

7. Sınıf Öğrencilerinin Matematikte Arduino ile Programlama Üzerine Farkındalıklarının Artırılması

Talip Gültekin^{1*} 
Serkan Narlı² 

¹Dokuz Eylül Üniversitesi Yüksek Lisans Öğrencisi, İzmir/Türkiye
talipgultekin1122@gmail.com

² Dokuz Eylül Üniversitesi Matematik Eğitimi Ana Bilim Dalı, İzmir/Türkiye
serkan.narli@deu.edu.tr

*Sorumlu Yazar

Geliş tarihi: 14.01.2025
Kabul tarihi: 25.02.2025
Yayın tarihi: 30.05.2025

Özet: Günümüz çağında, dijitalleşmenin başlaması, teknolojiye ileriye doğru ilerlemelere bağlı olarak eğitim uygulamalarında, değişim ve yenilik anlamında kayda değer bir artışa olanak sağlamıştır. Bu bağlamda çalışmamızın amacı, bilgisayar ve robotik kodlama bilgisi verirken öğrencilerin algoritmik düşüncelerini geliştirerek farkındalık oluşturmaktır. Bu karma yöntem çalışması, İzmir ilindeki bir vakıf ortaokulundan gönüllülük esasına göre seçilen 15 yedinci sınıf öğrencisi ile yürütülmüştür. Nitel veriler yarı yapılandırılmış görüşmeler yoluyla toplanırken, nicel veriler Teknoloji Hazırlık Ölçeği ve Dijital Okuryazarlık Ölçeği kullanılarak toplanmıştır. Öğrencilere altı haftalık bir algoritmik programlama kursu verilmiştir. Kursun tamamlanmasının ardından öğrencilerle birlikte Arduino kullanarak mini bir CNC plotter makinesi tasarlanmıştır. Veriler, kurs öncesinde ve sonrasında uygulanan görüşme formları ve ölçekler yardımıyla toplanmıştır. Nitel verilerin analizinde içerik analizi yöntemi, nicel verilerin analizinde ise Wilcoxon İşaretli Sıralar Testi kullanılmıştır. Çalışma sonunda öğrenciler kodlama yetkinliklerinin algoritmik becerilerini geliştirdiğini ve geometrik şekillerin farklı bir şekilde ifade edilebildiğini belirtmişlerdir.

Anahtar Kelime: Arduino, kodlama, matematik eğitimi, teknoloji, yazılım

GİRİŞ

İnsanlığın tarih sahnesine çıkışından günümüze kadar geçen sürede insanoğlu sürekli hayatı kolaylaştıracak ortaya yeni bir ürün çıkarma girişiminde bulunmuştur. Bu eylemlerin her biri teknoloji olarak ifade edilebilir (Agar, 2019). Kor insanın elini yakmasaydı maşaya, çiviye parmağımızla basabilseydik çekice, zihinsel yetilerimiz her türlü matematiksel ifadeyi çözebilecek kapasitede olsaydı hesap makinelerine gerek olmazdı (Günay, 2017). Teknoloji ifadesini sadece dijitalleşmeye bağlı olarak hayatımıza giren aletlerden ibaret görmek yerine genel bir çerçeve halinde insanlığın yapamadığı veya yapmakta zorlandığı her türlü iş ve eylemi kolaylaştırmak olarak ifade etmek daha doğru olacaktır. Teknoloji, bizlere sunduğu fırsatlar ile eğitim sektöründe bir araç olarak kullanılabilir (Akbulut vd., 2020). Bireyin sahip olduğu davranışları isteğe bağlı bir şekilde değiştirmeyi amaçlayan bir öğrenme sürecinde, öğrenciye sadece var olan bilginin aktarılması yeterli bir öğrenme olmamaktadır (Tekin ve Keser, 2020). Öğrencilere bilgiyi sadece önceden sindirilmiş bir biçimde sunmak yerine; eleştirel düşünmeyi, yorumlamayı veya bağımsız araştırmayı teşvik edici yöntemler izlenmelidir (Cameron, 2024). Öğrencilerin matematiği anlamalarını artırmak için, öğrenme sürecinde öğrenmeyi daha ilgi çekici ve anlamlı hale getirecek bir dizi eğitim materyali kullanmak gerekebilir (Antoino vd., 2024). Öğrencilerin matematik öğrenmeleri, onlara yaratıcı düşünme ve problem çözme becerileri kazandırmaktadır (Bal Sezerel, 2012). Teknolojinin ve yaşam standartlarının kalitesi ile bir doğrudan ilişkili olan matematik eğitiminin önemi her geçen gün daha çok anlaşılmaktadır (Işık vd., 2008).

Matematik dersi gerek günlük hayattaki kullanımından gerekse gelecekteki mesleklerinin gerektirdiği görev ve sorumluluklardan dolayı öğrenci için önemli bir yere sahiptir (İlhan ve Çağılhan Gülten, 2013). Matematiği öğrencilere daha iyi öğretmek ve bu öğrenmelerini kalıcı hale getirecek birçok yenilik bulunmaktadır (Bakırcı ve Özmantar, 2023). Bunlardan birisi de bilgisayar kodlamadır (Lupas vd., 2024). Jacobsen, vd. (2013) göre okullarda kodlama ve bilişimsel düşünmeye yönelik yapılan araştırma ve uygulamalar, öğrencileri zorlarken bir yandan da öğrenciler için anlamlı olan işlerle meşgul olduklarında ve

öğretmenlerine sürekli geri bildirim sağlayınca öğretmenleri tarafından desteklendiklerinde bilgiyi öğrenmelerinde faydalı olmuştur.

Bireyler, yaşamlarını sürdürebilmek ve çağın gereksinimlerine uyum sağlayabilmek için çeşitli becerilere ihtiyaç duymaktadır. yüzyıl becerileri, çağın gereklerine uyum sağlamak için bilgi ve yeterliliklerin sentezlenmesiyle ortaya çıkan, disiplinlerarası ve çağdaş yaşamın birçok yönüyle ilişkili yetkinliklerdir (Atış Akyok ve Avşar, 2022). 21. yüzyıl becerileri; “Öğrenme ve Yenilikçilik Becerileri”, “Yaşam ve Kariyer Becerileri” ve “Bilgi, Medya ve Teknoloji Becerileri” olmak üzere üç boyutlu olarak ele alınabilir (Kurudayıoğlu ve Gociaoğlu, 2021). Bilgi teknolojileri okuryazarlığının alt temalarından birini oluşturan ve son zamanlarda ismini sıklıkla duyduğumuz becerilerinden biri de kodlama becerisidir (European Commission, 2014a). Programlama, herhangi bir problemin bir programlama dili yardımıyla çözülmesi için yazılan kod satırlarına verilen addır (Şahin ve Namlı, 2017). Bir yazılımı kodlayabilmek için kod yazma becerisine sahip olunmalıdır. Yani program yazmak, üst düzey becerilere sahip olmayı gerektirir (Yükseltürk ve Altınok, 2015). Bu üst düzey becerileri kazanmak için algoritma yapmak da önemli bir yere sahiptir. Algoritmalar herhangi bir programlama dili ile yazılacağı gibi sözel olarak da ifade edilebilir. Dijital platformlarda algoritmaların uygulamaya geçirilebilmesi için bilgisayarın yazılım kısmının kullanılması gerekmektedir. C, C++, Java gibi programlama dilleri ortaokul öğrencileri için karmaşık ve bir o kadar da zor gelebilir (Kalelioğlu ve Gülbahar, 2014). Onun yerine geliştirilen Code.org, Scratch, mBlock gibi blok tabanlı kodlama platformlar, öğrencilerin kurduğu algoritmaları uygulamaya geçirmelerini hem daha kolay hem de öğrenci için daha keyifli hale getirmektedir (Olsson ve Granberg, 2022). Bu tip platformlar programlama bilgisi gerektirmeyen, blokları sürükleyip bırakarak işin derinliğine girmeden kod yazılabilen, basit ve anlaşılır ara yüzü ile her yaştan insana hitap edebilen uygulamalardır (Turan ve Yıldız, 2021). Aynı zamanda kullanıcıya görsellik açıdan kolaylık sağlayan farklı bileşenleri de içermektedir (Cheng vd., 2023). Scratch sayesinde öğrenciler matematik konularını öğrenirken daha hevesli ve aktif hale gelerek yaratıcı düşünme ve işbirlikli çalışma becerilerini geliştirmektedir (Bernard, Stiawan, 2020). Nitekim Mo ve arkadaşları (2021) tarafından yapılan bir çalışmada Scratch programı ile öğrencilere görsel matematiksel imgeler sunarak öğrencilerin bu kavramları keşfetmeye yönelttiği belirlenmiştir. Arduino ise basit bir mikro denetleyici kart olmakla birlikte sürekli kendini geliştirme özelliği ile de açık kaynak kodlu bir platformdur (Mollo vd., 2016, s. 60). Arduino eğitim/öğretimin farklı kategorilerinde ve farklı konularında kullanılan bir kaynaktır (Tudela ve Marin, 2023). Arduino'nun eğitim alanında kullanımı: potansiyeli, var olan türlerin çeşitliliği, sağladığı farklı deney olanakları ve düşük maliyeti nedeniyle önemli bir yere sahiptir (Marín vd., 2024). Ayrıca Belmonte ve arkadaşları (2020) eğitimde Arduino araçlarının kullanımını deney yaparak öğrencilere kavramların öğretiminde uygulama ve anlamalarını pekiştirme fırsatı sağladığını; tasarım faaliyetlerine kendilerinin yapmalarına teşvik ettiğini ve öğrencilerin yaratıcılıklarının geliştirdiği gibi faydalarının olduğunu ifade etmişlerdir. Arduino, matematik eğitiminde ilgi çekici bir eğitim aracı olarak kullanılabilir. Örneğin Arduino ile bazı kontrollü hareketler ve sensör ölçümlerini içeren kodlama projeleri, öğrencilere trigonometri ve cebir konularını öğretmek için kullanılabilir (Negrete, 2023). Aynı zamanda matematiksel denklemleri ve formülleri görselleştirmeye ve manipüle etmeye de yardımcı olur (Negrete, 2024).

Blok temelli kodlamanın matematik eğitiminde kullanılması konusuyla alakalı birçok çalışma yapılmıştır. Bu çalışmaların matematik eğitime katkı sağladığı görülmektedir. Rodríguez – Martínez vd. (2020) matematik eğitiminde bilgi işlemsel düşünme becerisini ve matematiksel kavramların öğrenci üzerindeki gelişimini incelemek amacıyla Scratch yazılımını kullanmışlardır. Sonuç olarak Scratch yazılımı öğrencilerin bilgi işlemsel düşünme becerilerinin gelişiminde fayda sağladığı sonucu çıkmıştır. Turan ve Yıldız (2021) tarafından yapılan bir çalışmada, Scratch yazılımının matematik eğitime etkisi incelenmiştir. Süreç içerisinde, öğrenciler kodlama içerisindeki temel matematiksel ifadeleri anlama yeteneği kazanırken, aynı zamanda programlama ile öğrenme sürecini eğlenceli ve yaratıcı bir şekilde deneyimleme imkanı bulmuşlardır. Tewes (2019) tarafından yapılan bir araştırmada, ortaokul matematik dersinde kodlama yapılmasının öğrencilerin matematiksel kavramları deneyimleme ve anlama üzerindeki etkileri incelenmiştir. Araştırmanın sonunda, kodlamanın ortaokul matematik dersinin içine entegre edilmesinin öğrencilerin matematik deneyimlerini ve matematiksel kavramları anlamalarını olumlu yönde etkileyebileceği görülmüştür.

Robotik kodlama alanlarının matematik eğitime dahil edilmesi konusuyla alakalı birçok çalışma yapılmıştır. Bu faaliyetlerin matematik ve matematik eğitime katkı sağladığı da görülmektedir. Meço ve

Görgülü Arı (2021) tarafından gerçekleştirilen bir araştırmada, STEM eğitimlerinde Arduino kartı kullanımının öğrencilere olan faydası öğrenci görüşlerine göre incelenmiştir. Sonuç olarak, öğrenciler tarafından etkinliklerin faydalı ve eğlenceli bulunduğu, fen dersine karşı bakış açılarının olumlu yönde değiştiği, dersi daha çok sevdikleri ve gelecekteki meslek seçimlerinin mühendislik ve fen alanlarına kaydığı söylenmiştir. Baysal vd. (2020) çalışmasında, kodlama ve Arduino eğitimleri alan ortaöğretim öğrencilerinin, bu eğitimler sonrasında kodlama, Arduino ve STEM hakkındaki görüşleri incelenmiştir. Bu çalışmanın sonunda, verilen eğitimlerin öğrencilerin probleme dayalı ve akran destekli öğrenme becerilerini olumlu yönde geliştirdiğini, karşılaştıkları problemleri çözmek için çeşitli robotlar geliştirdiği ve problemlerine çözüm üretme çabası gösterdikleri gözlenmiştir. Eğitimler, öğrencilere proje esnasında yaratıcı düşünme becerileri kazandırmış, olaylara farklı açılardan bakmayı ve problemleri çözme sürecine katkı sağlamayı öğretmiştir. Tekin ve Keser (2020) tarafından ortaokul 6. Sınıf öğrencileriyle gerçekleştirilen bu çalışmada, robotik kodlama etkinliklerinin öğrencilerin matematik başarısına etkisi incelenmiştir. Çalışmanın sonuçlarına göre, robotik etkinliklerle desteklenen matematik öğretim yönteminin, geleneksel yöntemlere kıyasla matematik başarısını artırmada daha etkili olduğu belirlenmiştir. Matematik başarı düzeyi düşük öğrencilerde robotik etkinliklerin daha fazla etki sağladığı, orta ve yüksek başarı düzeyine sahip öğrencilerde ise anlamlı bir fark yaratmadığı tespit edilmiştir. Marin vd. (2024) Arduino uygulamasının ortaokul öğrencilerine getirdiği fayda ve fırsatları analiz etmek amacıyla bir meta analiz çalışması yapmıştır. Bu çalışmanın sonunda Arduino kartının daha çok *teknoloji* ve fizik konularıyla birlikte STEM projelerinde de disiplinler arası çalışmalarda kullanıldığı görülmüştür. Yine Belmonte vd. (2020) yaptığı başka bir meta analiz çalışmasında Arduino'nun bilimsel literatürdeki gelişimini analiz etmek amacıyla 346 doküman incelemesi yapılmıştır. Bu çalışmanın sonunda eğitim alanında Arduino üzerine yapılan çalışmaların zeminin tam oturmadığı, hala günümüzde konuyla ilgili çalışmaların devam ettiği ifade edilmektedir. Yapılan çalışmaların genelinin fizik deneyleri, bilgi işlemsel düşünme ve bilgisayar tabanlı öğrenme üzerine olduğu da çalışmanın sonuçları arasındadır.

Literatürdeki konu ile ilgili çalışmalar incelendiğinde Code.Org, Scratch gibi uygulamalar öğrenme-öğretme sürecinde kalıcı öğrenme sağlamakla birlikte öğrencilerdeki ilginin artmasına ve eğlenceli öğrenme ortamları oluşturmada olumlu sonuçlar doğurmuştur. Arduino uygulamalarının öğrencilerdeki algoritmik düşünme yapılarını geliştirdiği, girişimcilik becerilerinin olumlu yönde etkilediği gözlemlenmiştir. Literatür taramamızın sonucunda Arduino kartının matematik eğitiminde doğrudan kullanıldığı çalışma sayısının azlığı, daha çok fen eğitiminde ve STEM çalışmalarında kullanıldığı dikkatimizi çekmiştir. Aynı zamanda eğitimde teknolojinin kullanımının 21. yüzyıl becerilerinin gelişimine sağladığı katkı dikkate alındığında, ortaokul öğrencilerine yönelik yazılım ve robotik kodlama ile matematik eğitimi alanında farkındalık oluşturmayı hedefleyen bir çalışma tasarlanmıştır. Bu doğrultuda, öğrencilerin analitik düşünme, problem çözme ve dijital okuryazarlık becerilerinin geliştirilmesine yönelik bir eğitim süreci yapılandırılmıştır.. Bu bilgiler ışığında bu çalışmanın amacı, dinamik halde olan teknolojinin eğitime entegrasyonu ile öğrencilerin matematik konularında hata miktarlarını minimuma indirmek, matematiği öğrenirken keyif alarak öğrenmelerini sağlamak ve öğrencilere “öğrenci merkezli” bir öğrenme ortamı sunarak bir farkındalık oluşturmaktır. Aynı zamanda öğrencilerin bilgisayar ve robotik kodlamaları öğrenirken algoritmik düşünce yapılarının gelişmesi hedeflenmektedir. Bundan dolayı çalışmanın problem cümlesi “Matematik eğitiminde Arduino ile programlama konusunda 7. sınıf öğrencilerinin farkındalıkları nasıldır?” olarak belirlenmiştir ve aşağıdaki alt problemlere cevap aranmıştır.

- a) Arduino ve matematik hakkında öğrencilerin görüşlerinde uygulama öncesinde ve sonrasında farklılık oluşmakta mıdır?
- b) Teknolojiye hazırlık ölçeğinde uygulama öncesinde ve sonrasında anlamlı fark oluşmakta mıdır?
- c) Dijital okuryazarlık ölçeğinde uygulama öncesinde ve sonrasında anlamlı fark oluşmakta mıdır?

YÖNTEM

Araştırmanın Deseni

Creswell (2006) karma yöntem çalışmalarının bir araştırma programı kapsamında yapılan tek bir çalışma veya çoklu çalışmalar (multiple studies) içerisinde, nicel ve nitel verilerin toplanması ve analiz edilmesini

kapsadığını söylemektedir. Bu bağlamda nicel ve nitel verilerin birlikte kullanılacağı bu çalışmada karma yöntem kullanılmaktadır.

Çalışma Grubu / Evren- Örneklem

Araştırmanın çalışma grubu, İzmir ilinde bir vakıf ortaokulunda 2022/2023 eğitim öğretim döneminde öğrenim görmekte olan 15 tane 7. sınıf öğrencisinden oluşmaktadır. Bu çalışmada rastgele olmayan örnekleme tekniklerinden uygun örnekleme yöntemi kullanılmıştır. Burada araştırmacı, ihtiyaç duyduğu büyüklükteki gruba ulaşana kadar, en kolay erişilebilir ve maliyet açısından en tasarruflu bireyleri seçerek veri toplar (Büyüköztürk vd. , 2010).

Veri Toplama Araçları

Öğrencilerin görüşleri yarı yapılandırılmış bir görüşme ile belirlenmiştir. Yarı yapılandırılmış görüşme tekniği, yapılandırılmış görüşme tekniğinden biraz daha farklılık gösterir. (Türnüklü, 2000). Bu teknikte, araştırmacı önceden sormayı planladığı soruları içeren görüşme protokolünü hazırlar (Türnüklü, 2000). Formun geliştirilme aşamasında önce literatür taraması, sonra uzman görüşü alınır. Bu çalışmada kullanılacak görüşme formu uzman görüşü doğrultusunda hazırlanmış 7 sorudan oluşmaktadır. Sorular aşağıda verilmektedir:

1. Bilgisayar kullanmayı ne derece biliyorsunuz, kısaca bahsediniz.
2. Hangi ölçüde bilgisayara erişim imkânınız var?
3. Algoritma ifadesini daha önce duydunuz mu? Cevabınız evetse kelime anlamını açıklayınız.
4. Programlama hakkında neler biliyorsunuz?
5. Bildiğiniz programlama dilleri var mı? Varsa hangileri?
6. Sizce matematik öğretiminde yazılım kullanılabilir mi?
 - i) Matematik öğreniminizi kolaylaştır mı zorlaştırır mı? Açıklayınız.
 - ii) Ne gibi yararlar sağlayabileceğini düşünüyorsunuz?
7. Bir yazılımcı olsanız hayatı kolaylaştıracak nasıl bir program yazardınız?

Öğrencilerin teknoloji kullanmaya istekliliklerini ölçmek için ise Parasuraman (2000) tarafından geliştirilen ve Özer, Eriş ve Özmen (2011) tarafından geçerlik güvenirlik çalışmaları yapılarak Türkçe'ye uyarlanan "Teknolojiye hazırlık ölçeği" (THÖ) kullanılmıştır. İyimserlik (7 madde), yenilikçilik (6 madde), güvensizlik (6 madde) ve huzursuzluk (4 madde) olmak üzere 4 boyuttan oluşan ölçekte 23 madde bulunmaktadır. Öğrencilerin dijital okuryazarlıklarını ölçmek için ise Ng (2012) tarafından geliştirilen ve Hamutoğlu vd. (2017) tarafından Türkçe'ye uyarlanan "Dijital okuryazarlık ölçeği" (DOÖ) kullanılmıştır.

Veri Toplama Süreci

Yarı yapılandırılmış görüşme formu, THÖ ve DOÖ öğrencilere verilen Bilgisayar ve Arduino Programlama Kursu öncesinde uygulanmıştır. Kurs sırasında öğrencilere teknolojinin ne demek olduğundan, dijitalleşme ile nelerin yapıldığından ve nelerin yapılabileceğinden bahsedilmiştir. Daha sonrasında Scratch yazılım platformu öğrencilere tanıtılmış ve Scratch üzerinden öğrencilerle birlikte çeşitli oyunlar yapılmıştır. Arduino öğrencilere tanıtılmış ve Arduino ile nelerin yapılabileceği öğrenciler ile tartışılmıştır. TinkerCat yazılım platformu üzerinden öğrencilerle çeşitli Arduino devrelerinin simülasyonu yapılmıştır. Öğrencilere mBlock uygulaması tanıtılmış ve bu uygulama ile Arduino devrelerin kodlarının nasıl oluşturulacağı ve bu devrelerin nasıl çalıştırılacağı gösterilmiştir. Daha sonrasında öğrencilere dağıtılan Arduino başlangıç kitleri her iki öğrenciye bir kit düşecek şekilde paylaştırılmıştır. Öğrencilerle birlikte mBlock uygulamasından çeşitli Arduino devreleri inşa edilmiştir. Sürecin en sonunda ise öğrencilerle birlikte bir adet mini CNC Plotter tasarımı yapıp makine (bkz. Şekil 1) çalıştırılmıştır.

Şekil 1

Mini CNC Plotter



Matematiksel ifadeler ve günlük hayattan çeşitli görseller öğrencilerle birlikte makineye çizdirilmiştir. 6 haftalık bu sürecin sonunda ise sürecin en başında yapılan yarı yapılandırılmış görüşme formu ile THÖ ile DOÖ öğrencilere tekrar uygulanmıştır.

Yapılan bu çalışmada araştırma etiği ilkeleri gözetilmiş olup gerekli etik kurul izinleri alınmıştır. Etik kurul izni kapsamında; Dokuz Eylül Üniversitesi Etik Komisyonundan 01.11.2022 tarih ve E-87347630-659-412496 sayılı belge alınmıştır.

Verilerin Analizi

Bu çalışmada görüşme verilerinin analizi için nitel araştırma yöntemine uygun olarak içerik analizi tekniğinden yararlanılmıştır. İçerik analizi, birbirine benzeyen verilerin belirlenen kodlar, kategoriler ve temalar kapsamında bütüncül bir bakış açısıyla net ve anlaşılır bir biçimde yorumlanarak okuyucuya sunulduğu bir analiz tekniğidir (Yıldırım ve Şimşek, 2021). Bu çalışmada da içerik analizi tekniğine tabi tutulan veriler, alt problemler bağlamında belirlenen kategori ve temalara göre özetlenmiş ve yorumlanmıştır. Verilerin analizi sürecinde çalışmanın güvenilirliğinin sağlanması için elde edilen veriler iki araştırmacı tarafından bağımsız bir şekilde kategorize edilmiş ve kodlanmıştır. Devamında ise araştırmacılar bir araya gelmiş, birbirinden bağımsız şekilde oluşturmuş oldukları kategori ve kodları tartışarak karşılaştırmışlar ve benzerlik olmayan kategori ve kodlarda fikir birliğine varmışlardır. Fikir birliği sonucunda netleştirilen kategori ve kodlardaki uyumun belirlenmesinde Miles ve Huberman'ın (1994) $[(\text{Görüş birliği} / (\text{Görüş birliği} + \text{Görüş ayrılığı})) \times 100]$ formülü kullanılmıştır. Gerçekleştirilen hesaplamalar sonucunda ise uyum oranının %81 olduğu tespit edilmiştir. Miles ve Huberman'a göre uyum oranı %70 ve üzerinde olan çalışmalar güvenilir olarak kabul edilmektedir (Miles ve Huberman, 1994). Ayrıca uyuşmayan kısımlar araştırmacılar tarafından uzman görüşü de alınarak tekrar tartışılmış, değerlendirilmiş ve fikir birliği sağlanarak çalışmanın güvenilirliği desteklenmiştir. Nicel verilerden elde edilen veriler SPSS 24.0 paket programı kullanılmıştır. Öğrencilerin öntest- sontest puanları referans alınarak frekans (f), yüzde (%), ortalama ($\bar{X}$) ve standart sapma (ss) değerleri hesaplanmıştır. Nicel verilerin analizi yapılırken araştırma bulgularının normal dağılım göstermemesinden ve uygulamanın yapıldığı örneklem 30 kişiden az olduğu için öntest ve sontest ortalamaları arasındaki farkın anlamlı olup olmadığına Wilcoxon İşaretli Sıralar Testi ile bakılmıştır. Wilcoxon İşaretli sıralar testi, "birbiriyle ilişkili iki ölçüm setine ait puanlar arasındaki farkın anlamlılığını test etmek amacıyla kullanılır" (Büyüköztürk, 2020). Bu teknik örneklemi az olan gruplarda ve örneklem normal dağılım göstermediği durumlarda ilişkili t-testi yerine kullanılabilir (Büyüköztürk, 2020). Son olarak öğrenciler sorulan sorulara birden fazla kategoriye ait olabilecek cevaplar verdikleri için toplam frekanslarda farklılıklar olmuştur.

Uygulanan THÖ'nin Cronbach Alpha güvenilirlik katsayısı 0,864, DOÖ'nin Cronbach Alpha güvenilirlik katsayısı 0.946 çıkmıştır.

BULGULAR

Çalışmanın bu aşamasında 6 haftalık kursun öncesinde ve sonrasında katılımcıların görüşme sorularına ve ölçeklere verdikleri yanıtların analiz edilmesiyle elde edilen bulgulara yer verilmiştir.

Birinci Alt Probleme İlişkin Bulgular

Çalışmanın “Arduino ve matematik hakkında öğrencilerin görüşlerinde uygulama öncesinde ve sonrasında farklılık oluşmakta mıdır?” alt problemine yönelik elde edilen veriler aşağıda sunulmuştur.

Tablo 1’de katılımcılara uygulanan görüşme formundaki “*Bilgisayar kullanmayı ne derecede biliyorsunuz? Kısaca bahsediniz.*” sorusuna uygulama öncesinde ve sonrasında öğrencilerin verdikleri cevaplar incelenmiştir.

Tablo 1

Öğrencilerin bilgisayar kullanmayı ne derece bildiklerine ilişkin görüşleri

Kategoriler	Uygulama Öncesi Frekanslar	Uygulama Sonrası Frekanslar
Yazılım ve tasarım yapabiliyorum	3	9
Kendi işimi yapacak kadar biliyorum	8	5
Film izlemek	2	1
Oyun oynamak	5	3
Ders çalışmak	4	2
Robotik kodlama yapmak	0	5
Bilgisayar donanımını biliyorum.	0	2

İncelemeler doğrultusunda Bilgisayar ve Arduino Programlama kursu almayan öğrencilerin bilgisayar kullanmayı daha çok kendi işimi yapacak kadar biliyorum (36,36) olarak ve oyun oynayacak (22,72) kadar biliyorum cevaplarını vermişlerdir. Arduino Programlama kursu alan öğrenciler ise daha çok bilgisayar kullanmayı yazılım ve tasarım yapacak kadar (%33,33) bilmektedir. Bilgisayar ve Arduino Programlama kursunu alan öğrencilerin cevaplarında nispeten bir farklılığın olduğu söylenebilir.

Katılımcılara uygulanan görüşme formundaki “*Hangi ölçüde bilgisayara erişim imkanınız var?*” sorusuna öğrencilerin 4 tanesinin evlerinde kendisine ait bilgisayarı varken 8 tanesinin evinde ailesine ait ortak kullandıkları bilgisayarları vardır. 2 tanesinin evinde bilgisayar yokken 2 tanesi de bilgisayara internet kafe üzerinden erişebilmektedir.

Tablo 2’de katılımcılara uygulanan görüşme formundaki “*Algoritma ifadesini daha önce duydunuz mu? Cevabınız evetse kelime anlamını açıklayınız.*” sorusuna uygulama öncesi ve sonrasında öğrencilerin verdikleri cevaplar incelenmiştir.

Tablo 2

Algoritma ifadesine dair öğrencilerin görüşleri

Kategoriler	Uygulama Öncesi Frekanslar	Uygulama Sonrası Frekanslar
Duymadım	5	0
Günlük hayatla ilişkilidir	2	3

Tablo 2 (Devam)

Kategoriler	Uygulama Öncesi Frekanslar	Uygulama Sonrası Frekanslar
Duydum fakat ne anlama geldiğini bilmiyorum	4	0
Bir işin yapılması için gerekli adımlardır	6	1
Bir işin yapılmadan önce planlanması	0	7
Yapılması gereken işlerin sıralanması	0	6
Bir soruna yönelik çözüm yolu	0	2

İncelemeler sonucunda uygulama öncesi almayan öğrencilerin algoritma kavramını daha çok bir işin yapılması için gerekli adımlar (%35,29) olarak tanımlamış ve %29,42'i de algoritma kavramını bilmediğini ifade etmiştir. Uygulama sonrasında öğrencilerin ise algoritma kavramını daha çok bir işin yapılmadan önce planlanması (%41,11) olarak ve yapılması gereken işlerin sıralanması (%35,29) olarak tanımlanmıştır. Uygulama sonrasında öğrencilerin cevaplarında nispeten bir farklılığın olduğu söylenebilir.

Tablo 3'te katılımcılara uygulanan görüşme formundaki “Programlama hakkında ne biliyorsunuz” sorusuna uygulama öncesi ve sonrası öğrencilerin verdikleri cevaplar incelenmiştir.

Tablo 3

Bilgisayar ve Programlama hakkında öğrencilerin görüşleri

Kategoriler	Uygulama Öncesi Frekanslar	Uygulama Sonrası Frekanslar
Gelecekte daha çok ilerleyecek.	2	0
Oyun ve günlük işler için kullanılır.	3	0
Gerekli teknik bilgileri biliyorum.	6	7
Algoritma kurmayı biliyorum.	2	0
Robotik kodlama yapabiliyorum.	0	6
Scratch ile programlama yapabilirim.	0	7
mBlok ile programlama yapabilirim.	0	3
TinkerCat ile Arduino devre simülasyonu yapabilirim.	0	4
Bilgi sahibi olmayanlar.	5	0

İnceleme sonucunda uygulama öncesinde öğrencilerin daha çok bilgisayar hakkında gerekli teknik bilgileri (işlemci, ram vs.) bildiklerini düşündükleri (%33,3) veya bilgisayar hakkında yeterince bilgi sahibi olmadıkları (%27,77) görülmüştür. Uygulama sonrasında ise öğrencilerin daha çok bilgisayar hakkında gerekli teknik bilgiyi (ram, işlemci vs.) bildiklerini düşündüklerini (%25,92), bilgisayar hakkında Scratch ile programlama yapmayı bildiklerini düşündüklerini (%25,92) ve robotik kodlama yapabileceklerini düşündükleri (%22,22) görülmüştür. Uygulama sonrasında öğrencilerin cevaplarında nispeten bir farklılığın olduğu söylenebilir.

Tablo 4’te katılımcılara uygulanan görüşme formundaki “*Bildiğiniz programlama dilleri var mı, varsa hangileri?*” sorusuna uygulama öncesinde ve sonrasında öğrencilerin verdikleri cevaplar incelenmiştir.

Tablo 4

Öğrencilerin bildikleri programlama dilleri

Kategoriler	Uygulama Öncesi	Uygulama Sonrası
Bilmiyorum	12	4
Bazı programlama dillerini duydum	3	11
Scratch kodlayabiliyorum	0	3
mBlock kodlayabiliyorum	0	5

İncelemeler sonucunda uygulama öncesinde öğrencilerin daha çok hiçbir programlama dilinin adını bilmediği (%75) görülürken uygulama sonrasında öğrencilerin ise daha çok bazı programlama dillerinin isimlerini (C, C++, Java vs.) bildikleri (%47,82) görülmüştür. Uygulama sonrasında öğrencilerin cevaplarında nispeten bir farklılığın olduğu söylenebilir.

Katılımcılara uygulanan görüşme formunda yer alan “*Sizce matematik eğitiminde yazılım kullanılabilir mi?*” Cevabınız evetse i) “*Matematik eğitiminizi kolaylaştırır mı, zorlaştırır mı açıklayınız?*” ii) “*Ne gibi yararlar sağlayabileceğini düşünüyorsunuz?*” soruları sorulmuştur.

Tablo 5’de “*Sizce matematik eğitiminde yazılım kullanılabilir mi?*” sorusuna uygulama öncesinde ve sonrasında öğrencilerin verdikleri cevaplar incelenmiştir.

Tablo 5

Matematik eğitiminde yazılımın kullanılmasına yönelik öğrencilerin görüşleri

Kategoriler	Uygulama Öncesi Frekanslar	Uygulama Sonrası Frekanslar
Kullanılır	14	15
Kullanılmaz	1	0

İnceleme uygulama öncesinde öğrencilerin sadece bir tanesi kullanılamaz cevabını verirken kalan tüm öğrenciler kullanılır (%93,33) cevabını vermiştir. Uygulama sonrasında öğrencilerin ise hepsi matematik eğitiminde yazılım kullanılır (%100) cevabını vermiştir. Uygulama sonrasında öğrencilerin sadece birinin cevabı değiştiği için anlamlı bir fark yoktur.

Tablo 6’te “*Sizce matematik eğitiminde yazılım kullanılabilir mi?*” sorusuna “*kullanılabilir*” diyen öğrencilerin “i) *Matematik eğitiminizi kolaylaştırır mı, zorlaştırır mı açıklayınız?*” sorusuna uygulama öncesinde ve sonrasında öğrencilerin verdikleri cevaplar incelenmiştir.

Tablo 6

Yazılımın matematik eğitimini kolaylaştırıp zorlaştırmasına ilişkin öğrencilerin görüşleri

Kategoriler	Uygulama Öncesi Frekanslar	Uygulama Sonrası Frekanslar
Yazılım ile matematik sevebilir	3	6
Yazılım ile matematik ilişkilidir	1	5
Kişiye göre değişir	3	0
Görsel olarak yardımcı olur	0	3

Tablo 6 (Devam)

Kategoriler	Uygulama Öncesi Frekanslar	Uygulama Sonrası Frekanslar
Algoritma ile matematik ilişkilidir.	0	1
Soru çözmede yardımcı olur	0	1
(Sadece) Kolaylaştırabilir ama bilgim yok diyenler	7	3
Konu ile alakalı bilgi sahibi olmayanlar	1	0

İncelemeler sonucunda öğrencilerin daha çok, matematik eğitiminde yazılım kullanımının matematik öğrenmelerini kolaylaştırabileceği ama bunun nasıl yapılacağı ile ilgili bilgisinin olmayanlardan (46,66) oluşmaktadır. Uygulama sonrasında ise öğrencilerin daha çok yazılım ile matematiğin sevebileceğini düşünenlerden (%31,57) ve yazılım ile matematiğin ilişkili (%26,31) olduğunu düşünenlerden oluştuğu görülmüştür. Uygulama sonrasında öğrencilerin cevaplarında nispeten bir farklılığın olduğu söylenebilir.

Tablo 7’te “Sizce matematik eğitiminde yazılım kullanılabilir mi?” sorusuna “kullanılabilir” diyen öğrencilerin “ii) Ne gibi yararlar sağlayabileceğini düşünüyorsunuz?” sorusuna uygulama öncesinde ve sonrasında öğrencilerin verdikleri cevaplar incelenmiştir.

Tablo 7

Yazılımın matematik eğitime ne gibi yararlar sağlayabileceğine dair öğrencilerin görüşleri

Kategoriler	Uygulama Öncesi Frekanslar	Uygulama Sonrası Frekanslar
Geogebra ile daha kolay öğrenilir	1	3
Matematiği öğrenmemizi kolaylaştırır.	5	12
Yarar sağlamaz.	3	0
Matematiksel şekilleri görselleştirir	0	3
Bilmiyorum	6	1
Matematiği daha eğlenceli hale getirir	0	3
Matematiğe farklı bir bakış açısı sağlar.	0	1

İncelemeler sonucunda uygulama öncesinde öğrencilerin daha çok konuya dair bir bilgilerinin olmadığı gözlemlenmiş (%40) ve matematik dersini kolaylaştıracağını (%33,33) düşünenlerden oluştuğu görülmüştür. Uygulama sonrasında ise daha çok matematik eğitiminde yazılım kullanımının matematik dersini öğrenmemizi kolaylaştırır (%52,17) diyenlerden oluştuğu görülmüştür. Uygulama sonrasında öğrencilerin cevaplarında nispeten bir farklılığın olduğu söylenebilir.

Tablo 8’da katılımcılara uygulanan görüşme formundaki “Bir yazılımcı olsaydınız hayatı kolaylaştıracak nasıl bir program yazardınız?” sorusuna uygulama öncesinde ve sonrasında öğrencilerin verdikleri cevaplar incelenmiştir.

Tablo 8*Öğrencilerin yazılımcı olmaları halinde ne yapabileceklerine dair görüşleri*

Kategoriler	Uygulama Öncesi Frekanslar	Uygulama Sonrası Frekanslar
Program tasarlayanlar	9	12
Bilmiyorum diyenler	6	3

İncelemeler sonucunda uygulama öncesi öğrencilerin bir yazılımcı olmaları durumunda yazacakları programları görüşme sırasında öğrencilerin %60'ı bir program tasarlayabilirken %40'ı tasarlayamamıştır. Uygulama sonrasında ise görüşme sırasında %80'i program tasarlayabilirken %20'si tasarlayamamıştır. Bilgisayar ve Arduino Programlama kursunu alan öğrencilerin cevaplarında nispeten bir farklılığın olduğu söylenebilir. Bu soruya uygulama öncesi ve sonrasında bazı öğrenciler tarafından yaratıcı cevaplar verilmiştir. Bunlardan bazıları şu şekildedir: “Çay servisi yapan akıllı bir robot tasarladım.”, “Kapılara teknolojik aletlerle çip sistemi uyguladım. Elimize çip yerleştirerek kapıya o çipi okuttuğumuzda kapı açılsın.”, “Evden VR gözlük ile canlı derse katılacak bir sistem kurardım.”

İkinci Alt Probleme İlişkin Alt Bulgular

Çalışmanın “Teknolojiye Hazırlık Ölçeği”nde uygulama öncesinde ve sonrasında anlamlı fark oluşmakta mıdır?” alt problemine yönelik elde edilen veriler aşağıda sunulmuştur. Bu bölümde THÖ’den elde edilen verilerin betimsel istatistiklerine yer verilmiştir. Bu istatistikler öğrencilerin öntest – sontest puanlarına ve ölçeğin alt boyutlarına göre yorumlanmıştır. Araştırma kapsamında alınan öğrencilerin verdiği cevaplar doğrultusunda THÖ’nden ve alt boyutlarından elde edilen ortalama puanları, standart sapmaları ve düzeyleri göz önünde bulundurularak incelenmiştir. Düzey değerleri hesaplanırken ortalama puan değerinden bir standart sapma değeri altında bulunan aralığa denk gelen değerler düşük düzey, bir standart sapma değeri altında ve üstünde bulunan aralığa denk gelen değerler orta düzey, bir standart sapma değeri üstünde bulunan değerleri ise yüksek düzey olacak şekilde kategorileştirilmiştir. Bu kategorileşme çerçevesinde elde edilen bulgular Tablo9’da gösterilmiştir.

Tablo 9*Öğrencilerin öntest-sontest sonuçlarına ait betimleyici istatistik değerleri*

	Boyut	Sınıf	N	Madde Sayısı	$\bar{X}$	Düzey	SS
Öntest	İyimserlik	7	15	7	3.657	Orta	0,776
	Yenilikçilik	7	15	6	3.400	Orta	0,632
	Güvensizlik	7	15	6	3.155	Orta	0,834
	Huzursuzluk	7	15	4	3.183	Orta	0,752
	Toplam	7	15	23	3.376	Orta	0.617
Sontest	Boyut	Sınıf	N	Madde Sayısı	$\bar{X}$	Düzey	SS
	İyimserlik	7	15	7	3,914	Orta	0,520
	Yenilikçilik	7	15	6	3,733	Orta	0,537
	Güvensizlik	7	15	6	3,300	Orta	0,652
	Huzursuzluk	7	15	4	3,716	Orta	0,549
	Toplam	7	15	23	3,675	Orta	0,218

Tablo 9 verileri incelendiğinde öğrencilerin Teknolojiye Hazırlık düzeyleri alt boyutlarının öntest sonuçlarına göre en yüksek ortalamanın İyimserlik ($\bar{X}=3.657$) alt faktörüne, en düşük ortalamanın ise Güvensizlik ($\bar{X}=3.255$) alt faktörüne ait olduğu görülmektedir. Yine öğrencilerin teknolojiye hazırlık algı düzeyleri alt boyutlarının sontest sonuçlarına göre en yüksek ortalamanın İyimserlik ($\bar{X}=3.914$) alt faktörüne, en düşük ortalamanın ise Huzursuzluk ($\bar{X}=3.300$) alt faktörüne ait olduğu görülmektedir. Ayrıca öntest verilerinden elde edilen genel aritmetik ortalama $\bar{X}=3.376$ ve standart sapma $ss=0,617$ olarak hesaplanmıştır. Bu veriler

ışığında öğrencilerin teknolojiye hazırlık düzeylerinin öntest verilerine göre orta düzeyde olduğu görülmektedir. Yine öğrencilerin sontest verilerinden elde edilen aritmetik ortalama $\bar{X}=3.675$, standart sapma $s=0.218$ olarak hesaplanmıştır. Bu sonuçlara göre öğrencilerin genel teknolojiye hazırlık düzeylerinin yine orta düzeyde olduğu ancak ortalamalar göz önüne alındığında teknolojiye hazırlık düzeylerinin yükseldiği ($\bar{X}_{\text{sontest}} = 3.675 > \bar{X}_{\text{öntest}} = 3.376$) söylenebilir.

Öğrencilerin THÖ'den Elde Edilen Verilerin Wilcoxon İşaretli Sıralar Testi Sonuçları

Çalışmada öncelikle öğrencilerin THÖ' ne vermiş olduğu cevaplara göre öğrencilerin teknolojiye hazırlık düzeylerinin alt boyutlarının öntest-sontest sonuçlarına göre anlamlı bir farklılık gösterip göstermediği Wilcoxon İşaretli Sıralar Testi uygulanarak incelenmiştir. Bu test sonucunda elde edilen bulgular Tablo 10'da gösterilmiştir.

Tablo 10

Deney Öncesi ve Sonrası Teknolojiye Hazırlık Ölçeği Puanlarının Wilcoxon İşaretli Sıralar Testi Sonuçları

Sontest-Öntest	N	Sıra Ortalaması	Sıra Toplamı	Z	p
Negatif Sıra	7	5	35	-1,422	0,155
Pozitif Sıra	8	10,63	85		
Eşit	0	-	-		

Tablo 10'da verilen test sonuçlarına göre, araştırmaya katılan öğrencilerin THÖ'ne verdikleri cevapların kurs öncesi ve sonrası cevaplar arasında anlamlı bir fark olmadığı görülmektedir ($z=1,422$; $p>0.05$). Bununla birlikte öğrencilerin sontestte verdikleri cevapların ortalamaları ($\bar{X}=3.675$), öntestte verdikleri cevapların ortalamalarından ($\bar{X} = 3.376$) fazla olduğu ($\bar{X}_{\text{sontest}}=3.675 > \bar{X}_{\text{öntest}}=3.376$) görülmektedir.

Üçüncü Alt Probleme İlişkin Alt Bulgular

Çalışmanın “Dijital Okuryazarlık Ölçeği”nde uygulama öncesinde ve sonrasında anlamlı fark oluşmakta mıdır?” alt problemine yönelik elde edilen veriler aşağıda sunulmuştur. Bu bölümde DOÖ'den elde edilen verilerin betimsel istatistiklerine yer verilmiştir. Bu istatistikler öğrencilerin öntest – sontest puanlarına ve ölçeğin alt boyutlarına göre yorumlanmıştır. Araştırma kapsamında alınan öğrencilerin verdiği cevaplar doğrultusunda DOÖ'den ve alt boyutlarından elde edilen ortalama puanları, standart sapmaları ve düzeyleri göz önünde bulundurularak incelenmiştir. Düzey değerleri hesaplanırken ortalama puan değerinden bir standart sapma değeri altında bulunan aralığa denk gelen değerler düşük düzey, bir standart sapma değeri altında ve üstünde bulunan aralığa denk gelen değerler orta düzey, bir standart sapma değeri üstünde bulunan değerleri ise yüksek düzey olacak şekilde kategorileştirilmiştir. Bu kategorileşme çerçevesinde elde edilen bulgular Tablo11'de gösterilmiştir.

Tablo 11

Öğrencilerin öntest-sontest sonuçlarına ait betimleyici istatistik değerleri

	Boyut	Sınıf	N	Madde Sayısı	$\bar{X}$	Düzey	SS
Öntest	Tutum	7	15	7	3,438	Orta	1,166
	Teknik	7	15	6	3,522	Orta	0,991
	Bilişsel	7	15	2	3,466	Orta	0,801
	Sosyal- Duygusal	7	15	2	3,100	Orta	0,929
	Toplam	7	15	17	3.41	Orta	0.992

	Boyut	Sınıf	N	Madde Sayısı	$\bar{X}$	Düzey	SS
Sontest	Tutum	7	15	7	3,609	Orta	0,926
	Teknik	7	15	6	3,655	Orta	1,075
	Bilişsel	7	15	2	4,000	Orta	0,801
	Sosyal- Duygusal	7	15	2	3,400	Orta	1,105
	Toplam	7	15	17	3,647	Orta	0,825

Tablo 11 verileri incelendiğinde öğrencilerin dijital okuryazarlık düzeyleri alt boyutlarının öntest sonuçlarına göre en yüksek ortalamanın Teknik ($\bar{X}=3,522$) alt faktörüne, en düşük ortalamasının ise Sosyal-Duygusal ($\bar{X}=3.100$) alt faktörüne ait olduğu görülmektedir. Yine öğrencilerin dijital okuryazarlık düzeyleri alt boyutlarının sontest sonuçlarına göre en yüksek ortalamanın bilişsel ($\bar{X}=4.000$) alt faktörüne, en düşük ortalamasının ise Sosyal-Duygusal ($\bar{X}=3.400$) alt faktörüne ait olduğu görülmektedir. Ayrıca öntest verilerinden elde edilen genel aritmetik ortalama $\bar{X}=3.431$ ve standart sapma $ss=0,992$ olarak hesaplanmıştır. Bu veriler ışığında öğrencilerin dijital okuryazarlık düzeylerinin öntest verilerine göre orta düzeyde olduğu görülmektedir. Yine öğrencilerin sontest verilerinden elde edilen aritmetik ortalama $\bar{X}=3.647$, standart sapma $s=0.825$ olarak hesaplanmıştır. Bu sonuçlara göre öğrencilerin genel dijital okuryazarlık düzeylerinin yine **orta** düzeyde olduğu ancak ortalamalar göz önüne alındığında dijital okuryazarlık düzeylerinin yükseldiği ($\bar{X}_{\text{sontest}}=3.647 > \bar{X}_{\text{öntest}}=3.431$) söylenebilir.

Öğrencilerin DOÖ'den Elde Edilen Verilerin Wilcoxon İşaretli Sıralar Testi Sonuçları

Çalışmada öncelikle öğrencilerin DOÖ'ne vermiş olduğu cevaplara göre öğrencilerin dijital okuryazarlık düzeylerinin alt boyutlarının öntest-sontest sonuçlarına göre anlamlı bir farklılık gösterip göstermediği Wilcoxon İşaretli Sıralar Testi uygulanarak incelenmiştir. Bu test sonucunda elde edilen bulgular Tablo 12'de gösterilmiştir.

Tablo 12

Deney Öncesi ve Sonrası Teknolojiye Hazırlık Ölçeği Puanlarının Wilcoxon İşaretli Sıralar Testi Sonuçları

Sontest-Öntest	N	Sıra Ortalaması	Sıra Toplamı	Z	P
Negatif Sıra	4	7,13	28,50	-1,792	0,073
Pozitif Sıra	11	8,32	91,50		
Eşit	0	-	-		

Tablo 12'de verilen test sonuçlarına göre, araştırmaya katılan öğrencilerin DOÖ'ne verdikleri cevapların kurs öncesi ve sonrası cevaplar arasında anlamlı bir fark olmadığı görülmektedir ($z=1,792$; $p>0.05$). Bununla birlikte öğrencilerin sontestte verdikleri cevapların ortalamaları ($\bar{X}=3,647$), öntestte verdikleri cevapların ortalamalarından ($\bar{X} = 3.431$) yüksek olduğu ($\bar{X}_{\text{sontest}}=3.647 > \bar{X}_{\text{öntest}}=3.431$) görülmektedir.

SONUÇ VE ÖNERİLER

Öğrencilerin uygulama öncesinde görüşme formuna verdikleri yanıtlardan bilgisayarı çoğunlukla günlük işlerinde kullandıkları, yazılım, programlama, robotik kodlama, Web 2.0 araçları (Geogebra) gibi konularda yeterince bilgi sahibi olmadıkları anlaşılmaktadır. Buradan hareketle öğrencilerin bilgisayar ve programlamaya olan farkındalıklarının az olabileceği düşünülmektedir. Gezgın, Azaz ve Atabey (2022) tarafından yapılan bir çalışmada ortaokul öğrencilerinin robotik ve kodlama eğitimi; öğrencilerin işbirlikli öğrenmeye karşı tutumlarını, problem çözme becerileri algılarını ve kişilik özelliklerine olan etkisi incelenmiştir. Araştırma sonucunda, öğrencilerin işbirlikli öğrenmeye yönelik tutumlarının ve problem çözme becerisi algı seviyelerinin arttığı anlaşılmıştır. Buradan hareketle robotik kodlamaya ve dolayısıyla bilgisayar programcılığı eğitimi öğrencilerdeki problem çözme becerilerine katkı sağladığı gözlenmiştir.

Öğrencilerin görüşme formuna verdikleri yanıtlarda bilgisayar ve programlama hakkında bilgi sahibi olmadıkları, varsa bile bilgisayarı günlük işlerde kullandıkları görülürken çalışma sonucunda öğrencilerin, bilgisayar ve programlama hakkında Scratch ve mBlock ile kodlama yapılabildiğini, robotik kodlama yapılabildiğini ve Tinkercat üzerinden Arduino simülasyonu yapabildiklerini ifade ettikleri görülmüştür. Böylece öğrenciler uygulama sürecinde bilgisayar ve programlama hakkında bilgi sahibi oldukları ve bir farkındalığa ulaştıkları gözlemlenmiştir. Sırakaya (2018) tarafından kodlama eğitime yönelik öğrenci görüşlerinin alındığı bir çalışmada ortaokul öğrencilerinin blok temelli kodlama eğitime yönelik görüşleri

incelenmiştir. Bu çalışmanın sonucunda öğrencilerin kodlama eğitimi almaktan keyif aldıkları ve bu eğitimleri ilginç ve eğlenceli olarak değerlendirdikleri gözlenmiştir. Fessakis vd. (2013) tarafından yapılan başka bir çalışmada 5-6 yaşındaki anaokulu çocuklarına bilgisayar programlaması öğretmek problem çözme becerilerini geliştirmeyi amaçlamışlardır. Çalışmanın sonunda anaokulu çocuklarının bilgisayar programlaması yoluyla problem çözme, matematiksel düşünme ve sosyal becerilerini geliştirebileceğini ve bu sürecin öğretmen rehberliğinde etkili şekilde uygulanabileceğini gösterilmiştir.

Yazılımın matematik eğitimine olan katkısı hakkında öğrencilerin çoğu uygulama öncesinde yeteri kadar bilgi sahibi olmadığı için kullanılabileceğini düşünmüş ama nasıl kullanılacağı hakkında herhangi bir bilgilerinin olmadığı gözlemlenmiştir. Uygulama sonrasında ise kodlama ile matematik eğitiminin algoritmik zekâyı geliştirdiği, geometrik şekillerin görsel olarak ifade edilmesinin daha kolay olacağını ifade etmişlerdir. Böylece öğrencilerin uygulama sürecinde yazılım ile matematiğin ilişkisini kavradıkları ve matematik eğitiminde yazılım kullanımının ne gibi fayda sağlayabileceğini düşündükleri gözlemlenmiştir. Bu sonuçlar Mercan ve Aktaş'ın (2018) yaptığı çalışma ile benzerlik göstermektedir. Ortaokul 6. sınıf Matematik dersine ait "Cebirsel İfadeler" konusunun, ders kitabındaki etkinliklerden farklı olarak Scratch yardımıyla matematik öğretiminin öğrencilerdeki akademik başarıya etkisini incelemiştir. Araştırma sonucunda, Scratch ile tasarlanan oyun ve etkinliklerin hem öğretmenlerin hem de öğrencilerin ders başarılarına olumlu katkı sağladığı görülmüştür. Öğrenilmesi zor ve sıkıcı olan konuların öğretiminde kullanılabilir (Mercan & Aktaş, 2018). Totan (2021) tarafından yapılan başka bir çalışmada ise blok tabanlı kodlama eğitimlerinin 5.sınıf kademesindeki öğrencilerin bilgi işlemsel düşünme becerileri ve kodlamaya yönelik tutumlarına etkisi Blocky kodlama ortamı kullanılarak incelenmiştir. Araştırmanın sonunda blok tabanlı kodlama eğitiminde Blocky ortamının, öğrencilerin bilgi işlemsel düşünme becerileri öz yeterlik algıları ve kodlama öğrenmeye yönelik tutumları üzerinde olumlu bir etkiye sahip olduğu söylenebilir. Momcilovic (2020) tarafından yapılan başka bir çalışmada ise temel geometrik şekillerin geleneksel yöntemlerle ve Scratch kullanılarak öğretiminde başarı düzeyleri arasında bir farkın olup olmadığı incelenmiştir. Çalışmanın sonunda ise geometrinin temel kavramlarını model ve cisimler üzerinde algılama ve tanıma yöntemiyle öğrenen öğrenciler ile Scratch tabanlı programları kullanarak öğrenen öğrenciler arasında istatistiksel olarak anlamlı bir başarı farkı olduğunu ortaya konulmuştur.

Belirli bir problemin çözümüne ulaşmak amacıyla tasarlanan sistematik ve adım adım ilerleyen işlem dizisine algoritma denir (Arabacıoğlu vd., 2007). Matematik gibi sayısal derslerde karşılaşılan problemlerin çözümünde algoritmaları kullanmak çözümü kolaylaştıracaktır. Öğrencilerin matematik problemlerinde çözüme ulaşmak için adım adım izledikleri süreçler de algoritmik bir yaklaşımdır (Jeyabal vd., 2016). Öğrencilerin algoritma bilgisinin ölçüldüğü görüşme formu sorusuna uygulama öncesinde öğrencilerin çoğu algoritma ifadesini çoğu daha önce isim olarak duyduğunu fakat ne anlama gelmediğini veya hiç duymadıklarını ifade ederken uygulama sonrasında öğrencilerin hepsinin temel düzeyde algoritma ifadesinin ne demek olduğu hakkında yorum yapabildiği gözlemlenmiştir. Konu ile ilgili literatür incelendiğinde Durası (2021) tarafında yapılan bir araştırmada 6. sınıf öğrencilerinin Scratch programı ile tam sayılar konusunda algoritma üretme süreçlerini ve oluşturdukları algoritmaların yapıları incelenmiştir. Araştırma sonucunda, öğrencilerin üç etkinlikte algoritma üretme süreçlerinde tanıma, kullanma ve oluşturma eylemlerini dinamik bir şekilde gerçekleştirdikleri gözlemlenmiştir. Lehmann (2024) tarafından yapılan başka bir çalışmada 12. sınıf öğrencilerinin algoritmikleştirme görevlerinin algoritmik düşünmeye nasıl dahil edildiklerini incelenmiştir. Çalışmanın sonunda öğrencilerin algoritma tasarlama süreçlerinde ilerleme kaydettikleri gözlenmiştir.

Araştırmaya katılan öğrencilerin Teknolojiye Hazırlık Ölçeği'nden (THÖ) aldıkları deney öncesi ve sonrası puanlar arasında anlamlı bir farklılık olmadığı gözlemlenmiştir. Fakat sontest puanlarının ortalaması öntest ortalamasından yüksek çıktığı görülmektedir. Araştırmaya katılan öğrencilerin Dijital Okuryazarlık Ölçeği'nden (DOÖ) aldıkları deney öncesi ve sonrası puanlar arasında anlamlı bir fark olmadığı gözlemlenmiştir. Fakat sontest puanlarının ortalaması öntest ortalamasından yüksek çıktığı görülmektedir.

Araştırma sonucunda Bilgisayar ve Arduino Programlama kursu alan öğrencilerin bilgisayara, bilgisayar bilimlerine, robotik kodlamaya, algoritma kurmaya ve tüm bunların matematikle olan ilişkilerini öğrendikleri görülmüştür. Bu sonuç Hangün'ün(2019) yaptığı çalışma ile benzerlik göstermektedir. Bu çalışmada robot

programlama eğitiminin öğrencilerin matematik başarı, matematik kaygı, programlama özyeterlilik STEM tutum değişkenleri üzerindeki etkisi incelenmiştir. Çalışmanın sonucunda robot uygulamalarının matematik eğitiminde etkili bir yöntem olduğu ve derslerin verimini arttırabileceği düşünülmüştür (Hangün,2019). Süreç içerisinde bilgisayar hakkında çok az bilgisi olan öğrencilerin bile sürecin sonunda Scratch üzerinden kodlama yaptığı, mBlock üzerinden Arduino programladığı vs. gözlemlenmiştir. Tüm bunların yanında matematik dersinin sadece bir dersle sınırlı kalmadığı, matematiğin başka alanlarla da ilişkili olduğu öğrenciler tarafından anlaşıldığı gözlemlenmiştir. Algoritma ifadesinin ne olduğu süreç içerisinde öğrenciler tarafından benimsenmiş olup algoritma ifadesinin günlük hayatın içinde yer edindiği de öğrenciler tarafından belirtilmiştir.

Tüm bunların yanında uygulama sonunda öğrencilerle birlikte arduino ile mini CNC Plotter yapımında öğrencilerin bilgisayar ve arduino ile neleri yapabileceklerini görecekları bir prototip tasarlanmıştır. Böylece öğrencilere bir teknolojik bir farkındalık oluşturulmuş ve öğrencilerin bilgisayara, yazılıma, robotik kodlamaya, algoritmaya ve tüm bunların matematikle olan ilişkisine dair bakış açılarına olumlu katkı sağladığı anlaşılmıştır. Shankar vd. (2013) tarafından 9. sınıf öğrencileriyle birlikte düşük maliyetli robotlar yapmak, programlamak ve çeşitli geometrik şekilleri çizdikleri bir çalışma yapılmıştır. Bu çalışmada öğrencilerin mühendislik ve matematiğe olan ilgilerini artırmak amaçlanmıştır. Robotlar matematik dersine entegre edilmiş ve sonuç olarak öğrencilerin robot kullanımının matematik derslerinde fayda sağladığı gözlemlenmiştir. Öğrenciler Pisagor teoremi ile çözülmüş ve ardından robot simülasyonu ile sonuçlar analiz edilmiş. Öğrenciler hem matematik problemi hem de robotları kullanarak ortaya çıkan öğrenci hataları yorumlanmıştır. Böylece öğrenciler kavramsal hata ile gerçek dünya hatası arasındaki farkı anlamışlardır. Literatürde yapılan bu çalışma, yaptığım çalışmayı destekler niteliktedir.

Yapılan çalışmada öğrenciler, yazılımın matematik dersine entegre edilmesi, algoritma kurmanın matematik öğrenmemize katkı sağladığını belirtmişlerdir. Ek olarak ortaokul matematik müfredatında yer alan geometrik kavramların daha iyi anlaşılması için dinamik geometri yazılımlarının daha sık kullanılması gerektiğini belirtmişlerdir.

Sonuç olarak öğrencilerin görüşme formuna verdikleri yanıtlara göre bilgisayar programlama ve yazılım ile öğrencilerin matematiği öğrenmeleri arasında öğrencilerin yakın ilişki kurduğu ve matematiğe olan bakışlarını etkilediği ortaya çıkmıştır. Bu bağlamda okullardaki matematik derslerine ders sırasında öğrencilerin algoritmik düşünce yapılarını geliştirecek etkinliklerin sayıca artırılması, Scratch gibi blok tabanlı uygulamaların matematik etkinliklerine daha fazla ilave edilmesi ve öğrencilerin görsel matematiksel okuryazarlığı artırma etkisine sahip dinamik geometri yazılımları kullanılması önerilmektedir.

NOT

Bu çalışma TUBİTAK 2209-A Üniversite Öğrencileri Araştırma Projeleri Destekleme Programı tarafından desteklenmiştir. Aynı zamanda 16. Ulusal Fen Bilimleri ve Matematik Eğitimi Kongresi (UFBMEK 2024) sempozyumunda sunulan bildirinin geniş halidir.

KAYNAKÇA

- Agar, J. (2019). Teknoloji nedir? Teknoloji: Bir kavramın eleştirel tarihi, Eric Schatzberg, Chicago ve Londra, Chicago Üniversitesi Yayınları, 2018, 352 sayfa, 27,45 \$ (kağıt kapaklı), ISBN: 978-0-226-58383-9. *Annals of Science*, 77 (3), 377–382. <https://doi.org/10.1080/00033790.2019.1672788>
- Akbulut, İ., Akyıldız, A., Yılmaz, S., Bayri, E., & Bayri, G. (2020). Eğitimde Teknoloji Kullanımının Öğrenci Başarısına Etkisi. *International QMX Journal*, 6. <https://doi.org/10.5281/zenodo.10729254>
- Antoino, J., Tudela, P. A., & Terron, P. D. (2024, 10). Computational Thinking and Programming with Arduino in Education: : A Systematic Review for Secondary Education. Heliyon.
- Arabacıoğlu, T., Bülbül, H. İ., & Filiz, A. (2007). Bilgisayar programlama öğretiminde yeni bir yaklaşım. IX. akademik bilişim konferansı bildirileri (s.193-197). Dumlupınar Üniversitesi, Kütahya.
- Atış Akyol, N., & Aşkar, N. (2022). Erken Çocukluk Döneminde 21. Yüzyıl Becerileri. Gazi Üniversitesi Gazi Eğitim Fakültesi Dergisi, 42(3), 2597-2629. <https://doi.org/10.17152/gefad.1081472>
- Bakırcı, G., & Özmantar, M. F. (2023). Matematik Eğitiminde Etkinlik Temelli Öğretim: Öğretmenlerin Etkinlik Değerlendirmeleri. Dokuz Eylül Üniversitesi Buca Eğitim Fakültesi Dergisi(58), 3158-3184. <https://doi.org/10.53444/deubefd.1359320>
- Bal-Sezerel, B. (2012). Seçici problem çözme (SPÇ) tekniğinin ilköğretim 6. ve 7. sınıf öğrencilerine yönelik matematik eğitimindeki sosyal geçerliğinin değerlendirilmesi. [Yüksek lisans tezi, Anadolu Üniversitesi].
- Belmonte, J. L., Marin, J. A. M., Costa, R. S., & Guerrero, A. J. M. (2020). Arduino Advances in Web of Science. A Scientific Mapping of Literary Production. IEEE Access, 6.
- Bernard, M., & Setiawan, W. (2020, October). Developing math games media using scratch language. In *Journal of Physics: Conference Series* (Vol. 1657, No. 1, p. 012064). IOP Publishing. <https://doi.org/10.1088/1742-6596/1657/1/012064>
- Büyüköztürk, Ş. (2020). Sosyal Bilimler İçin Veri Analizi El Kitabı . Ankara: Pegem Akademi.
- Büyüköztürk, Ş., Çakmak, K.E., Akgün, Ö.E., Karadeniz, Ş., & Demirel, F. (2010). Bilimsel araştırma yöntemleri. Ankara: Pegem Akademi.
- Cameron, K. (2024). *Progressive Education and the Quality Agenda: Contradictory Views of Children, Childhood, and the Purposes of Education for Young Children* [Yayımlanmamış Doktora Tezi].
- Cheng, Y. P., Lai, C. F., Chen, Y. T., Wang, W. S., Huang, Y. M., & Wu, T. T. (2023). Enhancing student's computational thinking skills with student-generated questions strategy in a game-based learning platform. *Computers & Education*, 6. <https://doi.org/10.1016/j.compedu.2023.104794>
- Creswell, J.W. (2006). Understanding Mixed Methods Research, (Chapter 1). Available At: http://Www.Sagepub.Com/Upm-Data/10981_Chapter_1.Pdf (Erişim Tarihi: 02/03/2022)
- Durası, E. (2021). *Akademik Başarısı Yüksek 6. Sınıf Öğrencilerin Scratch Programı ile Tamsayılar Konusunda Algoritma Üretme Süreçleri ve Yapılarının İncelenmesi* [Yayımlanmamış Yüksek Lisans Tezi].
- Fessakis, G., Gouli, E., & Mavroudi, E. (2013). Problem solving by 5–6 years old kindergarten children in a computer programming environment: A case study. *Computers & Education*. <https://doi.org/10.1016/j.compedu.2012.11.016>
- Gezgin, D. M., Azaz, E., & Atabey, E. (2022). Ortaokul Öğrencilerinin Robotik ve Kodlama Eğitimi Başarılarına İşbirlikli Öğrenme Tutumu, Problem Çözme Becerisi Algısı ve Kişilik Tiplerinin Etkisi: Bir Nedensel Karşılaştırma Araştırması. *Öğretim Teknolojileri ve Öğretmen Eğitimi Dergisi*, 28-42.
- Günay, D. (2017). Teknoloji Nedir? Felsefi Bir Yaklaşım. *Yüksek Öğretim ve Bilim Dergisi*, 164.

- Hamutoğlu, N. B., Güngören, Ö. C., Uyanık, G. K., & Erdoğan, D. G. (2017). Dijital Okuryazarlık Ölçeği: Türkçe'ye Uyarlama Çalışması. *Ege Eğitim Dergisi*, 18(1), 408- 429.
- Hangün, M. E. (2019). *Robot Programlama Eğitiminin Öğrencilerin Matematik Başarısına, Matematik Kaygısına, Programlama Özyeterliliğine ve Stem Tutumuna Etkisi* [Yayımlanmamış Yüksek Lisans Tezi].
- Iskrenovic-Momcilovic, O. (2020). Improving Geometry Teaching with Scratch. *International Electronic Journal of Mathematics Education*, 15(2), em0582. <https://doi.org/10.29333/iejme/7807>
- Işık, A., Çiltaş, A., & Bekdemir, M. (2008). Matematik eğitiminin gerekliliği ve önemi. *Atatürk Üniversitesi Kazım Karabekir Eğitim Fakültesi Dergisi*, (17), 174-184.
- Jacobsen, M, J Lock and S Friesen. 2013. "Strategies for Engagement: Knowledge Building and Intellectual Engagement in Participatory Learning Environments." *Education Canada* 53, no 1: npn.
- K. Jeyabal, M. Vasuki & A. Dinesh Kumar, "From Algorithms to Conceptual Understanding a Review of Instructional Strategies in Math", *Journal of Engineering Scientific Research and Applications*, Volume 2, Issue 1, January - June, Page Number 191-205, 2016.
- Kalelioglu, F., & Gülbahar, Y. (2014). The effects of teaching programming via Scratch on problem solving skills: A discussion from learners' perspective. *Informatics in Education*, 13(1), 33–50.
- Mehmet Kurudayıoğlu, Büşra Gociaoğlu .(0).21. Yüzyıl Becerileri Ve Türkçe Eğitimi .Nobel Yayıncılık
- Lehmann, T.H. How current perspectives on algorithmic thinking can be applied to students' engagement in algorithmatizing tasks. *Math Ed Res J* 36, 609–643 (2024). <https://doi.org/10.1007/s13394-023-00462-0>
- Lupas, L. B., Olchondra, J. A. A., Irinco, J. P. L., Nobis, M. L., & Bayri, G. (2024). Cracking the Math Code: Challenges and Strateg. *ETCOR*, 6.
- Marín, J. A. M., Tudela, P. A. G., & Terron, P. D. (2024). Computational thinking and programming with Arduino in education: A systematic review for secondary education. *Heliyon*, 6. <https://doi.org/10.1016/j.heliyon.2024.e29177>
- Meço, G., & Arı, A. G. (2021). Arduino Destekli STEM Etkinliklerine Yönelik Ortaokul Öğrencilerinin Görüşleri. *The Journal of International Social Research*, 498-504-505.
- Mercan, M., & Aktaş, M. (2018). 6. Sınıf Matematik Dersine Ait Cebirsel İfadeler Konusunun Scratch Destekli Öğretiminin Öğrenci Başarısına Etkisi. *Social Sciences Studies Journal*, 6395-6409.
- Miles, M. B. & Huberman, A.M. (1994). *Qualitative data analysis: an expanded sourcebook*. (2nd Edition). Calif.: SAGE Publications.
- Mollo, L., Bellotti, F., Berta, R., & Gorla, A. D. (2016). *Building Arduino-Based Tangible Serious Games for Elementary Mathematics and Physics*. Springer.
- Negrete, J. C. (2023). Arduino in agriculture and the teaching of mathematics. *Horticulture International Journal*. <https://doi.org/10.15406/hij.2023.07.00280>
- Negrete, J. C. (2024). Mechatronics for Teaching Mathematics. *Acta Scientific AGRICULTURE*.
- Ng, W. (2012). Can We Teach Digital Natives Digital Literacy?. *Computers & Education*, 59(3), 1065-1078.
- Olsson, J., & Granberg, C. (2022). Teacher-student interaction supporting students' creative mathematical reasoning during problem solving using Scratch. *Mathematical Thinking and Learning*. <https://doi.org/10.1080/10986065.2022.2105567>
- Özer, P. S., Eriş, E. D., & Özmen, Ö. N. T. (2011). Teknolojiye Hazırlık Düzeyinin Belirlenmesine Yönelik Bir Ölçek Uyarlama Çalışması. *Dokuz Eylül Üniversitesi İşletme Fakültesi Dergisi*, 12(2), 321-333.

- Parasuraman, A. (2000). Technology Readiness Index (Trı) A Multiple-İtem Scale To Measure Readiness To Embrace New Technologies. *Journal Of Service Research*, 2(4), 307-320.
- Rodríguez – Martínez, J. A., Gonzalez Calero, J. A., & Saez Lopez, J. M. (2020). Computational thinking and mathematics using Scratch: An experiment with sixth-grade students. *Interactive Learning Environments*, 6. <https://doi.org/10.1080/10494820.2019.1612448>
- Shankar, R., Ploger, D., Nemeth, A., & Hecht, S. A. (2013). Robotics: Enhancing Pre-College Mathematics Learning with Real-world Examples. *120th ASEE Annual Conference and Exposition* (s. 1-17). Florida: ASEE.
- Sırakaya, M. (2018). Kodlama Eğitimine Yönelik Öğrenci Görüşleri. *Ondokuz Mayıs University Journal of Education Faculty*, 37(2), 79-90. <https://doi.org/10.7822/omuefd.394649>
- Şahin, M. C., & Namlı, N. A. (2017). Algoritma Eğitiminin Problem Çözme Becerisi Üzerine Etkisi. *Recep Tayyip Erdoğan Üniversitesi Sosyal Bilimler Dergisi*, 136-140.
- Tekin, Y., & Keser, H. (2020). Matematik Öğretiminde Robotik Etkinlikler Kullanılmasının Başarıya Etkisi. *Akdeniz Eğitim Araştırmaları Dergisi*, 476-477-478-488.
- Tewes, A. (2019). *The Effects of Incorporating on Student Experience and Understanding of Middle School Mathematical Concepts* [Yayımlanmamış Yüksek Lisans Tezi].
- Totan, H. N. (2021). *Blok Tabanlı Kodlama Eğitiminin Ortaokul Öğrencilerinin Bilgi İşlemsel Düşünme Becerileri Ve Kodlama Öğrenimine Yönelik Tutumlarına Etkisi: Blocky Örneği* [Yüksek Lisans Tezi].
- Turan, B., & Yıldız, B. (2021). Scratch ile Kodlama Yaparken Matematik Öğreniyorum. 2. *Uluslararası Bilim, Eğitim, Sanat ve Teknoloji Sempozyumu* (s. 395-400). İzmir: Dokuz Eylül Üniversitesi Buca Eğitim Fakültesi.
- Tudela, P. A. G., & Marin, J. A. M. (2023). Use of Arduino in Primary Education: A Systematic Review. *Education Sciences*, 6. <https://doi.org/10.3390/educsci13020134>
- Türnüklü, A. (2000). Eğitim Bilim Araştırmalarında Etkin Olarak Kullanılabilecek Nitel Bir Araştırma Tekniği: Görüşme. *Kuram Ve Uygulamada Eğitim Yönetimi*, 547.
- Yıldırım, A., & Şimşek, H. (2021). Sosyal bilimlerde nitel araştırma yöntemleri. Ankara: Seçkin Yayıncılık
- Yükseltürk, E., & Altıok, S. (2015). Bilişim Teknolojileri Öğretmen Adaylarının Bilgisayar Programlama Öğretimine Yönelik Görüşleri. *Amasya Üniversitesi Eğitim Fakültesi Dergisi*, 4(1), 50-65. <https://doi.org/10.17539/aej.96735>
- W. Mo, Y. Zhang, Y. Fu, J. Lin, H. Gao and Y. Lin, "Effects of A Scratch-based Experiential Learning Approach on Students' Math Learning Achievements and Interest," 2021 International Symposium on Educational Technology (ISET), Tokai, Nagoya, Japan, 2021, pp. 261-265, <https://doi.org/10.1109/ISET52350.2021.00062>.



Enhancing 7th Grade Students' Awareness of Programming with Arduino in Mathematics

Talip Gültekin^{1*}
Serkan Narlı²

¹Dokuz Eylül University, Master's Student, Department of Mathematics Education, İzmir Türkiye
talipgultekin1122@gmail.com

²Dokuz Eylül University, Department of Mathematics Education, İzmir Türkiye
serkan.narli@deu.edu.tr

*Corresponding Author

Abstract: In today's era, the onset of digitalization has significantly contributed to changes and innovations in educational practices driven by technological advancements. In this context, our study aims to raise awareness by enhancing students' algorithmic thinking while providing knowledge of computer and robotic coding. This mixed-methods study was conducted with 15 seventh-grade students voluntarily selected from a foundation middle school in İzmir. Qualitative data were collected through semi-structured interviews, while quantitative data were gathered using the Technology Readiness Scale and the Digital Literacy Scale. Students participated in a six-week algorithmic programming course. Upon completing the course, they designed a mini CNC plotter machine using Arduino. Data were collected through interview forms and scales administered before and after the course. Content analysis was used for qualitative data analysis, while the Wilcoxon Signed-Rank Test was employed for quantitative data analysis. At the end of this study, students reported that their coding competencies enhanced their algorithmic skills and that geometric shapes could be expressed in different ways.

Keywords: Arduino, Coding, Mathematics Education, Technology, Software

Received:14.01.2025
Accepted: 25.02.2025
Available Online: 30.05.2025

INTRODUCTION

Throughout history, humanity has continuously sought to develop new products to simplify life. These efforts can be considered a form of technology (Agar, 2019). If embers did not burn human hands, there would be no need for tongs; if nails could be pressed with fingers, hammers would be unnecessary; and if our cognitive abilities were capable of solving all mathematical expressions, calculators would not be required (Günay, 2017). Rather than restricting the concept of technology to digital tools that have entered our lives, it would be more accurate to define it broadly as the facilitation of tasks and actions that humanity cannot perform or struggles to accomplish. With the opportunities it provides, technology can be utilized as a tool in the education sector (Akbulut et al., 2020). In a learning process that aims to voluntarily change an individual's behaviors, merely transferring existing knowledge to students does not constitute adequate learning (Tekin & Keser, 2020). Instead of presenting knowledge in a pre-digested form, methods that encourage critical thinking, interpretation, and independent research should be followed (Cameron, 2024). A range of educational materials that make learning more engaging and meaningful can enhance students' understanding of mathematics (Antoino et al., 2024). Learning mathematics equips students with creative thinking and problem-solving skills (Bal Sezerel, 2012). The growing recognition of the importance of mathematics education stems from its direct link to technological advancements and living standards (Işık et al., 2008).

Mathematics holds a significant place for students, both due to its applications in daily life and the responsibilities required by their future professions (İlhan & Çağılhan Gülten, 2013). There are various innovations designed to improve mathematics education and make learning more permanent (Bakırcı & Özmentar, 2023). One of these innovations is computer coding (Lupas et al., 2024). According to Jacobsen et al. (2013), research and practices related to coding and computational thinking in schools challenge students while enhancing their learning when they engage in meaningful tasks, receive continuous feedback from their teachers, and are supported throughout the process.

Individuals need various skills to sustain their lives and adapt to the requirements of the modern era. 21st-century skills are interdisciplinary competencies that emerge from the synthesis of knowledge and proficiency to meet contemporary demands and are related to many aspects of modern life (Atış Akyok & Avşar, 2022). These skills can be categorized into three dimensions: “Learning and Innovation Skills,” “Life and Career Skills,” and “Information, Media, and Technology Skills” (Kurudayıoğlu & Gociaoğlu, 2021). One frequently mentioned skills in recent times, and a sub-theme of information technology literacy, is coding (European Commission, 2014a). Programming refers to the set of written code lines used to solve a problem with the help of a programming language (Şahin & Namlı, 2017). Writing software requires coding skills, meaning programming demands high-level cognitive abilities (Yükseltürk & Altınok, 2015). Developing algorithms plays a crucial role in acquiring these high-level skills. Algorithms can be written using any programming language or expressed verbally. To implement algorithms in digital platforms, the software component of a computer must be utilized. Programming languages such as C, C++, and Java may be too complex and challenging for middle school students (Kalelioğlu & Gülbahar, 2014). As an alternative, block-based coding platforms like Code.org, Scratch, and mBlock make it easier and more enjoyable for students to implement their algorithms (Olsson & Granberg, 2022). These platforms do not require programming knowledge and allow users to create code by simply dragging and dropping blocks, providing a simple and intuitive interface accessible to people of all ages (Turan & Yıldız, 2021). Additionally, they incorporate various visual components that enhance user experience (Cheng et al., 2023). Scratch enables students to become more engaged and active while learning mathematical concepts, fostering their creative thinking and collaborative skills (Bernard & Stiawan, 2020). A study by Mo et al. (2021) found that using the Scratch program to present students with visual, mathematical images encouraged them to explore and understand these concepts. On the other hand, arduino is a simple microcontroller board and an open-source platform that can continuously evolve (Mollo et al., 2016, p. 60). It is a resource used across various categories and topics in education (Tudela & Marin, 2023). Arduino’s significance in education stems from its potential, the variety of available types, the diverse experimental opportunities it provides, and its low cost (Marín et al., 2024). Furthermore, Belmonte et al. (2020) highlighted the benefits of using Arduino tools in education, stating that it allows students to reinforce their understanding of concepts through hands-on experimentation, encourages them to engage in design activities, and enhances their creativity. Arduino can serve as an engaging educational tool in mathematics education. For instance, coding projects involving controlled movements and sensor measurements with Arduino can be used to teach students trigonometry and algebra (Negrete, 2023). Additionally, it helps visualize and manipulate mathematical equations and formulas (Negrete, 2024).

A significant corpus of research has been dedicated to the utilisation of block-based coding in the context of mathematics education. The findings of these studies have been shown to contribute to the field of mathematics education. Rodríguez-Martínez et al. (2020) utilised the Scratch software to examine the development of computational thinking skills and mathematical concepts in students enrolled in mathematics education. The study concluded that Scratch software is beneficial in developing students' computational thinking skills. In a subsequent study, Turan and Yıldız (2021) examined the impact of Scratch software on mathematics education. Their study revealed that while students gained the ability to understand basic mathematical expressions in coding, they also had the opportunity to experience the process of learning with programming in a fun and creative way. A further study by Tewes (2019) examined the effects of coding in secondary school mathematics lessons on students' experience and understanding of mathematical concepts. The study concluded that integrating coding into the secondary school mathematics curriculum can positively influence students' mathematical experiences and understanding of mathematical concepts.

Many studies have been conducted on integrating robotics coding into mathematics education, demonstrating its contributions to mathematics and mathematics instruction. In a study conducted by Meço and Görgülü Arı (2021), the benefits of using Arduino boards in STEM education were examined based on student perspectives. The findings indicated that students found these activities both beneficial and enjoyable, developed a more positive attitude toward science courses, increased their interest in the subject, and showed a tendency to pursue careers in engineering and science fields. Similarly, Baysal et al. (2020) investigated the perspectives of secondary school students who received training in coding and Arduino, focusing on their views regarding coding, Arduino, and STEM education. The results revealed that such training positively influenced students' problem-based and peer-supported learning skills, encouraged them to develop various robotic solutions, and fostered problem-solving efforts. Additionally, these activities enhanced students' creative thinking skills, encouraged them to approach problems from different perspectives, and contributed to their problem-solving processes. In another study conducted with sixth-grade middle school students, Tekin

and Keser (2020) examined the impact of robotics coding activities on students' mathematics achievement. Their findings suggested that mathematics instruction supported by robotics activities was more effective in enhancing mathematical achievement than traditional methods. It was also determined that robotics activities had a more significant impact on students with lower mathematics achievement levels, whereas no significant difference was observed among students with medium to high achievement levels. Furthermore, Marin et al. (2024) conducted a meta-analysis to examine the benefits and opportunities of Arduino applications for middle school students. The findings highlighted that Arduino boards were predominantly used in technology and physics topics and interdisciplinary STEM projects. Similarly, Belmonte et al. (2020) conducted another meta-analysis, reviewing 346 documents to analyze the development of Arduino in scientific literature. Their study concluded that research on Arduino in education is still evolving and remains an ongoing area of study. Additionally, most previous studies focused on physics experiments, computational thinking, and computer-based learning.

A literature review on this topic reveals that applications such as Code.Org and Scratch promote sustained learning in the teaching-learning process and increase student engagement and contribute to the creation of enjoyable learning environments. It has been observed that Arduino applications improve students' algorithmic thinking abilities and positively influence their entrepreneurial skills. Our literature review indicates that studies directly utilizing Arduino boards in mathematics education are relatively limited, with focusing predominantly on science education and STEM-related studies. Considering the role of technology in education and its contribution to the development of 21st-century skills, a study has been designed to raise awareness of mathematics education through software and robotic coding for middle school students. In this regard, an educational process has been structured to develop students' analytical thinking, problem-solving, and digital literacy skills. In light of these findings, the primary aim of this study is to integrate dynamic technology into education, minimize students' mathematical errors, ensure an enjoyable learning experience, and establish a "student-centered" learning environment that fosters awareness. Additionally, the study seeks to enhance students' algorithmic thinking skills while learning computer and robotic coding. Therefore, the research problem is defined as: "What are the awareness levels of 7th-grade students regarding programming with Arduino in mathematics education?" To address this central research question, the following sub-questions have been formulated:

- a) Is there a significant difference in students' perceptions of Arduino and mathematics before and after the implementation?
- b) Is there a significant difference in the technology readiness scale before and after the implementation?
- c) Is there a significant difference in the digital literacy scale before and after the implementation?

METHOD

Research Design

Creswell (2006) asserts that mixed-method studies entail the collection and analysis of quantitative and qualitative data within a single study or multiple studies conducted within the scope of a research programme. In this study, the term 'mixed method' is employed to denote the integration of quantitative and qualitative data.

Study Group/Population-Sample

The study group consisted of 15 seventh-grade students enrolled in a foundation middle school in İzmir during the 2022/2023 academic year. In this study, a non-random sampling technique, specifically the convenience sampling method, was employed. In this method, the researcher selects the most easily accessible and cost-effective individuals until reaching the required sample size for data collection (Büyüköztürk et al., 2010).

Data Collection Tools

Students' opinions were determined through a semi-structured interview. The semi-structured interview technique differs slightly from the structured interview technique (Türnüklü, 2000). In this technique, the

researcher prepares an interview protocol containing pre-planned questions (Türnüklü, 2000). During the development of the form, a literature review is conducted first, followed by expert consultation. The interview form used in this study consisted of seven questions prepared based on expert opinions. The questions are as follows:

1. How proficient are you in using a computer? Briefly explain.
2. To what extent do you have access to a computer?
3. Have you heard the term "algorithm" before? If yes, explain its meaning.
4. What do you know about programming?
5. Do you know any programming languages? If so, which ones?
6. Do you think software can be used in mathematics education?
 - i. Does it facilitate or complicate your mathematics learning? Explain.
 - ii. What benefits do you think it can provide?
7. If you were a software developer, what kind of program would you create to make life easier?

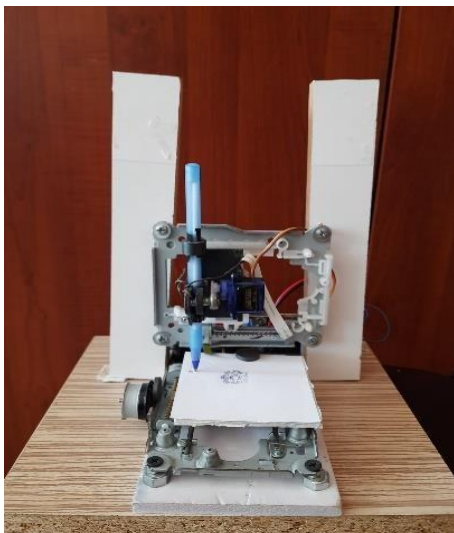
To measure students' willingness to use technology, the "Technology Readiness Index" (TRI), developed by Parasuraman (2000) and adapted into Turkish by Özer, Eriş, and Özmen (2011) with validity and reliability studies, was used. The scale consists of 23 items and four dimensions: optimism (7 items), innovativeness (6 items), insecurity (6 items), and discomfort (4 items). Additionally, to assess students' digital literacy, the "Digital Literacy Scale" (DLS), developed by Ng (2012) and adapted into Turkish by Hamutoğlu et al. (2017), was utilized.

Data Collection Process

The semi-structured interview form, TRS, and DLS were administered to students before the Computer and Arduino Programming Course. During the course, students were informed about what technology means, what has been done, and what can be done with digitalization. Subsequently, the Scratch software platform was introduced to the students, and various games were created with the students using Scratch. Arduino was introduced to the students, and what could be done with Arduino was discussed with the students. Various Arduino circuit simulations were carried out with the students using the TinkerCad software platform. The mBlock application was introduced to the students, and how to create the codes of Arduino circuits with this application and how to run these circuits were shown. Subsequently, the Arduino starter kits distributed to the students were shared so that each kit fell to two students. Various Arduino circuits were built with the students from the mBlock application. At the end of the process, a mini CNC Plotter design was made with the students, and the machine was operated (see Figure 1).

Figure 1

Mini CNC Plotter



Mathematical expressions and various visuals from daily life were drawn by the machine together with the students. At the end of this six-week process, the semi-structured interview form, the Technology Readiness Scale (TRS), and the Digital Literacy Scale (DLS), which were applied at the beginning of the process, were administered to the students again.

In this study, research ethics principles were observed, and the necessary ethical committee approvals were obtained. Within the scope of ethical approval, a document dated 01.11.2022 and numbered E-87347630-659-412496 was obtained from the Dokuz Eylül University Ethics Committee.

Data Analysis

In this study, the qualitative data obtained from the interviews were analyzed using the content analysis, which aligns with qualitative research methods. Content analysis is an analytical technique in which similar data are interpreted comprehensively and presented to the reader in a clear and understandable manner based on predefined codes, categories, and themes (Yıldırım & Şimşek, 2021). In this study, the data subjected to content analysis were summarized and interpreted according to the categories and themes determined about the sub-problems. To ensure the reliability of the data analysis process, the data were independently categorized and coded by two researchers. Subsequently, the researchers convened, compared their independently created categories and codes, and reached a consensus on the non-matching categories and codes. The agreement rate for the finalized categories and codes was determined using the formula proposed by Miles and Huberman (1994) $[\text{Agreement}/(\text{Agreement} + \text{Disagreement}) \times 100]$. The calculations revealed an agreement rate of 81%. According to Miles and Huberman (1994), studies with an agreement rate of 70% or higher are considered reliable. Additionally, the inconsistent sections were re-evaluated and discussed by the researchers in consultation with experts, ultimately reaching a consensus to support the reliability of the study.

The SPSS 24.0 software package was used quantitative data. Pre-test and post-test scores of the students were taken as references, and frequency (f), percentage (%), mean ($\bar{X}$), and standard deviation (SD) values were calculated. Since the research findings did not exhibit a normal distribution and the sample size was fewer than 30 participants, the significance of the difference between pre-test and post-test mean scores was analyzed using the Wilcoxon Signed-Rank Test. The Wilcoxon Signed-Rank Test is used "to test the significance of the difference between two related sets of measurements" (Büyüköztürk, 2020). This technique is preferred over the paired t-test in cases where the sample size is small, and the difference scores do not follow a normal distribution (Büyüköztürk, 2020). Lastly, since students provided responses that could be classified under multiple categories, there were variations in the total frequencies.

The Cronbach's alpha reliability coefficient for the Technology Readiness Scale (THÖ) was 0.864, while the Cronbach's alpha reliability coefficient for the Digital Literacy Scale (DOÖ) was 0.946.

RESULTS

At this stage of this study, the findings obtained from the analysis of participants' responses to the interview questions and scales before and after the six-week course are presented.

Findings about the First Sub-problem

The data obtained regarding the sub-problem of this study, *"Is there a difference in students' views on Arduino and mathematics before and after the implementation?"* are presented below. In Table 1, the responses given by students before and after the implementation of the interview question, *"To what extent do you know how to use a computer? Please briefly explain."*, are presented.

Table 1*Students' Opinions on their Level of Computer Proficiency*

Categories	Pre-application Frequencies	Post-Application Frequencies
I can do software and design.	3	9
I know enough to do my own work.	8	5
Watching movies.	2	1
Playing games.	5	3
Studying lessons.	4	2
Doing robotic coding.	0	5
I know computer hardware.	0	2

Based on the reviews, students who did not take the Computer and Arduino Programming course stated that they knew how to use a computer mainly to do their own work (36.36%) and play games (22.72%). On the other hand, students who took the Arduino Programming course stated that they knew how to use a computer to do software and design (33.33%). It can be said that there is a relatively noticeable difference in the responses of students who took the Computer and Arduino Programming course.

When participants were asked about their access to a computer in the interview form, four students stated that they had their own computers at home, while eight students had shared computers owned by their families. Two students reported that they had no computers at home, while two others could access a computer using an internet café.

In Table 2 below, the responses given by students to the question, "Have you ever heard of the term 'algorithm'? If yes, please explain its meaning" are presented both before and after the course.

Table 2*Students' Opinions on the Term "Algorithm"*

Categories	Pre-application Frequencies	Post-application Frequencies
I've never encountered this word before.	5	0
It is related to daily life.	2	3
I've heard of it but I don't know what it means.	4	0
They are the necessary steps to complete a task.	6	1
It is the planning of a task before it is carried out.	0	7
The arrangement of tasks to be done.	0	6
A solution path for a problem.	0	2

Following the analysis, it was found that before the course, students who did not take the Computer and Arduino Programming course defined the concept of algorithm primarily as the steps required to complete a task (35.29%), and 29.42% of them stated that they did not know the meaning of the term. After the course, however, students defined the concept of algorithm mostly as planning a task before it is performed (41.11%) and sequencing tasks to be performed (35.29%). It can be said that there was a relatively noticeable difference in the students' responses after the course.

Table 3 presents the participants' responses to the interview question "What do you know about programming?" both before and after the course.

Table 3*Students' opinions on Computer and Programming*

Categories	Pre-application Frequencies	Post-application Frequencies
It will progress more in the future.	2	0
It is used for games and daily tasks.	3	0
I know the necessary technical information.	6	7
I know how to create algorithms.	2	0
I can do robotic coding.	0	6
I can program with Scratch.	0	7
I can program with mBlock.	0	3
I can simulate Arduino circuits with TinkerCat.	0	4
Those who have no knowledge.	5	0

Based on the analysis, it was found that before the application, students mostly thought they knew the necessary technical information about computers (e.g., processor and RAM) (33.3%) or that they had insufficient knowledge about computers (27.77%). After the application, students mostly thought they knew the necessary technical information about computers (e.g., RAM and processor) (25.92%), that they could program with Scratch (25.92%), and that they could do robotic coding (22.22%). It can be said that there is a relatively noticeable difference in students' responses after the application.

In Table 4, the responses given by students to the question "Do you know any programming languages, and if so, which ones?" in the pre-application and post-application interviews are presented.

Table 4*Students' Knowledge of Programming Languages*

Categories	Pre-application Frequencies	Post-application Frequencies
I don't know	12	4
I have heard of some programming languages.	3	11
I can code with Scratch.	0	3
I can code with mBlock.	0	5

It was observed that before the implementation, most students did not know the names of any programming languages (75%). However, after the implementation, most students reported knowing the names of some programming languages, such as C, C++ and Java, (47.82%). It can be said that there was a relatively noticeable difference in students' responses after the implementation.

In the interview form administered to participants, the question "Do you think software can be used in mathematics education?" was asked, followed by i) "Do you think it would make your mathematics education easier or harder? Please explain." and ii) "What benefits do you think it could provide?"

In Table 5, the students' responses to the question "Do you think software can be used in mathematics education?" before and after the implementation are examined.

Table 5*Students' Opinions on the Use of Software in Mathematics Education*

Categories	Pre-application Frequencies	Post-application Frequencies
Used	14	15
Not used	1	0

Upon examination, before the implementation, only one student answered that software cannot be used, while the rest of the students (93.33%) answered that it can be used. After the implementation, all students answered that software can be used in mathematics education (100%). No significant difference was observed since only one student's answer changed after the implementation.

In Table 6, the answers given by students who said "software can be used" to the question "i) Do you think software will make your math education easier or harder? Please explain." that were analyzed both before and after the implementation presented.

Table 6*Students' Opinions on Whether Software Makes Mathematics Education Easier or Harder*

Categories	Pre-application Frequencies	Post-application Frequencies
Software can make mathematics more enjoyable.	3	6
Software is related to mathematics.	1	5
It depends on the individual	3	0
It helps visually	0	3
Algorithms are related to mathematics.	0	1
It helps with problem-solving.	0	1
It can only make it easier, but I don't have enough knowledge.	7	3
Individuals who are not knowledgeable about the subject.	1	0

The findings showed that before the implementation, most students (46.66%) believed that software could make learning mathematics easier, but they did not know how it could be done. After the implementation, however, most students thought that software could make mathematics more enjoyable (31.57%) and that software is related to mathematics (26.31%). The findings suggest that there is a relative difference in students' answers after the implementation.

In Table 7, the answers to the question "Do you think software can be used in mathematics education?" given by students who answered "yes" are analyzed regarding the question "ii) What kind of benefits do you think it can provide?". The students' responses before and after the implementation are compared.

Table 7*Students' Opinions on the Benefits of Software in Mathematics Education*

Categories	Pre-application Frequencies	Post-application Frequencies
It can be learned more easily with GeoGebra.	1	3
It makes learning mathematics easier.	5	12
It does not provide any benefits.	3	0
It visualizes mathematical shapes.	0	3
I don't know.	6	1

Table 7 (Continued)

Categories	Pre-application Frequencies	Post-application Frequencies
It makes mathematics more fun.	0	3
It provides a different perspective on mathematics.	0	1

It was observed that before the application, most students lacked knowledge of the topic (40%) and thought that mathematics would be made easier (33.33%). After the application, most students stated that using software in mathematics education would make learning mathematics easier (52.17%). It can be said that there was a relatively noticeable difference in the students' responses after the application.

Table 8 presents the answers given by the participants to the interview question, "If you were a software developer, what kind of program would you create to make life easier?" before and after the application.

Table 8

Students' Views on What they could Do if they were Software Developers

Categories	Pre-application Frequencies	Post-application Frequencies
Those Who Design Programs	9	12
Those Who Don't Know	6	3

Based on the analysis, before the intervention, 60% of students indicated that they would be able to design a program if they were software developers, while 40% stated that they would not be able to do so. After the intervention, 80% of students believed they could design a program, whereas 20% stated they still could not. It can be observed that there is a relatively noticeable difference in the responses of students who took the Computer and Arduino Programming course. Some creative responses were provided by students both before and after the intervention. Examples of these are: "I would design a smart robot that serves tea," "I would implement a chip system with technological devices on doors. By placing a chip in our hand and scanning it at the door, the door would open," and "I would set up a system that allows participation in live classes from home using VR glasses."

Sub-findings related to the second sub-problem

The data obtained for the subproblem of "Is there a significant difference in the Technology Readiness Scale before and after the application?" is presented below. This section presents the descriptive statistics obtained from the TRS, which are interpreted based on students' pre-test and post-test scores and the subdimensions of the scale. According to the responses provided by the students, the mean scores, standard deviations, and levels obtained from the TRS and its subdimensions were analyzed. When calculating level values, those falling within a range below one standard deviation of the mean were categorized as low level, those within a range both below and above one standard deviation were categorized as medium level, and values above one standard deviation were categorized as high level. The findings obtained within this categorization are presented in Table 9.

Table 9

Descriptive Statistical Values of Students' Pre-test and Post-test Results

	Factor	Class	N	Number of Items	$\bar{X}$	Level	SS
Pre-test	Optimism	7	15	7	3.657	Medium	0,776
	Innovativeness	7	15	6	3.400	Medium	0,632
	Insecurity	7	15	6	3.155	Medium	0,834
	Discomfort	7	15	4	3.183	Medium	0,752
	Total	7	15	23	3.376	Medium	0.617

Table 9 (Continued)

	Factor	Class	N	Number of Items	$\bar{X}$	Level	SS
Post-test	Optimism	7	15	7	3,914	Medium	0,520
	Innovativeness	7	15	6	3,733	Medium	0,537
	Insecurity	7	15	6	3,300	Medium	0,652
	Discomfort	7	15	4	3,716	Medium	0,549
	Total	7	15	23	3,675	Medium	0,218

When analyzing the data from Table 9, it was observed that the highest mean score for students' Technology Readiness subdimensions in the pre-test results was for the Optimism subfactor ($\bar{X} = 3.657$), while the lowest mean score was for the Distrust subfactor ($\bar{X} = 3.255$). Similarly, in the post-test results, the highest mean score was for the Optimism subfactor ($\bar{X} = 3.914$), and the lowest mean score was for the Anxiety subfactor ($\bar{X} = 3.300$). Additionally, the overall arithmetic mean obtained from the pre-test data was $\bar{X} = 3.376$, and the standard deviation was $ss = 0.617$. Based on this data, it can be concluded that students' technology readiness levels in the pre-test were at a medium level. For the post-test data, the arithmetic mean was $\bar{X} = 3.675$, and the standard deviation was $s = 0.218$. Based on these results, it can be said that students' overall technology readiness levels were also at a medium level, but considering the averages, the technology readiness levels improved ($\bar{X}_{\text{post-test}} = 3.675 > \bar{X}_{\text{pre-test}} = 3.376$).

Wilcoxon Signed-Rank Test Results for Students' Data from the Technology Readiness Scale (TRS)

In the present study, the Wilcoxon Signed-Rank Test was applied to examine whether there was a significant difference in students' technology readiness levels, based on their responses to the Technology Readiness Scale (TRS), between the pre-test and post-test results. The findings obtained from this test are presented in Table 10.

Table 10

Results of the Wilcoxon Signed-Rank Test for Pre-test and Post-test Technology Readiness Scale Scores

Post-test - Pre-test	N	Rank Average	Rank Sum	Z	p
Negative Rank	7	5	35	-1,422	0,155
Positive Rank	8	10,63	85		
Tied	0	-	-		

According to the test results presented in Table 10, it can be observed that there was no significant difference between the pre-test and post-test responses given by the students to the Technology Readiness Scale ($z=1.422$; $p>0.05$). However, it is noticeable that the average of the responses given in the post-test ($\bar{X}=3.675$) was higher than the average of the responses given in the pre-test ($\bar{X}=3.376$) ($\bar{X}_{\text{post-test}}=3.675 > \bar{X}_{\text{pre-test}}=3.376$).

Findings about the Third Sub-problem

The data obtained for the sub-problem of this study, "Is there a significant difference in the Digital Literacy Scale (DLS) before and after the application?" is presented below. This section includes descriptive statistics of the data obtained from the DLS. These statistics were interpreted based on the students' pre-test and post-test scores as well as the sub-dimensions of the scale. Based on the responses provided by the students in this research, the average scores, standard deviations, and levels derived from the DLS and its sub-dimensions were examined.

When calculating level values, the range corresponding to values below one standard deviation from the mean score was categorized as low level, the range between one standard deviation below and above the mean

score was categorized as medium level, and values above one standard deviation from the mean score were categorized as high level. The findings obtained based on this categorization are shown in Table 11.

Table 11

Descriptive Statistical Values Of Students' Pre-test and Post-test Results.

	Factor	Class	N	Number of Items	$\bar{X}$	Level	SS
Pre-test	Attitude	7	15	7	3,438	Orta	1,166
	Technical	7	15	6	3,522	Orta	0,991
	Cognitive	7	15	2	3,466	Orta	0,801
	Social-Emotional	7	15	2	3,100	Orta	0,929
	Total	7	15	17	3,41	Orta	0,992
Post-test	Attitude	7	15	7	3,609	Orta	0,926
	Technical	7	15	6	3,655	Orta	1,075
	Cognitive	7	15	2	4,000	Orta	0,801
	Social-Emotional	7	15	2	3,400	Orta	1,105
	Total	7	15	17	3,647	Orta	0,825

As shown in Table 11, it was observed that according to the pre-test results of the sub-dimensions of students' digital literacy levels, the highest average belonged to the Technical ($\bar{X}=3.522$) sub-factor, while the lowest average belonged to the Social-Emotional ($\bar{X}=3.100$) sub-factor. Similarly, according to the post-test results of the sub-dimensions of students' digital literacy levels, the highest average belonged to the Cognitive ($\bar{X}=4.000$) sub-factor, while the lowest average belonged to the Social-Emotional ($\bar{X}=3.400$) sub-factor. Additionally, the overall arithmetic mean obtained from the pre-test data was calculated as $\bar{X}=3.431$ and the standard deviation as $ss=0.992$. In light of these data, the findings showed that students' digital literacy levels were moderate according to the pre-test data. The arithmetic mean obtained from the post-test data was calculated as $\bar{X}=3.647$ and the standard deviation as $s=0.825$. According to these results, it can be said that the general digital literacy levels of the students were still at a moderate level, but considering the averages, the digital literacy levels have increased ($\bar{X}_{\text{post-test}}=3.647 > \bar{X}_{\text{pre-test}}=3.431$).

Results of the Wilcoxon Signed-Rank Test for Data Obtained from Students' Digital Literacy Scale (DLS)

In this study, the Wilcoxon Signed-Rank Test was applied to examine whether there was a significant difference in the pre-test and post-test results of the sub-dimensions of students' digital literacy levels based on their responses to the Digital Literacy Scale (DLS). The findings obtained from this test are shown in Table 12.

Table 12

Results of the Wilcoxon Signed-Rank Test for Pre-test and Post-test Scores of the Technology Readiness Scale

Post-test - Pre-test	N	Rank Average	Rank Sum	Z	p
Negative Rank	4	7,13	28,50	-1,792	0,073
Positive Rank	11	8,32	91,50		
Tied	0	-	-		

According to the test results given in Table 12, it was observed that there was no significant difference between the pre-course and post-course responses of the students participating in this study to the Digital Literacy Scale (DLS) ($z=1.792$; $p>0.05$). However, it was seen that the average of the responses given by the students in the post-test ($\bar{X}=3.647$) was higher than the average of the responses given in the pre-test ($\bar{X}=3.431$) ($\bar{X}_{\text{post-test}}=3.647 > \bar{X}_{\text{pre-test}}=3.431$).

DISCUSSION AND CONCLUSION

An analysis of the answers provided by the students in the interview form before application suggests that they predominantly utilise computers in their daily professional activities. However, the findings indicate a deficiency in their understanding of software, programming, robotic coding and Web 2.0 tools (e.g., Geogebra). This observation leads to the hypothesis that the students' awareness of computer science and programming may be limited. A study by Gezgin, Azaz, and Atabey (2022) examined the effects of robotics and coding education on secondary school students' attitudes toward cooperative learning, perceptions of problem-solving skills, and personality traits. The study's findings indicated that students' attitudes toward cooperative learning and their perception of problem-solving skills increased. The findings of this study indicated that robotic coding, and consequently computer programming education, contributed to enhancing students' problem-solving skills.

The students responses in the interview form indicated a lack of familiarity with computers and programming, despite the utilisation of technology in their daily work. However, the study revealed that the students possessed coding abilities in Scratch and mBlock, robotics coding capabilities, and the capacity to utilise Arduino simulation using Tinkercat. This finding suggests that the students had already developed a certain level of understanding of computer science and programming during the application process, which was subsequently confirmed through further studies. A study conducted by Sırakaya (2018) examined the perceptions of secondary school students regarding coding education, focusing specifically on block-based coding. The study revealed that students found coding education enjoyable and considered the training sessions engaging and entertaining. The Scratch programme, which incorporates a mathematical operation block, has been found to be effective in attracting students' attention and facilitating coding (Çubuköz, 2008). In a separate study, Fessakis et al. (2013) sought to enhance problem-solving skills by introducing computer programming to kindergarten children aged 5-6. The study's findings indicated that kindergarten children can develop problem-solving skills, mathematical thinking and social skills through computer programming. Furthermore, it was demonstrated that this process can be effectively implemented under the guidance of a teacher.

Before the application, most students believed that software could be used in mathematics education because they lacked sufficient knowledge about its contributions. However, it was observed that they did not have any information on how it could be utilized. After the application, students stated that coding in mathematics education enhances algorithmic intelligence and facilitates the visual representation of geometric shapes. This indicates that students understood the relationship between software and mathematics during the application process and considered the potential benefits of using software in mathematics education. These findings are consistent with the study conducted by Mercan and Aktaş (2018). They investigated the impact of teaching the topic "Algebraic Expressions" in the 6th-grade Mathematics course using Scratch, as opposed to traditional textbook activities, on students' academic achievement. The study concluded that games and activities designed with Scratch positively contributed to both teachers' and students' academic success, suggesting its utility in teaching challenging and monotonous subjects (Mercan & Aktaş, 2018). Similarly, Totan (2021) examined the effects of block-based coding training on the computational thinking skills and attitudes toward coding of 5th-grade students using the Blockly coding environment. The study found that the Blockly environment in block-based coding training positively influenced students' computational thinking skills, self-efficacy perceptions, and attitudes toward learning coding. Furthermore, Momcilovic (2020) explored the difference in success levels between teaching basic geometric shapes using traditional methods and Scratch. The study revealed a statistically significant difference in success between students who learned the basic concepts of geometry through models and objects and those who learned using Scratch-based programs.

An algorithm is defined as a systematic and step-by-step sequence of operations designed to solve a specific problem (Arabacıoğlu et al., 2007). Utilizing algorithms in solving problems encountered in numerical subjects, such as mathematics will facilitate the solution. The processes that students follow step-by-step to reach a solution to mathematical problems are also an algorithmic approach (Jeyabal et al., 2016). In response to the interview form question measuring students' knowledge of algorithms, it was observed that most students stated that they had heard the term "algorithm" as a name before but did not know what it meant or had never heard it before the application. However, after the application, all students were able to comment on what the term "algorithm" means at a basic level. When the literature on the subject is examined, a study conducted by Durası (2021) investigated the processes of generating algorithms and the structures of the algorithms created

by 6th-grade students on the topic of integers using the Scratch program. The study found that students dynamically performed the actions of recognizing, using, and creating algorithms in three activities. Another study by Lehmann (2024) examined how 12th-grade students' algorithmic tasks were incorporated into algorithmic thinking. At the end of this study, it was observed that students made progress in their algorithm design processes.

It was observed that there was no significant difference between the pre-test and post-test scores of the students participating in the study on the Technology Readiness Scale (TRS). However, it was noted that the average post-test scores were higher than the average pre-test scores. Similarly, it was observed that there was no significant difference between the pre-test and post-test scores of the students on the Digital Literacy Scale (DLS). However, the average post-test scores were higher than the average pre-test scores.

This study found that students who took the Computer and Arduino Programming course learned about computers, computer science, robotic coding, algorithm development, and the relationships between these topics and mathematics. This result is consistent with the findings of Hangün's (2019) study. In this study, the effects of robot programming education on students' mathematics achievement, mathematics anxiety, programming self-efficacy, and STEM attitude variables were examined. The study concluded that robot applications are an effective method in mathematics education and can increase the efficiency of lessons (Hangün, 2019). It was observed that even students with very little knowledge about computers were able to code using Scratch and program Arduino through mBlock by the end of the process. Additionally, it was noted that students understood that mathematics is not limited to a single subject but is related to other fields. The term "algorithm" was adopted by students during the process, and they indicated that the concept of an algorithm is embedded in daily life.

At the end of the application, a prototype was designed with the students to create a mini CNC Plotter using Arduino, allowing students to see what they could achieve with computers and Arduino. This created technological awareness among the students and positively contributed to their perspectives on computers, software, robotic coding, algorithms, and their relationship with mathematics. A study by Shankar et al. (2013) involved 9th-grade students building and programming low-cost robots and drawing various geometric shapes. This study aimed to increase students' interest in engineering and mathematics. Robots were integrated into the mathematics lessons, and it was observed that the use of robots benefited the mathematics classes. Students solved problems using the Pythagorean theorem and then analyzed the results with robot simulations. Errors made by students in both the mathematical problems and the use of robots were discussed. This helped students understand the difference between conceptual errors and real-world errors. Shankar et al.'s (2013) study supports the findings of this research.

In the present study, students stated that integrating software into mathematics lessons and developing algorithms contributed to their learning of mathematics. They also mentioned that dynamic geometry software should be used more frequently to understand better the geometric concepts in the middle school mathematics curriculum.

In conclusion, according to the responses given by students in the interview form, it was revealed that students established a close relationship between learning mathematics and computer programming and software, which influenced their perspectives on mathematics. In this context, it is recommended to increase the number of activities that develop students' algorithmic thinking structures during mathematics lessons in schools, incorporate more block-based applications like Scratch into mathematics activities, and use dynamic geometry software that enhances students' visual, and mathematical literacy.

Note: This study was supported by the TÜBİTAK 2209-A University Students Research Projects Support Program. It is also an extended version of the paper presented at the 16th National Science and Mathematics Education Congress (UFBMEK 2024) symposium.

REFERENCES

- Agar, J. (2019). Teknoloji nedir? Teknoloji: Bir kavramın eleştirel tarihi, Eric Schatzberg, Chicago ve Londra, Chicago Üniversitesi Yayınları, 2018, 352 sayfa, 27,45 \$ (kağıt kapaklı), ISBN: 978-0-226-58383-9. *Annals of Science*, 77 (3), 377–382. <https://doi.org/10.1080/00033790.2019.1672788>
- Akbulut, İ., Akyıldız, A., Yılmaz, S., Bayri, E., & Bayri, G. (2020). Eğitimde Teknoloji Kullanımının Öğrenci Başarısına Etkisi. *International QMX Journal*, 6. <https://doi.org/10.5281/zenodo.10729254>
- Antoino, J., Tudela, P. A., & Terron, P. D. (2024, 10). Computational Thinking and Programming with Arduino in Education: : A Systematic Review for Secondary Education. *Heliyon*.
- Arabacıoğlu, T., Bülbül, H. İ., & Filiz, A. (2007). Bilgisayar programlama öğretiminde yeni bir yaklaşım. IX. akademik bilişim konferansı bildirileri (s.193-197). Dumlupınar Üniversitesi, Kütahya.
- Atış Akyol, N., & Aşkar, N. (2022). Erken Çocukluk Döneminde 21. Yüzyıl Becerileri. *Gazi Üniversitesi Gazi Eğitim Fakültesi Dergisi*, 42(3), 2597-2629. <https://doi.org/10.17152/gefad.1081472>
- Bakırcı, G., & Özmantar, M. F. (2023). Matematik Eğitiminde Etkinlik Temelli Öğretim: Öğretmenlerin Etkinlik Değerlendirmeleri. *Dokuz Eylül Üniversitesi Buca Eğitim Fakültesi Dergisi*(58), 3158-3184. <https://doi.org/10.53444/deubefd.1359320>
- Bal-Sezerel, B. (2012). Seçici problem çözme (SPÇ) tekniğinin ilköğretim 6. ve 7. sınıf öğrencilerine yönelik matematik eğitimindeki sosyal geçerliğinin değerlendirilmesi. [Yüksek lisans tezi, Anadolu Üniversitesi].
- Belmonte, J. L., Marin, J. A. M., Costa, R. S., & Guerrero, A. J. M. (2020). Arduino Advances in Web of Science. A Scientific Mapping of Literary Production. *IEEE Access*, 6.
- Bernard, M., & Setiawan, W. (2020, October). Developing math games media using scratch language. In *Journal of Physics: Conference Series* (Vol. 1657, No. 1, p. 012064). IOP Publishing. <https://doi.org/10.1088/1742-6596/1657/1/012064>
- Büyüköztürk, Ş. (2020). Sosyal Bilimler İçin Veri Analizi El Kitabı . Ankara: Pegem Akademi.
- Büyüköztürk, Ş., Çakmak, K.E., Akgün, Ö.E., Karadeniz, Ş., & Demirel, F. (2010). Bilimsel araştırma yöntemleri. Ankara: Pegem Akademi.
- Cameron, K. (2024). *Progressive Education and the Quality Agenda: Contr duction and the Quality Agenda: Contradictory Views of Children, Childhood, and the Purposes of Education for Young Children* [Yayımlanmamış Doktora Tezi].
- Cheng, Y. P., Lai, C. F., Chen, Y. T., Wang, W. S., Huang, Y. M., & Wu, T. T. (2023). Enhancing student's computational thinking skills with student-generated questions strategy in a game-based learning platform. *Computers & Education*, 6. <https://doi.org/10.1016/j.compedu.2023.104794>
- Creswell, J.W. (2006). Understanding Mixed Methods Research, (Chapter 1). Available At: http://Www.Sagepub.Com/Upm-Data/10981_Chapter_1.Pdf (Erişim Tarihi: 02/03/2022)
- Durası, E. (2021). *Akademik Başarısı Yüksek 6. Sınıf Öğrencilerin Scratch Programı ile Tamsayılar Konusunda Algoritma Üretme Süreçleri ve Yapılarının İncelenmesi* [Yayımlanmamış Yüksek Lisans Tezi].
- Fessakis, G., Gouli, E., & Mavroudi, E. (2013). Problem solving by 5–6 years old kindergarten children in a computer programming environment: A case study. *Computers & Education*. <https://doi.org/10.1016/j.compedu.2012.11.016>
- Gezgin, D. M., Azaz, E., & Atabey, E. (2022). Ortaokul Öğrencilerinin Robotik ve Kodlama Eğitimi Başarılarına İşbirlikli Öğrenme Tutumu, Problem Çözme Becerisi Algısı ve Kişilik Tiplerinin Etkisi: Bir Nedensel Karşılaştırma Araştırması. *Öğretim Teknolojileri ve Öğretmen Eğitimi Dergisi*, 28-42.
- Günay, D. (2017). Teknoloji Nedir? Felsefi Bir Yaklaşım. *Yüksek Öğretim ve Bilim Dergisi*, 164.

- Hamutoğlu, N. B., Güngören, Ö. C., Uyanık, G. K., & Erdoğan, D. G. (2017). Dijital Okuryazarlık Ölçeği: Türkçe'ye Uyarlama Çalışması. *Ege Eğitim Dergisi*, 18(1), 408- 429.
- Hangün, M. E. (2019). *Robot Programlama Eğitiminin Öğrencilerin Matematik Başarısına, Matematik Kaygısına, Programlama Özyeterliliğine ve Stem Tutumuna Etkisi* [Yayımlanmamış Yüksek Lisans Tezi].
- Iskrenovic-Momcilovic, O. (2020). Improving Geometry Teaching with Scratch. *International Electronic Journal of Mathematics Education*, 15(2), em0582. <https://doi.org/10.29333/iejme/7807>
- Işık, A., Çiltaş, A., & Bekdemir, M. (2008). Matematik eğitiminin gerekliliği ve önemi. *Atatürk Üniversitesi Kazım Karabekir Eğitim Fakültesi Dergisi*, (17), 174-184.
- Jacobsen, M, J Lock and S Friesen. 2013. "Strategies for Engagement: Knowledge Building and Intellectual Engagement in Participatory Learning Environments." *Education Canada* 53, no 1: npn.
- K. Jeyabal, M. Vasuki & A. Dinesh Kumar, "From Algorithms to Conceptual Understanding a Review of Instructional Strategies in Math", *Journal of Engineering Scientific Research and Applications*, Volume 2, Issue 1, January - June, Page Number 191-205, 2016.
- Kalelioglu, F., & Gülbahar, Y. (2014). The effects of teaching programming via Scratch on problem solving skills: A discussion from learners' perspective. *Informatics in Education*, 13(1), 33–50.
- Mehmet Kurudayıoğlu, Büşra Gociaoğlu .(0).21. Yüzyıl Becerileri Ve Türkçe Eğitimi .Nobel Yayıncılık
- Lehmann, T.H. How current perspectives on algorithmic thinking can be applied to students' engagement in algorithmatizing tasks. *Math Ed Res J* 36, 609–643 (2024). <https://doi.org/10.1007/s13394-023-00462-0>
- Lupas, L. B., Olchondra, J. A. A., Irinco, J. P. L., Nobis, M. L., & Bayri, G. (2024). Cracking the Math Code: Challenges and Strateg. *ETCOR*, 6.
- Marín, J. A. M., Tudela, P. A. G., & Terron, P. D. (2024). Computational thinking and programming with Arduino in education: A systematic review for secondary education. *Heliyon*, 6. <https://doi.org/10.1016/j.heliyon.2024.e29177>
- Meço, G., & Arı, A. G. (2021). Arduino Destekli STEM Etkinliklerine Yönelik Ortaokul Öğrencilerinin Görüşleri. *The Journal of International Social Research*, 498-504-505.
- Mercan, M., & Aktaş, M. (2018). 6. Sınıf Matematik Dersine Ait Cebirsel İfadeler Konusunun Scratch Destekli Öğretiminin Öğrenci Başarısına Etkisi. *Social Sciences Studies Journal*, 6395-6409.
- Miles, M. B. & Huberman, A.M. (1994). *Qualitative data analysis: an expanded sourcebook*. (2nd Edition). Calif.: SAGE Publications.
- Mollo, L., Bellotti, F., Berta, R., & Gorla, A. D. (2016). *Building Arduino-Based Tangible Serious Games for Elementary Mathematics and Physics*. Springer.
- Negrete, J. C. (2023). Arduino in agriculture and the teaching of mathematics. *Horticulture International Journal*. <https://doi.org/10.15406/hij.2023.07.00280>
- Negrete, J. C. (2024). Mechatronics for Teaching Mathematics. *Acta Scientific AGRICULTURE*.
- Ng, W. (2012). Can We Teach Digital Natives Digital Literacy?. *Computers & Education*, 59(3), 1065-1078.
- Olsson, J., & Granberg, C. (2022). Teacher-student interaction supporting students' creative mathematical reasoning during problem solving using Scratch. *Mathematical Thinking and Learning*. <https://doi.org/10.1080/10986065.2022.2105567>
- Özer, P. S., Eriş, E. D., & Özmen, Ö. N. T. (2011). Teknolojiye Hazırlık Düzeyinin
- Belirlenmesine Yönelik Bir Ölçek Uyarlama Çalışması. *Dokuz Eylül Üniversitesi İşletme Fakültesi Dergisi*, 12(2), 321-333.

- Parasuraman, A. (2000). Technology Readiness Index (Trı) A Multiple-İtem Scale To Measure Readiness To Embrace New Technologies. *Journal Of Service Research*, 2(4), 307-320.
- Rodríguez – Martínez, J. A., Gonzalez Calero, J. A., & Saez Lopez, J. M. (2020). Computational thinking and mathematics using Scratch: An experiment with sixth-grade students. *Interactive Learning Environments*, 6. <https://doi.org/10.1080/10494820.2019.1612448>
- Shankar, R., Ploger, D., Nemeth, A., & Hecht, S. A. (2013). Robotics: Enhancing Pre-College Mathematics Learning with Real-world Examples. *120th ASEE Annual Conference and Exposition* (s. 1-17). Florida: ASEE.
- Sırakaya, M. (2018). Kodlama Eğitime Yönelik Öğrenci Görüşleri. *Ondokuz Mayıs University Journal of Education Faculty*, 37(2), 79-90. <https://doi.org/10.7822/omuefd.394649>
- Şahin, M. C., & Namlı, N. A. (2017). Algoritma Eğitiminin Problem Çözme Becerisi Üzerine Etkisi. *Recep Tayyip Erdoğan Üniversitesi Sosyal Bilimler Dergisi*, 136-140.
- Tekin, Y., & Keser, H. (2020). Matematik Öğretiminde Robotik Etkinlikler Kullanılmasının Başarıya Etkisi. *Akdeniz Eğitim Araştırmaları Dergisi*, 476-477-478-488.
- Tewes, A. (2019). *The Effects of Incorporating on Student Experience and Understanding of Middle School Mathematical Concepts* [Yayımlanmamış Yüksek Lisans Tezi].
- Totan, H. N. (2021). *Blok Tabanlı Kodlama Eğitiminin Ortaokul Öğrencilerinin Bilgi İşlemsel Düşünme Becerileri Ve Kodlama Öğrenimine Yönelik Tutumlarına Etkisi: Blocky Örneği* [Yüksek Lisans Tezi].
- Turan, B., & Yıldız, B. (2021). Scratch ile Kodlama Yaparken Matematik Öğreniyorum. 2. *Uluslararası Bilim, Eğitim, Sanat ve Teknoloji Sempozyumu* (s. 395-400). İzmir: Dokuz Eylül Üniversitesi Buca Eğitim Fakültesi.
- Tudela, P. A. G., & Marin, J. A. M. (2023). Use of Arduino in Primary Education: A Systematic Review. *Education Sciences*, 6. <https://doi.org/10.3390/educsci13020134>
- Türnüklü, A. (2000). Eğitim Bilim Araştırmalarında Etkin Olarak Kullanılabilecek Nitel Bir Araştırma Tekniği: Görüşme. *Kuram Ve Uygulamada Eğitim Yönetimi*, 547.
- Yıldırım, A., & Şimşek, H. (2021). Sosyal bilimlerde nitel araştırma yöntemleri. Ankara: Seçkin Yayıncılık
- Yükseltürk, E., & Altıok, S. (2015). Bilişim Teknolojileri Öğretmen Adaylarının Bilgisayar Programlama Öğretimine Yönelik Görüşleri. *Amasya Üniversitesi Eğitim Fakültesi Dergisi*, 4(1), 50-65. <https://doi.org/10.17539/aej.96735>
- W. Mo, Y. Zhang, Y. Fu, J. Lin, H. Gao and Y. Lin, "Effects of A Scratch-based Experiential Learning Approach on Students' Math Learning Achievements and Interest," 2021 International Symposium on Educational Technology (ISET), Tokai, Nagoya, Japan, 2021, pp. 261-265, <https://doi.org/10.1109/ISET52350.2021.00062>.