





# Düzce Üniversitesi Bilim ve Teknoloji Dergisi

*Araştırma Makalesi*

## Programlamaya Giriş Dersini Alan Öğrencilerin Programlama Öz Yeterlilik Algılarının ve Programlamaya Bakış Açılarının İncelenmesi

 Kristin S. BENLİ<sup>a,\*</sup>,  F. Boray TEK<sup>b</sup>

<sup>a</sup> Yazılım Mühendisliği Bölümü, Mühendislik ve Doğa Bilimleri Fakültesi, Üsküdar Üniversitesi, İstanbul, TÜRKİYE

<sup>b</sup> Bilgisayar Mühendisliği Bölümü, Mühendislik Fakültesi, Işık Üniversitesi, İstanbul, TÜRKİYE

\* Sorumlu yazarın e-posta adresi: kristin.benli@uskudar.edu.tr

DOI:10.29130/dubited.770726

### ÖZET

Bu çalışmada üniversite öğrencilerinin Java programlama öz yeterlilik algıları, programlama öğrenme istekleri ve çalışma alışkanlıkları çeşitli değişkenlere göre (cinsiyet, bölüm, eğitim dili, harf notu, ders tekrarları vb.) istatistiksel yöntemler kullanılarak (T-testi, Mann Whitney U-testi, Kruskal Wallis H testi, tek yönlü varyans analizi, Ki-Kare testi) incelenmiştir. Çalışma grubu, farklı bölümlerde zorunlu olarak programlamaya giriş dersini alan 191 lisans öğrencisinden oluşmaktadır. Elde edilen sonuçlara göre öğrencilerin Java programlama öz yeterlilik algıları bölümlerine ve programlama öğrenme isteklerine göre farklılaşmaktadır. Çalışmada ayrıca Apriori algoritması kullanılarak birliktelik kuralları çıkartılmıştır. En yüksek güven değeri elde edilen kurala göre, programlama öğrenmeyi çok fazla isteyen, programlama öğrenmenin iş hayatında kendisine fayda sağlayacağını düşünen ve programlama dersinden başarı ile geçen öğrencilerin programlama öz yeterlilikleri yüksektir.

**Anahtar Kelimeler:** Programlama öz yeterliliği, Programlama eğitimi, İstatistik, Birliktelik kuralı, Apriori algoritması

## Investigating Introduction to Programming Course Students' Programming Self Efficacy Perceptions and Perspectives on Programming

### ABSTRACT

This study investigates university student's (Java) programming self-efficacy, perspectives on programming, and study habits with respect to various variables (gender, department, course language, letter grade, course repeats) by using statistical methods (T-test, Mann Whitney U-test, Kruskal Wallis H test, one-way analysis of variance, Chi-Square test). The study group consisted of 191 undergraduate students who had to take the introductory programming course from different departments. The results indicate that students' Java programming self-efficacy have relationships with their department and desire to learn programming. Furthermore, in this study, association rules were obtained by using Apriori algorithm. The highest confidence value rule suggests that higher programming self-efficacy can be associated with higher levels of desire to learn programming, higher belief in the utility of programming in professional life and having successfully passed the programming course.

**Keywords:** Programming efficacy, Programming education, Statistics, Association rule, Apriori Algorithm

# I. GİRİŞ

Günümüzde teknolojinin hızlı bir şekilde gelişmesi ile yazılım insan hayatının her alanına girmiş ve beraberinde yazılım geliştirme becerisine sahip kişilere olan talebi de arttırmıştır. Gelişen talepler doğrultusunda üniversitelerin Mühendislik ve Fen Edebiyat Fakültelerinde yer alan bölümlerin müfredatlarına da programlamaya giriş dersleri eklenmiştir. Bölümlerinde gösterilen çoğu ders (matematik, fizik, kimya vb.) ile ilgili daha önceden bir alt yapıya sahip olan öğrenciler, programlamaya giriş dersinde ilk defa karşılaştıkları algoritma kavramını anlamakta güçlük yaşamaktadır.

Ezberden çok soyut düşünmeye dayalı programlama mantığını anlamada sıkıntı yaşayan öğrencilerin bir kısmı en baştan dersin zor olduğuna kanaat getirip çaba sarf etmeden ya direkt dersten çekilmekte veya düşük istekle derslere devam edip dersten başarısız olmaktadır. Sonraki süreçlerde bu öğrenciler programlamaya giriş dersini tekrar almak durumunda kalmaktadır. Karşılaşılan bu durumlar öğrencilerin öz yeterlilik algılarının düşük olması şeklinde yorumlanmaktadır [1].

Öz yeterlilik, bireyin bir görevi tamamlama veya bir hedefe ulaşmadaki becerisine olan inancı olarak ifade edilir [2]. Karşılarına çıkan zorluklarla başa çıkabilen, engelleri aşmak için çaba sarf eden, hedeflerine ulaşmak konusunda kararlı olan bireylerin öz yeterlilikleri yüksektir. Literatür incelendiğinde genel öz yeterlilik ölçeğinin matematik [3], bilgisayar kullanımı [4] ve programlama [5] gibi farklı alanlara uyarlandığı görülür.

Ramalingam ve Wiedenbeck [5] C++ programlama dili için bir öz yeterlilik ölçeği geliştirmiş, kişilerin cinsiyetlerine göre programlama öz yeterliliklerinin farklı olup olmadığını incelemişlerdir. Bu ölçek daha sonraki çalışmalarda araştırmacılar tarafından farklı programlama dillerine (Java [6], C# [7]) uyarlanmış, Türkçe diline çevrilmiştir [1,7,8].

Askar ve Davenport'un [6] yapmış oldukları çalışmada Java programlama diline uyarladıkları ölçek ile mühendislik fakültesi öğrencilerinin programlama öz yeterlilik algılarının cinsiyet, bölüm, bilgisayar becerileri, bilgisayar deneyimi, bilgisayar kullanım sıklığı ve aile bireylerinin bilgisayar kullanımı gibi çeşitli değişkenler ile olan ilişkilerini incelemiştir. Elde ettikleri sonuçlara göre erkek ve kadın öğrencilerin aralarında programlama öz yeterliliği açısından anlamlı bir fark vardır. Bilgisayar Mühendisliği bölümü öğrencilerinin öz yeterlilik puanları diğer mühendislik bölümü öğrencilerinin puanlarından yüksektir. Her gün bilgisayar kullanan öğrenciler ile haftada birkaç kez kullananlar arasında anlamlı fark bulunmuş, daha sık kullananların öz yeterlilik sonuçlarının daha yüksek olduğu belirtilmiştir. Aile bireylerinden annenin ve kardeşlerin bilgisayar kullanımının öğrencilerin öz yeterlilik algılarında etkili olduğu görülmüştür.

Altun ve Mazman [1] çalışmalarında Ramalingam ve Wiedenbeck'in [5] geliştirdikleri ölçeği herhangi bir programlama dilinden bağımsız olarak genel programlama öz yeterlilik ifadelerini içerecek şekilde Türkçe olarak düzenlemiştir. Ayrıca çalışmalarında öğrencilerin programlama öz yeterlilik algılarını cinsiyet, sınıf, bölüm, programlama deneyim yılı ve programlamaya ilişkin alınan ders sayısı gibi değişkenler açısından incelemiştir. Analiz sonuçlarında cinsiyet dışındaki tüm değişkenlere göre öğrencilerin programlama öz yeterlilik algılarının farklılaştığı görülmüştür.

Korkmaz ve Altun [8], Türkçe'ye uyarladıkları ölçek ile mühendislik fakültesi öğrencilerinin C++ programlama öz yeterlilik algılarının cinsiyete ve bölüme göre farklılık gösterip göstermediğini araştırmıştır. Çalışma sonucunda Bilgisayar Mühendisliği öğrencilerinin Elektronik Mühendisliği öğrencilerine göre daha yüksek programlama öz yeterliliğine sahip olduğu görülmüştür.

Çiğdem ve Yıldırım [7] yapmış oldukları çalışmada C# programlama diline Türkçe olarak uyarladıkları ölçek ile meslek yüksekokulu öğrencilerinin programlama öz yeterlilik algılarının yaş, mezun olunan lise türü, bilgisayar kullanma deneyimi, bilgisayar kullanım sıklığı ve programlama dersleri deneyimi gibi değişkenler ile olan ilişkilerini incelemiştir. Öğrencilerin bilgisayar kullanma deneyimi (yıl olarak)

ve önceki programlama dersi deneyimleri ile programlama öz yeterlilikleri arasında anlamlı bir ilişki bulunmuştur.

Özyurt ve Özyurt [9] çalışmalarında bilgisayar programcılığı bölümünde okuyan öğrencilerin programlamaya karşı tutumlarını ve programlama öz yeterliliklerini cinsiyet, sınıf düzeyi ve öğrenim türü değişkenlerine göre incelemiş ve hepsinde anlamlı farklılıklar bulmuştur. Elde edilen sonuçlarda erkek öğrencilerin programlama öz yeterliliklerinin kadın öğrencilerden daha yüksek olduğu görülmüştür. Sınıf düzeyinde 2.sınıfta okuyan öğrencilerin programlama öz yeterlilikleri 1.sınıfta okuyan öğrencilerden daha yüksek bulunmuştur. Öğrenim türüne göre yapılan incelemelerde ise uzaktan eğitim alan öğrencilerin programlama öz yeterliliklerinin I. Öğretim ve II. Öğretim öğrencilerinden daha yüksek olduğu belirtilmiştir.

Üst paragraflarda anlatılan çalışmalarda kullanılan ölçeklerde yer alan sorular öğrencilerin zorluk seviyesi gittikçe artan programlama problemlerini çözme öz yeterliliklerini derecelendirmektedir. Tek ve arkadaşları [10] bu ölçeklerden farklı olarak genel öz yeterlilik ölçeğinden uyarlanan bir ölçek oluşturmuştur. Çalışmalarında programlama yeteneğinin geliştirilebileceğine inanan öğrencilerin sabit programlama yeteneğine inanan gruptan daha yüksek programlama öz yeterliliğine sahip olduğu gösterilmiştir. Programlama dersinden geçen ve başarısız olan öğrencilerin programlama öz yeterlilikleri arasında önemli ölçüde fark olduğu belirtilmiştir. Ayrıca programlama dersini ilk kez alan ve tekrarlayan öğrencilerin programlama öz yeterliliklerinin farklılaştığı görülmüştür.

Tsai ve arkadaşları [11] çalışmalarında mantıksal düşünme, işbirliği, algoritma, kontrol ve hata ayıklama alt ölçeklerinden meydana gelen yeni bir programlama öz yeterlilik ölçeği geliştirmişlerdir. Farklı bölüm ve programlama deneyimine sahip öğrencilerle yapılan çalışma sonucunda bilgisayar programlama öz yeterliliği ile programlama deneyimi arasında pozitif bir ilişki tespit edilmiştir. Ayrıca düşük ve orta programlama deneyimine sahip öğrenciler için algoritma ve hata ayıklama alt ölçeklerinde önemli cinsiyet farklılıkları tespit edilmiştir.

Kittur [12] çalışmasında literatürdeki ölçeklerden yararlanarak herhangi bir programlama dilinden bağımsız bir ölçek oluşturmuş ve bu ölçek ile Hindistan'daki özel bir üniversitede Elektrik-Elektronik bölümünde okuyan öğrencilerin programlama öz yeterlilik algılarının sınıf düzeyi, cinsiyet, sosyal çevre ve programlama deneyimi gibi değişkenler ile olan ilişkilerini incelemiştir. Ölçek temel programlama görevleri ve karmaşık programlama görevleri olmak üzere iki faktörden oluşmaktadır. Temel programlama görevleri ile değişkenler arasında istatistiksel anlam bulunmazken öğrencilerin sınıf düzeyleri ve önceki programlama deneyimleri ile karmaşık programlama görevleri arasında istatistiksel anlam bulunmuştur.

Özyurt ve Özyurt'un [9] da çalışmalarında bahsettiği gibi programlama öz yeterliliğin ölçülmesi ile ilgili olarak sınırlı sayıda yapılmış çalışma vardır, bu çalışmalardan bazıları Tablo 1'de özetlenmiştir.

*Tablo 1. Programlama öz yeterliliği ile farklı değişkenlerin ilişkisini inceleyen çalışmaların özeti*

<b>Çalışma</b>	<b>Programlama Dili</b>	<b>Değişken</b>	<b>Sonuç</b>
Ramalingam ve Wiedenbeck [5]	C++	Cinsiyet	Cinsiyete göre farklılaşmamaktadır.
Askar ve Davenport [6]	Java	Cinsiyet, bölüm, bilgisayar becerileri, bilgisayar deneyimi, bilgisayar kullanım sıklığı ve aile bireylerinin bilgisayar kullanımı	Cinsiyet, bölüm, bilgisayar deneyimi, bilgisayar kullanım sıklığına göre farklılaşmaktadır.

*Tablo 1.(devam) Programlama öz yeterliliği ile farklı değişkenlerin ilişkisini inceleyen çalışmaların özeti*

<b>Çalışma</b>	<b>Programlama Dili</b>	<b>Değişken</b>	<b>Sonuç</b>
Altun ve Mazman [1]	Programlama dilinden bağımsız	Cinsiyet, sınıf, bölüm, programlama deneyim yılı ve programlamaya ilişkin alınan ders sayısı	Cinsiyet dışındaki tüm değişkenlere göre farklılaşmaktadır.
Korkmaz ve Altun [8]	C++	Cinsiyet, bölüm	Bölüme göre farklılaşmaktadır.
Çiğdem ve Yıldırım [7]	C#	Yaş, mezun olunan lise türü, bilgisayar kullanma deneyimi, bilgisayar kullanım sıklığı ve programlama dersleri deneyimi	Bilgisayar kullanma deneyimi ve önceki programlama dersi deneyimine göre farklılaşmaktadır.
Özyurt ve Özyurt [9]	Programlama dilinden bağımsız	Cinsiyet, sınıf düzeyi ve öğrenim türü	Tüm değişkenlere göre farklılaşmaktadır.
Tek ve arkadaşları [10]	Java	Programlama zihniyeti, dersten geçme/kalma, dersi tekrarlama	Tüm değişkenlere göre farklılaşmaktadır.
Tsai ve arkadaşları [11]	Programlama dilinden bağımsız	Programlama deneyimi, cinsiyet	Programlama deneyimine göre farklılaşmaktadır. Cinsiyet alt ölçekte farklılaşmaktadır.
Kittur [12]	Programlama dilinden bağımsız	Sınıf düzeyi, cinsiyet, sosyal çevre, programlama deneyimi	Sınıf düzeyi, programlama deneyimi alt ölçekte farklılaşmaktadır.
Mevcut çalışma	Java	Cinsiyet, bölüm, eğitim dili, programlama öğrenme isteği, çalışırken çözümlü soruları tercih etme durumu	Bölüm ve programlama öğrenme isteğine göre farklılaşmaktadır.

Programlamaya giriş dersini alan öğrencilerin programlama öz yeterlilik algılarının, programlama öğrenme isteklerinin ve programlama dersine çalışma alışkanlıklarının araştırıldığı bu çalışmada aşağıdaki sorulara cevap aranmıştır.

- Öğrencilerin programlama öz yeterlilik algıları; cinsiyete, eğitim gördüğü bölüme, eğitim diline, programlama öğrenme isteğine, çalışırken çözümlü soruları tercih etme durumuna göre anlamlı farklılık göstermekte midir?
- Öğrencilerin programlama öğrenme isteği ile harf notları, cinsiyetleri, ders tekrarları arasında ilişki var mıdır?
- Öğrencilerin programlama çalışırken bilgisayar kullanma sıklıkları ile bölümleri, cinsiyetleri arasında ilişki var mıdır?

Ayrıca ölçme aracında yer alan sorular kullanılarak Apriori algoritması ile öğrencilerin programlama öz yeterlilik algıları, öğrenme istekleri ve çalışma alışkanlıkları ile ilgili birliktelik kuralları araştırılmıştır.

## **II. YÖNTEM**

Bu araştırmada yer alan veriler 2016-2017 yıllarında FMV Işık Üniversitesi Mühendislik Fakültesi ve Fen-Edebiyat Fakültesinde öğrenim gören ve Programlamaya Giriş dersini alan öğrencilerden toplanmıştır. Programlamaya Giriş dersi bütün bölümlerde 3 saat teorik anlatım ve 2 saat laboratuvarında uygulama geliştirme olmak üzere haftada 5 saat olacak şekilde işlenmektedir. Ders kapsamında öğrencilere Java dili ile temel programlama yapıları (değişkenler, seçim yapıları, döngü kalıpları, metotlar, diziler) gösterilirken nesne yönelimli yapılar (kurucu, sınıf, nesne kavramları) gösterilmemektedir. Bu ders ile öğrencilere 14 haftalık dönem sonunda temel programlama disiplininin kazandırılması amaçlanmaktadır.

### **A. ÇALIŞMA GRUBU**

Programlamaya Giriş dersini alan öğrenciler okudukları bölüme ve eğitim dillerine göre üç farklı şekilde gruplandırılmıştır. Bilgisayar Mühendisliği ve Yazılım Mühendisliği bölümü öğrencilerine verilen Programlamaya Giriş dersinin kodu CSE111 olup eğitim dili İngilizcedir. Bu bölümler dışında kalan, Mühendislik Fakültesi veya Fen Edebiyat Fakültesinde öğrenim gören öğrencilere verilen Programlamaya Giriş dersinin İngilizce eğitim dilindeki kodu CSE101, Türkçe eğitim dilindeki kodu ise CSE101T'dir.

Tablo 2'de görüldüğü üzere çalışmaya katılanların 62'si (%32.5) kadın, 129'u (%67,5) erkektir. 158 kişi (%82.7) İngilizce dilinde eğitim görürken 33 kişi (%17.3) Türkçe dilinde eğitim görmektedir. Katılımcıların 29'u (%15.2) Bilgisayar Mühendisliği veya Yazılım Mühendisliği bölümünde öğrenim görürken, 162'si (%84.8) Mühendislik Fakültesi (Endüstri, Makine, Elektronik vb.) veya Fen Edebiyat Fakültesinde (Yönetim Bilişim, Matematik Müh. vb.) bulunan bölümlerde eğitim görmektedirler. Bu bilgilere ek olarak toplanan verilere göre, çalışmaya katılan öğrencilerin 121'i (%63.4) programlamaya giriş dersini ilk defa alırken, 70'i (%36.6) bu dersi tekrar almaktadır. Dönem sonunda ise çalışmaya katılan öğrencilerin 152'si (%79.6) dersi geçerken, 39'u (%20.4) dersten başarısız olmuştur.

*Tablo 2. Öğrencilerin demografik bilgileri*

Gruplar	Ders Kodu	Dil	Bölüm	Cinsiyet		
				Kadın	Erkek	Toplam
G1	CSE101	İngilizce	Müh./Fen Edebiyat	40	89	129
G2	CSE101T	Türkçe	Müh./Fen Edebiyat	13	20	33
G3	CSE111	İngilizce	Bilgisayar/Yazılım	9	20	29

### **B. VERİ TOPLAMA ARACI**

Araştırmada veriler dört bölümden oluşan bir ölçme aracı ile toplanmıştır. Ölçme aracının birinci bölümünü öğrencilerin demografik özellikleri ile ilgili olan sorular oluşturmaktadır. Öğrencilerin ders çalışma alışkanlıkları ile ilgili olan sorular ikinci bölümde, programlama ile olan etkileşimleri ile ilgili olan sorular üçüncü bölümde bulunmaktadır. Dördüncü bölümde ise programlama özyeterlilik ölçeğinde yer alan sorular bulunmaktadır [13]. Araştırma verileri çevrimiçi olarak toplanmıştır.

Çalışmaya katılan öğrencilerin cinsiyet, bölüm, eğitim dili, programlamaya giriş dersinden aldıkları dönem sonu harf notları, dersi tekrarlama bilgileri demografik özellikler başlığı altında toplanmıştır. Öğrencilerin ders çalışma alışkanlıklarının ve programlama ile olan etkileşimlerinin incelendiği bölümlerde yer alan sorular literatürdeki benzer çalışmalardan yararlanılarak [14-18] araştırmacılar tarafından oluşturulmuş ve konuyla ilgili akademik çalışmaları bulunan uzmanların görüş ve önerileri alınmıştır. Bu bölümlerde yer alan sorulara Tablo 10 ve Tablo 11'de yer verilmiştir.

Bu çalışmada verinin toplandığı Programlamaya Giriş dersinde nesne yönelimli yapılar işlenmediği için Tek ve arkadaşları [10] tarafından geliştirilen programlama öz yeterlilik ölçeği kullanılmıştır. Ölçeğin

cronbach- $\alpha$  güvenilirlik katsayısı arařtırmacı tarafından 0.86 olarak belirtilmiřtir. Bu katsayıya gre lek yksek derecede gvenilirdir [19]. Programlama z yeterlilik leđinde đrencilere olumlu (“Programlamayı đrenebileceđime eminim.”) ve olumsuz (“Yeni bir programlama problemini zmeye alıřırken bařlangıta bařarılı olamazsam vazgeerim.”) sorular sorulmuř ve cevapları iin 5 seenekli likert tipi lek (1-Hi Tanımlamıyor, 2-ok az tanımlıyor, 3-Kısmen tanımlıyor, 4-Byk lde tanımlıyor, 5-ok iyi tanımlıyor) kullanılmıřtır. đrencilerin puanları olumlu ifade ieren sorular iin 1,2,3,4,5 řeklinde, olumsuz ifade ieren sorular iin ise 5,4,3,2,1 řeklinde hesaplanmıřtır.

## C. VERİLERİN ANALİZİ

alıřmada toplanan verilerin istatistiksel analizlerinde parametrik ve parametrik olmayan testlerden hangilerinin uygulanacađını belirlemek iin verilere normallik testleri yapılmıřtır. Programlama z yeterlilik ile ilgili analizlerde verilerin normal dađıldıđı cinsiyet deđiřkeni iin bađımsız iki rneklem T-testi yapılırken, rneklem sayısının uygun olmadıđı blm deđiřkeni iin ve verilerin normal dađılmadıđı eđitim dili deđiřkeni iin Mann Whitney U-testi kullanılmıřtır. İki den fazla gruptan oluřan programlama đrenme isteđi deđiřkeninde normallik varsayımı karřılanmadıđı iin analizlerde Kruskal Wallis H testi kullanılmıřtır. zml soruları tercih etme deđiřkenindeki beř grubun verileri normal dađılım gsterdiđi iin programlama z yeterlilik ile olan iliřkisi tek ynl varyans analizi ile incelenmiřtir.

Programlama đrenme isteđi ile dnem sonu harf notu, cinsiyet ve ders tekrarları arasındaki iliřkiler ve programlama alıřırken bilgisayar kullanma sıklıkları ile blm ve cinsiyet deđiřkenleri arasındaki iliřkiler incelenirken veriler nominal lekli olduđu iin Ki-Kare testi uygulanmıřtır. alıřmanın istatistiksel analiz ařamasında IBM SPSS programı kullanılırken [20], birliktelik kuralları oluřturulması ařamasında Apriori algoritması kullanılmıřtır [21]. Verilerin analiz ařamalarında kullanılan testler ve birliktelik kuralları hakkında detaylı bilgilere ařađıdaki alt blmlerde yer verilmiřtir.

### C.1. İstatistiksel Analiz

#### C.1.1. Bađımsız İki rneklem T-Testi

Bađımsız iki rneklem T-testi iki farklı grubun ortalamalarını karřılařtırmak iin kullanılır. Grupların normal dađılım gsterdiđi ve varyansların homojen olduđu varsayımlarından sonra Denklem 1 [22] kullanılarak T-test istatistiđi hesaplanır. Formlde yer alan  $N_1$  ve  $N_2$  deđiřkenleri gruplarda bulunan gzlem sayısını,  $\bar{X}_1$  ve  $\bar{X}_2$  deđiřkenleri grupların ortalamalarını,  $S_1^2$  ve  $S_2^2$  deđiřkenleri ise grupların varyanslarını gstermektedir.

$$t = \frac{(\bar{X}_1 - \bar{X}_2)}{\sqrt{\frac{(N_1-1)S_1^2 + (N_2-1)S_2^2}{(N_1+N_2-2)} \cdot \left(\frac{1}{N_1} + \frac{1}{N_2}\right)}} \quad (1)$$

T-test istatistiđi ile karřılařtırılacak olan kritik deđerin bulunması iin serbestlik derecesine ihtiya duyulur. Serbestlik derecesi  $N_1$  ve  $N_2$  deđerleri toplamından iki sayısının ıkartılması ile bulunur ( $N_1+N_2-2$ ). Kritik deđer, anlamlılık dzeyi ve serbestlik derecesi kullanılarak hazır t tablosu yardımı ile belirlenir. Daha sonra bu deđer T-test istatistiđi ile karřılařtırılır ve hipotezlerden hangisinin kabul edileceđine karar verilir.

#### C.1.2. Tek Ynl Varyans Analizi

Tek ynl varyans analizi bađımsız ikiden fazla grubun ortalamalarını karřılařtırmak iin kullanılır. Grupların normal dađılım gsterdiđi ve varyansların homojen olduđu varsayımlarında bulunulur. Bađımsız iki rneklem T-testinin ikiden fazla grup iin uygulanan halidir. F-test istatistiđi Tablo 3’te [23] belirtilen formller kullanılarak hesaplanır. Formlde bulunan k deđiřkeni grup sayısını,  $N_k$  deđiřkeni k. grupta bulunan gzlem sayısını,  $X_{ij}$  deđiřkeni i. gruptaki j. gzlemi gstermektedir.

Tablo 3. Varyans analiz tablosu

Varyasyon Kaynağı	Kareler Toplamı	SD	Karelerin Ort.	F Değeri
Gruplar Arası	$GAKT = \sum_{i=1}^k \left( \sum_{j=1}^{N_k} X_{ij} \right)^2 - \frac{\left( \sum_{i=1}^k \sum_{j=1}^{N_k} X_{ij} \right)^2}{N}$	k-1	GAKO= GAKT/k-1	GAKO/ GİKO
Gruplar İçi	$GİKT = \sum_{i=1}^k \sum_{j=1}^{N_k} X_{ij}^2 - \left[ \sum_{i=1}^k \left( \sum_{j=1}^{N_k} X_{ij} \right)^2 - \frac{\left( \sum_{i=1}^k \sum_{j=1}^{N_k} X_{ij} \right)^2}{N} \right]$	N-k	GİKO= GİKT/N-k	

\*SD=serbestlik derecesi, Ort.=ortalama, GAKT= Gruplar arası kareler toplamı, GİKT= Grup içi kareler toplamı, GAKO= Gruplar arası kareler ortalaması, GİKO= Grup içi kareler ortalaması

F-test istatistiği ile karşılaştırılacak olan kritik değerin bulunması için iki tane serbestlik derecesine ihtiyaç vardır. İlk serbestlik derecesi gruplar arası serbestlik derecesidir ve değeri grup sayısının bir eksiğidir (k-1). İkinci serbestlik derecesi ise grup içi serbestlik derecesidir ve değeri toplam gözlem sayısından grup sayısının çıkartılması ile bulunur (N-k). Kritik değer, anlamlılık düzeyi ve serbestlik dereceleri kullanılarak F tablosu yardımı ile belirlenir. Daha sonra bu değer F-test istatistiği ile karşılaştırılır ve hipotezlerden hangisinin kabul edileceğine karar verilir.

### C.1.3. Mann Whitney U-Testi

Mann Whitney U-testi, bağımsız örneklem T-testi için istenilen varsayımların karşılanmadığı durumlarda kullanılan parametrik olmayan bir testtir. İki farklı grubun medyan değerlerini karşılaştırır. U-testi hesaplaması sırasında ilk olarak  $N_1$  ve  $N_2$  hacmine sahip iki grubun verileri tek bir dizide birleştirilir ve gözlemler küçükten büyüğe doğru sıralanır. Daha sonra her bir gözleme bir sıra numarası atanır. İki gruba ait sıra toplamları,  $R_1$  ve  $R_2$ , ayrı ayrı hesaplanır (Denklem 2) [23].

$$R_1 = \sum_{i=1}^{N_1} X_{1i} \quad ve \quad R_2 = \sum_{i=1}^{N_2} X_{2i} \quad (2)$$

Sıra toplamları arasındaki farkları incelemek için Denklem 3 [24] kullanılır:

$$U_1 = N_1 N_2 + \frac{N_1(N_1 + 1)}{2} - R_1 \quad ve \quad U_2 = N_1 N_2 + \frac{N_2(N_2 + 1)}{2} - R_2 \quad (3)$$

$U_1$  ve  $U_2$  değerlerinden küçük olan U-test istatistiği olarak belirlenir.  $N_1$  ve  $N_2$  değerlerinin 8 ve daha büyük olduğu durumda U normal dağılıma yaklaşır [24]. U-test istatistiği Denklem 4 yardımıyla z-test istatistiğine dönüştürülür.

$$z = \frac{U - \mu_U}{\sigma_U}, \quad \mu_U = \frac{N_1 N_2}{2} \quad ve \quad \sigma_U^2 = \frac{N_1 N_2 (N_1 + N_2 + 1)}{12} \quad (4)$$

Hesaplanan z-test istatistiğinin mutlak değeri alınır ve belirlenen anlamlılık düzeyindeki kritik değer ile karşılaştırılıp hipotezlerden hangisinin kabul edileceğine karar verilir.

### C.1.4. Kruskal Wallis H Testi

Kruskal Wallis H testi, tek yönlü varyans analizinin varsayımlarının karşılanmadığı durumlarda kullanılan parametrik olmayan bir test olup Mann Whitney-U testinin ikiden fazla grup için uygulanan halidir. H testi hesaplamaları sırasında U-testinde olduğu gibi bütün gruplar tek bir dizide birleştirilir, gözlemler artan şekilde sıralanır ve her bir gözleme bir sıra numarası verilir. Bu işlemlerden sonra H test istatistiği Denklem 5 [24] yardımıyla hesaplanır.

$$H = \frac{12}{N(N+1)} \sum_{j=1}^k \frac{R_j^2}{N_j} - 3(N+1) \quad (5)$$

Denklemde yer alan k değişkeni toplam grup sayısını, N değişkeni bütün gruplarda yer alan gözlem sayılarının toplamı, N<sub>j</sub> değişkeni j.grupta bulunan gözlemlerin sayısını, R<sub>j</sub> değişkeni ise j. gruba ait gözlemlerin sıra numaraları toplamını göstermektedir. H test istatistiği ile karşılaştırılacak olan kritik değerin bulunması için gerekli olan serbestlik derecesinin değeri k-1'dir. Kritik değer, anlamlılık düzeyi ve serbestlik derecesi kullanılarak hazır Ki-Kare tablosu yardımı ile belirlenir. Daha sonra bu değer H test istatistiği ile karşılaştırılır ve hipotezlerden hangisinin kabul edileceğine karar verilir.

### C.1.5. Ki-Kare Testi

Ki-Kare testi kategorik değişkenler arasındaki ilişkileri incelemek için kullanılır. İki kategoriye sahip iki değişkenin bağımsızlık analizleri yapılırken 2x2 çaprazlık tablosu kullanılır. Bu tablonun her bir hücresine değişkenin o kategorideki gözlem sayısı yerleştirilir ve Denklem 6 [23] yardımıyla Ki-Kare değeri hesaplanır.

$$\chi^2 = \sum \frac{(Gözlenen_{ij} - Beklenen_{ij})^2}{Beklenen_{ij}}, \quad Beklenen_{ij} = \frac{\text{satır}_i \text{sütun}_j}{N} \quad (6)$$

Beklenen değerler çaprazlık tablosunun i.satır toplamı ve j.sütun toplamı çarpımının toplam gözlem sayısına bölünmesiyle elde edilir. Kritik değeri bulmada kullanacağımız serbestlik derecesi ise satır ve sütun sayılarından bir çıkartılıp bu değerlerin birbirleriyle çarpılması sonucunda elde edilir ((satır-1) x (sütun-1)). Anlamlılık düzeyi ve serbestlik derecesi ile Ki-Kare tablosu kullanılarak kritik değer bulunur. Ki-Kare test istatistiği ile kritik değer karşılaştırılıp hipotezlerden hangisinin kabul edileceğine karar verilir.

## C. 2. Birliktelik Kuralı

Birliktelik kuralları büyük veri setinden anlamlı ve yararlı ilişkileri çıkarmak için kullanılan veri madenciliği yöntemlerinden biridir [25]. Yaygın olarak pazar sepet analizlerinde müşterilerin satın alma alışkanlıklarının bulunmasında kullanılmakla beraber tıp, eğitim, finans gibi farklı alanlardaki problemlere de uygulanmaktadır. Birliktelik kurallarında, birlikte sıklıkla tekrarlanan öge kümeleri arasındaki ilişki  $X \Rightarrow Y$  şeklinde gösterilir ve beraberlerinde destek ve güven değerlerini de içerirler.

Destek ve güven değerleri hesaplanırken destek sayısı adı verilen bir değer kullanılır. Destek sayısı ( $\sigma$ ) belirli bir öge kümesini içeren işlemlerin sayısıdır. Destek değeri X ve Y ögelerinin birlikte bulunduğu işlem sayısının toplam işlem sayısına (N) oranı ile bulunur (Denklem 7) [26].

$$\text{Destek}(X \Rightarrow Y) = \frac{\sigma(X \cup Y)}{N} \quad (7)$$

Güven değeri ise X ve Y ögelerinin birlikte bulunduğu işlem sayısının X'i içeren işlem sayısına oranlanması ile hesaplanır (Denklem 8) [26]. Güven değerinin 100 olması, X ögesinin bulunduğu bütün kayıtlarda Y ögesinin de bulunduğu anlamına gelir.



$$\text{Güven}(X \Rightarrow Y) = \frac{\sigma(X \cup Y)}{\sigma(X)} \quad (8)$$

Elde edilen kuralların ilginçlik derecesinin tespitinde ise Lift değeri kullanılmaktadır (Denklem 9) [26]. Lift değerinin 1'den büyük olduğu durumlarda X ve Y arasında pozitif korelasyon, 1'den küçük olduğu durumlarda ise negatif korelasyon vardır, 1 değerine eşit olduğunda ise X ve Y birbirlerinden bağımsızdır [27].

$$\text{Lift}(X \Rightarrow Y) = \frac{\text{Güven}(X \Rightarrow Y)}{\text{Destek}(Y)} \quad (9)$$

Birliktelik kuralları çıkarımında en fazla bilinen ve kullanılan algoritma Agrawal ve Srikant tarafından geliştirilen Apriori Algoritmasıdır [21]. Bu algoritma uygulanmadan önce destek ve güven ölçütleri için eşik değerler belirlenir. İlk adımda öge kümesindeki her bir ögenin destek değeri hesaplanır ve destek eşik değeri ile karşılaştırılır. Eğer bulunan değer eşik değerden küçükse bu öge daha sonraki hesaplamalarda yer almaz. İkinci aşamada, ilk aşamada elde edilen yaygın tekli ögeler kullanılarak ikiyeşli öge kümeleri oluşturulur. Bu kümelerin de destek değerleri hesaplanır, eşik değerden büyük olanlar bir sonraki aşamada yeni kümelerin oluşturulması aşamasında yer alırlar. Bu süreç yeni kümeler oluşmayana kadar yani eşik değerinin altında kalmayana kadar devam eder. Daha sonra oluşturulan kuralların her biri için güven değerleri hesaplanır.

### **III. BULGULAR VE TARTIŞMA**

#### **A. PROGRAMLAMA ÖZ YETERLİLİK BULGULARI**

Öğrencilerin programlama öz yeterlilik algıları çeşitli değişkenler açısından incelenirken bölüm değişkeni dışındaki tüm analizlerde CSE101 ve CSE101T derslerinin verileri kullanılmış, bölüm değişkeni ile yapılan analizde ise Tablo 2'de belirtilen bütün grupların verileri kullanılmıştır.

Bu bölümdeki birinci alt problem “*Programlamaya giriş dersini alan öğrencilerin programlama öz yeterlilik algılarında cinsiyetlerine göre anlamlı bir farklılık var mıdır?*” olarak belirlenmiştir. Tablo 4 incelendiğinde cinsiyet değişkenine göre öğrencilerin programla öz yeterlilik puanlarında anlamlı fark bulunmamıştır ( $t(160)=1.398, p>.05$ ).

**Tablo 4.** Öğrencilerin programlama öz yeterlilik puanlarının cinsiyet değişkenine göre T-testi sonuçları

	Cinsiyet	N	Ortalama	Standart Sapma	sd	t	p
<b>Programlama</b>	Kadın	53	22,19	4,494	160	1,398	0,164
<b>Öz Yeterlilik</b>	Erkek	109	21,08	4,833			

\*N=örneklem sayısı, sd=serbestlik derecesi

Bu bölümün ikinci alt problemi “*Programlamaya giriş dersini alan öğrencilerin programlama öz yeterlilik algılarında eğitim görülen bölüme göre anlamlı bir farklılık var mıdır?*” olarak belirlenmiştir. Tablo 5'teki sonuçlara göre Bilgisayar/Yazılım Mühendisliği öğrencileri ile programlamaya giriş dersini (Türkçe/İngilizce) alan diğer bölümlerin öğrencilerinin programlama öz yeterlilikleri arasında anlamlı bir farklılık vardır ( $U=1714.50, Z=2.320, p=.02$ ).

191 öğrencinin programlama öz yeterlilik puanları küçükten büyüğe doğru sıralandığında, 29 Bilgisayar/Yazılım Mühendisliği bölümü öğrencilerine ilişkin sıralamaların ortalaması 117,88 olup bu sıralamaların toplamı 3418,50'dir. Diğer bölümlerde okuyan öğrencilerin programlama öz yeterlilik puanlarına karşılık gelen sıra puanlarının ortalaması 92,08 olup, toplamı ise 14917,50'dir. Sıra

ortalamlarına bakıldığında Bilgisayar/Yazılım Mühendisliği öğrencilerinin ortalaması diğer bölümlerin öğrencilerinden daha yüksektir. Başka bir deyişle Bilgisayar/Yazılım mühendisliği öğrencilerinin programlama öz yeterlilik puanları diğer bölümlerin öğrencilerinden yüksektir.

**Tablo 5.** Öğrencilerin programlama öz yeterlilik puanlarının bölüm değişkenine göre U-testi sonuçları

	Bölüm	N	Sıra Ortalaması	Sıra Toplamı	U değeri	p değeri
<b>Programlama Öz Yeterlilik</b>	Bilgisayar /Yazılım	29	117,88	3418,50	1714,500	0,020
	Diğer	162	92,08	14917,50		

\*N=örneklem sayısı

Bu bölümün üçüncü alt problemi “Programlamaya giriş dersini alan öğrencilerin programlama öz yeterlilik algularında eğitim diline göre anlamlı bir farklılık var mıdır?” olarak belirlenmiştir. Tablo 6 incelendiğinde eğitim dili değişkenine göre öğrencilerin programlama öz yeterlilik puanlarında anlamlı bir fark bulunmamıştır (U=2063.00, Z=0.273, p=.785). İki grubun sıra ortalamaları birbirine yakın olmakla birlikte, Türkçe dilinde eğitim alan öğrencilerin programlama öz yeterlilik ortalamalarının daha yüksek olduğu görülmüştür.

**Tablo 6.** Öğrencilerin programlama öz yeterlilik puanlarının eğitim dili değişkenine göre U-testi sonuçları

	Eğitim Dili	N	Sıra Ortalaması	Sıra Toplamı	U değeri	p değeri
<b>Programlama Öz Yeterlilik</b>	İngilizce	129	80,99	10448,00	2063,000	0,785
	Türkçe	33	83,48	2755,00		

\*N=örneklem sayısı

Bu bölümün dördüncü alt problemi “Programlamaya giriş dersini alan öğrencilerin programlama öz yeterlilik algularında programlama öğrenme isteklerine göre anlamlı bir farklılık var mıdır?” olarak belirlenmiştir. Tablo 7 incelendiğinde öğrencilerin programlama öz yeterlilik değerlerinin beş farklı öğrenme isteği grubunda farklılık gösterdiği görülmektedir ( $\chi^2(4, N=162)=24.896, p<.05$ ).

Grupların sıra ortalamaları incelendiğinde öğrenme isteğini “Çok Fazla” şeklinde ifade eden grubun programlama öz yeterlilik algularının diğer gruplara göre daha yüksek olduğu görülmektedir. Çoklu karşılaştırmalar yapılarak hangi grupların birlerinden farklı olduğu incelendiğinde, “Hiç” ile “Çok Fazla” ve “Az” ile “Çok Fazla” programlama öğrenme isteği gruplarında farklılık olduğu gözlemlenmiştir.

**Tablo 7.** Öğrencilerin programlama öz yeterlilik puanlarının programlama öğrenme isteğine göre Kruskal Wallis H testi sonuçları

	Programlama Öğrenme İsteği	N	Sıra Ortalaması	SD	$\chi^2$	p
<b>Programlama Öz Yeterlilik</b>	Hiç	12	54,08	4	24,896	,000
	Az	22	51,16			
	Orta	30	74,73			
	Fazla	39	81,83			
	Çok Fazla	59	101,61			

\*N=örneklem sayısı, SD=serbestlik derecesi,  $\chi^2$ =Ki-Kare

Dördüncü alt probleme ek olarak “Programlamaya giriş dersini alan Bilgisayar/Yazılım Mühendisliği öğrencilerinin programlama öz yeterlilik algularında programlama öğrenme isteklerine göre anlamlı bir farklılık var mıdır?” sorusunun analizi yapılmıştır. Tablo 8’deki sonuçlar incelendiğinde öğrencilerin programlama öz yeterlilik puanları ile öğrenme isteği grupları arasında anlamlı bir fark bulunmamıştır

( $\chi^2(2, N=29)=2.964, p>.05$ ). Bu durum Bilgisayar/Yazılım Mühendisliğinde okuyan öğrencilerin programlama öz yeterlilik algılarının yüksek olması ile açıklanabilir.

**Tablo 8.** Bilgisayar/yazılım müh. öğrencilerinin programlama öz yeterlilik puanlarının programlama öğrenme isteğine göre Kruskal Wallis H testi sonuçları

	Programlama Öğrenme İsteği	N	Sıra Ortalaması	SD	$\chi^2$	p
<b>Programlama Öz Yeterlilik</b>	Hiç	0	0	2	2,964	,227
	Az	0	0			
	Orta	3	9,17			
	Fazla	4	11,13			
	Çok Fazla	22	16,50			

\* N=örneklem sayısı, SD=serbestlik derecesi,  $\chi^2$ =Ki-Kare

Bu bölümün beşinci alt problemi “Programlamaya giriş dersini alan öğrencilerin programlama öz yeterlilik algılarında çalışırken çözümlü soruları tercih etme durumuna göre anlamlı bir farklılık var mıdır?” olarak belirlenmiştir. Tablo 9 incelendiğinde öğrencilerin programlama öz yeterlilik değerlerinin beş farklı tercih grubunda farklılık göstermediği görülmektedir ( $F(4,157)=0.359, p>.05$ ).

**Tablo 9.** Öğrencilerin programlama öz yeterlilik puanlarının çözümlü soruları tercih etme durumuna tek yönlü varyans analiz sonuçları

		SD	Kareler Toplamı	Kareler Ortalaması	F	p
<b>Programlama Öz Yeterlilik</b>	Gruplar Arası	4	32,801	8,200	,359	,837
	Gruplar İçi	157	3583,199	22,823		
	Toplam	161	3626,000			

\*SD=serbestlik derecesi

## B. PROGRAMLAMA ÖĞRENME İSTEĞİ BULGULARI

Programlama öğrenme isteği ile ilgili yapılan analizlerde Bilgisayar/Yazılım Mühendisliği bölümü öğrencilerinin verileri kullanılmamıştır. Veri seti CSE101 ve CSE101T derslerine kayıtlı öğrencilerden oluşmaktadır.

Bu bölümdeki birinci alt problem “Programlamaya giriş dersini alan öğrencilerin programlama öğrenme isteği ile harf notları arasında ilişki var mıdır?” olarak belirlenmiştir. Analiz sırasında CC ve üstü harf notunu alan öğrenciler başarılı, CC harf notunun altındaki öğrenciler ise dersi tekrar edilmesi önerilenler olmak üzere iki gruba ayrılmıştır. Örneklem uygulanan Ki-Kare test sonuçlarına göre öğrencilerin programlama öğrenme istekleri ile harf notları arasında anlamlı bir ilişki bulunmamıştır ( $\chi^2(4, N=162)=7.654, p=.105$ ).

Bu bölümün ikinci alt problemi “Programlamaya giriş dersini alan öğrencilerin programlama öğrenme isteği ile cinsiyetleri arasında ilişki var mıdır?” olarak belirlenmiştir. Analiz sırasında çaprazlık tablosunda 1’den küçük frekans değerine sahip olan bir hücre bulunmuştur (kadın öğrencilerden hiçbiri programlama öğrenmeyi ne kadar istiyorsunuz sorusuna “Hiç” yanıtını vermemiştir). Ki-Kare testinin varsayımını gerçekleyebilmek için programlama öğrenme isteği verileri 3 farklı grup (az istekli, orta istekli, çok istekli) olacak şekilde yeniden düzenlenmiştir [28]. Tekrarlanan test sonuçlarında cinsiyet ile programlama öğrenme isteği arasında anlamlı bir ilişki bulunmamıştır ( $\chi^2(2, N=162)=5.218, p=.074$ ).

Bu bölümün üçüncü alt problemi “Programlamaya giriş dersini alan öğrencilerin programlama öğrenme isteği ile ders tekrarları arasında ilişki var mıdır?” olarak belirlenmiştir. Test sonuçlarına göre dersi ilk kez alan ve tekrarlayan öğrencilerin programlama öğrenme istekleri arasında anlamlı bir ilişki bulunmamıştır ( $\chi^2(4, N=162)=4.333, p=.363$ ).

## C. PROGRAMLAMA ÇALIŞIRKEN BİLGİSAYAR KULLANMA SIKLIĞI

Bu bölümündeki birinci alt problem “Programlamaya giriş dersini alan öğrencilerin programlama çalışırken bilgisayar kullanma sıklıkları ile bölümleri arasında ilişki var mıdır?” olarak belirlenmiştir. Analiz sırasında Tablo 2’de yer alan bütün öğrencilerin verilerine Ki-Kare testi uygulanmıştır. Çaprazlık tablosu incelendiğinde 5’ten küçük frekans değerine sahip olan hücrelerin verinin %20’sinden büyük olduğu görülmüştür. Ki-Kare testinin varsayımını gerçekleyebilmek için bilgisayar kullanma sıklığı verileri 3 farklı grup (sık sık, bazen, seyrek kullanım) olacak şekilde yeniden düzenlenmiştir [28]. Yeni gruplamayla testler tekrarlanmış ve öğrencilerin programlamaya çalışırken bilgisayar kullanma sıklıkları ile eğitim gördükleri bölüm arasında ilişki bulunmamıştır ( $\chi^2(2, N=191)=1.514, p=.469$ ).

Bu bölümündeki ikinci alt problem “Programlamaya giriş dersini alan öğrencilerin programlama çalışırken bilgisayar kullanma sıklıkları ile cinsiyetleri arasında ilişki var mıdır?” olarak belirlenmiştir. Analiz sırasında Bilgisayar/Yazılım Mühendisliği dışındaki bölümlerde okuyan öğrencilerin verileri kullanılmıştır. Örneklemeye uygulanan Ki-Kare test sonuçlarına göre kadın ve erkek öğrencilerin programlama öğrenirken bilgisayar kullanım sıklıkları arasında bir ilişki bulunmamıştır ( $\chi^2(5, N=162)=3.353, p=.646$ ).

## D. DİĞER BULGULAR

Bu bölümde öğrencilerin ölçme aracında yer alan programlama ile olan etkileşimler ve ders çalışma alışkanlıkları bölümlerinde bulunan sorulara vermiş oldukları yanıtlar incelenmektedir. Tablo 10’da öğrencilerin gruplar bazında sorulan sorulara vermiş oldukları yanıtlar, CSE101 dersine kayıtlı öğrenciler için G1 satırı ile CSE101T dersine kayıtlı öğrenciler için G2 satırı ile CSE111 dersine kayıtlı öğrenciler için ise G3 satırı ile gösterilmektedir.

Tablo 10. Programlamaya ilişkin görüşlerin gruplara göre dağılımı

İfadeler	Gruplar	Kesinlikle	Katılıyor	Kısmen	Kısmen	Katılmıyor	Kesinlikle
		Katılıyor	Katılıyor	Katılıyor	Katılmıyor	Katılmıyor	Katılmıyor
		%	%	%	%	%	%
Programlamayı öğrenmek için İngilizce'ye hakim olmak gereklidir.	G1	19,4	22,5	21,7	13,2	15,5	7,7
	G2	15,1	27,3	39,4	12,1	6,1	0,0
	G3	20,7	44,8	20,7	3,5	10,3	0,0
İngilizce gerektirdiği için programlamayı öğrenmek daha zordur.	G1	8,5	10,1	32,6	9,3	24,0	15,5
	G2	6,1	15,1	27,3	18,2	24,2	9,1
	G3	17,2	10,4	17,2	13,8	24,2	17,2
Matematik altyapısı iyi olan kişiler iyi programcı olurlar.	G1	19,4	24,0	27,9	15,5	9,3	3,9
	G2	12,1	36,3	27,3	12,1	6,1	6,1
	G3	34,4	27,6	17,2	3,5	13,8	3,5

**Tablo 10. (devam) Programlamaya ilişkin görüşlerin gruplara göre dağılımı**

İfadeler	Gruplar	Kesinlikle	Katlıyorum	Katlıyorum	Kısmen	Katlıyorum	Kısmen	Katılmıyorum	Katılmıyorum	Kesinlikle	Katılmıyorum
		%	%	%	%	%	%	%			
Matematik gerektirdiği için programlamayı öğrenmek daha zordur.	G1	3,1	10,0	31,0	21,7	23,3	10,9				
	G2	9,1	6,1	39,4	15,1	24,2	6,1				
	G3	6,9	13,8	31,0	20,7	13,8	13,8				
Programlama bilgisi ileride iş hayatında benim işime yarayabilir.	G1	31,8	22,5	25,6	5,4	5,4	9,3				
	G2	36,4	27,3	15,1	12,1	6,1	3,0				
	G3	65,5	24,1	3,5	6,9	0,0	0,0				
Programlamayı öğrenmenin en iyi yolu mümkün olduğu kadar fazla program yazmaktır.	G1	32,6	39,5	22,5	3,1	0,8	1,5				
	G2	45,5	24,2	18,2	3,0	6,1	3,0				
	G3	51,7	31,0	10,3	3,5	0,0	3,5				
Derslerde/laboratuvarda yazılan programları izleyerek/not alarak programlamayı öğreniyorum.	G1	18,6	32,6	36,4	4,6	7,0	0,8				
	G2	21,2	42,5	33,3	0,0	0,0	3,0				
	G3	44,8	27,6	10,3	13,8	0,0	3,5				
Derslerde/laboratuvarda çözülen problemleri bilgisayarda kodlayarak programlamayı öğreniyorum.	G1	18,6	40,3	27,2	5,4	5,4	3,1				
	G2	18,2	39,4	24,2	9,1	6,1	3,0				
	G3	37,9	31,0	13,8	13,8	0,0	3,5				
Arkadaşlarımla beraber ikili veya küçük gruplar halinde bilgisayarda program yazarak programlamayı öğreniyorum.	G1	13,2	30,2	26,4	7,7	11,6	10,9				
	G2	18,2	21,2	27,3	12,1	15,1	6,1				
	G3	20,7	20,7	17,2	13,8	6,9	20,7				
Kendi başıma program yazarak programlamayı öğreniyorum.	G1	18,6	20,9	32,6	9,3	11,6	7,0				
	G2	21,2	30,3	33,3	6,1	9,1	0,0				
	G3	24,1	37,9	13,8	17,2	3,5	3,5				

\*G1= CSE101, G2= CSE101T, G3= CSE111

Tabloda yer alan *programlama öğrenmek için İngilizce'ye hakim olmak gerektiği* ifadesine Grup 1'de yer alan öğrencilerin %63,6'sı, Grup 2'de yer alan öğrencilerin %81,8'i ve Grup 3'te yer alan öğrencilerin %86,2'si katılmaktadır. Burada dikkat çekici olan sonuç Türkçe dilinde programlama dersi alan öğrencilerin yer aldığı Grup 2'deki ifadeye katılma oranının yüksek olmasıdır. Programlama dillerinin teknik dilinin İngilizce olması sebebiyle öğrenciler dersi Türkçe dilinde görseller bile kullanılan terimleri anlamak ve güncel kaynakları takip etmek için İngilizce dilini bilmeleri gerektiğini düşünmektedir. *İngilizce gerektirdiği için programlamayı öğrenmek daha zordur* ifadesine Grup 1'de yer alan öğrencilerin %51,2'si katılırken, Grup 2'de yer alan öğrencilerin %51,5'i ve Grup 3'te yer alan öğrencilerin %55,2'si katılmamaktadır. Bu ifadeye katılıp katılmama oranlarının birbirlerine çok yakın olduğu görülmektedir.

*Matematik altyapısı iyi olan kişiler iyi programcı olurlar* ifadesine bütün gruplar yüksek oranlarla (G1: 71,3; G2: 75,7; G3: 79,2) katılmaktadırlar. *Matematik gerektirdiği için programlamayı öğrenmenin zor olduğu* ifadesine Grup 1'de yer alan öğrencilerin %55,9'u katılmazken, Grup 2'de yer alan öğrencilerin %54,6'sı ve Grup 3'te yer alan öğrencilerin %51,7'si katılmaktadır. Bu ifadenin katılıp katılmama oranlarına bakıldığında cevaplarda bir çoğunluğun sağlanamadığı görülmüştür.

*Programlama bilgisi ileride iş hayatında benim işime yarayabilir* ifadesine Bilgisayar/Yazılım Mühendisliği dışındaki bölümlerde okuyan öğrencilerin (162 kişi) %79,63'ü katılmaktadır. Bu

öğrenciler asıl bölümleri farklı olmasına rağmen ders kapsamında öğrendikleri programlama bilgisinin iş hayatlarında kendilerine artı bir değer katacağı inancındadır.

Bütün gruplar, *Programlama öğrenmenin en iyi yolu mümkün olduğu kadar fazla program yazmaktır* ifadesine %85 üzerinde oranlarla (G1: 94,6; G2: 87,9; G3: 93) katılmaktadırlar. Programlama öğrenme yöntemlerini anlamak için sorulan sorularda öğrenciler, *derslerde/laboratuvarlarda yazılan programları izleyerek/not alarak* (G1: 87,6; G2: 97; G3: 82,7) ve *derslerde/laboratuvarlarda çözülen problemleri bilgisayarda kodlayarak* (G1: 86,1; G2: 81,8; G3: 82,7) programlamayı öğrendiklerini ifade etmişlerdir. Çalışmada yer alan bütün gruplar bu iki ifadeye %80'nin üstünde oranlarla katılmaktadırlar.

Öğrenciler, *arkadaşlarıyla beraber ikili veya küçük gruplar halinde bilgisayarda program yazarak* (G1: 69,8; G2: 66,7; G3: 58,6) veya *kendi başlarına program yazarak* programlamayı öğrendiklerini (G1: 72,1; G2: 84,8; G3: 75,8) belirtmişlerdir. Son ifadeye bütün gruplarda katılma oranı %70'in üstündedir.

Tablo 11'de öğrencilerin programlama öğrenirken kullanılan kaynaklar ile ilgili sorulara vermiş oldukları yanıtlar gruplar bazında gösterilmiştir.

**Tablo 11.** Programlama öğrenirken bilgisayar ve ders/laboratuvar notlarını kullanma sıklıklarının gruplara göre dağılımı

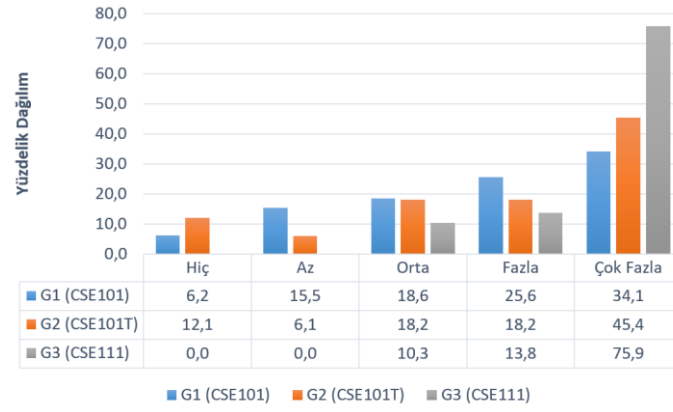
İfadeler	Gruplar	Gruplar					
		Hiç	Çok az	Az	Zaman zaman	Çoğu zaman	Her zaman
		%	%	%	%	%	%
Programlamayı öğrenirken bilgisayarı hangi sıklıkla kullanıyorsunuz.	G1	5,4	19,5	27,9	24,0	11,6	11,6
	G2	3,0	12,1	30,3	24,3	12,1	18,2
	G3	3,4	20,7	20,7	20,7	20,7	13,8
Programlamayı öğrenirken ders/laboratuvar notlarını hangi sıklıkla kullanıyorsunuz.	G1	3,9	4,7	23,2	15,5	23,2	29,5
	G2	3,0	3,0	27,3	24,2	15,2	27,3
	G3	6,9	17,2	20,7	27,6	10,4	17,2

\*G1= CSE101, G2= CSE101T, G3= CSE111

*Programlamayı öğrenirken hangi sıklıkla bilgisayar kullanıldığı* sorusuna Grup 1'deki öğrencilerin yaklaşık olarak %53'ü hiç, çok az, az seçeneklerinden biri ile cevap verirken geri kalan %47'si zaman zaman, çoğu zaman, her zaman seçeneklerinden biri ile cevap vermiştir. Bu durum Grup 2 ve Grup 3'te %45 (hiç, çok az, az) ve %55 (zaman zaman, çoğu zaman, her zaman) şeklinde oluşmuştur. Oranlar arasında çok fark olmadığı için öğrencilerin yarısı kadarının bilgisayarı kullanmadan programlamayı öğrenmeye çalıştıkları görülmektedir.

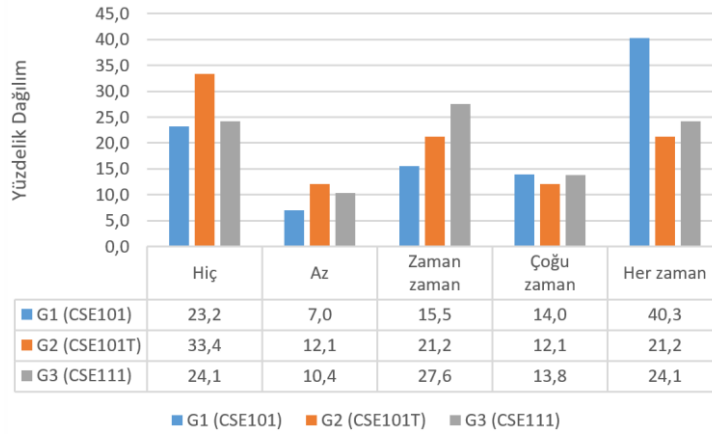
*Programlamayı öğrenirken ders/laboratuvar notlarını hangi sıklıkla kullanıyorsunuz* sorusuna zaman zaman, çoğu zaman, her zaman seçeneklerinden biri ile cevap verenlerin oranı Grup 1'de %68,2 olurken bu oran Grup 2'de %66,7 ve Grup 3'te %55,2 şeklindedir. Bilgisayar/Yazılım Mühendisliği bölümü öğrencilerinden oluşan Grup 3'teki oran diğer gruplara göre daha düşüktür.

Ankette yer alan *programlama öğrenmeyi ne kadar istiyorsunuz* sorusuna ders bazında öğrencilerin verdikleri cevaplar Şekil 1'de gösterilmiştir. Bilgisayar/Yazılım Mühendisliği öğrencilerinin bulunduğu Grup 3'ün "Çok Fazla" yanıtını verme oranı %75,9 olurken, geri kalan öğrencilerde aynı yanıtı verme oranı İngilizce dilinde eğitim gören Grup 1 için %34,1 ve Türkçe dilinde eğitim gören Grup 2 için %45,4'tür. Türkçe dilinde eğitim gören grubun "Çok Fazla" yanıtını verme oranı İngilizce grubundan daha yüksek olmakla birlikte "Çok Fazla" ve "Fazla" yanıtları birlikte incelendiğinde, Grup 1 ve Grup 2 için yakın oranların elde edildiği görülür (G1: %59,7; G2: %63,6). Bu soruya Bilgisayar/Yazılım Mühendisliği öğrencilerinden "Az" ve "Hiç" yanıtını veren olmamıştır.



**Şekil 1.**Ders bazında programlama öğrenme isteği dağılımı

*Programlama çalışırken çözümlü soruları çözümsüz sorulara tercih ederim* sorusuna ders bazında öğrencilerin verdikleri cevaplar Şekil 2’de gösterilmiştir. Bu soruya en yüksek oranla “hiç” cevabını veren grup Türkçe dilinde eğitim görmektedir. Bölümü Bilgisayar/Yazılım Mühendisliği olmayıp İngilizce dilinde eğitim gören öğrencilerin büyük çoğunluğu (%40,3) ve Türkçe dilinde eğitim görenlerin beşte biri (%21,2) çalışırken her zaman çözümü bilinen soruları tercih etmektedir. Bu oran Bilgisayar/Yazılım Mühendisliğinde yaklaşık olarak dörtte biri kadardır (%24,1).



**Şekil 2.**Ders bazında çözümlü soruları tercih etme dağılımı

## E. APRİORİ ALGORİTMASI İLE BİRLİKTELİK KURALI ÇIKARIMLARI

Apriori algoritması ile ölçme aracında yer alan bölümler (demografik bilgiler, ders çalışma alışkanlıkları, programlama ile olan etkileşimler, programlama öz yeterlilik algıları) arasındaki birliktelikler belirlenmeye çalışılmıştır. Birliktelik kuralları programlamaya giriş dersini alan bütün öğrencilerin verileri kullanılarak oluşturulmuştur. Buna göre en yüksek güven değerine sahip 5 kural Tablo 12’de verilmiştir.

**Tablo 12.** Apriori algoritması ile elde edilen kurallar

<b>Kural</b>	<b>f</b>	<b>G</b> <b>%</b>	<b>D</b> <b>%</b>	<b>L</b>
Programlama Öğrenme İsteği= Çok Fazla, İş Hayatında Gerekli=Kesinlikle Katılıyorum, Geçme/Kalma Durumu=Geçti⇒ Programlama Öz Yeterliliği=Yüksek	37/37	100	19	1,91
Ders İlk/Tekrar=İlk, İş Hayatında Gerekli=Kesinlikle Katılıyorum, Geçme/Kalma Durumu=Geçti⇒Programlama Öz Yeterliliği=Yüksek	30/32	94	16	1,79
Programlama Öğrenme İsteği=Çok Fazla, Öğrenmek İçin Çok Pratik Yap=Kesinlikle Katılıyorum, Programlama Öz Yeterliliği=Yüksek ⇒ İş Hayatında Gerekli=Kesinlikle Katılıyorum	30/32	94	16	2,49
Programlama Öğrenme İsteği=Çok Fazla, Geçme/Kalma Durumu=Geçti⇒Programlama Öz Yeterliliği=Yüksek	42/45	93	22	1,78
İş Hayatında Gerekli=Kesinlikle Katılıyorum, Geçme/Kalma Durumu=Geçti ⇒Programlama Öz Yeterliliği=Yüksek	41/44	93	22	1,78

\* f: Frekans, G: Güven, D: Destek, L: Lift

İlk kurala göre “Programlamayı Öğrenmeyi Ne Kadar İstiyorsunuz?” sorusuna çok fazla cevabını veren, “Programlama bilgisi ileride iş hayatında benim işime yarayabilir.” sorusuna kesinlikle katılıyorum cevabını veren ve dersten başarılı bir şekilde (CC ve üstü) geçen 37 öğrencinin tamamının programlama öz yeterliliği yüksektir. Bu kuralın güven değeri %100 olduğu için kesin kural olarak adlandırılır [29].

İkinci kuralda ise dersi ilk defa alan, programlama bilgisinin ileride iş hayatında kendisine faydalı olacağını düşünen ve dersten başarı (CC ve üstü) ile geçen 32 öğrencinin 30’unun (%94) programlama öz yeterliliğinin yüksek olduğu görülmektedir.

Üçüncü kuralda ise programla öğrenmeyi çok fazla isteyen, programlamayı öğrenmek için çok fazla pratik yapmak gerektiğini düşünen ve programlama öz yeterliliği yüksek olan öğrencilerin %94’ü programlamanın iş hayatında gerekli olduğu görüşündedir. Tablo 12’de yer alan tüm kuralların Lift değerleri 1’den büyük olmakla beraber üçüncü kural en yüksek Lift değerine sahiptir.

## **IV. SONUC**

Bu çalışmada Mühendislik ve Fen Edebiyat Fakültesinde öğrenim gören ve Programlamaya Giriş dersini alan öğrencilerin, programlama öz yeterlilik algıları, programlama öğrenme istekleri ve programlama dersi çalışma alışkanlıkları farklı değişkenlere göre incelenmiş ve bu kavramlar arasındaki birliktelik kuralları çıkarılmıştır.

Elde edilen sonuçlara göre Bilgisayar/Yazılım Mühendisliği bölümleri dışındaki bölümlerde öğrenim gören öğrencilerin programlama öz yeterlilik puanları cinsiyete, eğitim diline, çalışırken çözümlü soruları tercih etme durumuna göre değişmemektedir. Literatürde programlama öz yeterlilik algısının cinsiyete göre anlamlı farklılık gösterdiğini belirten çalışmalar olmakla birlikte bu çalışmada olduğu gibi cinsiyete göre farklılaşmadığı sonucunu destekleyen çalışmalar da bulunmaktadır [1, 5, 8]. Programlama özyeterlilik puanlarının eğitim diline göre farklılık göstermemesi Türkçe dilinde eğitim gören öğrencilerin İngilizce dilinde eğitim görenler kadar programlama konusunda kendilerini yeterli hissettikleri şeklinde yorumlanabilir.



Çalışmanın bir diğer bulgusu öğrencilerin programlama öz yeterlilik algılarının programlama öğrenme isteklerine göre farklılık göstermesidir. Programlama öğrenmeyi çok fazla isteyen öğrencilerin programlama öz yeterlilik puanlarının diğerlerinden yüksek olduğu görülmektedir. Bu durum programlama öğrenmeyi çok fazla isteyen öğrencilerin programlama ile ilgili karşılarına çıkan zorluklarda yılmadan, vazgeçmeden sorunu çözmeye çalıştıklarını göstermektedir.

Ayrıca, Bilgisayar/Yazılım mühendisliği öğrencileri ile diğer bölümlerin öğrencilerinin programlama öz yeterlilik puanlarının istatistiksel açıdan farklı olduğu görülmüştür. Bununla birlikte kendilerine meslek olarak Bilgisayar/Yazılım mühendisliğini seçen öğrencilerin programlama öz yeterlilik puanları diğer bölümlerin öğrencilerinden yüksektir. Elde edilen bu sonuç literatürdeki diğer çalışmalarla [1,6,8] da benzerlik göstermektedir.

Çalışmamızda öğrencilerin programlama öğrenme istekleri ile harf notları, cinsiyetleri ve ders tekrarları arasında anlamlı bir ilişki bulunmamıştır. Öğrencilerin programlama çalışırken bilgisayar kullanma sıklıkları ile bölümleri ve cinsiyetleri arasında bir ilişki tespit edilmemiştir.

Çalışmaya katılan öğrenciler programlamayı öğrenmek için İngilizceye hakim olmak gerektiği görüşündedirler. İlt'er'in [14] elektrik-elektronik mühendisleri ile yabancı dilde eğitim üzerine yapmış olduğu çalışmasında da, katılımcıların %41,5'i yabancı dil bilmeden alanda ilerlemenin mümkün olamayacağı görüşünde olduklarını belirtmişlerdir.

Öğrenciler iyi bir programcı olmak için iyi bir matematik altyapısına sahip olmak gerektiği görüşündedirler. Bilgisayar biliminin tarihine bakıldığında da temeli oluşturan kişilerin matematikçiler olduğu görülmektedir [30]. Gomes ve arkadaşları [15] programlama dersinden başarısız olan öğrencilerin matematikteki temel eksikliklerini tespit etmek ve bu eksikliklerin onların programlama becerileri üzerindeki etkilerini analiz etmek için yaptıkları çalışmada, programlama becerileri eksik olan öğrencilerin temel matematiksel kavramlarda da eksiklikleri olduğu sonucuna ulaşmışlardır.

Bilgisayar/Yazılım Mühendisliği dışındaki bölümlerde okuyup programlama dersini alan öğrenciler öğrendikleri bilgilerin iş hayatında kendilerine fayda sağlayacağı inancındadırlar. Öğrenciler farklı disiplinlere ait bilgiler ile kendilerini donatmaları gerektiğinin farkındadırlar.

Çalışmaya katılan bütün gruplar iyi bir programcı olmak için çok fazla pratik yapmak gerektiği konusunda hemfikirdir. Gomes ve Mendes [16] programlama eğitiminde yaşanan zorluklardan biri olarak öğrencilerin doğru çalışma yöntemini bilmemesini gösterir. Öğrencilerin bazılarının olabildiğince fazla programlama problemi çözmek yerine ders kitabını okuyarak programlamayı öğreneceğini düşündüğünden bahseder. Bizim çalışmamıza katılan öğrenciler ise programlamayı ezberleyerek değil pratik yaparak öğrenebileceklerinin farkındadırlar.

Öğrencilerin kendi başlarına veya gruplar halinde programlama öğrenme şekillerine karşı gösterdikleri olumlu görüşler Tan ve arkadaşlarının [17] yapmış oldukları çalışmanın sonuçlarıyla da örtüşmektedir. Bu çalışmada öğrenciler hocaları ve arkadaşlarıyla istişare ederek, tartışarak ve tek başlarına çalışarak programlamayı daha etkin olarak öğrendiklerini belirtmişlerdir.

Çalışmadan elde edilen bir başka sonuç da bölümlerine bakılmaksızın çalışmaya katılan öğrencilerin yarısı kadarının bilgisayarı kullanmadan programlamayı öğrenmeye çalıştıklarını belirtmesidir. Öğrencilerin bilgisayarı kullanmadan programlamayı öğrenmeye çalışmalarının bir nedeni olarak yazılan kodları derlemekte kullanılan editörün programlamaya yeni başlayan kişiler için yeterince kullanıcı dostu ve anlaşılır olmaması gösterilebilir.

Okudukları bölümden bağımsız olarak ankete katılan tüm öğrencilerin %65'i programlama çalışırken çözümünü bildiği soruları tekrar çözmeye çalışmaktadır. Böylelikle bir hata yaptıklarında kolaylıkla hatanın nerede olduğunu görebilmektedirler, kendilerini güvende hissetmektedirler. Tan ve arkadaşlarının [17] yaptıkları çalışmada lisans öğrencilerinin çoğunluğu çözümleri mevcut olan sorularla çalışmanın, programlamayı öğrenmelerine yardımcı olabileceği görüşünde hemfikirdir. Bu durumu değiştirmek ve öğrencileri çözümlerini bilmedikleri programlama problemleri ile uğraşmaya

teşvik etmek için onlara sık sık kısa sınavlar yapılabilir veya küçük ölçekli programlama projeleri ödev olarak verilebilir.

Apriori algoritması ile çıkarılan birliktelik kuralları incelendiğinde ise kesin kural olarak, programlama öğrenmek konusunda çok fazla istekli olan, programlama öğrenmenin iş hayatında kendisine fayda sağlayacağı görüşünde olan ve programlama dersinden başarı ile geçen (CC ve üstü) öğrencilerin programlama öz yeterlilik seviyesi de yüksektir sonucuna ulaşılmıştır. Bu noktadan hareketle bir öneri olarak, programlama bilgisinin profesyonel hayatta nasıl faydalı olacağı farklı alanlardan başarı/kariyer hikayeleri kullanılarak anlatılıp öğrencilerin programlama öğrenme konusundaki istekleri artırılabilir ve programlama öz yeterlilik seviyeleri yükseltilebilir.

Çalışmanın sadece bir üniversitede yapılmış olması, öğrencilerin sadece Java ile programlamaya giriş dersini almış olmaları, Bilgisayar ve Yazılım Mühendisliği öğrencilerinin ve Türkçe dilinde eğitim alan öğrencilerin diğerlerine göre daha az sayıda olması bu çalışmanın sonuçları ile ilgili genel çıkarımlar yapmamızı kısıtlamaktadır.

Sonuçların genelleştirilebilmesi bu çalışmanın birden fazla üniversitede (devlet, vakıf) daha geniş katılımlı öğrenci gruplarıyla (ulusal ve uluslararası öğrenciler) belki birkaç dönem tekrarlanması ile mümkün olabilecektir.

## **V. KAYNAKLAR**

- [1] A. Altun ve S. G. Mazman, “Programlamaya ilişkin öz yeterlilik algısı ölçeğinin Türkçe formunun geçerlilik ve güvenilirlik çalışması,” *Journal of Measurement and Evaluation in Education and Psychology*, c. 3, ss. 297-308, 2012.
- [2] A. Bandura, *Social Foundations of Thought and Action: A Social Cognitive Theory*, New Jersey, USA: Prentice Hall Yayıncılık, 1986.
- [3] N. E. Betz ve G. Hackett, “The relationship of mathematics self-efficacy expectations to the selection of science-based college majors,” *Journal of Vocational Behavior*, c. 23, s. 3, ss. 329–345, 1983.
- [4] C. A. Murphy, D. Coover ve S. V. Owen, “Development and validation of the computer self-efficacy scale,” *Educational and Psychological Measurement*, c. 49, s. 4, ss. 893–899, 1989.
- [5] V. Ramalingam ve S. Wiedenbeck, “Development and validation of scores on a computer programming self-efficacy scale and group analyses of novice programmer self-efficacy,” *Journal of Educational Computing Research*, c. 19, s. 4, ss. 367-381, 1998.
- [6] P. Askar ve D. Davenport, “An investigation of factors related to self-efficacy for Java programming among engineering students,” *The Turkish Online Journal of Educational Technology*, c. 8, s. 1, ss. 26-32, 2009.
- [7] H. Çiğdem ve O. G. Yıldırım, “Predictors of C# programming language self efficacy among vocational college students,” *International Journal on New Trends in Education and Their Implications*, c. 5, s. 3, ss. 145-153, 2014.
- [8] Ö. Korkmaz ve H. Altun, “Adapting computer programming self-efficacy scale and engineering students’ self-efficacy perceptions,” *Participatory Educational Research*, c. 1, s. 1, ss. 20–31, 2014.

- [9] Ö. Özyurt ve H. Özyurt, "A study for determining computer programming students' attitudes towards programming and their programming self-efficacy," *Eğitimde Kuram ve Uygulama*, c. 11, s. 1, ss. 51-67, 2015.
- [10] F. B. Tek, K. S. Benli ve E. Deveci, "Implicit theories and self-efficacy in an introductory programming course," *IEEE Transactions on Education*, c. 61, s. 3, ss. 218-225, 2018.
- [11] M.-J. Tsai, C.-Y. Wang ve P.-F. Hsu, "Developing the computer programming self-efficacy scale for computer literacy education," *Journal of Educational Computing Research*, c. 56, s. 8, ss. 1345–1360, 2019.
- [12] J. Kittur, "Measuring the programming self-efficacy of electrical and electronics engineering students," *IEEE Transactions on Education*, c. 63, s. 3, ss. 216-223, 2020.
- [13] F. B. Tek ve K. S. Benli. (2021, 24 Ocak). *Programlama özyeterlilik ölçeği*. [Online]. Erişim: <https://github.com/btekgit/MindsetForProgramming>
- [14] B. G. İltter, "Mühendis Bakış Açısıyla Yabancı Dilde Eğitim," *IV. Elektrik Elektronik Bilgisayar Biyomedikal Mühendislikleri Eğitimi Sempozyumu*, Eskişehir, Türkiye, 2009.
- [15] A. Gomes, L. Carmo, E. Bigotte ve A. Mendes, "Mathematics and programming problem solving," *In 3rd E-Learning Conference – Computer Science Education*, Coimbra, Portugal, 2006.
- [16] A. Gomes ve A. J. Mendes, "Learning to program - difficulties and solutions," *International conference on Engineering Education*, Coimbra, Portugal, 2007.
- [17] P. Tan, C. Ting ve S. Ling, "Learning Difficulties in Programming Courses: Undergraduates' Perspective and Perception," *2009 International Conference on Computer Technology and Development*, Kota Kinabalu, 2009.
- [18] M. Başer, "Bilgisayar programlamaya karşı tutum ölçeği geliştirme çalışması," *The Journal of Academic Social Science Studies*, c. 6, s. 6, ss. 199-215, 2013.
- [19] S. Kılıç, "Cronbach's alpha reliability coefficient," *Journal of Mood Disorders*, c. 6, s. 1, ss. 47-48, 2016.
- [20] S. Landau ve B. Everitt, *A Handbook of Statistical Analyses Using SPSS*, 1.baskı, USA: Chapman and Hall/CRC Yayıncılık, 2003.
- [21] R. Agrawal ve R. Srikant, "Fast algorithms for mining association rules in large databases," *In Proceedings of the 20th International Conference on Very Large Data Bases*, Santiago de Chile, Chile 1994, ss. 478-499.
- [22] R. G. Lomax ve D. L. Hahs-Vaughn, *An Introduction to Statistical Concepts*, 3.baskı, New York, USA: Routledge Yayıncılık, 2012.
- [23] M. Dikmen, "Veri toplama ve veri analiz yöntemleri: WPF ve Silverlight ile uygulama geliştirme," Yüksek lisans tezi, Bilgisayar Mühendisliği, Fen Bilimleri Enstitüsü, Süleyman Demirel Üniversitesi, Isparta, Türkiye, 2011.
- [24] M. R. Spiegel ve L. J. Stephens, *Schaum's Outline of Theory and Problems of Statistics*, 4.baskı, New York, USA: McGraw-Hill Yayıncılık, 2008.
- [25] Y. Ateş ve M. Karabatak, "Nicel birliktelik kuralları için çoklu minimum destek değeri," *Fırat Üniversitesi Mühendislik Bilimleri Dergisi*, c. 29, s. 2, ss. 57-65, 2017.

- [26] S. S. Jambhorkar ve V. S. Jondhale, *Data Mining Technique: Fundamental Concept and Statistical Analysis*, Horizon Books Yayıncılık, 2015.
- [27] N. Hussein, A. Alashqur ve B. Sowan, "Using the interestingness measure lift to generate association rule," *Journal of Advanced Computer Science & Technology*, c. 4, s. 1, ss. 156-162, 2015.
- [28] S.Kılıç, "Chi-square Test," *Journal of Mood Disorders*, c. 6, s. 3, ss. 180-182, 2016.
- [29] F. C. Özçakır ve A. Y. Çamurcu, "Birliktelik kuralı yöntemi için bir veri madenciliği yazılımı tasarımı ve uygulaması," *İstanbul Ticaret Üniversitesi Fen Bilimleri Dergisi*, c. 6, s. 12, ss. 21-37, 2007.
- [30] İ. Yusubov, "Bilgisayar mühendisliği eğitiminde matematik altyapı ve tarihe başvurunun önemi hakkında," *I. Elektrik-Elektronik Bilgisayar Mühendislikleri Eğitimi Sempozyumu*, Ankara, Türkiye, 2003.