



**Ondokuz Mayıs Üniversitesi Eğitim Fakültesi Dergisi**  
Ondokuz Mayıs University Journal of Faculty of Education  
e-ISSN: 2548-0278 OMU EFD, December 2023, 42(2): 1101-1148

# Algoritma ve Programlama Dersinin Matematik Öğretmen Adayları Perspektifinden Değerlendirilmesi

Evaluating the Algorithm and Programming Course  
from the Perspective of Pre-Service Mathematics  
Teachers

Nilgün GÜNBAŞ<sup>1</sup>, Şükrü İLGÜN<sup>2</sup>

<sup>1</sup>Kafkas Üniversitesi, Kars  
· ngunbas@gmail.com · ORCID > 0000-0003-2578-3083

<sup>2</sup>Kafkas Üniversitesi, Kars  
· mat.ilgun@hotmail.com · ORCID > 0000-0002-2842-2032

## Makale Bilgisi/Article Information

**Makale Türü/Article Types:** Araştırma Makalesi/Research Article

**Geliş Tarihi/Received:** 17 Mayıs/May 2023

**Kabul Tarihi/Accepted:** 19 Aralık/December 2023

**Yıl/Year:** 2023 | **Cilt-Volume:** 42 | **Sayı-Issue:** 2 | **Sayfa/Pages:** 1101-1148

**Atıf/Cite as:** Günbaş, N., İlgün, Ş. "Algoritma ve Programlama Dersinin Matematik Öğretmen Adayları Perspektifinden Değerlendirilmesi-Evaluating the Algorithm and Programming Course from the Perspective of Pre-Service Mathematics Teachers" Ondokuz Mayıs Üniversitesi Eğitim Fakültesi Dergisi, Ondokuz Mayıs University Journal of Faculty of Education, 42(2), December 2023: 1101-1148.

**Sorumlu Yazar/Corresponding Author:** Nilgün GÜNBAŞ

**Etik Kurul Beyanı/Ethics Committee Approv:** "Araştırma için Kafkas Üniversitesi Sosyal ve Beşerî Bilimler Bilimsel Araştırma ve Yayın Etiği Kurulu'ndan 17.11.2022 tarihli ve E-37673- Sayı: 39 karar sayısı ile etik kurul izni alınmıştır."

## ALGORİTMA VE PROGRAMLAMA DERSİNİN MATEMATİK ÖĞRETMEN ADAYLARI PERSPEKTİFİNDEN DEĞERLENDİRİLMESİ

### ÖZ

Programlama becerisi 21. yüzyıl becerileri arasında öne çıkan becerilerden biridir. Yükseköğretim Kurulu tarafından, 2018-2019 yılı itibarıyla eğitim fakülteleri ilköğretim matematik öğretmenliği lisans programlarına dahil edilen Algoritma ve Programlama dersi ile matematik öğretmen adaylarının programlama becerilerini geliştirmeleri hedeflenmiştir. Bu bağlamda, çalışma eğitim fakültelerinde güncellenmiş müfredatla birlikte Algoritma ve Programlama Dersi ile ilgili öğretmen adaylarının görüşlerini ortaya çıkarmayı amaçlamıştır. Aynı zamanda dersin öğretmen adaylarının teknoloji, matematiksel beceriler, matematiksel düşünme, analitik düşünme gibi becerilerine olan katkılarının belirlenmesi hedeflenmiştir. Programlama ile doğrudan ilişkisi olmamasına rağmen, Algoritma ve Programlama dersi ilköğretim matematik eğitimi programında zorunlu dersler listesinde yer almaktadır. Bu derse ilişkin öğretmen adaylarının görüşlerini almak ve dersin etkililiğini sorgulamak amacıyla nitel araştırma yöntemlerinden durum çalışması modeli kullanılmıştır. Bu kapsamda ölçüt örnekleme yöntemiyle seçilen matematik öğretmen adaylarıyla yarı yapılandırılmış görüşmeler yapılarak veriler toplanmıştır. Görüşmelerden elde edilen veriler, içerik analizi yöntemiyle analiz edilmiştir. Araştırmanın bulgularına göre öğretmen adaylarının Algoritma ve Programlama dersine karşı olumlu görüşe sahip olup, dersin teknoloji kullanım becerisini artırdığı, matematiği eğlenceli hale getirdiği, dijital matematik anlayışlarını geliştirdiği ve yaratıcı düşünme ile analitik düşünme gibi becerileri geliştirdiği yönünde görüş bildirmişlerdir. Sonuç olarak, dersin ilköğretim matematik eğitimi programlarında verilmesinin uygun olduğu, matematik öğretmen adaylarının kolaylıkla edinebileceği bir kazanım olduğu, öğretmen adaylarının düşünme becerilerini, teknolojik yeterliliklerini ve farklı öğretim yöntemleri konusundaki farkındalıklarını artırdığı düşünülmektedir.

**Anahtar Kelimeler:** Algoritma, Programlama, Matematik Öğretmen Adayı, Teknoloji Okuryazarlığı, Matematik Eğitimi.



## EVALUATING THE ALGORITHM AND PROGRAMMING COURSE FROM THE PERSPECTIVE OF PRE-SERVICE MATHEMATICS TEACHERS

### ABSTRACT

Programming skill is seen as one of the 21st-century skills. For pre-service teachers to acquire this skill, the Council of Higher Education in Türkiye included the Algorithm and Programming course in the education faculties as of the 2018-2019 academic year. In this context, it was aimed to get the opinions of pre-service teachers about the Algorithm and Programming Course, which was started to be given with the renewed curriculum. It was also aimed to reveal the contributions of the course to pre-service teachers in terms of technology, mathematical skill improvement, mathematical thinking, analytical thinking, etc. In mathematics education departments, although it is not directly related to programming such as computer engineering and information technologies, this course was listed as one of the required courses. Accordingly, it has become a necessity to get the opinions of the pre-service teachers and question the effectiveness of the course. A case study, one of the qualitative research methods was used. The participants were reached with the criterion sampling method and were interviewed using a semi-structured interview form. The data obtained from the interviews were analyzed with the content analysis method. As per the research findings, pre-service teachers expressed a positive opinion of the Algorithm and Programming course. They highlighted its role in enhancing their technology skills, making mathematics more enjoyable, fostering a deeper understanding of digital mathematics, and cultivating skills such as creative and analytical thinking. In conclusion, the appropriateness of offering the course in mathematics education departments is affirmed. It is considered an easily attainable course for mathematics pre-service teachers, contributing to the enhancement of their cognitive abilities, technological proficiency, and awareness of diverse teaching methodologies.

**Keywords:** Algorithm, Programming, Pre-service Mathematics Teacher, Technology Literacy, Mathematics Education.



## GİRİŞ

Gelişen ve değişen dünya ile bireylerin 21.yüzyıl becerilerine sahip olmaları beklenmektedir. Bu becerilerden biri de teknolojik okuryazarlık becerisi olarak değerlendirilen programlama becerisidir. Programlama anlayışı bilişim çağının ve gelişmekte olan teknolojilerin çok önemli bir unsuru olarak karşımıza çıkmaktadır. Üst düzey düşünme, sistematik düşünme, farklı bakış açıları geliştirip problemlere çözümler sunma ve yaratıcı düşünmeyi gerektiren programlama karmaşık ve zor bir süreç olarak tanımlanmaktadır (Yıldız, Çiftçi ve Karal, 2017). Alanyazında programlama becerisi ile birtakım düşünme becerileri arasında ilişki olduğunu gösteren çalışmalara rastlanmaktadır. Örneğin, programlama becerisinin analitik düşünme ile pozitif yüksek düzeyde ilişkili olduğu, mantıksal düşünme ile de orta düzeyde pozitif ilişki olduğu görülmüştür (Sebetci & Aksu, 2014). Ayrıca, mantıksal- matematiksel zekâ ve kritik düşünme becerisinin programlamada algoritma tasarımı yapma ile pozitif yönde ilişkili olduğuna dair bulgular bulunmaktadır (Korkmaz, 2012). Programlama becerisi ile öğrenciler program geliştirmenin yanında eleştirel, algoritmik ve analitik düşünme, problem çözme ve sorgulama gibi becerileri de kazanmaktadır (Yıldız, Çiftçi ve Karal, 2017).

Geleneksel olarak öğretilen programlama dillerinin karmaşık yapısı nedeniyle öğrencilerin programlama öğrenmelerinin oldukça güç olduğu bildirilmektedir. Özellikle blok-tabanlı görsel programlama dilleri sayesinde ise programlama eğitimi kolaylaştığından bu dillerin programlama dili öğreniminde bir giriş aktivitesi olarak öğretilmesi gerektiği savunulmaktadır (Çatlak, Tekdal ve Baz, 2015). Küçük yaştaki çocukların dahi kolaylıkla öğrenebileceği blok-tabanlı görsel programlama dillerinin, öğrencilerin hedef belirleme, fikir üretme ve test etme gibi problem çözme becerilerini kullanarak öğrenmelerine olanak sağladığı görülmüştür (Taylor, Harlow ve Forret, 2010). Blok-tabanlı görsel programlama dilleri, öğrencilerin programlama mantığını kolayca anlamalarını sağladığı gibi, yaratıcılıklarını (Ouahbi vd., 2015; Pinto ve Escudeiro, 2014) ve problem çözme yeteneklerini de (Shin ve Park, 2014) geliştirir.

Programlama eğitiminde ilk ve en önemli noktalardan biri programlama mantığı öğretimidir. Programlamada bir problemin çözümüne yönelik yazılan kodlar programlama dillerine göre farklılık göstermektedir. Fakat tüm programlama dillerinde programlama mantığı aynıdır. Bu sebeple programlama eğitiminde ilk önce programlama mantığı öğretiminin gerçekleşmesi gerekmektedir (Arabacıoğlu, Bül-bül ve Filiz, 2007). Programlama eğitiminden önce algoritma kavramının verilmesi aslında programlama mantığının kavranması için elzemdir. Programlama öğretimi yapılırken algoritma tasarımı yapmak çözülmesi gereken problemler için önemli bir gereksinim olarak görülmektedir. Programların oluşturulması, algoritmalar kullanılarak daha basit ve anlamlı bir hale gelmektedir. Bu sebeple de algoritmaların anlaşılması programlama eğitimi için önem arz etmektedir (Gökoğlu, 2017).

Bilgisayar programlama sadece Bilgisayar Mühendisliği gibi programlama ile direkt ilişkili olan programlarda değil aynı zamanda eğitim fakültelerinde öğrenim gören öğrenciler için de verilen bir ders halini almıştır. Bunun nedeni, sadece yazılım öğrenmekle kalmayıp aynı zamanda etkileşimli multimedya öğrenme ortamları tasarlamak amacıyla yazılım öğrenmeleridir (Fesakis ve Serafeim, 2009). Ülkemizde ve diğer ülkelerde öğretim programlarına öğrencilerin programlama becerilerini geliştirecek farklı dersler ve uygulamalar eklenmiştir (Sayın ve Seferoğlu, 2016; Arslan ve Akçelik, 2019). Yükseköğretim programlarının güncellenmesiyle öğretmen adaylarının "...teknoloji okuryazarı, araştırmacı öğretmen niteliği kazanmış olarak mezun olması beklenmektedir" (Yükseköğretim Kurulu (YÖK), 2018, s.13). Teknoloji okuryazarlığını artıran becerilerden programlama becerisi ise gerek Millî Eğitim Bakanlığı (MEB) gerekse YÖK tarafından programlara entegre edilmiştir. 2012 yılında MEB tarafından 5. ve 6. sınıflarda Bilişim Teknolojileri ve Yazılımı dersi (Pala ve Mıhçı Türker, 2018), birçok üniversitede de 2018-2019 eğitim öğretim yılı itibariyle Algoritma ve Programlama dersi kapsamında programlama eğitimine başlanmıştır.

2018-2019 eğitim-öğretim yılında eklenen Algoritma ve Programlama dersi, matematik öğretim programının hedeflediği beceri ve yetkinliklerle uyumlu olması gereken yeni derslerden biridir (Akıncı ve Dübüş, 2022). Dersin algoritma kısmı, programlama mantığının günlük yaşam örnekleriyle konuşma dili ile verilmesi süreciyken (Arabacıoğlu, Bülbül ve Filiz, 2007), programlama kısmı algoritmalar kullanılarak bilgisayara komutlar verilmesi ve uygulamalar geliştirilmesi sürecidir (Saygıner ve Tüzün, 2017). Algoritmik düşünme becerisinin takip ettiği programlama sürecinin teknolojik okuryazarlık becerilerinden biri olduğu ve bilimsel düşünme becerisini geliştirdiği de savunulmaktadır (Yıldız, Çiftçi ve Karal, 2017).

Günümüzde teknolojiye verilen önem ile yapılan çalışmalarda algoritma ve programlama eğitimine ilgi artmıştır. Alanyazın incelendiğinde programlama eğitimi ile ilgili yapılan çalışmalara rastlanmaktadır (Gökoğlu, 2017; Oh, 2017; Göksoy ve Yılmaz, 2018; Pala ve Mıhçı Türker, 2018; Alsancak Sırakaya, 2019; Arslan ve Akçelik, 2019; Öztürk ve Alper, 2019; Uzun ve Baltalı, 2019; Agbo vd. 2019; Akkaya ve Öztürk, 2020; Göncü, Çetin ve Şendurur, 2020). Örneğin, Gökoğlu (2017) çalışmasında 372 Bilgisayar Programcılığı öğrencisinin programlama eğitiminde algoritma kavramına dair metaforlarını incelemiştir. Çalışmanın sonucuna göre bazı öğrencilerin algoritmaya yönelik olumsuz görüşe sahip olduğu görülmüştür. Ancak, öğrencilerin çoğunluğunun algoritma kavramını programlama sürecinde bir rehber olarak gördüğü ve hedeflerine ulaştıran bir adım olarak değerlendirdiği ortaya çıkmıştır. Alsancak Sırakaya (2019) tarafından yapılan bir çalışmada, programlama öğretiminin bilgisayar programcılığı bölümü öğrencilerinin bilişimsel düşünme becerilerine etkisi incelenmiştir. Çalışmanın bulgularına göre, programlama eğitiminin cinsiyet ve programlama becerilerinden bağımsız olarak, öğrencilerin bilgi-işlemsel düşünme becerileri üzerinde anlamlı ve olumlu bir

etkisi olduğu ortaya çıkmıştır. Saygıner ve Tüzün (2017) tarafından yapılan programlamanın öğretimiyle ilgili çalışmaların analizinin yapıldığı bir çalışmada karşılaşılan zorlukların başında olayın mantık boyutunun çözülemediğinin ve öğrenenlerin bunu halledemedikleri için konuyu iletmedikleri ortaya konulmuştur. Bunun üstesinden gelmek için görsel tabanlı ortamların kullanımı önerilmiştir. Göksoy ve Yılmaz (2018) çalışmasında bilişim teknolojileri öğretmenleri ve öğrencilerinin robotik ve kodlama dersine ilişkin görüşlerini incelemiştir. Bulgular, öğretmen ve öğrencilerin söz konusu dersin verimli çalışma, sistematik, analitik, yaratıcı ve sayısal düşünme, problem çözme ve tasarlama gibi çeşitli beceriler kazandırdığını, ayrıca öğrencilerin sayısal derslerdeki başarılarının ve problem çözme becerilerinin arttığını göstermiştir.

Bilgisayar teknolojilerini verimli kullanan, elde edilen bilgiyi kalkınmaya ve üretime yansıtan yüksek eğitilmiş bireylerin varlığı toplumlar için elzemdir (Sayın ve Seferoğlu, 2016). Tüm bu bilgiler bize gelişmişlik seviyesi yüksek bir toplum olmak için programlama eğitiminin zorunluluğundan bahsetmek gerektiğini göstermiştir. Mantıksal akıl yürütmenin ana fikrini oluşturan ve akademik bir kazanım olarak kabul gören programlama eğitimi, yakın zamanda birçok ülke tarafından eğitim programlarına dahil edilmektedir. Bu çalışmada yenilenen ilköğretim matematik öğretmenliği lisans programında yer alan Algoritma ve Programlama dersine yönelik öğretmen adaylarından görüşler alınmıştır. Önceki çalışmalar genellikle bilgisayar mühendisliği öğrencileri (Benli ve Tek, 2021), bilişim teknolojileri öğretmenleri (Göksoy ve Yılmaz, 2018; Göncü, Çetin ve Şendurur, 2020) ve bilgisayar programcılığı öğrencileri (Özyurt & Özyurt, 2015; Gököğlü, 2017; Alsancak Sırakaya, 2019) ile yürütülmüştür. Bu çalışmalar genel olarak programlamayı hedef alan programlarda öğrenim gören ya da öğretmen olan bireyler ile ilgilidir. Örneğin, Göncü, Çetin ve Şendurur (2020) programlamayı öğrencilerine öğretmesi gereken öğretmenlerle; Alsancak Sırakaya (2019) programlamayı bir mesleki gereksinim olarak öğrenen bilgisayar programcılığı öğrencileriyle; Benli ve Tek (2021) ise programlama becerisinin geleceğin mühendislerine derinlemesine öğretildiği Bilgisayar ve Yazılım Mühendisliği öğrencileri ile yürütülmüştür. Bu açıdan bakıldığında matematik öğretmen adayları algoritma ve programlama becerilerini öğrencilerine direkt olarak öğretmeyecek, fakat öğrendikleri programlama becerilerini kullanarak matematik bilgilerini geniş kapsamda gerçek yaşam durumlarına uygulama imkânı elde edeceklerdir (Cline vd. 2020). Algoritma ve Programlama dersinin, programlama ile doğrudan ilişkili olmayan bir bölümde verilmesi, dersin kazanımlarının bölüm öğrencileri için uygun olup olmadığını sorgulamayı gerektirmiştir. Bu nedenle, bu çalışma, programlama ile doğrudan ilgisi olmayan eğitim fakültesi ilköğretim matematik öğretmenliği programında öğrenim gören öğretmen adaylarının görüşlerini ortaya çıkarmak amacıyla yürütülmüştür. İlgili literatür incelendiğinde, ilköğretim matematik öğretmenliği programında öğrenim gören öğretmen adaylarıyla bu konuda yapılmış herhangi

bir çalışmaya rastlanmamıştır. Bu nedenle, bu çalışma, eğitim fakültelerinde güncellenmiş müfredatla birlikte Algoritma ve Programlama Dersi ile ilgili matematik eğitimi öğretmen adaylarının görüşlerini ortaya çıkarmayı amaçlamaktadır. Aşağıdaki araştırma problemine cevap aranmıştır:

*Matematik öğretmen adaylarının ilköğretim matematik eğitimi programında yer alan Algoritma ve Programlama dersine yönelik görüşleri nelerdir?*

## YÖNTEM

### Araştırma Modeli

Bu çalışmada nitel araştırma yöntemlerinden durum çalışması kullanılmıştır. Durum çalışmalarında bir veya birden fazla birey, sosyal grup, toplum, durum, ortam ya da program doğal ortamında derinlemesine zaman ve mekâna bağlı olarak incelenir (McMillan, 2012). Bu araştırma modelinin seçilmesinin nedeni, Algoritma ve Programlama dersini öğrenim gördükleri bölümde henüz tamamlayan İlköğretim Matematik Öğretmenliği programı öğrencilerinin derste edindikleri kazanımları bizzat kendi görüşlerinden yola çıkarak ortaya çıkarmaktır. Bu nedenle, matematik öğretmen adaylarının dersi almış, dersteki ödev ve projeleri tamamlamış, final ödevini ve sınavını henüz tamamlamış bireyler olarak, dersin hedeflenen kazanımlarını ne ölçüde elde ettiklerini belirlemek amacıyla durum çalışmasına başvurulmuştur.

### Çalışma Grubu

Bu çalışmaya bir devlet üniversitesinde İlköğretim Matematik Öğretmenliği Bölümü 2. Sınıfı tamamlayan 15 öğretmen adayı katılmıştır. Katılımcıların seçiminde ölçüt örnekleme yöntemi kullanılmıştır. Bu örnekleme yönteminde genellikle belirli ölçütleri taşıyan evrenin temsilcisi olan az sayıda katılımcı seçilir (McMillan, 2012). Katılımcıların Algoritma ve Programlama dersini almış, ara sınav ve final ödevlerini tamamlamış, dersin içeriği ile ilgili dönem boyunca ürünler ortaya koymuş olması ölçüt olarak belirlenmiştir. Çeşitliliği sağlamak ve evrenin iyi bir temsilcisi olması için katılımcıların yukarıda değinilen ölçütleri iyi ( $n=5$ ), orta ( $n=5$ ) ve düşük ( $n=5$ ) seviyede taşımalarına dikkat edilmiştir. Bu seviyeler seçilirken dönem boyunca yaptıkları ödevleri istenilen kriterlerde ne seviyede tamamladıkları (üst, orta ve basit seviye) baz alınmıştır. Katılımcıların demografik bilgileri Tablo 1’de gösterilmiştir.

**Tablo 1.** *Katılımcıların Demografik Bilgileri*

| Değişkenler            |                | f | %     |
|------------------------|----------------|---|-------|
| Cinsiyet               | Kadın          | 9 | 60    |
|                        | Erkek          | 6 | 40    |
| Yaş                    | 19             | 2 | 13,33 |
|                        | 20             | 9 | 60    |
|                        | 21             | 3 | 20    |
|                        | 22             | 1 | 6,67  |
| Mezun olduğu lise türü | Anadolu lisesi | 6 | 40    |
|                        | Fen lisesi     | 5 | 33,33 |
|                        | Temel lise     | 2 | 13,33 |
|                        | Özel lise      | 1 | 6,67  |
|                        | Açık lise      | 1 | 6,67  |
| Genel not ortalaması   | 2.00-2.50      | 1 | 6,67  |
|                        | 2.51-2.99      | 9 | 60    |
|                        | 3.00-3.50      | 5 | 33,33 |
|                        | 3.50-4.00      | 0 | 0     |

### Veri Toplama Aracı ve Verilerin Toplanması

Bu çalışmanın veri toplama aracı araştırmacılar tarafından geliştirilmiştir. Geçerliliğini sağlamak üzere konu ile ilgili uzman görüşüne başvurulmuştur. Uzmanların görüşleri doğrultusunda bazı sorular üzerinde düzenlemeler yapılarak görüşme formu son şeklini almıştır. Veri toplama aracı, öğretmen adaylarının Algoritma ve Programlama dersinden elde ettikleri kazanımları, matematik ile Algoritma ve Programlamanın ilişkisini, dersin matematiğe bakış açılarını, dersin avantaj ve dezavantajlarını, kişisel ve mesleki katkıları, Algoritma ve Programlamanın matematik dersini anlamada rolünü içeren yarı yapılandırılmış görüşme sorularını içermiştir. Bazı örnek sorular “*Algoritma ve Programlama dersinde edindiğiniz kazanımlar nelerdir ve matematik dersi ile ilişkisi nedir? Algoritma ve Programlama dersinin gelecekte konuşulan dijital matematik anlayışına katkısı yorumlayınız. Algoritma ve Programlama dersinin avantaj ve dezavantajlarını açıklayınız?*” şeklindedir.

Çalışmanın verileri 2021-2022 eğitim-öğretim yılı bahar döneminde 14 hafta süren Algoritma ve Programlama dersinin tamamlanmasının ardından 2. sınıf öğretmen adaylarından toplanmıştır. Ders süresince algoritma kavramı, algoritma şemaları, akış diyagramları, blok tabanlı kodlama programı Scratch ve uygulamaları ve ardından Java programlama dili ve uygulamaları gösterilmiştir. Dönem boyunca öğrencilerden ilgili konuya ilişkin haftalık ödevler istenmiş ve dönüt



sağlanmıştır. Ara sınav ve final sınavında ise öğretmen adaylarından uygulamalı ödev yapmaları istenmiştir. Dönem sonunda yarı yapılandırılmış, yukarıda bahsi geçen soruların da içerisinde yer aldığı, görüşme soruları yazılı olarak öğretmen adaylarına sunulmuş ve bireysel olarak bu soruları cevaplandırmaları istenmiştir. Öğretmen adaylarına toplamda dokuz soru yöneltilmiş ve soruları yanıtlarken süre sınırı konulmamıştır. Ortalama 30-35 dk. da tamamlanan cevaplandırma süreci sonunda cevaplar toplanmış ve analiz edilmiştir.

### Verilerin Analizi

Öğretmen adaylarının cinsiyet, yaş, sınıf düzeyi, genel not ortalaması ve mezun olduğu lise türü gibi demografik bilgileri frekans ve yüzdelerle verilmiştir (Tablo 1). Katılımcıların açık uçlu sorulara verdiği cevaplardan elde edilen veriler ise içerik analizi yöntemiyle analiz edilmiştir. Bu bağlamda öğretmen adaylarının verdikleri cevaplar içerisinde en çok tekrarladıkları kavramlar tümevarım yoluyla ulaşılan kategorileri belirlemiştir: teknoloji, programlama ve kodlama, matematik ve yazılım ilişkisi, programlama öğrenim ve öğretim yöntemleri, dersin matematik anlamaya katkısı, dersin müfredatta yer alma durumu, avantajlar ve dezavantajlar şeklindedir. Araştırmacıların birbirinden bağımsız olarak oluşturduğu kodlar için kodlayıcılar arası güvenilirlik değeri hesaplanmıştır. Bunun için Miles ve Huberman'ın (1994) Görüş Birliği / (Görüş Birliği + Görüş Ayrılığı) formülü kullanılmış ve güvenilirlik değeri 0,78 olarak hesaplanmıştır.

## BULGULAR

Bu çalışma eğitim fakültelerinde güncellenmiş müfredatla birlikte Algoritma ve Programlama Dersi ile ilgili öğretmen adaylarının görüşlerini ortaya çıkarmayı amaçlamıştır. Bu doğrultuda dersin öğretmen adaylarının teknoloji, matematiksel becerilerini geliştirme, matematiksel düşünme, analitik düşünme gibi alanlara olan katkılarının belirlenmesi hedeflenmiştir. Daha önce Algoritma ve Programlama dersini hiç almayan (n=12) ya da teknoloji ile ilişkili diğer dersleri (örn. Bilişim Teknolojileri, Öğretim Teknolojileri) (n= 3) aldığını ifade eden öğretmen adaylarıyla yarı yapılandırılmış görüşme sorularıyla çalışma yürütülmüştür. Yapılan görüşmeler sonunda elde edilen veriler ışığında aşağıdaki bulgulara ulaşılmıştır:

### Algoritma ve Programlama Dersinin Teknoloji Açısından Değerlendirilmesi

Algoritma ve programlama dersiyle ilgili görüşleri alınan öğretmen adaylarının sıklıkla teknoloji vurgusu yaptığı gözlemlenmiştir. Katılımcıların %30.43'ü (n=7), dersin teknoloji çağına uygunluğuna vurgu yapmış, aynı oranda yani %30.43'ü (n=7) teknolojik yeterliliklerinin arttığını düşündüklerini belirtmiştir. %17.39'u (n=4), matematiğin teknolojiye uyarlanabileceğine dikkat çekerken, %8.70'i (n=2)

dersin verimli ve yararlı teknoloji kullanımına katkı sağladığını ifade etmiştir. Ayrıca, %8.70'i (n=2) dersin teknoloji kullanımının yaygınlaşması nedeniyle gerekli olduğunu vurgulamıştır (Tablo 2).

**Tablo 2. Teknoloji**

| Kodlar   | f         | %          | Katılımcı                           |
|--|-----------|------------|-------------------------------------|
| Teknoloji çağına uygunluk                              | 7         | 30,43      | ÖA1, ÖA3, ÖA5, ÖA6, ÖA7, ÖA9, ÖA14  |
| Teknoloji kullanımı / becerisi / yeterliliği artırıyor | 7         | 30,43      | ÖA1, ÖA2, ÖA3, ÖA7, ÖA8, ÖA10, ÖA13 |
| Matematiğin teknolojiye uyarlanması                    | 4         | 17,39      | ÖA3, ÖA5, ÖA13, ÖA14                |
| Verimli/yararlı teknoloji kullanımı                    | 2         | 8,70       | ÖA5, ÖA13                           |
| Teknoloji kullanımının yaygınlaşması                   | 2         | 8,70       | ÖA7, ÖA9                            |
| Teknolojik imkansızlıklar                              | 1         | 4,35       | ÖA5, ÖA12                           |
| <b>Toplam:</b>   | <b>23</b> | <b>100</b> | <b>12</b>                           |

Tablo 2 incelendiğinde teknoloji kategorisinde toplam altı kodun yer aldığı görülmektedir. Teknoloji kategorisindeki kodlardan bazıları “Teknoloji çağına uygunluk, teknoloji kullanım becerisinin/yeterliliğinin artması, teknoloji kullanımının yaygınlaşması” olduğu görülmektedir. Kodlarla ilgili bazı örnek alıntılar aşağıdaki gibidir:

*Teknoloji çağına uygunluk: “Dijital çağa geçtiğimiz için artık hayatımızın her alanında yazılım, internet ve teknoloji vardır. Bu yüzden yeni teknolojiye daha kolay adapte olmamızı sağlar” (ÖA1).*

*Matematiğin teknolojiye uyarlanması: “Teknoloji çağında bulunduğumuz için gelecek öğrenciler bilgisayar, telefon ve benzeri teknolojik araçları kullanmaya meyillidirler. Öğrencinin bu ilgisini teknoloji, oyun ve eğitici matematik programları ile birleştirerek öğrenci ilgisini çeker, derse motivasyonu artırır ve matematiği daha eğlenceli bir şekilde sunma imkânı sağlar” (ÖA2).*

*Teknoloji kullanımı becerisi/ yeterliliği artırma: “Bilgisayar ve teknoloji hayatımızın bir parçası haline geldiği için yazılım ve benzeri şeylere hâkim olmak dijital teknolojiye yatkınlığımızı geliştirir” (ÖA10).*

*Teknoloji kullanımının yaygınlaşması: “Dijital matematiğin çağımızla birlikte daha hâkim olması açısından bakış açımızı değiştirdi” (ÖA9)*

## Programlama ve Kodlama Öğrenimi

Teknoloji vurgusuna ek olarak öğretmen adaylarının %44'ünün (n= 11) kodlama ve %44'ünün programlama vurgusunu dile getirdiği (n= 11) ve %12 sinin (n=3) Algoritma ve programlama dersi sayesinde kolay yoldan yazılım öğrendikleri vurgusunu yaptıkları görülmüştür. Bu kategorideki kodlar Tablo 3. te gösterilmiştir.

**Tablo 3.** Programlama/Kodlama

| Kodlar                       | f         | %          | Katılımcı   |
|------------------------------|-----------|------------|---|
| Kodlama                      | 11        | 44         | ÖA1, ÖA2, ÖA3, ÖA4, ÖA6, ÖA7, ÖA8, ÖA10, ÖA11, ÖA13, ÖA14 |
| Programlama                  | 11        | 44         | ÖA1, ÖA2, ÖA3, ÖA4, ÖA6, ÖA7, ÖA8, Ö11, ÖA13, Ö14, Ö15    |
| Kolay yoldan yazılım öğrenme | 3         | 12         | ÖA8, ÖA10, ÖA13   |
| <b>Toplam:</b>               | <b>25</b> | <b>100</b> | <b>12</b>   |

Tablo 3 incelendiğinde programlama-kodlama kategorisinde toplam üç kod bulunduğu görülmektedir. Bu kodlar kodlama, programlama ve kolay yoldan yazılım öğrenme şeklindedir kodlara ilişkin örnek alıntılar aşağıdaki gibidir:

**Kodlama:** “Yazılım ve kodlama öğrendim. Kodlamanın günlük hayattaki tüm yazılımlarda olduğunu anladım” (ÖA1).

**Programlama:** “Yeni bir bilgisayar programlama dili öğrendim. İleride öğrencilerime yazılım konusunda kendilerini geliştirebilmeleri açısından bu programı öğreteceğim” (ÖA8). “Birçok uygulamada kullanabileceğim bir altyapı olan Java programlama dilini öğrendim” (ÖA4)

**Kodlama/Programlama:** “Algoritma ve programlama dersi sayesinde dersinde üst düzey yazılımlar öğreniyoruz matematiksel kazanımları cevap programına uygulama öğreniyoruz. Scratch ile çeşitli kodlama teknikleri öğreniyoruz” (ÖA10). “Kodlamanın ne kadar detaylı bir süreç olduğunu ve en temelinden başlayarak yavaş yavaş mantığının kavranması gerektiğini gören öğretmen adayları, bu yeteneklerini matematikte uygulamak isteyeceklerdir.” (ÖA11).

## Matematik ve Yazılım İlişkisinin Kurulması

Öğretmen adaylarının matematik dersi ile yazılım arasında ilişki kurdukları da gözlenmiştir. Katılımcıların %32.15 inin matematik ve programlamanın harmanlanabileceğini (n=9), %21.43 ünün yazılımın matematiğe ilgiyi artırdığını (n= 6), %17. 86 sınıfın matematiği programlama da görmeyen mümkün olduğunu (n=5)), %0.71 inin algoritma ve günlük yaşamın ilişkili olduğunu (n= 3), %10.71

inin matematiksel yazılımlar geliştirilebileceklerini (n= 3) ve bir kişinin ise örüntü ve yazılımın ilişkili olduğunu belirttikleri görülmüştür. İlgili kodlar Tablo 4'te gösterilmiştir.

**Tablo 4.** *Matematik ve Yazılım İlişkisi*

| Kodlar                                   | f         | %          | Katılımcı                                    |
|--|-----------|------------|--|
| Matematik ve programlama harmanlanabilir | 9         | 32,15      | ÖA2, ÖA4, ÖA5, ÖA6, ÖA8, Ö10, Ö13, Ö14, ÖA15 |
| Matematiğe ilgiyi arttırma               | 6         | 21,43      | ÖA2, ÖA4, ÖA7, ÖA9, ÖA13, ÖA14               |
| Matematiği programlamada görme           | 5         | 17,86      | ÖA3, ÖA4, ÖA6, ÖA8, ÖA10                     |
| Algoritma ve günlük yaşam                | 3         | 10,71      | ÖA1, ÖA9, ÖA12                               |
| Matematik yazılımları geliştirme         | 3         | 10,71      | ÖA4, ÖA8, ÖA11                               |
| Örüntü ve Yazılım                        | 1         | 3,57       | ÖA7  |
| İlgisi yok                               | 1         | 3,57       | ÖA15   |
| <b>Toplam:</b>                           | <b>28</b> | <b>100</b> | <b>15</b>                                    |

Tablo 4 incelendiğinde öğretmen adaylarının matematik ve yazılım arasında ilişki kurduğuna dair toplam yedi kod olduğu görülmektedir. Bu kodlardan bazıları matematik ve programlama harmanlanabilir, matematiğe ilgi artırılabilir, matematik programla da görülebilir, Algoritma ve günlük yaşam ilişkilidir şeklindedir. Bu kategoriye ait örnek alıntılar aşağıdaki gibidir:

*Matematik ve programlama harmanlanabilir:* “Öğretmen adaylarına bilgisayar kullanma ve program becerilerini kazandırır. Öğretmen adaylarının derste matematiği sevdirecek eğitici oyunlar tasarlamalarına yardımcı olur” (ÖA2).

*Matematiği programlamada görme:* “Programlama dilinde kodlar olduğundan kodların mantığını algılamak matematiği gerektirir. Böylelikle matematiğin dijitaldeki kullanımını sağlar” (ÖA3).

*Matematiğe ilgiyi arttırma:* “Algoritma da matematik gibi bir dildir ve bu diller birbirine benzerdir. Biz bu dilleri birbiriyle ilişkilendirirsek ve birbirine uyarlasak bizim açımızdan birçok fayda sağlayacaktır. Teknolojiye uyarlanan matematik öğrencilerin ilgisini daha çok çekecektir” (ÖA14)

*Matematik yazılımları geliştirme:* “Algoritma ve programlama dersinde öğrendiğimiz kodlama sayesinde matematik dersindeki bazı kazanımları programlama diline uygun bir şekilde döküp kodlamayı öğrendim (ÖA8).

## Algoritma ve Programlama Dersinin Öğrenim ve Öğretim Yöntemleri ile İlgili Farkındalık Arttırması

Öğretmen adayları ile yapılan görüşmede onların öğrenim ve öğretim yöntemlerinin vurgusunu yaptıkları görülmektedir. Örneğin %19.05'i yaparak yaşayarak öğrenme (n=4), %14.29'unun bilgisayar tabanlı öğrenme (n=3), %14.29'unun dikkat çekme (n=3), altı öğretmen adayının ise anlamlı öğrenme, işbirlikçi öğrenme gibi öğretim yöntemlerinden bahsettikleri görülmüştür. İlgili kodlar tablo 5'te gösterilmiştir.

**Tablo 5.** Programlama Öğrenim ve Öğretim Yöntemleri

| Kodlar                              | f         | %          | Katılımcı                 |
|-------------------------------------|-----------|------------|---------------------------|
| Farklı, etkili öğretim yöntemi      | 5         | 23,81      | ÖA2, ÖA3, ÖA6, ÖA13, ÖA14 |
| Yaparak yaşayarak öğrenme           | 4         | 19,05      | ÖA2, ÖA8, ÖA11, ÖA12      |
| Bilgisayar tabanlı öğrenme          | 3         | 14,29      | ÖA1, ÖA7, ÖA9             |
| Dikkat çekme                        | 3         | 14,29      | ÖA9, ÖA12, ÖA14           |
| Kalıcı öğrenme                      | 1         | 4,76       | ÖA4                       |
| Yaratıcı düşünme                    | 1         | 4,76       | ÖA3                       |
| Çoklu öğrenme ortamı                | 1         | 4,76       | ÖA1                       |
| Anlamlı öğrenme                     | 1         | 4,76       | ÖA5                       |
| İşbirlikli öğrenme                  | 1         | 4,76       | ÖA12                      |
| Örnek çeşitliliği ile konu öğretimi | 1         | 4,76       | ÖA3                       |
| <b>Toplam:</b>                      | <b>21</b> | <b>100</b> | <b>12</b>                 |

Tablo 5 incelendiğinde algoritma ve programlama dersi ile ilgili öğretmen adaylarının yorumlarında öğrenim ve öğretim yöntemlerine değindikleri ve bu kategoride toplam 10 kod olduğu görülmektedir. Bu kodlardan bazıları farklı ve etkili öğretim yöntemi, yaparak yaşayarak öğrenme, bilgisayar tabanlı öğrenme, dikkat çekme ve işbirlikçi öğrenme olduğu görülmektedir. Örnek alıntılar aşağıdaki gibidir:

*Farklı, etkili öğretim yöntemi:* “Öğretmen adaylarının derste kullanacağı farklı bir yöntem ve teknikle bu program kullanılabilir. Öğrenciye farklı bakış açısı kazandırabilir. Dijital bir araç kullanarak farklılık yaratabilir” (ÖA6).

*Bilgisayar tabanlı öğrenme:* Simülasyon hazırlama adına yaptığımız çalışmalar bilgisayar tabanlı, ders içi uygulamaları yapma konusunda fayda sağlamıştır” (ÖA9).

*Dikkat çekme:* “Gündelik yaşama uygun, öğrencilerin ilgilerini çekecek bir metot. Hem uygulama hem de öğretim sağlar. Etkinlikler ile öğrenciler arası bağları güçlendirir” (ÖA12.)

Çoklu öğrenme ortamı: “Öğretmenlerin yazılım araç gereçlerini kullanabiliyor olması derslerde çoklu duyu organlarına hitap edecek öğrenmeler sağlamaları açısından önemlidir” (ÖA1).

Yaratıcı düşünme: “Kazanımları teknolojiye uygun olarak yaptığımızda yaratıcı düşünmeyi gerçekleştirir” (ÖA3).

### Algoritma ve Programlama Dersinin Matematik Anlamaya Katkısı

Öğretmen adaylarının Algoritma ve Programlama dersinin matematik anlamaya katkısı olduğunu düşündükleri görülmektedir. %33,33 ünün bu yolla eğlenceli matematik (n=7), %25.57 sinin dijital çağda matematik (n=6), % 14.29 unun matematiksel işlemlerin kolaylaşması (n=3) ve % 9.53 ünün öğretimde verimlilik (n=2) vurgusu yapıldığı görülmektedir. Sadece bir öğrencinin matematik ile ders arasında hiçbir ilgi olmadığını belirttiği görülmektedir. İlgili kodlar Tablo 6’da verilmiştir.

**Tablo 6.** Dersin Matematik Anlamaya Katkısı

| Kodlar                               | f         | %          | Katılımcı                           |
|--------------------------------------|-----------|------------|-------------------------------------|
| Eğlenceli matematik                  | 7         | 33,33      | ÖA2, ÖA4, ÖA6, ÖA7, ÖA9, ÖA11, ÖA13 |
| Dijital çağda matematik              | 6         | 28,57      | ÖA1, ÖA2, ÖA3, ÖA6, ÖA7, ÖA10       |
| Matematiksel işlemlerin kolaylaşması | 3         | 14,29      | ÖA3, ÖA4, ÖA6                       |
| Öğretimde verimlilik                 | 2         | 9,53       | ÖA2, ÖA6                            |
| Matematiği somutlaştırma             | 1         | 4,76       | ÖA9                                 |
| Matematiği farklı alanlarda kullanma | 1         | 4,76       | ÖA13                                |
| Faydası yok                          | 1         | 4,76       | ÖA15                                |
| <b>Toplam:</b>                       | <b>21</b> | <b>100</b> | <b>12</b>                           |

Tablo 6’da öğretmen adayları ile yapılan görüşmelere göre belirlenen Algoritma ve Programlama dersinin matematik anlamaya ne derece katkısı olduğunu gösteren toplam yedi kod olduğu görülmektedir. Bu kodlardan bazıları, eğlenceli matematik, dijital çağda matematik, matematiksel işlemlerin kolaylaşması, öğretimde verimlilik, matematiği somutlaştırma gibi kodlar olduğu görülmektedir. Bazı örnek alıntılar aşağıdaki gibidir:

Eğlenceli matematik: “Matematiğe olan ilgiyi arttırabilir. Matematiği daha eğlenceli yapabilir. Matematik farklı alanlarda kullanılabilir” (ÖA13).

Matematiksel işlemlerin kolaylaşması: “Java programında yazdığımız kodlar ile oluşturduğumuz açılır pencere ile sayı girip işlem yapıyoruz ortalama hesaplama üs alma vs. gibi birçok matematik işleminin sonucunu yazdığımız kodlarla bulabiliyoruz” (ÖA4).

Dijital çağda matematik: “Dijital dünyada olduğumuz için matematik de artık bunun içine girmektedir Algoritma ve programlama dersinde dijital matematik kullanırız kodlamalar yaparız ve matematik dilini orada görebiliriz” (ÖA6). “Teknoloji geliştikçe öğretim hayatımı da etkilemiştir. Bu sebepten dolayı derslerde de kullanılan teknoloji artmıştır. Bu yüzden bu ders hem matematik hem de diğer bütün dersler için önem arz etmiştir” (ÖA7).

Matematiği somutlaştırma: “Matematik dersinin daha ilgi çekici olması açısından çalışmalar yapılabilmesi, somutlaştırma konusunda daha etkili olabileceğini düşündürdü” (ÖA9).

## Algoritma ve Programlama Dersinin Müfredatta Yer Alma Durumu

Öğretmen adaylarının Algoritma ve programlama dersinin müfredatta yer almasını değerlendirmeleri sorulduğunda büyük bir çoğunluğunun buna bir gerekçe göstererek yer alması gerektiğini bildirdikleri görülmüştür. % 33.34 ünün dersin katkısı olduğunu (n=5), % 20 sinin farklı bakış açısı kazandırdığını (n=3), %13.33 ünün yeni neslin ilgisini çekeceğini (n=2), %13.33 ünün bu dersin daha önceden verilmesi gerektiğini (n=2) ve % 6,67 sinin ise öğretmenlerin yazılım bilmeleri gerektiğini (n=1) belirttikleri görülmüştür. Bu sebeplerle de dersin müfredatta yer alması gerektiğini bildirdikleri görülmüştür. Bu kategorideki ilgili kodlar tablo 7 de gösterilmiştir.

**Tablo 7.** Dersin Müfredatta Yer Alma Durumu

| Kodlar                              | f         | %          | Katılımcı                |
|-------------------------------------|-----------|------------|--------------------------|
| Katkı sağlar, faydalı               | 5         | 33,34      | ÖA6, ÖA7, ÖA8, ÖA9, ÖA11 |
| Derse farklı bakış açısı kazandırır | 3         | 20         | ÖA4, ÖA9, ÖA13           |
| Yeni neslin ilgisini çeker          | 2         | 13,33      | ÖA12, ÖA14               |
| Daha önce olmalıydı                 | 2         | 13,33      | ÖA1, ÖA12                |
| Geleceğin dersi                     | 1         | 6,67       | ÖA3                      |
| Öğretmen yazılım/bilgisayar bilmeli | 1         | 6,67       | ÖA10                     |
| Hayır, yer almasın                  | 1         | 6,67       | ÖA15                     |
| <b>Toplam:</b>                      | <b>15</b> | <b>100</b> | <b>13</b>                |

Tablo 7 incelendiğinde dersin müfredatta yer alma durumu ile ilgili toplam yedi kod olduğu görülmektedir. Bu kodlardan bazıları, katkı sağlayıp faydalı ola-

bileceği, derse farklı bakış açısı kazandırabileceği, yeni neslin ilgisini çekebileceği ve geleceğin dersi olmasıdır. Öğretmen adaylarının yorumlarından örnek alıntılar aşağıdaki gibidir:

*Katkı sağlar, faydalı:* “Bence benzer dersler okutulmadan önce üniversite, ortaokul ve tüm okulların sağlam bir altyapıya sahip olması için çalışılmalıdır. Ondan sonra benzer dersler artırılabilir ve matematik öğretimine katkıda bulunabilir” (ÖA6). “Kodlama mantığını kavrayan birisi daha analitik düşünebilir. Analitik düşünme de matematiğin temelinde yatan bir unsur olduğu için bu dersin programda yer alması matematik öğretmenleri için faydalı olacaktır” (ÖA11).

*Daha önce olmalıydı:* “Programın üniversite düzeyinde verilmesinin öğretmen adaylarının bakış açılarının değişmesi için geç olduğunu düşünüyorum” (ÖA12).

*Derse farklı bakış açısı kazandırır:* “Matematik dersine yeni bir bakış açısı getirebilir. Farklı yollarla anlatım ya da oyunlaştırma ile matematik öğrencilere daha çekici gelebilir” (ÖA13).

### Algoritma ve Programlama Dersinin Matematik Öğretmen Adayları Açısından Kişisel Avantajları

Öğretmen adaylarının birtakım kişisel avantajlardan da söz ettikleri görülmüştür. Örneğin bilgisayar kullanım becerisinin artması (n=10), ileride farklı bir meslek sahibi olabilme imkânı (n=8), matematik ve programlama arasında daha iyi ilişki kurabilecekleri (n=3), programlama sayesinde İngilizce kavramlar öğrendikleri (n=3), ileride daha zengin içerikli ve etkili dersler verebilecekleri (n=2) gibi vurgular yaptıkları görülmektedir. Aynı zamanda kendilerine analitik düşünme, plan yapma gibi becerileri kazandırdığını ve özgüven kazandıklarını belirttikleri görülmektedir. İlgili kodlar tablo 8’de verilmiştir.

Tablo 8. Kişisel Avantajları

| Kodlar                                       | f  | %     | Katılımcı   |
|--|----|-------|---|
| Bilgisayar kullanım becerisi                 | 10 | 32,26 | ÖA1, ÖA2, ÖA5, ÖA6, ÖA7, ÖA8, ÖA9, ÖA10, ÖA12, ÖA13 |
| Yazılımcı olma/farklı bir meslek sahibi olma | 8  | 25,80 | ÖA1, ÖA3, ÖA6, ÖA7, ÖA11, ÖA13, ÖA14, ÖA15          |
| Matematik ve programlama ilişkisi kurma      | 3  | 9,68  | ÖA5, ÖA8, ÖA14                                      |
| İngilizce kavramlar öğrenme                  | 3  | 9,68  | ÖA1, ÖA2, ÖA6                                       |
| Zengin içerikli/etkili dersler               | 2  | 6,45  | ÖA12, ÖA13  |
| Matematik dili                               | 2  | 6,45  | ÖA3, ÖA6  |



|                   |           |            |           |
|-------------------|-----------|------------|-----------|
| Analistik düşünme | 1         | 3,23       | ÖA11      |
| Planlama yapma    | 1         | 3,23       | ÖA12      |
| Özgüven kazanma   | 1         | 3,23       | ÖA12      |
| <b>Toplam:</b>    | <b>31</b> | <b>100</b> | <b>13</b> |

Tablo 8 incelendiğinde kişisel avantajlar kategorisinde toplam 10 kod olduğu görülmektedir. Bu kodlardan bazıları bilgisayar kullanımı becerisi, farklı bir meslek sahibi olma, matematik ve programlama ilişkisi kurabilme, İngilizce kavramlar öğrenme ve analitik düşünme gibidir. Öğretmen adaylarının görüşlerinden elde edilen örnek alıntılar aşağıdaki gibidir:

*Yazılımcı olma/farklı bir meslek sahibi olma:* “Programlamayı öğretir, teknoloji kullanımını geliştirir, geleceğe temel sağlar” (ÖA3). “Teknoloji çağında yazılım öğrenmek. Bilgisayar kullanımını artırması, yazılım ve programlama öğrenme, ek iş, gelecek mesleği” (ÖA7). “Diğer matematik öğretmenlerinden farklı olarak yazılım biliyor olmamız meslek olarak öğretmenlik değil de yazılım falan yapmayı düşünürsek iyi bir avantaj” (ÖA14)

*İngilizce kavramlar öğrenme:* “Algoritma ve programlama dersinde matematik kazanımları ile konuya ilişkin sorularla kod yazmayı öğrendik. Kodlama becerilerimiz arttı. Java programında İngilizce dilini kullandığımız için İngilizcemiz gelişti. Programda hesap makinesi yaparak matematiği kolayca eğlenceli olarak anlamamızı sağladı” (ÖA6).

*Analitik düşünme:* “Matematik analitik düşünmeyi gerektiren bir ders olduğu için ve Algoritma ve Programlama dersi de bunu geliştirdiği için katkısı olduğunu düşünüyorum” (ÖA11).

### Algoritma ve Programlama Dersinin Matematik Öğretmen Adayları Açısından Dezavantajları

Olumlu görüşlere ek olarak öğretmen adaylarının dersin bazı dezavantajlarına değindikleri görülmektedir. % 58.82 sinin bilgisayar sahibi olmayanlar için bir dezavantaj olabileceği (n=10), % 23,53 ünün sürekli bilgisayar karşısında oturmanın bazı sağlık problemlerine yol açabileceğini (n=4), % 5.88 inin programlama öğrenmenin sabır gerektirdiğini (n=1), % 5.88 inin yorucu ve zaman alıcı olduğunu (n=1) ve bir öğretmen adayının ise öğretmenler için gereksiz bir ders olduğunu vurguladığı görülmüştür. Dezavantajlar ile ilgili kodlar Tablo 9’da verilmiştir.

Tablo 9. Dezavantajları

| Kodlar   | f         | %          | Katılımcı  |
|--|-----------|------------|--|
| Bilgisayar sahibi olmayanlar                           | 10        | 58,82      | ÖA1, ÖA2, ÖA3, ÖA4, ÖA5, ÖA6, ÖA8, ÖA9, ÖA12, ÖA14 |
| Sürekli bilgisayar karşısında olmak-sağlık problemleri | 4         | 23,53      | ÖA3, ÖA6, ÖA7, ÖA13                                |
| Sabır  | 1         | 5,88       | ÖA10   |
| Yorucu-Zaman alıcı                                     | 1         | 5,88       | ÖA11   |
| Öğretmenler için gereksiz                              | 1         | 5,88       | ÖA15   |
| <b>Toplam: 5</b>                                       | <b>17</b> | <b>100</b> | <b>15</b>  |

Tablo 9 incelendiğinde dezavantajlar kategorisinde toplam 5 kod olduğu görülmektedir. Görüşlerden elde edilen kodlar, bilgisayar sahibi olmayanlar, sağlık problemleri, sabır, yorucu ve zaman alıcı ve öğretmenler için gereksiz oluşudur. Bazı örnek alıntılar aşağıda verilmiştir:

Sürekli bilgisayar karşısında olmak-sağlık problemleri: “Bilgisayarı olmayan öğrenciler için zor olabilir. Çok fazla bilgisayara bakıldığı için göz bozukluğu sırt ağrısı olabilir” (ÖA6).

Bilgisayar sahibi olmayanlar: “Bilgisayarı olmayan kişiler derse etkin katılım sağlayamadı ve bu da onlar için dezavantaj oldu” (ÖA8). “Sınıfta uygulamalı olarak öğrendiğimiz için herkeste bilgisayar olması gerekir ama herkesin bilgisayarı yok” (ÖA14)

Sabır: “Yazılımla uğraşmak sabırlı olmamızı gerektirir. Küçük bir hata yapsak bile hata veriyor” (ÖA10).

Öğretmenler için gereksiz: “Şu an bizim için yani öğretmenler için bir şey ifade etmiyor” (ÖA15).

Elde edilen bulgulardan hareketle, teknoloji, programlama ve kodlama, matematik ve programlama ilişkisi, programlama öğrenme ve öğretim yöntemleri, dersin matematik anlamaya katkısı, dersin müfredatta bulunma durumu, avantajlar ve dezavantajlar şeklinde kategoriler elde edilmiştir. Bu bulgular tartışma bölümünde literatürden hareketle detaylı olarak yorumlanmıştır.

## TARTIŞMA

Bu çalışmanın amacı, matematik öğretmen adaylarının Algoritma ve Programlama dersi hakkındaki görüşlerini ortaya koymaktadır. Algoritma ve Programlama dersi 2018-2019 eğitim-öğretim yılından itibaren ilköğretim matematik öğretmenliği programında okutulmaktadır. Bu dersin programlama ile doğrudan ilgili olmayan bir bölümde verildiği düşünüldüğünde, dersin kazanımlarının matematik öğretmeni adayları için uygun olup olmadığını sorgulamak bir gereksinim halini almıştır. Bu nedenle bu dersi öğretmen adaylarının bakış açısıyla değerlendirmek için yarı yapılandırılmış görüşmeler yapılmış ve elde edilen bulgular literatüre dayalı olarak değerlendirilmiştir.

Öncelikle öğretmen adaylarının hemen hepsinin derse karşı olumlu görüşler belirttiği ve dersin kazanımlarına sıkça vurgu yaptıkları görülmüştür. Bu bulgunun sebebi öğretmen adaylarının matematikle ilgili bilişsel becerilere sahip olmaları olabilir. Çünkü yapılan araştırmalara göre programlama becerisinin matematikte kullanılan benzer bilişsel beceri ve deneyim gerektirdiği (Korkmaz, 2012; Ayalew, Tshukudu ve Lefoanea, 2018) bilinmektedir. Ek olarak mantıksal ve matematiksel düşünmenin programlama öğrenmeyi kolaylaştırdığı (Cevahir ve Özdemir, 2017) ve programlama ile matematik öğrenmenin birbiriyle ilişkili olduğu (Kong ve Kwok, 2022) ifade edilmektedir. Dolayısıyla ilköğretim matematik öğretmenliği programı bünyesinde Algoritma ve Programlama dersi için hedeflenen kazanımların program bünyesinde eğitim gören öğrenciler için uygun olduğu sonucuna varılabilir.

Öğretmen adaylarının dersin kazanımları ile ilgili teknoloji vurgusunu çokça yaptıkları görülmüştür. Dersin teknoloji çağına uygunluğu, teknoloji kullanım becerisini ve teknolojik yeterliliği artırması ve aynı zamanda matematik dersinin teknolojiye uyarlanması gibi görüşler ortaya çıkmıştır. Bu bulgular onların teknoloji okuryazarlığı becerilerinin arttığı bir göstergesi olarak gösterilebilir. Programlama becerisi, aslında teknoloji okuryazarlığı olarak da adlandırılmıştır (Yıldız, Çiftçi ve Karal, 2017). Ayrıca, bireylere programlama becerisi öğretildiğinde onların dijital okuryazarlığının geliştirilebileceği düşünülmektedir (Akpınar ve Altun, 2014; Durak ve Şahin, 2018). Bu bağlamda, öğretmen adaylarının programlama öğrenme sürecinde, teknoloji okuryazarlığı becerisi kazandıklarını belirtmeleri önemlidir.

Öğretmen adaylarının kodlama ve programlamayı öğrenmeye istekli olmaları, hatta lisans eğitimi sürecinde bunu kolaylıkla öğrenmek istemeleri, onların 21. yüzyıl becerilerini kazanmaya açık olduklarını gösteren bir işaret olarak değerlendirilebilir. Kodlama, yazmanın yeni bir formu, matematiksel düşünmenin, üretmenin ve problem çözmeyi desteklemenin bir yolu, 21. yüzyıl becerisi ve mantıksal akıl yürütmenin bir parçası olduğu kabul edilmektedir (Sayın ve Seferoğlu,

2016). Yazılım, öğrencilere hayal etme, keşfetme, analiz yapma, varsayımda bulunma, genelleme yeteneklerini artırarak, öğrenenlerin problem çözme becerisine katkıda bulunduğu savunulmaktadır. (Baki, 2001). Programlama öğretimi uygulamaları ile öğrencilerin yaşam boyu öğrenme yeterliliklerini artırdığı görülmektedir (Durak ve Şahin, 2018). Dolayısıyla, öğretmen adaylarının programlama sayesinde matematiksel düşünme, problem çözme, analiz etme ve varsayımlarda bulunabilme gibi 21. yüzyıl becerileri kazanmalarının bir yolu olarak programlama eğitimi almaları gerektiği ve buna istekli oldukları söylenebilir.

Öğretmen adayları sırasıyla algoritma, blok tabanlı görsel programlama ve programlama şeklinde eğitime tabi tutulmuşlardır. Özellikle algoritmanın asıl amacının programlama mantığını kavratmak olduğu (Gökoğlu, 2017) ve programlama mantığının kavratılmasının önem arz ettiği (Arabacıoğlu, Bülbül ve Filiz, 2007) bilinmektedir. Programlamanın mantığının kavranması, doğal olarak programlamanın problem çözme ve analitik düşünme gibi bahsi geçen becerileri kazandırmasına da (Akpınar ve Altun, 2014) imkân sağlayacaktır. Öğretmen adayları programlama mantığının öğrenilmesi gerektiğini vurgulayarak programlamanın analitik düşünme ve matematikle ilgisi olduğunu ifade etmeleri, programlama mantığını kavradıklarının ve dolayısıyla bahsi geçen becerileri kazandıklarının bir göstergesi olabilir.

Programlamada matematiksel mantık yer almaktadır. Öğretmen adayları matematiksel yazılımlar geliştirerek matematik ile programlamanın harmanlanabileceğini, matematiğin zaten programlama içerisinde yer aldığını ve böylece matematiğe olan ilginin artabileceğinden bahsetmişlerdir. Ayrıca programlama eğlenceli bir şekilde matematik öğretimini gerçekleştirme ve özellikle dijital çağda dijital matematik anlayışına katkıda bulunma gibi yararlıdır. Yapılan araştırmalarda, programlama öğrenmenin öğrencilerin matematiği anlamalarını kolaylaştırdığını ve matematik öğrenmeye karşı onları motive ettiğini (Barak ve Assal, 2014) ve öğrencilerin matematik ve teknolojik yeterliliklerini artırdığını (Leonard vd., 2016) göstermektedir. Buna göre öğretmen adaylarının programlamanın alanyazında savunulan avantajlar ile ilgili farkındalık kazandıkları ve bilhassa matematik yönünden geliştikleri söylenebilir.

Ders süresince öğretmen adaylarına ders kapsamında verilen konular önerildiği üzere (Arabacıoğlu, Bülbül ve Filiz, 2007; Akıncı ve Dübüş, 2022) uygulamalı olarak verilmiş ve çeşitli ödevlerle desteklenmiştir. Öğretmen adaylarının farklı öğretim yöntemlerinin vurgusunu da ders sayesinde yaptıkları görülmüştür. Bu tür bir dersin verilmesinin farklı öğretim yöntemlerini destekleyeceğini özellikle yaparak yaşayarak öğrenme gibi etkili öğretim yöntemlerine değindikleri görülmüştür. Bu bağlamda dersin uygulamalı olarak verilmesinin öğretmen adaylarına öğretim yöntemleri ile ilgili de farkındalıklarını artırdığı sonucu çıkarılabilir. Sadece teknoloji, yazılım gibi ders ile doğrudan ilgisi olan kavramlara ek olarak öğretmen-

lik becerilerinden öğretim yöntem ve ilkelerini kavratma yolunda da algoritma ve programlama dersinin faydası olacağı söylenebilir.

Öğretmen adaylarının dersin ilköğretim matematik eğitimi programı lisans müfredatında yer alarak bireylere katkıda bulunacağını, farklı bakış açısı edinmelerine yardımcı olacağı, yaratıcı düşünme, problem çözme, analitik düşünme becerilerine katkıda bulunduğunu ve yaşadığımız dijital çağda öğrencilerin ilgisini çekebileceğini belirttikleri görülmektedir. Program yazabilmenin, verilen probleme farklı perspektiflerden bakabilme, en kısa çözümü bulabilme, olaylar arasındaki ilişkiyi görebilme, sistematik düşünme ve yaratıcı düşünme gibi bilişsel becerileri geliştirdiği bilinmektedir (Fesakis ve Serafeim, 2009; Göksoy ve Yılmaz, 2018). Bireylerin, dijital yeterlilikleri, problem çözme, analitik düşünme, uzamsal düşünme, proje oluşturma, çözüm üretme, işbirlikli çalışma ve yaparak yaşayarak öğrenme becerilerini kazanmaları için programlama öğrenmeleri önerilmektedir (Akçınar & Altun, 2014). Buradan hareketle öğretmen adaylarının bahsedilen bu yararların vurgusunu kendi öğrenme süreçlerinden yola çıkarak belirtmelerinin literatürü destekler nitelikte olduğu görülmektedir.

Kişisel olarak özellikle bilgisayar kullanım becerilerinin arttığını, matematik ile programlama ilişkisi kurabildiklerini, ileride farklı bir meslek sahibi olma yolunda yeni bir beceri edindiklerini ve hatta İngilizce kavramlar öğrendiklerini belirtmişlerdir. Alanyazında programlama dillerinin İngilizce olmasının programlama öğrenmede yaşanan en büyük zorluklardan biri olduğu (Arabacıoğlu, Bül-bül ve Filiz, 2007) ve öğrenenlerin yabancı dil düzeylerinin programlama ile ilgili öz yeterliklerine anlamlı bir etkisi olduğu (Akçay ve Çoklar, 2018) gösterilmiştir. Bu çalışmada ise öğretmen adaylarının dersin katkısı olarak İngilizce kavramlar öğrendikleri vurgusunu yaptıkları görülmüştür. Bu bulgunun dersin uygulamalı olarak verilmesinin ve ders sürecinde algoritma tasarımı, blok tabanlı programlama ve programlama dili öğrenimi gibi bir sıralamada öğrenmelerinin bir sonucu olduğu düşünülebilir.

Dersin dezavantajları olarak öğretmen adayları bilgisayar sahibi olmayanlar için bir sorun teşkil ettiğini, çok uzun süre bilgisayar karşısında oturma'nın sırt ağrısı gibi birtakım problemlere yol açabileceğini, sabır isteyen bir süreç olduğunu belirtmişlerdir. Katılımcılardan sadece biri programlamaya yönelik olumsuz görüşler bildirmiş ve dersin gereksizliğinden bahsetmiştir. Programlama öğrenmenin matematiksel düşünme (Cevahir ve Özdemir, 2017), motivasyon (Yılmaz ve Çakır, 2019), öz yeterlik inancı ve tutum ile pozitif yönlü olarak ilişkili olduğu (Özyurt ve Özyurt) ele alınırsa, olumsuz düşünen öğrencinin bu unsurlardan bazılarını sahip olma durumu olasıdır. Ayrıca programlamaya yönelik olumlu tutumun (Şahin, Korkmaz, Çakır ve Erdoğan, 2019) programlamayı öğrenme isteği ve bunun profesyonel yaşamda faydalı olacağına dair inançla olumlu bir ilişkisi olduğu kabul edilmektedir (Benli ve Tek, 2021). Ayrıca motivasyon eksikliğinin de

programlama öğrenmeye olumsuz etkisi olduğu bildirilmiştir (Saygıner ve Tüzün, 2017). Dolayısıyla olumsuz yorumlarda bulunan öğretmen adayının programlamaya olan tutumu alanyazında bahsi geçen sebeplerle ilgili olabilir.

## SONUÇ VE ÖNERİLER

Öğretmen adaylarının genellikle Algoritma ve Programlama dersi ile ilgili olumlu görüş bildirmeleri, literatürle desteklenir nitelikte olup ilköğretim matematik öğretmenliği programı bünyesinde verilen uygun bir ders olduğu sonucunu doğurmaktadır. Araştırmanın bulgularına göre, dersin matematik öğretmenliği müfredatına uygun olduğu sonucuna ulaşılabilir. Öğretmen adaylarının görüşlerine göre ise dijital yeterliliklerinin arttığı ve 21. yüzyıl becerilerinden programlama yeteneği kazandıkları sonucu çıkarılabilir. Ayrıca, öğrencilerin programlama mantığını kavradığı ve bu sayede matematikle programlama arasında bir ilişki gördükleri belirlenmiştir. Öğretmen adayları, derste öğrenme süreçlerinden ilham alarak öğretim yöntemlerine vurgu yapmış, ayrıca analitik ve yaratıcı düşünme gibi düşünme becerilerinin arttığını ve farklı bir mesleki deneyim kazandığını düşünmektedir.

Bu çalışma bir devlet üniversitesinin ilköğretim matematik öğretmenliği programında Algoritma ve Programlama dersi alan ve dersi yukarıda değinilen ölçütlerde tamamlayan 15 öğretmen adayı ile sınırlıdır. Öğretmen adayları dersin yürütücüsü tarafından algoritma kavramı, blok tabanlı görsel programlama ve Java programlama dili sıralamasıyla ders içeriklerini öğrenmişlerdir. Çalışmanın farklı bir programlama diliyle, farklı bir içerikle sunulmasının nasıl sonuçlanacağı bundan sonra yürütülecek olan çalışmalarda sorgulanabilir. Aynı zamanda öğretmen adaylarının programlama becerisiyle de yakından ilişkili olabilecek mevcut matematik düzeyleri farklı bir üniversitedeki öğretmen adaylarında farklı bir düzeyde olabilir. Bu sebeple benzer çalışmaların farklı öğretmen adaylarıyla yürütülmesi sonuçların genellenebilirliği açısından önem arz etmektedir.

Bu sonuçlar doğrultusunda araştırmanın bulgularına dayanarak aşağıdaki öneriler yapılabilir:

- Dersin uygulamalı bir şekilde verilmesi,
- Dersin algoritma kavramı, görsel blok-tabanlı programlama uygulamaları ve programlama dili sırasıyla yürütülmesi,
- Dersin kazanımlarının yeterince edinilmesi için bilgisayar sahibi olmayan öğrenciler için fakültelerin alt yapısının uygun hale getirilmesi önerilmektedir.

## TEŞEKKÜR VE AÇIKLAMALAR

Çalışmamıza katkı sağlayan tüm öğrencilere teşekkür ederiz.

### Çıkar Çatışması

Makalenin yazarları arasında, çalışma kapsamında herhangi bir kişisel ve finansal çıkar çatışması bulunmamaktadır.

### Yazar Katkı Oranları

Çalışmanın Tasarlanması (Design of Study): NG(%50), Şİ(%50)

Veri Toplanması (Data Acquisition): NG(%50), Şİ(%50)

Veri Analizi (Data Analysis): NG(%50), Şİ(%50)

Makalenin Yazımı (Writing Up): NG(%50), Şİ(%50)

Makalenin Gönderimi ve Revizyonu (Submission and Revision): NG(%50), Şİ(%50)

### Etik Kurul İzin Bilgileri

Yapılan bu çalışmada “Yükseköğretim Kurumları Bilimsel Araştırma ve Yayın Etiği Yönergesi” kapsamında uyulması belirtilen tüm kurallara uyulmuştur.

Etik Değerlendirmeyi Yapan Kurul Adı: Kafkas Üniversitesi Sosyal ve Beşerî Bilimler Bilimsel Araştırma ve Yayın Etiği Kurulu

Etik Değerlendirme Kararının Tarihi: 17.11.2022

Etik Değerlendirme Belgesi Sayı Numarası: E-37673- Sayı: 39

## KAYNAKLAR

- Agbo, F. J., Oyelere, S. S., Suhonen, J., & Adewumi, S. (2019, Kasım). A systematic review of computational thinking approach for programming education in higher education institutions. *Proceedings of the 19th Koli Calling International Conference on Computing Education Research* (pp. 1-10)
- Akçay, A., & Çoklar, A. N. (2018). Bilişim teknolojileri ve yazılım dersi öğretmen adaylarının programlamaya ilişkin algılanan öz yeterliklerinin farklı değişkenler açısından incelenmesi. *Kastamonu Eğitim Dergisi*, 26(6), 2163-2176.
- Akinci, M., & Dübüş, M. (2022). Ortaokul matematik öğretmenlerinin ilköğretim matematik öğretmenliği lisans programı hakkındaki görüşleri. *Karaelmas Eğitim Bilimleri Dergisi*, 10(2022), 1-14.
- Akkaya, A. ve Öztürk, G. (2020). Algoritma yazma ve öğrenimi hakkında meslek yüksekokulu öğrencilerinin görüşleri. *Balıkesir Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, 22(1), 367-380.
- Akpınar, Y., & Altun, A. (2014). Bilgi toplumu okullarında programlama eğitimi gereksinimi. *Elementary Education Online*, 13(1), 1-4.
- Alsancak Sırakaya, D. (2019). Programlama öğretiminin bilgi işlemsel düşünme becerisine etkisi. *Türkiye Sosyal Araştırmalar Dergisi*, 23(2), 575-590.
- Arabacıoğlu, T., Bülbül, H. İ., & Filiz, A. (2007). Bilgisayar programlama öğretiminde yeni bir yaklaşım. *Akademik Bilişim*, 2007, 193-197.
- Arslan, K. ve Akçelik, M. (2019). Programlama eğitiminde Scratch'ın kullanılması: Öğretmen adaylarının tutum ve algıları. *Ulusal Eğitim Akademisi Dergisi*, 3(1), 41-61.
- Ayalew, Y., Tshukudu, E., & Lefoanea, M. (2018). Factors affecting programming performance of first year students at a University in Botswana. *African Journal of Research in Mathematics, Science and Technology Education*, 22(3), 363-373.
- Baki, A. (2001). Bilişim teknolojisi ışığında matematik eğitiminin değerlendirilmesi. *Milli eğitim dergisi*, 149(1), 26-31.
- Barak, M., & Assal, M. (2018). Robotics and STEM learning: students' achievements in assignments according to the P3 Task Taxonomy—practice, problem solving, and projects. *International Journal of Technology and Design Education*, 28, 121-144.
- Benli, K. S., & Tek, F. B. (2021). Programlamaya giriş dersini alan öğrencilerin programlama öz yeterlilik algılarının ve programlamaya bakış açılarının incelenmesi. *Düzce Üniversitesi Bilim ve Teknoloji Dergisi*, 9(3), 328-347.
- Cevahir, H., & Özdemir, M. (2017). Programlama öğretiminde karşılaşılan zorluklara yönelik öğretmen görüşleri ve çözüm önerileri. *Uluslararası Bilgisayar ve Öğretim Teknolojileri Sempozyumu* içinde, İnönü Üniversitesi, Malatya.
- Cline, K., Fasteen, J., Francis, A., Sullivan, E., & Wendt, T. (2020). Integrating programming across the undergraduate mathematics curriculum. *Primus*, 30(7), 735-749.
- Çatlak, Ş., Tekdal, M., & Baz, F. Ç. (2015). Scratch yazılımı ile programlama öğretiminin durumu: Bir doküman inceleme çalışması. *Journal of Instructional Technologies and Teacher Education*, 4(3), 13-25.
- Durak, H. Y., & Şahin, Z. (2018). Kodlama eğitiminin öğretmen adaylarının yaşam boyu öğrenme yeterliliklerinin gelişmesine katkısının incelenmesi. *Ege Eğitim Teknolojileri Dergisi*, 2(2), 55-67.
- Fesakis, G., & Serafeim, K. (2009). Influence of the familiarization with "scratch" on future teachers' opinions and attitudes about programming and ICT in education. *Acm SIGCSE Bulletin*, 41(3), 258-262.
- Gökoğlu, S. (2017). Programlama eğitiminde algoritma algısı: Bir metafor analizi. *Cumhuriyet International Journal of Education*, C1JE, 6(1), 1-14.
- Göksoy, S. ve Yılmaz, İ. (2018). Bilişim teknolojileri öğretmenleri ve öğrencilerinin robotik ve kodlama dersine ilişkin görüşleri. *Düzce Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, 8(1), 178-196.
- Göncü, A., Çetin, İ. ve Şendurur, P. (2020). Bilişim teknolojileri ve yazılım dersi öğretmenlerinin kodlama eğitimine yönelik görüşleri. *Mersin Üniversitesi Eğitim Fakültesi Dergisi*, 16(2), 301-321.
- Kong, S. C., & Kwok, W. Y. (2022). From mathematical thinking to computational thinking: Use scratch programming to teach concepts of prime and composite numbers. *Proceedings of 29th International Conference on Computers in Education Conference* içinde. *Asia-Pacific Society for Computers in Education* (s. 549-558).
- Korkmaz, Ö. (2012). The impact of critical thinking and logico-mathematical intelligence on algorithmic design skills. *Journal of Educational Computing Research*, 46(2), 173-193.
- Leonard, J., Buss, A., Gamboa, R., Mitchell, M., Fashola, O. S., Hubert, T., & Almughyrah, S. (2016). Using robotics and game design to enhance children's self-efficacy, STEM attitudes, and computational thinking skills. *Journal of Science Education and Technology*, 25, 860-876.
- McMillan, J. H. (2012). *Educational Research Fundamentals for the Consumer* (6. Baskı) Boston: Pearson.
- Miles, M. B., & Huberman, A. M. (1994). *An Expanded Sourcebook Qualitative Data Analysis* (2nd Ed.). London: Sage Publications.



- Oh, M. J. (2017). Non-Major Students' Perceptions of Programming Education Using the Scratch Programming Language. *The Journal of Korean Association of Computer Education*, 20(1), 1-11.
- Ouahbi, I., Kaddari, F., Darhmaoui, H., Elachqar, A., & Lahmine, S. (2015). Learning basic programming concepts by creating games with scratch programming environment. *Procedia-Social and Behavioral Sciences*, 191, 1479-1482.
- Öztürk, S. & Alper, A. (2019). Programlama öğretimindeki ters-yüz öğretim yönteminin öğrencilerin başarılarına, bilgisayara yönelik tutumuna ve kendi kendine öğrenme düzeylerine etkisi. *Bilim, Eğitim, Sanat ve Teknoloji Dergisi (BEST Dergi)*, 3(1), 13-26.
- Özyurt, Ö., & Özyurt, H. (2015). Bilgisayar programcılığı öğrencilerinin programlamaya karşı tutum ve programlama öz-yeterliliklerinin belirlenmesine yönelik bir çalışma. *Eğitimde Kuram ve Uygulama*, 11(1), 51-67.
- Pala, F. K. ve Mihçı Türker, P. (2018). Öğretmen adaylarının programlama eğitimine yönelik görüşleri. *Kuramsal Eğitimbilim Dergisi*, 12(1), 116-134.
- Pinto, A., & Escudeiro, P. (2014). The use of Scratch for the development of 21st century learning skills in ICT. *In Information Systems and Technologies (CISTI), 2014 9th Iberian Conference on IEEE*, 1-4.
- Saygıner, Ş., & Tüzün, H. (2017). Programlama eğitiminde yaşanan zorluklar ve çözüm önerileri. *Uluslararası Bilgisayar ve Öğretim Teknolojileri Sempozyumu*, İnönü Üniversitesi, 78-90.
- Sayın, Z., & Seferoğlu, S. S. (2016). Yeni bir 21. yüzyıl becerisi olarak kodlama eğitimi ve kodlamanın eğitim politikalarına etkisi. *Akademik Bilişim Konferansı*, 3(5), 1-13.
- Sebetci, Ö. & Aksu, G. (2014). Öğrencilerin mantıksal ve analitik düşünme becerilerinin programlama dilleri başarısına etkisi. *Eğitim Bilimleri ve Uygulama*, 13 (25), 65-83
- Shin, S., & Park, P. (2014). A study on the effect affecting problem solving ability of primary students through the scratch programming. *Advanced Science and Technology Letters*. 59,117-120. <https://doi.org/10.14257/astl.2014.59.27>
- Şahin, H., Korkmaz, Ö., Çakır, R., & Erdoğmuş, F. U. (2019). Bilişim teknolojileri öğretmenlerinin kodlamaya dönük tutumları, öz-yeterlilikleri ve kodlama öğretimi için kullanıkları yöntemler. *Ondokuz Mayıs University Journal of Education Faculty*, 38(2), 1-16.
- Taylor, M., Harlow, A., & Forret, M. (2010). Using a computer programming environment and an interactive whiteboard to investigate some mathematical thinking. *Procedia-Social and Behavioral Sciences*, 8, 561-570. [Çevrim-içi:10.1016/j.sbspro.2010.12.078, Erişim tarihi: 04.10.2022].
- Uzun A. ve Baltalı, S. (2020). Programlama öğretiminde kullanılabilecek yazılımlara ilişkin öğretmen görüşleri. *Uludağ Üniversitesi Eğitim Fakültesi Dergisi*, 33(1), 129-156.
- Yıldız, A. G. M., Çiftçi, E., & Karal, H. (2017). Bilişimsel düşünme ve programlama. H.F. Odabaşı, B. Akkoynlu, A. İşman (Ed.), *Eğitim teknolojileri okumaları içinde (75-86)*. Ankara: Sakarya Üniversitesi-TOJET.
- Yılmaz, F., & Çakır, H. (2019). Meslek yüksekokulu öğrencilerinin programlama başarılarını etkileyen faktörlerin incelenmesi. *Eğitim Teknolojisi Kuram ve Uygulama*, 9(2), 416-437.
- YÖK (2018). *Öğretmen Yetiştirme Lisans Programları*. 15 Kasım 2022 tarihinde <https://www.yok.gov.tr> sitesinden erişilmiştir.

## EVALUATING THE ALGORITHM AND PROGRAMMING COURSE FROM THE PERSPECTIVE OF PRE-SERVICE MATHEMATICS TEACHERS

### ABSTRACT

Programming skill is seen as one of the 21st-century skills. For pre-service teachers to acquire this skill, the Council of Higher Education in Türkiye included the Algorithm and Programming course in the education faculties as of the 2018-2019 academic year. In this context, it was aimed to get the opinions of pre-service teachers about the Algorithm and Programming Course, which was started to be given with the renewed curriculum. It was also aimed to reveal the contributions of the course to pre-service teachers in terms of technology, mathematical skill improvement, mathematical thinking, analytical thinking, etc. In mathematics education departments, although it is not directly related to programming such as computer engineering and information technologies, this course was listed as one of the required courses. Accordingly, it has become a necessity to get the opinions of the pre-service teachers and question the effectiveness of the course. A case study, one of the qualitative research methods was used. The participants were reached with the criterion sampling method and were interviewed using a semi-structured interview form. The data obtained from the interviews were analyzed with the content analysis method. As per the research findings, pre-service teachers expressed a positive opinion of the Algorithm and Programming course. They highlighted its role in enhancing their technology skills, making mathematics more enjoyable, fostering a deeper understanding of digital mathematics, and cultivating skills such as creative and analytical thinking. In conclusion, the appropriateness of offering the course in mathematics education departments is affirmed. It is considered an easily attainable course for mathematics pre-service teachers, contributing to the enhancement of their cognitive abilities, technological proficiency, and awareness of diverse teaching methodologies.

**Keywords:** Algorithm, Programming, Pre-service Mathematics Teacher, Technology Literacy, Mathematics Education.



## ALGORİTMA VE PROGRAMLAMA DERSİNİN MATEMATİK ÖĞRETMEN ADAYLARI PERSPEKTİFİNDEN DEĞERLENDİRİLMESİ

### ÖZ

Programlama becerisi 21. yüzyıl becerileri arasında öne çıkan becerilerden biridir. Yükseköğretim Kurulu tarafından, 2018-2019 yılı itibarıyla eğitim fakülteleri ilköğretim matematik öğretmenliği lisans programlarına dahil edilen Algoritma ve Programlama dersi ile matematik öğretmen adaylarının programlama becerilerini geliştirmeleri hedeflenmiştir. Bu bağlamda, çalışma eğitim fakültelerinde güncellenmiş müfredatla birlikte Algoritma ve Programlama Dersi ile ilgili öğretmen adaylarının görüşlerini ortaya çıkarmayı amaçlamıştır. Aynı zamanda dersin öğretmen adaylarının teknoloji, matematiksel beceriler, matematiksel düşünme, analitik düşünme gibi becerilerine olan katkılarının belirlenmesi hedeflenmiştir. Programlama ile doğrudan ilişkisi olmamasına rağmen, Algoritma ve Programlama dersi ilköğretim matematik eğitimi programında zorunlu dersler listesinde yer almaktadır. Bu derse ilişkin öğretmen adaylarının görüşlerini almak ve dersin etkililiğini sorgulamak amacıyla nitel araştırma yöntemlerinden durum çalışması modeli kullanılmıştır. Bu kapsamda ölçüt örnekleme yöntemiyle seçilen matematik öğretmen adaylarıyla yarı yapılandırılmış görüşmeler yapılarak veriler toplanmıştır. Görüşmelerden elde edilen veriler, içerik analizi yöntemiyle analiz edilmiştir. Araştırmanın bulgularına göre öğretmen adaylarının Algoritma ve Programlama dersine karşı olumlu görüşe sahip olup, dersin teknoloji kullanım becerisini artırdığı, matematiği eğlenceli hale getirdiği, dijital matematik anlayışlarını geliştirdiği ve yaratıcı düşünme ile analitik düşünme gibi becerileri geliştirdiği yönünde görüş bildirmişlerdir. Sonuç olarak, dersin ilköğretim matematik eğitimi programlarında verilmesinin uygun olduğu, matematik öğretmen adaylarının kolaylıkla edinebileceği bir kazanım olduğu, öğretmen adaylarının düşünme becerilerini, teknolojik yeterliliklerini ve farklı öğretim yöntemleri konusundaki farkındalıklarını artırdığı düşünülmektedir.

**Anahtar Kelimeler:** Algoritma, Programlama, Matematik Öğretmen Adayı, Teknoloji Okuryazarlığı, Matematik Eğitimi.



## INTRODUCTION

In the evolving and changing world, individuals are expected to possess 21st-century skills. One of these skills is programming, considered a technological literacy skill. The concept of programming emerged as a crucial element in the information age and rapidly advancing technologies. Programming, described as a complex and challenging process, requires high-level thinking, systematic thinking, the development of different perspectives to offer solutions to problems, and creative thinking (Yıldız, Çiftçi, & Karal, 2017). In the literature, there are studies indicating a relationship between programming skills and various thinking skills. For instance, programming skills are highly positively correlated with analytical thinking and moderately positively correlated with logical thinking (Sebetci & Aksu, 2014). Additionally, findings suggest a positive relationship between logical-mathematical intelligence, critical thinking skills, and algorithm design in programming (Korkmaz, 2012). Through programming skills, students acquire not only the ability to develop applications but also critical, algorithmic, and analytical thinking, problem-solving, and questioning skills (Yıldız, Çiftçi, & Karal, 2017).

The traditional teaching of programming languages is reported to be challenging for students due to the complex structure. In particular, block-based visual programming languages facilitate programming education. It is argued that these languages should be taught as an introductory activity for programming language learning (Çatlak, Tekdal, & Baz, 2015). Block-based visual programming languages, easy to learn even for young children, enable students to learn by using problem-solving skills such as goal setting, idea generation, and testing (Taylor, Harlow, & Forret, 2010). These languages not only help students understand programming logic easily but also enhance their creativity (Ouahbi et al., 2015; Pinto & Escudeiro, 2014) and problem-solving abilities (Shin & Park, 2014).

One of the first and most important points in programming education is the instruction of programming logic. The codes written for solving a problem in programming vary according to programming languages however programming logic is the same in all programming languages. Therefore, the instruction in programming logic should take place before the programming education (Arabacıoğlu, Bülbül, & Filiz, 2007). Introducing the concept of algorithms before programming education is essential for grasping programming logic. While teaching programming, designing algorithms is seen as an important requirement for solving problems. Creating programs becomes simpler and more meaningful by using algorithms. Therefore, understanding algorithms is crucial for programming education (Gökoğlu, 2017).

Computer programming has become a course not only in programs directly related to programming, such as Computer Engineering but also for students stud-

ying in faculties of education. The reason for this is to learn programming with the aim of designing interactive multimedia learning environments. (Fesakis & Serafeim, 2009). In our country and other countries, various courses and practices have been added to the curriculum to develop students' programming skills (Sayın & Seferoğlu, 2016; Arslan & Akçelik, 2019). With the update of higher education programs, it is expected that pre-service teachers graduate with the qualification of being 'technologically literate and research-oriented teachers' (Higher Education Council (YÖK), 2018, p.13). Programming skills, one of the skills that enhance technological literacy, have been integrated into programs by both the Ministry of National Education (MEB) and YÖK. In 2012, MEB introduced the Information Technologies and Programming course in the 5th and 6th grades (Pala & Mıhçı Türker, 2018). In many universities, programming education within the scope of the Algorithm and Programming course started in the 2018-2019 academic year.

The Algorithm and Programming course, added in the 2018-2019 academic year, is one of the new courses that should be in line with the skills and competencies targeted by the mathematics education program. (Akıncı and Dübüş, 2022). The algorithmic part of the course involves teaching programming logic with everyday life examples in spoken language (Arabacıoğlu, Bülbül, and Filiz, 2007), while the programming part is the process of giving commands to the computer and developing applications using algorithms (Saygıner and Tüzün, 2017). It is argued that algorithmic thinking, inherent in the programming process, is one of the technological literacy skills and enhances scientific thinking skills (Yıldız, Çiftçi, and Karal, 2017).

There is a growing interest in algorithm and programming education in today's world due to the increasing importance given to technology. A review of the literature reveals various studies related to programming education (Gökoğlu, 2017; Oh, 2017; Göksoy and Yılmaz, 2018; Pala and Mıhçı Türker, 2018; Alsancak Sırakaya, 2019; Arslan and Akçelik, 2019; Öztürk and Alper, 2019; Uzun and Baltalı, 2019; Agbo et al., 2019; Akkaya and Öztürk, 2020; Göncü, Çetin, and Şendurur, 2020). For instance, Gökoğlu (2017) examined metaphors related to the concept of algorithms in the programming education of 372 Computer Programming students. The results indicated that some students had a negative view of the algorithm. However, most students considered the algorithm as a guide in the programming process, valuing it as a step that helps them reach their goals. Alsancak Sırakaya (2019) conducted a study on the impact of programming education on the information-processing thinking skills of Computer Programming department students. The findings revealed that regardless of gender and programming skills, programming education had a meaningful and positive impact on students' information-processing thinking skills. In a study by Saygıner and Tüzün (2017) analyzing studies related to the teaching of programming, one of the difficulties encountered was the inability to solve the logical dimension of the problem, le-

ading learners to struggle with the topic. To overcome this issue, the use of visual-based environments was recommended. Göksoy and Yılmaz (2018) examined the views of information technologies teachers and students on the robotics and coding course. The findings showed that the course provided various skills such as efficient work, systematic, analytical, creative, and numerical thinking, problem-solving, and design. Additionally, students' achievements in numerical courses and problem-solving skills improved.

The existence of highly educated individuals who efficiently utilize computer technologies and translate acquired knowledge into development and production is essential for societies (Sayın and Seferoğlu, 2016). All this information indicates the necessity of programming education to become a highly developed society. Programming education, recognized as a key academic achievement that forms the core of logical reasoning, has recently been included in the educational programs of many countries. In this study, the opinions of pre-service teachers were gathered regarding the Algorithm and Programming course, which has been included in the updated undergraduate mathematics teacher education program. Previous studies were mainly conducted with computer engineering students (Benli and Tek, 2021), information technology teachers (Göksoy and Yılmaz, 2018; Göncü, Çetin, and Şendurur, 2020), and computer programming students (Özyurt & Özyurt, 2015; Gökoğlu, 2017; Alsancak Sırakaya, 2019). These studies are generally related to individuals who are learning or teaching in programs focusing on programming. For example, Göncü, Çetin, and Şendurur (2020) conducted research on teachers who need to teach programming to their students. Alsancak Sırakaya (2019) focused on computer programming students learning programming as a professional requirement. Benli and Tek (2021) conducted their study with Computer and Software Engineering students where programming skills were taught in-depth to future engineers. In this context, pre-service mathematics teachers are not expected to directly teach algorithm and programming skills to their students but are anticipated to apply their learned programming skills to a wide range of real-life situations with mathematics content (Cline et al., 2020). The fact that the Algorithm and Programming course is offered in a department not directly related to programming raises questions about whether the objectives of the course are suitable for department students. Therefore, this study was conducted to reveal the opinions of pre-service teachers studying in the undergraduate mathematics teacher education program of an education faculty that does not have a direct focus on programming. Upon reviewing the relevant literature, no studies were found on pre-service teachers studying in the undergraduate mathematics teacher education program regarding this subject. Therefore, this study aims to reveal the opinions of pre-service teachers on the Algorithm and Programming course within the context of the updated curriculum in education faculties. The following research problem was addressed:

*What are the opinions of primary mathematics teacher candidates about the Algorithm and Programming course included in the primary mathematics education program?*

## METHOD

### Research Design

In this study, a qualitative research method, specifically the case study approach, was employed. Case studies involve an in-depth exploration of one or more individuals, social groups, communities, situations, environments, or programs in their natural context over time and space (McMillan, 2012). The reason for choosing this research design is to uncover the insights of pre-service teachers who have recently completed the Algorithm and Programming course in their department, based on their perspectives. Therefore, a case study was conducted to determine the extent to which pre-service mathematics teachers had achieved the targeted learning outcomes of the course. This involved obtaining the perspectives of individuals who had taken the course completed assignments and projects, and finished their final assignments and exams.

### Sample

Fifteen pre-service mathematics teachers who completed the 2nd year of the Primary School Mathematics Teaching Department at a state university participated in this study. The criterion sampling method was utilized in the selection of participants, where a small number of participants representing specific criteria in the population were chosen (McMillan, 2012). The criteria for selecting participants included completing the Algorithm and Programming course, completing midterm and final assignments, and demonstrating their understanding of the course content by producing products throughout the semester. To ensure diverse representation, participants were selected from high (n=5), medium (n=5), and low (n=5) performance levels on assignments of varying difficulty (advanced, intermediate, and basic) throughout the semester. The demographic information of the participants is presented in Table 1.

**Table 1.** Demographic Information of Participants

| Variables     |        | <i>f</i> | %     |
|---------------|--------|----------|-------|
| <b>Gender</b> | Female | 9        | 60    |
|               | Male   | 6        | 40    |
| <b>Age</b>    | 19     | 2        | 13,33 |
|               | 20     | 9        | 60    |

|                                      |                                |   |       |
|--------------------------------------|--------------------------------|---|-------|
|                                      | 21                             | 3 | 20    |
|                                      | 22                             | 1 | 6,67  |
| <b>Type of High School Graduated</b> | Anatolian High School          | 6 | 40    |
|                                      | Science High School            | 5 | 33,33 |
|                                      | General High School            | 2 | 13,33 |
|                                      | Private High School            | 1 | 6,67  |
|                                      | Distance Education High School | 1 | 6,67  |
| <b>GPA</b>                           | 2.00-2.50                      | 1 | 6,67  |
|                                      | 2.51-2.99                      | 9 | 60    |
|                                      | 3.00-3.50                      | 5 | 33,33 |
|                                      | 3.50-4.00                      | 0 | 0     |

### Data Collection Tool and Data Collection

The data collection tool for this study was developed by the researchers. To ensure its validity, expert opinions in the field were sought. In line with the opinions of the experts, adjustments were made to some questions, and the interview form took its final form. The data collection tool included semi-structured interview questions aimed at gathering the perspectives of teacher candidates on the gains they believe they acquired in the Algorithm and Programming course. The questions addressed topics such as the connection between mathematics and Algorithms and Programming, the impact of the course on their view of mathematics, the advantages and disadvantages of the course, and the personal and professional contributions they attributed to it. Some sample questions included: “*What gains do you think you acquired in the Algorithm and Programming course, and what is its relationship with mathematics? Interpret the contribution of the Algorithm and Programming course to the digital mathematical understanding discussed in the future. Can you explain the advantages and disadvantages of the Algorithm and Programming course?*”

The data for this study were collected from 2nd-year pre-service teachers after the completion of the 14-week Algorithm and Programming course in the spring semester of the 2021-2022 academic year. Throughout the course, the following concepts were covered: algorithms, algorithm diagrams, flowcharts, and the block-based coding program Scratch, along with its applications. Then, Java programming language and its applications were introduced. Weekly assignments related to the subject were requested from the students, and feedback was provided during the semester. In the midterm and final exams, pre-service teachers were asked to perform practical assignments. At the end of the semester, the semi-structured interview questions were provided in written form to the pre-service teachers. They were then asked to respond to these questions individually. A total of nine



questions were directed to the pre-service teachers, and there was no time limit for answering the questions. After an average completion time of 30-35 minutes, the response process was concluded, and the answers were collected and analyzed.

### Data Analysis

The demographic information of the pre-service teachers, such as gender, age, grade level, general GPA, and the type of high school they graduated from, is presented in frequencies and percentages (Table 1). The data obtained from the answers of the pre-service teachers to open-ended questions were analyzed using the content analysis method. In this context, the concepts most repeated in the answers of the pre-service teachers determined the categories reached through induction: technology, programming and coding, the relationship between mathematics and programming, programming learning and teaching methods, the contribution of the course to understanding mathematics, the inclusion status of the course in the curriculum, advantages, and disadvantages. The inter-coder reliability value has been calculated for codes created independently by researchers. For this purpose, Miles and Huberman's (1994) formula was used for intercoder reliability,  $(\text{Agreement} / (\text{Agreement} + \text{Disagreement}))$ , and the reliability value was calculated as 0.78.

## RESULTS

This study aimed to reveal the opinions of pre-service teachers regarding the Algorithm and Programming Course in education faculties with updated curricula. In this context, the goal was to determine the contributions of the course to areas such as technology, the improvement of mathematical skills, mathematical thinking, and analytical thinking among pre-service teachers. Semi-structured interview questions were conducted with pre-service teachers who had never taken the Algorithm and Programming course before (n=12) or expressed having taken other technology-related courses (e.g., Information Technologies, Instructional Technologies) (n=3). The data obtained from these interviews led to the following findings:

### Evaluation of the Algorithm and Programming Course in Terms of Technology

Pre-service teachers frequently emphasized technology when expressing their opinions about the Algorithm and Programming course. About 30.43% of the participants (n=7) highlighted the course's alignment with the technology era, and 30.43% (n=7) stated that they believed their technological competencies had increased. While 17.39% (n=4) of the participants drew attention to the adaptability of mathematics to technology, 8.70% (n=2) expressed that the course contributed

to the efficient and beneficial use of technology. Additionally, 8.70% (n=2) emphasized the necessity of the course due to the widespread use of technology (Table 2).

**Table 2.** *Technology*

| <i>Codes</i>                                      | <i>f</i>  | <i>%</i>   | <i>Participant</i>                  |
|---|-----------|------------|-------------------------------------|
| Alignment with the technology era                 | 7         | 30,43      | PT1, PT3, PT5, PT6, PT7, PT9, PT14  |
| Improvement of technology usage skills/competence | 7         | 30,43      | PT1, PT2, PT3, PT7, PT8, PT10, PT13 |
| Adapting mathematics to technology                | 4         | 17,39      | PT3, PT5, PT13, PT14                |
| Use of efficient/useful technology                | 2         | 8,70       | PT5, PT13                           |
| Prevalence of technology usage                    | 2         | 8,70       | PT7, PT9                            |
| Lack of technological tools                       | 1         | 4,35       | PT5, PT12                           |
| <b>Total:</b>                                     | <b>23</b> | <b>100</b> | <b>12</b>                           |

Table 2 reveals a total of six codes in the technology category. Some of the codes in the technology category include “Alignment with the technology era, improvement of technology usage skills/competence, prevalence of technology usage.” Some sample quotations related to the codes are as follows:

Alignment with the technology era: “Since we have entered the digital age, programming, the internet, and technology are now present in every aspect of our lives. Therefore, the course allows us to adapt to new technology more easily” (PT1).

Adapting mathematics to technology: “As we are in the age of technology, future students are inclined to use computers, phones, and similar technological tools. Combining the student’s interest with technology, games, and educational math applications attracts student interest, increases motivation, and provides an opportunity to present mathematics in a more enjoyable way” (PT2).

Improvement of technology usage skills/competence: “Since computers and technology have become a part of our lives, mastering programming and similar things enhances our digital technology literacy” (PT10).

Prevalence of technology usage: “Our perspective has changed in terms of the prevalence of digital mathematics in our age” (PT9).

## Learning Programming and Coding

In addition to the emphasis on technology, it was observed that 44% of pre-service teachers (n=11) highlighted coding, and an equal percentage, 44% (n=11),

emphasized programming. Furthermore, 12% (n=3) emphasized that they learned programming easily through the Algorithm and Programming course. The codes in this category are shown in Table 3.

**Table 3.** *Programming/coding*

| <i>Codes</i>                        | <i>f</i>  | <i>%</i>   | <i>Participant</i>  |
|-------------------------------------|-----------|------------|---|
| Coding                              | 11        | 44         | PT1, PT2, PT3, PT4, PT6, PT7, PT8, PT10, PT11, PT13, PT14 |
| Programming                         | 11        | 44         | PT1, PT2, PT3, PT4, PT6, PT7, PT8, PT11, PT13, PT14, PT15 |
| Learning programming in an easy way | 3         | 12         | PT8, PT10, PT13   |
| <b>Total:</b>                       | <b>25</b> | <b>100</b> | <b>12</b>   |

Table 3 reveals a total of three codes in the programming-coding category. These codes are related to coding, programming, and learning programming easily. Sample quotations related to these codes are as follows:

Coding: “I learned programming and coding. I understood that coding is present in all software in daily life” (PT1).

Programming: “I learned a new computer programming language. In the future, I will teach this program to my students so that they can improve themselves in terms of programming” (PT8). “I learned the Java programming language, which is an infrastructure I can use in many applications” (PT4).

Coding/Programming: “Thanks to the Algorithm and Programming course, we learn advanced programming, apply mathematical gains to the question-answer application. We learn various coding techniques with Scratch” (PT10). “Teacher candidates who recognize how detailed process the coding is and understand that its logic needs to be grasped from the very basics will want to apply these skills in mathematics.” (PT11).

### Establishing the Relationship between Mathematics and Programming

Pre-service teachers establish a relationship between the mathematics course and programming in the interviews. 32.15% of participants believe that mathematics and programming can be blended (n=9), 21.43% think that programming increases interest in mathematics (n=6), 17.86% believe that seeing mathematics in programming is possible (n=5), 10.71% think that algorithms are related to daily life (n=3), 10.71% believe that mathematical programming can be developed (n=3), and one person mentioned the relationship between patterns and programming. The relevant codes are shown in Table 4.

**Table 4.** Relationship between mathematics and programming

| <i>Codes</i>                               | <i>f</i>  | <i>%</i>   | <i>Participant</i>                              |
|--|-----------|------------|---|
| Mathematics and programming can be blended | 9         | 32,15      | PT2, PT4, PT5, PT6, PT8, PT10, PT13, PT14, PT15 |
| Increasing interest in mathematics         | 6         | 21,43      | PT2, PT4, PT7, PT9, PT13, PT14                  |
| Seeing math in programming                 | 5         | 17,86      | PT3, PT4, PT6, PT8, PT10                        |
| Algorithm and Daily life                   | 3         | 10,71      | PT1, PT9, PT12                                  |
| Developing mathematical software           | 3         | 10,71      | PT4, PT8, PT11                                  |
| Math patterns and programming              | 1         | 3,57       | PT7   |
| Not related                                | 1         | 3,57       | PT15  |
| <b>Total:</b>                              | <b>28</b> | <b>100</b> | <b>15</b>                                       |

Table 4 reveals a total of seven codes indicating that pre-service teachers establish a relationship between mathematics and programming. Some of these codes suggest that mathematics and programming can be blended, interest in mathematics can be increased, mathematics can be seen in programming, and algorithms are related to daily life. Sample quotations related to this category are as follows:

Mathematics and programming can be blended: “It provides pre-service teachers with computer usage and programming skills. It helps pre-service teachers design educational games that will make students love mathematics” (PT2).

Seeing math in programming: “Since there are codes in the programming language, understanding the logic of the codes requires mathematics. Thus, it enables the use of mathematics in the digital realm” (PT3).

Increasing interest in mathematics: “Algorithm is like mathematics, and these languages are similar. If we relate and adapt these languages to each other, it will benefit us in many ways. Mathematics adapted to technology will attract students’ interest even more” (PT14).

Developing mathematical software: “Thanks to the coding I learned in the Algorithm and Programming course, I acquired the ability to document and code certain objectives in the mathematics lesson using a programming language” (PT8).

### **Increasing Awareness of Learning and Teaching Methods in the Algorithm and Programming Course**

In the interviews with pre-service teachers, it is observed that they emphasize their learning and teaching methods. For example, 19.05% mentioned experiential

learning (n=4), 14.29% computer-based learning (n=3), 14.29% drawing attention (n=3), and six pre-service teachers mentioned meaningful learning, collaborative learning, and other teaching methods. The relevant codes are shown in Table 5.

**Table 5.** *Programming learning and teaching methods*

| <i>Codes</i>                                | <i>f</i>  | <i>%</i>   | <i>Participant</i>        |
|---|-----------|------------|---------------------------|
| Different, effective teaching method        | 5         | 23,80      | PT2, PT3, PT6, PT13, PT14 |
| Learning by doing                           | 4         | 19,05      | PT2, PT8, PT11, PT12      |
| Computer-based learning                     | 3         | 14,29      | PT1, PT7, PT9             |
| Drawing attention                           | 3         | 14,29      | PT9, PT12, PT14           |
| Permanent learning                          | 1         | 4,76       | PT4                       |
| Creative thinking                           | 1         | 4,76       | PT3                       |
| Multimedia learning                         | 1         | 4,76       | PT1                       |
| Meaningful learning                         | 1         | 4,76       | PT5                       |
| Cooperative learning                        | 1         | 4,76       | PT12                      |
| Subject teaching with a variety of examples | 1         | 4,76       | PT3                       |
| <b>Total:</b>                               | <b>21</b> | <b>100</b> | <b>12</b>                 |

Table 5 reveals 10 codes in the learning and teaching methods category. Some of these codes include different and effective teaching methods, experiential learning, computer-based learning, drawing attention, and collaborative learning. Sample quotations are as follows:

Different, effective teaching method: “A different method and technique that pre-service teachers can use in class, which can be used with programming. It can provide the student with a different perspective. Creating distinctiveness by using a digital tool” (PT6).

Computer-based learning: “The works we conducted for preparing simulations have been beneficial in creating computer-based in-class applications” (PT9).

Drawing attention: “A method suitable for daily life, which will attract the students’ interest. Provides both application and instruction. Strengthens connections among students through activities” (PT12).

Multimedia learning environments: “It is important for teachers to be able to use programming tools to provide learning experiences that appeal to multiple sensory organs in classes” (PT1).

Creative thinking: “When we align the objectives with technology, it enables creative thinking” (PT3).

## Contribution of the Algorithm and Programming Course to Mathematics Knowledge

The pre-service teachers believe that the Algorithm and Programming course contributes to understanding mathematics. This includes a 33.33% emphasis on making mathematics enjoyable (n=7), 25.57% on mathematics in the digital age (n=6), 14.29% on simplifying mathematical operations (n=3), and 9.53% on efficiency in teaching (n=2). Only one student mentioned that there is no connection between mathematics and the course. The relevant codes are provided in Table 6.

**Table 6.** Contribution of the course to mathematics knowledge

| <i>Codes</i>                          | <i>f</i>  | <i>%</i>   | <i>Participants</i>                 |
|---------------------------------------|-----------|------------|-------------------------------------|
| Fun math                              | 7         | 33,33      | PT2, PT4, PT6, PT7, PT9, PT11, PT13 |
| Mathematics in the digital age        | 6         | 28,57      | PT1, PT2, PT3, PT5, PT7, PT10       |
| Simplifying mathematical operations   | 3         | 14,29      | PT3, PT4, PT6                       |
| Efficiency in teaching                | 2         | 9,53       | PT2, PT6                            |
| Concretizing mathematics              | 1         | 4,76       | PT9                                 |
| Using mathematics in different fields | 1         | 4,76       | PT13                                |
| No benefits                           | 1         | 4,76       | PT15                                |
| <b>Total:</b>                         | <b>21</b> | <b>100</b> | <b>12</b>                           |

According to the interviews with pre-service teachers, Table 6 shows a total of seven codes indicating the extent to which the Algorithm and Programming course contributes to understanding mathematics. Some of these codes include enjoyable mathematics, mathematics in the digital age, simplifying mathematical operations, efficiency in teaching, and concretizing mathematics. Some sample quotations are as follows:

Fun mathematics: “It can increase interest in mathematics. It can make mathematics more enjoyable. Mathematics can be used in different fields” (PT13).

Simplifying mathematical operations: “We perform operations by entering numbers using the pop-up window we created with the codes we wrote in the Java program. With the codes we write, we can find the results of many mathematical operations such as calculating averages, exponentiation, etc.” (PT4).

Mathematics in the digital age: “Since we are in the digital world, mathematics is now becoming a part of it. In the Algorithm and Programming course, we use digital mathematics, make coding, and can see the language of mathematics there” (PT6). “As technology advances, it has also affected educational life. For

this reason, the technology used in courses has increased. Therefore, this course has become important for both mathematics and all other courses” (PT7).

Concretizing mathematics: “The possibility of making studies to make mathematics courses more interesting suggests that it could be more effective in concretizing” (PT9).

### Status of the Algorithm and Programming Course in the Curriculum

The pre-service teachers evaluated the inclusion of the Algorithm and Programming course in the curriculum. The majority reported that it should be included, providing reasons. It was observed that 33.34% mentioned the contribution of the course (n=5), 20% mentioned gaining a different perspective (n=3), 13.33% mentioned attracting the interest of the new generation (n=2), 13.33% mentioned that the course should have been offered earlier (n=2), and 6.67% mentioned that teachers should have knowledge of programming (n=1). They stated that the course should be included in the curriculum for these reasons. The relevant codes in this category are shown in Table 7.

**Table 7.** *Inclusion of the course in the curriculum*

| <i>Codes</i>                                   | <i>f</i>  | <i>%</i>   | <i>Participants</i>      |
|--|-----------|------------|--------------------------|
| Contribution and usefulness                    | 5         | 33,34      | PT6, PT7, PT8, PT9, PT11 |
| Provides a different perspective on the course | 3         | 20         | PT4, PT9, PT13           |
| Attracts the new generation                    | 2         | 13,33      | PT12, PT14               |
| Should have been offered earlier               | 2         | 13,33      | PT1, PT12                |
| Lesson of the future                           | 1         | 6,67       | PT3                      |
| Teacher should know programming/computer       | 1         | 6,67       | PT10                     |
| No, do not include                             | 1         | 6,67       | PT15                     |
| <b>Total:</b>                                  | <b>15</b> | <b>100</b> | <b>13</b>                |

Table 7 reveals a total of seven codes related to the inclusion of the course in the curriculum. Some of these codes include their potential contribution and usefulness, the ability to provide a different perspective on the course, their potential to attract the interest of the new generation, and their potential to be the course of the future. Here are some sample quotations from the comments of pre-service teachers:

Contribution and usefulness: “In my opinion, efforts should be made to have a strong infrastructure in universities, middle schools, and all schools before similar courses are taught. After that, similar courses can be increased, contributing to mathematics education” (PT6). “Someone who understands coding

logic can think more analytically. Since analytical thinking is a fundamental element underlying mathematics, having this course in the program will be beneficial for mathematics teachers” (PT11).

Should have been offered earlier: “I think it’s too late for the course to be given at the university level to change the perspectives of pre-service teachers” (PT12).

Provides a different perspective on the course: “It can bring a new perspective to the mathematics course. Mathematics can be more appealing to students through different methods of explanation or gamification” (PT13).

### Personal Advantages of the Algorithm and Programming Course for Mathematics Pre-service Teachers

The pre-service teachers also mention some personal advantages. For example, they emphasize the increase in computer usage skills (n=10), the possibility of acquiring a different profession in the future (n=8), the ability to establish a better relationship between mathematics and programming (n=3), learning English concepts through programming (n=3), and the ability to teach more content-rich and effective lessons in the future (n=2). At the same time, they mention gaining skills such as analytical thinking, planning, and building self-confidence. The relevant codes are provided in Table 8.

**Table 8.** *Individual advantages*

| <i>Codes</i>  | <i>f</i>  | <i>%</i>   | <i>Participant</i>                                  |
|---|-----------|------------|---|
| Computer skills   | 10        | 32,26      | PT1, PT2, PT5, PT6, PT7, PT8, PT9, PT10, PT12, PT13 |
| Being a programmer/having a different profession                | 8         | 25,80      | PT1, PT3, PT6, PT7, PT11, PT13, PT14, PT15          |
| Establishing a relationship between mathematics and programming | 3         | 9,68       | PT5, PT8, PT14                                      |
| Learning English concepts                                       | 3         | 9,68       | PT1, PT2, PT6                                       |
| Rich/effective lessons  | 2         | 6,45       | PT12, PT13  |
| Math language   | 2         | 6,45       | PT3, PT6  |
| Analytical thinking   | 1         | 3,23       | PT11  |
| Planning  | 1         | 3,23       | PT12  |
| Gaining self-confidence   | 1         | 3,23       | PT12  |
| <b>Total:</b>   | <b>31</b> | <b>100</b> | <b>13</b>   |



Table 8 reveals a total of 10 codes in the personal advantages category. Some of these codes include computer usage skills, the possibility of having a different profession, the ability to establish a relationship between mathematics and programming, learning English concepts, and analytical thinking. Some sample quotations obtained from the opinions of pre-service teachers are as follows:

Becoming a programmer/having a different profession: “It teaches programming, improves technology usage, and provides a foundation for the future” (PT3). “Learning programming in the age of technology. Increasing computer usage, learning software and programming, additional job, future profession” (PT7). “Knowing programming, unlike other math teachers, is an advantage if we consider not teaching as a profession but creating software or something” (PT14).

Learning English concepts: “In the Algorithm and Programming course, we learned to write code with questions related to mathematics objectives. Our coding skills improved. Since we used the English language in the Java program, our English improved. By making a calculator in the program, it made it easy for us to understand mathematics in a fun way” (PT6).

Analytical thinking: “Since mathematics is a course that requires analytical thinking, and the Algorithm and Programming course also develops this, I think it contributes” (PT11).

### Disadvantages of the Algorithm and Programming Course for Mathematics Pre-service teachers

In addition to positive views, pre-service teachers also address some disadvantages of the course. It is observed that 58.82% think it could be a disadvantage for those who do not own a computer (n=10), 23.53% think sitting in front of a computer constantly could lead to some health problems (n=4), 5.88% think learning programming requires patience (n=1), 5.88% find it tiring and time-consuming (n=1), and one teacher candidate emphasizes that it is an unnecessary course for teachers. Disadvantages-related codes are provided in Table 9.

**Table 9.** *Disadvantages*

| <i>Codes</i>                            | <i>f</i>  | <i>%</i>   | <i>Participant</i>                                 |
|---|-----------|------------|--|
| Not owning a computer                   | 10        | 58,82      | PT1, PT2, PT3, PT4, PT5, PT6, PT8, PT9, PT12, PT14 |
| Continuous computer use-health problems | 4         | 23,53      | PT3, PT6, PT7, PT13                                |
| Patience                                | 1         | 5,88       | PT10   |
| Exhausting-Time consuming               | 1         | 5,88       | PT11   |
| Unnecessary for teachers                | 1         | 5,88       | PT15   |
| <b>Total: 5</b>                         | <b>17</b> | <b>100</b> | <b>15</b>  |

Table 9 indicates a total of five codes in the disadvantages category. The codes derived from the opinions include not owning a computer, health problems due to continuous computer use, patience, exhausting-time-consuming, and being deemed unnecessary for teachers. Here are some sample quotations:

Continuous computer use-health problems: “It can be challenging for students who do not have a computer. Looking at the computer too much can cause eye strain and back pain” (PT6).

Not owning a computer: “People who do not have a computer could not actively participate in the course, and this became a disadvantage for them” (PT8). “Since we learn practically in the classroom, everyone should have a computer, but not everyone has one” (PT14).

Patience: “Dealing with programming requires us to be patient. Even a small mistake results in an error” (PT10).

Unnecessary for teachers: “At the moment, it doesn’t mean anything for us, for teachers” (PT15).

Based on the findings obtained, eight categories were identified: technology, programming and coding, the relationship between mathematics and programming, programming learning and teaching methods, the contribution of programming to understanding mathematics, the presence of the course in the curriculum, and its advantages and disadvantages. These findings are discussed in detail in the discussion section, interpreted in line with the literature.

## DISCUSSION

The aim of this study is to present the views of pre-service mathematics teachers on the Algorithm and Programming course. The Algorithm and Programming course has been included in the primary mathematics teacher education program since the 2018-2019 academic year. The course is offered in a program that is not directly related to programming. Thus, it has become necessary to question whether the objectives of the course are suitable for pre-service mathematics teachers. Semi-structured interviews were conducted to evaluate this course from the perspective of pre-service teachers. The findings obtained were analyzed based on the literature.

First, it was observed that almost all pre-service teachers expressed positive views and frequently emphasized the objectives of the course. The reason for this finding could be that pre-service teachers have cognitive skills related to mathematics. According to research, programming skills require similar cognitive skills and experience used in mathematics (Korkmaz, 2012; Ayalew, Tshukudu, and

Lefoanea, 2018), logical and mathematical thinking facilitates learning programming (Cevahir and Özdemir, 2017), and there is a connection between learning programming and mathematics (Kong and Kwok, 2022). Therefore, it can be concluded that the objectives set for the Algorithm and Programming course within the primary mathematics teacher education program are appropriate for students enrolled in the program.

Pre-service teachers often emphasized the use of technology regarding the objectives of the course. Opinions, such as the course being suitable for the age of technology, the enhancement of technology usage skills, and the adaptation of mathematics classes to technology, have emerged. These findings can be considered as an indicator that their technology literacy skills might be improved. Programming skill has also been referred to as technology literacy (Yıldız, Çiftçi, and Karal, 2017). Moreover, it is believed that teaching programming skills to individuals can enhance their digital literacy (Akpınar and Altun, 2014; Durak and Şahin, 2018). In this context, it is important that pre-service teachers mention gaining technology literacy skills during the programming learning process.

Pre-service teachers' willingness to learn coding and programming, even wanting to learn it easily during their undergraduate education, can be considered a sign that they are open to acquiring 21st-century skills. Coding is considered a new form of writing, a way to support mathematical thinking, creativity, problem-solving, and a part of logical reasoning, which is a 21st-century skill (Sayın and Seferoğlu, 2016). It is argued that programming contributes to learners' problem-solving skills by increasing their abilities to imagine, explore, analyze, make assumptions, and generalize, thus enhancing their lifelong learning competencies (Baki, 2001). Therefore, pre-service teachers need programming education to acquire 21st-century skills, including mathematical thinking, problem-solving, analysis, and assumption-making. In fact, they are willing to do so.

Pre-service teachers were subjected to education in algorithm, block-based visual programming, and programming, respectively. Especially, it is known that the primary objective of an algorithm is to understand programming logic (Gökoğlu, 2017), a crucial aspect (Arabacıoğlu, Bülbül, and Filiz, 2007). Gaining skills such as problem-solving and analytical thinking, as mentioned, naturally comes from understanding the logic of programming. Pre-service teachers highlighting the importance of learning programming logic and stating the connection between programming, analytical thinking, and mathematics may indicate their understanding of programming logic and acquisition of the mentioned skills.

Mathematics is inherent in programming. Pre-service teachers have mentioned that blending mathematics with programming can be achieved by developing mathematical software. They expressed that mathematics is already integrated into

programming, and this integration could potentially increase interest in mathematics. Furthermore, they highlighted the benefits of conducting fun mathematics teaching, especially contributing to digital mathematics understanding in the digital age. Research has shown that learning programming facilitates students' understanding of mathematics and motivates them to learn mathematics (Barak and Assal, 2014). Moreover, it increases students' mathematical and technological competencies (Leonard et al., 2016). Accordingly, it can be said that pre-service teachers have gained awareness about the advantages advocated in the literature regarding programming, especially in terms of mathematics.

Throughout the course, topics given to pre-service teachers were presented practically, supported by various assignments, as recommended (Arabacıoğlu, Bülbül, and Filiz, 2007; Akıncı and Dübüş, 2022). Pre-service teachers were observed to emphasize different teaching methods thanks to the course. It is seen that the course, which provides applied education, has increased the awareness of pre-service teachers in terms of teaching methods. It can be said that the Algorithm and Programming course will be beneficial not only in terms directly related to concepts such as technology and software but also in imparting teaching skills, methods, and principles associated with teaching.

Pre-service teachers highlight the potential contributions of the Programming course, integrated into the undergraduate curriculum of primary mathematics education. They believe that the course can offer individuals a fresh perspective, aid in developing creative thinking, problem-solving, and analytical skills, and capture the interest of students in the digital age. It is known that writing codes enhance cognitive skills such as looking at a given problem from different perspectives, finding the shortest solution, recognizing the relationship between events, systematic thinking, and creative thinking (Fesakis and Serafeim, 2009; Göksoy and Yılmaz, 2018). Individuals are recommended to learn programming to acquire digital literacy, problem-solving, analytical thinking, spatial thinking, project creation, solution generation, collaborative work, and hands-on learning skills (Akpınar & Altun, 2014). Therefore, it is seen that the emphasis of pre-service teachers on these benefits based on their learning processes is in line with the literature.

On a personal level, pre-service teachers specifically mention that their computer usage skills have improved, they can establish a relationship between mathematics and programming, they have acquired a new skill towards a different profession, and they have even learned English concepts. In the literature, it is shown that one of the biggest challenges in learning programming is that programming languages are in English (Arabacıoğlu, Bülbül, and Filiz, 2007), and learners' proficiency in a foreign language has a meaningful impact on their self-efficacy in programming (Akçay and Çoklar, 2018). However, in this study, pre-service teachers emphasize that they learned English concepts as a contribution of the course.

This conclusion can be attributed to the practical teaching approach of the course, which involves hands-on learning through a structured sequence covering algorithm design, block-based programming, and programming language instruction.

The drawbacks of the course, as mentioned by teacher candidates, include potential issues for those without access to a computer. Additionally, spending extended periods in front of a computer may result in problems such as back pain. The candidates emphasized that the learning process requires patience. Only one participant expressed negative views about programming and mentioned the unnecessary nature of the course. Considering that learning programming is positively associated with mathematical thinking (Cevahir and Özdemir, 2017), motivation (Yılmaz and Çakır, 2019), self-efficacy belief, and attitude (Özyurt and Özyurt), it is possible that the student expressing negative thoughts may lack some of these factors. Moreover, a positive attitude toward programming (Şahin, Korkmaz, Çakır ve Erdoğmuş, 2019) is recognized to have a positive correlation with the eagerness to learn programming and the conviction that it will be beneficial in one's professional life (Benli and Tek, 2021). Also, a lack of motivation has been reported to have a negative impact on learning programming (Saygıner and Tüzün, 2017). Therefore, the negative attitude of the teacher candidate may be related to the reasons mentioned in the literature.

## RESULTS AND RECOMMENDATIONS

The positive opinions shared by pre-service teachers regarding the Algorithm and Programming course align with existing literature, leading to the conclusion that the course is appropriate within the context of the primary mathematics education program. According to the findings of the study, it can be inferred that the course is compatible with the primary mathematics teacher education curriculum. Based on the opinions of pre-service teachers, there is an increase in their digital literacy and the acquisition of programming skills, which are 21st-century skills. Additionally, it is determined that pre-service teachers grasp the logic of programming and, as a result, perceive a relationship between mathematics and programming. Pre-service teachers, inspired by their learning processes in the course, emphasize teaching methods and believe that their analytical and creative thinking skills have improved, providing them with a different professional experience.

This study is limited to 15 pre-service teachers who took the Algorithm and Programming course in the primary mathematics education program of a state university and completed the course based on the criteria mentioned above. The pre-service teachers learned course content with the concepts of algorithms, block-based visual programming, and the Java programming language sequence by the course instructor. The outcomes of presenting the course with a different programming language or content can be explored in future studies. Moreover, the current

mathematics levels of pre-service teachers, which may be closely related to their programming skills, can vary in pre-service teachers from a different university. Therefore, conducting similar studies with different pre-service teachers is important for the generalizability of the results.

Based on the findings of the research, the following recommendations can be made:

- The course should be delivered in an applied manner.
- The course content should be structured to cover the concepts of algorithm, visual block-based programming applications, and programming language sequentially.
- To ensure that the course outcomes are adequately acquired, the infrastructure of faculties should be made suitable for students who do not own a computer.

## ACKNOWLEDGMENTS AND EXPLANATIONS

We would like to thank all the students who contributed to our study.

### Conflict of Interest

Among the authors of the article, there is no personal or financial conflict of interest within the scope of the study.

### Author Contributions

Design of Study: GSA(%50), EK(%50)

Data Acquisition: GSA(%50), EK(%50)

Data Analysis: GSA(%50), EK(%50)

Writing Up: GSA(%50), EK(%50)

Submission and Revision: GSA(%50), EK(%50)

### Ethical Approval Information

In this study, all rules stipulated within the scope of the “Higher Education Institutions Scientific Research and Publication Ethics Directive” were adhered to.

Ethical Evaluation Committee Name: Kafkas University Social and Humanity Sciences Scientific Research and Publication Ethics Committee

Date of Ethical Evaluation Decision: 17.11.2022

Ethical Evaluation Document Number: E-37673- Number: 39

## REFERENCES

- Agbo, F. J., Oyelere, S. S., Suhonen, J., & Adewumi, S. (2019, Kasım). A systematic review of computational thinking approach for programming education in higher education institutions. *Proceedings of the 19th Koli Calling International Conference on Computing Education Research* (pp. 1-10)
- Akçay, A., & Çoklar, A. N. (2018). Bilişim teknolojileri ve yazılım dersi öğretmen adaylarının programlamaya ilişkin algılanan öz yeterliklerinin farklı değişkenler açısından incelenmesi. *Kastamonu Eğitim Dergisi*, 26(6), 2163-2176.
- Akinci, M., & Dübüş, M. (2022). Ortaokul matematik öğretmenlerinin ilköğretim matematik öğretmenliği lisans programı hakkındaki görüşleri. *Karaelmas Eğitim Bilimleri Dergisi*, 10(2022), 1-14.
- Akkaya, A. ve Öztürk, G. (2020). Algoritma yazma ve öğrenimi hakkında meslek yüksekokulu öğrencilerinin görüşleri. *Balıkesir Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, 22(1), 367-380.
- Akpınar, Y., & Altun, A. (2014). Bilgi toplumu okullarında programlama eğitimi gereksinimi. *Elementary Education Online*, 13(1), 1-4.
- Alsancak Sırakaya, D. (2019). Programlama öğretiminin bilgi işlemsel düşünme becerisine etkisi. *Türkiye Sosyal Araştırmalar Dergisi*, 23(2), 575-590.
- Arabacıoğlu, T., Bülbül, H. İ., & Filiz, A. (2007). Bilgisayar programlama öğretiminde yeni bir yaklaşım. *Akademik Bilişim*, 2007, 193-197.
- Arslan, K. ve Akçelik, M. (2019). Programlama eğitiminde Scratch'ın kullanılması: Öğretmen adaylarının tutum ve algıları. *Ulusal Eğitim Akademisi Dergisi*, 3(1), 41-61.
- Ayalew, Y., Tshukudu, E., & Lefoanea, M. (2018). Factors affecting programming performance of first year students at a University in Botswana. *African Journal of Research in Mathematics, Science and Technology Education*, 22(3), 363-373.
- Baki, A. (2001). Bilişim teknolojisi ışığı altında matematik eğitiminin değerlendirilmesi. *Milli eğitim dergisi*, 149(1), 26-31.
- Barak, M., & Assal, M. (2018). Robotics and STEM learning: students' achievements in assignments according to the P3 Task Taxonomy—practice, problem solving, and projects. *International Journal of Technology and Design Education*, 28, 121-144.
- Benli, K. S., & Tek, F. B. (2021). Programlamaya giriş dersini alan öğrencilerin programlama öz yeterlilik algılarının ve programlamaya bakış açılarının incelenmesi. *Düzce Üniversitesi Bilim ve Teknoloji Dergisi*, 9(3), 328-347.
- Cevahir, H., & Özdemir, M. (2017). Programlama öğretiminde karşılaşılan zorluklara yönelik öğretmen görüşleri ve çözüm önerileri. *Uluslararası Bilgisayar ve Öğretim Teknolojileri Sempozyumu* içinde, İnönü Üniversitesi, Malatya.
- Cline, K., Fasteen, J., Francis, A., Sullivan, E., & Wendt, T. (2020). Integrating programming across the undergraduate mathematics curriculum. *Primus*, 30(7), 735-749.
- Çatlak, Ş., Tekdal, M., & Baz, F. Ç. (2015). Scratch yazılımı ile programlama öğretiminin durumu: Bir doküman inceleme çalışması. *Journal of Instructional Technologies and Teacher Education*, 4(3), 13-25.
- Durak, H. Y., & Şahin, Z. (2018). Kodlama eğitiminin öğretmen adaylarının yaşam boyu öğrenme yeterliliklerinin geliştirmesine katkısının incelenmesi. *Ege Eğitim Teknolojileri Dergisi*, 2(2), 55-67.
- Fesakis, G., & Serafeim, K. (2009). Influence of the familiarization with "scratch" on future teachers' opinions and attitudes about programming and ICT in education. *Acm SIGCSE Bulletin*, 41(3), 258-262.
- Gökoğlu, S. (2017). Programlama eğitiminde algoritma algısı: Bir metafor analizi. *Cumhuriyet International Journal of Education, CIJE*, 6(1), 1-14.
- Göksoy, S. ve Yılmaz, İ. (2018). Bilişim teknolojileri öğretmenleri ve öğrencilerinin robotik ve kodlama dersine ilişkin görüşleri. *Düzce Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, 8(1), 178-196.
- Göncü, A., Çetin, İ. ve Şendurur, P. (2020). Bilişim teknolojileri ve yazılım dersi öğretmenlerinin kodlama eğitimine yönelik görüşleri. *Mersin Üniversitesi Eğitim Fakültesi Dergisi*, 16(2), 301-321.
- Kong, S. C., & Kwok, W. Y. (2022). From mathematical thinking to computational thinking: Use scratch programming to teach concepts of prime and composite numbers. *Proceedings of 29th International Conference on Computers in Education Conference* içinde. *Asia-Pacific Society for Computers in Education* (s. 549-558).
- Korkmaz, Ö. (2012). The impact of critical thinking and logico-mathematical intelligence on algorithmic design skills. *Journal of Educational Computing Research*, 46(2), 173-193.
- Leonard, J., Buss, A., Gamboa, R., Mitchell, M., Fashola, O. S., Hubert, T., & Almughyirah, S. (2016). Using robotics and game design to enhance children's self-efficacy, STEM attitudes, and computational thinking skills. *Journal of Science Education and Technology*, 25, 860-876.
- McMillan, J. H. (2012). *Educational Research Fundamentals for the Consumer* (6. Baskı) Boston: Pearson.
- Miles, M. B., & Huberman, A. M. (1994). *An Expanded Sourcebook Qualitative Data Analysis* (2nd Ed.). London: Sage Publications.

- Oh, M. J. (2017). Non-Major Students' Perceptions of Programming Education Using the Scratch Programming Language. *The Journal of Korean Association of Computer Education*, 20(1), 1-11.
- Ouahbi, I., Kaddari, F., Darhmaoui, H., Elachqar, A., & Lahmine, S. (2015). Learning basic programming concepts by creating games with scratch programming environment. *Procedia-Social and Behavioral Sciences*, 191, 1479-1482.
- Öztürk, S. & Alper, A. (2019). Programlama öğretimindeki ters-yüz öğretim yönteminin öğrencilerin başarılarına, bilgisayara yönelik tutumuna ve kendi kendine öğrenme düzeylerine etkisi. *Bilim, Eğitim, Sanat ve Teknoloji Dergisi (BEST Dergi)*, 3(1), 13-26.
- Özyurt, Ö., & Özyurt, H. (2015). Bilgisayar programcılığı öğrencilerinin programlamaya karşı tutum ve programlama öz-yeterliklerinin belirlenmesine yönelik bir çalışma. *Eğitimde Kuram ve Uygulama*, 11(1), 51-67.
- Pala, F. K. ve Mihci Türker, P. (2018). Öğretmen adaylarının programlama eğitimine yönelik görüşleri. *Kuramsal Eğitimbilim Dergisi*, 12(1), 116-134.
- Pinto, A., & Escudeiro, P. (2014). The use of Scratch for the development of 21st century learning skills in ICT. *In Information Systems and Technologies (CISTI), 2014 9th Iberian Conference on IEEE*, 1-4.
- Saygıner, Ş., & Tüzün, H. (2017). Programlama eğitiminde yaşanan zorluklar ve çözüm önerileri. *Uluslararası Bilgisayar ve Öğretim Teknolojileri Sempozyumu*, İnönü Üniversitesi, 78-90.
- Sayın, Z., & Seferoğlu, S. S. (2016). Yeni bir 21. yüzyıl becerisi olarak kodlama eğitimi ve kodlamanın eğitim politikalarına etkisi. *Akademik Bilişim Konferansı*, 3(5), 1-13.
- Sebetci, Ö. & Aksu, G. (2014). Öğrencilerin mantıksal ve analitik düşünme becerilerinin programlama dilleri başarısına etkisi. *Eğitim Bilimleri ve Uygulama*, 13 (25), 65-83
- Shin, S., & Park, P. (2014). A study on the effect affecting problem solving ability of primary students through the scratch programming. *Advanced Science and Technology Letters*. 59,117-120. <https://doi.org/10.14257/astl.2014.59.27>
- Şahin, H., Korkmaz, Ö., Çakır, R., & Erdoğmuş, F. U. (2019). Bilişim teknolojileri öğretmenlerinin kodlamaya dönük tutumları, öz-yeterlilikleri ve kodlama öğretimi için kullanıkları yöntemler. *Ondokuz Mayıs University Journal of Education Faculty*, 38(2), 1-16.
- Taylor, M., Harlow, A., & Forret, M. (2010). Using a computer programming environment and an interactive white-board to investigate some mathematical thinking. *Procedia-Social and Behavioral Sciences*, 8, 561-570. [Çevrim-içi:10.1016/j.sbspro.2010.12.078, Erişim tarihi: 04.10.2022].
- Uzun A. ve Baltalı, S. (2020). Programlama öğretiminde kullanılacak yazılımlara ilişkin öğretmen görüşleri. *Uludağ Üniversitesi Eğitim Fakültesi Dergisi*, 33(1), 129-156.
- Yıldız, A. G. M., Çiftçi, E., & Karal, H. (2017). Bilişimsel düşünme ve programlama. H.F. Odabaşı, B. Akkoyunlu, A. İşman (Ed), *Eğitim teknolojileri okumaları* içinde (75-86). Ankara: Sakarya Üniversitesi-TOJET.
- Yılmaz, F., & Çakır, H. (2019). Meslek yüksekokulu öğrencilerinin programlama başarılarını etkileyen faktörlerin incelenmesi. *Eğitim Teknolojisi Kuram ve Uygulama*, 9(2), 416-437.
- YÖK (2018). *Öğretmen Yetiştirme Lisans Programları*. 15 Kasım 2022 tarihinde <https://www.yok.gov.tr> sitesinden erişilmiştir.



