

Meslek yüksekokulu bilgisayar programcılığı öğrencilerinin problem çözme becerileri**Rahime ÇELİK GÖRGÜT¹** ¹Kastamonu Üniversitesi, Meslek Yüksekokulu, Bilgisayar Teknolojileri Bölümü, Kastamonu, Türkiye**Araştırma Makalesi/Research Article****DOI: 10.70736/ijoess.604**

Gönderi Tarihi/ Received:

Kabul Tarihi/ Accepted:

Online Yayın Tarihi/ Published:

11.09.2024

22.04.2025

15.06.2025

Öz

Bu çalışmanın amacı, meslek yüksekokulu bilgisayar programcılığı öğrencilerinin matematiksel problemleri çözme becerilerini belirlemek, bilgisayar programlama dillerini kullanma ile matematik problemlerini çözme becerileri ve bunlar arasındaki ilişki hakkındaki görüşlerini ortaya çıkarmaktır. Çalışmanın deseni açıklayıcı durum çalışmasıdır. Çalışmanın katılımcıları, kolay ulaşılabilir örneklem yöntemiyle belirlenen bilgisayar programcılığı programı öğrencilerinden oluşmaktadır. Araştırma verileri, araştırmacı tarafından hazırlanan “problem çözme beceri testi” ve yarı yapılandırılmış görüşme formu aracılığıyla toplanmıştır. Problem çözme beceri testi 243 öğrenciye uygulanırken yarı yapılandırılmış görüşmeler 14 öğrenci ile yürütülmüştür. Verilerin analizi sürecinde ise problem çözme beceri testinde yer alan her bir sorunun çözümüne yönelik cevaplar doğru ve yanlış olarak değerlendirilmiş ve buna yönelik frekans, yüzde değerleri hesaplanmıştır. Yarı yapılandırılmış görüşme formundan elde edilen veriler ise geleneksel (konvansiyonel) içerik analiz yöntemi ile analiz edilmiştir. İçerik analizi sürecinin güvenilirliğini sağlamak için kararlılık, tekrarlanabilirlik ve doğruluk ölçütleri ele alınırken çalışmanın güvenilirliği için ise inandırıcılık, aktarılabilirlik ve tekrarlanabilirlik ölçütleri ele alınmıştır. Verilerinin analizi sonucunda, bilgisayar programcılığı öğrencilerinin matematiksel problem çözme becerilerinin oldukça düşük düzeyde olduğu görülmüştür. Ancak öğrencilerin matematiksel problem çözme becerilerinin bilgisayar programlama konusunda kendilerine fayda sağladığını bununla birlikte bilgisayar programlama becerilerinin de matematiksel problem çözme becerilerine olumlu etki ettiğini düşündükleri görülmüştür. Bu bağlamda öğrencilerin bilgisayar programlama konusunda ihtiyaç duyulan gerekli düşünme becerilerini geliştirebilmeleri için programcılık eğitim müfredatlarında matematiksel problem çözme becerilerinin kazandırılmasına yönelik ders içeriklerine yer verilebilir.

Anahtar Kelimeler: Bilgisayar programlama, matematiksel problem çözme, problem çözme.***Problem solving skills of computer programming students of vocational school*****Abstract**

The aim of this study is to determine the mathematical problem-solving skills of students in the computer programming of vocational schools and to reveal their opinions on the relationship between their skills in using computer programming languages and their mathematical problem-solving skills. The design of the study is an explanatory case study, which is one of the types of case studies. The participants of the study consist of students enrolled in a computer programming program at a university in Türkiye, determined through the convenience sampling method. The research data were collected through a “problem-solving skills test” and a semi-structured interview form prepared by the researcher with expert opinions. While the problem-solving skills test was administered to 243 students, semi-structured interviews were conducted with 14 students. During the data analysis process, the frequency and percentage values of the answers to each question in the problem-solving skills test were calculated. The data obtained from the semi-structured interview form were analyzed using the traditional (conventional) content analysis method. To ensure the reliability of the content analysis process, the criteria of consistency, repeatability, and accuracy were addressed, while the criteria of credibility, transferability, and repeatability were considered for the reliability of the study. As a result of the data analysis, it was observed that the mathematical problem-solving skills of computer programming students were at a very low level. However, it has been determined that the students consider themselves to be at an intermediate level in terms of mathematical problem-solving and using computer programming languages. Additionally, it has been observed that students believe their mathematical problem-solving skills benefit them in computer programming, and conversely, their computer programming skills positively impact their mathematical problem-solving abilities. To enable students to develop the necessary critical thinking skills required for computer programming, course content aimed at imparting mathematical problem-solving skills can be included in programming education curricula.

Keywords: Computer programming, mathematical problem solving, problem solving.**Sorumlu Yazar/ Corresponded Author:** Rahime ÇELİK GÖRGÜT, **E-posta/ e-mail:** rcelik@kastamonu.edu.tr

GİRİŞ

Programlama, bir problemin bir programlama dili kullanılarak çözülmesi için yazılan kod satırlarıdır (Arabacıoğlu ve ark., 2007). Bu, bir problemi anlama, problem için bir çözüm formüle etme, kavramsal bilgiyi yapılandırma ve bir bilgisayarın çözüm üretebileceği ve kullanabileceği şekilde yazma süreçlerini içerir (Helminen & Malmi, 2010). Bu süreçte bir programcıdan beklenen ise problem çözme becerilerini kullanmasıdır (Rahmat ve ark., 2012). Problem çözme becerisi kişiyi çözüme götürecek bilgilerin kazanılması ve bir problemin çözümünü uygulayabilme düzeyidir (Elkin & Karadağlı, 2015). Ancak çoğu programcılık öğrencisi bilgisayar programlama süreçlerini ve problem çözme becerilerini gerçekleştirmede zorluk çekmektedir (Baist & Pamungkas, 2017). Bu zorluklar problem için çözüm üretememe ve temel programların planlanamamasından kaynaklanmaktadır (Mhashi & Alakeel, 2013). Matematiğin bilgisayar programlama için temel bir alan (McCoy & Burton, 1988) olduğu göz önüne alındığında matematik derslerinde öğrencilere kazandırılacak olan problem çözme becerilerinin öğrencilerin bilgisayar programlamada karşılaştıkları zorlukları yenmede yardımcı olacağı açıktır (Mathew ve ark., 2019). Nitekim matematiksel problem çözme yeteneğinin bilgisayar programlamada başarı için bir ön koşul olduğu (Cheah, 2020) ve matematiksel problem çözme yeteneği ile bilgisayar programlama başarısı arasında pozitif bir ilişkinin olduğu da bilinmektedir (Antonakos, 2016). Dolayısıyla bilgisayar programcılığı öğrencilerinin problem çözme becerilerini geliştirme programcılık kalitesinin yükseltilmesi açısından önemlidir (Barlow-Jones & Westhuizen, 2017). Bu bağlamda bu çalışmanın amacı meslek yüksekokulu bilgisayar programcılığı öğrencilerinin matematiksel problemleri çözme becerilerini belirlemek ve bilgisayar programlama dillerini kullanma becerileri ile matematik problemlerini çözme becerileri ve bunlar arasındaki ilişki hakkındaki görüşlerini ortaya çıkarmaktır. Böylece öğrencilerin problem çözmedeki temel boşlukları belirlenebilecek ve bilgisayar programcılığı programı mezunlarından beklenen, gerçek dünya problemlerini analiz edebilme ve çözebilme becerilerine sahip bireylerin yetiştirilmesine katkı sağlanabilecektir.

KAVRAMSAL ÇERÇEVE

Problem çözme ve problem çözme becerisi

Bir problem, çözüm prosedürü belli olmayan fakat kişinin bilgi ve deneyimlerini kullanarak çözebileceği durumlar olarak tanımlanmaktadır (Olkun & Toluk, 2004). Problem çözme ise verilen bir problemi analiz etme, bir çözüm geliştirme ve çözümü sunmayı gerektiren zihinsel bir süreçtir (Muller, 2005). Bu süreç problemi anlama, plan geliştirme, planı uygulama ve geriye bakma aşamaları ile açıklanmaktadır. Yani, ilk olarak neyin gerekli olduğunun tespit

edilmesi, ardından çözüm fikrini elde etmek ve bir plan yapmak için çeşitli öğelerin birbirleri ile nasıl bağlantılı olduğunun görülmesi gerekir. Sonrasında plan uygulanır ve çözüm gözden geçirilerek üzerine tartışılır (Polya, 2004). Bu süreç soyutlama, temsil etme, bütünleştirme, genelleme, eleştirel, yaratıcı ve yansıtıcı düşünme, analiz ve sentezleme ve önceki bilgileri kullanma gibi çeşitli bilişsel becerileri barındırır. Benzer şekilde matematiksel problem çözme becerisi de bir durumu matematiksel olarak yorumlama süreci olarak tanımlanmaktadır ve bu süreç genellikle matematiksel yorumu ifade etme, test etme ve gözden geçirme ve matematik içindeki ve dışındaki çeşitli konulardan matematiksel kavram kümelerini sıralama, bütünleştirme, değiştirme, gözden geçirme veya rafine etme döngülerini içerir (Lesh & Zawojewski, 2007). Bu bireye çok sayıda konuyu kendi başına bütünsel, anlamlı, gerçek ve uygulanabilir bir şekilde analiz etmeyi öğretir (Hariawan ve ark., 2013). Dolayısıyla öğrencilerin problem çözme becerileri, yalnızca matematikte kullanılan beceriler olarak görülmeyip, farklı alanlarda da kullanılan temel bir beceri olarak görülmektedir (Prayekti ve ark., 2020). Bilgisayar programlamanın temel amacının da problem çözen programları uygulamak olduğu göz önüne alındığında, öğrencilerin problem çözme becerilerinin geliştirilmesi ile aslında programlama becerilerinin geliştirilebileceği söylenebilir (Barlow-Jones & Westhuizen, 2017).

Programlama, problem çözme ve matematiksel problem çözme becerisi

Programlama, bir problem için çözüm geliştirme ve bilgisayarın bu çözümü üretmesi için kullanılabilecek şekilde kod yazma eylemidir (Aho, 2012). Bu, eleştirel ve sistematik düşünme, analitik ve nicel akıl yürütme ile çözüm üretmede yaratıcı ve esnek olma gibi çok sayıda beceriyi talep eden karmaşık bir süreçtir (Ambrosio ve ark., 2011). Programlamada bir problem çözme süreci ise problemi anlama, çözüm çerçevesini tasarlama, stratejileri karşılaştırma, algoritma geliştirme, algoritmayı kodlama, çözüm sunma, planlanan çözümü uygulama, çözümü test etme, koddaki hatayı belirleyip düzeltmeden oluşmaktadır (Levitin, 2012). Benzer şekilde, matematiğin kendisi de bir problem çözme ve bunun için teori ve yöntemler geliştirme bilimidir (Davydov, 2008). Temelde matematiksel süreçleri, kavramları, işlemleri ve problemleri ileri düzeyde analiz edebilme ve yaratıcı çözüm bulma becerilerini içermektedir. Dolayısıyla matematiksel problem çözme, programlamada sıkça karşılaşılan mantık temelli görevlerle benzerlik gösterir. Matematiksel problem çözme, bilgisayar programlama için gerekli olan eleştirel düşünme becerilerini geliştirerek (Syahlan & Simamora, 2022) bilgisayar programlamayı öğrenmeyi kolaylaştırabilir (İnce, 2021). Nitekim matematiksel problem çözme, öğrencilere bir problemin alt bileşenlere ayrılması ve bu bileşenlerin sıralı bir şekilde

ele alınması gerektiğini öğretir (Polya, 2004). Bu süreç, programlamadaki algoritma geliştirme süreciyle doğrudan ilişkilidir (Cormen ve ark., 2022). Benzer şekilde matematiksel problemlerde yapılan hataların analizi ve doğru çözüm yollarının belirlenmesi de programlama sürecindeki hata ayıklama (debugging) süreciyle örtüşür. Yine matematiksel problem çözme, bir problemin çözümüne farklı yöntemlerle yaklaşabilme becerisini geliştirir. Programlama süreçlerinde alternatif çözüm yolları geliştirme veya daha verimli kod yazma konularında da bu beceri büyük önem taşımaktadır. İlgi literatürde de programlamanın matematiksel düşünmeyi bir araç olarak kullandığı ve öğrencilerin düşünme süreçlerini somutlaştırmalarına yardımcı olduğunu (Papert, 1980), yeterli matematiksel bilgi ve beceriye sahip olamayan öğrencilerin ise programlamayı da yapamadıkları belirtilmektedir (Gomes & Mendes, 2007). Bu durum, matematiksel problem çözme becerilerinin programlama öğreniminde temel bir rol oynadığını ortaya koymakta; aynı zamanda, bu becerilerin mesleki uygulamalarda karşılaşılan karmaşık görevleri yerine getirebilmek için vazgeçilmez bir yetkinlik olduğunu da göstermektedir. Nitekim, yetkin bir programcıdan yalnızca teknik bilgiye sahip olması değil, aynı zamanda bir probleme uygulanabilecek teknikleri ve kuralları öğrenmesi, anlaması, keşfetmesi, uygulaması ve uyguladığı yöntemleri değerlendirmesi beklenir (Kreitzberg & Swanson, 1974). Ayrıca bir problemi çözmek için programlama kullanılırken, problemin en iyi nasıl çözüleceğine dair bir hipotez geliştirilmesi ve çözümü test etmek için kullanılabilecek bir dizi kural oluşturulması istenir. Ardından veriler ve çözümü gözden geçirmesi, bir çözüm bulunduğunda ise çözümü rafine etmesi ve daha zarif bir hale getirmesi gerekir (Einhorn, 2012). Dolayısıyla bir programcının programlama süreçlerini öğrenmesi ve geliştirmesi için problem çözme becerilerine sahip olması önem arz etmektedir (Antonakos, 2016). Bu bağlamda şimdiki çalışmada da problem çözme becerilerinin ölçülmesi amacıyla, yukarıda belirtilen problem çözme aşamaları esas alınmıştır. Bahsedilen davranışları gösteren öğrencilerin, problem çözme becerilerine sahip olacağı düşünülmektedir.

Araştırmanın amacı ve önemi

Bu çalışmanın amacı meslek yüksekokulu bilgisayar programcılığı programı öğrencilerinin matematiksel problemleri çözme becerilerini belirlemek ve bilgisayar programlama dillerini kullanma becerileri ile matematik problemlerini çözme becerileri ve bunlar arasındaki ilişki hakkındaki görüşlerini ortaya çıkarmaktır. Nitekim programlama öğretimi 21. yüzyıl becerileri olarak adlandırılan yaratıcılık, eleştirel ve analitik düşünme, problem çözme, üretkenlik, süreç odaklı düşünme ve ürün odaklı projeler hazırlama gibi becerilerin kazandırılmasında önemli bir rol oynamaktadır (Einhorn, 2012). Ancak

programlamanın getirebileceği bu beceriler yapısından dolayı karmaşık görülmektedir (Elçiçek & Karal, 2020) ve çoğu öğrenci için programlamayı öğrenmek oldukça güçtür (Prasad ve ark., 2022). Bu duruma neden olarak konunun doğasında var olan zorluklar gösterildiği gibi asıl sebebin öğrencilerin kazanmış olması gereken matematiksel ve mantıksal bilgiyi içeren problem çözme yeterliliklerinin eksikliği olduğu öne sürülmektedir (Cheah, 2020). Öğrenciler problemleri soyut kavramlar üzerinden analiz edememekte ve çözüm aşamasında algoritmik düşünme süreçlerini etkili bir şekilde kullanamamaktadırlar. Öğrencilerin programlamayı nasıl oluşturacaklarını bilmemeleri aslında onların problemlerin nasıl çözüleceğini bilmiyor olmalarından kaynaklıdır (Pacheco ve ark., 2008a). Bir programlama problemini çözerken sıklıkla karşılaşılan bir diğer engel de metinsel bir problemi matematik diline dönüştürememektir (Gomes ve ark., 2006). Öğrenciler, matematiksel düşünme becerilerini yeterince geliştiremediklerinden, programlama süreçlerinde de güçlüklerle karşılaşmaktadırlar. Bu durum iyi bir matematik altyapısının ve problem çözme becerisinin programlama öğrenimi için gerekli ve önemli olduğunu göstermektedir (Prasad ve ark., 2022). Dolayısıyla öğrencilerin programlama performanslarının artmasında etkili olan matematiksel problem çözme becerilerinin belirlenmesi matematik ve programlama öğretiminde ele alınması gereken temel konulardan biri olarak görülmektedir. Özellikle, öğrencilerin matematiksel problem çözme becerilerinin belirlenmesi ve bu becerilerin programlama başarısına etkisinin ortaya konulması, hem akademik hem de mesleki yeterliliklerinin geliştirilmesi açısından büyük önem taşımaktadır. Bu sayede öğrencilerin problem çözmedeki temel boşlukları belirlenebilecek ve bir bilişim programı mezunundan beklenen becerilere sahip bireylerin yetiştirilmesine katkı sağlanabilecektir. Ek olarak, bilgisayar programcılığı okuyan ön lisans öğrencilerinin akademik başarılarını artırmaları ve öğrenme süreçlerini daha etkili hale getirmeleri için matematik ve programlama arasındaki simbiyotik ilişkiyi anlamaları da kritik önem taşımaktadır. Çünkü matematik ve programlama birbirini tamamlayan disiplinlerdir; matematik, problem çözme ve algoritma tasarımı için teorik bir altyapı sunarken, programlama bu teoriyi pratikte uygulanabilir hale getirir. Dolayısıyla bu ilişkiyi anlamak, öğrencilerin soyut kavramları somut bir şekilde görmelerine, karmaşık problemlere daha analitik yaklaşımlar geliştirmelerine ve algoritmik düşünme yeteneklerini artırmalarına olanak tanır. Bu ilişki kavranılmazsa özellikle mesleki hayatta yazılım geliştirme ve analiz gibi işlerde karşılarına çıkan karmaşık sorunlarla baş etmekte yetersiz kalabilirler. Bu durum sadece akademik başarı değil, aynı zamanda analitik düşünme, yaratıcılık ve profesyonel yeterlilik için de vazgeçilmezdir. Ayrıca, eğitimciler ve müfredat geliştiricileri açısından da bu tür bir araştırma, programlama öğretimini daha etkili

hale getirmek ve öğrencilerin karşılaştıkları zorluklara yönelik çözümler üretmek için önemli veriler sağlayacaktır.

YÖNTEM

Araştırma deseni

Bu çalışma nitel bir araştırma modelindedir. Çalışmanın deseni durum çalışması türlerinden açıklayıcı durum çalışmasıdır. Durum çalışması, bir olgunun bir veya birkaç örneğinin detaylıca incelendiği bir araştırma yaklaşımı iken (Yin, 2014), açıklayıcı durum çalışması, belirli bir bağlamla ilgili ayrıntıları aydınlatmak, aşına olunmayan anlaşılır kılmak ve pratik, gerçek dünya senaryolarıyla ilişkileri açıklamak için kullanılır (Yin, 2009). Şimdiki çalışmanın amacı göz önüne alındığında açıklayıcı durum çalışması yönteminin tercih edilmesi, bu yöntemin öğrencilerin matematiksel problem çözme ile problem çözme ve programlama becerileri arasındaki ilişkiye ilişkin [görüşlerini](#) ayrıntılı bir şekilde ele alınabilmesine olanak tanınmasından kaynaklanmaktadır. Çalışmada incelenen durum, meslek yüksekokulu bilgisayar programcılığı öğrencilerinin problem çözme ve bilgisayar programlamaya ilişkin görüşleri ile matematiksel problem çözme becerileridir.

Çalışma grubu

Araştırmanın katılımcıları, ilk olarak kolay ulaşılabilir örnekleme yöntemi (Creswell, 2012) ile 2022–2023 eğitim-öğretim yılında Türkiye’de bir devlet üniversitesine bağlı üç farklı meslek yüksekokulunda öğrenim görmekte olan bilgisayar programcılığı programı öğrencileri arasından belirlenmiştir. Kolay ulaşılabilir örnekleme yöntemi, araştırmacının elindeki zaman, maliyet ve erişim kısıtları nedeniyle, evrenden kolayca ulaşılabilen bireylerin seçilmesiyle yapılan örnekleme türüdür (Creswell, 2012). Katılımcı seçiminde ise ölçüt örnekleme yöntemi kullanılmıştır. Ölçüt örnekleme, araştırmacının çalışmasında kritik öneme sahip belirli kriterleri tanımlaması ve bu kriterleri karşılayan örneklerin sistematik olarak seçilmesi esasına dayanır (Patton, 2002). Bu bağlamda, genel matematik, programlama temelleri, veritabanı yönetimi, web tasarımı ve nesne tabanlı programlama gibi bilgisayar programcılığı alanındaki temel dersleri almış ve bu derslerden başarılı olmuş öğrenciler çalışma grubuna dahil edilmiştir. Bunun sebebi söz konusu derslerin, yazılım geliştirme süreçlerinin anlaşılması ve modern programlama dillerinin temel yapı taşlarının kavranması açısından önemli bir temel oluşturmasıdır. Ölçütlere uygun öğrenciler arasından çalışmaya dâhil edilecek katılımcıların seçimi amacıyla amaçlı örnekleme yaklaşımlarından biri olan kolay ulaşılabilir örnekleme yöntemi kullanılmıştır (Creswell, 2012). Uygulama sürecinde, araştırmanın problem çözme becerisi testi toplam 243 öğrenciye uygulanmıştır. Bu öğrenciler arasından yarı yapılandırılmış

görüşmelere katılacak katılımcıların belirlenmesinde gönüllülük esasına dayalı amaçlı örnekleme yöntemi kullanılmıştır. Bu yöntem, katılımcıların kendi istekleriyle ve bilinçli olarak araştırmaya katılmayı tercih ettiği bir örnekleme yöntemidir (Patton, 2002). Bu süreçte öncelikle, test uygulamasına katılan tüm öğrencilere çalışmanın nitel aşamasında yer almak üzere görüşmelere katılım çağrısı yapılmıştır. Gönüllü olan öğrenciler arasından veri çeşitliliğini ve temsil gücünü artırmak amacıyla öğrencilerin akademik başarı düzeyleri, problem çözme başarı testi puanları ve öğrenim gördükleri meslek yüksekokulları göz önünde bulundurulmuştur. Bu ölçütlere göre seçilen ve çalışmaya aktif katılım sağlayabileceği öngörülen 14 öğrenciyle yarı yapılandırılmış bireysel görüşmeler gerçekleştirilmiştir.

Veri toplama araçları

Araştırma verileri, araştırmacı tarafından hazırlanan “problem çözme beceri testi” ve yarı yapılandırılmış görüşme formu aracılığıyla toplanmıştır. Katılımcılara ilk olarak “problem çözme beceri testi” uygulanmıştır, arkasından yarı yapılandırılmış görüşmeler yürütülmüştür.

Yarı yapılandırılmış görüşme formu

Araştırmacı tarafından bilgisayar programcılığı programı öğrencilerinin bilgisayar programlama dillerini kullanma becerileri ile matematik problemlerini çözme becerileri ve bunlar arasındaki ilişki hakkındaki fikirlerini belirlemek üzere sekiz maddeden oluşan yarı-yapılandırılmış bir görüşme formu tasarlanmıştır. Tasarlanan form bir matematik eğitimi bir de bilgisayar mühendisliği alanında doktora derecesine sahip iki uzmanın görüşüne sunulmuştur. Uzmanlar formda yer alan maddelerin öğrencilerin matematik problemlerini çözme ve programlamaya ilişkin fikirlerini belirlemede yeterli düzeyde bilgi sağlayabileceğini bildirmiştir. Görüşme formunda yer alan sorulardan ikisi aşağıda yer almaktadır:

1. *Matematik problemlerini çözme becerileriniz ile bilgisayar programlama dillerini kullanma becerileriniz arasında nasıl bir ilişki görüyorsunuz?*
2. *Matematik derslerinde öğrendiğiniz problem çözme stratejilerinin bilgisayar programlama alanında problemleri çözmede size yardımcı olup olamayacağı konusunda ne düşünürsünüz?*

Problem çözme beceri testi

Bu test bilgisayar programcılığı programı öğrencilerinin matematiksel problemleri çözme becerilerini belirlemek üzere araştırmacı tarafından tasarlanmıştır. İlgili test matematiksel problem çözme süreçlerini gerektiren 11 tane günlük hayat probleminden oluşmaktadır ve meslek yüksekokulu bilgisayar programcılığı programındaki temel matematik ders içeriğine

uygun olarak tasarlanmıştır. Genel olarak temel matematik derslerinde öğrencilere aktarılması gereken konular arasında; sayılar ve problemler, temel istatistik kavramları, çokgenler, fonksiyonlar ve grafikleri gibi konular yer almaktadır. Ancak çalışmanın doğası gereği problem çözme beceri testinde problemler konusuna odaklanılmış olup, sayı problemleri, kar-zarar problemleri, yaş problemleri, hız problemleri, işçi problemleri, yüzde ve karışım problemleri ile aritmetik ortalama ve çokgenlerin alanlarının hesaplanmasını gerektiren problemlere yer verilmiştir. Tasarlanan test ilk olarak bir üniversitesinin bilgisayar programcılığı programında görev yapmakta olan biri matematik eğitimi konusunda diğeri de bilgisayar mühendisliği alanında doktora derecesine sahip iki uzmanın görüşüne sunulmuş, uzmanlardan alınan dönütler doğrultusunda testte gerekli düzenlemeler yapılmıştır. Örneğin 2. soru için uzmanlar, öncelikle sorunun bağlamının gerçek yaşama uygunluğunu ve öğrencilerin ilgisini çekici nitelikte olduğunu belirtmiştir; ancak soruda Seda'nın kaç testten not aldığının netleştirilmesi gerektiği önerisinde bulunmuşlardır. Bunun üzerine soru metnine “Böylece yine 5 puandan oluşan bir ortalama olacaktır.” ifadesi eklenmiştir. Bu düzenlemeler sonrasında sorunun, öğrencilerin yalnızca işlem becerilerini değil, problem çözme stratejisi geliştirme ve tahmin yürütme gibi düşünme becerilerini değerlendirebilecek nitelikte olduğu uzman tarafından teyit edilmiştir. Ardından testin, öğrenciler tarafından anlaşılabilirliğini görebilmek amacıyla ise bir devlet üniversitenin bilgisayar programcılığı programında öğrenim görmekte olan beş gönüllü öğrenciye test uygulanmıştır. Uygulama sürecinde herhangi bir olumsuz durumla karşılaşmadığından testte bir değişikliğe gidilmemiştir. Tablo 1’de problem çözme beceri testinde yer alan örnek sorulara yer verilmektedir.

Tablo 1. Problem çözme beceri testi örnek soruları

Problem Çözme Beceri Testi						
Soru No	Soru Metni					Sorunun Amacı
Soru 9	Aşağıdaki tabloda Bursa ilindeki sabah 07:00 ile 08:00 saatleri arası trafiğin akış hızı gösterilmektedir.					Hız problemlerini çözebilme
	Saat	07:00-07:15	07:15-07:30	07:30-07:45	07:45-08:00	
	Hız (km/s)	12	10	8	6	
	Evi ile işyeri arası 8 km olan Mehmet’in saat 08:00’ da başlayacak işine geç kalmaması için en geç saat kaçta evinden işine doğru hareket etmiş olması gerekir?					
Soru 10	Bir kuru temizleme firmasında çalışan Elis ve Ecem’in gelen bir kıyafeti temizleme ve ütöleme sürelerine ilişkin tablo aşağıda verilmiştir.					İşçi problemlerini çözebilme
		Ütüleme		Temizleme		
	Elis	4 dk		12 dk		
	Ecem	6 dk		6 dk		
	Gün içinde gelen 20 kıyafet bu kişiler tarafından en erken kaç dakikada temizlenip ütülenebilir?					

Problem Çözme Beceri Testi		
Soru No	Soru Metni	Sorunun Amacı
Soru 11	Bir babanın oğluna hediye ettiği saat için aralarına geçen diyalog aşağıda verilmiştir:	Yaş problemlerini çözebilme
	Baba: Oğlum ata yadigârı bu saat babamdan bana miras kaldı. Ben bu saati babamın kullandığından senin yaştan kadar daha fazla kullandım.	
	Oğul: Eğer ben bu saati senin kullandığından yaşlarımızın farklı kadar daha fazla kullanabilirsem saat 90 yıllık olacak.	
	Bu baba ile oğlun yaşları toplamı 63 olduğuna göre dede bu saati kaç yıl kullanmıştır?	

Veri toplama süreci

Araştırmada veri toplama süreci, açıklayıcı durum çalışması desenine uygun biçimde, problem çözme beceri testi uygulaması ve yarı yapılandırılmış bireysel görüşmeler olmak üzere ardışık iki aşamada yürütülmüştür. Sürecin ilk aşamasında, problem çözme becerisi testi, 2022–2023 eğitim-öğretim yılı bahar döneminde gerçekleştirilmiştir. Uygulamalar, öğrencilerin ders saatleri dışında, öğrencilerin öğrenim gördükleri okulların dersliklerinde bireysel olarak, araştırmacı gözetiminde yüz yüze yürütülmüştür. Problem çözme başarı testi için öğrencilere yaklaşık 50 dk tanınmıştır. İkinci aşamada ise yarı yapılandırılmış bireysel görüşmeler gerçekleştirilmiştir. Görüşmeler, her öğrencinin öğrenim gördüğü yüksekokulda uygun görülen bir öğretim elemanı ofisinde, sessiz ve dikkat dağınık unsurlardan arındırılmış şekilde gerçekleştirilmiştir. Her bir görüşme, araştırmacı tarafından bire bir yürütülmüş ve yaklaşık 25–35 dakika sürmüştür. Katılımcıların onayı doğrultusunda görüşmeler ses kayıt cihazı ile kaydedilmiş, daha sonra yazılı metne dönüştürülerek analiz sürecine dahil edilmiştir. Veri toplama süreci boyunca etik ilkelere özen gösterilmiş, katılımcılara araştırmanın amacı ve kapsamı hakkında bilgilendirme yapılmış, gizlilik ve gönüllülük ilkeleri doğrultusunda yazılı aydınlatılmış onam formları alınmıştır.

Verilerin analizi

Çalışmanın verilerinin toplanabilmesi amacıyla katılımcılara ilk olarak “problem çözme beceri testi” uygulanmıştır. Daha sonra ise bu öğrenciler arasından seçilen 14 öğrenci ile yarı yapılandırılmış görüşmeler yürütülmüştür. Toplanan verilerin analizi sürecinde problem çözme beceri testinde yer alan sorulara verilen cevaplar doğru, eksik-yanlış ve boş olmak üzere sınıflandırılarak frekans ve yüzde değerleri hesaplanmıştır. Bu süreçte doğru olarak değerlendirilen cevaplar; problemin türüne uygun matematiksel yöntem veya formülün seçildiği, çözüm sürecinin adım adım ve açık bir şekilde yazıldığı, her adımın önceki adımla uyumlu olduğu ve çözümün matematiksel mantığa uygun ilerlediği, yapılan işlemlerin matematiksel olarak doğru ve sıralı bir şekilde gerçekleştirildiği, çözüme uygun bir grafik, tablo

ya da görselin çizildiği cevaplar olmuştur. Yarı yapılandırılmış görüşme formundan elde edilen veriler ise geleneksel (konvansiyonel) içerik analiz yöntemi ile analiz edilmiştir. Bunun için ilk olarak görüşme kayıtları transkript edilerek dikkatlice okunmuş ve verinin genel yapısı hakkında bilgi edinilmiştir. Ardından elde edilen veriler, anlamsal olarak benzerliklerine göre birleştirilip, bu görüşleri temsil edebilecek kodlar oluşturulmuştur. Tümevarımsal bir süreç yürütülerek kodlar arasındaki ilişkiler incelenip bağlantılar kurulmuş, ortak özelliklere sahip kodlar bir araya getirilerek temalar oluşturulmuştur (Yıldırım & Şimşek, 2000). Bulguların sunumunda ise farklı veri toplama araçlarından elde edilen veriler birbirleri ile ilişkilendirilerek sunulmuştur.

Güvenirlilik

Çalışmanın güvenirliliği için; inandırıcılık, aktarılabilirlik ve tekrarlanabilirlik ölçütleri ele alınmıştır (Guba & Lincoln, 1982). Araştırmanın inandırıcılığı için çoklu veri toplama araçlarından yararlanılmış, bu araçlarının hazırlanması ve verilerin analiz edilmesi aşamalarında uzman görüşleri alınmış ve katılımcı kontrolüne (Creswell, 1998) başvurulmuştur. Aktarılabilirliği sağlayabilmek için ise veri toplama sürecine uygun katılım sağlanmaya çalışılmış, araştırmanın yapıldığı ortam ve katılımcılar hakkında betimlemeler yapılmıştır. Analiz sürecinde tüm işlem basamakları detaylıca ele alınmış ve doğrudan alıntılara yer verilerek bulgular tanımlanmıştır. Tekrarlanabilirlik konusunda, yapılan tüm görüşmeler ses kaydına alınarak elde edilen veriler eksiksiz bir şekilde yazıya aktarılmış ve veriler birer hafta arayla bir nitel veri analizi programı yardımıyla iki defa kodlanmıştır. Ek olarak araştırmada var olan sonuçlar ile yapılan yorumlar araştırmacı yanlılığı olmadan sunulmuştur.

Yürütülen içerik analizi sürecinin güvenilirliği sağlamak için ise kararlılık, tekrarlanabilirlik ve doğruluk ölçütleri ele alınmıştır (Güler ve ark., 2015). Bu süreçte ilk olarak araştırmacı yazılı verilerin güvenirliliğinin sağlanmasında “üye kontrolü”nden (Creswell, 1998) yararlanmıştır. Ardından araştırmacı yazıya aktarılan verileri iki hafta ara ile üçer defa kodlanmıştır. Kodlama sonuçlarını birbiriyle karşılaştırarak kodlamalar arasında tutarlılık sağlamıştır. Daha sonra matematik eğitimi üzerine doktora derecesine sahip bir uzmandan görüşme verilerini kodlaması istenmiş ve uzmanın yaptığı kod ve kategoriler ile araştırmacının yaptıkları kod ve kategoriler karşılaştırılarak bunlar arasındaki tutarlılığa bakılmıştır. İki kodlayıcı arasındaki puanlayıcı güvenirliliğini sağlamak ve anlaşma düzeyini ölçmek için ise Cohen’s Kappa istatistiği kullanılmış (Cohen, 1960) olup, Kappa değeri 0.76 olarak hesaplanmıştır. Bu sonuç araştırmacılar arasında önemli derecede uyum olduğunu göstermektedir (Landis & Koch, 1977). Daha sonra uzman ile görüşmeler yapılarak kod ve

kategorilere son şekli verilmiştir. Son olarak, katılımcıların ifadelerine metin içinde yer verilmiş, bulguların sunumunda derinlemesine bir betimleme yapılmıştır (Creswell, 2012).

BULGULAR

Problem çözme beceri testine ilişkin bulgular

Problem çözme beceri testine ilişkin öğrencilerin testte yer alan her bir probleme vermiş olduğu yanıtlar doğru, eksik-yanlış ve boş olmak üzere sınıflandırılmıştır. Öğrencilerin vermiş oldukları yanıtlara dair frekans dağılımları Tablo 2’ de yer almaktadır.

Tablo 2. Problem çözme beceri testine ilişkin frekans dağılımları

Sorular	Doğru Yanıt <i>f</i>		Yanlış- Eksik Yanıt <i>f</i>		Boş <i>f</i>	
	<i>f</i>	%	<i>f</i>	%	<i>f</i>	%
1. Soru	124	51,02	76	31,27	43	17,69
2. Soru	141	58,02	74	30,45	28	11,52
3. Soru	104	42,79	84	34,56	55	22,63
4. Soru	106	43,62	68	27,98	69	28,39
5. Soru	56	23,04	84	34,56	103	42,38
6. Soru	100	41,15	87	35,80	56	23,04
7. Soru	59	24,27	88	36,21	96	39,50
8. Soru	81	33,33	76	31,27	86	35,39
9. Soru	72	29,62	82	33,74	89	36,62
10. Soru	32	13,16	113	46,50	98	40,32
11. Soru	59	24,27	126	51,85	58	23,86

Tablo 2 incelendiğinde çalışmaya katılan öğrencilerin testte yer alan soruları önemli ölçüde boş bıraktıkları (5, 7, 8 ve 9. soru) veya yanlış yanıtladıkları (10 ve 11. soru) görülmektedir. Benzer şekilde çalışmaya katılan öğrencilerin testte yer alan sorulardan en çok 2. soruyu doğru yanıtlarken en az ise 10. soruyu doğru yanıtladıkları görülmektedir. Soruların genelinin doğru yanıtlanma oranına $\left(\frac{\text{Doğru yanıt yüzdesi}}{\text{Soru sayısı}}\right)$ bakıldığında ise soruların yalnızca %34,94 oranında doğru yanıtladığı görülmüştür.

Öğrencilerin problem çözme beceri testinde yer alan sorulara vermiş oldukları yanıtlar incelendiğinde çoğunlukla aritmetik ortalamaya ilişkin hesapların yapılmasını gerektiren problemleri doğru yanıtlayabildikleri (2. ve 3. soru) görülmüştür. Tablo 3’te öğrencilerin bu konudaki sorulara vermiş oldukları doğru yanıtlardan örnekler yer almaktadır.

Tablo 3. Problem çözme beceri testindeki 2 ve 3. sorulara verilen doğru yanıt örnekleri

Soru No	Doğru yanıt örnekleri	
Soru 2	Soru2. Seda matematik testlerinden 88, 90, 78, 77, 85, 80. notlarını almıştır. Öğretmen sınıfı geçmek için bu testlerin ortalamasını kullanacaktır. Herhangi bir öğrenci mevcut test ortalamasını koruma veya en yüksek puanı ile en düşük puanını final sınavından alacağı puanla değiştirme seçeneğine sahiptir. Böylece yine beş puandan oluşan bir ortalama olacaktır. Eğer Seda final sınavı olmayı tercih ederse bu sınavdan en az kaç puan almalıdır ki şuan ki ortalamasından daha iyi bir ortalamaya sahip olsun?	(Ö16)
Soru 3	Soru3. Tarık, oynadığı son 6 basketbol maçı için maç başına ortalama 15 sayı atmıştır. Oysaki Tarık maç başına ortalama 18 sayı ile oynamak istemektedir. Tarık'ın hedefine ulaşabilmesi için her bir maçta 20 sayı attığı kaç tane daha maç oynaması gerekir?	(Ö127)

Seda matematik testlerinden 88, 90, 78, 77, 85 ve 80 puan almıştır. Öğretmen, sınıfı geçmek için bu testlerin ortalamasını kullanacaktır. Öğrenciler, isterlerse final sınavına girip düşük notlarından birini final notlarıyla değiştirebilir ya da hiç final sınavına girmeyip mevcut ortalamalarını koruyabilirler. Buna göre Seda'nın daha yüksek bir ortalama elde edebilmesi için final sınavından en az kaç puan alması gerekir?

Öğrencilerin yarısına yakını rasyonel sayılara ilişkin cebirsel hesaplamaların yapılabilmesi (1. soru) ile kar-zarar problemlerine ilişkin hesaplamaların yapılabilmesini gerektiren (4. ve 6. soru) problemlerde başarı gösterebilmiştir. Tablo 4'te öğrencilerin bu konulardaki problemlere vermiş oldukları doğru yanıtlardan örnekler yer almaktadır.

Tablo 4. Problem çözme beceri testindeki 1., 4. ve 6. sorulara verilen doğru yanıt örnekleri

Soru No	Doğru yanıt örnekleri
Soru 1	Soru1. 30 cm yüksekliğinde bir dolabın 4 cm kalınlığında bir alt taban ve 1 cm kalınlığında da bir üst tabanı olmalıdır. Geriye kalan alana 4 eşit boyda çekmece yerleştirilecektir ve her bir çekmece arasında 1/4 cm boşluk olmalıdır. Buna göre her bir çekmecenin yüksekliği kaç cm olmalıdır?

Soru10. Bir kuru temizleme firmasında çalışan Elis ve Ecem'in gelen bir kıyafeti temizleme ve ütöleme sürelerine ilişkin tablo aşağıda verilmiştir.

	Ütöleme	Temizleme
Elis	24 dk	12 dk
Ecem	6 dk	6 dk

Gün içinde gelen 20 kıyafet bu kişiler tarafından en erken kaç dakikada temizlenip ütölenebilir?

Soru 10

Handwritten solution for Soru 10:

Temizleme

$$\begin{array}{r} 12 \text{ dk} \\ \times 20 \\ \hline 240 \end{array}$$

Ütöleme

$$\begin{array}{r} 24 \text{ dk} \\ \times 20 \\ \hline 480 \end{array}$$

Toplam

$$240 + 480 = 720 \text{ dk}$$

720 dk / 6 dk = 120 adet

(Ö26)

Öğrencilerin matematik problemlerini çözme ve bilgisayar programlama dillerini kullanma konusundaki fikirlerine ilişkin bulgular

Meslek yüksekokulu bilgisayar programcılığı öğrencilerinin matematiksel problemleri çözme becerileri ile problem çözme ve bilgisayar programlama dillerini kullanmaya ilişkin görüşlerinden elde edilen veriler içerik analizi yöntemi ile analiz edilmiştir. Verilerin analizinden elde edilen tema ve alt tema ile bu temalara ait kodlar Tablo 7'de özetlenmiştir.

Tablo 7. Öğrencilerinin matematiksel problemleri çözme ile bilgisayar programlama dillerini kullanmaya ilişkin görüşleri

Tema	Açıklama	Kod
Matematiksel başarı	Matematiğe karşı ilgi duyma ve matematiksel problemleri çözmede kendini başarılı görme.	Matematiği bir ihtiyaç olarak görme. Matematiğe ilgi duyma. Matematiği sevmek. Matematikte kendini başarılı görme.
Programlama başarı	Programlamaya karşı ilgi duyma ve programlamada kendini başarılı görme.	Programlamadan keyif alma. Programlamada kendini başarılı görme. Programlamada kendini yetenekli görme. Program için çözüm üretebilme. Program için yönlendirme yapabilme. Programcılıkta kendini geliştirebilme.
Matematik ve programlama arasında ilişki	Matematikteki problem çözme ve algoritma mantığının programlamanın temeli olduğunu, programlama için matematiğin ön koşul olduğunu ve ikisinin de benzer iş akışları gerektirdiğini görme.	Matematik programlama için ön koşuldur. Matematiksel problemler bilgisayarların altyapısını oluşturur. Bilgisayar ortamının temeli matematiktir. Programlama kodları matematiksel kodlardır. Programlama matematiksel mantık üzerine kuruludur. Matematik ve programlama birbirini tamamlar. Program diline dönüştürmek için önce matematiksel çözüm yapılmalı. Programlama temelindeki kavramlar matematik kaynaklı kavramlardır. Matematik ve programlama mantığı aynıdır. Matematik problemleri ile programlamanın benzer adımları vardır. Matematik ve programlama sorun çözme üzerine odaklıdır. Matematik ve programlamada alternatif yollar üretilir. Matematik ve programlamada benzer sayı sistemleri vardır. Matematik ve programlamanın algoritmaları aynıdır.

Tema	Açıklama	Kod
		Matematik ve programlamada benzer iş akışları yürütülür. Matematik problemleri ile programlamanın benzer çözüm yolları vardır.
Matematiğin programlamaya etkisi	Matematiksel problemleri çözme becerisinin programlama becerisini geliştirebilmesi ve matematikteki başarısızlığın programlamada da başarısızlığa yol açabilmesi.	Matematikte zorlanınca programlamada da zorlanma. Matematiği anlayınca programlamayı da anlama. Matematik algıyı genişlettiği için programlama becerilerini geliştirir. Matematik teknik düşünmeyi sağladığı için programlama becerisini geliştirir. Matematiksel problemleri çözmek programlama becerisini geliştirir. Matematik bilmeden program yazılamaz. Matematik becerilerini geliştirme programlama becerisini de geliştirir.
Programlamanın matematiğe etkisi	Programlama dillerini kullanma becerilerinin matematik problemlerini çözme becerilerini geliştirmesi.	Programlama yapmak matematiksel düşünceyi geliştirir. Programlama yapmak matematiksel problem çözme sürecini hızlandırır. Programlama becerileri matematik becerilerini geliştirir.

Verilerin analizi sonucunda elde edilen matematiksel başarı teması öğrencilerin matematiğe karşı ilgi duyduklarını ve matematiksel problemleri çözmede kendini başarılı gördüklerini temsil eden temadır. Bu tema kapsamında, öğrencilerin matematiği sevdiğini ve kendini bu alanda başarılı gördüğünü ifade ettikleri buna karşın derin bir matematik bilgisine sahip olmadıklarını ancak çalışarak matematiği başarabildiklerini dile getirdikleri görülmüştür. Matematiksel başarı temasına ilişkin bazı katılımcılara ait görüşler şu şekildedir:

GÖ2: Matematiği severim, çocukluğumdan beri sevdiğim derslerden biridir. Onun için de bu bölümü tercih ettim.

GÖ6: Çok derin bir matematik bilgisine sahip değilim ama çalışarak matematiği başarabiliyorum. Ama bu bölüm için matematik bizim için bir ihtiyaç.

GÖ7: İleri derece olmasa da genel olarak matematikte başarılıyım.

Verilerin analizi sonucunda elde edilen bir diğer tema olan programlama başarısı öğrencilerin programlamaya karşı ilgi duyduklarını ve programlama dillerini kullanma konusunda kendilerini başarılı gördüklerini temsil etmektedir. Öğrencilerin çoğu programlamadan keyif aldıklarını, programlama konusunda problemlere çözüm üretebildiklerini ve programda yönlendirme yapabildiklerini dile getirdikleri görülmüştür. Ek olarak programlama konusunda daha yolun başında olduklarını ve kendilerini geliştirmeleri gerektiğini de ifade edenler de olmuştur. Programlama başarısı temasına ilişkin bazı katılımcılara ait görüşler şu şekildedir:

GÖ1: Programlama dilleri konusunda iyi derecede görüyorum kendimi. Ama öğrenecek çok şey var. Programlamadan zevk alıyorum ve severek yapıyorum.

GÖ8: Programcılık çok geniş ve her geçen gün yenilenen bir alan. Bizim de kendimizi bu manada sürekli geliştirmemiz gerekiyor.

GÖ12: Programlamada bir program yazmaktan ziyade bir çözüm yolu geliştirebiliyorum. Programlama üzerinde vakit harcamak gerek içine girince birçok hataları çıkabiliyor ama yönlendirmek babında çözüm üretebilirim.

Verilerin analizinde üçüncü olarak elde edilen tema matematik ve programlama arasında ilişki temasıdır. Bu tema matematikteki problem çözme ve algoritma mantığının programlamanın temeli olduğu, programlama için matematiğin ön koşul olduğu ve her ikisinin de benzer iş akışları gerektirdiği üzerine kuruludur. Bu bağlamda tema, matematik programlamanın temelidir ve matematik ve programlama benzerdir olmak üzere iki alt temaya ayrılmıştır. Matematik programlamanın temelidir alt temasına ilişkin olarak, öğrencilerin programlama temelindeki kavramların matematik kaynaklı kavramlar olduğunu, matematik ve programlamanın birbirini tamamladığını, matematiksel problemlerin aslında bilgisayarların altyapısını oluşturduğunu, bilgisayar ortamının temelini matematik olduğunu, programlama kodlarının aslında matematiksel kodlar olduğunu ve programlamanın matematiksel mantık üzerine kurulu olduğunu belirttikleri görülmüştür. Bu temaya ilişkin bazı katılımcılara ait görüşler şu şekildedir.

GÖ3: Matematiğin genel olarak bilgisayarların altyapısını oluşturur. Bilgisayardaki bütün işlemler matematiksel mantık üzerine kurulu ve böyle yürüyor. Bilgisayarlar programlama dilleri tamamen matematik üzerine kurulu. Bilgisayar ortamına girilen ve bilgisayar ortamından alınan verilerin tamamı matematik üzerine kurulu. Yani bilgisayar ortamının temeli matematiktir.

GÖ5: Programlama kodlarını yazarken bunlar matematiksel kodlar zaten. Kesinlikle matematiğe bağlı. Bunlar birbirini tamamlar ve birbiriyle ayrılmaz ilişki içerisindedir.

Matematik ve programlama benzerdir alt teması konusunda ise öğrencilerin matematik ve programlama mantığının aynı olduğu, ikisinin de problem çözme üzerine kurulu olduğu, benzer algoritmalar ve iş akışları kullandıkları yönünde görüş bildirdikleri görülmüştür. Bu konuda bazı öğrenci yorumlarına aşağıda yer verilmiştir.

GÖ3: Matematik yorumlamak ve çözümlemekle alakalı. Programlama kodları da öyle, ikisi de çözümleme üzerine kurulu.

GÖ7: Matematikte de bir yol deniyorsunuz olmazsa başka bir yol denersiniz birçok yol bulduysanız en pratik olanı tercih edersiniz. Programlamada da amaç en kısa ve pratik kodu yazmak. Mesela programlamada fonksiyonları kullanıyoruz. Bu fonksiyonların mantığı matematikte de aynı. Bir döngü kullanıyoruz, bir dizi kullanıyoruz. Bunlar matematikte de aynı. Yani programlama temelindeki kavram matematik kaynaklı kavramlar.

GÖ9: Programlama yaparken bir takım işlem basamakları vardır, bunlar algoritmalar. Matematik problemleri çözerken de işlem süreçleri var. Dolayısıyla nasıl programlamadaki işlem basamakları ilerletebiliyorsak matematikteki işlem basamakları da benzer şekilde ilerletiliyor.

GÖ11: Matematikte nasıl bir yol izliyorsak programlama dilinde de bu matematiksel işlemi yorumlayıp programlama diline çeviriyoruz. Mesela $a = 1$, $b = 2$ ve $c = 3$ dediğimiz zaman burada matematikte $a + b + c$ 'yi sorduğumuzda cevap 6 çıkıyor. Programlama dilinde ise bunu program diline çevirerek yazıyoruz yani mantık aynı, süreç aynı birbirine çok benzer ve çok yakın.

Analiz sonucu elde edilen matematiğin programlamaya etkisi teması ise matematiksel problemleri çözme becerisinin aslında programlama becerisini de geliştirebileceği aksine matematikteki başarısızlığın da programlamada başarısızlığa yol açabileceği fikrini barındırmaktadır. Bu konuda öğrencilerin genel olarak, matematik anlaşıldığında bilgisayar programlamanın da daha kolay anlaşılacağı, matematiğin aslında algıyı, teknik ve analitik düşünmeyi genişlettiği için programlama becerilerini de geliştirebileceği dolayısıyla matematik bilmeden program kodlarının yazılamayacağı fikrini savundukları görülmüştür. Öğrencilerin bu temaya ilişkin görüşlerine aşağıda yer verilmiştir.

GÖ1: Matematik problemleri çözmek ve matematik bilmek daha teknik düşünmenizi sağlar. Bu da programlama becerinizi geliştirir.

GÖ2: Bir problemi çözerken programlama dilini çok iyi biliyorum. C# dilini çok iyi biliyorum. Ondan sonra bana da problem verildi hemen bilgisayarın başına geçip başlıyorum. Öncelikle ben bu problemi ben çözebilirsem şayet o zaman bilgisayara da yaptırabilirim. Örneğin iki sayıyı nasıl toplarsın mesela bu soruyu ben kendim yapamıyorsa bilgisayara da yaptırırım. Bilgisayara yaptırmam için aslında çözümü kendim yapmam gerekiyor. Bu ise aslında matematik problem çözümüdür. Matematik problemi adım adım nasıl çözülebiliyorsa programa da o şekilde aktarabiliyorsunuz.

GÖ3: Programlama kodları matematiğe benzer şekildedir. Dolayısıyla matematik problemleri çözmek programı becerilerini geliştirebilir.

GÖ5: Matematiksel problemleri ne kadar iyi çözümlerseniz kodları yazmak da o kadar kolay oluyor.

GÖ6: Matematiksel olarak anladığınız zaman programlama olarak da daha iyi anlıyorsunuz. Ama matematikte başarısızsanız matematikle alakalı bir durum olduğundan programlamayı da başaramıyorsunuz, takılı kalıyorsunuz.

Verilerin analizi sonucu elde edilen programlamanın matematiğe etkisi teması ise programlama dillerini kullanma becerilerinin matematik problemlerini çözme becerilerini geliştirebileceği fikrini içerir. Çalışmaya katılan öğrenciler matematiksel problemleri çözme becerisinin programlamaya etkisi olduğu gibi tersi yönde de bir etkinin olabileceğini yani programlama becerisinin de matematiksel problemleri çözme becerisini geliştirebileceğini ve problem çözme sürecini de hızlandırabileceğini belirtmişlerdir. Bu temaya ilişkin bazı öğrencilere ait görüşler ise şu şekildedir:

GÖ3: Mantık olarak aynı olduğu için programlama beceriniz matematikte de meyvesini gösteriyor.

GÖ6: Programlamanın da bir getirisi vardır matematiğe. Programlama daha teknik düşünmenizi sağlar. Bu da matematiği geniş açıdan düşünmenizi sağlar. Programlama, matematiğe daha çok kafa yormanızı ve geniş açıdan bakmanızı sağladığı için matematiği de geliştirir.

Çalışmaya katılan öğrencilerden programlama dillerini kullanma becerilerinin aslında matematik problemlerini çözme becerilerini geliştirmeyeceği yönünde fikrini bildiren bir öğrenciye de rastlanılmıştır. Örneğin GÖ4 “Bu konuda benim için programlama dillerini kullanmak daha kolay. Çünkü programlamada belli bir kod ve kod mantığı var. Bütün işlemleri bilgisayar yapıyor. Mesela bir logaritma alacağınız zaman $\log_{10} 10 = 1$ 'dir. Bunu matematiksel olarak hesaplamanıza gerek yok. Bizim bunu matematiksel olarak bilmemiz yerine program dili bunu bildiği için bu tip hesapları yapmak kolay. Dolayısıyla bu matematiksel gelişiminizi etkilemez. Ama tabi programlamada bunu nasıl programlayacağınızı bilmek gerekir.” şeklinde görüş bildirerek programlama becerilerinin matematiksel beceriyi geliştirmediğini düşündüğünü belirtmiştir.

TARTIŞMA VE SONUÇ

Bu çalışmanın sonuçları çalışmaya katılan meslek yüksekokulu bilgisayar programcılığı öğrencilerinin matematiksel problemleri çözmede genel olarak düşük başarılar elde ettiklerini göstermektedir. Çalışmaya katılan öğrencilerin yaklaşık yarısının aritmetik ortalamaya ilişkin hesapların yapılmasını gerektiren problemleri doğru yanıtlayabildikleri görülürken, 1/4'ne yakını yüzde ve karışım problemleri ile yaş problemlerine ilişkin hesapların yapılabilmesini gerektiren problemlerin çözümünde başarı gösterebilmişlerdir. Çalışmada yer alan öğrenciler en düşük başarıyı ise işçi problemlerine ilişkin hesapların yapılabilmesi ile çokgenlerin alanlarının hesaplanabilmesini gerektiren problemlerinin çözülmesinde göstermişlerdir. Bu durum genel olarak meslek yüksekokulu bilgisayar programcılığı öğrencilerinin matematiksel problem çözme becerisine düşük ya da yetersiz düzeyde sahip olduklarını göstermektedir. İlgili literatürde de programcılık öğrencilerin genellikle temel matematik becerilerini gösteremedikleri, sayı teorileri ile ilgili yeterli becerilere sahip olmadıkları ve temel matematiksel kavramlarda dahi derin zorluklar yaşadıkları sonuçlarına ulaşan çalışmalara rastlanılmaktadır (Gomes & Mendes, 2014; Ismail ve ark., 2010; Kaya ve ark., 2013; Pacheco ve ark., 2008b). Genel olarak matematik öğretiminin amaçları arasında öğrencilerin problem çözme stratejilerini geliştirmek yer alsa da öğrencilerin problem çözmede gerekli matematik yaklaşımlarını etkili ve başarılı bir biçimde ortaya koyamadıkları görülmektedir (Altun & Arslan, 2006). Programcılık öğrencilerinin matematikteki bu başarısızlığın nedenleri arasında öğrencilerin birçoğunun meslek liselerinden mezun olmaları ve liselerde matematik derslerini yeterli düzeyde görmemeleri, bu nedenle sağlam bir matematik alt yapısına yeterli düzeyde sahip olmamaları ve bu ön yargılarla yüksek öğretim kurumlarındaki matematik derslerine katılmaları gösterilebilir. Öğrencilerin aldıkları matematik eğitiminin yetersiz veya uygulamadan uzak, dersin öneminin anlaşılmasında eksikliklerin olması veya öğrencilerin matematiği çalışma metotlarındaki yanlışlıklar da bu başarısızlığın sebepleri olarak görülebilir.

Öğrencilerin matematiksel problemleri çözme becerilerinin genel olarak bilgisayar programlama becerileri üzerinde olumlu etkiye sahip olduğu yönünde görüş bildirdikleri görülmüştür. Öğrenciler, matematiği programcılık için değerli ve gerekli görmekte, matematiğin programlama için bir ön koşul olduğunu ve bilgisayarların altyapısını oluşturduğunu, programlamanın matematiksel mantık üzerine kurulu olduğunu düşünmektedirler. Bu sebeple matematiksel problemleri çözmenin programlama becerisini geliştirebileceğini belirtmektedirler. Nitekim öğrencilerin bilgisayar programlama becerilerini geliştirmek problem çözme süreciyle yakından ilgili görülmektedir (Oluk & Çakır, 2019; Wang

ve ark., 2014). Çünkü programlama; soyutlama, matematiksel mantık, sınama, hata ayıklama ve sorunları çözme gibi karmaşık görevleri içerir (Saeli ve ark., 2011). Bu ise problemin nasıl çözüleceğinin ilk olarak matematiksel olarak yapılmasını gerektirir (Psycharis & Kallia, 2017). Dolayısıyla problem çözme, başarılı bir bilgisayar programı tasarlamak ve geliştirmek adına oldukça önemlidir (Razak & Ismail, 2018). İlgili literatürde de problem çözme becerisinin programlama başarısı üzerinde etkili olduğunu (Altın ve ark., 2021; Gomes & Mendes, 2014; Ismail ve ark., 2010; Pacheco ve ark., 2008b; Yılmaz & Çakır, 2019), matematik yeteneğinin bilgisayar bilimlerine kabul edilmek için bir ön koşul olduğunu (Byrne & Lyons, 2001), bilgisayar donanımının ve yazılımının nasıl çalıştığını anlamak açısından matematik becerisine ihtiyaç duyulduğunu (Truss, 1991), öğrencilerin problem çözme becerilerinin geliştirilmesinin aynı zamanda programlama becerilerini geliştirdiğini (Barlow-Jones & Westhuizen, 2017), problem çözme becerilerine sahip olmamanın ise düşük programlama becerilerine yansıdığını (Gomes ve ark., 2006; Pacheco ve ark., 2008a) bildiren çalışmalar yer almaktadır. Programcılık mantıksal düşünmeyi gerektirir ve bu, bir programcının sahip olması gereken en önemli özelliklerden biridir. Bir şeyleri mantıksal düşünce üzerine inşa etmek için, kavramı anlamak gerekir, bu da daha fazla fikre yol açar ve problem çözme daha da basit hale gelir. Dolayısıyla, programlama hakkında öğrenmeye başlamak için programlamanın temellerini yani matematiği anlamayı gerektirir (Prasad ve ark., 2022).

Çalışmaya katılan öğrenciler programlama becerilerinin de matematiksel problemleri çözme becerileri üzerinde olumlu etkisi olduğunu belirtmişlerdir. Buna sebep matematiğin programlama ile aynı bilişsel becerileri gerektirmesi görülmektedir (Byrne & Lyons, 2001). İlgili literatürde de programlama becerilerinin problem çözme becerileri üzerinde olumlu etkileri olduğunu (Erümit ve ark., 2019; Gökçe ve ark., 2017; Pala & Mihci Türker, 2019; Pardamean ve ark., 2011; Taylor ve ark., 2010) ve matematik problemlerini çözmeye karşılaşılan hataları düzeltmeyi sağladığını (Lovric, 2018) gösteren araştırmalar bulunmaktadır. Programlama süreçleri farklı düşünme becerilerini içerdiğinden, bu süreçler problem tanımlama, formül oluşturma ve çözüm üretme gibi becerilerin geliştirilmesine yardımcı olur (Partnership for 21st Century Skill, 2007). Bir programlama dilini kullanmak öğrencilerin sadece kaynak kodu yazmaları anlamına gelmediği gibi bunun soyut ve mantıksal düşünmenin farklı yönlerini içerdiği açıktır. Bu bilgi, yalnızca bilgisayar bilimi için değil matematik gibi bir alan için de önemli bir katkı sağlamaktadır (Psycharis & Kallia, 2017).

Günümüzde bir bilişim programı mezunundan, gerçek dünya problemlerini analiz edebilmesi, mevcut donanım ve yazılım sistemlerini kullanarak problemler için verimli

çözümler sunabilmesi, problemleri çözmek için ilgili algoritmalara, veri yapılarına ve kalıplara hakim olabilmesi, bir çözümün doğruluğunu ve etkinliğini matematiksel yöntemlerle değerlendirebilmesi beklenmektedir. Bu yetenekler ise, sağlam bir matematik eğitime dayanmaktadır (Ottmann ve ark., 2008). Nitekim çalışmaya katılan öğrenciler matematiksel problemleri çözmede düşük başarılar elde etmelerine rağmen matematiğin bilgisayar programlamanın temeli olduğunu ve aslında matematiksel problem çözme becerilerini geliştirmenin programlama becerilerini de geliştireceğinin farkında olduklarını dile getirmişlerdir. Bu durum, öğrencilerin algıları ile pratik becerileri arasındaki farklardan kaynaklanabilir. Öğrenciler matematiksel kavramları derinlemesine öğrenmede, bu kavramları algoritmik düşünceyle birleştirmede ve gerçek hayattaki problemlere uygulamada zayıf olsalar da bu öğrenciler bilgisayar programlamada karşılaşılan karmaşık problemlerin çözümü için algoritmaların kurulması ve sistem tasarlamada matematiksel düşünceden faydalanmaktadırlar. Programlama yaparken öğrenciler, bir problemi çözmek için adımları belirleyerek bir algoritma oluştururlar. Benzer şekilde matematiksel problemleri çözerken de adımları belirlemek ve bir çözüm stratejisi oluşturmak gerekir. Bu nedenle, matematiksel problem çözme becerilerini geliştirmenin beraberinde programlama becerilerini de geliştirebileceği sonucuna ulaşabilirler. Dolayısıyla öğrencilerin programlama konularıyla uyumlu problem çözme, formülasyon, algoritmik düşünme, olası çözümlerin tanımlanması ve genellemeyi içeren süreçler dahilinde öğrenim görmesi bu öğrencilerin hem programlamada hem de matematikte daha etkili sonuçlar elde etmesini sağlayacaktır.

Sınırlılıklar ve öneriler

Problem çözme yeteneği bilgisayar programcılarında beklenen profesyonel bir davranıştır. Ancak bu çalışmada öğrencilerin problem çözme becerilerinin çok iyi düzeylerde olmadığı saptanmıştır. Öğrencilerin bilgisayar programlama konusunda ihtiyaç duyulan gerekli eleştirel düşünme becerilerini edinebilmeleri için programcılık eğitim müfredatlarında problem çözme becerilerinin kazandırılmasına yönelik ders içeriklerine yer verilebilir. Bu derslerde öğrencilerin problem çözme becerileri onları problem çözme pratiği yapmaya teşvik ederek geliştirilebilir ve öğrenciler problem çözerken geçirdikleri süreçlere benzer süreçleri programlama becerilerini geliştirmek için nasıl kullanacakları konusunda cesaretlendirilebilir. Ek olarak matematik derslerinde problem çözme stratejilerinin etkin bir şekilde uygulanması ve öğrencilerinin de bunları farklı alanlarda kullanmaları desteklenebilir.

Bu çalışma ile bilgisayar programcılığı programı öğrencilerinin matematiksel problem çözme becerileri ile problem çözme ve bilgisayar programlama dillerini kullanmaya dair

görüşlerine ilişkin bir bakış açısı sunulsa da araştırma sonuçlarının genellenebilirliğini artırmak için benzer çalışmalar farklı örneklem grupları üzerinde yürütülebilir. Ayrıca öğrencilerin programlama ve matematiksel problem çözme becerilerinin bir arada ortaya konulacağı ileri araştırmalar yapılabilir. Özellikle problem çözme sürecini ve bu süreçteki öğrenci davranışlarını ölçme tekniklerini geliştirici çalışmalar yapılabilir. Bu çalışmalarda öğrencilerin hangi zihinsel süreçlerden geçtikleri daha ayrıntılı biçimde değerlendirilerek problem çözme öğretimi etkinliklerine ışık tutulabilir. Öğrencilerin zorlandıkları alanların bilinmesi ve anlaşılması ile programlama öğrenimini ve öğretimini geliştirmek için daha verimli bir şekilde hareket edilebilir. Ek olarak problem temelli öğretim ile tasarlanmış öğrenme ortamlarının öğrencilerin programlama derslerindeki başarısı üzerine etkisinin araştırıldığı çalışmalar yürütülebilir. Ayrıca programlama öğretimi sürecine etki eden farklı faktörlerin de belirlenerek bu faktörlere yönelik çözüm önerilerinin geliştirilebileceği çalışmalar tasarlanabilir.

Araştırmada öğrencilerin problem çözme becerileri araştırmacı tarafından tasarlanan problem çözme beceri testi aracılığı ile belirlenmiştir. Bu durum araştırmanın bir sınırlılığı olarak ele alınabilir. Öğrencilerin daha farklı problem türlerini çözebilmelerine ve farklı problem çözme stratejilerini uygulayabilmelerine hatta kendi problem çözme stratejilerini geliştirmelerine olanak tanıyan ölçme araçlarının kullanıldığı ek olarak öğrencilerin kullandıkları problem çözme stratejilerini programlama becerileri ile nasıl ilişkilendirdiklerini ortaya koymak için nitel ve nicel çalışma süreçlerinin bir arada yürütülebileceği ileri araştırmalar yapılabilir.

Etik metni

Bu makalede dergi yazım kurallarına, yayın ilkelerine, araştırma ve yayın etiği kurallarına, dergi etik kurallarına uyulmuştur. Makale ile ilgili doğabilecek her türlü ihlallerde sorumluluk yazara aittir.

KAYNAKLAR

- Aho, A. V. (2012). Computation and computational thinking. *The Computer Journal*, 55(7), 832-835. <https://doi.org/10.1093/comjnl/bxs074>
- Altın, R., Tokel, T., & Delialioğlu, O. (2021). The effects of mathematics on programming skills and its retention: An experimental study. *Journal of Computers in Mathematics and Science Teaching*, 40(3), 183-199.
- Altun, M., & Arslan, Ç. (2006). İlköğretim öğrencilerinin problem çözme stratejilerini öğrenmeleri üzerine bir çalışma. *Uludağ Üniversitesi Eğitim Fakültesi Dergisi*, 19(1), 1-21.
- Ambrosio, A. P., Costa, F. M., Almeida, L., Franco, A., & Macedo, J. (2011). Identifying cognitive abilities to improve CS1 outcome. In *Frontiers in Education Conference (FIE)*, F3G-1. <https://doi.org/10.1109/FIE.2011.6142824>

- Antonakos, J. L. (2016). *Computer technology and computer programming: Research and strategies*. Boca Raton, Florida: CRC Press.
- Arabacıoğlu, T., Bülbül, H. İ., & Filiz, A. (2007). Bilgisayar programlama öğretiminde yeni bir yaklaşım. *Akademik bilişim*, 193-197.
- Baist, A., & Pamungkas, A. S. (2017). Analysis of student difficulties in computer programming. *VOLT: Jurnal Ilmiah Pendidikan Teknik Elektro*, 2(2), 81-92. <https://doi.org/10.30870/VOLT.V2I2.2211>
- Barlow-Jones, G., & Westhuizen, D. V. D. (2017). Problem solving as a predictor of programming performance. In *Annual Conference of the Southern African Computer Lecturers' Association* (pp. 209-216). Springer, Cham.
- Byrne, P., & Lyons, G. (2001). The effect of student attributes on success in programming. *SIGCSE Bulletin*, Vol. 33 (3), pp. 49-52. <https://doi.org/10.1145/507758.377467>
- Cheah, C. S. (2020). Factors contributing to the difficulties in teaching and learning of computer programming: A literature review. *Contemporary Educational Technology*, 12(2), ep272. <https://doi.org/10.30935/cedtech/8247>
- Cohen, J. (1960). A coefficient of agreement for nominal scales. *Cohe*, (1), 37-46.
- Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2022). *Introduction to algorithms*. MIT press.
- Creswell, J. W. (1998). *Qualitative inquiry and research design: Choosing among five traditions*. Thousand Oaks, California: Sage.
- Creswell, J. W. (2012). *Research design: Qualitative, quantitative, and mixed methods approaches*. (4th Edition), Sage publications.
- Davydov, V. V. (2008). *Problems of developmental instruction. A theoretical and experimental psychological study*. In D. Robbins & V. Lektorsky (Eds.), *International perspectives in nonclassical psychology*. New York, NY: Nova Science.
- Einhorn, S. (2012). Microworlds, computational thinking, and 21st century learning. *LCSI White Paper*, 1-10.
- Elçiçek, M., & Karal, H. (2020). Cognitive problems in the process of programming teaching in higher education: learner-instructor experiences. *Turkish Online Journal of Qualitative Inquiry*, 11(1), 140-160. <https://doi.org/10.17569/tojqi.639139>
- Elkin, N., & Karadağlı, F. (2015). Üniversite öğrencilerinin problem çözme becerilerinin değerlendirilmesi. *Adıyaman Üniversitesi Sağlık Bilimleri Dergisi*, 1(1), 11-18.
- Erümit, K. A., Karal, H., Şahin, G., Aksoy, D. A., Aksoy, A., & Benzer, A. I. (2019). A model suggested for programming teaching: Programming in seven steps. *Eğitim ve Bilim*, 44(197). <https://doi.org/10.15390/EB.2018.7678>
- Gökçe, S., Yenmez, A. A., & Özpınar, I. (2017). An analysis of mathematics education students' skills in the process of programming and their practices of integrating it into their teaching. *International Education Studies*, 10(8), 60-76. <https://doi.org/10.5539/ies.v10n8p60>
- Gomes, A., Carmo, L., Bigotte, E., & Mendes, A. (2006). *Mathematics and programming problem solving*. In 3rd e-learning conference—computer science education (pp. 1-5).
- Gomes, A., & Mendes, A. J. (2007, September). Learning to program—difficulties and solutions. In *International Conference on Engineering Education—ICEE* (Vol. 7, pp. 1-5).

- Gomes, A., & Mendes, A. (2014). A teacher's view about introductory programming teaching and learning: Difficulties, strategies and motivations. In *2014 IEEE Frontiers in Education Conference (FIE) Proceedings* (pp. 1-8). IEEE.
- Guba, E. G., & Lincoln, Y. S. (1982). Epistemological and methodological bases of naturalistic inquiry. *Ectj*, 30(4), 233-252.
- Hariawan, H., Kamaluddin, K., & Wahyono, U. (2013). Pengaruh model pembelajaran creative problem solving terhadap kemampuan memecahkan masalah fisika pada siswa kelas XI SMA Negeri 4 Palu. *JPFT (Jurnal Pendidikan Fisika Tadulako Online)*, 1(2), 48-54. <https://doi.org/10.22487/j25805924.2013.v1.i2.2395>
- Helminen, J., & Malmi, L. (2010). *Jype-a program visualization and programming exercise tool for Python*. In Proceedings of the 5th international symposium on Software visualization (pp. 153-162). ACM. <https://doi.org/10.1145/1879211.1879234>
- İnce, M. (2021). LSTM Based Source Code Generation for Turkish Pseudo Code of Algorithm. *Düzce Üniversitesi Bilim ve Teknoloji Dergisi*, 9(1), 104-113. <https://doi.org/10.29130/dubited.824799>
- Ismail, M. N., Ngah, N. A., & Umar, I. N. (2010). Instructional strategy in the teaching of computer programming: a need assessment analyses. *TOJET: The Turkish Online Journal of Educational Technology*, 9(2).
- Kreitzberg, C. B., & Swanson, L. (1974). *A cognitive model for structuring an introductory programming curriculum*. In Proceedings of the May 6-10, 1974, national computer conference and exposition (pp. 307-311).
- Kaya, Y., Özdemir, S., & Utkun, E. (2013). Meslek yüksekokulu öğrencilerinin matematik başarısını etkileyen faktörler: öğrenci görüşleri bakımından. *Electronic Journal of Vocational Colleges* 62-68
- Landis, J., R., & Koch, G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, 33, 159-174.
- Lesh, R., & Zawojewski, J. S. (2007). *Problem solving and modeling*. In F. K. Lester (Ed.), *Handbook of research on mathematics teaching and learning* (2nd Edition, pp. 763–804). Information Age.
- Levitin, A. (2012). *Introduction to design and analysis of algorithm*. Pearson, Boston.
- Lovric, M. (2018). Programming and mathematics in an upper-level university problem-solving course. *Primus*, 28(7), 683-698. <https://doi.org/10.1080/10511970.2017.1403524>
- Mathew, R., Malik, S. I., & Tawafak, R. M. (2019). Teaching Problem Solving Skills using an Educational Game in a Computer Programming Course. *Informatics in Education*, 18(2), 359-373. <https://doi.org/10.15388/infedu.2019.17>
- McCoy, L. P., & Burton, J. K. (1988). The relationship of computer programming and mathematics in secondary students. *Computers in the Schools*, 4(3-4), 159-166.
- Mhashi, M. M., & Alakeel, A. (2013). Difficulties facing students in learning computer programming skills at Tabuk University. *Recent Advances in Modern Educational Technologies*, 15-24.
- Miles, M. B. & Huberman, A. M. (1994). *Qualitative data analysis: An expanded Sourcebook*. (2nd ed). Thousand Oaks, CA: Sage
- Muller, O. (2005). Pattern oriented instruction and the enhancement of analogical reasoning. In *Proceedings of the first international workshop on Computing education research* (pp. 57-67).
- Olkun, S., & Toluk, Z. (2004). *İlköğretimde etkinlik temelli matematik öğretimi*. Ankara: Anı Yayıncılık.

- Oluk, A., & Çakır, R. (2019). Üniversite öğrencilerinin bilgisayarca düşünme becerilerinin mantıksal matematiksel zekâ ve problem çözme becerileri açısından incelenmesi. *Journal of Theoretical Educational Science*, 12(2), 457-473.
- Ottmann, T., Hermann, C., & Heumann, C. (2008). Accreditation practice for degree programs in Computer Science: Experience gained at a classical research university in Germany. *ACM-IFIP*.
- Pacheco, A., Gomes, A., Henriques, J., Almeida, A. M., & Mendes, A. (2008a). A study on basic mathematics knowledge for the enhancement of programming learning skills. *Proceedings of IEEE/08-Informatics Education Europe III*.
- Pacheco, A., Gomes, A., Henriques, J., de Almeida, A. M., & Mendes, A. J. (2008b). *Mathematics and programming: Some studies*. In Proceedings of the 9th International Conference on Computer Systems and Technologies and Workshop for PhD Students in Computing (pp. V-15).
- Pala, F. K., & Mihci Türker, P. (2019). The effects of different programming trainings on the computational thinking skills. *Interactive Learning Environments*, 1-11. <https://doi.org/10.1080/10494820.2019.1635495>.
- Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. Basic Books, Inc
- Pardamean, B., Honni, H., & Evelin, E. (2011). The effect of logo programming language for creativity and problem solving. *Proceedings of the 10th WSEAS international conference on E-Activities. World Scientific and Engineering Academy and Society (WSEAS)*, 151–156.
- Patton, M. Q. (2002). *Qualitative research and evaluation methods* (Vol. 3). Sage.
- Partnership for 21st Century Skills (P21)., (2007). *Partnership for 21st century skills*. 09.12.2022 tarihinde https://www.marietta.edu/sites/default/files/documents/21st_century_skills_standards_book_2.pdf adresinden erişilmiştir.
- Polya, G. (2004). *How to solve it: A new aspect of mathematical method* (Vol. 85). Princeton university press.
- Prasad, A., Chaudhary, K., & Sharma, B. (2022). Programming skills: Visualization, interaction, home language and problem solving. *Education and Information Technologies*, 27(3), 3197-3223. <https://doi.org/10.1007/s10639-021-10692-z>
- Prayekti, N., Nusantara, T., Sudirman, Susanto, H., & Rofiki, I. (2020). Students' mental models in mathematics problem-solving. *Journal of Critical Reviews*, 7(12), 468–470. <http://dx.doi.org/10.31838/jcr.07.12.83>
- Psycharis, S., & Kallia, M. (2017). The effects of computer programming on high school students' reasoning skills and mathematical self-efficacy and problem solving. *Instructional science*, 45(5), 583-602. <https://doi.org/10.1007/s11251-017-9421-5>
- Rahmat, M., Shahrani, S., Latih, R., Yatim, N. F. M., Zainal, N. F. A., & Ab Rahman, R. (2012). Major problems in basic programming that influence student performance. *Procedia-Social and Behavioral Sciences*, 59, 287-296.
- Razak, M. R. B., & Ismail, N. Z. B. (2018). Influence of mathematics in programming subject. In *AIP Conference Proceedings* (Vol. 1974, No. 1, p. 050011). AIP Publishing LLC.
- Saeli, M., Perrenet, J., Jochems, W. M. G., Zwaneveld, B., Nederland, O. U., & Centrum, R. D. M. (2011). Teaching programming in secondary school: A pedagogical content knowledge perspective. *Informatics in Education*, 10(1), 73–88. <http://doi.org/10.1145/2016911.2016943>

- Syahlan, S., & Simamora, R. (2022). Improve students' mathematical critical thinking skills sumatera utara islam: using models for problem-based learning. *Aksioma: Jurnal Program Studi Pendidikan Matematika*, 11(4), 3040-3051. <http://dx.doi.org/10.24127/ajpm.v11i4.5503>
- Taylor, M., Harlow, A., & Forret, M. (2010). Using a computer programming environment and an interactive whiteboard to investigate some mathematical thinking. *Procedia Social and Behavioral Sciences*, 8(1), 561–570. <https://doi.org/10.1016/j.sbspro.2010.12.078>
- Truss, J. K. (1991). *Discrete Mathematics for Computer Scientists (International Computer Science Series)*. Addison-Wesley Longman Publishing Co., Inc.
- Wang, H. Y., Huang, I., & Hwang, G. J. (2014). *Effects of an integrated Scratch and projectbased learning approach on the learning achievements of gifted students in computer courses*. Paper presented at the Advanced Applied Informatics (IIAIAI), 2014 IIAI 3rd International Conference on, Kitakyushu, Japan.
- Wing, J. M. (2006). Computational Thinking. *Communications of the ACM*, 49(3), 33-35
- Yıldırım, A., & Şimşek, H. (2000). *Sosyal Bilimlerde Nitel Araştırma Yöntemleri*. Ankara: Seçkin Yayıncılık.
- Yılmaz, F., & Çakır, H. (2019). Meslek yüksekokulu öğrencilerinin programlama başarılarını etkileyen faktörlerin incelenmesi. *Eğitim teknolojisi kuram ve uygulama*, 9(2), 416-437. <https://doi.org/10.17943/etku.527202>
- Yin, R.K. (2009). *Case study research: Design and methods*. 4th ed. Sage, Thousand Oaks.
- Yin, R.K. (2014). *Case study methods: design and methods*. Thousand Oaks: Sage Pbc.

KATKI ORANI CONTRIBUTION RATE	AÇIKLAMA EXPLANATION	KATKIDA BULUNANLAR CONTRIBUTORS
Fikir ve Kavramsal Örgü Idea or Notion	Araştırma hipotezini veya fikrini oluşturmak Form the research hypothesis or idea	Rahime ÇELİK GÖRGÜT
Tasarım Design	Yöntem ve araştırma desenini tasarlamak To design the method and research design.	Rahime ÇELİK GÖRGÜT
Literatür Tarama Literature Review	Çalışma için gerekli literatürü taramak Review the literature required for the study	Rahime ÇELİK GÖRGÜT
Veri Toplama ve İşleme Data Collecting and Processing	Verileri toplamak, düzenlemek ve raporlaştırmak Collecting, organizing and reporting data	Rahime ÇELİK GÖRGÜT
Tartışma ve Yorum Discussion and Commentary	Elde edilen bulguların değerlendirilmesi Evaluation of the obtained finding	Rahime ÇELİK GÖRGÜT
Destek ve Teşekkür Beyanı/ Statement of Support and Acknowledgment		
Bu çalışmanın yazım sürecinde katkı ve/veya destek alınmamıştır. No contribution and/or support was received during the writing process of this study.		
Çatışma Beyanı/ Statement of Conflict		
Araştırmacının araştırma ile ilgili diğer kişi ve kurumlarla herhangi bir kişisel ve finansal çıkar çatışması yoktur. Researchers do not have any personal or financial conflicts of interest with other people and institutions related to the research.		
Etik Kurul Beyanı/ Statement of Ethics Committee		
Bu araştırma, Kastamonu Üniversitesi Etik Kurulunun 27.02.2023 tarihli ve 3/1 sayılı kararı ile yürütülmüştür. This research was conducted with the decision of Kastamonu University Ethics Committee dated 27.02.2023 and numbered 3/1.		



Bu eser [Creative Commons Atıf-Gayri Ticari 4.0 Uluslararası Lisansı \(CC BY 4.0\)](https://creativecommons.org/licenses/by/4.0/) ile lisanslanmıştır.