



Bilişim Teknolojileri Öğretmenlerinin Görsel Programlama Öğretimi Deneyimleri¹

ICT Teachers' Visual Programming Teaching Experiences

Esra Yecan, Pamukkale Üniversitesi, eyecan@pau.edu.tr

Hüseyin Özçınar, Pamukkale Üniversitesi, hozcinar@pau.edu.tr

Tayfun Tanyeri, Pamukkale Üniversitesi, ttanyeri@pau.edu.tr

ÖZ. Hesaplamalı düşünmenin (computational thinking) her öğrencinin kazanması gereken bir “21 yüzyıl becerisi” olduğu düşüncesi bir çok ülkenin ilköğretim düzeyindeki öğretim programlarına görsel kodlama derslerinin dahil olmasını sağlamıştır. Bu araştırma da, halihazırda bilişim teknolojileri dersinde görsel programlama öğretimi yapan öğretmenlerin gözünden ilköğretimde görsel programlama öğretiminin incelenmesini amaçlamaktadır. Araştırma karma yöntem araştırma yaklaşımıyla yürütülmüş; öğretmenlerle yüz yüze görüşmeler sonrasında bir anket geliştirilerek uygulanmıştır. Bulgular, öğretmenlerin, programlamaya giriş amacıyla birbirlerinden farklı yöntemler kullandıklarını, gerçek hayattan örnekler, oyun tasarımı ve grup çalışmalarıyla ders işlediklerini, öğrencilerin genel akademik başarı durumları ve programlamaya ilgi düzeyleri arasındaki farklılıkların önemli olduğunu ortaya çıkarmıştır. Ayrıca programlamanın, öğrencilerin problem çözme, mantık yürütme ve algoritma oluşturma gibi farklı alanlardaki becerilerinin, yaratıcılık ve üreticiliklerinin gelişmesine katkı sağlayabileceği sonucuna varılmıştır.

Anahtar kelimeler: Programlama eğitimi, bilişim teknolojileri öğretmenleri, hesaplamalı düşünme

ABSTRACT. Since computational thinking is being discussed as a necessary skill for 21st century, many countries decided to integrate visual programming into the elementary school curricula. This study aims to investigate “teaching to code at elementary school level” through the opinions of teachers who have chosen to teach coding as a part of ICT and Software course. The research used a mixed methodology approach, and data were collected through interviews with ICT teachers and followed by a survey. Findings revealed that teachers use various strategies for introducing the coding concept, teach the lesson through examples from daily life, game development, and group studies. Based on interview data, the individual differences of students in terms of overall academic achievement status and interest in coding seemed to be important for coding lessons. Furthermore, findings indicate that learning to code may contribute to students’ development on problem solving, logical thinking, and algorithm design skills, creativity and productivity.

Keywords: Programming education, ICT teachers, computational thinking

SUMMARY

Purpose and Significance: Computational thinking is being discussed as a necessary skill for 21st century. In addition to many other strategies, teaching visual programming is used to teach computational thinking at elementary school level. It is known that some of the ICT teachers at Turkish upper-elementary schools started to teach programming at their courses. The purpose of this study is to investigate the teachers’ point of view about teaching visual programming at elementary education.

Method: The study used exploratory sequential mixed method design incorporating qualitative and quantitative approaches to provide a holistic view about the research problem. For the first phase of the study, a purposeful sampling strategy was used and ICT teachers with visual programming teaching experience were contacted. Six teachers participated in the qualitative phase. Data was collected through semi-structured interviews. The second phase of the research consisted of a survey study. A questionnaire was developed based on the findings of the qualitative phase. The questionnaire was distributed on Internet, and a total of 107 ICT teachers filled out. Collected data were analyzed through descriptive statistical analysis including percentages and frequencies.

¹ Bu çalışmanın bir kısmı 3. Uluslararası Eğitimde Yeni Yönelimler Konferansı’nda (ICNTE 2016) sunulmuştur.

Results: The qualitative findings of the study were examined in five themes including the justifications for teaching/not teaching programming, justifications for using Scratch as the programming environment, activities used to teach programming, problems faced with while teaching programming concepts, effects of learning programming on students, difficulties in teaching programming at elementary schools, and suggestions about improving programming lessons. The sub-themes of *justifications for teaching/not teaching programming* are following the national and worldwide trends, and changing needs of students over years. Survey findings revealed that justifications for teaching programming as indicated by the majority of the participants are that it will force the students' productivity or creativity, the new generation should know at least one programming language, programming provides a way of thinking in daily life for students, and the need for keeping up with the changing times. The sub-themes of *justifications for using Scratch as the programming environment* are that it is suitable for the students' age- level, provides a visual environment, and suitable for teaching programming logic and game design. Survey findings supported the qualitative findings on this theme. The sub-themes of *activities used to teach programming* are introduction to programming activities, designing/examining games, using analogies from real life, and working in groups. Survey findings of this theme revealed that most of the participants use game design/examination, and using analogies from real life to teach programming. Furthermore, survey findings showed that teachers mostly face with *problems while teaching programming concepts* like variables, loops, and conditions. The sub-themes for *effects of learning programming on students* are that teachers think that learning programming improves students' problem solving, reasoning, algorithm creating, and productivity and creativity skills. Survey findings supported the qualitative findings on this theme. The *difficulties in teaching programming at elementary schools* theme included sub-themes like dealing with individual differences of students, preparation and infrastructural deficiencies at schools, the general lack of importance given to ICT course, and students' lack of readiness in terms of knowledge and background. In addition to qualitative findings, survey results exposed one more difficulty which is the inadequacy of time period given to teach programming.

Discussion and Conclusion: In conclusion, teaching programming – as one of the strategies to teach computational thinking- at elementary school level may contribute to children's cognitive and/or perceptual development. Teachers' experiences provide some knowledge about the teaching process. However, strategies to teach programming, its integration into elementary school curricula, and teachers' and students' points of views still needs to be discovered.

GİRİŞ

Hesaplamalı düşünme kavramının kökenleri çok daha eskilere dayansa da ilk ve orta öğretimde hesaplamalı düşünme öğretimi fikri Seymour Papert (1980) tarafından ortaya atılmıştır. Kavramın öğretimi ile ilgili tartışmalar ise Wing'in (2006) hesaplamalı düşünmenin okuma, yazma ve aritmetik gibi temel alanlardan biri olarak 21. yüzyılda her öğrenciye öğretilmesi gerektiği görüşü ile yaygınlaşmıştır. Hesaplamalı düşünme en genel anlamda problemin çerçeveselendirilmesinde ya da çözümünde bir hesaplayıcının, bu insan ya da bilgisayar olabilir, etkin bir biçimde kullanabileceği düşünme biçimi olarak tanımlanmaktadır (Wing, 2006).

Hesaplamalı düşünmenin öğretilmesinde kullanılan yöntemlerden biri olan programlamanın her öğrencinin kazanması gereken bir "21 yüzyıl becerisi" olduğu düşüncesi bir çok ülkenin ilköğretim ya da orta öğretim programına kodlama derslerinin dahil olmasını sağlamıştır. Bazı ülkeler programlamayı zorunlu olarak öğretim programına eklerken, bazı ülkelerde resmi olarak öğretim programının parçası olmasa bile öğretim etkinliklerinin bir parçası olması teşvik edilmeye başlanmıştır (Grover ve Pea, 2013; Mannila, Dagiene, Demo, Grgurina, Mirola, Rolandsson ve Settle, 2014).

Hesaplamalı düşünmenin öğretilmesinde kullanılacak birçok yöntem bulunmaktadır. CSTA (Computer Science Teachers Association), ilköğretim ve ortaöğretim düzeyinde bilgisayar bilimi standartlarını belirlemiş ve ilk ve ortaokul seviyesi için problem çözme ve hesaplamalı düşünme becerilerinin ayrı bir bilgisayar dersinde veya programlarda zaten var olan fen bilgisi,

matematik ve sosyal bilgiler gibi derslere bütünleştirilecek modüller yoluyla öğretilmesini önermiştir (2011).

Görsel programlama yazılımlarının kullanımı, hesaplamalı düşünmenin ilköğretim düzeyinde öğretilmesi amacıyla sıklıkla kullanılan araçlardandır (Armoni, Meerbaum-Salant ve Ben-Ari, 2015; Brennan ve Resnick, 2012). Özellikle küçük yaşlarda başlayan programlama etkinliklerinin çocukların bilişsel gelişimine (Lewis, 2010; Akpınar ve Altun, 2014; Atmatzidou ve Demetriadis, 2016) ya da programlamaya yönelik algılarına (Lewis, 2010; Kalelioğlu ve Gülbahar, 2014) olumlu katkısı bilinmektedir. Görsel programlama yazılımları, kullanım kolaylığı ve az deneyimli kişilerin kod yazım hatalarına düşmeden tasarıma ve üretime odaklanmalarına izin veren yapısı nedeniyle çocuklara kodlama öğretimi için oldukça uygun ortamlar sağlamaktadırlar (Grover ve Pea, 2013).

Prosedürel ve nesne yönelimli programlama dillerinin yükseköğretim düzeyinde öğretimi, kullanılan yöntemler ve bu yöntemlerin etkililikleri konusunda geniş bir araştırma alt yapısı olmasına karşın (Milne ve Rowe, 2002) ilköğretim düzeyinde görsel programlama öğretimi görece yeni bir araştırma alanıdır. Mannila ve diğ. (2014) yaptıkları çalışmada öğretmenlerin de hesaplamalı düşünme öğretimi ile ilgili ortak bir bakış açılarının tam anlamıyla oluşmadığını, hesaplamalı düşünmenin öğretim programlarına nasıl eklenilebileceği ve öğretmenlerin hesaplamalı düşünme ile ilgili bakış açılarının ortaya konulmasına gereksinim olduğunu ortaya koymuştur.

Ülkemizde de Bilişim Teknolojileri ve Yazılım dersine ait çerçeve programda programlamaya yönelik kazanımlar bulunmakta, öğretmenlerin bir kısmının dersin bir parçası olarak görsel programlama dilleri öğretimine ilişkin etkililiklere yer verdikleri bilinmektedir. Öğretim programında kazanımların geniş bir çerçevede tutulmuş olması, öğretmen kılavuz kitabı ve öğrenci çalışma kitaplarının bulunmaması gibi nedenlerle öğretmenlerin konu ve yazılım seçimi konularında birbirlerinden farklı uygulamalar yaptığı görülmektedir (Uzgun ve Aykaç, 2016). Dolayısıyla bazı bilişim teknolojileri öğretmenleri derslerinde görsel programlama eğitimi verirken, bazıları vermemektedir.

Bu araştırma da, halihazırda bilişim teknolojileri dersinde görsel programlama öğretimi yapan öğretmenlerin gözünden ilköğretimde görsel programlama öğretimini incelemeyi amaçlamaktadır. Çalışma kapsamında öğretmenlerin görsel programlama öğretimi ile ilgili deneyimleri, görüşleri ve önerileri ortaya çıkarılmaya çalışılmıştır.

YÖNTEM

Araştırmada nitel ve nicel tekniklerin bir arada kullanıldığı karma yöntem araştırması uygulanmıştır. Karma yöntem araştırmaları, farklı felsefi varsayımlardan yola çıkan farklı sorgulama yöntemlerini harmanlayarak daha kapsamlı bir anlayış sunmaktadır (Tashakkori ve Teddlie, 2003).

Birçok türü olan karma yöntem araştırmalarından, araştırma problemine uygun yöntem olan keşfedici sıralı desen kullanılmıştır. Öncelikle bilişim teknolojileri öğretmenlerinin programlama öğretimi ile ilgili deneyimlerini anlamak üzere derinlemesine veri toplanarak analiz edilmiş, ardından elde edilen bulgulara dayanılarak nicel bir veri toplama aracı geliştirilerek uygulanmıştır. Creswell (2012) tarafından belirtildiği gibi, bu araştırma yönteminde nicel veri toplanması ve analizi nitel bulgular sonrasında gerçekleştirilmiş, veriler nitel bulgular çerçevesinde yapılandırılarak sunulmuş, nicel veriler nitel bulguları destekleyici ve genellenebilmesine yardımcı olmuştur. Nitel ve nicel veri analizleri ayrı zamanlarda yapılmakla birlikte, bulgular, bütünlüğün bozulmaması amacıyla, nitel temalara göre başlıklandırılarak bir arada sunulmuştur.

Araştırma Grubu

Araştırmanın nitel aşamasının verileri, 2016 Ocak ve Şubat ayları içerisinde Denizli il merkezindeki vakıf ve devlet ortaokullarında görev yapmakta olan bilişim teknolojileri öğretmenlerinden toplanmıştır. Katılımcılar amaçlı örneklem yoluyla belirlenmiş, kartopu örneklem yöntemiyle Denizli il merkezinde programlama öğretimi yapan öğretmenlere ulaşılmaya çalışılmıştır. Belirlenen altı adet bilişim teknolojileri öğretmeniyle iletişim kurularak gönüllü katılım sağlanmıştır.

İkinci aşama olan nicel veri toplanması sürecinde ise çevrimiçi bir anket hazırlanarak, Bilişim Teknolojileri öğretmenleriyle paylaşılmış ve yine gönüllü katılım sağlanmıştır. Ankete Türkiye'nin çeşitli illerinde bilişim teknolojileri öğretmenliği yapmakta olan 107 öğretmen katılmıştır.

Verilerin Toplanması

Araştırmanın nitel verileri yüz yüze görüşme yoluyla toplanmış, bu amaçla yarı-yapılandırılmış görüşme formu kullanılmıştır. Görüşme formunun hazırlanmasında, araştırma probleminin yanı sıra ilgili alan yazın taramasında ortaya çıkan konular da belirleyici olmuştur. Bu bağlamda görüşme formunda ilk olarak demografik bilgilerine yönelik sorular yer almış, sonrasında katılımcılara, programlama öğretimi için hangi yazılımı kullandıkları, neden bu yazılımı seçtikleri, programlama öğretiminde kullandıkları yöntemler, karşılaştıkları zorluklar, programlama öğrenmenin öğrenciler üzerindeki etkilerine dair görüşleri ve son olarak programlamanın ilköğretim düzeyinde öğretim programına eklenmesiyle ilgili görüş ve önerileri sorulmuştur.

Katılımcıların tercihleri doğrultusunda, kendileri üniversite kampüsüne davet edilerek yada görev yaptıkları okulda ziyaret edilerek görüşmeler gerçekleştirilmiştir. Görüşmeler, ortalama süresi 22 dakika olmak üzere, 10-45 dakika arası sürmüş, katılımcıların izniyle ses kaydı yapılmıştır. Görüşmelerde katılımcılara programlama öğretimiyle ilgili görüşleri, deneyimleri ve önerileri sorularak ortaokul düzeyinde programlama öğretimi sürecinin mevcut durumunu ortaya çıkarmaya yönelik bilgi elde edilmeye çalışılmıştır. İç geçerliği sağlamak amacıyla, Yıldırım ve Şimşek (2013) tarafından önerildiği gibi, görüşme formunun içerik ve biçimiyle ilgili uzman görüşleri alınmış, bu amaçla çalışmayı yürüten üç araştırmacı, görüşleriyle formun geliştirilmesine katkı sağlamıştır.

Çalışmanın ikinci aşaması için, nitel verilere ait temalar, kodlar, ortak bulgular ve alternatif fikirler yardımıyla bir anket oluşturulmuş ve daha fazla sayıda öğretmenden veri toplanması hedeflenmiştir. Bu aşamada nitel bulguların temel alınmasına ek olarak, yine içeriksel ve biçimsel açıdan geçerliliğin sağlanması amacıyla üç araştırmacı anket geliştirme sürecinde ortak çalışmıştır. Hazırlanan ankette, öğretmenlerin demografik bilgilerine ek olarak, programlama öğretimi süreciyle ilgili 13 adet seçenekli soru ve bir tane de açık uçlu olmak üzere 14 soru sorulmuş, bu soruların 3 tanesi programlama öğretimi yapmayan öğretmenlere, geri kalanı ise programlama öğretimi yapmakta olan öğretmenlere yönelik hazırlanmıştır. İnternet ortamında hazırlanan anket, kolay ulaşılabilir örneklem tekniğiyle bilişim teknolojileri öğretmenlerine ulaştırılmıştır. Gönüllü katılımın esas olduğu bu sürece toplamda 107 öğretmen katılmıştır.

Verilerin Analizi

Yüz yüze görüşme kayıtları öncelikle çözümlenerek yazılı hale getirilmiştir. Daha sonra yazılı olarak e-posta yoluyla katılımcılara gönderilerek meslektaş teyidi yapılmıştır. Böylece katılımcılara verinin doğruluğunu kontrol edebilme, ekleme ve çıkarma yapabilme imkanı sağlanmıştır.

Elde edilen veriler, Marshall ve Rossman (1999) tarafından önerilen aşamalar izlenerek analiz edilmiştir: verilerin organize edilmesi, kategori ve temaların oluşturulması, verilerin kodlanması, ortak bulguların ortaya çıkarılması, alternatif fikirlerin ortaya çıkarılması. Tüm veriler analiz edilip kategori ve temalar üç araştırmacı tarafından oluşturulmuştur. Sonrasında kodlar önce bir araştırmacı tarafından belirlenmiş, daha sonra ikinci araştırmacı tarafından teyit edilmiştir. Son olarak, ortak ve alternatif bulgular belirlenerek nitel bulgular raporlandırılmıştır.

Çalışmanın ikinci aşaması olan anket çalışmasında, çoğunluğu nicel olmak üzere veriler toplanmıştır. Toplanan nicel veriler üzerinde betimleyici istatistiksel işlemler yapılmış ve yüzde, frekans sonuçları ortaya çıkarılmıştır. Açık uçlu anket sorusu ise nitel yöntemlerle analiz edilerek tema ve kodlar ortaya çıkarılmış, sonrasında ortak bulgular gruplanarak sayısallaştırılmış ve tablo şeklinde raporlandırılmıştır. Anket çalışmasına ait tüm bulgular, nitel bulguları destekleyecek ve genellenmesine yardımcı olacak şekilde, önceden oluşturulan tema ve kategoriler altında sunulmuş ve yorumlanmıştır.

BULGULAR

Araştırmanın nitel kısmı için görüşme verileri analiz edilerek temalar, kodlar ve kategoriler oluşturulmuştur. Buna göre ana temaların; öğretmenlerin neden programlama öğretmeye karar

verdikleri, Scratch programını kullanma gerekçeleri, programlama öğretiminde kullandıkları yöntemler, programlama öğrenmenin öğrenciler üzerindeki etkileri, programlama öğretiminde karşılaşılan sorunlar ve ilköğretim programına programlama dersi eklenmesi ile ilgili görüş ve önerileri çerçevesinde şekillendiği görülmüştür. Veri analizi de bu kapsamda gerçekleştirilmiştir. Nicel veriler, ilgili oldukları nitel tema başlığı altında sunulmuş, katılımcılardan yapılan alıntılarda, öğretmenlerin isimleri verilmeden takma isimlerle anılmıştır.

Katılımcı Özellikleri

Araştırmanın nitel aşamasında, çalışmaya 6 bilişim teknolojileri öğretmeni katılmıştır. Aşağıda, görüşmelere katılan öğretmenlerin özellikleri verilmiştir.

Tablo 1. Görüşme katılımcılarının özellikleri

Cinsiyet	f	%
Kadın	0	0
Erkek	6	100
Kaç yıldır Görsel Programlama Öğretiyorlar	f	%
1 yıl	3	50
3 yıl	2	33.3
5 yıl	1	16.7
Okul Türü	f	%
Devlet okulu	4	66.7
Vakıf Okulu	2	33.3
Hangi Sınıflara Programlama Öğretiyorlar	f	%
Yalnızca 6.sınıf	5	83.3
5. ve 6. sınıf	1	16.7

Tablo 1 incelendiğinde, öğretmenlerin görsel programlama öğretimi ile ilgili deneyimlerinin 1-5 yıl arasında değiştiği görülmektedir. Programlama öğretimi ile ilgili ayrıntılara baktığımızda ise, öğretmenlerin tamamı Scratch programını kullanmakta ve 4 öğretmen programlama öğretimini 6. sınıfın ikinci döneminde gerçekleştirmektedir. Özel okul öğretmenlerinden bir tanesi 5. sınıfın ikinci döneminde Scratch programını kullanmaya başladığını ve 6. sınıfta Kodu Game yazılımı ile öğretime devam ettiğini ifade etmiştir. Diğer özel okul öğretmeni ise okullarında ortaokul programında yalnızca 6. sınıfta bilişim teknolojileri ve yazılım dersi olduğunu, orada da birinci dönemin yaklaşık olarak ortalarından itibaren Scratch'a başladığını, ikinci dönem başka bir programlama diliyle devam edip, grafik tasarım ile ikinci dönemi tamamladığını belirtmektedir.

Çalışmanın ikinci aşaması olan anket uygulamasına ise Türkiye'nin çeşitli illerinde çalışan ve ortaokul düzeyinde bilişim teknolojileri ve yazılım dersi vermekte olan 107 öğretmen katılmıştır. Aşağıda anket çalışmasına katılan öğretmenlerin özellikleri verilmiştir.

Tablo 2. Çevrimiçi anket katılımcılarının özellikleri

Cinsiyet	f	%
Kadın	35	32.7
Erkek	72	67.3
Okul türü	f	%
Devlet okulu	104	97
Vakıf okulu	3	3
Derslerinizde Programlama Öğretiyor musunuz?	f	%
Evet	61	57
Hayır	46	43
Kaç Dönemdir Programlama Öğretiyorsunuz?	f	%
1-2 dönem	31	50.8
3-4 dönem	18	29.5
5-6 dönem	9	14.8
6 dönemden fazla	3	4.9

Tablo 2'nin devamı

	f	%
Programlama Öğretimini Hangi Sınıfta Gerçekleştiriyorsunuz?		
5. sınıf	0	0
6. sınıf	27	40.9
5. ve 6. sınıf	31	47
Diğer	8	12.1

Tablo 2 incelendiğinde, öğretmenlerin %67.3' ünün erkek, %32.7'sinin ise kadınlardan oluştuğu görülmektedir. Katılımcıların tamamına yakını devlet okullarında çalışmakta, %57'si derslerinde en az bir dönemdir programlama öğretmekte, %43'ü ise programlama öğretmemektedir. Öğretmenlik deneyimine bakıldığında, büyük çoğunluğunun 5-10 yıl arasında deneyime sahip olduğu, programlama öğretimindeki deneyimin ise çoğunlukla 1-2 yıl arasında olduğu görülmektedir. Ayrıca programlama öğrettiğini belirten katılımcı öğretmenlerin %47'si programlamayı 5. ve 6. sınıflarda öğrettiğini, %40.9'u yalnızca 6. sınıfta öğrettiğini, %12.1'i ise 7. ve/veya 8. sınıflarda öğrettiğini belirtmişlerdir. Diğer seçeneğini seçen öğretmenlerin bir kısmı 7-8. Sınıfların yanı sıra 5-6. sınıfta da programlama öğretmektedir.

Öğretmenlerin Ders Kapsamında Programlama Öğretme-Öğretmeme Nedenleri

Nitel Bulgular

Görüşme verilerinin analizi sonucunda öğretmenlerin neden programlama öğretmeye başladıklarına dair bulgular ortaya çıkarılmıştır. Buna göre, öğretmenlerin programlama öğretimine başlamalarına yönelik gerekçelerinden bir tanesinin değişimi takip etmek olduğu ortaya çıkmıştır. Bir öğretmen, kodlama öğretimiyle ilgili uluslararası eğilimlerin kararını etkilediğini belirtmiş, bir öğretmen devlet büyüklerinin dile getirdiği kodlama öğretiminin gerekliliği söylemlerinin önemli olduğunu söylemiştir. Aynı katılımcı, temel ofis programları öğretiminin öğrenciler açısından basit düzeyde ve sıkıcı olduğunu, yıllar içinde değişen öğrenci profiline uyum sağlamak adına programlama öğretimine başlamayı uygun gördüğünü belirtmiştir.

Nicel Bulgular

Görüşme verilerine ek olarak, anket yoluyla da öğretmenlere görsel programlama öğretimine başlama nedenleri sorulmuştur. Sorunun seçenekleri oluşturulurken, görüşme verilerinin analizinde ortaya çıkan programlama öğretimine başlama gerekçeleri ve programlama öğrenmenin öğrencilere olası etkileri temasında yer alan kodlar dikkate alınmıştır. Anket verilerine göre öğretmenlerin programlama öğretimine başlama gerekçeleri aşağıda verilmiştir.

Tablo 3. Görsel programlama öğretimine başlama gerekçeleri

Derslerinizde Niçin Görsel Programlama Öğretiyorsunuz?	f	%
Öğrencileri üretime ya da yaratıcılığa teşvik edeceğini düşünüyorum	54	88.5
Yeni neslin en az bir programlama dili bilmesi gerektiğini düşünüyorum	48	78.7
Programlama öğrencilere gündelik hayatlarında kullanabilecekleri bir düşünme biçimi sağlıyor	46	75.4
Değişime ayak uydurmak gerektiğini düşünüyorum	33	54.1
Öğretim programındaki diğer konulara kıyasla öğrencilerin daha çok ilgisini çekiyor	28	45.9
Bilişim Teknolojileri ve Yazılım dersi öğretim programında yer aldığı için	21	34.4
Toplam	230	

Tablo 3 incelendiğinde, ankete katılan ve programlama öğrettiğini belirten 61 öğretmenin soruyu cevapladığı ve birden fazla seçeneğin işaretlenebildiği bu soruda toplamda 230 işaretleme yapıldığı görülmektedir. Buna göre öğretmenlerin programlama öğretimine başlama gerekçeleri sırasıyla, öğrencileri üretime ya da yaratıcılığa teşvik edeceğini düşünmeleri (%88.5), yeni neslin en az bir programlama dili bilmesi gerektiğini düşünmeleri (%78.7), programlamanın öğrencilere

günelik hayatlarında kullanabilecekleri bir düşünme biçimi sağladığını düşünmeleri (%75.4), değişime ayak uydurmak gerektiğini düşünmeleri (%54.1), öğretim programındaki diğer konulara kıyasla öğrencilerin daha çok ilgisini çektiğini düşünmeleri (%45.9) ve programlamanın Bilişim Teknolojileri ve Yazılım dersi öğretim programında yer alması (%34.4) şeklindedir.

Anket uygulamasında, programlama öğretmediğini belirten 46 öğretmene de, programlama öğretiminden haberdar olup olmadıkları, öğretmeyi düşünüp düşünmedikleri ve öğretmeme nedenleri sorulmuştur. Anket seçenekleri oluşturulurken yine görüşme analizlerinde elde edilen bulgular kullanılmıştır. Aşağıda öğretmenlerin programlama öğretmemeye ilişkin durumları sunulmuştur.

Tablo4. Katılımcıların programlama öğretmemeye ilişkin durumları

Bilişim Teknolojileri ve Yazılım derslerinde programlama öğretildiğinden haberdar mısınız?	f	%
Evet	40	86.96
Hayır	6	13.04
Öğrencilerinize programlama öğretmeyi düşündünüz mü?	f	%
Evet	43	93.49
Hayır	3	6.51
Programlama öğretimi yapmamanızın nedeni nedir?	f	%
Okulumdaki laboratuvar koşullarının uygun olmaması	25	54.35
Öğretim programında bu konunun bulunmaması	21	45.65
Öğrencilere ağır geleceğini düşünüyor olmam	20	43.48
Konuyla ilgili bilgi sahibi olmamam	4	8.70
Diğer	6	13.04

Tablo 4 incelendiğinde, 46 öğretmenin 40'ının Bilişim Teknolojileri ve Yazılım dersinde programlama öğretildiğinden haberdar oldukları, 43'ünün de programlama öğretmeyi düşündüğü görülmektedir.

Öğretmenlerin programlama öğretmeme nedenlerine baktığımızda ise, %54.35'i okulundaki laboratuvar koşullarının uygun olmaması, %45.65'i öğretim programında konunun bulunmaması %43.48'i, öğrencilere ağır geleceğini düşünüyor olması, %8.7'si ise konuyla ilgili bilgi sahibi olmaması gerekçelerini belirtmişlerdir. Bu soruda "Diğer" seçeneğini seçen 6 öğretmenden 3'ü öğrencilerin ön bilgi ve hazır bulunuşluk düzeyi açısından yetersiz durumda olmaları nedeniyle programlama öğretimine başlamadığını belirtmiştir. Diğer 3 öğretmen ise diğer nedenler olarak sırasıyla, tek başına kod yazmanın yeterli olmayacağı, öncelikle içerik ve yöntemin belirlenmesinin gerekliliğini ve robotik şeklinde bir etkinlik olması için yeterli bütçe bulunmaması gibi sebepler belirtmişlerdir

Scratch Programını Kullanma Gerekçeleri

Nitel Bulgular

Katılımcılara, programlama öğretimine başlama gerekçelerinin yanı sıra, öğretimde hangi yazılımı kullandıkları ve neden o yazılımı seçtikleri de sorulmuştur. Buna göre, yüz yüze görüşmelere katılan altı öğretmenin tamamı, programlama öğretiminde Scratch yazılımını kullandıklarını belirtmiştir.

Yüz yüze görüşmelerde, Scratch yazılımını seçme gerekçesi olarak, öğretmenlerin tamamı programın öğrencilerin yaş grubuna uygun olmasını vurgulamıştır. Ayrıca, programın görsel bir programlama ortamı sunması, programlama mantığını öğretmeye ve oyun geliştirmeye uygun bir yapıda olması da öğretmenler tarafından Scratch'ın olumlu bulunan özelliklerindedir. 6. sınıf öğrencilerine daha önce görsel olmayan bir programlama dili öğretme deneyimi olan bir öğretmen, bu iki programlama dilini karşılaştırarak, Scratch'ın görsel olmasından dolayı öğrencilere daha kolay ve daha sevimli geldiğini belirtmiştir.

Nitel veriler ayrıca, öğretmenlerin Scratch programından nasıl haberdar olduklarıyla ilgili bulgular sunmuştur. Buna göre öğretmenlerin çoğunluğunun, genel anlamda, diğer bilişim teknolojileri öğretmenleriyle olan iletişimleri aracılığıyla bu programlama dilinden haberdar oldukları ortaya çıkmıştır. Bu bağlamda bir katılımcı, bilişim teknolojileri öğretmenlerine yönelik bir eğitim sırasında yazılımdan haberdar olduğunu belirtmiştir. Üç öğretmen, bilişim teknolojileri öğretmenlerinin bulunduğu çevrimiçi platformlar aracılığıyla, iki öğretmen ise zümre toplantılarında veya başka ortamlarda bilişim teknolojileri öğretmenleriyle kurdukları iletişim aracılığıyla Scratch öğretimini planlamaya başladığını belirtmiş, öğretmenlerin tamamı kendi programlama altyapıları sayesinde Scratch'ı kişisel olarak inceleyip sonrasında öğretimine başladıklarını belirtmiştir.

Nicel Bulgular

Ayrıca ankete katılan öğretmenlerden programlama öğrettiğini belirten 61 katılımcıya hangi yazılımı kullandıkları sorulmuştur. Öğretmenlerin programlama öğretiminde hangi yazılımları kullandıklarına dair bulgular aşağıdaki tabloda verilmiştir.

Tablo 5. Programlama öğretiminde kullanılan yazılımlar

Programlama öğretiminde hangi yazılımı kullanıyorsunuz?	f	%
Scratch	59	96.72
Kodu Game	8	13.11
Alice	4	6.56
Logo	1	1.64
Diğer	10	16.39
Toplam	82	

Tablo 5 incelendiğinde, birden fazla seçeneğin işaretlenebildiği bu soruda toplamda 82 işaretleme yapıldığı görülmektedir. Buna göre öğretmenlerin tamamına yakını (%96.72) Scratch yazılımıyla öğretim yapmaktadır. İkinci sırada ise Kodu Game yazılımı yer almaktadır (%13.11). bu soruya “Diğer” cevabını veren katılımcılardan ikisi Small Basic, ikisi Lego Mindstorms, geri kalanlar ise Java, Code.org, Lightbot, Code Monkey, Appinventor, Arduio gibi ortamları kullandıklarını belirtmişlerdir.

Anket katılımcısı öğretmenlere ayrıca programlama öğretiminde kullandıkları yazılımı niçin seçtikleri sorulmuş, yüz yüze görüşme verilerinde ortaya çıkan kodlar seçenek olarak sunulmuştur. Aşağıda katılımcıların programlama öğretiminde kullandıkları yazılımı seçme gerekçelerine ait bulgular verilmiştir.

Tablo 6. Programlama öğretiminde kullanılan yazılımı seçme gerekçeleri

Programlama öğretiminde niçin bu yazılım(lar)ı seçtiniz?	f	%
Programlama mantığını öğretebilmek açısından daha kullanışlı	50	81.97
Öğrenmesi daha kolay	49	80.33
Görsel yönden daha başarılı	48	78.69
Bu yaş grubuna en çok bu programın uygun olduğunu düşünüyorum	46	75.41
Bu yazılım(lar)ı daha önceden biliyordum	13	21.31
Bu yazılım(lar) meslektaşlarım tarafından da sıklıkla tercih ediliyor.	9	14.75
Toplam	215	

Tablo 6 incelendiğinde, birden fazla seçeneğin seçilebildiği bu soruda katılımcıların toplamda 215 seçenek işaretlediği görülmektedir. Buna göre, öğretmenlerin kullandıkları yazılımı seçme gerekçesi öncelikle söz konusu yazılımın programlama mantığını öğretebilmek açısından daha kullanışlı olduğunu düşünmeleridir (%82). Bunu sırasıyla, öğrenilmesinin daha kolay olması (%80.33), görsel yönden daha başarılı olması (%78.69), yaş grubuna en uygun yazılım olması (%75.41), öğretmenin yazılım(lar)ı daha önceden biliyor olması (%21.31) ve bu yazılım(lar)ın meslektaşları tarafından da sıklıkla tercih ediliyor olması (%14.75) gibi gerekçeler takip etmektedir.

Programlama Öğretimi Süreci: Kullanılan Yöntemler ve Yaşanan Sorunlar

Nitel Bulgular

Öğretmenlerle yapılan görüşmelerin analizi sonucunda, programlama öğretiminde uygulanan etkinlikler ve bu süreçte yaşanan sorunlar şeklinde iki tema ortaya çıkmıştır. Buna göre öğretmenler programlama öğretiminde öncelikle çeşitli başlangıç etkinlikleriyle programlamaya giriş yapmakta, sonrasında ise oyun tasarlama, var olan oyunları inceleme, konuları gerçek hayatla bağlantı kurma ve gruplar halinde çalışma gibi yöntemler kullanmaktadırlar.

Programlamaya Giriş Etkinlikleri

Öğretmen görüşmelerinden elde edilen bulgular, programlama öğretimi yapan öğretmenlerin en çok önemstedikleri konunun, öğrencilerin algoritma mantığını kavraması olduğunu göstermiştir. Bu bağlamda, öğretmenler öncelikle programlamaya giriş amacıyla bazı hazırlık etkinlikleri yapmakta ve akış diyagramlarını anlatmak, yazılımın yapısını ve kod bloklarını anlatmak, elektronik tablolama programında programlamaya hazırlık uygulamaları yapmak yada code.org sitesini kullanmak gibi yöntemler kullanmaktadırlar.

Yüz yüze görüşmelere katılan öğretmenlerin tamamı Scratch'da programlamaya başlamadan önce öğrencilerle hazırlık amaçlı bazı uygulamalar yaptıklarını belirtmiştir. Bazı öğretmenler programlamaya hazırlık olarak akış diyagramlarını anlatmaktadır. İki öğretmen bu şekilde bir ön hazırlık yaparak derslerinde günlük hayattaki basit problemlerin çözümüyle ilgili akış diyagramlarını öğrencilerle birlikte oluşturduklarını söylemişlerdir. Diğer iki katılımcı öğretmen ise programlama öğretiminin başlangıcında Scratch yazılımının yapısını ve kod bloklarını tanıttığını ya da bu yazılımla yapılmış bazı oyunları örnek olarak gösterdiğini, böylece öğrencilerin programlamanın ne olduğuyla ilgili genel bir anlayışa sahip olmalarını hedeflediklerini belirtmişlerdir.

Katılımcı öğretmenlerden biri, programlamaya hazırlık amaçlı olarak elektronik tablolama programını kullanarak basit tasarımlar, koşullu biçimlendirmeler ve matematiksel formüller yoluyla program benzeri uygulamalar geliştirdiklerini, bu hazırlığın sonrasında programlamaya giriş yaptığını belirtmiştir. Programlamaya hazırlık amaçlı etkinliklerden bahseden son katılımcı ise, Scratch öğretimine ön hazırlık olarak, öğrencilerden code.org sitesindeki basamakları yerine getirip sertifika almalarını istemektedir. Ayrıca akış diyagramlarıyla başlangıç yapan öğretmenlerden biri de, bir sonraki etkinlik olarak code.org adresindeki uygulamaları yaptırmakta ve daha sonra Scratch'a başlamaktadır.

Oyun Tasarlama/Var Olan Oyunları İnceleme

Yüz yüze görüşme yapılan öğretmenlerin programlama öğretiminde en yaygın olarak yaptıkları etkinliklerden birinin oyun tasarlama olduğu görülmüştür. Her şeyden önce öğrencilerin oyunlara meraklı olmaları ve bir oyunun nasıl oluşturulabileceği konusuna ilgili olmaları, öğretmenlerin bu yöntemi kullanmalarında etkili olmuştur. Öğretmenlerden biri, oyunun kurallarını adım adım belirleme ve her kural için gerekli tasarımı oluşturma yolunu izlediğini ve bunun öğrencileri sıkmadan algoritma mantığını anlamalarına yardımcı olduğunu belirtmiştir. İki katılımcı öğretmen de, oyun tasarımları için öğrencilerin günlük hayatlarında oynadıkları incelediklerini, onlara benzer oyunlar tasarladıklarını belirtmiştir. Öğretmenler, bu durumun öğrencilerin güdülenmesini artırdığını ve her gün oynadıkları oyunlara artık daha farklı bir gözle bakmaya başladıklarını belirtmişlerdir.

Gerçek Hayatla Bağlantı Kurma

Görüşme analizlerinden elde edilen bulgular, katılımcı öğretmenlerin, programlamaya hazırlık etkinliklerinde olduğu gibi programlama öğretimi sırasında da gündelik hayattan örnekleri kullanmaya gayret ettiğini göstermiştir. Öğretmenler bu yöntemin kavramları somutlaştırdığını ve öğrencilerin güdülenmesini artırdığını düşünmektedir. Özellikle programlama öğretimi kapsamında akış diyagramlarını işleyen öğretmenler, konunun öğrenciler açısından daha somut ve daha kolay kavranabilir olması amacıyla akış diyagramı örneklerini günlük hayattan seçmektedirler. Bunun dışında, bir öğretmen, bilişim teknolojileri ve yazılım dersi kapsamında öğrencilerin geliştirdiği tüm uygulamaları, onların gerçek hayatlarıyla bağlantılı olabilecek konulardan seçtiğini, bir diğer öğretmen ise programlama öğretimi sürecinde

telefon defteri ya da not ortalaması hesaplayıcı gibi gerçek hayatlarında kullanabilecekleri uygulamalar geliştirdiklerini ve bunun öğrencilerin motivasyonunu artırdığını belirtmiştir.

Gruplar Halinde Çalışma

Görüşme yapılan öğretmenlerin tamamının derslerinde öğrenciler uygulama geliştirmektedirler. Öğretmenler laboratuvarlarda bir bilgisayar başında en az iki öğrenci oturduğunu ve dolayısıyla programlama uygulamalarının en az iki kişilik gruplar halinde yapıldığını söylemiştir. Bu uygulamalar sırasında geride kalan öğrenciler olup olmadığı sorulduğunda ise, iki öğretmen geride kalma durumunun pek yaşanmadığını, zaten kendilerinin küçük adımlar şeklinde herkesin aynı hızda ilerleyebileceği şekilde uygulama geliştirdiklerini söylemiştir. Başka iki katılımcı ise bu sorunun grup çalışması sayesinde kendiliğinden çözüldüğünü, öğrencilerin ikili olarak çalışması nedeniyle en azından birinin ilerleyerek diğerinin de ilerlemesini sağladığını belirtmiştir. Grup çalışması konusu üzerinde çok duran bir öğretmen ise, sınıfta bazı öğrencilerin programlama konusunda diğerlerinden daha fazla ilgili ve yetenekli olması nedeniyle 3-4'er kişilik grupları oluştururken her grupta bu öğrencilerden en az bir tane bulunmasını hedeflediğini belirtmiştir. Bu öğrenciler gruptaki arkadaşlarının ilerlemesini sağlamakta ve öğretmen de daha hızlı giden öğrencilere zaman ayırmaktadır.

Nicel Bulgular

Nitel verilerden elde edilen bulgular kullanılarak oluşturulan anket aracılığıyla, öğretmenlere, programlama kavramlarını öğretirken hangi etkinlikleri kullandıkları sorulmuştur. Aşağıda, uygulanan etkinliklere yönelik bulgular sunulmuştur.

Tablo 7. Öğretmenlerin programlama kavramlarını öğretmede uyguladıkları etkinlikler

Programlama Kavramlarının Öğretilmesinde Yapılan Etkinlikler	f	%
Oyun Tasarlama	44	72.13
Programlama kavramlarını gündelik hayattan örneklerle açıklama	41	67.21
Var olan oyun/yazılımları inceleme	24	39.34
Gruplar halinde çalışma	22	36.07
Diğer	1	1.64
Toplam	132	

Tablo 7 incelendiğinde, birden fazla seçeneğin seçilebildiği bu soruda, programlama öğrettiğini belirtmiş olan 61 katılımcının toplamda 132 seçenek işaretlediği görülmektedir. Buna göre programlama kavramlarının öğretilmesinde uygulanan etkinlikler sırasıyla, oyun tasarlama (%72.13), kavramları gündelik hayattan örneklerle açıklama (%67.21), var olan oyun/yazılımları inceleme (%39.34), gruplar halinde çalışma (%36.07) olarak belirlenmiştir. Ayrıca bir katılımcı da, anket seçenekleri arasında bulunmayan tersyüz edilmiş sınıf yöntemini kullandığını belirtmiştir.

Kavramların Öğretiminde Yaşanan Sorunlar

Programlama öğretimi deneyimlerini ortaya çıkarmak amacıyla öğretmenlere, programlamaya ait hangi kavramların öğretiminde sorun yaşadıkları ve çözüme yönelik neler yaptıklarıyla ilgili açık uçlu sorular anket yoluyla sorulmuştur. Öğretmenlerin programlama öğretiminde en çok sorun yaşadıkları kavramlara ait bulgular aşağıda verilmiştir.

Tablo 8. Öğretmenlerin programlama öğretiminde zorluk yaşadıkları kavramlar

En çok hangi programlama kavram(lar)ının öğretiminde sorun yaşıyorsunuz?	f	%
Değişkenler	16	32.65
Döngüler	14	28.57
Koşullar	9	18.37
Algoritma Mantığı	6	12.24
Fonksiyonlar	3	6.12
Matematiksel yada Geometrik İşlemler	3	6.12
Diğer	12	24.49
Toplam	63	

Tablo 8 incelendiğinde, en çok hangi programlama kavram(lar)ının öğretiminde sorun yaşıyorsunuz sorusuna cevap veren 49 katılımcının toplamda 63 farklı konuda sorun yaşadığı görülmektedir. Buna göre katılımcıların %32.65'i, en çok değişken kavramının öğretiminde %28.57'si döngü, %18.37'si koşul yapıları, %12.24'ü algoritma mantığı, %6.12'si fonksiyonlar, yine %6.12'si matematiksel ya da geometrik işlemler konularının öğretiminde sorun yaşamaktadırlar.

Programlama Öğrenmenin Öğrenciler Üzerindeki Etkisi Nitel Bulgular

Programlama öğrenmenin öğrencileri ne şekilde etkilediğine yönelik gözlem ve görüşleri konusunda ise öğretmenler, öğrencilerin çeşitli alanlardaki becerilerinin geliştiğini düşünmektedirler. Özellikle problem çözme, mantık yürütme ya da algoritma oluşturma ve yaratıcılık ve üreticilik becerilerinin geliştiği düşünülmektedir.

Problem Çözme, Mantık Yürütme ya da Algoritma Oluşturma Becerileri

Yüz yüze görüşmelere katılan altı öğretmenden üçü, programlama öğrenen öğrencilerin problem çözme, mantık yürütme ya da algoritma oluşturma konusundaki becerilerinin geliştiğini düşünmektedir. Bu konuda Nedim öğretmen şöyle demiştir: *"Yani günlük hayatta karşılaştığı probleme nasıl çözüm önerisi getirebilir; oradaki akış diyagramları ile mantıksal çözümler üretmeye çalışıyor. Mutlaka problem çözme becerisini artırdığını düşünüyorum."*

Yaratıcılık ve Üreticilik Becerileri

Problem çözme, mantık yürütme ya da algoritma oluşturma becerisinin yanı sıra, programlama öğrenmenin öğrencilerin yaratıcılık ve üreticilik becerilerinin gelişmesine katkıda bulunduğu da katılımcı öğretmenlerden dördü tarafından belirtilmiştir. Öğretmenlerden ikisi, programlama öğrenilen bir bilgisayar dersinin, diğer birçok dersten farklı olarak, öğrencilerin bir ürün ortaya koyabildikleri bir ders olduğunu belirtmiştir. Katılımcılardan Davut öğretmenin görüşü şöyledir: *"Şimdiye kadar TEOG'daki fen dersi, sosyal dersi gibi derslerde herhangi bir ürün üretmiyor. Ama ilk defa bilgisayarda bir ürün yapabiliyor. Bunu yapabildiğini fark ediyor. O bakımdan seviniyorlar."*

Programlama öğrenmenin birçok alandaki beceriyi geliştirdiğini söyleyen İbrahim öğretmen ise şöyle demiştir: *"Bu konu sadece programlama öğrenmeleri değil, aynı zamanda programlamanın içinde her şey var. Matematik var, mantık yürütme var, sorun çözme var, önceden hazırlık yapma var. Bu süreç, onun gibi bölümleri kapsadığı için diğer derslerden çok farklı bir ders bence... Onları hem rahatlatıp aynı zamanda hem de düşündürmeyi sağlayan bir ders olduğunu düşünüyorum. Tek bir beceri vermiyor programlama bence."*

Öğretmenlerle yapılan görüşmelerde, özellikle bazı öğrencilerin programlama konusuna özel bir ilgisinin olması ve okul dışında da kod yazmaya ve yeni ürünler ortaya çıkarmaya devam etmelerinin öğretmenler açısından heyecan verici olduğu belirlenmiştir. Okul dışında programlama ile uğraşan öğrencilerden bahseden öğretmenler genellikle bu öğrencilerin sayısı konusunda *"her sınıftan birkaç öğrenci"* ya da *"öğrencilerin yaklaşık yüzde 10'u"* gibi bir orandan bahsetmişlerdir.

Görsel programlama öğretiminde kullanılan yöntemler başlığında da belirtildiği gibi, oyun tasarlama, öğretmenler tarafından sıklıkla kullanılan bir yöntem olarak ortaya çıkmıştır. Katılımcı öğretmenlerden ikisi bu konuda, öğrencilerin gündelik hayatlarında oynadıkları oyunlara karşı bakış açılarının değişip, oyunlar üzerinde bazı değişiklikler yapmaya yönelik bir bakış açısı geliştirmeye başladıklarını söylemişlerdir. Bu da öğrencilerin "kullanıcı" rolünden "üretici" rolüne doğru bir değişim yaşamalarını sağlamaktadır. Bu konuda Galip hoca *"Bir oyunu alıyorlar sonra karakterleri değiştiriyorlar, kuralları değiştiriyorlar mesela... Ya da bir oyun var, iki kişilik yapıyorlar gibi... Öyle şeyler yapıyorlar... Gerçek hayatla mesela Scratch ta gerçek hayattaki bir oyunu oraya uyarlamaya çalışıyorlar, oynadıkları bir oyuna benzetmeye çalışıyorlar..."*

Bu konuda Davut hoca da *"Çocukların tam dönemi... 6. sınıfların özellikle... Çok oyun oynadığı bir dönemde basit internet üzerindeki flash oyunları oynayacaklarına kendi oyunlarını tasarlıyorlar bu sayede..."* diyerek, öğrencilerin bilgisayarda sadece oyun oynayan değil aynı zamanda kendi oyununu üreten kişilere dönüştüğünü belirtmektedir.

Nicel Bulgular

Görüşme bulgularından yola çıkarak hazırlanan anketle de katılımcılara programlama öğrenmenin öğrenciler üzerinde ne tür etkileri olduğunu düşündükleri sorulmuştur. Bu etkilere yönelik bulgular aşağıdaki tabloda verilmiştir.

Tablo 9. Programlama öğrenmenin öğrenciler üzerindeki etkileri

Programlama Öğrenmenin Öğrenciler Üzerindeki Etkileri	f	%
Mantık yürütme becerileri gelişti	51	83.61
Problem çözme becerileri gelişti	43	70.49
Algoritma oluşturma becerileri gelişti	41	67.21
Yaratıcılıkları gelişti	40	65.57
Teknoloji kullanımında tüketicilikten üreticiliğe geçtiler	37	60.66
Diğer	3	4.92
Toplam	215	

Tablo 9 incelendiğinde, 61 katılımcının büyük çoğunluğunun, öğrencilerin mantık yürütme (%83.61), problem çözme (%70.49) ya da algoritma oluşturma (%67.21) becerilerinin geliştiğini düşündüğü görülmektedir. Ayrıca katılımcıların %65.57'si öğrencilerin yaratıcılık becerilerinin geliştiğini, %60.66'sı ise öğrencilerin tüketicilikten üreticiliğe geçtiğini düşünmektedir. "Diğer" seçeneğini seçmiş olan öğretmenlerden biri, öğrencileri, üzerlerindeki etkisini gözlemleyecek kadar fazla görmediklerini söylemiş, bir öğretmen öğrencilerin kendisinin bir şeyler yapabildiğini görmesinin onları mutlu ettiğini, bir diğer öğretmen ise çocukların ufuklarının genişlediğini, derse yönelik tutumlarının olumlu yönde değiştiğini ve ailelerden destek gördüklerini belirtmiştir.

Programlama Öğretiminde Karşılaşılan Zorluklar

Nitel Bulgular

Görüşme verilerinin analiziyle ortaya çıkan temalardan biri de öğretmenlerin kodlama öğretimi sürecinde karşılaştıkları zorluklar olmuştur. Öğretmenler için, öğrencilerin farklı açılardan birbirlerinden farklı seviyelerde olmaları, bilişim teknolojileri dersinin eğitim sistemi içerisindeki yeri ve önemine yönelik algılar, programlama öğretimine yönelik altyapı yada hazırlık çalışmaları yetersizlikleri gibi konular, çeşitli zorluklar olarak karşımıza çıkmaktadır.

Öğrenciler Arasındaki İlgi, Güdülenme ve Önbilgi Düzeyi Farklılıkları

Yüz yüze görüşme verilerine göre, sınıfta bulunan öğrencilerin programlamaya olan ilgisinin çok farklı seviyelerde olması, çok ilgili ve çok ilgisiz öğrencilerin aynı sınıfta bulunması, öğretim etkinliklerini düzenleme konusunda öğretmenlerin zaman zaman sıkıntı yaşamasına neden olmaktadır. Öğretmenlerin tamamı öğrenciler arasında bazı bireysel farklılıklar olduğunu, bunun da öğrenmede farklı sonuçlar ortaya çıkardığını belirtmişlerdir.

Katılımcıların en çok vurguladığı bireysel farklılık, öğrencilerin programlamaya olan ilgi düzeyleridir. Görüşme katılımcılarının tamamı, bazı öğrencilerin ilgi düzeyinin çok yüksek olduğunu, her sınıfta birkaç öğrencinin kendi evlerinde de programlama etkinlikleri yaptığını, derste yapılanlardan daha ileri seviyede uygulamalar geliştirdiklerini, bazı öğrencilerin ise programlamaya ilgi duymamaları nedeniyle geride kaldıklarını belirtmişlerdir. Bu konuda Galip öğretmen, programlamaya olan ilgisi yüksek öğrenci grubuna şu örneği vermiş; "Gündelik hayatta şöyle ki içlerinden şöyle her sınıfta 1 ya da 2 kişi var... Eve de kurayım, evde de bir şeyler yapayım... Yarışma düzenliyoruz biz bu arada oyun tasarım yarışması. Mesela orada da yapayım güzel bir şey olsun diyenler var...", programlamaya olan ilgisi düşük bir öğrenci grubu için ise şu örneği vermiştir; "Herkesin ilgisi aynı değil, seviyesi aynı değil, 3 tane öğrenci var bir sınıfta, çok ilgili değiller onlar... Animasyon yapıyorlar mesela, oyun yapmıyorlar. Zorlamıyorum ben de..."

Programlamaya olan ilgi düzeyi açısından, altı katılımcı öğretmenden ikisi cinsiyetler arası bir farktan bahsetmiştir. Bu bağlamda, erkek öğrencilerin kız öğrencilerden genel olarak daha ilgili ve daha başarılı olduğunu belirtilmiştir. Arada çok başarılı ve ilgili kız çocuklar da olduğu, ancak genele bakıldığında erkek öğrencilerin başarı ve ilgi seviyesinin daha yüksek olduğu bir katılımcı tarafından belirtilmiştir.

Ayrıca genel akademik durumu iyi düzeyde olan öğrencilerin programlamada da başarılı olduğu üç katılımcı tarafından belirtilmiştir. Genel akademik durumun yanı sıra, bir öğretmen, matematiksel düşünme becerisi gelişmiş bir öğrencinin programlamada da başarılı olduğunu, başka iki öğretmen ise problem çözme becerisinin programlama başarısını etkilediğini söylemiştir.

Programlama Öğretimi İçin Gerekli Altyapı/ Hazırlık Çalışmaları Yetersizlikleri

Görüşme verilerinin analizlerinde ortaya çıkan, öğretmenlerin zorluk olarak belirttiği bir diğer konu ise sınıfların kalabalık, laboratuvarlardaki bilgisayar sayılarının eksik olmasıdır. Görüşme katılımcısı iki öğretmen, bir bilgisayar başında ikili ya da üçlü oturan öğrencilerin hepsine yeterince zaman ayıramadığını, özellikle farklı seviyelerde olan öğrencilere rehberlik etme konusunda sıkıntı yaşadıklarını belirtmişlerdir.

Ayrıca, yine hazırlık çalışmaları yetersizliği olarak görülebilecek konulardan bir tanesi, görüşme katılımcısı öğretmenler tarafından belirtilen, programlama öğretimi konusuyla ilgili hissettikleri teorik bilgi eksikliğidir. Bu konuda iki görüşme katılımcısı, programlama öğretimi konusunda yaşadıkları bilgi eksikliğini dile getirmiştir. Konunun soyutluğu yada programlamanın öğretmenler tarafından bilinip, öğretiminin nasıl yapılacağı konusunun tam bilinmemesi gibi bulgular ortaya çıkmıştır.

Bilişim Teknolojileri Dersinin Eğitim Sistemi İçerisindeki Yeri ve Önemine Yönelik Algılar

Katılımcı öğretmenler tarafından belirtilen bir diğer sorun ise, bilişim teknolojileri dersinin ortaokul programındaki yeri ve öğrenciler, öğretmenler, veliler ve okul yöneticileri tarafından derse verilen önem olmuştur. Öğretmenlerden ikisi, bilişim teknolojileri dersinin, ortaöğretime geçiş sınavı kapsamında olmayan bir ders olması nedeniyle, yeterince önemsenmediğini, 7. ve 8. sınıflarda seçmeli dersler arasında bulunmasına rağmen, bu nedenle öğrenciler tarafından tercih edilmediğini belirtmişlerdir.

Vakıf okulunda çalışan Davut öğretmen, okulun özel programı gereği bilgisayar dersinin 5. sınıftan kaldırıldığını, yalnızca 6. sınıfta bulunduğunu belirtmiştir. Devlet okulunda çalışan Gazi öğretmen, bilişim teknolojileri dersine yönelik bakış açısını şöyle ifade etmiştir: “Zaten haftada iki saat dersimiz var ve hani çocuklar bu dersi zorunlu bir ders olarak görmüyorlar. Bunda biraz ailelerin suçu da var; oğlum sen zaten evde bir sürü bilgisayar başındasın, bir de bilgisayar dersine mi çalışacaksın diyorlar. O yüzden çok önemsemiyorlar. Devlet de zaten önemsemiyor, dersin bir kitabı yok, müfredatı yok, yardımcı materyali yok, hiçbir şeyi yok dersin.”

Bilişim teknolojileri dersinin programdaki yeriyle bağlantılı olarak, katılımcılar tarafından belirtilen bir diğer sorun ise dersin sürekliliğinin olmamasıdır. Genellikle 6. sınıfın ikinci döneminde başlayan programlama öğretimi, çok kısa bir sürede tamamlanmaktadır ve katılımcı 3 öğretmen bu konuda, öğrenci gelişimini takip edemediklerini, öğrenmenin kesintiye uğradığını ve zamanla unutulduğunu belirtmişlerdir.

Öğrencilerin Altyapı Yada Ön Bilgi Düzeyi Olarak Hazır Durumda Olmamaları

Öğrencilerin matematik yada problem çözme konularındaki becerilerinin farklı seviyelerde olmasının yanı sıra, zaman zaman öğrencilerin genelinde matematik dersinin bazı konularıyla ilgili bilgi eksikliği nedeniyle zorluk yaşadıklarını belirten iki öğretmen olmuştur. Özellikle Scratch'ta tasarım yaparken gerekli olan koordinat sistemi ya da geometriyle ilgili bazı temel konuların 5. ve 6. Sınıfa kadar olan dönemde matematik derslerinde verilmemiş olması, bilişim teknolojileri öğretmenlerinin bu konuları öğrencilere anlatmasını gerektirmektedir.

Ayrıca katılımcıların programlama öğretiminde en çok zorlandıkları kavramlara yönelik açık uçlu anket sorusuna verilen cevaplar analiz edildiğinde, soruyu yanıtlayan 49 öğretmenden yedisinin, öğrencilerin farklı açılardan programlama öğrenmeye hazır olmamaları nedeniyle sorun yaşadığı görülmüştür. Buna göre, öğrenciler her şeyi hazır bekledikleri, kendi başlarına algoritma oluşturmada sorun yaşadıkları, algoritma oluşturmak yerine kodları anlamadan ezbere yazmaya yöneldikleri ya da matematiksel alt yapı ya da ön bilgileri açısından eksiklikleri bulunduğu için öğretmenler programlama öğretiminde zorluk yaşamaktadırlar. Katılımcı öğretmenlerden biri öğrenci altyapısı açısından yaşadığı zorluğu ve kendi çözümünü şu şekilde belirtmiştir: “Öğrencilerde ortaokula gelene kadar matematiksel ve mantıksal düşünme biçiminin

gelişmemesi bir sorun. Bunu aşmak için öğrencilerde matematiksel mantıksal düşünmeyi sağlayacak ön çalışmalar yapıyoruz.”

Nicel Bulgular

Görüşme verilerinden yola çıkılarak hazırlanan anket yoluyla da öğretmenlere programlama öğretimiyle ilgili karşılaştıkları en büyük zorluk sorularak öğretmenlerin deneyimleri ortaya çıkarılmaya çalışılmıştır. Öğretmenlerin karşılaştıkları zorluklara yönelik bulgular aşağıda verilmiştir.

Tablo 10. Öğretmenlerin programlama öğretiminde karşılaştıkları zorluklar

Programlama öğretimiyle ilgili karşılaştığınız en büyük zorluk nedir?	f	%
Öğrencilerin bilgi, motivasyon vb. konularda birbirlerinden farklı seviyelerde olması	20	32.79
Okuldaki laboratuvar koşullarının yetersizliği	19	31.15
Dersin (programlama öğretiminin) devamlılığının olmaması, tek dönemde bitmesi.	11	18.03
Öğrencilerin genel olarak altyapı yada ön bilgilerinin yetersiz olması	10	16.39
Diğer	1	1.64
Toplam	61	

Tablo 10 incelendiğinde, katılımcıların %32.79'unun programlama öğretimiyle ilgili karşılaştığı en önemli zorluğun güdülenme, ön bilgi vb. konularındaki düzey farklılıkları olduğu görülmektedir. Ayrıca katılımcıların %31.15'i okullardaki laboratuvar koşullarının yetersizliğini, %18.03'ü dersin (programlama öğretiminin) devamlılığının olmamasını, %16.39'u ise öğrencilerin genel olarak altyapı ya da ön bilgilerinin yetersiz olmasını en büyük zorluk olarak görmektedir.

Programlama Öğretimine Yönelik Öneriler

Çalışmanın bulguları arasında, yüz yüze görüşmelere katılan öğretmenlerin önerileri de bulunmaktadır. Bu bağlamda, katılımcı öğretmenlerin, programlamanın hangi sınıflarda verilmesi gerektiğiyle ilgili önerileri, ayrıca ilkököl ve/veya ortaokul programında zorunlu hale getirilmesi halinde ihtiyaç duyulacak hazırlık çalışmalarıyla ilgili görüşleri ortaya çıkmıştır.

Görüşmelerde öğretmenlere, programlama öğretiminin hangi sınıfta olması gerektiğiyle ilgili fikirleri sorulmuş ve katılımcıların tamamı bilişim teknolojileri dersinin yalnızca iki sınıfta verilmesinin yeterli olmadığını söylemişlerdir. Öğretmenlerden beşi, ilkökolden itibaren, hatta birinci sınıftan başlayarak bilişim teknolojileri dersinin verilmesinin yararlı olacağını belirtmişlerdir. Bir öğretmen, 5. ve 6. sınıfın yeterli olmadığını, özellikle 7. ve 8. sınıfta dersin TEOG sınavı kaygısıyla seçilmemesinin, öğrencilerin bu alandaki gelişimini engellediğini düşünmektedir. Örneğin, 6. sınıftaki programlama öğretiminden sonra, mobil programlama gibi daha ileri düzey uygulamalar öğretmek isteyen iki öğretmen, bunun için fırsat bulamadıklarını belirtmektedir.

Bilişim teknolojileri dersinin ilkököl başlangıcından itibaren verilmesi gerektiğini düşünen Davut öğretmen, dersin sürekliliğinin özellikle öğrencilerin gelişimini izleme açısından çok faydalı olduğunu, geçmiş yıllardaki deneyimine dayanarak belirtmektedir: “Şimdi mesela ana sınıftaki ilk gelen bir çocuğun fareyi kullanamaması tuşları bilmemesi, klavyenin üzerindeki tuşların çok karmaşık gelmesi, gibi sıkıntıları çözdük. Bilgisayarın nasıl çalıştığını biraz daha onların yaşayacağı şekilde maketler yaparak sınıfta onlarla gösterdik. Anasınıfından tutun ilkököl 1 de hani artırarak ilerledik. Mesela anasınıfında programlama gibi değildi, daha çok boyama programlarını kullanıyorduk ama ilkököl 1'den itibaren Scratch'ın temellerini başlattık.”

Bu konuda üç öğretmen, çocukların küçük yaşta daha yaratıcı düşünebildiklerini, bu nedenle programlama öğrenmeye küçük yaşlarda başlamanın daha etkili olacağını belirtmişlerdir. Ayrıca bir öğretmen de küçük yaşta çocukların programlama yada oyun tasarlama konularıyla daha ilgili olduklarını, 7. veya 8. sınıfa gelen çocukların sınav kaygısıyla hareket ederek dersin gereksiz olduğunu düşündüklerini ifade etmişlerdir.

Katılımcı öğretmenlerin, programlama dersinin ilkokul ya da ortaokul programında zorunlu hale gelmesi tartışmalarıyla ilgili olarak da fikirleri sorulduğunda, altı katılımcıdan beşi, zorunlu hale getirilmeden önce bazı hazırlık çalışmaları yapılması gerektiğine yönelik görüş bildirmişlerdir. İhsan öğretmen, programlama öğretimiyle ilgili çalışmalar yapan uzmanlardan da bu konuda görüş alınması gerektiğini düşünmektedir.

Öğretmenlerden dördü, okullarda öncelikle uygun altyapının oluşturulması gerektiğini belirtmektedir. Örneğin İhsan ve Galip öğretmen, eğer ilkokulun küçük sınıflarında kodlama öğretilecekse, ona uygun donanım ve yazılımın okullarda bulunması gerektiğini, öncelikle bu tür eksikliklerin tamamlanması gerekeceğini belirtmişlerdir. Davut öğretmen, örneğin robot programlama öğretimi için materyal satın alınması gerektiğini ama bunun için bir bütçe bulunmadığını söylemiştir. İhsan ve Galip öğretmen ise, Türkiye'deki okulların alt yapı koşullarının birbirlerinden çok farklı seviyelerde olabildiğini, bu durumda tüm okullarda aynı programın yürütülmesinin zor olacağını, o nedenle öncelikle koşulların eşit hale getirilmesinin gerektiğini belirtmiştir. Gazi öğretmen ise, sınıfların kalabalık olması ve laboratuvarlarda bir bilgisayarda 2- 3 öğrencinin çalışması gibi altyapı eksikliklerinin düzeltilmesinin, programlama öğretimi konusunda olumlu etkisi olacağını düşünmektedir.

TARTIŞMA ve SONUÇ

Yürütülen bu araştırma, karma bir yaklaşımla gerçekleştirilmiş ve ortaokul seviyesinde görsel programlama dersi veren az sayıda katılımcı grubundan derinlemesine veri toplanarak, sonrasında bu bulgular temelinde hazırlanan bir anket uygulamasıyla aynı konular hakkında daha geniş bir katılımcı grubuyla çalışılmıştır. Bulgular bu bağlamda değerlendirildiğinde, programlama öğretimine küçük yaşlarda başlamanın öğrencilerin gelişimi açısından olumlu katkılar sağlama potansiyeli olduğu söylenebilir.

Öncelikle görsel programlamanın, ilköğretim çağındaki çocuklara programlama öğretmek için uygun bir araç olduğu, öğrencilerin üreticilik, yaratıcılık, problem çözme, mantık yürütme gibi becerilerinin gelişmesine katkıda bulunabileceği düşünülmektedir (Akpınar ve Altun, 2014; Atmatzidou ve Demetriadis, 2016; Kalelioğlu ve Gülbahar, 2014; Lewis, 2010; Seiter ve Foreman, 2013). Programlama etkinliklerinde öğrenciler soyutlama, algoritmik düşünme, örüntü arama ve genelleme, yineleme gibi bilgisayar bilimleri kavramlarının problem çözmeye kullanımını içeren hesaplamalı düşünme ile tanışmaktadır (Brennan ve Resnick, 2012). Hesaplamalı düşünme problemlerin ve problem çözümlerinin bilgisayarların etkili bir biçimde kullanılabilmesi için yapılandırılmasına olanak sağlamaktadır. Dolayısıyla 21. Yüzyılda her öğrencinin edinmesi gereken temel becerilerden biridir (Wing, 2006).

Bu çalışmanın bulgularında da belirtildiği gibi, Bilişim Teknolojileri ve Yazılım dersinde kodlama yapmak ayrıca, öğrencilere ürün geliştirme, diğer derslerden farklı olarak kendilerine ait tasarımlar yapma ve yaratıcılıklarını ortaya çıkarma olanağı sunmaktadır. Öğretmen görüşlerine göre, programlamayla küçük yaşta tanışan öğrencilerden bazıları özel ilgi alanı ve yeteneklerini keşfetme ve kendilerini geliştirme fırsatı bulmaktadırlar.

Öncüsü olan LOGO programlama dili gibi "programlanabilir tuğlalar" yöntemini kullanan Scratch bu çalışmada en çok kullanılan görsel programlama dili olarak ortaya çıkmıştır. Scratch öğrencilere programlama etkinliklerinde yazım hatalarıyla uğraşmadan kodlamanın mantığına odaklanma olanağı sunmaktadır. Bu şekilde programlamaya yeni başlayan öğrencilere kolay öğrenilebilir ve anlamlı yazılımlar üretmeye olanak sağlayabilecek yetenekte bir programlama dili sunulmuştur (Resnick ve diğ, 2009). İlk ve ortaöğretim düzeyinde programlama öğretimine ilişkin alan yazın incelendiğinde bu özelliklerin Scratch'ın yaygın bir biçimde kullanılmasına olanak sağladığı görülmektedir (Kafai, Fields ve Burke, 2010; Kordaki, 2012).

Programlama öğretiminde kullanılan yöntemlere bakıldığında ise öğretmenler arasında ortak bazı uygulamalar bulunduğu gibi birbirlerinden tamamen farklı yöntemler kullandıkları ya da programlama öğretimi konusunda farklı bakış açılarına sahip oldukları da görülmektedir. Oyun tasarımı ve geliştirilmesi, ilgili alan yazında olduğu gibi (Basawapatna, Koh ve Repenning, 2010) bu çalışmada da programlama ve hesaplamalı düşünme öğretiminde en sık kullanılan yöntemlerden biri olarak ortaya çıkmıştır. Yapılandırıcılık kuramından yola çıkılarak oluşturulan Yapıcılık (constructionism) yaklaşımına göre öğrenciler metin, oyun, bilgisayar

programı gibi paylaşılabılır ürünler ortaya çıkardıklarında bilgi yapılandırma süreçleri daha başarılı olmaktadır (Papert ve Harel, 1991). Bu bağlamda oyun tasarımı öğrencilere uğraşlarını ürün haline dönüştürme ve bu süreçte bilgilerini yapılandırma olanağı sunmaktadır. Scratch ve Alice gibi görsel programlama dilleri, programlama konusunda önbilgiye sahip olmayan öğrencilere hem hızlı bir biçimde oyun tasarlama hem de koşul ifadelerinden, nesne tabanlı tasarım ilkelerine kadar programlama kavramlarını öğrenme olanağı sağlamaktadır. Bunun yanında öğrenciler için eğlenceli ve ilgi çekici bir uğraş olması oyun tasarımı ve geliştirilmesinin programlama öğretiminde 1990'lardan beri yaygın bir biçimde kullanılmasını sağlamıştır (Basawapatna, Koh ve Repenning, 2010).

Karmaşık programlama kavramlarını basitleştirerek öğretmek için kullanılan bir diğer yöntem gündelik hayatla benzetimler kurmadır (Dagdilelis, Satratzemi ve Evangelidis, 2004). Benzetimler yeni tanışılan, karmaşık kavramları ve soyutlamaları öğrenmek için etkili bir yöntem sunmaktadır. Açıklayıcı, sınırları kesin bir biçimde belirlenmiş ve sistematik bir yaklaşımla oluşturulan benzetimlerin programlama öğretiminde kullanılmasının programın anlaşılmasını ve oluşturulmasını kolaylaştırdığı deneysel çalışmalarla ortaya konulmuştur (San Chee, 1993). Benzetimler farklı düzeylerdeki öğrenciler için kullanılabilir. Ancak kullanılacak benzetimlerin öğrenenlerin içinde yaşadıkları kültür, cinsiyet ve yaşları göz önüne alınarak oluşturulması gerekmektedir (Forisek ve Steinova, 2012). Öğrenenlerin gündelik hayatları ile programlama kavramları arasında benzetim kurulması ya da programlama kavramlarının gündelik hayattan seçilen örneklerle anlatılması karmaşık ve bilinmeyen programlama kavramlarının daha kolay anlaşılmasına olanak sağlayabilir.

Çalışmanın bulguları ayrıca okullardaki altyapı eksiklikleri, öğrencilerin ön bilgi, motivasyon vb. açılardan birbirlerinden farklı seviyelerde olmaları gibi durumların öğretmenleri zorlayan faktörler olduğunu ortaya çıkarmıştır. Farklı beceri düzeyleri, hazır bulunuşluklar ve güdülenme biçimleri, aynı programlama öğrenme etkinliğinin başarılı ya da başarısız olmasına neden olabilmektedir (Robins, Rountree ve Rountree, 2003). Dolayısıyla bilişim teknolojileri öğretmenlerinin, bireysel farklılıkları dikkate alarak programlama öğretimini tasarlayabilecek deneyim ve bilgi düzeyine sahip olması programlama eğitiminin başarılı olabilmesi açısından oldukça önemlidir.

Okullarda programlama öğretimine başlarken öncelikle her okula eşit altyapı olanaklarının sunulması, sonrasında eğitime yönelik olarak Kalelioğlu ve Gülbahar (2014) tarafından belirtildiği gibi geçerliliği kanıtlanmış öğretim yöntemleri, başarılı uygulama örnekleri ve öğretmenler için mesleki gelişim olanakları gerekmektedir. Ancak böyle bir mesleki gelişim olanağının sunulabilmesi için öncelikle bu eğitime temel oluşturacak bilgi tabanının oluşması gerekmektedir. Dolayısıyla ilköğretim düzeyinde programlama öğretimine ilişkin bilimsel çalışmalara dayanan bir bilgi altyapısının oluşması oldukça önemlidir.

Yükseltürk ve Altıok (2015) tarafından yürütülen çalışmada da öğretmen adaylarının görsel programlama öğretimine yönelik eğitimlere ihtiyaç duyduğu ortaya çıkmıştır. Bu bağlamda, öğretmenlerin ilköğretim düzeyinde görsel programlama öğretimi ile ilgili pedagojik alan bilgisi düzeylerinin araştırılması, bilgisayar bilimlerine ait pedagojik alan bilgisi konusunda çalışmalar yapılması (Grover ve Pea, 2013) ve mevcut Bilgisayar ve Öğretim Teknolojileri Eğitimi lisans programlarının bu bağlamda incelenerek güncellenmesi önerilmektedir.

KAYNAKÇA

- Akpınar, Y. ve Altun, Y. (2014). Bilgi toplumu okullarında programlama eğitimi gereksinimi. *İlköğretim Online*, 13(1), 1- 4.
- Armoni, M., Meerbaum-Salant, O., & Ben-Ari, M. (2015). From scratch to "real" programming. *ACM Transactions on Computing Education (TOCE)*, 14(4), 25.
- Atmatzidou, S., & Demetriadis, S. (2016). Advancing students' computational thinking skills through educational robotics: A study on age and gender relevant differences. *Robotics and Autonomous Systems*, 75, 661-670.
- Basawapatna, A. R., Koh, K. H., & Repenning, A. (2010, June). Using scalable game design to teach computer science from middle school to graduate school. In *Proceedings of the fifteenth annual conference on Innovation and technology in computer science education* (pp. 224-228). ACM.

- Brennan, K., & Resnick, M. (2012). Using artifact-based interviews to study the development of computational thinking in interactive media design. *Paper presented at annual American Educational Research Association meeting, Vancouver, BC, Canada.*
- Creswell, J. W. (2012). *Educational research: Planning, conducting and evaluating quantitative and qualitative research.* Pearson Education.
- CSTA (2011). K-12 Computer Science Standards. Erişim Tarihi: 3 Eylül 2016, <http://csta.acm.org/Curriculum/sub/K12Standards.html>.
- Dagdilelis, V., Satratzemi, M., & Evangelidis, G. (2004). Introducing secondary education students to algorithms and programming. *Education and Information Technologies, 9(2)*, 159-173.
- Forišek, M., & Steinová, M. (2012, February). Metaphors and analogies for teaching algorithms. In *Proceedings of the 43rd ACM technical symposium on Computer Science Education* (pp. 15-20). ACM.
- Grover, S., & Pea, R. (2013). Computational thinking in K-12 a review of the state of the field. *Educational Researcher, 42(1)*, 38-43.
- 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)*.
- Kordaki, M. (2012). Diverse categories of programming learning activities could be performed within Scratch. *Procedia-Social and Behavioral Sciences, 46*, 1162-1166.
- Lewis, C. M. (2010, March). How programming environment shapes perception, learning and goals: logo vs. scratch. In *Proceedings of the 41st ACM technical symposium on computer science education* (pp. 346-350). ACM.
- Mannila, L., Dagiene, V., Demo, B., Grgurina, N., Mirolo, C., Rolandsson, L., & Settle, A. (2014, June). Computational thinking in k-9 education. In *Proceedings of the working group reports of the 2014 on innovation & technology in computer science education conference* (pp. 1-29). ACM.
- Marshall, C., & Rossman, G.B. (1999). *Designing qualitative research* (3rd ed.). Thousand Oaks: Sage Publications.
- Milne, I., & Rowe, G. (2002). Difficulties in learning and teaching programming—views of students and tutors. *Education and Information Technologies, 7(1)*, 55-66.
- Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas.* Basic Books, Inc.
- Papert, S., & Harel, I. (1991). Situating constructionism. *Constructionism, 36*, 1-11.
- Robins, A., Rountree, J., & Rountree, N. (2003). Learning and teaching programming: A review and discussion. *Computer Science Education, 13(2)*, 137-172.
- San Chee, Y. (1993). Applying Gentner's theory of analogy to the teaching of computer programming. *International journal of man-machine studies, 38(3)*, 347-368.
- Seiter, L., & Foreman, B. (2013). Modeling the learning progressions of computational thinking of primary grade students. In *Proceedings of the ninth annual international ACM conference on international computing education research* (pp. 59-66). ACM.
- Sisman Eren, E., & Sahin-Izmirli, O. (2012). Problems and Solution Suggestions Related to Information Technology Course According to Elementary School Principals and Information Technology Teachers (A Case from Eskisehir). *Educational Sciences: Theory and Practice, 12(4)*, 2882-2888.
- Tashakkori, A., & Teddlie, C. (2003). *Handbook of mixed methods in the social and behavioral sciences.* Thousand Oaks: Sage Publications.
- 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)*.
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM, 49(3)*, 33-35.
- Yıldırım, A. ve Şimşek, H. (2013). *Sosyal Bilimlerde Nitel Araştırma Yöntemleri.* Ankara: Seçkin Yayıncılık.
- Yükseltürk, E. ve Altıok, S. (2015). Bilişim Teknolojileri Öğretmen Adaylarının Bilgisayar Programlama Öğretimine Yönelik Görüşleri. *Amasya Üniversitesi Eğitim Fakültesi Dergisi, 4(1)*, 50-65.