

## BİLGİSAYAR MÜHENDİSLİĞİ ÖĞRENCİLERİ İÇİN PROGRAMLAMA YETENEĞİ ÖLÇEĞİ GELİŞTİRİLMESİ: HARRAN ÜNİVERSİTESİ'NDE BİR UYGULAMA

Ammar ALBAKRİ

Fatma Nur CÜCÜ

Pınar MACİT

Mehmet MİMAN

### ÖZET

Bu çalışma, Harran Üniversitesi bilgisayar mühendisliği okuyan öğrencilerin kod yazma yeteneklerini ölçmek üzere Bilgisayar Programlama Yeteneği Ölçeği (BPYÖ) geliştirmekte, yapılan analizler geliştirilen ölçeklerin yüksek derecede geçerlik ve güvenilirliğe sahip olduğunu göstermektedir. Harran Üniversitesi'nde 2018-2019 eğitim – öğretim döneminde okuyan öğrencilerin ve kolayda örnekleme ile seçilen 100 öğrenci ile gerçekleştirilmiş olan bu çalışmada çalışanların demografik özelliklerinin geliştirilen ölçeklerle belirlenen BPYÖ puanlarına (temel, nesnye, genel) etkisi araştırıldığı, sınıf, yaş, kişisel bilgisayar sahibi olma, bilinen programlama dil sayısı, en iyi bilinen programlama dili özelliklerinin temeli nesneye dayalı ve genel programlama yeteneklerini; yazılım kaynaklarının kullanımının nesneye dayalı programlama yeteneğini 0,05 anlamlılık düzeyinde etkilediği görülmektedir.

**Anahtar sözcükler:** Programlama, Programlama dili, Kod yazma yeteneği, Ölçek geliştirme, Harran Üniversitesi.

### DEVELOPMENT OF THE PROGRAMMING CAPABILITY SCALE FOR COMPUTER ENGINEERING STUDENTS: AN APPLICATION IN HARRAN UNIVERSITY

### ABSTRACT

This study develops Computer Programming Ability Scale (BPYÖ) in order to measure the students' ability to write code in Harran University computer engineering. Analysis shows that the developed scales have high reliability and validity. The aim of this study was to investigate the effect of the demographic characteristics of the employees on the BPYÖ scores (basic, object, general), which were determined by the scales developed in Harran University. being known, the number of known programming languages, the basis of the best known programming language features are object-oriented and general programming aliases; It is seen that the use of software resources affects object-oriented programming innovation at the level of 0.05 significance.

**Key words:** Programming, Programming language, Code writing, Scale development, Harran University.

## GİRİŞ

Programlamada, problemin iyi anlaşılması ve iyi analiz edilmesinin önemi herkes tarafından bilinmektedir. Programlama dillerine ait komutlar birbirleri arasında farklılık göstermesine rağmen, çözüm için kullanılacak programlama mantığı tüm dillerde benzerdir. Programlama öğretiminden önce gerçekleştirilen ve programlama için temel olarak adlandırılan algoritma diğer deyişle programlama mantığı öğretiminde ise konuşma dili kullanılmaktadır. Bunun sebebi ise herhangi bir programlama diline bağlı olmaksızın geliştirilmesinden ve kolay anlaşılabilmesinden kaynaklanmaktadır (Akademik bilişim, 2007).

Yapılan araştırmalar son on yılda bilgisayar bilimlerini seçen öğrencilerin sayısında önemli düşüşlerin olduğunu göstermektedir (Heersink & Moskal, 2010; Hoegh & Moskal, 2009). Bunun bir nedeni öğrencilerin bilgisayar programlamanın zor, sıkıcı ve asosyal olduğu algıdır (Farkas & Murthy, 2005). Programlamayı öğrenmenin zor olduğu gerçeği biliniyorken (Feldgen & Clúa, 2004; Jenkins, 2002; Lahtinen, Ala-Mutka, & Järvinen, 2005) bir de bu algılar öğrencilerin programlamaya karşı negatif tutum geliştirmesine neden olabilmektedir, çünkü tutum, algıların sonucunda oluşmaktadır (Akande, 2009).

Programlama, öğrencilerin mühendislik, bilgisayar eğitimi ve öğretim teknolojileri ve bilgisayar bilimleri gibi birçok bölümlerinde sahip olması gereken temel yetkinliklerden biridir. Lau ve Yuen (2011) 'e göre, programlama, sosyal ağ araçlarının ortaya çıkışıyla popülerliğini yitirmiş olsa da, okuryazarlık eğitiminin hafife alındığı gibi, okuryazarlık okuryazarlığının önemsiz boyutlarından biridir. Bu kurumların çoğunda beceri görmezden gelinir (Akpınar ve Altun, 2014).

Programlama dili dersi hala öğrencilerin başarısız olduğu en zor derslerden biridir. Literatürde, öğrencilerin programlamada ustalaşmaları için çeşitli talepler olsa da, çalışmaların çoğu bilgisayar bilimi öğrencileri ile akılda tutulmuştur. Bilgisayar öğretmenlerinin daha düşük dereceler için eğitildiği durumlarda, öğretmen adaylarının bu süreçte yaşadıkları zorlukları keşfetmeye ihtiyaç vardır. Bu nedenle, bu çalışmada, lisans öğrencilerinin eğitim fakültelerindeki programlama derslerindeki başarısızlıklarının nedenlerini ortaya koymak ve programlamada karşılaştıkları sorunlar hakkındaki görüşleri araştırmak amaçlanmıştır.

Bir programlama dilini öğrenmek oldukça uzun bir zaman gerektiren zor bir süreçtir. Özellikle lisans programcılığı dersleri, temel programlama bilgisine sahip olan öğrenciler tarafından zor olarak algılanmaktadır çünkü çoğu zaman yüksek düzeyli düşünme becerilerini gerektirir (Tan, Ting & Ling, 2009). Araştırmalar, öğrencilerin çoğunluğunun programlama dillerini öğrenmede zorluk yaşadıklarını göstermektedir (Ambrosio ve ark., 2011; Hawi, 2010; Aşkar ve Davenport, 2009). Bu durum programlama derslerinde başarısızlığa neden olmuştur (Robins, Rountree ve Rountree, 2003). Öğrencilerin tekrar eden başarısızlık deneyimi özellikle programlama dilini öğrenmeye yönelik heyecan ve ilgiyi yitirmesine neden olmuştur (Law, Lee ve Yu, 2010). Bu durum araştırmacıların programlama başarısını artırmaya ve bu konuda farklı yöntemler geliştirmeye çalışmasına neden olmuştur. Jiau, Chen ve Su (2009) programlama eğitiminin somutlaştırılmasına ilişkin materyaller geliştirmiştir.

## ÇALIŞMA MODELİ VE YÖNTEM

Ölçek geliştirme çalışmasında geçerlik analizi keşfedici faktör analizi ile gerçekleştirilmiş, geçerlilik analizleri gerçekleştirilmeden önce örneklem uygunluğu Kaiser-Meyer Olkin (KMO) ve Bartlett testi ile incelenmiştir. KMO katsayısı 0,70'in üzerinde değer veren veri kümeleri faktörleşme için uygundur (Pett *vd.*, 2003). Bartlett testi sonucunda elde edilen değerler istatistiksel açıdan anlamlı olduğu durumda verilerin faktör analizi için uygun olduğu kabul edilmiş olur (Munro, 2005). Geliştirilen ölçeğin güvenilirlik analizleri Cronbach's Alpha ile ölçülmüştür. Geçerlilik ve güvenilirlik analizlerinden öğrencilerin kod yazma yeteneklerine karşı tutumları tarama yaklaşımıyla (Karasar, 1999) araştırılmıştır.

Çalışma kolayda örnekleme ile seçilen 100 öğrenci ile gerçekleştirmiş, öğrencilerin kişisel demografik özellikleriyle (cinsiyet, sınıf, yaş, kişisel bilgisayara sahip olma, yazılım kaynaklarını kullanma, yazılım ile ilgilenme, bilinen programlama dili sayısı ve en iyi bilinen programlama dili) Ki-kBenzer şekilde geliştirilen Bilgisayar Programlama Yeteneği Ölçeği (BPYÖ) nde üzerinde en çok ve en az katılan maddeler için öğrencilerin katılım derecelerine kişisel özelliklerin etkisi de Ki-Kare testi ile araştırılmıştır. Gerçekleştirilen ki-kare testlerinde ilişkileri ifade eden çapraz tablolarda her bir gözdeki beklenen değer 5'ten büyükse Pearson ki-kare testi; herhangi bir gruptaki beklenen değer 5'ten küçükse Fisher kesin ki-kare testi kullanılmıştır (Boyacıoğlu ve Güneri, 2006). Öğrencilerin kod yazma yeteneklerinden etkilenmelerine onların demografik özelliklerinin etkisini incelemek için normallik testi yapılmıştır. 50 den daha az boyuttaki gruplar için Shapiro-Wilk normallik analizi sonuçları dikkate alınmıştır (Bakınız: Ek). Yapılan normallik testleri sonucunda öğrencilerin kod yazma yeteneklerinden etkilenme durumlarına etkisi araştırılan demografik faktörlerin tamamının normal dağılıma sahip olduğu görülmüş olup, ikili değişkenler için bağımsız örneklemler için t-testi; çoklu değişkenler için tek yönlü varyans analizi kullanılmıştır. Yapılan bütün testlerde anlamlılık seviyesi olarak  $p < 0.05$  kabul edilmiştir. İstatistiksel analizler SPSS v22.0 ile gerçekleştirilmiştir.

## ANALİZ VE BULGULAR

Bu bölüm çalışmaya katılanların demografik özellikleri, geliştirilen anketin geçerlik ve güvenilirlik analizlerini ve demografik faktörlerin birbirleriyle olan ilişkileri ile ölçeklerde en çok katılan ve en az katılan ifadelerle olan etkisini incelemektedir.

### Demografik Özellikler

Öğrencilerin kod yazma yeteneklerini değerlendirmeye yönelik geliştirilen ölçekler Harran Üniversitesi'nde okuyan ve kolayda örnekleme (uygun örnekleme) yolu ile seçilen toplam 500 civarındaki aktif öğrenciden 100 kişiye uygulanmış ve çalışmaya katılanların kişisel özellikleriyle kod yazma yeteneği sırasıyla Tablo 1 ve Tablo 2 de betimsel olarak belirtilmiştir. Uygun örnekleme, özellikle insana yönelik araştırmalarda çok yaygın olarak kullanılan bir örnekleme tekniği olup, pratik ve ekonomik olması açısından tercih edilir (Monette *vd.*, 1990). Betimsel araştırmalar için, evrenin %10'unu oluşturan bir örneklem, olabilecek en az oran olarak dikkate alınırken, daha küçük evrenler için %20 gerekebileceği, korelasyona bakan araştırmalarda en azından %30'luk bir örneklem, bir ilişkinin varlığı ya da yokluğunu kanıtlamada gerekli görüldüğü (Özen ve Gül, 2007) düşünüldüğünde örneklemin boyutunun uygun olduğu kanısına varılabilir.

**Tablo 1.** Katılımcıların Kişisel Demografik Özelliklerine göre Dağılımları.

Demografik Değişkenler		Frekans	Yüzde	Kümülatif Yüzde
Cinsiyet	Kadın	66	66,0	66,0
	Erkek	34	34,0	100,0
Sınıf	1.Sınıf	14	14,0	14,0
	2.Sınıf	32	32,0	46,0
	3.Sınıf	30	30,0	76,0
	4.Sınıf	24	24,0	100,0
Yaş	18-altı	5	5,0	5,0
	19-21	58	58,0	63,0
	22-24	32	32,0	95,0
	25-üzeri	5	5,0	100,0
Kişisel Bilgisayar	Evet	92	92,0	92,0
	Hayır	8	8,0	100,0
Yazılım Kaynakları	Evet	59	59,0	59,0
	Hayır	41	41,0	100,0
Yazılım İlgilenme	Evet	32	32,0	32,0
	Hayır	68	68,0	100,0
Programlama Dili Sayısı	0	11	11,0	11,0
	1	23	23,0	34,0
	2	23	23,0	57,0
	3	25	25,0	82,0
	4-üzeri	18	18,0	100,0
Programlama Dili	Yok	13	13,0	13,0
	C	34	34,0	47,0
	C++	23	23,0	70,0
	Java	21	21,0	91,0
	Diğer	9	9,0	100,0

Veri analizinden yararlanılarak oluşturulmuş tabloya bakacak olursak: Ankete katılan kadın yüzdelik oranı (%66.0) ve erkeklerin yüzdelik oranı (%34.0), çoğunun ikinci sınıf öğrencisi olduğu (%32.0) görülmektedir. Yine yaş olarak 19-21 aralığında olup (%58.0) kişisel bilgisayara sahip olan kişi oranı ise (%92.0)'dir.

Üniversitenin yazılım kaynaklarından faydalananların oranı (%59.0); yazılımla ilgilenmeyen öğrencilerin oranı (%68.0); bilinen programlama dili sayısı en çok 3 (%25.0) tane olup, en iyi bilinen programlama dili ise C (%34.0) dilidir.

### Geçerlilik Analizleri

Bu bölüm geliştirilen Bilgisayar Programlama Yeteneği Ölçeklerinin (BPYÖ) geçerlilik analizleri ve bulgularına yer vermektedir.

### Bilgisayar Programlama Yeteneği Ölçeği (BPYÖ)

Öğrencilerinin kod yazma yeteneklerini ölçmeye yönelik BPYÖ'deki 12 ifadeye benzer 5-li likert tipi (1:Çok Kötü, 2:Kötü, 3:Orta, 4:İyi, 5:Çok İyi) 12 ifade için (Tablo 6) benzer şekilde geçerlilik analizi yapılmıştır.

**Tablo 2.** BPYÖ için Maddeler.

Madde No	İfade
M1	Bildiğiniz programlama dillerindeki seviyeniz
M2	Evde oturup kod yazma alışkanlığınız
M3	Döngü kurabilme seviyeniz
M4	Dizi manipülasyonları oluşturma seviyeniz
M5	Fonksiyon yazabilme seviyeniz
M6	Recursive fonksiyon yazabilme seviyeniz
M7	String manipülasyon yapabilme seviyeniz
M8	Nesne tabanlı programlama dili kullanabilme seviyeniz
M9	Struct yazabilme seviyeniz
M10	Class oluşturma durumunuz
M11	Sıralama teknikleri oluşturma durumunuz
M12	Binary search algoritması kurabilme seviyeniz

**Tablo 3.** BPYÖ için KMO ve Bartlett Testi Sonuçları.

<b>Kaiser-Meyer-Olkin (KMO)</b>	Measure of Sampling Adequacy	0.876
	Ki-Kare	633,359
<b>Bartlett Testi</b>	Serbestlik Derecesi	66
	Sig.	0,000

Test sonuçlarına göre KMO katsayısı ,876; Bartlett testi ki-kare değeri 633,359 (sd=66) ve istatistiksel açıdan anlamlı (sig.=.000) bulunmuştur. Bu sonuçlara göre varimax rotationı kullanılarak gerçekleştirilen keşfedici faktör analizi sonuçları Tablo 3'te verilmiştir.

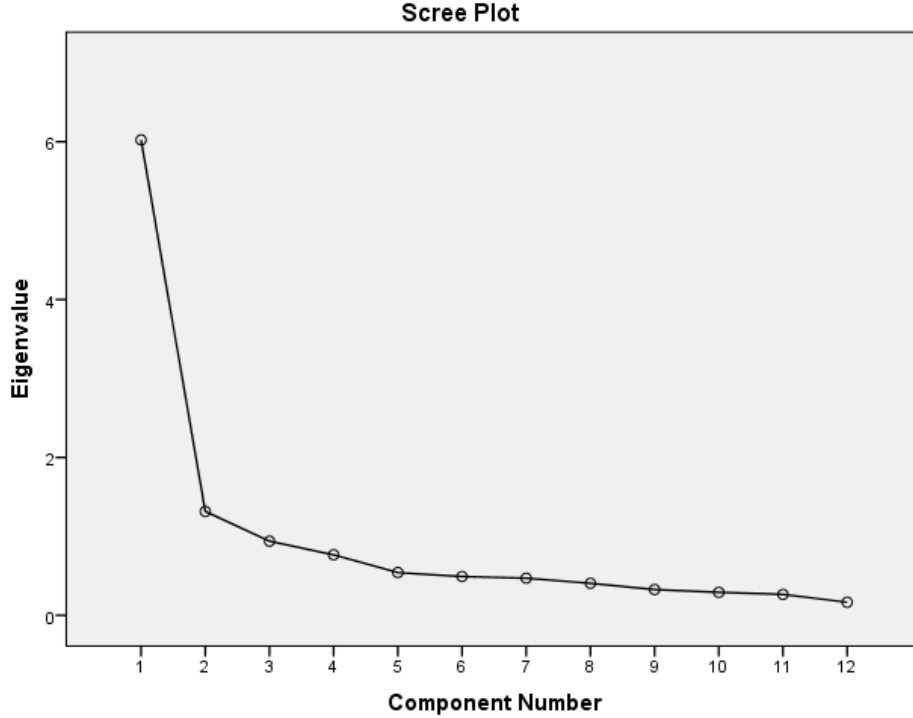
30

**Tablo 4.** BPYÖ'nin Ayırt Edici Özellikleri.

Madde	$\bar{X}$	SS	Extraction	Döndürülmüş Bileşenler Matrisi	
				1	2
M1	3,03	0,85	0,519	<b>0,694</b>	0,192
M2	3,01	1,08	0,643	<b>0,802</b>	0,011
M3	3,43	1,13	0,730	<b>0,791</b>	0,323
M4	2,92	1,05	0,626	<b>0,722</b>	0,323
M5	3,22	1,09	0,689	<b>0,762</b>	0,329
M6	2,70	1,15	0,601	<b>0,605</b>	0,484
M7	2,81	1,07	0,493	<b>0,547</b>	0,440
M8	2,73	1,10	0,494	0,348	<b>0,595</b>
M9	2,71	1,09	0,635	0,178	<b>0,777</b>
M10	3,13	1,18	0,593	0,427	<b>0,641</b>
M11	3,02	1,03	0,687	0,289	<b>0,777</b>
M12	2,95	1,18	0,632	0,072	<b>0,791</b>

Tablo 4'e göre BPYÖ'ği 2 faktörden oluşmaktadır. Her bir maddenin içinde yer aldığı faktör sırasıyla mavi ve kırmızı ile işaretlenmiş, eşit yük değerlerinin (0.40) altında kalan madde olmadığı görülmüştür. Ayrıca her maddenin iki faktördeki yük değerleri arasındaki fark 0.1'den büyük

olduğundan binişik madde olmadığı sonucuna varılmıştır. Buna göre madde 1, 2, 3, 4, 5, 6 ve 7 bir boyutta yer alırken madde 9,10,11 ve 12 diğer boyutta yer almaktadır. Bu maddelerin içeriklerine bakıldığında ilk boyutun Temel Programlama Yeteneği hakkında ikinci boyutun ise Nesneye Dayalı Programlama yönelik tutumları içerdiği görülmekte olup bundan dolayı bu boyutlar sırasıyla Temel Programlama Yeteneği ve Nesneye Dayalı Programlama olarak adlandırılmıştır.



**Şekil 1.** BPYÖ için Scree Plot

Şekil 2’deki Yamaç-Birikinti, Scree Plot, tan ikinci noktaya kadarki keskin düşüş ve ikinci noktadan sonra çizginin yatay bir seyre geçtiği görülmekte, ikinci noktaya kadar olan nokta aralıkları sayıldığında bunun iki olduğu görülmekte olup bu da keşfedici faktör analizinin ölçek için verdiği boyut sayısının iki olarak belirlenmesinin tutarlılığını göstermektedir. Geliştirilen BPYÖ ölçeğinin varyansı açıklama oranının %61,16 olduğu tespit edilmiştir.

Tablo 4’de Programlama yeteneği ölçeğini oluşturan 12 ifadeye ortalama katılım incelendiğinde genellikle 3 (Orta) puanın altında değerler aldığını bu da genel olarak Harran Üniversitesinin Bilgisayar Mühendisliği Öğrencilerin Programlama Yeteneğine çalışmaya yönelik özel bir tutuma sahip olmadığını göstermektedir. Çalışmaya katılanların kendilerini en iyi gördükleri bilgisayar programlama yeteneği M5 (“Fonksiyon yazabilme seviyeniz”,  $\bar{X} = 3.22$ ) en kötü gördükleri bilgisayar programlama yeteneği M6(“Recursive fonksiyon yazabilme seviyeniz”,  $\bar{X} = 2.70$ ) olarak belirlemektedir.

### Güvenirlilik Analizi

BPYÖ ölçeğindeki ifadeler için her bir madde silindiğinde Cronbach’s Alpha değerinin 0,896 ile 0,907 arasında olduğu görülmektedir. Buna göre ölçekteki bütün sorular gereklidir. Düzeltilmiş madde toplam korelasyonlarına bakıldığında her bir değer 0,20 den büyük olduğu görülmekte bu da ölçekte bütün maddelerin birbiriyle ilişkili olduğunu göstermektedir. Sonuç olarak BPYÖ’deki bütün maddeler oldukça güvenilir olup, maddeler birbiriyle ilişkili ve ölçek için gereklidir.

**Tablo 5.** BPYÖ için Madde Toplam İstatistikleri.

Madde	Madde Silindiğinde Ölçek Ortalaması	Madde Ölçek Varyansı	Düzeltilmiş Madde Toplam Korelasyon	Madde Silindiğinde Cronbach's Alpha Değeri
M1	32.6300	75.912	0,574	0,903
M2	32,6500	74,189	0,527	0,905
M3	32,2300	69,734	0,746	0,895
M4	32,7400	71,851	0,689	0,898
M5	32,4400	70,734	0,721	0,896
M6	32,9600	70,039	0,714	0,896
M7	32,8500	72,573	0,632	0,900
M8	32,9300	72,470	0,611	0,901
M9	32,9500	73,098	0,585	0,903
M10	32,5300	70,332	0,678	0,898
M11	32,6400	72,273	0,676	0,898
M12	32,7100	73,481	0,510	0,907

Alt boyutların güvenilirliği incelendiğinde Temel Programlama Yeteneği boyutu 0,884 alpha değerine sahipken, Nesneye Dayalı Programlama Yeteneği boyutu 0,829 alpha değerine sahip olup her iki boyutun da yüksek derecede güvenilirliğe sahip olduğu görülmektedir.

32

### Puanlandırma

Bilgisayar mühendisliği öğrencilerinin programlama yeteneklerini ölçen BPYÖ ölçeği 12 maddeden oluşmaktadır. Beşli Likert şeklinde oluşturulan ölçekte maddelerde belirtilen düşünceye bireylerden “çok iyi (5), iyi (4), orta (3), kötü (2), çok kötü (1)” olmak üzere beş derecede görüş bildirmeleri istenmektedir. BPYÖ’den alınabilecek puanlar 12-60 arasında olup, yüksek puanlar programlama yeteneğinin yüksek olduğunu ifade etmektedir. Benzer şekilde temel programlama yeteneğinden (7-35) arası, nesneye dayalı programlama yeteneğinden (5-25) arası puan alınabilmektedir.

### Hipotez Testleri

Çalışmaya katılan öğrencilerin demografik özelliklerinin (cinsiyet, sınıf, yaş, kişisel bilgisayar sahipliği, yazılım kaynakları kullanımı, yazılım ilgilenme, bilinen programlama dili sayısı, ve en iyi bildikleri programlama dili) temel programlama yeteneği, nesneye dayalı programlama yeteneği ve genel programlama yeteneği ne olan etkileri Tablo 6’da belirtilen hipotez testleriyle incelenmiştir. Programlama yetenekleri puanlarının ilgili demografik özelliklere göre normal dağılıma sahip olup olmadığı Ek-1 de belirtilen normallik testleri ortaya konulduktan sonra normal dağılıma sahip ikili değişkenler için bağımsız örneklemlili t-testi, normal dağılıma sahip olmayan ikili değişkenler için Man-Whitney U testi; normal dağılımı sahip çoklu değişkenler için ANOVA, normal dağılımı sahip olmayan çoklu değişkenler için Kurskal-Wallis analizi Ek-2 deki gibi kullanılmış, sonuçlar Tablo 7’de gösterilmiştir.



**Tablo 6.** Araştırma Hipotezleri.

Hipotez No	Açıklama
1	Bilgisayar Mühendisliği öğrencilerinin cinsiyeti ile temel programlama yetenekleri arasında anlamlı bir ilişki vardır.
2	Bilgisayar Mühendisliği öğrencilerinin sınıfları ile temel programlama yetenekleri arasında anlamlı bir ilişki vardır.
3	Bilgisayar Mühendisliği öğrencilerinin yaşları ile temel programlama yetenekleri arasında anlamlı bir ilişki vardır.
4	Bilgisayar Mühendisliği öğrencilerinin kişisel bilgisayar sahiplikleri ile temel programlama yetenekleri arasında anlamlı bir ilişki vardır.
5	Bilgisayar Mühendisliği öğrencilerinin bölümdeki yazılım kaynaklarını kullanmaları ile temel programlama yetenekleri arasında anlamlı bir ilişki vardır.
6	Bilgisayar Mühendisliği öğrencilerinin yazılıma duydukları ilgi ile temel programlama yetenekleri arasında anlamlı bir ilişki vardır.
7	Bilgisayar Mühendisliği öğrencilerinin bildikleri programlama dili sayısı ile temel programlama yetenekleri arasında anlamlı bir ilişki vardır.
8	Bilgisayar Mühendisliği öğrencilerinin en iyi bildikleri programlama dili ile temel programlama yetenekleri arasında anlamlı bir ilişki vardır.
9	Bilgisayar Mühendisliği öğrencilerinin cinsiyeti ile nesneye dayalı programlama yetenekleri arasında anlamlı bir ilişki vardır.
10	Bilgisayar Mühendisliği öğrencilerinin sınıfları ile nesneye dayalı programlama yetenekleri arasında anlamlı bir ilişki vardır.
11	Bilgisayar Mühendisliği öğrencilerinin yaşları ile nesneye dayalı programlama yetenekleri arasında anlamlı bir ilişki vardır.
12	Bilgisayar Mühendisliği öğrencilerinin kişisel bilgisayar sahiplikleri ile nesneye dayalı programlama yetenekleri arasında anlamlı bir ilişki vardır.
13	Bilgisayar Mühendisliği öğrencilerinin bölümdeki yazılım kaynaklarını kullanmaları ile nesneye dayalı programlama yetenekleri arasında anlamlı bir ilişki vardır.
14	Bilgisayar Mühendisliği öğrencilerinin yazılıma duydukları ilgi ile nesneye dayalı programlama yetenekleri arasında anlamlı bir ilişki vardır.
15	Bilgisayar Mühendisliği öğrencilerinin bildikleri programlama dili sayısı ile nesneye dayalı programlama yetenekleri arasında anlamlı bir ilişki vardır.
16	Bilgisayar Mühendisliği öğrencilerinin en iyi bildikleri programlama dili ile nesneye dayalı programlama yetenekleri arasında anlamlı bir ilişki vardır.
17	Bilgisayar Mühendisliği öğrencilerinin cinsiyeti ile genel programlama yetenekleri arasında anlamlı bir ilişki vardır.
18	Bilgisayar Mühendisliği öğrencilerinin sınıfları ile genel programlama yetenekleri arasında anlamlı bir ilişki vardır.
19	Bilgisayar Mühendisliği öğrencilerinin yaşları ile genel programlama yetenekleri arasında anlamlı bir ilişki vardır.
20	Bilgisayar Mühendisliği öğrencilerinin kişisel bilgisayar sahiplikleri ile genel programlama yetenekleri arasında anlamlı bir ilişki vardır.
21	Bilgisayar Mühendisliği öğrencilerinin bölümdeki yazılım kaynaklarını kullanmaları ile genel programlama yetenekleri arasında anlamlı bir ilişki vardır.
22	Bilgisayar Mühendisliği öğrencilerinin yazılıma duydukları ilgi ile genel programlama yetenekleri arasında anlamlı bir ilişki vardır.
23	Bilgisayar Mühendisliği öğrencilerinin bildikleri programlama dili sayısı ile genel programlama yetenekleri arasında anlamlı bir ilişki vardır.
24	Bilgisayar Mühendisliği öğrencilerinin en iyi bildikleri programlama dili ile genel programlama yetenekleri arasında anlamlı bir ilişki vardır.



Tablo7. Demografik Faktörlerin Bağımlı Değişkenlere Etkileri

Demografik Değişkenler		Bağımlı Değişken								
		BPYÖ Temel Programlama			BPYÖ Nesneye Dayalı Programlama			BPYÖ Genel Programlama		
		$\bar{X}$	SS	p	$\bar{X}$	SS	p	$\bar{X}$	SS	p
Cinsiyet	Kadın	20,51	5,10	0,144	14,31	4,05	0,478	34,83	8,48	0,214
	Erkek	22,29	6,77		14,97	4,95		37,26	10,48	
Sınıf	1.Sınıf	17,50	4,55	0,000*	11,85	3,46	0,004*	29,35	6,93	0,000*
	2.Sınıf	19,09	5,72		13,46	4,22		32,56	9,19	
	3.Sınıf	22,06	5,60		16,03	4,52		38,10	9,43	
	4.Sınıf	24,75	4,29		15,66	3,70		40,41	6,67	
Yaş	18-altı	16,40	2,88	0,005*	11,00	2,64	0,008*	27,40	4,27	0,001*
	19-21	19,89	6,03		13,82	4,42		33,72	9,83	
	22-24	23,31	4,29		15,75	3,32		39,06	5,96	
	25-üzeri	26,00	5,52		18,60	5,98		44,60	10,28	
Kişisel Bilgisayar	Evet	21,60	5,60	0,003*	14,89	4,20	0,008*	36,50	8,91	0,002*
	Hayır	15,50	4,50		10,50	3,77		26,00	7,50	
Yazılım Kaynakları	Evet	21,59	5,74	0,326	15,28	4,36	0,033*	36,88	9,14	0,113
	Hayır	20,43	5,76		13,46	4,08		33,90	9,18	
Yazılım İlgilenme	Evet	22,37	6,87	0,167	15,15	4,98	0,331	37,53	10,65	0,166
	Hayır	20,52	5,09		14,25	3,98		34,77	8,42	
Programlama Dili Sayısı	0	13,54	3,93	0,000*	9,45	3,14	0,000*	23,00	5,98	0,000*
	1	17,91	3,64		12,04	3,69		29,95	6,24	
	2	21,43	5,52		14,52	2,98		35,95	7,43	
	3	25,08	3,77		17,68	3,03		42,76	5,36	
	4-üzeri	23,94	4,92		16,50	4,14		40,44	8,16	
Programlama Dili	Yok	14,84	4,68	0,000*	10,61	3,59	0,000*	25,46	7,44	0,000*
	C	19,67	5,15		13,26	3,65		32,94	7,78	
	C++	23,34	4,23		16,68	3,88		40,13	7,44	
	Java	23,47	4,81		15,42	3,35		38,90	6,22	
	Diğer	24,44	6,78		17,22	5,56		41,66	12,08	

\* p&lt;0.05, ilişki anlamlı

Tablo 7'ye göre cinsiyet temel, nesneye dayalı ve genel programlama yeteneğine 0,05 hata düzeyinde etkisi saptanamamıştır. Sınıf, yaş, kişisel bilgisayar sahipliği, ve bilinen programlama dili sayısının ise incelenen temel, nesneye dayalı ve genel programlama yeteneklerine 0,05 anlamlılık düzeyinde etkili olduğu görülmektedir. Buna göre temel ve genel programlama yetenekleri sınıf düzeyi arttıkça artmakta (temel yetenek en çok 4. sınıflarda  $\bar{x} = 24,75$  , en az 1. sınıflarda  $\bar{x} = 17,50$  ; genel yetenek en çok 4. sınıflarda  $\bar{x} = 40,41$ , en az 1. sınıflarda  $\bar{x} = 29,35$ ) iken nesneye dayalı programlama yeteneği en fazla 3. sınıflarda ( $\bar{x} = 16,03$ ) değerlendirilmektedir. Bu da bilgisayar mühendisliği öğrencilerin müfredatları boyunca yeteneklerini geliştirmekte olduklarından üçüncü sınıfta ise daha çok nesneye dayalı eğitim alıyor olmalarından kaynaklanabilir. Temel, nesneye dayalı ve genel programlama yetenekleri yaş arttıkça artmakta (temel yetenek en çok 25 ve üzeri  $\bar{x} = 26,0$  , en az 18 - altı  $\bar{x} = 16,40$ ; nesneye dayalı yetenekte en çok 25 ve üzeri  $\bar{x} = 18,60$  en az 18 - altı  $\bar{x} = 11,0$  , genel yetenek en çok 25 ve üzeri  $\bar{x} = 44,60$  en az 18 - altı  $\bar{x} = 27,40$  ) 'dır. Kişisel bilgisayara sahip olma (temel yetenekte  $\bar{x} = 21,60$ ; nesneye dayalı yetenekte ise  $\bar{x} = 14,89$  , genel yetenekte  $\bar{x} = 36,50$ ). Öğrencilerin

yazılım kaynaklarından faydalanma konusunda temel ve genel programlama yeteğini 0,05 hata düzeyin saptanamamıştır.Yazılım ile ilgilenmede temel, nesneye dayalı ve genel programlama yeteneğine 0,05 hata düzeyinde etki saptanamamıştır.Programlama dili sayısı ise (temel yetenek en çok  $\bar{x} = 25,08$  , en az  $\bar{x} = 13,54$ ; nesneye dayalı yetenekte en çok  $\bar{x} = 17,68$  en az  $\bar{x} = 9,45$  , genel yetenek en çok 3 ve üzeri  $\bar{x} = 42,76$  en az  $\bar{x} = 23,00$  )'dır.Yani genel olarak istatistik verilerine bakıldığı zaman fazla bilinen dil sayısı 3 olup en az bilin dil sayısı ise 0'dır.Son olarak programlamada dili sayısında ise (temel yetenek en çok bilinen diğer olup  $\bar{x} = 24,44$ , en az bilinen dil sayısı 0 olup değeri  $\bar{x} = 14,84$  ; nesneye dayalı yetenekte en iyi bilinenin diğer olup  $\bar{x} = 17,22$  en az bilinenin ise hiç olmadığı  $\bar{x} = 10,61$  , genel yetenek en çok bilinen konusunda diğer olup  $\bar{x} = 41,66$  en az bilinen ise yok olup  $\bar{x} = 25,46$ )'dir.Yani bu kısımda genel veri sonucuna bakacak olursak en iyi bilinen programlama dili diğer olup en az bilinen programlama dili ise hiç yoktur.

## SONUÇ

Bu çalışmada Bilgisayar Programlama Yeteneği Ölçeği (BPYÖ) nin geçerlik ve güvenilirlik çalışması yapılmıştır. Öğrencileri kod yazma yeteneklerini ölçmek üzere geliştirilen ölçekler 12 madden oluşmakta olup, ölçeklerin geçerlik ve güvenilirliklerinin yüksek olması öğrencilerin kod yazma yeteneklerinin değerlendirmede kullanılabileceğini göstermektedir. Harran Üniversitesinde eğitim gören katılımcıların cevapları analiz edildiğinde öğrencilerin cinsiyetlerinin, yaşlarının, sınıflarının, yazılım kaynaklarının, kişisel bilgisayarlarının, daha önce yazılım ile ilgilenmelerinin, bildiği programlama dili sayısının ve en iyi bildiği programlama dilinin kod yazma yeteneklerine etkili olduğu görülmüştür.

Test edilen 24 hipotezin sonuçları toplu olarak Tablo 8 de verilmektedir.

**Tablo 8.** Araştırılan Hipotez Test Sonuçları.

Hipotez No	Sonuç	Hipotez No	Sonuç	Hipotez No	Sonuç
1	X	9	X	17	X
2	√	10	√	18	√
3	√	11	√	19	√
4	√	12	√	20	√
5	X	13	√	21	X
6	X	14	X	22	X
7	√	15	√	23	√
8	√	16	√	24	√

X: İlişki %95 güvenilirlik seviyesinde anlamlı değil.

√: İlişki %95 güvenilirlik seviyesinde anlamlı.

Çalışma Harran Üniversite'sinin kolayda örnekleme ile ulaşılabilen eğitim gören öğrencilerinin katılımıyla sınırlıdır. Aynı çalışma güncel olarak daha büyük örneklemlerle tekrarlanabilir. Geliştirilen ölçekler yıllara göre kod yazma yeteneğinin seviyesini belirlemek amacıyla düzenli olarak kullanılabilir. Benzer şekilde geliştirilen ölçekler diğer üniversitelerde ve iş kollarında uygulanabilir ve öğrencilerin kod yazma yeteneklerinden nasıl etkilendikleri belirlenebilir.

## KAYNAKÇA

Akademik bilişim, 2007, Bilgisayar Programlama Öğretiminde Yeni Bir Yaklaşım. Taner Arabacıoğlu , Halil İbrahim Bülbül , Ali Filiz.

- AKANDE, S. O. (2009). Knowledge, Perception, and Attitudes of Library Personnel towards Preservation of Information Resources in Nigerian Federal University Libraries. *Library Philosophy and Practice (e-journal)*, 303.
- Akpınar, Y., & Altun, A. (2014). Bilgi toplumu okullarında programlama eğitim gereksinimi. *İlköğretim Online* (13)1, 1-4.
- Aşkar, P., & Davenport, D. (2009). An investigation of factors related to self-efficacy for Java programming among engineering students. *The Turkish Online Journal of Educational Technology (TOJET)*, 8(1).
- FARKAS, D., & MURTHY, N. (2005). Attitudes toward computers, the introductory course and recruiting new majors: Preliminary results. İçinde 17th Workshop of the Psychology of Programming Interest Group, Sussex University (ss 268–277). Citeseer.
- FELDGEM, M., & CLÚA, O. (2004). Games as a motivation for freshman students learn programming. İçinde *Frontiers in Education, 2004. FIE 2004. 34th Annual* (ss S1H/11–S1H/16 Vol. 3). IEEE.
- Hawi, N. (2010). Causal attributions of success and failure made by undergraduate students in an introductory-level computer programming course. *Computers and Education* 54 (2010), 1127-1136.
- HEERSINK, D., & MOSKAL, B. M. (2010). Measuring high school students' attitudes toward computing. İçinde *Proceedings of the 41st ACM technical symposium on Computer science education* (ss 446–450). ACM.
- Jiau, H. C., Chen, J. C., & Su, K. F. (2009). Enhancing self-motivation in learning programming using game-based simulation and metrics. *IEEE Transactions on Education*, 52(4), 555-562.
- Lau, W. W. F., & Yuen, A. H. K. (2011). Modeling programming performance: Beyond the influence of learner characteristics. *Computers and Education*, 57(1), 1202-1213.
- Law, K., Lee, V., & Yu, Y. T. (2010). Learning motivation in e-learning facilitated computer programming courses. *Computers and Education* 55 (2010), 218-228.
- Robins, A., Rountree, J., & Rountree, N. (2003). Learning and Teaching Programming: A Review and Discussion. *Computer Science Education*, 13(2), 137-172.
- Turkish Online Journal of Qualitative Inquiry, (2014). Undergraduate students' experiences in programming: Difficulties and obstacles.

## EK

## Normallik Test Sonuçları

Demografik Değişkenler		BPYÖ Temel Programlama			BPYÖ Nesneye Dayalı Programlama			BPYÖ Genel Programlama		
		İstatistik	sd	Sig.	İstatistik	sd	Sig.	İstatistik	sd	Sig.
Cinsiyet	Kadın	0,100	66	0,171 <sup>b</sup>	0,100	66	0,096 <sup>b</sup>	0,086	66	0,200 <sup>b</sup>
	Erkek	0,966	34	0,366 <sup>a</sup>	0,942	34	0,073 <sup>a</sup>	0,964	34	0,320 <sup>a</sup>
Sınıf	1.Sınıf	0,974	14	0,929 <sup>a</sup>	0,897	14	0,102 <sup>a</sup>	0,934	14	0,348 <sup>a</sup>
	2.Sınıf	0,962	32	0,319 <sup>a</sup>	0,970	32	0,495 <sup>a</sup>	0,952	32	0,163 <sup>a</sup>
	3.Sınıf	0,964	30	0,399 <sup>a</sup>	0,941	30	0,098 <sup>a</sup>	0,939	30	0,086 <sup>a</sup>
	4.Sınıf	0,979	24	0,881 <sup>a</sup>	0,976	24	0,811 <sup>a</sup>	0,970	24	0,676 <sup>a</sup>
Yaş	18-altı	0,871	5	0,269 <sup>a</sup>	0,942	5	0,679 <sup>a</sup>	0,804	5	0,087 <sup>a</sup>
	19-21	0,199	58	0,040 <sup>b*</sup>	0,104	58	0,181 <sup>b</sup>	0,098	58	0,200 <sup>b</sup>
	22-24	0,972	32	0,544 <sup>a</sup>	0,978	32	0,734 <sup>a</sup>	0,987	32	0,959 <sup>a</sup>
	25-üzeri	0,885	5	0,331 <sup>a</sup>	0,669	5	0,004 <sup>b*</sup>	0,894	5	0,379 <sup>a</sup>
Kişisel Bilgisayar	Evet	0,076	92	0,200 <sup>b</sup>	0,108	92	0,010 <sup>b*</sup>	0,074	92	0,200 <sup>b</sup>
	Hayır	0,974	8	0,924 <sup>a</sup>	0,850	8	0,095 <sup>a</sup>	0,952	8	0,736 <sup>a</sup>
Yazılım Kaynakları	Evet	0,103	59	0,189 <sup>b</sup>	0,124	59	0,024 <sup>b*</sup>	0,079	59	0,200 <sup>b</sup>
	Hayır	0,972	41	0,404 <sup>a</sup>	0,975	41	0,488 <sup>a</sup>	0,982	41	0,737 <sup>a</sup>
Yazılım İlgilenme	Evet	0,965	32	0,371 <sup>a</sup>	0,976	32	0,677 <sup>a</sup>	0,976	32	0,677 <sup>a</sup>
	Hayır	0,113	68	0,032 <sup>b*</sup>	0,087	68	0,200 <sup>b</sup>	0,087	68	0,200 <sup>b</sup>
Programlama Dili Sayısı	0	0,983	11	0,982 <sup>a</sup>	0,966	11	0,838 <sup>a</sup>	0,980	11	0,968 <sup>a</sup>
	1	0,944	23	0,220 <sup>a</sup>	0,960	23	0,469 <sup>a</sup>	0,984	23	0,959 <sup>a</sup>
	2	0,963	23	0,524 <sup>a</sup>	0,952	23	0,324 <sup>a</sup>	0,982	23	0,942 <sup>a</sup>
	3	0,956	25	0,333 <sup>a</sup>	0,946	25	0,207 <sup>a</sup>	0,913	25	0,037 <sup>a*</sup>
	4-üzeri	0,946	18	0,363 <sup>a</sup>	0,946	18	0,370 <sup>a</sup>	0,911	18	0,089 <sup>a</sup>
Programlama Dili	Yok	0,970	13	0,894 <sup>a</sup>	0,977	13	0,963 <sup>a</sup>	0,968	13	0,874 <sup>a</sup>
	C	0,977	34	0,681 <sup>a</sup>	0,961	34	0,263 <sup>a</sup>	0,929	34	0,222 <sup>a</sup>
	C++	0,966	23	0,589 <sup>a</sup>	0,971	23	0,714 <sup>a</sup>	0,957	23	0,407 <sup>a</sup>
	Java	0,953	21	0,380 <sup>a</sup>	0,965	21	0,617 <sup>a</sup>	0,842	21	0,003 <sup>a*</sup>
	Diğer	0,948	9	0,668 <sup>a</sup>	0,884	9	0,172 <sup>a</sup>	0,927	9	0,455 <sup>a</sup>

37

a:Shapiro-Wilk Normallik testi ile elde edilmiştir

b:Kolmogorov-Simirov Normallik testi ile elde edilmiştir

\* p&lt;0.05, normal dağılıma sahip değil

## Ek-2: Uygulanan Hipotez Testleri

Faktör	Durum	Bağımlı Değişken		
		BPYÖ-Temel	BPYÖ-Nesneye Dayalı	BPYÖ-Genel
Cinsiyet	Dağılım	Normal	Normal	Normal
	Test	t-test	t-test	t-test
Sınıf	Dağılım	Normal	Normal	Normal
	Test	ANOVA	ANOVA	ANOVA
Yaş	Dağılım	Normal Değil	Normal Değil	Normal
	Test	Kruskal Wallis	Kruskal Wallis	ANOVA
Kişisel Bilgisayar	Dağılım	Normal	Normal Değil	Normal
	Test	t-test	Man-Whitney U	t-test
Yazılım Kaynakları	Dağılım	Normal	Normal Değil	Normal
	Test	t-test	Man-Whitney U	t-test
Yazılım İlgilenme	Dağılım	Normal Değil	Normal	Normal
	Test	Man-Whitney U	t-test	t-test
Programlama Dili Sayısı	Dağılım	Normal	Normal	Normal Değil
	Test	ANOVA	ANOVA	Kruskal-Wallis
Programlama Dili	Dağılım	Normal	Normal	Normal Değil
	Test	ANOVA	ANOVA	Kruskal-Wallis