışma, medyada şiddet farkındalığı, programlama	Programlamayla geliştirilen video oyunları ile eleştirel medya okuryazarlığı öğretimi	Öğrencilere bilgisayar programlama eğitimi verilmesi eleştirel medya okuryazarlık becerileri öğretimi açısından anlamlı bir farklılık yaratmamıştır.
Kalelioğlu (2015)	Problem çözme açısından yansıtıcı düşünme becerileri	Programlama öğretimi	t-testi sonuçlarına göre, code.org sitesinde ilköğretim öğrencilerine programlama öğretmek, problem çözme açısından yansıtıcı düşünme becerilerinde herhangi bir farklılığa neden olmamıştır. Bununla birlikte, kız öğrencilerin problem çözme yeteneğine yönelik yansıtıcı düşünme becerileri, erkeklerin yansıtıcı düşünme becerilerinden daha fazla artış göstermektedir.

Tablo 2'deki veriler incelendiđinde, deneysel alıřmaların tamamına yakınında bađımsız deđiřken olarak "programlama đretimi" deđiřkeninin ele alındıđı, en ok incelenen bađımlı deđiřkenlerin ise akademik bařarı ile dūřınme becerileri olduđu belirlenmiřtir. "programlama đretimi" bađımsız deđiřkeninin yanı sıra "cinsiyet", "mobil destekli programlama đretimi", "harmanlanmış đrenme yntemi", "programlama dili đretiminde ters yz đretim yntemi", "programlamadaki temel kavramlarını web tabanlı oyunla đretim yntemi" ve "programlamayla geliřtirilen video oyunlarıyla eleřtirel medya okuryazarlıđı đretimi" deđiřkenlerinin de bađımlı deđiřkenler zerindeki etkisinin arařtırıldıđı grlmektedir. Deneysel alıřmaların byk bir kısmında deney grubunda yer alan đrenciler programlama đretimini ocuklara zel geliřtirilen bir ara kullanarak alırken kontrol grubunda yer alan đrenciler ise geleneksel ortamda eđitim almıřlardır.

Alanyazın incelendiđinde, harmanlanmış đrenme, dijital hikye anlatımı, tersyz đrenme ve mobil destekli ortamlarda đretimin etkililiđini arařtırmak iin programlama đretiminin kullanıldıđı alıřmalara (Ceylan, 2015; Dehmenođlu, 2015; ztrk, 2016) rastlanmaktadır. Mevcut deneysel alıřmalar, erken yařta programlama đretiminin programlama, matematik gibi alıřılan alanda akademik bařarı zerinde pozitif bir etki oluřturduđunu (Aslan, 2014; Ceylan, 2015; Dehmenođlu, 2015; Glmez, 2009; ztrk, 2016) gstermektedir. İncelenen deneysel alıřmalar, erken yařta programlama đretiminin programlama, matematik gibi alıřılan alanda akademik bařarı zerinde pozitif bir etki oluřturduđunu (Aslan, 2014; Ceylan, 2015; Dehmenođlu, 2015; Glmez, 2009; ztrk, 2016) gstermektedir.

Erken yařta programlama đretiminin problem zme ile yansıtıcı ve eleřtirel dūřınme becerileri zerinde olan etkisinin incelendiđi alıřmalarda (Aslan, 2014; etin, 2012; Dođan, 2015; Olgun, 2014) programlama đretimin problem zme ve dūřınme becerilerinin geliřtirilmesinde olumlu etkisinin olduđu belirtilmektedir. Ayrıca programlama đretiminin ve đretimde kullanılan araların đrencilerin derse karřı ilgi, motivasyon ve bilgisayara karřı tutuma olumlu etkileri bulunmaktadır (ztrk, 2016; Tilki, 2016). te yandan Gregg (2014) tarafından yapılan alıřmada programlamayla geliřtirilen video oyunlarıyla eleřtirel medya okuryazarlıđı eđitimi verilmesi eleřtirel medya okuryazarlık becerileri đretimi aısından anlamlı bir farklılık yaratmamıřtır.

### 6.1.3. Programlama Eğitimiyle İlgili Çalışmaların Çalışma Grubunun Demografik Özellikleri

Bu bölümde tarama sonucunda elde edilen çalışmaların çalışma grubu özellikleri katılımcı profili, katılımcı sayısı, yaş, sınıf düzeyi, uygulama yapılan ve örneklem seçim şekli bölge açısından ele alınmıştır. Tablo 3 incelendiğinde belirli bir örneklem grubu üzerinde uygulama yapılan çalışmaların örneklemeleri incelendiğinde, herhangi bir özellik belirtilmeyen, ortaokul, 9-14 yaş arası öğrencilerin daha çok tercih edildiği görülmektedir. Özel eğitim kapsamında üstün yetenekli öğrencilerin örneklem olarak alındığı çalışmalar da (Durak, 2016) bulunmaktadır. Örneklem sayısı ise en sık 31 ile 100 arasında değişiklik göstermektedir. Uygulama yapılan çalışmalarda kentsel bölgeler tercih edilmiştir. Örneklem seçiminde de en sık tercih edilen yöntem amaca uygun örnekleme yöntemidir.

Tablo 3. İncelenen Çalışmalarda Üzerinde Çalışılan Çalışma Grubunun Özelliklerinin Dağılımı

	Seçenekler	f	%
Katılımcı Profili	Üstün zekâlı/yetenekli	1	4.17
	Herhangi bir özellik belirtilmemiş	23	95.83
Katılımcı Sayısı	1-10 arası	2	8
	11-30 arası	6	24
	31-100 arası	11	44
	101-300 arası	4	16
	301-1000 arası	2	8
Yaş	0-5 yaş	3	13.63
	6-8 yaş	2	9.10
	9-14 yaş	15	68.17
	15-18 yaş	1	4.55
Sınıf Düzeyi	Öğretmenler	1	4.55
	Okul Öncesi	3	13.04
	İlköğretim (1-5)	5	21.74
	Ortaokul (6-8)	11	47.83
	Ortaöğretim (9-12)	2	8.69
	Lisans (Eğitim Fak.)	1	4.35
	Öğretmenler	1	4,35
Uygulama Yapılan Bölge	Kırsal	1	4.3
	Kentsel	18	78.3
	Diğer	4	17.4

Örneklem Seçim Şekli	Rastgele	1	4
	Kolay ulaşılabilir örnekleme	6	24
	Amaca uygun	15	60
	Evrenin tamamı	3	12

#### 6.1.4. Programlama Eğitiminde Kullanılan Veri Toplama Araç Türlerinin Dağılımı

Erken yaşta programlama eğitimiyle ilgili olarak ulusal ve uluslararası düzeyde yapılan çalışmalarda kullanılan veri toplama araç türleri incelenmiş ve edinilen veriler Tablo 4'te sunulmuştur.

Tablo 4. İncelenen Çalışmalarda Kullanılan Veri Toplama Araçları

Veri Toplama Araçları	f	%
Görüşme (Görüşme/Odak Grup Görüşmesi)	12	30.77
Başarı Testi	8	20.51
Anket	6	15.38
Gözlem Formu	4	10.26
Proje Değerlendirme Ölçeği	2	5.13
Öz yeterlilik Ölçeği	1	2.56
Motivasyon Ölçeği	1	2.56
Diğer	5	12,82

\*Diğer: Öğrenci akademik ders notu ortalamaları, video kayıtları, araştırmacı notları, alan notları, öğrenci dergileri, çalışma yapıtları ve çevrim-içi projeler.

Tablo 4'e göre veri toplama araçları çalışmaların amacına göre farklılık göstermekte, öğrencilerin, öğretmenlerin programlama araçları ve ortamlarla ilgili görüşlerini almak amacıyla yoğunlukla kullanılan veri toplama aracı görüşme formlarıdır. İkinci sırada başarı testlerinin kullanıldığı tespit edilmiştir. Ayrıca çeşitli anketler ile katılımcıların görüşlerinin alındığı görülmektedir. Veri analiz yöntemleri açısından çoğunlukla içerik analizi gibi nicel yöntemler tercih edilmiştir.

#### 6.2. Erken Yaşta Programlama Eğitimi ile İlgili İncelenen Çalışmalarda Saptanan Sorunlar

Bu çalışmanın alt araştırma sorularından bir diğeri "incelenen çalışmalarda erken yaşta programlama eğitimi ile ilgili hangi sorunlar gözlenmiştir?" şeklinde belirlenmiştir. Bu soruya yanıt vermek üzere erken yaşta programlama eğitimi ile

ilgili incelenen makalelerde saptanan sorunlar incelenmiştir. Araştırmada erken yaşta programlama eğitimiyle ilgili çalışmalarda karşılaşılan sorunlara bakıldığında Dehmenoğlu (2015) okuldaki kız öğrenci sayısının az olması sebebiyle araştırmada cinsiyete bağlı bir analizin yapılamadığını ve bu bağlamda uygulamayla ilgili bazı sorunların yaşandığını belirtmiştir. Uygulama sürecinde yaşanan bir diğer sorun ise Kod Her Yerde uygulamasının kod çalıştırma kısmında güvenlik nedeniyle bazı kodların çalıştırılmasına izin verilmemesi ve bu engellenen kodların öğrencinin ileri seviye denemeler yapması konusunda önünde bir set oluşturduğuna işaret edilmiştir.

Dış çevre ile ilgili saptanan sorunlara bakıldığında Ceylan (2015) programlama öğretimi konusunda öğretmenler dahil tüm paydaşların bilinçli olmaması, Durak (2016) ise velilerin öğrencilerin bilişim teknolojileri kullanımıyla ilgili ders dışı kısıtlamaları, dersin zorunlu ders olmamasından ötürü sınıf öğretmenlerinin bu dersi gerekli bir ders olarak görmemesinin programlama öğretimi açısından sınırlılıklara neden olduğunu belirtmiştir. Bers, Flannery, Kazakoff ve Sullivan (2014) çalışmalarında öğretmenlerin programlama ile ilgili yeterince bilgi sahibi olmamasının erken yaşta programlama eğitiminde bir sorun olduğunu dile getirmiştir.

Programlamanın öğretiminde kullanılan araçlarla ilgili ortaya çıkan sorunlara bakıldığında Patana (2016) göre ana sınıfında programlama öğretiminde öğrencilerin okuma-yazma bilmemesi araçların kullanılabilirliğini düşürmektedir. Benzer şekilde Bers, Flannery, Kazakoff ve Sullivan (2014) da özellikle küçük yaşta çocukların gelişim özelliklerine uygun çok fazla araç bulunmamasının bir sorun olduğunu belirtmiştir. Öte yandan Hill (2015) tarafından yapılan çalışmada Scratch'e bazı yeni bölümler eklenmiş ancak bu çalışmada ilköğretim sınıflarında blok tabanlı ortamların kullanılmasında çok fazla seçeneğe sahip ortamların öğrencileri programlamadan uzaklaştırdığı görülmüştür.

### **6.2.1. Erken Yaşta Programlama Eğitimi ile İlgili İncelenen Makalelerde Öneriler**

Bu araştırma çerçevesinde yapılan tarama sonucunda Gülmez (2009) tarafından yapılan çalışmada uygulamaya yönelik bazı öneriler yapılmıştır. Bu çalışmada yaş düzeyine göre özellikle ilköğretim öğrencilerine hitap edebilecek araç ve yöntemlerin kullanılması ve programlama konusunda başarılı olan öğrencilerin ilgili bölümlere ve mesleklere yönlendirilirken akademik ders başarılarının dikkate alınması gerektiği önerilmiştir. Çünkü çalışmada matematik, fen, bilgisayar derslerinin akademik başarısı ile programlama başarısının pozitif yönde ilişkili olduğu bulunmuştur. Öte yandan Çetin (2012) tarafından yapılan çalışmada öğrencilerin

problem çözme becerilerine bakılmıştır. Gelecekte yapılan çalışmalarda eleştirel düşünme, analitik düşünme gibi becerilerin incelenmesi önermiştir. Olgun (2014) tarafından yürütülen çalışmada ise programlama eğitiminin daha erken yaşta (en geç 6-7 yaş) ve uzun sürece yayılarak verilmesi önerilmiş, Hill (2015) tarafından yapılan çalışmada ise giriş seviyesinde çocuklar için programlama dilleri seçeneklerinin genişletilmesi gerekliliği belirtilmiştir. Böylece bilgisayar bilimi, daha fazla çocuğun programlama ile ilgilenmesini ve bir bütün olarak bu alana daha fazla çeşitlilik getirmesini teşvik edebileceğini öne sürmüştür. Sáez-López, Román-González ve Vázquez-Cano (2016) ilköğretim 5. ve 6. sınıf eğitim ortamlarında müfredat dışı bir uygulama yoluyla bir görsel programlama dili uygulamasını önermektedirler. Erken yaşta programlama öğretiminde uygulamaya dönük eksikleri göz önüne alan önerilere bakıldığında Ceylan (2015) Scratch programının mobil kullanım için tasarlanmış tablet cihazların kullanımına yönelik uygulamasının geliştirilmesi öğrencilerin Scratch ile daha fazla zaman ayırmalarını sağlayacağından faydalı görülmüştür.

Adleberg (2013) programlamanın öğretim süreciyle ilgili bazı öneriler sunmuştur. Örneğin dışarıdan gelen geribildirim ve ödül, çocuklara açık uçlu bir tasarım ve uygulama özerkliğini veren platformlar ile birleşince programlama daha motive edici bir öğrenme türü olacaktır. Dehmenoglu (2015) ise programlama öğretiminde oyunlaştırmanın başarıyı artırdığını ve bu nedenle etkili bir yöntem olarak kullanılmasını önermektedir. Bu öneriler bağlamında her yaş grubu için yüz yüze programlama öğretimine mobil öğrenmenin dahil edilmesinin başarıya önemli katkı getirebileceği söylenebilir. Ayrıca öğrencilerin programlama faaliyetlerinde nasıl bir düşünme süreci izlediği, çevrim-içi programlama faaliyetleri izlenerek analiz edilebilir (Lye & Koh, 2014).

Gelecekteki araştırmacılara sunulan önerilere bakıldığında Durak'ın (2016) çalışmasında dersin kitabının olmayışının önemli bir eksiklik olarak görüldüğü ve programlamayla ilgili ilkökul ders kitabı veya e-kitap tasarımına yönelik çalışmaların yapılmasının önerildiği görülmektedir. Kayabaşı (2016) ise Alice programının özellikle programlamaya giriş derslerinde kullanılmak üzere Türkiye'de alternatif bir öğretim teknolojisi olma potansiyeline sahip olduğunu ve bu doğrultuda kullanılması gerektiğini belirtmiştir.



### 6.3. Programlama Eğitiminde Kullanılan Ortam/Araç Türü ve Öğrenme Alanları

Araştırmanın ikinci alt problemi “Erken yaşta programlama eğitimi ile ilgili çalışmalarda yer verilen öğretim sürecinde hangi ortamlar, araçlar kullanılmakta, hangi öğrenme alanları (Matematik/ Fen/ Bilgisayar vb..) tercih edilmektedir?” şeklinde belirlenmiştir. Bu bölümde tarama sonucunda elde edilen çalışmalarda kullanılan ortam, araç türü ve öğrenme alanları Tablo 5 ve Tablo 6’da ele alınmıştır.

#### 6.3.1. Programlama Eğitiminde Kullanılan Ortam/Araç Türünün Dağılımı

Erken yaşta programlama eğitimi ile ilgili incelenen makalelerde programlama öğretiminde kullanılan araçlar ve ortamlar incelenmiştir.

Tablo 5. İncelenen Çalışmalarda Kullanılan Araçlar

Araç	f	%
Scratch/Scratch Jr	10	40
Lego Mindstorms	2	8
Alice	1	4
Small Basic	1	4
Kodu	1	4
Python	0	0
App Inventor	0	0
Game Maker	0	0
Diğer...	10	40

\***Diğer:** Fcpro, Kodable, Code.org, logo tabanlı yazılımlar, TangibleK Robotics Program, araştırmada geliştirilmiş ortam, KELP-CS: 1.versiyon: Octopi-2. versiyon: LaPalaya (Scratch’in dijital storytelling ve animasyon düzenleme bölümleri eklenerek değiştirilmiş hali), Android uygulama-programlamanın temelleri

Tablo 5 incelendiğinde araştırmada incelenen çalışmaların çoğunun erken yaşta programlama eğitiminde en sık kullanılan aracın Scratch programı (n=10) olduğu görülmektedir. İkinci olarak robot programlama çalışmaları çerçevesinde Lego Mindstorms (n=2) sık kullanılmıştır. Öte yandan “diğer” seçeneğiyle araştırma kapsamında ortam geliştirimi, Scratch’in düzenlenmesiyle oluşturulan ortamların vb. kullanıldığı görülmüştür.

Tablo 6. İncelenen Çalışmalarda Kullanılan Ortam

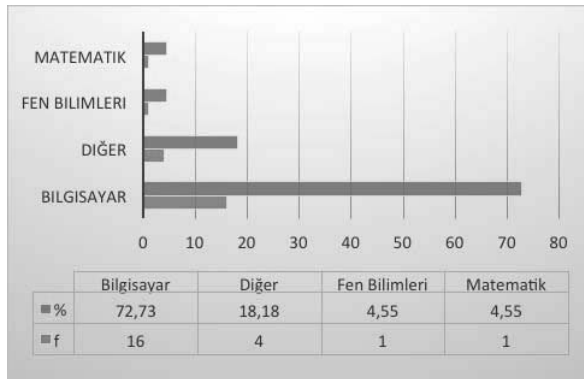
Öğrenme Ortamı	f	%
Sınıf Ortamı (Yüzyüze)	16	64
Web Tabanlı Öğrenme Ortamları	2	8
Harmanlanmış Öğrenme Ortamları	1	4
Mobil İnternet Uygulaması	1	4
Öğrenme Yönetim Sistemleri	0	0
Sosyal Medya Ortamları	0	0
Blog	0	0
Diğer	5	20

**Diğer:** Alanyazın taraması, okul dışı etkinlikler, kurslar

Tablo 6 incelendiğinde araştırmada incelenen çalışmaların çoğunda erken yaşta programlama eğitiminin en sık yüzyüze sınıf ortamında gerçekleştirildiği (n=16) görülmektedir. Daha sonra web tabanlı, sınıf ve web tabanlı birlikte (harmanlanmış) ve mobil öğrenme ortamlarında programlama öğretimi gerçekleştirilmiştir. Diğer seçeneğinde ise okul dışı etkinliklerle programlama öğretiminin gerçekleştirildiği görülmüştür.

### 6.3.2. Erken Yaşta Programlama Eğitimi İle İlgili Çalışmalarda Öğrenme Alanları

Erken yaşta programlama eğitimi ile ilgili incelenen makalelerde programlama öğretiminde kullanılan öğrenme alanları incelenmiştir.



Şekil 5. İncelenen Çalışmalarda Üzerinde Çalışılan Öğrenme Alanlarının Dağılımı

\***Diğer:** Sanat Öğretimi, Ana sınıfı oyun uygulamaları, STEM-Mühendislik, Medya okuryazarlığı

Şekil 5 incelendiğinde araştırmada incelenen çalışmaların çoğunda programlama eğitiminin Bilgisayar dersinde (n=16) kullanıldığı anlaşılmaktadır. Dört çalışmada da programlama, sanat öğretimi, medya okuryazarlığı, ana sınıfı etkinlikleri ve STEM eğitiminde disiplinlerarası bütünleştirici uygulamalar olarak uygulanmıştır.

## 7. Sonuçlar

Yapılan çalışmalar incelendiğinde SSCI indeksli dergilerde makale formatındaki çalışmaların çok fazla olmadığı görülmektedir. İncelenen çalışmalar arasında tez çalışmaları ağırlıktadır. Bunun nedeni olarak ise müfredatın erken yaşta programlama eğitimine yer vermemesi ve bu nedenle uygulamada yaşanan zorluklar, öğretmenlerin konu ile ilgili yeterince bilgi sahibi olmayışları, veli ve okul yönetimi gibi paydaşların bilinçli olmayışı ve kaynak yetersizlikleri sayılabilir. Öte yandan konu ile ilgili bir derleme çalışmanın bulunmaması da alanyazında bir boşluk olarak değerlendirilebilir.

İncelenen çalışmalarda erken yaşta programlama eğitimi uygulamalarının ortaokul düzeyinde yoğunlaştığı görülmektedir. Birçok araştırmada (Bers, Flannery, Kazakoff & Sullivan, 2014; Hill, 2015; Olgun, 2014) anasınıfı düzeyinden başlanarak programlama eğitiminin verilmesi önerilirken uygulamada bunun gerçekleşmediği dikkat çekmektedir.

Alanyazında erken yaşta programlama eğitiminde özellikle son dönemlerde öne çıkan araç Scratch'tir. Bu noktada Scratch'in kullanılabilirliği ve etkililiği tartışılmalı, araştırmacıların geleneksel yöntemle tasarlanan süreç ile Scratch'in kullanıldığı süreci karşılaştırmak yerine Scratch ile farklı programlama araçlarının etkililiği konusunda çalışma yapılması daha uygun görünmektedir. Çalışmalarda çoğunlukla uygulamanın bir çalışma çerçevesinde ve o çalışma süresince gerçekleştirildiği görülmektedir. Hatta müfredat dışı çalışmalarla uygulamaların gerçekleştirilmesi gerektiğini belirten çalışmalar da mevcuttur (Sáez-López, Román-González & Vázquez-Cano, 2016). Öte yandan programlama eğitiminin müfredat kapsamında ana sınıftan başlayarak sürekliliği sağlanarak verilmesi gerektiğini belirten çalışmalar bulunmaktadır (Hill, 2015; Olgun, 2014).

İncelenen deneysel çalışmaların tamamına yakınında bağımsız değişkenin "programlama öğretimi", en çok incelenen bağımlı değişkenlerin ise "akademik başarı" ve "düşünme becerileri" olduğu belirlenmiştir. Öte yandan deneysel çalışmaların büyük bir kısmında deney ve kontrol grubunda yöntem-araç karşılaştırması yapılmıştır. incelenen makalelerde/tezlerde erken yaşta programlama eğitimi ile ilgili saptanan sorunlara bakıldığında en sık uygulamayla ilgili sorunlar öne

çıkıştır. Programlamanın öğretiminde kullanılan araçlarla ilgili ortaya çıkan sorunlara bakıldığında araçların ekstra özelliklerden arınmış, sade ve basit olması gerektiği aksi takdirde çok fazla seçeneğe sahip ortamların öğrencileri programlamadan uzaklaştırdığı sonucuna varılmıştır. Ayrıca programlama başarısında diğer derslerdeki başarının önemli bir gösterege olacağı (Gülmez, 2009) ve programlama başarısının problem çözme, eleştirel ve analitik düşünme gibi becerileri de geliştirdiği sonucuna ulaşılmıştır.

İncelenen çalışmaların çoğunda programlama eğitiminin verilmesinin Bilgisayar dersi çerçevesinde tercih edildiği görülmektedir. Oysaki uluslararası alanyazında son dönemlerde fen, teknoloji, mühendislik ve matematik disiplinlerinin bilgi ve becerilerini bütünleştiren bir eğitim hareketi olan FeTeMM (STEM-Science Technology Engineering Mathematics) kapsamında programlama eğitimi etkin şekilde kullanılabilir (Israel, Maynard & Williamson, 2013; Lin, 2015).

Erken yaşta programlama eğitimine yönelik çalışmalarda genel olarak nitel araştırma yöntemleri kullanılmaktadır. Bu duruma neden olarak küçük yaş grubunun özelliklerinin araştırmacıların nitel yöntemi tercih etmeye yönelmesi gösterilebilir. Ayrıca çalışmalar az sayıda çalışma grubu üzerinde ve kısa sürede gerçekleştirilmiştir. Okullarda uygulama yapmanın sıkıntısı, öğretim programında ilgili bir dersin yer almaması gibi nedenler gerekçe olarak sayılabilir. Elde edilen sonuçların genellenebilirliğini sağlamak için daha geniş kitleler üzerinde daha uzun süre uygulanma yapılmasının önemli olduğu düşünülmektedir. Ayrıca özel eğitime ihtiyaç duyan öğrenciler üzerinde çalışmalar (Durak, 2016) olsa da sayıca yetersizdir. Öte yandan örneklem grubu olarak genellikle ortaokullar tercih edilmiştir. Bu sonucun, kolay erişilebilir bir örneklem seçme yönteminin tercih edilmesi nedeniyle ortaya çıktığı söylenebilir. Çalışmadan elde edilen sonuçlar doğrultusunda aşağıdaki öneriler geliştirilmiştir.

- Alanyazında erken yaşta programlama eğitiminde özellikle son dönemlerde öne çıkan araç Scratch'tir. Bu nedenle araştırmacılara, Scratch ile öğrenme ortamı tasarımı ve bu ortamların etkililiği konusunda çalışma yapımları önerilebilir.
- Programlama eğitimi çalışmalarının ders dışı faaliyet olarak belli araçlarla ya da çevrimiçi ortamlarda öğretilmesi hareketinin ötesine geçebilmesi için çalışmaların müfredat geliştirme, içerik geliştirme, diğer derslerle bütünleşik eğitsel etkinlik tasarımı ve benzeri konularla örgün eğitim boyutunda desteklenmesi önemlidir.
- Uluslararası nitelikte, dil çeşitliliği olan uygulamaya dönük nitelikli çalışmalar arttırılabilir.

## **Yansıtma Soruları**

1. Türkiye’de programlama öğretiminin uygulanabilirliğini tartışınız.
2. Programlama öğretimi çalışmalarının FATİH Projesi kapsamında yürütülebilirliğini tartışınız.
3. Programlama ile kodlama kavramları farklı kavramlar mıdır? Tartışınız.
4. K12 düzeyinde programlama öğretiminin öğrencilerin matematik, fen bilimleri gibi derslerdeki öğrenme yaşantılarına yansımalarını tartışınız.
5. Sizce çocuklara programlama öğretiminde hangi ortam ve araçlar kullanılmalı? Nedenleriyle tartışınız.

## Kaynaklar

- Akçay, T. (2009). *Perceptions of students and teachers about the use of a kid's programming language in computer courses*. Yayınlanmamış yüksek lisans tezi, Ortadoğu Teknik Üniversitesi, Ankara.
- Al-Jarrah, A.A.M. (2016). *Collaborative virtual environments for introductory programming (cveip)*. Doktora tezi, New Mexico State Üniversitesi, New Mexico.
- Almatary, M., Bakar, M. A., & Zin, A. M. (2014). The block store of block-based programming approach. *Journal of Theoretical and Applied Information Technology*, 60(2), 237-244.
- Burke, Q., & Kafai, Y. B. (2010). Programming & storytelling: opportunities for learning about coding & composition. 15.10.2016 tarihinde [https://www.seas.upenn.edu/~eas285/Readings/IDC\\_StorytellingAndProgramming.pdf](https://www.seas.upenn.edu/~eas285/Readings/IDC_StorytellingAndProgramming.pdf) adresinden erişilmiştir.
- Burke, W. Q. (2012). *Coding & composition: Youth storytelling with Scratch programming*. Yayınlanmamış Doktora tezi, Pensilvanya Üniversitesi, Pensilvanya, ABD.
- CSTA (2010). Running on empty: The failure to teach K–12 computer science in the digital age. 01.10.2016 tarihinde <http://runningonempty.acm.org/fullreport2.pdf> adresinden erişildi.
- Çetin, E. (2012). *Bilgisayar programlama eğitiminin çocukların problem çözme becerileri üzerine etkisi*. Yayınlanmamış yüksek lisans tezi, Gazi Üniversitesi, Ankara.
- DiSessa, A. A. (2001). *Changing minds: Computers, learning, and literacy*. MIT Press.
- Durak, H. (2016). Üstün yetenekli öğrencilere yazılım geliştirme süreçlerinin öğretilmesine yönelik bir öğretim programının tasarlanması ve geliştirilmesi. Yayınlanmamış Doktora tezi, Gazi Üniversitesi, Ankara.
- Einhorn, S. (2011). *Microworlds, computational thinking, and 21<sup>st</sup> century learning*. Logo Computer System Inc, White Paper. 15.10.2016 tarihinde <http://www.microworlds.com/> adresinden erişildi.
- Ersoy, Y. (2005). Matematik eğitimi yenileme yönünde ileri hareketler-I: Teknoloji destekli matematik öğretimi. *The Turkish Online Journal of Educational Technology*, 4(2), 51-63.
- 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.
- Göktaş, Y., Küçük, S., Aydemir, M., Telli, E., Arpacık, Ö., Yıldırım, G. ve Reisoğlu, İ. (2012). Türkiye'de eğitim teknolojileri araştırmalarındaki eğilimler: 2000-2009 dönemi makalelerinin içerik analizi. *Kuram ve Uygulamada Eğitim Bilimleri Dergisi*, 12(1), 177-199.
- Grover, S., & Pea, R. (2013). Computational thinking in K-12: A review of the state of the field. *Educational Researcher*, 42(1), 38-43.
- Hill, C. (2015). *Programming environments for children: Creating a language that grows with you*. Yayınlanmamış Yüksek lisans tezi, Kaliforniya Üniversitesi, Kaliforniya.
- Ismail, M. N., Ngah, N. A., & Umar, I. N. (2010). Instructional strategy in the teaching of computer programming: A need assessment analyses. *TOJET: The Turkish Online Journal of Educational Technology*, 9(2).

- Israel, M., Maynard, K., & Williamson P. (2013). Promoting literacy-embedded, Authentic STEM instruction for students with disabilities and other struggling learners. *Teaching Exceptional Children*, 45(4), 18-25.
- Kalelioğlu, F., & Gülbahar, Y. (2014). The effects of teaching programming via Scratch on problem solving skills: A discussion from learners' perspective. *Informatics in Education*, 13(1), 33-50.
- Kazakoff, E. R. (2014). *Cats in space, pigs that race: Does self-regulation play a role when kindergartners learn to code?* Yayınlanmamış Doktora tezi, Tufts Üniversitesi, Massachusetts.
- Kazakoff, E. R. (2015). *Technology-based literacies for young children: Digital literacy through learning to code*. 02.10.2016 tarihinde [http://link.springer.com/chapter/10.1007/978-94-017-9184-7\\_3#page-1](http://link.springer.com/chapter/10.1007/978-94-017-9184-7_3#page-1) adresinden erişilmiştir.
- Kiss, G. (2010). A comparison of programming skills by genders of Hungarian grammar school students. Paper presented at *Ubiquitous Intelligence & Computing and 7th International Conference on Autonomic & Trusted Computing (UIC/ATC)*, IEEE.
- Kiss, G. (2013). Teaching programming in the higher education not for engineering students. *Procedia-Social and Behavioral Sciences*, 103, 922-927.
- Kurland, D. M., Pea, R. D., Clement, C., & Mawby, R. (1986). A study of the development of programming ability and thinking skills in high school students. *Journal of Educational Computing Research*, 2(4), 429-458.
- Lamb, A., & Johnson, L. (2011). Scratch: computer programming for 21st century learners. *Teacher Librarian*, 38(4), 64-68.
- Lau, W. W., & Yuen, A. H. (2011). Modelling programming performance: Beyond the influence of learner characteristics. *Computers & Education*, 57(1), 1202-1213.
- Lee, Y. J. (2011). Scratch: Multimedia programming environment for young gifted learners. *Gifted Child Today*, 34(2), 26-31.
- Lin, V. J. (2015). *Computational thinking and technology toys*. 15.10.2016 tarihinde <http://repository.wellesley.edu/cgi/viewcontent.cgi?article=1379&context=thesiscollection> adresinden erişilmiştir.
- 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.
- 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.
- Merriam, S. B. (1998). *Qualitative research and case study applications in education*, (2nd ed.). San Francisco: Jossey-Bass.
- Microsoft, (2014). *Hour of code*. 08.11.2016 tarihinde <http://www.microsoft.com/about/corporatecitizenship/enus/youthspark/youthsparkhub/hourofcode/> adresinden erişilmiştir.
- Mohamad, S. N. H., Patel, A., Latih, R., Qassim, Q., Na, L., & Tew, Y. (2011). Block-based programming approach: challenges and benefits. In *International Conference on Electrical Engineering and Informatics (ICEEI)* (pp. 1-5).

- Partnership for 21st Century Skills (P21) (2007). *Partnership for 21<sup>st</sup> century skills*. 20.09.2016 tarihinde <http://www.p21.org/about-us/p21-framework/60> adresinden erişilmiştir.
- 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.
- Reigeluth, C. M., & Frick, T. W. (1999). Formative research: A methodology for creating and improving design theories. In C. M. Reigeluth, (Ed.). *Instructional-design theories and models, Volume II: A new paradigm of instructional theory* (633-651). New York: Routledge.
- 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. 08.10.2016 tarihinde <http://www.ijcce.org/papers/181-J028.pdf> adresinden erişilmiştir.
- Şimşek, A., Özdamar, N., Uysal, Ö., Kobak, K., Berk, C., Kılıçer, T., & Çiğdem, H. (2009). İki binli yıllarda Türkiye'deki eğitim teknolojisi araştırmalarında gözlenen eğilimler. *Kuram ve Uygulamada Eğitim Bilimleri Dergisi*, 9(2), 115-120.
- Yen, C.Z., Wu, P.H., & Lin, C.F. (2012). Analysis of expert's and novice's thinking process. Engaging Learners through Emerging Technologies. *Communication in Computer and Information Science*, 302, 122-134.
- Şimşek, H., & Yıldırım, A. (2011). *Sosyal bilimlerde nitel araştırma yöntemleri*. Ankara: Seçkin Yayıncılık.
- 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.