Ahi Evran Üniversitesi Kırşehir Eğitim Fakültesi Dergisi (KEFAD)
Cilt 17, Sayı 2, Ağustos 2016, Sayfa 509-525

Makine Mühendisliği ve Ekonometri Öğrencilerinin Programlamaya İlişkin Öz Yeterlik Algılarının İncelenmesi

Deniz Mertkan GEZGİN[1], Müge ADNAN[2]

Geliş Tarihi: *01.03.2016* ***Kabul Ediliş Tarihi:****26.06.2016*

ÖZ

Bilgisayar programlama, bilişim sektöründe yaşanan hızlı değişimler sayesinde sektörün önemli bir dalı haline gelmiştir. Yükseköğretimde Bilgisayar ve Yazılım Mühendisliği bölümlerinin yanı sıra, farklı mühendislik bölümleri ile matematik ve fen bilimlerine dayalı diğer bölümlerde de bilgisayar programcılığına verilen önem artmaktadır. Bilgisayar bilimleri haricindeki bölümlerde akademik başarı ve öğrenci memnuniyeti açısından bilgisayar programlama öğretiminde bir takım sıkıntılar ile karşılaşılmaktadır. Bu sıkıntılar ve öz yeterlik algısının akademik başarı üzerindeki rolünden yola çıkarak, bu çalışmanın amacı Türkiye'de bir devlet üniversitesinin makine mühendisliği ve ekonometri bölümlerinde öğrenim gören lisans öğrencilerinin bilgisayar programlamaya ilişkin öz yeterlik algılarını incelemektir. Altun ve Mazman (2012) tarafından Türkçeye uyarlanan Programlamaya İlişkin Öz Yeterlik Algısı Ölçeği'nin veri toplama aracı olarak kullanıldığı ve toplam 138 öğrencinin katıldığı çalışmada, veriler betimsel istatistikler ve bağımsız örneklem t-testi ile analiz edilmiştir. Araştırmanın bulguları, öğrencilerin programlamaya ilişkin öz yeterlik algılarının düşük olduğunu göstermektedir. Öğrenciler basit programlama görevlerini algılayabilmekte, ancak karmaşık programlama görevlerini algılamada sorun yaşamaktadırlar. Araştırmanın diğer bulgularına göre, erkek öğrencilerin kız öğrencilere, Makine Mühendisliği öğrencilerinin Ekonometri öğrencilerine kıyasla programlamaya yönelik öz yeterlik algılarının yüksek olduğu görülmektedir. Öğrencilerin bilgisayar programlamaya ilişkin öz yeterlik algılarının yanı sıra hem tutum hem de önceki programlama deneyimine ek olarak öğrencilerin temel düzeyde programlama bilgisi içeren derslerdeki problem çözme yeteneklerinin ve yansıtıcı düşünme, algoritmik düşünme gibi üst düzey beceriler ile arasındaki ilişkilerin araştırılması faydalı olacaktır. Bilgisayar programlama ile ilgili yapılacak tüm çalışmalar, günümüzde son derece önemli bir yere sahip bu becerinin en etkin şekilde kazanılmasını sağlamaya ışık tutacaktır.

Anahtar kelimeler: Bilgisayar programlama, programlama dili, öz yeterlik

[1] Yrd. Doç. Dr., Trakya Üniversitesi, Eğitim Fakültesi, BÖTE Bölümü, mertkan@trakya.edu.tr
[2] Yrd. Doç. Dr., Muğla Sıtkı Koçman Üniversitesi, Eğitim Fakültesi, BÖTE Bölümü, mugea@mu.edu.tr

An Investigation of Perceived Self Efficacy of Engineering and Econometrics Students on Computer Programming

ABSTRACT

Computer programming has come to the fore of the sector due to unprecedented pace of developments in information and communication technologies. Besides computer and software engineering, other engineering programmes and the programmes based on mathematics and natural sciences also give significant importance to programming. In view of the problems encountered in teaching computer programming in terms of academic achievement and student satisfaction for programmes other than computer sciences, and based on the role of perceived self-efficacy on academic achievement, this study aims to investigate perceived computer programming self-efficacy of undergraduate students of mechanical engineering and econometrics at a Turkish state university. Data were collected from 138 students through a Programming Self-Efficacy Scale, adapted by Altun and Mazman (2012) and analysed using descriptive statistics and independent sample t-test. Findings from the study indicates a low level of perceived self-efficacy for computer programming. Despite understanding simple programming tasks, students tend to encounter problems in understanding complex tasks. Other findings indicate that male students have a higher level of programming self-efficacy compared to female students; similarly, mechanical engineering students' perceived self-efficacy is higher compared to econometrics students. It is important to continue to advance research studies on perceived self-efficacy for computer programming, and investigating relationship between self-efficacy, attitude, prior programming experience, and ownership of higher order thinking skills. Studies on computer programming will shed light and contribute to efforts for acquiring this very important skill of 21st century in the most efficient manner.

Keywords: Computer programming, programming language, self-efficacy.

GİRİŞ

21. yüzyıl bireyinin sahip olması beklenen becerilerden biri de bilgisayar programlama becerisidir (Balanskat & Engelhardt, 2014). Bilgisayar programlama, bilişim sektöründe yaşanan hızlı değişimler sayesinde sektörün önemli bir dalı haline gelmiş, buna paralel olarak birçok üniversitenin bünyesinde Bilgisayar Mühendisliği bölümlerine ek olarak Yazılım Mühendisliği bölümleri açılmıştır. Aynı zamanda farklı mühendislik bölümleri ile matematik ve fen bilimlerine dayalı diğer bölümlerde de bilgisayar programcılığına verilen önem artmakta, bu bölümlerin müfredatlarında farklı isimler altında seçmeli ya da zorunlu ders olarak bilgisayar programlama yer almaktadır.

Bilgisayar programlama denildiğinde akla bilgisayar komutları ve bunların belirli bir yapıya göre düzenlenmesi gelmektedir. Hâlbuki bilgisayar programlamada kodlama sürecine geçilmeden önce problemin iyi anlaşılması, mantıksal ve matematiksel düşünme becerileri sayesinde iyi analiz edilerek parçalara ayrılması gerekmektedir (Saeli vd., 2011). Nitekim bir bilgisayar programlama dilinin içerdiği komutlar ile farklı yazılış biçimlerinde olmasına

rağmen problemin çözümü için kullanılacak mantık ve yapılar tüm diller için benzerdir. Bilgisayar programlamayı soyutlama, genelleme, transfer ve eleştirel düşünme gibi birçok özelliğin bileşkesi olarak düşünebiliriz (Gomez 2007). Bu yüzden bireyin bilgisayar programlama yapabilmesi için, problem çözme, mantıksal ve matematiksel düşünme gibi üst düzey düşünme becerilerine sahip olması gerektiği düşünülmektedir (Fang 2012; Korkmaz 2012; Lau & Yuen 2009; Sebetci & Aksu 2014; Wang vd. 2012).

Korkmaz (2012) akademik başarı ve öğrenci memnuniyeti açısından bilgisayar programlama öğretiminde sorunlarla karşılaşıldığını ifade etmektedir. Özellikle de bilgisayar bilimleri ile ilgili olmayan bölümlerde bilgisayar programlama dersinin akademik başarı düzeyinin düşük olduğu ve başlangıç düzeydeki öğrencilerin bilgisayar programlama öğrenimine ilişkin güçlükler yaşadığı farklı çalışmalarda vurgulanmıştır (Ambrosio vd. 2011; Aşkar & Davenport 2009; Başer 2013; Feldgen & Clúa 2004, Gerald & Mark 1989; Jenkins 2002; Lahtinen, Ala-Mutka, & Järvinen 2005; Milne & Rowe, 2002; Pillay & Jugoo 2005; Tan, Ting & Ling 2009; Wiedenbeck, LaBelle, & Kain 2004). Bu öğrenim güçlüğünün sebeplerine yönelik tartışmalar ve araştırmalar sürmektedir. Bilgisayar programlama derslerinde başarıyı etkileyen faktörler arasında öğrencilerin üst düzey becerilere sahip olmamaları ile programlama sürecinin karmaşıklığı ve bilgisayar programlama derslerinde doğru öğretim stratejilerinin, yöntem ve tekniklerin kullanılmaması gibi faktörler sayılabilir (Jenkins 2002; Korkmaz 2012; Lau & Yuen 2011). Ayrıca öğrencilerin programlamaya karşı tutumları ile öz yeterlik algılarının bu derslere ilişkin akademik başarıda önemli rol oynadığı da söylenmektedir (Aşkar & Davenport 2009; Mazman & Altun 2013; Wiedenbeck, LaBelle, & Kain 2004)

Bandura (1977) öz yeterlik inancını "bireyin olası durumlar ile başa çıkabilmek için gerekli olan eylemleri ne kadar iyi yapabildiklerine ilişkin inançları" olarak tanımlar (Akt. Akkoyunlu & Orhan 2003). Öz yeterlik, bireyin bir görevi başarmadaki seçimlerini, zorluklarla başa çıkmadaki direncini, çaba seviyesini ve en önemlisi de performansını etkilemektedir ve yüksek öz yeterlik sahibi bireylerin karşılaştıkları problemler karşısında çözüm üretebildikleri görülmektedir (Bandura 1977; Sue & Garry 2008).

Öz yeterlik özellikle de eğitim alanında önemli ve faydalı olarak değerlendirilir. Zimmermann (2000) öz yeterlik inancının akademik motivasyon üzerindeki etkisinden bahseder ve Bandura'ya (1997) atıfta bulunarak yeteneklerinden şüphe duyan öğrencilerle karşılaştırıldıklarında, öz yeterlik düzeyi yüksek öğrencilerin hazırbulunuşluklarının da daha yüksek olduğunu, daha çok çalıştıklarını, daha dayanıklı olduklarını ve problemle karşılaştıkları zaman daha az duygusal tepki verdiklerini ifade eder. Benzer olarak öz yeterlik inancı, öz düzenleme süreçlerinin kullanımı yoluyla öğrencilerin kendi öğrenmelerini de güdülemesine olanak sağlar. Öz yeterlik düzeyi yüksek olan öğrencilerin öğrenmeye ilişkin motivasyonlarının ve öz-düzenleme becerilerinin de yüksek

olmasının akademik performansla olan ilişkilerini araştıran çalışmalar bulunmaktadır (Aşkar & Davenport 2009; Zimmermann 2000).

Aşkar ve Davenport (2009) bilgisayar öz yeterlik inancına ilişkin yapılan çalışmaların çeşitliliğine dikkat çekerken, bilgisayar programlama becerilerinin ediniminde öz yeterliğin rolüne fazla önem verilmediğini ifade etmektedir. Alanyazında bulunan çalışmalar arasında Ramalingam ve Wiedenbeck'in (1998) geliştirdiği ve C++ programlama diline ilişkin öz yeterlik ölçeği kullanılarak elde edilen bulguların, öğrencilerin programlamaya ilişkin öz yeterlik algılarında cinsiyetin istatistiksel olarak anlamlı bir fark oluşturmadığı, ancak, kuramla paralel olarak, programlama dersi deneyiminin özellikle de öz yeterlik puanları başlangıçta düşük olan öğrenciler için anlamlı bir fark oluşturduğuna işaret ettiği görülmektedir. Yine başka bir çalışmada (Ramalingam, La Belle & Wiedenbeck 2004), programlama deneyiminin programlama öz yeterlik inancını etkilediği tespit edilmiştir. Aşkar ve Davenport'un (2009) farklı alanlardaki mühendislik öğrencileri ile yürüttüğü çalışmada, programlamaya ilişkin öz yeterlik algısının cinsiyete ve bölümlere göre anlamlı farklılık gösterdiği, erkeklerde ve bilgisayar mühendisliği öğrencilerinde öz yeterlik algısının daha yüksek olduğu sonucuna ulaşılmıştır. Jegede (2009) de, farklı branşlardaki mühendislik öğrencileri ile yaptığı çalışmada, programlama öz yeterlik algısında cinsiyete göre anlamlı farklılık olduğu ve erkek öğrencilerin programlama öz yeterliklerinin kız öğrencilere göre yüksek olduğu sonucuna ulaşmıştır. Uluslararası alanyazında öğrencilerin programlamaya ilişkin öz yeterlik algılarının orta düzeyde olduğuna dair çalışmalar bulunmaktadır (Hawi 2010; Hernane, Gilney & Marcelo 2010; Pereira, Zebende & Moret 2010; Robins, Rountree & Rountree 2003).

Türkiye'de son dönemde yapılan çalışmalara bakıldığında, Altun ve Mazman (2012) farklı branşlardaki mühendislik öğrencileri ve bilgisayar öğretmeni adayları ile yaptıkları çalışmada programlamaya ilişkin öz yeterlik algısının cinsiyete göre farklılık göstermediğini, ancak sınıf, programlamaya yönelik alınan dersler ve programlama deneyimi değişkenlerine göre farklılıklar tespit edildiğini ortaya koymuştur. Korkmaz'ın (2013) BÖTE öğrencileriyle yaptığı çalışmada da, öğretmen adaylarının programlamaya ilişkin öz yeterlik algılarının orta düzeyde olduğu ve cinsiyet ya da sınıf düzeyine göre bir farklılık gözlenmediği sonucuna ulaşılmıştır. Yine Korkmaz ve Altun (2014) da, 378 mühendislik öğrencisi ile yaptıkları çalışmada programlamaya ilişkin öz yeterlik algısının cinsiyete göre anlamlı farklılık göstermediği, ancak bilgisayar mühendisliği öğrencilerinin öz yeterlik algısının elektrik-elektronik mühendisliği öğrencilerinden daha yüksek olduğu sonucuna ulaşmıştır. Özyurt ve Özyurt'un (2015) üç farklı bilgisayar programcılığı ön lisans programına devam eden 325 öğrenciyle yaptığı çalışmanın sonucunda ise, öğrencilerin programlama öz yeterliklerinin orta düzeyde olduğu ve cinsiyet, sınıf düzeyi ve öğrenim türü değişkenlerine göre istatistiksel olarak anlamlı fark taşıdığı tespit edilmiştir.

Görüleceği üzere, özellikle de ulusal alanyazında geçen çalışmaların çoğunluğu mühendislik ve bilgisayar bilimleri alanına odaklanmıştır. Ancak, programlama

dersleri yalnızca bu alanlarla sınırlı değildir. Programlama becerilerinin her geçen gün daha da önem kazanması, bu becerilerin kazandırılmasına yönelik derslerin yükseköğretim düzeyinde farklı çalışma alanlarında da rağbet görmesine yol açmaktadır. Dolayısıyla, bilgisayar bilimleri haricindeki bölümlerde eğitim gören ve programlama dersi alan öğrencilerin katkı sağladığı çalışmaların alanyazına kazandırılması önemli görülmektedir. Bu anlamda, farklı bölümlerdeki öğrencilerin ders hakkındaki öz yeterlik algısını tespit etmenin gerek yapılacak çalışmalara gerekse verilen eğitimin niteliğinin artırılması ve öğretim yöntemleri açısından olası problemlerin çözülmesine katkı sağlayacağı düşünülmektedir.

Araştırmanın Amacı

Mevcut çalışma, Türkiye'de bir devlet üniversitesinde makine mühendisliği ve ekonometri bölümlerinde öğrenim gören ve 2014-2015 öğretim yılı bahar döneminde bölüm dışı seçmeli ders olarak bilgisayar programlama dersi alan lisans öğrencilerinin bilgisayar programlamaya ilişkin öz yeterlik algılarının incelenmesini amaçlamaktadır. Çalışma kapsamında, öğrencilerin programlamaya ilişkin öz yeterlik algıları ölçeğin geneli ve ölçek boyutları olan 'basit programlama görevleri gerçekleştirme' ve 'karmaşık programlama görevleri gerçekleştirme' açısından incelenmiş, cinsiyet ve bölüm değişkenlerine göre de çözümleme yapılmıştır. Bu amaç doğrultusunda aşağıdaki sorulara cevap aranmıştır.

1. Bilgisayar Programlama dersi alan öğrencilerin programlamaya ilişkin öz yeterlik algıları nedir?
2. Bilgisayar Programlama dersi alan öğrencilerin programlamaya ilişkin öz yeterlik algıları 'basit programlama görevleri' ve 'karmaşık programlama görevleri' boyutlarına göre değişiklik göstermekte midir?
3. Bilgisayar Programlama dersi alan öğrencilerin programlamaya ilişkin öz yeterlik algılarında cinsiyete göre anlamlı farklılık var mıdır?
4. Bilgisayar Programlama dersi alan öğrencilerin programlamaya ilişkin öz yeterlik algılarında devam edilen bölüme göre anlamlı farklılık var mıdır?

YÖNTEM

Bu çalışmada tarama modeli kullanılmıştır. Tarama modelleri, geçmişte var olmuş ya da var olan bir durumu mevcut şekliyle betimlemeyi amaçlayan modellerdir. Araştırmaya konu olan olay, birey ya da nesne, kendi koşulları içinde ve olduğu gibi tanımlanmaya çalışılır. Onları, herhangi bir şekilde değiştirme, etkileme çabası gösterilmez. Önemli olan anlaşılmak istenen durumu uygun bir biçimde belirlemektir (Karasar, 2005). Çalışmada, Bilgisayar Programlama-I dersini alan üniversite öğrencilerinin programlamaya yönelik öz yeterlilik algıları belirlenerek çeşitli değişkenler bakımından incelenmiştir.

Çalışma Grubu

Araştırmanın çalışma grubunu, bir devlet üniversitesinde 2014-2015 öğretim yılı bahar döneminde zorunlu ortak ders havuzundan Bilgisayar Programlama-I dersini seçen Ekonometri ve Makine Mühendisliği bölümü birinci sınıf öğrencileri oluşturmaktadır. Bilgisayar programlama dersi programlamaya giriş niteliğinde olup, algoritmaya giriş, akış şemaları, temel programlama dili komutları, değişkenler, mantıksal ve aritmetiksel işlemler, kontrol ve döngü yapıları, fonksiyon, prosedür ve dizilerle ilgili temel konulardan oluşmaktadır. Programlama dili olarak C kullanılmıştır.

Araştırmanın evreni, Bilgisayar Programlama-I dersi alan 216 öğrencidir. Bu öğrencilerden 144'ü çalışmaya katılmış ve örneklemi oluşturmuştur. Daha önce bilgisayar programlama dersi alan altı öğrenci çalışmadan çıkartılmış, sonuç olarak toplam 138 öğrenci çalışmaya katılmıştır. Öğrencilerin 58'i kadın (% 42,0), 80'i erkektir (% 58,0). Öğrencilerin 74'ü (% 53,6) Ekonometri, 64'ü (% 46,4) Makine Mühendisliği bölümünde öğrenim görmektedir.

Tablo 1. *Demografik istatistikler*

		n	*%*
Cinsiyet			
	Erkek	80	58.0
	Kadın	58	42.0
Bölüm			
	Makine Mühendisliği	64	46.4
	Ekonometri	74	53.6
Toplam		138	100

Veri Toplama Aracı

Programlamaya ilişkin öz yeterlilik ölçeği ilk olarak Ramalingam ve Wiedenbeck (1998) tarafından üniversite öğrencilerinin bilgisayarda programlamaya yapmaya ilişkin algılanan öz yeterliliklerini ölçmek üzere geliştirilmiştir. Orjinal ölçekte toplamda 32 soru yer almaktadır. Çalışmada ise veri toplama aracı olarak Altun ve Mazman (2012) tarafından Türkçeye uyarlanan Programlamaya İlişkin Öz Yeterlik Algısı Ölçeği kullanılmıştır. Ölçeğin Türkçe formunda temel düzeyde ortalama hesaplama, ekrana mesaj yazdırabilme gibi görevleri içeren *Basit Programlama Görevleri (3 madde)* ve hata ayıklama, birden çok dosyada çalışabilme, programı yeniden yazma gibi görevleri *içeren Karmaşık Programlama Görevleri (6 madde)* boyutu altında toplam 9 madde yer almaktadır. Yedili Likert tipine göre maddelenen ölçekte, maddeler '1=kendime hiç güvenmiyorum, 2=genellikle güvenmiyorum, 3=biraz güveniyorum, 4=%50/%50, 5=oldukça güveniyorum, 6=genellikle güveniyorum, 7=tamamen güveniyorum' şeklinde puanlanmıştır. Ölçeğin iç tutarlık katsayısı .86, ölçeğin iki alt boyutunun tutarlık katsayıları ise .91 olarak bulunmuştur. Demografik bilgi formunda cinsiyet, bölüm, programlamaya ilişkin önceden

alınan ders bilgileri sorulmuştur. Ölçek öğrencilere dönemin son haftası uygulanmıştır.

Verilerin Analizi

Araştırmada analizler SPSS 23 (The Statistical Package for the Social Sciences) paket programı kullanılarak gerçekleştirilmiştir. Araştırmanın temel ve alt problemlerine yanıt bulmak amacıyla ortalama, standart sapma ve bağımsız örneklem t-testi analizinden yararlanılmıştır. Analiz sürecine geçilmeden önce verilerin normallik testleri kapsamında dağılımların normallik gösterip göstermediği Kolmogorov-Smirnov normallik testi ile kontrol edilmiştir. Testin sonucunda verilerin normal dağılım gösterdiği tespit edilmiştir (p.>.05). Ayrıca yapılan istatistiksel çözümlemelerde istatistiksel anlamlılık düzeyi .05 olarak alınmıştır.

BULGULAR

Araştırmadan elde edilen veriler, yanıt aranılan sorulara uygun başlıklar altında sırasıyla ele alınmış ve yorumlanmıştır.

Ölçekten alınabilecek en düşük puan 9, en yüksek puan 63 ve ortalama puan 36 olarak hesaplanmıştır. Bu çalışmada öğrencilerin aldığı ortalama puan ($\overline{X}$=25.44) olarak bulunmuştur. Ölçeğin alt boyutları üzerinden ortalama puanlara bakıldığında, Tablo 2'de de görüleceği gibi basit programlama görevleri gerçekleştirme boyutu için alınabilecek ortalama puan $\overline{X}$=12.00'dir. Bu çalışmanın örnekleminin bu boyut için vermiş olduğu yanıtların ortalama puanı ise $\overline{X}$=11.70'tir. Öğrencilerin karışık programlama görevleri gerçekleştirme boyutundan aldığı ortalama puan olan $\overline{X}$=13.74, bu boyuttan alınabilecek ortalama puanın ($\overline{X}$=24) altındadır. Bu bulgular ışığında, ölçekten alınan ortalama puana göre öğrencilerin programlamaya yönelik öz yeterlik algılarının düşük olduğu, basit programlama görevleri gerçekleştirme boyutundan alınan ortalama puana göre orta düzeyde ve karışık programlama görevleri gerçekleştirme boyutundan alınan ortalama puana göre düşük olduğu söylenebilir.

Tablo 2. *Ölçeğe İlişkin Ortalama ve Standart Sapmalar*

	n	*En Düşük*	*En Yüksek*	*Ortalama* ($\overline{X}$)	*Std. Sap.*
Basit Programlama Görevleri	138	3.00	21.00	11.70	6.10
Karışık Programlama Görevleri	138	6.00	38.00	13.74	7.27
Toplam Puan	138	9.00	58.00	25.44	11.43

Bireylerin programlamaya ilişkin öz yeterlik algılarının cinsiyete göre farklılaşıp farklılaşmadığını görmek amacıyla bağımsız örneklem t-testi yapılmıştır. Analiz sonuçlarına göre bireylerin programlamaya ilişkin öz yeterlik algılarının cinsiyete göre farklılık gösterdiği bulunmuştur. Tablo 3'de görüleceği gibi, programlamaya ilişkin öz yeterlik algısında erkek öğrenciler lehine istatistiksel olarak anlamlı bir fark olduğu görülmektedir [t(136)=3,00, p<0,05]. Bu bulguya göre erkek öğrencilerin programlamaya ilişkin öz yeterlik algılarının kız öğrencilere göre daha yüksek olduğu söylenebilir.

Tablo 3. *Programlamaya İlişkin Öz Yeterlik Algısı t-Testi Sonuçları*

Cinsiyet	*n*	$\overline{X}$	*SS*	*Sd*	*t*	*p*
Erkek	80	3,10	1,26	136	3,00	.003*
Kadın	58	2,46	1,19			

*p<0,05 düzeyinde anlamlı bir farklılık vardır.

Öğrencilerin programlamaya ilişkin öz yeterlik algılarının öğrenim gördükleri bölüme göre farklılık gösterip göstermediğini görmek amacıyla bağımsız örneklem t-testi yapılmıştır. t-testi sonuçlarına göre öğrencilerin programlamaya ilişkin öz yeterlik algılarının kayıtlı oldukları bölüme göre farklılık gösterdiği bulunmuştur.

Tablo 4. *Programlamaya İlişkin Öz Yeterlik Algısı t-Testi Sonuçları*

Bölüm	*n*	$\overline{X}$	*SS*	*Sd*	*t*	*p*
Ekonometri	74	2,44	1,23	136	4,06	.000*
Makine Mühendisliği	64	3,27	1,17			

*p<0,05 düzeyinde anlamlı bir farklılık vardır.

Tablo 4 incelendiğinde, programlamaya ilişkin öz yeterlik algısında Makine Mühendisliği Bölümü öğrencileri lehine istatistiksel olarak anlamlı bir fark olduğu görülmektedir [t(136)=4,06, p<0,05]. Bu bulguya göre Makine Mühendisliği bölümünde öğrenim gören öğrencilerin programlamaya ilişkin öz yeterlik algılarının Ekonometri bölümünde öğrenim gören öğrencilere göre daha yüksek olduğu söylenebilir.

TARTIŞMA ve SONUÇ

Araştırmanın sonuçları, öğrencilerin programlamaya ilişkin öz yeterlik algılarının düşük olduğunu göstermektedir. Bununla birlikte ölçeğin iki boyutu açısından inceleme yapıldığında, birinci boyut olan 'basit programlama görevleri gerçekleştirme' açısından algı puan ortalamalarının orta düzeyde olduğu, ikinci boyut olan 'karışık programlama görevleri gerçekleştirme' açısından ise algı puan ortalamasının düşük olduğu saptanmıştır. Bilgisayar programlamaya karşı öz yeterlik algısının düşük olmasının, başarıyı engelleyen faktörlerden biri

olduğu düşünülmektedir (Aşkar & Davenport 2009; Altun & Mazman 2012). Bunun sebeplerinden biri, öğrencilerin bilgisayar programlama eğitiminin kendi kariyerleri için önem taşımadığını düşünmeleri olabilir. Nitekim Aşkar ve Davenport da (2009) çalışma alanı bilgisayar bilimleri olmayan elektronik veya endüstri gibi diğer mühendislik alanlarında eğitim alan öğrencilerin bilgisayar programlamaya ilişkin öz yeterlik algılarının düşüklüğünden bahsetmiştir. Bu çalışmada da, araştırma grubunu oluşturan Ekonometri ve Makine Mühendisliği Bölümü öğrencilerinin aldıkları programlama dersini kendi alanlarıyla bağlantılı görmemesinden dolayı derse ve konuya karşı ilgisizlik söz konusu olabilir. Diğer bir neden, öğrencilerin programlamaya yönelik problem çözme veya eleştirel düşünme, yansıtıcı düşünme gibi üst düzey düşünme becerilerinin düşük olabileceğidir.

Cinsiyet açısından bakıldığında, erkek öğrencilerin programlamaya ilişkin öz yeterlik algılarının kız öğrencilere göre daha yüksek olduğu bulunmuştur. Alanyazında, bu sonuca paralel olarak, programlamaya ilişkin öz yeterlik algısının cinsiyete göre anlamlı farklılık gösterdiği ve erkeklerde öz yeterlik algısının daha yüksek olduğunu söyleyen çalışmalar bulunmaktadır (Aşkar & Davenport 2009; Alvarado, Dodds, & Libeskind-Hadas 2012; Başer 2013; Carter & Jenkins 1999; Jegede 2009; Korkmaz & Altun 2013). Dolayısıyla, kadın ve erkek öğrencilerinin programlama algılarının farklı olduğu söylenebilir (Rubio vd., 2015). Araştırmanın bu bulgusuna karşıt olarak, alanyazında her ne kadar erkeklerin öz yeterlilik algıları kadınlara göre yüksek olarak bulunsa da anlamlı farklılık göstermediği sonucuna varan çalışmalar da bulunmaktadır (Altun & Mazman 2012; Ramalingam & Wiedenbeck 1998).

Çalışma alanı açısından bakıldığında Makine Mühendisliği Bölümü'nde öğrenim gören öğrencilerin programlamaya ilişkin öz yeterlik algısının Ekonometri Bölümü'nde öğrenim gören öğrencilere göre daha yüksek düzeyde olduğu bulunmuştur. Fatin ve diğerleri (2010), mühendislik eğitimi alan öğrenciler için kıyaslama yapabilme, çıkarımda bulunma, problem çözme, analitik ve eleştirel düşünmenin oldukça önemli bir yere sahip olduğunu belirtmektedir.

Programlama deneyiminin de öz yeterlik algısı açısından önemli olduğuna dair çalışmalar vardır. Daha önce programlama dersi alan ya da bir program geliştirmiş olan öğrencilerin akademik başarısı ve derse ilişkin öz yeterliğinin yüksek olduğu ifade edilmektedir (Barchino vd., 2012; Davidson, Larzon & Ljunggren 2010; Doyle, Stamouli & Huggard 2005; Mazman & Altun 2013). Bundan yola çıkarak, bugün Türkiye'de tartışılan ve üzerinde çalışılan ortaokul ve lise müfredatlarına programlama dersi eklenmesi doğru bir yaklaşım olacaktır (Işık 2016). Bilgisayar bilimleri, İngiltere ve Yeni Zelanda gibi birçok ülkede ilköğretim müfredatında yer almaktadır, programlamaya ilişkin çalışmalar giderek yoğunlaşmaktadır (Bell, Andreae & Robins 2014; Brown vd., 2014). Bu müfredatlı kurslarda Scratch (Resnick vd., 2009), App Inventor (Wolber vd., 2011) ve Light-Bot (Light-Bot 2015) gibi öğrencilerin sürekli gelişimini de gösteren çıktılar veren başarılı uygulamalar bulunmaktadır (Goadrich

2014, Gouws, Bradshaw, & Wentworth 2013; Guzdial vd., 2014). Önceki programlama deneyiminin öz yeterlik algısı üzerindeki etkisi düşünüldüğünde, özellikle de bilgisayar bilimleri haricindeki çalışma alanlarında öğrenim gören öğrencilerin önceki deneyimleri bu derslerdeki akademik başarıyı da olumlu etkileyeceği düşünülmektedir.

Yükseköğretim kurumlarında bilgisayar programlama dersleri, genellikle tek dönemlik dersler olarak verilmektedir. Dersin iki döneme genişletilerek ilk dönemin problem çözme, algoritma oluşturma ve temel program komutlarını, ikinci dönemin ise bu temel üzerine yapılandırılarak daha karmaşık programlama yapılarının içeriğe dâhil edilmesi ve sınıf içi projelerle öğrencilerin öğrendiklerini pekiştirmelerinin sağlanması gerekmektedir. Müfredat, öğrencinin kendi alanı ile ilgili problemlerde algoritmik çözümler üretebilmesini desteklemelidir. Bilgisayar programlama mühendislik bölümlerinin yanı sıra Ekonometri gibi ders müfredatları matematik ve istatistiğe dayalı bölümlere devam eden öğrencilerin diğer derslerinde karşılaşacakları problemlerin çözümüne daha analitik yaklaşmalarına ve hatta problem çözümlerini bilgisayar programlama ile daha hızlı ve hatasız yapabilmelerine yardımcı olacaktır. Ders saatinin arttırılması, öğrencilerin programlama deneyimlerini zenginleştirecektir.

Mazman (2013) tarafından da belirtildiği üzere, alanyazında bilgisayar programlamaya ait başarıyı etkileyen çalışma belleği ve uzamsal beceriler gibi bilişsel ve öz yeterlilik algısı, deneyim ve cinsiyet gibi bilişsel olmayan faktörlerden bahsedilmektedir (Aşkar & Davenport 2009; Bergersen & Gustafsson 2011; Byrne & Lyons 2001; Erdoğan 2005; Jones & Burnett 2008; Mancy & Reid 2004; Ramalingam vd., 2004). Özellikle de bilgisayar bilimleri haricindeki çalışma alanlarında bilgisayar programlamaya ilişkin öz yeterlik algıları üzerinde çalışmaların devam etmesi önemlidir. Yanı sıra, hem tutum hem de önceki programlama deneyimine ek olarak öğrencilerin temel düzeyde programlama bilgisi içeren bu derslerde problem çözme yeteneklerinin ve yansıtıcı düşünme, algoritmik düşünme gibi üst düzey beceriler ile arasındaki ilişkilerin araştırılması da faydalı olacaktır. Ayrıca, bilgisayar programlama öğretiminde kullanılan yöntem ve tekniklerin etkinliğinin ölçülmesi, nitel araştırma yöntemleri kullanılarak duruma dair elde edilen nicel bulguların desteklenmesi ve derinlemesine bilgi sahibi olunması da önemlidir. Bilgisayar programlama ile ilgili yapılacak tüm çalışmalar, günümüzde son derece önemli bir yere sahip bu becerinin en etkin şekilde kazanılmasını sağlamaya ışık tutacaktır.

KAYNAKLAR

Akkoyunlu B. ve Orhan F. (2003). Bilgisayar ve Öğretim Teknolojileri Eğitimi (BÖTE) Bölümü Öğrencilerinin Bilgisayar Kullanma Öz Yeterlik İnancı ile Demografik Özellikleri Arasındaki İlişki. *The Turkish Online Journal of Educational Technology*, 2(3), 86-93.

Altun, A. ve Mazman, S.G. (2012). Programlamaya ilişkin öz yeterlik algısı ölçeğinin Türkçe formumun geçerlilik ve güvenirlik çalışması. *Eğitimde ve Psikolojide Ölçme ve Değerlendirme Dergisi*, 3(2), 297- 308.

Alvarado, C., Dodds, Z., & Libeskind-Hadas, R. (2012). Increasing women's participation in computing at Harvey Mudd College. *ACM Inroads*, 3(4), 55-64.

Ambrosio, A. P., Costa, F. M., Almeida, L., Franco, A., & Macedo, J. (2011,October). Identifying cognitive abilities to improve CS1 outcome. Paper presented by Frontiers in Education Conference (FIE)., Rapid City, South Dakota.

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), 26-32.

Balanskat, A. & Engelhardt, K. (2014). Computing our Future: Computer programming and coding - Priorities, school curricula, and initiatives across Europe. European Schoolnet http://www.eun.org/c/document_library/get_ file?uuid=521cb928-6ec4-4a86-b522-9d8fd5cf60ce&groupId=43887

Bandura, A. (1977). Self-efficacy: Toward a unifying theory of behavioural change. *Psychological Review*. 84(2), 191-215.

Barchino, R., Gutiérrez, J. M., De-Marcos, L., Martínez, J. J., Jiménez, L., Otón, S., Hilera, J. R. Et al. (2012). Experiences in the use of Mobile Games to improve Programming Skills in Computer Engineering. *International Journal of Innovative Computing, Information and Control*, 8(2), 1167-1174.

Başer, M. (2013). Developing attitude scale toward computer programming. *International Journal of Social Science*, 6(6), 199-215.

Bell, T., Andreae, P., & Robins, A. (2014). A case study of the introduction of computer science in NZ schools. *Transactions on Computing Education (TOCE)*, 14(2), 10.

Bergersen, G. R., & Gustafsson, J.-E. (2011). Programming Skill, Knowledge, and Working Memory Among Professional Software Developers from an Investment Theory Perspective. *Journal of Individual Differences, 32*(4), 201-209. doi: 10.1027/1614-0001/a000052

Brown, N. C., Sentance, S., Crick, T., & Humphreys, S. (2014). Restart: The resurgence of computer science in UK schools. *Transactions on Computing Education (TOCE)*, 14(2), 9.

Byrne, P., & Lyons, G. (2001). The Effect of Student Attributes on Success in Programming. *SIGCSE Bulletin*, 33(3), 49-52.

Carter, J., & Jenkins, T. (1999). Gender and programming: What's going on? *SIGCSE Bulletin*, 31(3), 1-4.

Davidson, K., Larzon, L. & Ljunggren, K. (2010). *Self-Efficacy in Programming among STS Students.* Technical Reports from Computer Science Education course of Upssala University.http://www.it.uu.se/edu/course/homepage/datadidaktik/ht10/reports adresinden 11.02.2016 tarihinde erişilmiştir.

Doyle, E., Stamouli, I., & Huggard, M. (2005). Computer anxiety, self-efficacy, computer experience: An investigation throughout a computer science degree. Proceedings of the 35th Annual Frontiers in Education Conference , FIE'05. S2H-3

Erdoğan, B. (2005). *Programlama başarısı ile akademik basarı, genel yetenek, bilgisayara karsı tutum, cinsiyet ve lise türü arasındaki ilişkilerin incelenmesi*. Yüksek lisans, Marmara Üniversitesi İstanbul.

Evans, G. E., & Mark G. S. (1989). *What best predicts computer proficiency?*. Communications of the ACM, 32(11), 1322-1327.

Fang, X. (2012). Application of the participatory method to the computer fundamentals course, Affective Computing and Intelligent Interaction. *Advances in Intelligent and Soft Computing*, 137, 185-189.

Fatin, A.P., Mohamad, B.A., Bakar, M.N., Noor,F.A.R., Lilia, E.M., Normah, M.G. (2010,April). *Engineering elements profile among first- and final-year engineering students in Malaysia*. Paper presented at IEEE Global Engineering Education Conference (EDUCON) , Amman, Jordan.

Feldgen, M., & Clua, O.(2004). Games as a Motivation for Freshman to Learn Programming. *Proceedings of the 34th ASEE/IEEE Frontiers in Education Conference*, 3, S1H/11–S1H/16

Goadrich, M. (2014). Incorporating tangible computing devices into CS1.*Journal of Computing Sciences in Colleges*, 29(5), 23-31.

Gomes, A. & Mendes, A. (2007). *Learning to program – difficulties and solutions*. Paper presented at International Conference on Education – ICEE 2007, Coimbra, Portugal.

Gouws, L. A., Bradshaw, K., & Wentworth, P. (2013, July). Computational thinking in educational activities: an evaluation of the educational game light-bot. Proceedings of the 18th ACM Conference On Innovation And Technology İn Computer Science Education ,10-15.

Guzdial, M., Ericson, B., Mcklin, T., & Engelman, S. (2014). Georgia computes! An intervention in a US state, with formal and informal education in a policy context. *ACM Transactions on Computing Education (TOCE)*, 14(2), 13.

Hawi, N. (2010). Causal attributions of success and failure made by undergraduate students in an introductory-level computer programming course. *Computers & Education.* 54, 1127–1136.

Hernane B., P., Gilney F., Z., & Marcelo A., M.(2010). Learning computer programming: Implementing a fractal in a Turing machine. *Computers & Education.* 55,767-776

Horzum, M. B., & Çakır, Ö. (2009). The validity and reliability study of the Turkish version of the online technologies self-efficacy scale. *Educational Sciences: Theory & Practice*, 9(3), 1343-1356.

Işık F. (2016, Ocak 17). Milli Eğitim Bakanlığımızla yapacağımız çalışma ile Kodlama dersini Ortaokul ve Liselerde Müfredata alacağız. https://twitter.com/TCSanayi/status/688646739834843136 'dan erişildi.

Jegede, P. O. (2009). Predictors of java programming self-efficacy among engineering students in a Nigerian university. *International Journal of Computer Science and Information Security*, 4(1&2).

Jenkins, T. (2002). *On the difficulty of learning to program.* Paper presented at 3rd annual Conference of LTSN-ICS, Loughbrorough University, Leicestershire, UK

Jones, S. J., & Burnett, G. (2008). Spatial Ability and Learning to Program. *Human Technology, 4*(1), 47-61.

Karasar, N. (2005). *Bilimsel Araştırma Yöntemi*, Ankara: Nobel Kitabevi.

Korkmaz, Ö. (2012). The impact of critical thinking and logical-mathematical intelligence on algorithmic design skills. *Journal of Educational Computing Research*. 46(2), 173-193.

Korkmaz, Ö. (2013). Prospective CITE teachers' self-efficacy perceptions on programming. Proceeding of the 2nd World Conference on Educational Technology Researches (WCETR-2012). 83, 639-643

Korkmaz, Ö., & Altun, H. (2013). Mühendislik ve BÖTE Öğrencilerinin Bilgisayar Programlama Öğrenmeye Dönük Tutumları. *International Journal of Social Science*, 6(2), 1169-1185.

Korkmaz, Ö., & Altun, H. (2014). Adapting Computer Programming Self-Efficacy Scale and Engineering Students' Self-Efficacy Perceptions. *Participatory Educational Research (PER)*, 1(1), 20-31.

Lahtinen, E., Ala-Mutka, K., & Järvinen, H.M. (2005). *A study of the difficulties of novice programmers*. in inroads - ACM SIGCSE Bulletin, 37, 14–18. ACM Press.

Lau, W. W. F., & Yuen, A. H. K. (2009). Exploring the effects of gender and learning styles on computer programming performance: implications for programming pedagogy. *British Journal of Educational Technology*, 40(4), 696-712.

Light-Bot (2015), Hour of code, https://lightbot.com/hour-of-code-2015.html adresinden 11.02.2016 tarihinde alınmıştır.

Mancy, R., & Reid, N. (2004). *Aspects of Cognitive Style and Programming*. Paper presented at the 16 th Workshop of the Psychology of Programming Interest Group (PPIG 16). Carlow, Ireland.

Mazman, S.G. (2013). Programlama Performansını Etkileyen Faktörlerin Bilişsel Tabanlı Bireysel Farklılıklar Temelinde Modellenmesi. Doktora Tezi, Hacettepe Üniversitesi, Ankara.

Mazman, S.G., ve Altun, A. (2013). Programlama–I Dersinin BÖTE Bölümü Öğrencilerinin Programlamaya İlişkin Öz Yeterlik Algıları Üzerine Etkisi. *Journal of Instructional Technologies & Teacher Education*, 2(3),24-29

Milne, I., & Rowe, G. (2002). Difficulties in learning and teaching programming views of students and tutors. *Education and Information Technologies*. 7(1), 55-66

Özyurt, Ö., & Özyurt, H. (2015). Bilgisayar programcılığı öğrencilerinin programlamaya karşı tutum ve programlama öz-yeterliklerinin belirlenmesine yönelik bir çalı. *Eğitimde Kuram ve Uygulama.* 11(1), 51-67.

Pereira, H. B. D. B., Zebende, G. F., & Moret, M. A. (2010). Learning computer programming: Implementing a fractal in a Turing Machine. *Computers & Education*, 55(2), 767-776.

Pillay N., & Jugoo V. R. (2005). *An Investigation into Student Characteristics Affecting Novice Programming Performance*, in inroads - ACM SIGCSE Bulletin. 37(4), 107-110, ACM Press.

Ramalingam, V. & Wiedenbeck, S. (1998). Development and validation of scores on a computer programming self-efficacy scale and group analysis of novice programmer self-efficacy. *Journal of Educational Computing Research*, 19(4), 367-381.

Ramalingam, V., LaBelle, D., & Wiedenbeck, S. (2004, June). Self-efficacy and mental models in learning to program. *SIGCSE Bulletin 36*(3), 171-175.

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., ... & Kafai, Y. (2009). Scratch: programming for all.*Communications of the ACM*, 52(11), 60-67.

Robins, A. Rountree, J., & Rountree, N. (2003). Learning and Teaching Programming: A Review and Discussion. *Computer Science Education*, 13(2), 137–172.

Rubio, M. A., Romero-Zaliz, R., Mañoso, C., & Angel, P. (2015). Closing the gender gap in an introductory programming course. *Computers & Education*,82, 409-420.

Saeli, M.,Perrenet, J., Jochems, W. M. G. ve Zwaneveld, B. (2011). Teaching programming in secondary school: A pedagogical content knowledge perspective.*Informatics in Education*, 10(1), 73-88.

Sebetci, Ö., & Aksu, G. (2014). Öğrencilerin mantıksal ve analitik düşünme becerilerinin programlama dilleri başarısına etkisi. *Journal of Educational Sciences & practices*, 13(25).

Sue J., & Gary B. (2008). Spatial Ability and Learning to Program. *Interdisciplinary Journal on Humans in ICT Environments* ,4 (1), 47–61.

Wiedenbeck S., LaBelle D., & Kain V. N.R. (2004). Factors Affecting Course Outcomes in Introductory Programming. *16th Annual Workshop of the Psychology of Programming Interest Group (PPIG16)*, 97-109.

Wolber, D., Abelson, H., Spertus, E., & Looney, L. (2011). *App Inventor*. O'Reilly Media, Inc.

Tan, P. H., Ting, C. Y., & Ling, S. W. (2009). *Learning difficulties in programming courses: Undergraduates' perspective and perception*. International Conference on Computer Technology and Development, Kota Kinabalu, Malaysia.

Wang, Y., Liu, H., Feng, Y., Jiang, Y., & Liu, Y. (2012). Assessment of programming language learning based on peer code review model: Implementation and experience report. *Computers & Education*, 59,412–422.

Zimmerman, B. J. (2000). Self-efficacy: An essential motive to learn. *Contemporary Educational Psychology*, 25(1), 82-91.

SUMMARY

Computer programming has come to the fore of the sector due to unprecedented pace of development and change in information and communication technologies. Besides computer and software engineering, other engineering programmes and the programmes based on mathematics and natural sciences also give significant importance to programming. Programming requires higher order thinking skills like problem solving, logical or computational thinking. There are studies in the literature indicating certain problems encountered in teaching computer programming in terms of academic achievement and student satisfaction particularly for programmes other than computer sciences caused by lack of such skills, use of incorrect teaching methods in programming classes or negative attitudes and low level of perceived self-efficacy on students' side. Based on limited number of studies on perceived computer programming self-efficacy of undergraduate students of programmes other than computer sciences, this study aims to investigate perceived computer programming self-efficacy of undergraduate students of mechanical engineering and econometrics at a Turkish state university. Students' perceived self-efficacy is investigated in terms of basic and complex programming tasks, gender, and programme.

This is a descriptive survey study aiming at investigating possible reasons for low academic performance of in a computer programming class. Participants of the study is 138 first year undergraduate students attending mechanical engineering and econometrics programmes at a Turkish state university. Of participants, 58 (42.0%) are female and 80 (58.0%) are male; 74 (53.6%) enrolled in Econometrics, and 64 (46.4%) in Mechanical Engineering. A Programming Self-Efficacy Scale, adapted by Altun and Mazman (2012), was used for data collection. The instrument included nine 7-point Likert questions under two dimensions for simple and complex programming tasks. Data were analysed using SPSS 23 using descriptive statistics and independent sample t-test.

The study indicates a low level of perceived self-efficacy of participants for computer programming ($\underline{X}$=25.44 out of $\underline{X}$=36). An examination of sub-dimensions shows that students' perceived self-efficacy is at medium level for simple programming tasks, and at low level for complex programing tasks. The study points out a significant difference in terms of gender where male students' perceived self-efficacy for programming is higher than female students. The study also points a statistically significant difference on behalf of mechanical engineering students compared to others for perceived self-efficacy for programming.

Findings from the study shows a low level of general perceived self-efficacy for computer programming, while a medium level and low level for basic and complex programming tasks respectively. There are studies connecting low self-efficacy level with low academic performance (Aşkar & Davenport 2009; Altun

& Mazman 2012) due to several reasons. This is particularly noteworthy for students of programmes other than computer sciences since they may consider computer programming as not an essential subject for their career (e.g. Aşkar & Davenport 2009). This may also be the case for the participants of this very study. Another reasons may be students' lack of higher order thinking skills required for computer programming.

From a gender perspective, findings indicate a higher self-efficacy level for male students. There are other studies in the literature with parallel findings where male students have higher levels of perceived self-efficacy for programming (e.g. Aşkar & Davenport 2009; Alvarado, Dodds, & Libeskind-Hadas 2012; Başer 2013; Carter & Jenkins 1999; Jegede 2009; Korkmaz & Altun 2013). Yet, there are also studies indicating a no significant difference for perceived self-efficacy for programming for gender (Altun & Mazman 2012; Ramalingam & Wiedenbeck 1998).

This study also indicates a high level of perceived self-efficacy for engineering students, regarding the area of study. Engineering students are supposed to be equipped with higher order thinking skills such as problem solving, analytical thinking or critical thinking (Fatin et al. 2010). There are also studies on the importance of prior programming experience on perceived self-efficacy, where students having completed a programming course or having developed a programme previously tend to have a higher level of perceived self-efficacy and higher academic achievement (e.g. Barchino et al. 2012; Davidson, Larzon & Ljunggren 2010; Doyle, Stamouli & Huggard 2005; Mazman & Altun 2013). Hence, it would be useful to integrate programming courses into secondary education curriculum as being discussed for initiation in Turkey very recently (Işık 2016).

Computer programming courses at higher education institutions are generally designed as single-term courses. It is suggested to extend these courses to two terms to include basic problem solving, creating algorithms, and learning basic programming commands in the first term followed by more complex programming tasks and intensive in-class activities. Students should also be provided with opportunities to apply acquired skills in other courses. Such increase in class hours will also enrich their programming experience.

Studies in the literature mentions several factors influencing academic performance in computer programming. It is important to continue to advance research studies on perceived self-efficacy for computer programming, particularly for areas of study other than computer sciences. It is also recommended to conduct studies investigating relationship between self-efficacy, attitude, prior programming experience, and ownership of higher order thinking skills. Examination of teaching methods and techniques used in computer programming classes is also to be considered in future studies. Use of qualitative research will provide opportunities to complement quantitative data,

and provide an in-depth perspective. Studies on computer programming will shed light and contribute to efforts for acquiring this very important skill of 21st century in the most efficient manner.