



ONDOKUZ MAYIS ÜNİVERSİTESİ

EĞİTİM BİLİMLERİ ENSTİTÜSÜ

BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ EĞİTİMİ

ANABİLİM DALI

**PROGRAMLAMA ÖĞRETİMİNDE BİLGİSAYARSIZ
BİLGİSAYAR BİLİMİ ETKİNLİKLERİNİN ÖĞRENCİLERİN
AKADEMİK BAŞARILARINA ETKİSİ**

Ercan ÇİMŞİR

Danışman

Dr. Öğr. Üyesi Polat ŞENDURUR

YÜKSEK LİSANS TEZİ

Haziran, 2019

TELİF HAKKI

2547 Sayılı Yükseköğretim Kanunu Ek Madde 40 hükümleri çerçevesinde (Ek:22/2/2018-7100/10 md.) “*Lisansüstü tezler yetkili kurum ve kuruluşlar tarafından gizlilik kararı alınmadıkça, bilime katkı sağlamak amacıyla Yükseköğretim Kurulu Ulusal Tez Merkezi tarafından elektronik ortamda erişime açılır.*”

Araştırmacılar tezlerin tamamı veya bir bölümünü yazarın izni olmadan ticari veya mali kazanç amaçlı kullanamaz, yayınlamayaz, dağıtamaz ve kopyalayamaz. Ulusal Tez Merkezi Web Sayfasını kullanan araştırmacılar, tezlerden bilimsel etik ve atıf kuralları çerçevesinde yararlanırlar.

YAZARIN

Adı : Ercan

Soyadı : ÇİMŞİR

Bölümü : Bilgisayar ve Öğretim Teknolojileri Eğitimi Anabilim Dalı

İmza :

Teslim Tarihi :

TEZİN

Türkçe Adı : Programlama Öğretiminde Bilgisayarsız Bilgisayar Bilimi Etkinliklerinin Öğrencilerin Akademik Başarılarına Etkisi

İngilizce Adı : Effect of Computer Science Unplugged Activities on Students' Academic Achievements in Programming Teaching

ETİK İLKELERE UYGUNLUK BEYANI

Tez yazma sürecinde bilimsel ve etik ilkelere uyduđumu, yararlandıđım tüm kaynakları kaynak gösterme ilkelerine uygun olarak kaynakçada belirttiđimi ve bu bölümler dışındaki tüm ifadelerin şahsıma ait olduđunu beyan ederim.

Yazar Adı Soyadı:

Ercan ÇİMŞİR

İmza:

.....

KABUL VE ONAY

Ercan ÇİMSİR tarafından hazırlanan “**Programlama Öğretiminde Bilgisayarsız Bilgisayar Bilimi Etkinliklerinin Öğrencilerin Akademik Başarılarına Etkisi**” adlı tez çalışması aşağıdaki jüri tarafından oy birliği ile Ondokuz Mayıs Üniversitesi **Bilgisayar ve Öğretim Teknolojileri Eğitimi** Anabilim Dalı’nda Yüksek Lisans tezi olarak kabul edilmiştir.

Danışman: Dr. Öğr. Üyesi Polat ŞENDURUR

Bilgisayar ve Öğretim Teknolojileri Eğitimi, Ondokuz Mayıs Üniversitesi

Başkan: Dr. Öğr. Üyesi Polat ŞENDURUR

Bilgisayar ve Öğretim Teknolojileri Eğitimi, Ondokuz Mayıs Üniversitesi

Üye: Dr. Öğr. Üyesi Hatice Gökçe BİLGİÇ DOĞAN

Bilgisayar ve Öğretim Teknolojileri Eğitimi, Ondokuz Mayıs Üniversitesi

Üye: Dr. Öğr. Üyesi Neşet MUTLU

Okul Öncesi Eğitimi, Erciyes Üniversitesi

Bu tezin **Bilgisayar ve Öğretim Teknolojileri Eğitimi** Anabilim Dalı’nda Yüksek Lisans tezi olması için şartları yerine getirdiğini onaylıyorum.

Tarihi: __/__/__

Prof. Dr. Ali ERASLAN

Eğitim Bilimleri Enstitüsü Müdürü

(İmza ve Mühür)



“Sevgili Eşime”

TEŞEKKÜRLER

Araştırma sürecimin her bölümünde bana bilgisi ve tecrübesiyle yol gösteren saygıdeğer hocam ve danışmanım Dr. Öğr. Üyesi Polat ŞENDURUR'a vermiş olduğu bu desteklerinden dolayı sonsuz teşekkür ederim.

Lisans ve lisansüstü eğitimim boyunca bilgi, görüş ve tavsiyeleriyle bana yardımcı olan hocam, Sayın Prof. Dr. Ufuk ÇORUH'a teşekkürlerimi sunarım. Lisansüstü eğitimim esnasında yaşadığım birtakım sıkıntılara çare olmak için, öğle aralarını feda ederek değerli zamanını bana ayıran saygıdeğer hocam Doç. Dr. Özgür TAŞKIN'a teşekkürü borç bilirim.

Derslerine katılmaktan keyif aldığım ve bana çok şey kattıklarına inandığım Sayın hocalarım Dr. Öğr. Üyesi Hatice Gökçe BİLGİÇ DOĞAN ve Dr. Öğr. Üyesi Emine ŞENDURUR'a teşekkürlerimi sunarım.

Araştırmamın çalışma grubunu oluşturan, aynı zamanda öğretim görevlisi olarak hocaları olmaktan gurur duyduğum, çok sevgili Giresun Üniversitesi Bilgisayar Teknolojileri Bölümü öğrencilerine teşekkür ederim.

Son olarak bana bu yolda inanan ve teşvik eden, sadece araştırma sürecinde değil hayatımın her anında yanımda olan sevgili eşim Burcu TURAN ÇİMŞİR'e sonsuz teşekkürlerimi sunarım.

**PROGRAMLAMA ÖĞRETİMİNDE BİLGİSAYARSIZ
BİLGİSAYAR BİLİMİ ETKİNLİKLERİNİN ÖĞRENCİLERİN
AKADEMİK BAŞARILARINA ETKİSİ**

Yüksek Lisans Tezi

Ercan ÇİMŞİR

ONDOKUZ MAYIS ÜNİVERSİTESİ

EĞİTİM BİLİMLERİ ENSTİTÜSÜ

Haziran, 2019

ÖZ

Bilgisayarsız bilgisayar bilimi (B³), bilgisayar kullanmadan bilgisayar bilimlerinin temelini öğrencilere kazandırmayı hedefleyen etkinliklerle eğitimci ve araştırmacıların dikkatini çekmeye başlamıştır. Bu araştırmanın amacı programlama öğretiminde B³ etkinliklerinin üniversite öğrencilerinin akademik başarılarına etkisini araştırmaktır. Ayrıca öğrenci görüşleri ile akademik başarı arasındaki etkinin değerlendirilmesidir. Araştırmada karma araştırma desenlerinden sıralı açıklayıcı desen kullanılmıştır. Araştırmanın nicel bölümü deneysel bir çalışma olup rastgele atamalı eşleştirilmiş öntest ve sontest kontrol gruplu araştırma modeline göre gerçekleştirilmiştir. Araştırmanın örneklemini Giresun Üniversitesi Bilgisayar Teknolojileri Bölümünde öğrenim gören araştırmaya katılmaya gönüllü 54 öğrenci oluşturmaktadır. Nicel verilerin toplanmasında araştırmacı tarafından geçerlik güvenilirlik çalışması yapılmış olan akademik başarı testi kullanılmıştır. Nitel verilerinin toplanmasında ise programlama öğretiminde B³ etkinliklerinin kullanılmasına dair hazırlanan görüş formundan yararlanılmıştır. Araştırmada öncelikle tüm katılımcıların programlama bilgisini ölçmek amacıyla akademik başarı testi uygulanmıştır. Elde edilen puanlar yukarıdan aşağıya sıralanmış ve öğrenciler başarı puanları dikkate alınarak eşleştirilmiştir. Daha sonra eşleştirilen ikililerden her biri deney ve kontrol gruplarına rastgele atanmıştır. Bu işlemin sonunda 27 öğrenci deney grubunda, 27 öğrenci de kontrol grubunda yer almıştır. Altı hafta süren araştırmada, deney grubunda yer alan öğrenciler programlama öğretiminde B³ etkinliklerine katılmıştır. Kontrol grubu öğrencileri ise aynı konuyla ilgili geleneksel yöntemle programlama öğretimine tabi tutulmuştur. Sonrasında her iki gruptan akademik başarı testi ile nicel veriler toplanmıştır. Ayrıca deney grubundaki öğrencilerden görüş formu ile nitel veriler elde edilmiştir. Nicel verilerin analizi sonucunda öğrencilerin programlama akademik başarılarında deney grubu lehine anlamlı fark olduğu görülmüştür. Nitel verilerin

analizi sonucunda da, B³ etkinliklerinin öğrencilerin birbirleri ve öğretmenleriyle olan etkileşimlerine, uzun süreli hatırlama sürelerine ve motivasyonlarına olumlu katkı sağladığı söylenebilir.

Anahtar Kelimeler : Bilgisayarsız bilgisayar bilimi, programlama, programlama öğretimi

Sayfa Sayısı : 78

Danışman : Dr. Öğr. Üyesi Polat ŞENDURUR



**EFFECT OF COMPUTER SCIENCE UNPLUGGED ACTIVITIES
ON STUDENTS' ACADEMIC ACHIEVEMENTS IN
PROGRAMMING TEACHING**

MS Thesis

Ercan ÇİMŞİR

ONDOKUZ MAYIS UNIVERSITY

GRADUATE SCHOOL OF EDUCATIONAL SCIENCES

June, 2019

ABSTRACT

Computer Science Unplugged (CS Unplugged) have begun to draw the attention of educators and researchers with activities aiming at providing students with the basis of computer science without using computers. The aim of this study is to investigate the effects of CS Unplugged activities on academic achievement of the undergraduate students in teaching programming course, and also to evaluate the relationship between the students' ideas and academic achievement. In this research, explanatory sequential design that is one of the mixed research designs was used. The quantitative part of the study was an experimental study, and it was carried out based on the research model including randomly assigned pretest and posttest control group. The sample of the study consisted of 54 volunteer students studying in the Department of Computer Technologies at Giresun University. In order to collect the quantitative data, researchers used the academic achievement test, and the validity and reliability analyses of this test were conducted by the researcher. The opinion form prepared on the use of CS Unplugged activities in teaching programming was used to collect the qualitative data. Firstly, academic achievement test was applied to measure the programming knowledge of all participants. The scores were ranked from top to bottom and the students were matched regarding their achievement scores. Each pair was then randomly assigned to the experimental and control groups. At the end of this process, there were 27 students in the experimental group and 27 students in the control group. During six weeks, the students in the experimental group participated in CS Unplugged activities in teaching programming. The students in the control group were subjected to the traditional method of teaching programming on the same subject. Afterwards, the quantitative data from both groups was collected with the academic achievement test. Furthermore, the qualitative data were obtained from the experimental group with opinion forms. As a result of the quantitative data analysis, it

was seen that there was a significant difference in the academic achievement of the students in favor of the experimental group. In addition to the quantitative part of the research, qualitative data analysis indicated that CS Unplugged activities contribute positively to students' interactions with each other and with their teachers, their long-term memory and motivation.

Key Words : Computer science unplugged, programming, programming teaching

Number of Pages : 78

Advisor : Assist. Prof. Dr. Polat ŞENDURUR



İÇİNDEKİLER

TELİF HAKKI.....	II
ETİK İLKELERE UYGUNLUK BEYANI.....	III
KABUL VE ONAY	IV
TEŞEKKÜRLER	VI
ÖZ.....	VII
ABSTRACT	IX
İÇİNDEKİLER	XI
TABLolar LİSTESİ.....	XIII
ŞEKİLLER LİSTESİ.....	XIV
BİRİNCİ BÖLÜM.....	1
I. GİRİŞ	1
1.1 Problem Durumu.....	1
1.2 Araştırmanın Amacı ve Araştırma Soruları.....	4
1.3 Araştırmanın Önemi	5
1.4 Varsayımlar	6
1.5 Araştırmanın Sınırlılıkları.....	6
1.6 Tanımlar	7
İKİNCİ BÖLÜM	8
II. KURAMSAL ÇERÇEVE.....	8
2.1 Programlama Tanımı.....	8
2.2 Programlama Öğretimi.....	10
2.2.1 Türkiye’de ve Dünya’da Programlama Öğretimi	12
2.2.2 Programlama Öğretiminde Kullanılan Yaklaşımlar	15
2.3 Bilgisayarsız Bilgisayar Bilimi (B ³).....	16
ÜÇÜNCÜ BÖLÜM	22
III. YÖNTEM.....	22
3.1 Araştırmanın Deseni	22
3.2 Çalışma Grubu	23
3.3 Veri Toplama Araçları.....	26
3.3.1 Akademik Başarı Testi.....	26
3.3.2 Görüş Formu	28
3.4 Veri Analizi	29

3.4.1 Nicel Veri Analizi	29
3.4.2 Nitel Veri Analizi	30
3.5 Süreç	30
3.5.1 Kontrol Grubunda Uygulanan Geleneksel Yöntem	30
3.5.2 Deney Grubunda Uygulanan B ³ Etkinlikleri	30
DÖRDÜNCÜ BÖLÜM	36
IV. BULGULAR	36
4.1 Nicel Verilere Ait Bulgular	36
4.1.1 Tanımlayıcı İstatistikî Veriler	36
4.1.2 Karma Varyans Analizi Varsayımları.....	37
4.1.3 Karma Varyans Analizi Sonuçları.....	41
4.2 Nitel Verilere Ait Bulgular	42
4.2.1 B ³ Etkinliklerinin Olumlu Katkıları	42
4.2.2 B ³ Etkinliklerinde Sınıf Yapısı	47
4.2.3 B ³ 'e Yönelik Öğrenci Önerileri	48
4.2.4 B ³ Etkinliklerine Ait Olumsuz Görüşler.....	50
BEŞİNCİ BÖLÜM	51
V. SONUÇ, TARTIŞMA ve ÖNERİLER.....	51
5.1 Sonuç ve Tartışma	51
5.2 Öneriler	54
KAYNAKÇA	57
EKLER.....	64

TABLÖLAR LİSTESİ

Tablo 1: Programlama Öğretiminin Ülkelerin Okul Seviyelerine Göre Entegrasyon Durumu	14
Tablo 2: Araştırma Deseni	23
Tablo 3: Çalışma Grubuna Ait Cinsiyet Bilgileri	24
Tablo 4: Çalışma Grubuna Ait Programlama Deneyimi Bilgileri	24
Tablo 5: Deney Grubu Bilgileri	25
Tablo 6: Akademik Başarı Testi (Öntest-Sontest) Madde Ayırt Edicilik İndeksi	28
Tablo 7: Tanımlayıcı İstatistik	37
Tablo 8: Normallik Testi	40
Tablo 9: Levene'nin Hata Varyanslarının Eşitliği Testi	40
Tablo 10: Mauchly'nin Küresellik Testi	40
Tablo 11: Varyans Analizi Sonuçları	41
Tablo 12: Olumlu Katkıları	43
Tablo 13: Sınıf Yapısı	47
Tablo 14: Öneriler	48
Tablo 15: Olumsuz Görüşler	50

ŞEKİLLER LİSTESİ

Şekil 1: Bilgi İşlemsel Düşünme Teknikleri (Weinberg, 2013).....	11
Şekil 2: CSunplugged.org Sitesi Ekran Görüntüsü	18
Şekil 3: Code.org Sitesi Ekran Görüntüsü	19
Şekil 4: Eba.gov.tr Sitesi Ekran Görüntüsü	20
Şekil 5: Teste Dahil Edilen Soru Örneği (rjx= .63)	26
Şekil 6: Teste Dahil Edilen Soru Örneği (rjx= .50)	27
Şekil 7: Testten Çıkarılan Soru Örneği (rjx= .19).....	27
Şekil 8: Testten Çıkarılan Soru Örneği (rjx= .25).....	27
Şekil 9: 1. Hafta Etkinliğine Ait Örnek Bir Fotoğraf.....	31
Şekil 10: 2. Hafta Etkinliğine Ait Örnek Bir Fotoğraf.....	32
Şekil 11: 3. Hafta Etkinliğine Ait Örnek Bir Fotoğraf.....	33
Şekil 12: 4. Hafta Etkinliğine Ait Örnek Bir Fotoğraf.....	34
Şekil 13: 5. Hafta Etkinliğine Ait Örnek Bir Fotoğraf.....	35
Şekil 14: 6. Hafta Etkinliğine Ait Örnek Bir Fotoğraf.....	35
Şekil 15: Deney Grubuna Ait Box Plot Grafiği	38
Şekil 16: Kontrol Grubuna Ait Box Plot Grafiği	39

SİMGELER VE KISALTMALAR

B ³	Bilgisayarsız Bilgisayar Bilimi
BİD	Bilgi İşlemsel Düşünme
CS Unplugged	Computer Science Unplugged
ISTE	Uluslararası Eğitim Teknolojileri Topluluğu



BİRİNCİ BÖLÜM

I. GİRİŞ

Bu bölümde araştırmanın problem durumu ile gerekçesi, amacı, önemi, varsayımları, sınırlılıkları, araştırmada kullanılan tanımlar ve temel kavramlar yer almaktadır. Problem durumu başlığı altında eğitim teknolojilerindeki değişimlere, bilgisayar bilimi eğitime, bilgisayarsız bilgisayar bilimi eğitiminin ortaya çıkışına ve programlama öğretiminde karşılaşılan zorluklara değinilmiştir. Daha sonra araştırmanın amacı, soruları, önemi ve sınırlılıklarından söz edilerek araştırmada kullanılan kavramların açıklamalarına yer verilmiştir.

1.1 Problem Durumu

İçerisinde bulunulan ve adına bilgi çağı denilen bu yüzyılda bilginin ve öğrenci sayısının artması beraberinde bazı problemleri de getirmiş, bunun çözümü olarak da eğitim süreç ve niteliğinin gelişmesinde önemli paya sahip olan yeni teknolojilerin eğitim kurumlarına girmesi zorunluluğa dönüşmüştür. Günümüzde öğretimin nasıl daha etkili hale gelebileceği konusunda bilgisayar teknolojileri önemli yere sahiptir (Parlak, 2017). Çağa ayak uydurmada ve toplumsal gelişimi sağlamada en önemli rol eğitime düşmekte, eğitim ve eğitimde teknoloji kullanımı, birbirinden ayrılamayan iki kavram haline gelmektedir. Fakat eğitim teknolojisi yalnızca bilgisayar teknolojisi anlamına gelmemektedir. Öğrenci ile öğretilecek konu arasındaki etkileşimin öğrencinin anlayacağı düzeye gelmesine yardımcı olan tüm materyaller eğitim teknolojisinin çalışma alanında yer almaktadır (Demirel ve Altun, 2017). Eğitim teknolojilerinin, öğretme ve öğrenmeyi kolay hale getirdiği, zamanın verimliliğini sağladığı ve maliyeti düşürüp eğitim ortamını zenginleştirerek öğrenciyi aktif hale getirdiği düşünülmektedir (Akkaya, 2017). Ayrıca öğretim süreçlerinde eğitim teknolojilerinden faydalanmak, anlaşılmasında güçlük çekilen konuları somutlaştırarak konunun anlaşılır bir duruma gelmesine yardımcı olmaktadır. Bu sayede öğrenme - öğretme süreci anlamlı hale gelmektedir (Demirel, 2002).

Teknolojik gelişmeler eğitim uygulamalarında birçok değişikliği beraberinde getirdiğinden eğitim teknolojisi uygulamalarının da giderek yaygınlaştığı söylenebilir. Eğitim teknolojilerinde olduğu gibi bilgisayar bilimi eğitiminde de Dünya’da ve Türkiye’de hızlı bir değişim yaşanmaktadır. Bu yaşanan değişim içinde müfredat ve sınıf ortamlarında yeni değişiklikler yapılması gerekmektedir (Göncü, Çetin ve Top, 2018). Günlük hayatta teknoloji bugüne kadar hiç olmadığı seviyede her alanda karşımıza çıktığından, bilgisayar bilimi ile bireylerin teknolojik araçları etkin kullanmaları hedeflenmektedir. Bu sebeple başta bilgisayar olmak üzere bu teknolojilerin nasıl çalıştığını anlamak ve bu doğrultuda eğitimle teknolojiyi bütünleştirmek için bilgisayar bilimi büyük önem taşımaktadır.

Bilgisayar biliminin kavramlarından faydalanarak problem çözme, sistem tasarlama ve insan davranışlarını anlamak bilgi işlemsel düşünme olarak tanımlanmaktadır (Gülbahar, Kalelioğlu ve Karataş, 2017). Eğitim alanında söz sahibi topluluklardan olan Uluslararası Eğitim Teknolojileri Topluluğu’nun (ISTE) belirlediği 21. yüzyılda öğrencilerin sahip olması gereken beceriler içerisinde bilgi işlemsel düşünme becerisinin yer aldığını görülmektedir (ISTE, 2018). Artık günümüzde öğrencilerin teknoloji kullanmayı öğrenmesi yeterli görülmemekte, onların teknolojinin nasıl oluşturulduğunu bilmelerinde bilgi işlemsel düşünmeyi anlamalarının önemli olduğu öne sürülmektedir (Kafai, Burke ve Resnick, 2014). Bilgi işlemsel düşünme, soyutlama, bilginin sistematik işlenmesi, koşullu mantık, problemi ayrıştırma, hata ayıklama ve sistematik hata tespiti gibi öğelerden meydana gelmektedir (Wing, 2006). Bilgi işlemsel düşünme becerilerinin öğeleri ve beceri alanları değerlendirildiğinde programlamayla benzer nitelikte olduğu görülmektedir. Dolayısıyla bilgi işlemsel düşünme becerilerinin öğrencilere kazandırılmasında programlama öğretiminin bir aracı olarak görüldüğü düşünülmektedir. Bu nedenle her bireye programlama becerilerinin kazandırılması önem arz etmektedir (Barut, Tuğtekin ve Kuzu, 2016). Programlama öğretimindeki amaç öğrencilerin sadece programlamanın nasıl yapılacağını öğrenmesiyle kalmayıp, öğrenme için programlamayı kullanabilmelerini sağlamak, bilgi işlemsel düşünme becerisini geliştirerek uygulama yeteneği kazandırmak ve teknoloji çağı gerekliliklerine uymalarını sağlamaktır (Çetin ve Toluk Uçar, 2017; Selby ve Woollard, 2013; Wing, 2006). Programlama öğretimi üst düzey zihinsel beceriler gerektirmesi sebebiyle zor ve uzmanlık isteyen bir süreçtir. Bu zor

ve karmaşık becerilerin öğretilmesinde kullanılan programlama dillerinin de aynı zorlukta olması sebebiyle öğrencilerin programlama öğrenimine bakış açısı olumsuz olmuştur (Ersoy, Madran ve Gülbahar, 2011). Araştırmacılar da, programlamanın öğrenciler tarafından öğrenilmesi konusunda farklı bakış açılarına sahip olsalar da öğrencilerin programlama öğrenirken zorlandıkları konusunda aynı görüştedir (Çetin, 2013). Bu kadar karmaşık ve soyut yapıya sahip programlama eğitimi sürecinde yaşanan zorluklar her ne kadar programlamanın zor olmasına bağlansa da, başarısızlık sebebi olarak eğitimciler tarafından süreci yeterli olarak zenginleştirip kalıcı öğrenme ve öğrenme transferi için yeterli etkinlik yapılmaması gösterilebilir (Hawi, 2010; Özmen ve Altun, 2014). Programlama öğretiminin gerekliliği önemle bahsedilen bir konu olsa da, öğretim yöntemi (Çatlak, Tekdal ve Baz, 2015), algoritmaları anlama ve uygulama (Futschek ve Moschitz, 2010), metin tabanlı programlamada kod dizimi (Özmen ve Altun, 2014) gibi konularda birtakım problemler öne çıkmaktadır. Bu sorunların yanında programlamanın doğası, programlama dilinin özellikleri, programlamaya karşı tutum, motivasyon (Gomes ve Mendes, 2007), özyeterlilik inancı (Aşkar ve Davenport, 2009) gibi faktörler de yer almaktadır. Programlama öğretiminde yaşanan zorluklar için birçok faktör öne çıksa da, ana faktörün geleneksel programlama öğretim yönteminden kaynaklandığı söylenebilir (Byrne ve Lyons, 2001).

Programlama öğretiminde birçok yaklaşımın kullanılması mümkün olsa da bunların arasında dikkat çeken, eğlenceli ve aynı zamanda geleneksel yöntemler kadar etki sağlayabilen Bell ve arkadaşları tarafından tasarlanan Bilgisayarsız Bilgisayar Bilimi (B³) öğretimi uygulamasıdır. Bilgisayar kullanmadan bilgisayar bilimlerinin temelini öğrencilere kazandırmayı hedefleyen bu öğretim etkinlikleri araştırmacıların ve eğitimcilerin dikkatini çekmiş ve günden güne yaygınlaşmaya başlamıştır (Kukul ve Karataş, 2016). B³'te öğrenciler, çoğunlukla bilgisayar kullanırken karşılaştıkları dikkati dağıtıcı ve teknik ayrıntılardan arınmış bir biçimde, ikili sayı, algoritma ve veri sıkıştırma gibi temel kavramları ilgi çekici oyun, bulmaca, kart ve boya kalemi gibi çeşitli öğrenme etkinlikleri sayesinde öğrenirler. Öğrenenler B³ etkinliklerinde bilgisayar kullanmadan, aktif öğrenme gerçekleştirirler. Ayrıca öğrenenlerin yapılan bütün etkinliklerde bilgisayar bilimi kavramlarını öğrenmeleri sağlanır (Kalelioğlu ve Keskinlikçi, 2017). Ayrıca B³ ile teknik altyapı eksiği bulunan eğitim kurumlarında

öğrencilerin bilgisayar olmadan öğrenebilmeleri için farklı etkinlik ve uygulamaların gerçekleştirilmesi, aynı zamanda zengin öğrenme ortamları oluşturulması amaçlanmalıdır.

Öğrencilerin zor ve karmaşık kavramları doğrudan programlama etkinlikleri ile öğrenmesi birçok sorunu da yanında getirebilir. Özellikle kavramsal bilgi eksikliği, programlama dili söz dizim kuralları ile bütünleştğinde öğrencilerin program yazma veya problem çözme süreçlerinde sorun yaşamalarına sebep olabilir (Kalelioğlu, 2015). Özetlenecek olursa gerçekleştirilen araştırmalar, öğrencilerin programlama eğitimini genellikle zor olarak algıladığını (Aşkar ve Davenport, 2009) ve bilgisayarsız bilgisayar bilimi etkinliklerinin günden güne yaygınlaşacağını (Bell, Rosamond ve Casey, 2012) göstermektedir. Bu nedenle çalışmada, programlama öğretiminde bilgisayarsız bilgisayar bilimi etkinliklerinin öğrencilerin akademik başarılarına etkisi ve öğrenci görüşlerinin değerlendirilmesi amaçlanmaktadır.

1.2 Araştırmanın Amacı ve Araştırma Soruları

Bu araştırmanın amacı programlama öğretiminde bilgisayarsız bilgisayar bilimi etkinliklerinin öğrencilerin akademik başarılarına etkisi ve öğrenci görüşlerinin değerlendirilmesidir. Bu amaç doğrultusunda aşağıdaki sorulara yanıt aranacaktır.

1. Bilgisayarsız bilgisayar bilimi etkinliklerinin programlamaya yönelik akademik başarı üzerinde etkisi var mıdır?
 - 1.1. Kontrol grubunun programlamaya yönelik öntest başarı puanı ile sontest başarı puanı arasında anlamlı düzeyde farklılık var mıdır?
 - 1.2. Deney grubunun programlamaya yönelik öntest başarı puanı ile sontest başarı puanını arasında anlamlı düzeyde farklılık var mıdır?
 - 1.3. Deney grubunun programlamaya yönelik sontest başarı puanı ile kontrol grubunun sontest başarı puanı arasında anlamlı düzeyde farklılık var mıdır?
2. Bilgisayarsız bilgisayar bilimi etkinlikleriyle programlama öğretimine katılan öğrencilerin uygulamaya yönelik görüşleri nasıldır?

1.3 Araştırmanın Önemi

Teknolojinin gelişmesi göz önünde bulundurulduğunda öğrenenlerin bu değişimleri yakalamaları ve çağa ayak uydurabilmeleri için bir takım standartlara sahip olmaları gerekmektedir (ISTE, 2018). Bu standartlara dayanarak öğrenenlerden; hedef belirlemeleri, bu hedefe ulaşmada teknolojiyi kullanarak verimli öğrenmeler sağlamaları, dijital dünyaya ayak uydurmaları ve dijital vatandaş olmaları beklenmektedir. Ayrıca öğrencilerden beklenenler arasında kişiye özgü araştırma stratejilerini kullanarak bilgiye ulaşma, yenilikçi yaratıcı çözümler oluşturma, problemi tanımlayarak çözümünde teknoloji kullanma, bilgi işlemsel düşünebilme, kendini ifade etmede uygun ortamlar seçme, yaratıcı iletişime ve küresel işbirliğine sahip olma da bulunmaktadır. Söz konusu becerilerden bazılarının kazandırılmasında programlama öğretiminin de olumlu etkisi bulunmaktadır. Gerçekleştirilen araştırmalar programlama öğretiminin problem çözme becerilerini pozitif yönde etkilediğini gösteren çalışmaların da bulunduğunu göstermektedir (Nam, Kim ve Lee, 2010; Taylor, Harlow ve Forret, 2010; Yıldırım, 2016; Yünkül, Durak, Çankaya ve Mısırlı, 2017). Dünyada ve Türkiye’de hızla yayılan ve popüler hale gelen programlamanın, eğitim programları içerisindeki yeri de gün geçtikçe önem kazanmaktadır. Günümüzde birçok ülke eğitim müfredatlarında programlama eğitimine yer vermiş veya vermeyi düşünmektedir. Programlama eğitimine müfredatında yer veren ülkelere bakıldığında Belçika, Estonya, İngiltere, Portekiz, İspanya, Fransa ve birçok Avrupa Birliği ülkesi ile Avustralya, Güney Kore gibi önemli ülkelerin bulunduğu göze çarpmaktadır (Demirer ve Sak, 2016; Sayın ve Seferoğlu, 2016). Heintz, Mannila ve Farnqvist (2016)’in yayınladığı rapora göre, geçmiş yıllarda yükseköğretimin bir bölümü olan programlama öğretiminin birçok ülkede temel eğitimde zorunlu dersler arasında yer aldığı ifade edilmektedir. Fakat kullanılan programlama dilleri veya öğretim yöntemlerinden ötürü algoritma ve programlama mantığının öğretilmesi ve öğrenilmesinde bir takım güçlüklerle karşılaşıldığı görülmektedir (Saygıner ve Tüzün, 2017). Öğrencilerin programlamaya karşı istek ve motivasyonlarının azalması da bu güçlüklerdendir (Cassel, McGettrick, Guzdial ve Roberts, 2007). Programlama öğretiminin gerçekleştirildiği yöntemlerden biri de B³ eğitimidir. Hermans ve Aivaoglou (2017) tarafından gerçekleştirilen araştırmanın sonucunda B³ etkinliklerinin öğrencilerin programlama ortamlarına olan adaptasyonunu olumlu etkilediği söylenebilir. Ayrıca programlama ve bilgisayar

bilimine karşı ilgi ve motivasyonu arttırdığını vurgulayan birçok araştırmadan söz etmek mümkündür (Jiang ve Wong, 2018; Mano, Allan ve Cooley, 2010).

Alanyazın incelendiğinde B³ eğitimiyle ilgili gerçekleştirilen çalışmaların birçoğunun K-12 eğitim düzeyinde yer aldığı (Rodriguez, Rader ve Camp, 2017; Taub, Ben-Ari ve Armoni, 2009) dikkat çekmekte, söz konusu eğitimin üniversite düzeyindeki öğrencilerin akademik başarısına fayda sağlayıp sağlamayacağının araştırılması önem arz etmektedir. Ayrıca araştırma sonuçlarının uygulayıcılara ve araştırmacılara programlama öğretiminde B³ etkinliklerinin nasıl kullanılması gerektiği konusunda yol göstereceği düşünülmektedir.

1.4 Varsayımlar

1. Eşleştirme sonunda eşleşen öğrencilerin deney ve kontrol gruplarına rastgele atanması akademik başarı dışında diğer özellikleri eşitlediği,
2. Deney ve kontrol gruplarındaki öğrencilerin çalışma kapsamındaki konular açısından bir etkileşimde bulunmadıkları,
3. Araştırmada kullanılan ölçme araçlarının öğrencilerin akademik başarılarını ve görüşlerini ölçtüğü,
4. Cinsiyetin sonuçları etkilemediği varsayılmıştır.

1.5 Araştırmanın Sınırlılıkları

Bu çalışma aşağıdaki sınırlılıkları içermektedir.

Çalışma;

1. Giresun Üniversitesi'nde öğrenim gören araştırmaya gönüllü 54 öğrenciyle,
2. 2018-2019 öğretim yılı güz dönemi programlama öğretiminin sınıfta işlendiği 6 hafta (ikilik düzende sayılar, algoritma, akış şeması, sabitler ve değişkenler, operatörler, döngüler konuları) ile,
3. Akademik başarı testinin geçerlik güvenirlik çalışması 60 öğrenciyle,
4. Kullanılan programlama dilleri C ve C# ile sınırlıdır.

1.6 Tanımlar

Akademik başarı: Programlamaya yönelik gerçekleştirilen akademik başarı testi sonucu öğrencilerin aldığı puanlardır.

Akademik başarı testi: Öğrencilerin programlama konusundaki akademik başarı durumunu gösteren testtir.

Algoritma: Belirli bir problem durumunu çözme ya da belli bir amaca ulaşmak amacıyla tasarlanan yol.

Deney grubu: B³ etkinlikleri kullanılarak programlama öğretimine tabi tutulan öğrencilerin oluşturduğu gruptur.

Görsel programlama: Resimler, grafikler ya da bloklar gibi görsel öğelerden faydalanılarak gerçekleştirilen programlama işlemine verilen ad.

Kod: Program yazmak amacıyla kullanılan ve programlama dilinden diline farklılık gösterebilen, programın en küçük yapı taşıdır.

Kodlama: Belli şart ve düzene uygun bir biçimde yapılması öngörülen işlemlerin tümü.

Kontrol grubu: Geleneksel yöntemle programlama öğretimine tabi tutulan öğrencilerden oluşan gruptur.

Program: İhtiyaca uygun bir biçimde geliştirilen, problemleri çözmek, aynı zamanda rutin olarak gerçekleştirilen işleri daha kolay bir biçimde gerçekleştirmeyi sağlayan bilgisayar yazılımlarıdır.

Programlama: Bilgisayarın donanıma karşı davranışlarının nasıl olması gerektiğini anlatan, bilgisayarı yönlendiren komut satırları, bir dizi aritmetik işlemlerdir.

Programlama Dili: Kişinin yapmak istediklerini bilgisayara yorumlamasını sağlayan dildir.

İKİNCİ BÖLÜM

II. KURAMSAL ÇERÇEVE

Bu bölümde, çalışmanın kuramsal çerçevesini içeren programlama öğretimi, bilgi işlemsel düşünme ve bilgisayarsız bilgisayar bilimi öğretimi kavramları açıklanarak ilgili araştırma örneklerine yer verilmiştir.

2.1 Programlama Tanımı

Günlük hayatta karşılaştığımız bir problemin bilgisayarlar tarafından çözümlenmesi istendiğinde, problemin gerçek hayattan bilgisayar diline dönüştürülmesi gerekmektedir. Bu sebeple bilgisayarın anlayacağı komutların belirli kurallar çerçevesinde birleştirilip çalıştırılması sonucunda problemin çözümü için ortaya çıkan algoritmaya programlama denir. Bir diğer ifade ile programlama, bilgisayarın, donanımın nasıl hareket etmesi gerektiğini anlatan ve bilgisayarın anlayacağı komut, kelime ve aritmetik işlemlerden oluşan yazı dizisidir (Kesici ve Kocabaş, 2007). Günümüzde sıklıkla programlama ile kodlama kavramlarının birbiri yerine kullanıldığı görülmektedir. Aslında bu kavramlar birbiri için kullanılsa da aralarında bazı temel farklılıkların olması da bir gerçektir. Programlama, bilgisayarın belirlenmiş bir görevi yerine getirmesini, problemleri çözmesini ve insan etkileşimini sağlamasını mümkün kılmak amacıyla çeşitli komutlar geliştirilerek uygulama sürecidir. Fakat kodlama, belirli bir mantığa dayanarak görevi gerçekleştirmek için talimatlar yazılması, algoritmanın önceden belirlenmiş programlama dilinde kullanılmasını sağlayan programlamanın bir alt görevi şeklinde ifade edilebilir (EuropeanSchoolnet, 2015). Yine de alanyazında programlama ile kodlama kavramlarının birbiri yerine kullanıldığına dair çalışmaların da var olduğu görülmektedir (Sayın ve Seferoğlu, 2016).

Tüm programlama yapıları gerçek yaşamda çözümü aranan soruların, soyutlanarak bilgisayar yardımıyla çözümlenmesini amaçlar. Karşılaşılan problemleri bilgisayarların anlayacağı yapılara dönüştürülmesi işine bilgisayar programlama denir. Bilgisayar

ortamında komutlar yardımıyla yazılan, bilgisayardan istenen işlemin yerine getirilmesini sağlayan bilgisayar programları, bilgisayar programcıları tarafından problem çözme aşamaları takip edilip bilgisayar dili ile kodlanarak oluşturulur (Coşar, 2013). Programlamada, problemin iyi anlaşılması ve iyi analiz edilmesi önemlidir. Yazılan komutların bir araya gelmesiyle bilgisayarın belirlenmiş bir işi yapmasını gerçekleştiren programlar oluşmaktadır. Farklı bilgisayar ve teknolojiler için oluşturulmuş birçok programlama dili mevcut olsa da dillerin kendine has yazım kuralları ile amacına uygun komutlar oluşturulup çalıştırılır. Dile has komutların yazılmasının gerçekleştiği süreçte, meydana çıkan son ürün de program olarak isimlendirilir (Ersoy, Madran ve Gülbahar, 2011). Programlama dillerinde yer alan komutlar birbirinden farklı gibi görünsede, çözüm için kullanılan programlama mantığı tüm dillerde benzerdir (Arabacıoğlu, Bülbül ve Filiz, 2007).

Programlama yapacak kişiler çeşitli bilgi ve becerilere ihtiyaç duymaktadır. Bayman ve Mayer (1988) bu bilgi ve becerileri Sözdizimsel Bilgi (Syntactic Knowlegde), Kavramsal Bilgi (Conceptual Knowlegde) ve Stratejik Bilgi (Strategic Knowlegde) olarak üç gruba ayırmıştır. Söz konusu bilgi ve beceriler aşağıdaki şekilde açıklanabilir (Dinçer, 2018, s.11):

1. Sözdizimsel Bilgi (Syntactic Knowlegde): Programlama dilinde kodlar yazılırken hangi kuralları kullanarak nasıl yazılacağını gösteren bilgidir. Yani o programlama dilinin dil bilgisi kuralı olarak tanımlanabilir. Örneğin, C# programlama dilinde satır sonlarına noktalı virgül (;) işaretine yer verilirken Visual Basic'de satır sonlarına hiçbir işaret konmamaktadır. Aynı sorunun çözümüne her iki programlama diliyle de ulaşılrken, yazılışlarının farklı olması sözdizimsel bilgiyi gösterir. Sözdizimsel bilgide soruna çözüm üretmez, yalnızca dilin yazılışına ait bilgidir.

2. Kavramsal Bilgi (Conceptual Knowlegde): Programlama dillerindeki kavramların ne olduklarını ve ne iş yaptıklarını gösteren bilgidir. Programlamada yer alan kod ve yapıların anlamı kavramsal bilgi olarak ifade edilebilir. Örnek olarak koşul ifadesini söylediğimizde Visual Basic programlama dilinde IF (eğer) kavramını kullanırız. Kavramsal bilgiyi If kavramı görüldüğünde ne anlam ifade ettiğini anlamanın bilgisi olarak da

açıklanabilir. İlaveten bu kavramlar başka programlama dillerinde de benzer yapıda olabilir. Örneğin C#'de koşul ifadesi IF (eğer) kullanılır. Buradan kavramsal bilgilerin farklı programlama dillerine de transfer edilebileceği sonucu çıkarılabilir.

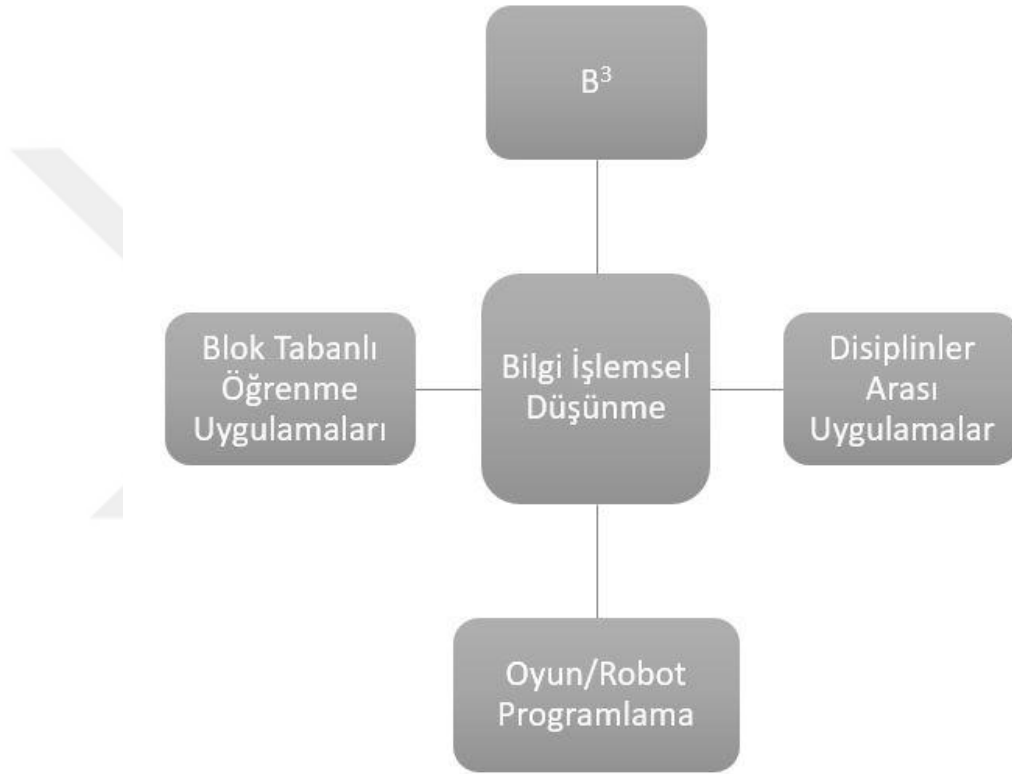
3. *Stratejik Bilgi (Strategic Knowledge)*: Programlama gerçekleştirirken problemin çözülmesi için sözdizimsel ve kavramsal bilgilerin bir arada kullanılmasını sağlayan beceridir. Stratejik bilgi sayesinde programlama dillerinin gramerini bilmeye (sözdizimsel bilgi) aynı zamanda kavramların hangi işi gerçekleştirdiğini bilmeye (kavramsal bilgi) gereksinim vardır. Stratejik bilgi aşamasında program yazılır ve sonuca ulaşılır. Örneğin kullanıcıdan alınan iki sayının hangisinin küçük olduğunu bulmak amacıyla bir program yazılmak istendiğinde, kişinin hem kod satırlarını nasıl yazacağını hem de programın tamamının nasıl yazılması gerektiğinin bilinmesi gereklidir. Bir programın çözümlenmesi ancak bu şekilde mümkün olmaktadır.

2.2 Programlama Öğretimi

Bilgi İşlemsel Düşünme'nin (BİD) bilgisayar bilimiyle ilgili uzun bir tarihi bulunmaktadır. Tarihsel süreçte, özellikle de 1960 ve 1970'lerdeki bilindiği haliyle algoritmik düşünme, problemleri girdi ve çıktı bağlamında ele alarak, algoritmik bağıntıları formüle etme süreci olarak tanımlanmaktaydı (Knuth, 1985). Bugün ise bu kavram, soyutlamanın birçok basamağına genişlemiş, matematiği algoritmalar geliştirmek için kullanma ve farklı büyüklüklerdeki problemler için çözüm önerilerinin en iyi nasıl işlediğini belirleme üzerinde yoğunlaşmıştır (Denning, 2009). Wing (2006) bilgi işlemsel düşünmeyi, bir bilgisayarın etkili bir biçimde gerçekleştirebileceği şekilde bir sorunun formüle edilmesinde ve bununla ilişkili çözümü/çözümlerini ifade etmeyi içeren bir düşünce süreci olarak tanımlamaktadır.

Günümüzde bilgisayar bilimi eğitimi öğrenenlerin bilgi işlemsel düşünme becerilerini geliştirme amacıyla da kullanılmaktadır. Gerçekleştirilen çalışmalar, programlama eğitiminin bilgi işlemsel düşünme becerilerini geliştirmede etkili bir yol olduğunu öne sürmektedir. Bilgi işlemsel düşünme yardımıyla öğrenenler bilgisayar ile ürettikleri çözümleri otomatikleştirerek sorunları etkin bir biçimde çözebilecek ve düşünmenin sınırlarını aşabilecektir. Ayrıca, öğrenenler bilgisayar bilimine ait kavram ve ilkeleri

öğrendiklerinde, sürekli gelişen teknolojik yaşama ve iş hayatına kolayca hazırlanabilecektir. Bilgisayarın bilgi işlem sürecini etkili bir biçimde anlayan öğrenenler, gelişen araç ve uygulamalardan etkilenmeyip, yaşam boyu öğrenen bireyler olabilecektir. Bilgi işlemsel düşünmenin dört ayrı yaklaşım ve teknik yardımıyla öğretilebileceğinden söz edilmektedir (Weinberg, 2013). Bu uygulamalar B³ uygulamaları, yeni öğrenenler için blok tabanlı öğrenme uygulamaları, oyun ya da robotik programlama ve disiplinler arası uygulamalardır.



Şekil 1: Bilgi İşlemsel Düşünme Teknikleri (Weinberg, 2013)

B³ etkinliklerini kullanarak öğrenciler bilgisayar bilimini, programlamanın temel ilkelerini, problem çözmeyi çeşitli kart oyunları ve fiziksel etkinlikler yardımıyla öğrenebilirler (Bell, Witten ve Fellows, 2015). Bilgisayara erişime uzak öğrenme ortamlarında, bu etkinlikler bireysel veya grup çalışmaları ile kolaylıkla gerçekleştirilebilmektedir. Programlamada deneyime sahip olmayan öğrencilere blok tabanlı araçlar yardımıyla programlamanın temel ilkeleri ve bilgi işlemsel düşünme öğretilebilir (Weintrop ve Wilensky, 2015). Oyun ya da robot programlama sayesinde öğrenciler, açık uçlu problem çözüme kapsamında tasarım, uygulama, hata ayıklama ya da daha üst düzey konularda kendilerini geliştirebilir (Ko, 2013). Multidisipliner

etkinlikler sayesinde de öğrenciler programlama etkinliklerini diğer dersler içerisinde uygulamalı bir biçimde öğrenebilir (Cortina ve diğerleri, 2012; Freudenthal, Ogrey, Roy ve Siegel, 2010; Aktaran: Gülbahar ve Kalelioğlu, 2018).

Programlama, bilgisayar biliminde temel yeterliliklerden birisidir. Alanyazındaki araştırmalar incelendiğinde, programlamanın sorgulama (Fox ve Farmer 2011; Psycharis ve Kallia, 2017), matematiksel düşünme gibi becerileri (Taylor, Harlow ve Forret, 2010) geliştirerek, öğrencileri açık uçlu ve karmaşık sorunları çözmeye hazırladığı göze çarpmaktadır (Wing, 2006). Çalışmalarla önemi kavranmaya başlanan programlama öğretiminin birçok faydası bulunsa da bu faydalar aşağıdaki şekilde sınıflandırılabilir (Akpınar ve Altun, 2014, s.3):

Programlama öğretimi,

- ✓ Öğrencilerin dijital okuryazarlık seviyelerini geliştirebilir.
- ✓ Öğrencilerin hayal gücünü ve yaratıcılığını artırabilir.
- ✓ Sonuç ve süreç odaklı düşünmeyi sağlar.
- ✓ Derse olan motivasyonu arttırabilir.
- ✓ Problem çözme, uzamsal ve analitik düşünme gibi becerilerin kazanılmasına yardımcı olur.
- ✓ Ürün odaklı büyük projeler üretmek ve küçük projelerin bütünleşmesiyle karmaşık problemlere çözümler üretme becerisi kazanmalarını sağlar.
- ✓ İşbirlikli çalışma, yaparak-yaşayarak öğrenme ile öğrenme alışkanlıkları geliştirilebilir.

2.2.1 Türkiye’de ve Dünya’da Programlama Öğretimi

Türkiye’de Bilgisayar Bilimi dersi ilköğretim seviyesinde ilk olarak 1997 yılında seçmeli olarak “Bilgisayar” dersi ismiyle eğitim sistemine girmiştir. Sonrasında ise 2007 yılında “Bilişim Teknolojileri” adı altında 4 ve 5. sınıfta haftada 2 saat diğer sınıflarda 1 saat olarak belirlenmiştir. 2013 yılında ise dersin ismi “Bilişim Teknolojileri ve Yazılım” dersi olarak değiştirilmiş 5 ve 6. sınıfta da haftada 2 saat diğer sınıflarda seçmeli olarak sisteme eklenmiştir. Son olarak 2017 yılında dersin ismi aynı kalmakla beraber içerik hedef ve kazanımlar güncellenmiştir (Tebliğler Dergisi, 1997; Tebliğler Dergisi, 2007; Tebliğler Dergisi, 2013; Tebliğler Dergisi, 2017). Yükseköğretimde de programlamaya ilişkin temel kavramlar, programlama mantığı ve

problem çözme süreci; programlamaya yeni başlayanlar için temel ders niteliğinde olan "Programlamaya Giriş", "Programlamanın Temelleri" veya "Programlama Dilleri I" gibi farklı isimlerdeki programlama derslerinde gösterilmektedir. Bu dersler, genel olarak her yerde bilgisayar bilimi ders programında öğretilen ilk dersler arasındadır (Helminen ve Malmi, 2010). Programlama dersinin en önemli hedeflerinden biri öğrencilerin sistematik bir bilgisayar programı geliştirme yaklaşımı öğrenmeleridir. Yeni başlayan öğrenciler için programlama sürecini ortaya çıkarmak bunun önemli bir parçasıdır (Bennedsen ve Caspersen, 2008).

Türkiye’de programlama eğitimi başta devlet okulları olmak üzere özel okullar ve özel kurumlar tarafından verilmektedir. Öğrenciler kendi eğitim düzeylerine göre programlama eğitimleri alarak programlama bilgilerini geliştirmektedirler. İlköğretimden sonra ortaöğretimde özellikle meslek liselerinin bilgisayar bilimleri bölümlerinde bilgisayar okuryazarlığı kazandırmak amacıyla verilen derslerin içerisinde mevcut programlama becerileri bulunmaktadır. Mesleki ve teknik liselerin bilgisayar bölümlerinde ise uzun zamandır programcılık eğitimi verilmektedir (Ersoy, Madran ve Gülbahar, 2011; Yıldız ve Kaya, 2013). Yükseköğretimde ise meslek yüksekokulları bünyesinde olan bilgisayar programcılığı bölümleri ile programlama üzerine eğitim verilmektedir. Bu bölümün yanı sıra bilgisayar bilimleri üzerine kurulan bölümlerde de (örn. Bilgisayar Mühendisliği, Bilgisayar ve Öğretim Teknolojileri Öğretmenliği vb.) bilgisayar programlama dersi adı altında genellikle birinci veya ikinci sınıflarda programlama becerileri öğretilmektedir.

Türkiye’de olduğu gibi dünya üzerindeki ülkeler arasında Almanya, İngiltere, Estonya, ABD, İsrail, Kanada, Tayvan, Hindistan gibi bilgisayar programlamanın önemini farkında olup ortaöğretim hatta ilköğretim müfredatlarında bulunduran ülkeler bulunmaktadır. Almanya’da ve İngiltere’de 2004 yılından günümüze kadar programlama becerileri geliştirilmesi becerisi ilköğretim müfredatında yer almaktadır (Eurydice, 2004). Bunun yanı sıra Amerika Birleşik Devletleri’nde ise öğrencilerin sadece %2’si programlama eğitimi görmektedir (Partovi, 2013). Estonya’da programlama eğitimi ise öğrencilere 6 yaşından itibaren zorunlu olarak verilmektedir (Akçay, 2015). Tayvan’da son yıllarda ilköğretim okullarında Lego Mindstorms’a yer vermeye başlamıştır (Lin, Yen, Yang ve Chen, 2005). Programlamanın küçük yaşta verilmesi gerekliliğine karşın İsrail ilköğretim müfredatlarında bulundurmamakla

birlikte programlama öğretimi 10,11 ve 12. sınıflarda bilgisayar bilimleri dersi kapsamında verilmektedir (Gülmez, 2009). Kanada’da programlama eğitimi ortaöğretim kurumlarında bilgisayar mühendisliği ve bilgisayar bilimleriyle ilgili açılan dersler kapsamında verilmektedir (Stephenson, 2001). Hindistan’da da programlama dilleri öğretiminde en yüksek pazar payına sahiptir. Ülke birçok başarılı bilgisayar mühendisi yetiştirmektedir (Akçay, 2015). Dünyada programlamayla ilgili gerçekleştirilen araştırmalar incelendiğinde, erken yaşta programlama eğitimine verilen önemin giderek yaygınlaştığı görülmektedir. Çocuk ve gençlerin daha kolay bir biçimde programlama yapabilmesi amacıyla programlama araçları geliştirilmekte, eğitim kurumları tarafından da her geçen gün farklı projeler gerçekleştirilmektedir (Demirer ve Sak, 2016). European School net tarafından gerçekleştirilen bir araştırmaya göre araştırmaya dahil edilen 21 Avrupa ülkesinden 16’sı programlamayı öğretim programının bir parçası haline dönüştürmüştür. Bu ülkelerden bazılarının okul seviyelerine göre programlama öğretim durumları Tablo 1’deki şekilde özetlenmiştir (Şimşek, 2018, s.38).

Tablo 1: Programlama Öğretiminin Ülkelerin Okul Seviyelerine Göre Entegrasyon Durumu

	İlkokul	Ortaokul	Ortaokul (Meslek)	Lise	Lise (Meslek)
Avusturya			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Belçika	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
Bulgaristan				<input type="radio"/>	<input type="radio"/>
Çekya					<input type="radio"/>
Danimarka		<input type="radio"/>		<input type="checkbox"/>	<input type="checkbox"/>
Estonya	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
Finlandiya	<input type="radio"/>	<input type="radio"/>			
Fransa	<input type="checkbox"/>	<input type="radio"/>			
Macaristan				<input type="radio"/>	<input type="radio"/>
İngiltere	<input type="radio"/>	<input type="radio"/>		<input type="radio"/>	
İrlanda		<input type="checkbox"/>			
İspanya	<input type="checkbox"/>	<input type="checkbox"/>		<input type="checkbox"/>	
İsrail	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Litvanya				<input type="checkbox"/>	
Malta				<input type="checkbox"/>	
Polonya				<input type="checkbox"/>	<input type="checkbox"/>
Portekiz		<input type="radio"/>		<input type="radio"/>	
Slovakya	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Zorunlu Seçmeli

2.2.2 Programlama Öğretiminde Kullanılan Yaklaşımlar

Bilgisayar teknolojilerindeki hızlı gelişmeler programlama dillerini de etkilemiş, farklı özellik ve kabiliyete sahip birçok programlama dili geliştirilmiştir. Bu diller kullanılarak birçok program geliştirilmiş ve bu programlar sayesinde günlük hayatta karşılaşılan birçok problem kolaylıkla çözülmeye başlanmıştır. Birbirini etkileyen bu sistem neticesinde programlama bilen bireylere olan ihtiyaç her geçen gün artarak devam etse de (Clancy, 2004), programlama eğitimi çoğunlukla üst düzey beceri gerektirdiği için öğrenilmesi zor bir süreç olarak algılanmaktadır (Gültekin, 2006). Bu nedenle gerçekleştirilen çalışmalarda bazı araştırmacılar var olan öğretim yöntemlerini programlama öğretimi için uyarlarlarken, bazıları da programlama dersine özgü kendi müfredatlarını oluşturarak ona uygun yöntemler geliştirmeyi tercih etmiştir. Lye ve Koh (2014) ilk ve ortaokul öğrencilerinin etkili bir biçimde programlama öğrenmeleri için sesli düşünme tekniği yardımıyla kodlamayı ilk önce sözel olarak ifade edebilmelerini, sonrasında programlama çalışmalarına başlamaları gerektiğini belirtmiştir. Çalışmalardan bir tanesinde de Arabacıoğlu, Bülbül ve Filiz (2007) programlama mantığını öğretebilmek amacıyla bir uygulama dili tasarlamışlardır. Teorik yöntemin sıkıcı ve etkili olmadığını söyleyen araştırmacılar geliştirdikleri materyal ile bu kısıtlayıcıların önüne geçmeyi amaçlamışlardır. Bu çalışmayla araştırmacılar soyut bir kavram olan programlamayı daha etkili hale getirmeye çalışmışlardır. Bunun gibi birçok çalışmada da çeşitli araçlar, yöntemler geliştirilerek programlama eğitiminin daha başarılı olması amaçlanmıştır. Yükseltürk ve Altıok (2015) tarafından Bilişim teknolojileri öğretmen adaylarının bilgisayar programlama öğretimine yönelik görüşlerinin değerlendirilmesi amacıyla gerçekleştirilen çalışma 25 bilişim teknolojileri öğretmeni adayıyla gerçekleşmiştir. Betimsel analiz yöntemiyle gerçekleştirilen araştırma sonucunda öğretmen adayları bilişim teknolojileri eğitiminde alternatif yöntem, araçlar konusunda ve etkinlik sonrası kazanımlar hakkında olumlu görüşler belirtmiştir. Üniversite öğrencilerinin programlama derslerindeki başarısızlıklarının sebeplerini ortaya koymak ve programlama sürecinde karşılaştıkları zorlukları değerlendirmek amacıyla gerçekleşen çalışmada ise 12 üniversite öğrencisine açık uçlu sorular yönlendirilmiş araştırma sonucunda öğrencilerin, programlamadaki başarısızlıklarının en büyük nedenlerini pratik eksikliği, algoritma oluşturmama ve bilgi eksikliği olduğunu öne sürdükleri görülmüştür (Özmen ve Altun, 2014).

Programlama yapabilmek için önce problemi anlayarak, uygun adımları belirlemek, bu adımları uygulamak ve değerlendirmek gereklidir. Belirtilen becerilerin geliştirilmesi için programlama öğretiminde hangi etkinliklerin kullanıldığı ve nasıl yapıldığı önem arz etmektedir. Bu nedenle literatürde programlama öğretimi için bilgisayarlı ve bilgisayarsız etkinliklere yer verildiği göze çarpmaktadır (Giannakos, Jaccheri ve Proto, 2013; Milkova ve Hülkova, 2013). Gerçekleştirilen araştırmalarda, bilgisayarsız etkinliklerde öğrencilerin algoritmik düşünme becerilerini geliştirmek için problem odaklı bir yaklaşım izlenmektedir (Futschek, 2006; Kubica ve Radoszewski, 2010). Literatürün öngördüğü şekilde kodlama eğitimine giriş için üç önemli yönelim bulunmaktadır; görsel programlama, robotik ile programlama ve bilgisayarı hiç kullanmadan kodlama kavramlarını anlatmak olan bilgisayarsız bilgisayar bilimidir (Bower ve Falkner, 2015).

2.3 Bilgisayarsız Bilgisayar Bilimi (B³)

Bilgisayar kullanmadan (unplugged) gerçekleştirilen etkinliklerle öğrenciler bilgisayar bilimini, programlamanın temellerini ve problem çözmeyi öğrenebilirler (Bell, Witten ve Fellows, 2015). Bilgisayarlara erişimi olmayan öğrenme ortamlarında, bu etkinlikler bireysel veya grup çalışması ile kolay bir biçimde gerçekleştirilebilir.

Bilgisayar bilimleri, bilimsel bir disiplin olarak enfemasyon süreçleri ve bu süreçlerin dünya ile etkileşimini ele almaktadır. Alanyazında bilgisayar biliminin tanımlarından ortak bir çerçeve içerisinde değerlendirmek gerekirse bu bilim, matematiksel temellere dayanan teknik bir disiplin olarak açıklanmıştır (Booth, 2001). Bilim alanını karşılayan bilgisayar günümüzde yalnız ilgili teknolojik cihazın karşılığı değil çok boyutlu dallara ayrılan bir araştırma disiplini haline dönüşmüştür. Sonuç olarak bu disiplin mekanik bir mühendislik alanı olmanın ötesinde disiplinler arası çalışmaların konusunu oluşturan çok yönlü kuramları ve bu kuramların uygulama alanlarını merkezine alan sistematik işlem süreçlerini kapsayarak toplumları şekillendiren bir bilim alanı olarak tanımlanabilmektedir.

Birçok öğrencinin bilgisayar bilimi ile ilgili olumsuz tutum içerisinde bulunması nedeniyle zorlandığını gözlemleyen Bell, Alexander, Freeman ve Grimley (2009) CS Unplugged adlı öğrenme etkinlikleri geliştirmiştir. Canterbury üniversitesinde ortaya

ıkan bu ğretim etkinlikleri ilgi ekici oyunlar, kartlar, boya kalemleri gibi farklı materyallerin kullanıldıđı ücretsiz bir koleksiyondur. Bu kaynaklar ğrencilerin bilgisayar kullanmadan eğlenceli bir biçimde bilgisayar bilimini keşfetmelerini sağlamakta, programlamanın eğlenceli ve iş birliđi içerisinde gerçekleştirilebildiđini göstermeyi hedeflemektedir. Bell yapılan kişisel bir görüşmede bu etkinliklerin amaçlarını dört maddede açıklamıştır;

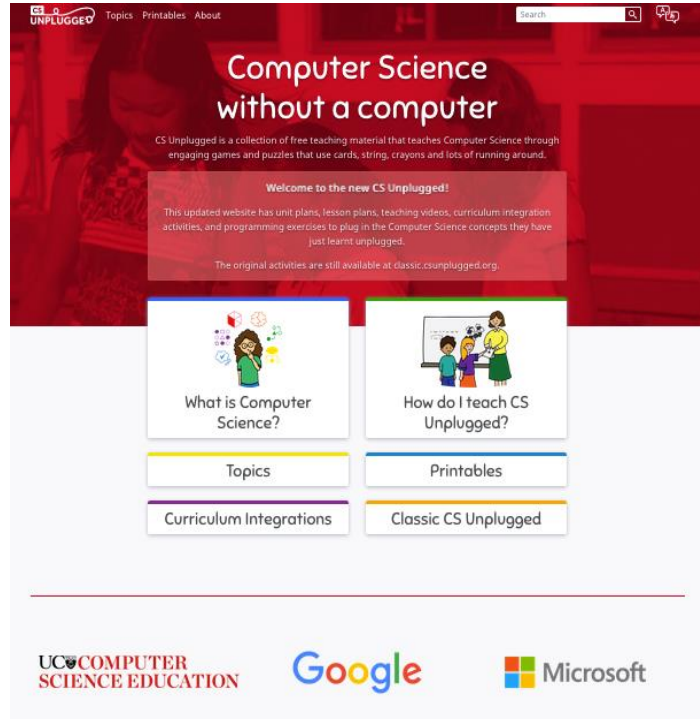
- ✓ Öğrencilerin bilgisayar bilimine olan ilgilerini artırmak,
- ✓ Öğrencilerin daha önce hiç olmadığı kadar işbirlikli ve geri bildirimlerle desteklenecek şekilde bilgisayar biliminin zorluklarıyla karşılaştırmak,
- ✓ Öğrencilerin bilgisayar biliminin ne olduğunu anlamalarını sağlamak ve programlamayla arasındaki farkın anlaşılmasını sağlamak,
- ✓ Kadınların, bilgisayar bilimi kariyer olanaklarına yönelmesini sağlamak (Taub, Ben Ari ve Armoni, 2009).

B³ olarak da isimlendirilen bilgisayarsız bilgisayar bilimi etkinliklerini birçok formda görmek mümkün olsa da bu etkinliklerin temelinde kinestetik etkinlikler olarak tasarlanması önem arz etmektedir. Bu türde geliştirilen öğretim etkinliklerinde dikkate alınacak ilkeler Tim Bell ve meslektaşları tarafından aşağıdaki şekilde belirlenmiştir.

- ✓ Bilgisayar bilimi kavramlarına odaklanılmalı
- ✓ Öğrencilerin grup olarak bir arada olması sağlanmalı
- ✓ Etkinlikler eğlenceli olmalı, sıkıcı bir süreç tasarlanmamalı
- ✓ Malzeme ve materyaller düşük maliyetli olmalı
- ✓ Materyaller Creative Commons lisansı ile yapılmalı, diğer kişilerin deđiştirmesine olanak sağlamalı
- ✓ Etkinlikler cinsiyet açısından tarafsız olmalı
- ✓ Bireyci yaklaşımlardan çok grup ile olan yaklaşımlara odaklanmalı
- ✓ Öğrencilerin ilgisini çekecek veya hayal dünyalarını etkileyecek bir kurgusu olmalı
- ✓ Öğrencilerin sorular sorarak kavramları ve çözümleri keşfetmeleri sağlanmalı
- ✓ Kavramlar aracılıđıyla oyun oynamaları sağlanmalı
- ✓ Etkinliklerde öğrenciler hata yapabilir bu durum tüm etkinliđin ilerlemesini etkilememeli; hata ayıklama da öğretim bir parçası olarak görülmelidir (Bell, Alexander, Freeman ve Grimley, 2009).

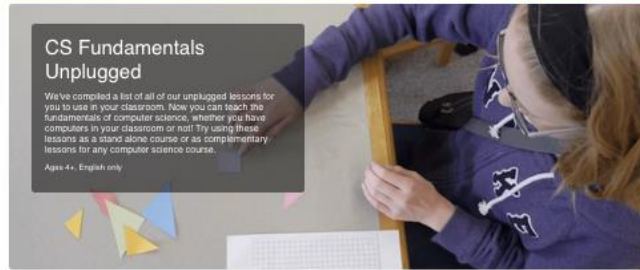
B³ eğitimi için etkinlikler geliştirilirken, öğrencilerin, bilgisayar kullanmadan bilgisayar biliminin temel kavramlarını öğrenmeleri amaçlanmıştır. Etkinlikler içerisinde bilgisayarların bilgi depolama şekilleri, sıralama algoritmaları, arama algoritmaları, görüntü gösterimi gibi temel bilgisayar bilimi kavramları yer almaktadır. Etkinlikler öğrencilerin hem işbirlikli çalışmalarına olanak sağlamakta, hem de sadece bilgisayar başında oturarak değil kinestetik etkinliklerle de bilgisayar bilimini öğrenmelerine olanak sağlamaktadır. Bu özellikleriyle geliştirilmiş olan etkinlikler öğrencilerin herhangi bir donanıma bağlı kalmadan, sıkılmadan derse katılabilmelerini sağlamaktadır. Etkinliklerin bilgisayardan bağımsız olarak geliştirilmiş olmasının, öğrencilerin bilgisayarı bir araç ya da oyuncak olarak görüp, üzerinde çalışılacak bir alan olarak görmemelerinin de önüne geçeceği düşünülmüştür (Bell, Alexander, Freeman ve Grimley, 2009). B³ eğitimi için öğrenme etkinliklerinin yer veren önemli eğitim platformları aşağıdaki gibidir.

- ✓ **csunplugged.org:** CS Unplugged, kart, ip, boya kalem gibi eğlenceli araçlar ile bulmaca ve çekici oyunlar aracılığıyla Bilgisayar Bilimleri derslerine yönelik ücretsiz öğrenme etkinlikleri ve ders planları bulunan öğrenme ortamıdır.



Şekil 2: CSUnplugged.org Sitesi Ekran Görüntüsü

- ✓ **code.org/curriculum/unplugged:** Bill Gates, Mark Zuckerberg, Barack Obama gibi ünlü isimlerin de destek verdiği bu proje kısa zamanda milyonlarca kullanıcıya ulaşmış bulunmaktadır. Herhangi bir program yüklemeden çevrimiçi olarak kullanılabilen uygulama ile kullanıcılar çizgi film ve oyun kahramanları eşliğinde hem keyifli vakit geçirmekte hem de algoritma geliştirme ve problem çözme becerilerini geliştirmektedir. Site de yer alan dersler incelendiğinde; programlamada yer alan temel kavramlar, sıralama, tekrar yapıları, döngüler, koşul ifadeleri, değişkenler, fonksiyonlar, desenler, algoritma, algoritmik düşünme, problem çözme gibi konular öne çıkmaktadır. Web sayfasında öğretmenlerin sınıfta kullanabilmesi amacıyla bilgisayarsız derslerin bir listesi bir araya getirilmiştir. Bu liste yardımıyla sınıfta bilgisayar olup olmama durumuna bakılmadan, bilgisayar bilimlerinin temelleri öğretilir. Bu derslerin yalnız başına bir ders ya da herhangi bir bilgisayar bilimleri dersi için tamamlayıcı olarak kullanılması mümkündür. Ve bu web sayfası aralarında Türkçe'nin de olduğu kırktan fazla dili desteklemektedir.



Each of these activities can either be used alone or with other computer science lessons on related concepts.

2018 Course A-F Curriculum Book

2017 Course A-F Curriculum Book (v2)

Course 1-4 Curriculum Book

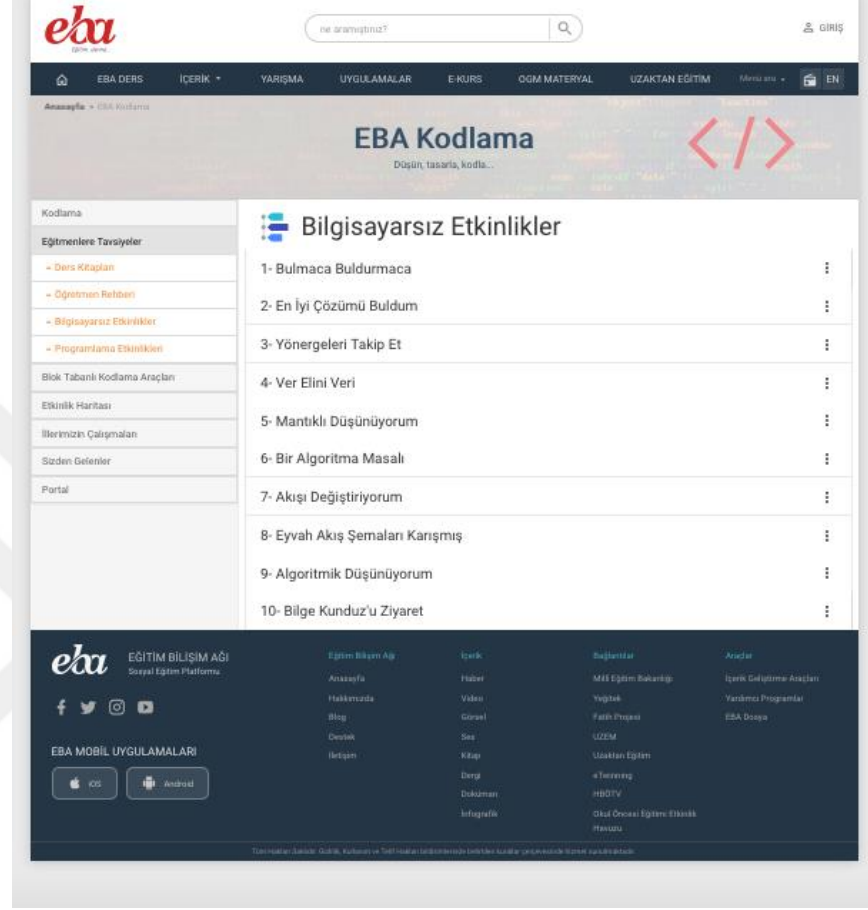
Course A-F Supply List

CSF Flashcards

Concept	Lesson	Curriculum Video	Additional Resources
Algorithms	Dice Race Course E : Lesson 10 Course 2 : Lesson 18 (page 8v) In this lesson, students will relate the concept of algorithms back to real-life activities by playing the Dice Race game. The goal here is to start building the skills to translate real-world situations to online scenarios and vice versa. Lesson Plan Teacher Video Activity Answer Key Assessment Answer Key	 https://youtu.be/WQpgBCKvAs	 See Lesson in Action
	Graph Paper Programming Course D : Lesson 9 Express Course : Lesson 1 Course 2 : Lesson 9 (page 8v) By "programming" one another to draw pictures, students will begin to understand what coding is really about. The class will begin by having students instruct each other to color squares on graph paper in an effort to reproduce	 Unplugged - Graph Paper ...	 Unplugged Lesson in Action - Graph Paper Programming

Şekil 3: Code.org Sitesi Ekran Görüntüsü

- ✓ **eba/bilgisayarsiz-etkinlikler:** Öğretmenlerin programlama eğitiminde kullanabilmesi için ders kitapları, öğretmen rehberi, programlama etkinlikleri ve etkinliklerde kullanılacak görsel, işitsel materyallere yer verilmiştir.



Şekil 4: Eba.gov.tr Sitesi Ekran Görüntüsü

Alanyazın incelendiğinde B³ etkinlikleriyle ilgili araştırmalara rastlansa da var olan araştırmaların kapsamlı olmadığı görülmektedir (Rodriguez, Reder ve Camp, 2016). 2009 yılında Lambert ve Guiffre içerisinde B³ etkinliklerinin yer aldığı etkinlikleri gerçekleştirmiş, bu etkinliklerin öğrencilerin bilgisayar bilimine olan ilgileri ve bilişsel becerilerine etkisini değerlendirmiştir. Araştırma sonucunda, B³ etkinliklerine katılan öğrencilerin bilgisayar bilimi ile daha fazla ilgilendikleri ve bilişsel becerilerinde bir artış meydana geldiği ifade edilmiştir. Mano, Allan ve Cooley (2010) öğrencilerin bilgisayar bilimine olan ilgisini arttırmak amacıyla Alice programlama ortamıyla B³ etkinliklerini bir arada kullanmıştır. Araştırma sonucunda B³ etkinlikleriyle derse katılan öğrencilerin derse karşı ilgilerinin olumlu olduğu belirtilmiştir.

Kukul ve Karataş tarafından 2016 yılında bilişim teknolojileri öğretmen adaylarının B³ etkinlikleri ile ilgili görüşlerinin değerlendirilmesi amacıyla gerçekleştirdikleri araştırmaya 24 öğretmen adayı katılmıştır. Durum çalışması olarak gerçekleştirilen bu araştırmaya göre öğretmen adayları etkinlikleri, öğrencilerin eğlenerek öğrenebilecekleri etkinlikler olarak değerlendirmiş, etkinliklerin öğrencilerin motivasyonlarını artıracaklarını, bilgisayarın temel mantığını öğrenirken üst düzey düşünme becerileri geliştirebildiklerini, kalıcı öğrenme sağladıklarını belirtmişlerdir.



ÜÇÜNCÜ BÖLÜM

III. YÖNTEM

Bu bölümde, araştırmanın desenine, çalışma grubuna, veri toplama araçlarının hazırlanması ve uygulanmasına, araştırma sürecine yer verilmiştir.

3.1 Araştırmanın Deseni

Araştırma karma araştırma desenlerinden sıralı açıklayıcı desen (Explanatory sequential design) kullanılarak gerçekleştirilmiştir. Creswell (2014) sıralı açıklayıcı deseni “ilk olarak nicel verilerin toplandığı ve sonrasında nitel veriler toplanarak nicel sonuçların açıklandığı ya da detaylandırıldığı” bir süreç olarak tanımlamaktadır. Bu çalışmada nicel veriler deneysel bir araştırma sonucunda toplanmıştır. Devamında ise deney sonucunda elde edilen bulguların açıklanabilmesi için deney grubunda yer alan katılımcılarla nitel bir veri toplama süreci gerçekleştirilmiştir. Nitel araştırma sonucunda elde edilen bulgular nicel verilerin açıklanabilmesi için kullanılmıştır.

Araştırmanın nicel bölümü deneysel bir çalışma olup rastgele atamalı eşleştirilmiş öntest ve sontest kontrol gruplu araştırma modeline göre gerçekleştirilmiştir. Bu desen denek gruplarının denk olma olasılıklarını arttırmak amacıyla kullanılmaktadır. Araştırma deseninde karşılaştırılan belli değişkenler üzerinde denek çifti oluşturularak bu çiftlerdeki denekler seçkisiz olarak deney ve kontrol gruplarına atanmaktadır (Fraenkel, Wallen ve Hyun, 2012). Bu çalışmada da deney ve kontrol gruplarının oluşturulması seçkisizlik kuralı göz önünde bulundurularak olmuştur. Programlama öğretiminde B³ etkinliklerine tabi tutulan öğrenciler deney grubunu, geleneksel yöntemle programlama öğretimine tabi tutulan öğrenciler ise kontrol grubunu temsil etmektedir. Araştırma öncesi tüm katılımcılara akademik başarı testi uygulanmıştır. Elde edilen puanlar yukarıdan aşağıya sıralanmış ve öğrenciler başarı puanları dikkate alınarak eşleştirilmiştir. Eşleştirmede her bir ikilinin eşit ya da birbirine en yakın puanı alması durumu göz önünde bulundurulmuştur. Daha sonra ikililerden her biri deney

ve kontrol gruplarına rastgele atanmıştır. Araştırma modelinin şekilsel görünümü ve kullanılan simgelerin anlamları Tablo 2’de sunulmuştur.

Tablo 2: Araştırma Deseni

Deney	E	R	O ₁	X	O ₁ , O ₂
Kontrol	E	R	O ₁	C	O ₁

E= Eşleştirme

R= Seçkisiz atama

O₁= Başarı testi

O₂= Deney Grubu Görüş Formu

X= Bilgisayarsız bilgisayar bilimi etkinlikleri ile programlama eğitimi

C= Geleneksel programlama eğitimi

Deneyisel araştırma sonucunda deney grubu ve kontrol grubu arasındaki olası akademik başarı farkının neden kaynaklandığını ortaya çıkarabilmek adına deney grubunda yer alan katılımcıların görüşleri yazılı olarak alınmıştır. Hazırlanan görüş formu ile deney grubu katılımcılarının B³ etkinlikleri süresinde yaşadıkları deneyimler ortaya çıkarılmaya çalışılmıştır.

3.2 Çalışma Grubu

Bu çalışmanın örneklemini 2018-2019 güz döneminde Giresun Üniversitesi Bilgisayar Teknolojileri bölümünde öğrenim gören araştırmaya katılmaya gönüllü 27’si deney, 27’si kontrol grubunda olmak üzere 54 öğrenci oluşturmaktadır. Uygulama öncesinde uygun örnekleme yöntemi kullanılmıştır. Uygun örnekleme yöntemine göre belirlenen katılımcılar deney ve kontrol gruplarına rastgele bir şekilde atanmıştır. Çalışma grubuna ait cinsiyet ve programlama deneyimi durumlarına ait demografik bilgiler aşağıda Tablo 3 ve Tablo 4’te verilmektedir.

Tablo 3: Çalışma Grubuna Ait Cinsiyet Bilgileri

	Kız	Erkek
Deney	9	18
Kontrol	4	23
Toplam	13	41

Çalışma grubuna ait cinsiyet demografik bilgilerine bakıldığında deney grubunda 9 kız, 18 erkek olmak üzere 27 öğrencinin, kontrol grubunda da 4 kız, 23 erkek olmak üzere 27 öğrencinin yer aldığı görülmektedir. Deney ve kontrol gruplarında cinsiyet sayısı farklılıklarının sonuçları etkilemediği varsayılmıştır.

Tablo 4: Çalışma Grubuna Ait Programlama Deneyimi Bilgileri

	Programlama Deneyimi Olan	Programlama Deneyimi Olmayan
Deney	11	16
Kontrol	13	14
Toplam	24	30

Tablo 4'e bakıldığında deney grubunda yer alan 11 öğrenci daha önce programlama öğretimine tabi tutulduğunu belirtirken, 16 öğrenci daha önce herhangi bir dilde programlama öğretiminde yer almadıklarını ifade etmiştir. Kontrol grubunda ise 13 öğrenci daha önce programlama öğretimine katıldıklarını, 14 öğrenci de katılmadığını dile getirmiştir. Programlama deneyimi olduğunu belirten öğrencilerin bu deneyimi lise eğitimi sırasında edindikleri gözlemlenmiştir. Ayrıca deney grubunu oluşturan 27 öğrenciye ait cinsiyet ve programlama deneyimi bilgileri aşağıda Tablo 5'te verilmiştir.

Tablo 5: Deney Grubu Bilgileri

Öğrenci	Cinsiyet	Programlama Deneyimi
Öğrenci 1	Erkek	Evet
Öğrenci 2	Erkek	Hayır
Öğrenci 3	Kız	Evet
Öğrenci 4	Erkek	Hayır
Öğrenci 5	Erkek	Evet
Öğrenci 6	Erkek	Hayır
Öğrenci 7	Erkek	Evet
Öğrenci 8	Erkek	Hayır
Öğrenci 9	Kız	Hayır
Öğrenci 10	Erkek	Hayır
Öğrenci 11	Kız	Evet
Öğrenci 12	Kız	Hayır
Öğrenci 13	Kız	Hayır
Öğrenci 14	Erkek	Evet
Öğrenci 15	Kız	Hayır
Öğrenci 16	Erkek	Hayır
Öğrenci 17	Erkek	Hayır
Öğrenci 18	Kız	Hayır
Öğrenci 19	Erkek	Evet
Öğrenci 20	Kız	Evet
Öğrenci 21	Erkek	Evet
Öğrenci 22	Erkek	Evet
Öğrenci 23	Erkek	Hayır
Öğrenci 24	Kız	Hayır
Öğrenci 25	Erkek	Evet
Öğrenci 26	Erkek	Hayır
Öğrenci 27	Erkek	Hayır

3.3 Veri Toplama Araçları

Bu bölümde akademik başarı testi ve görüş formu başlıkları altında ilk olarak akademik başarı testi açıklanmış, ardından nitel verileri toplamak için kullanılan görüş formu detaylı olarak sunulmuştur.

3.3.1 Akademik Başarı Testi

Araştırmada nicel verilerin toplanmasında araştırmacı tarafından geçerlik güvenirlik çalışması yapılmış olan akademik başarı testi kullanılmıştır. Başarı testini oluşturan soruların öğretim programının kazanımlarını karşılayacak seviyede olmasına özen gösterilmiştir. Dört alan uzmanı, bir eğitim uzmanı ve bir dil uzmanının değerlendirmesinden sonra hazırlanan 40 sorudan oluşan test (Ek-4) 60 öğrenciye uygulanmıştır. Sonrasında geçerlik analizi kapsamında soruların madde ayırt edicilik indeksleri hesaplanmıştır. Ayırt edicilik indeksi (r_{jx}) .40 ve daha büyük olan maddeler, ayırt etme gücü yüksek olan maddeler iken, .20 - .39 arası ayırt edicilik indeksine sahip maddeler ayırt etme gücü orta, .19 ve daha küçük olan maddelerin ise ayırt etme gücü düşüktür (Tekin, 2000). Bu sebeple madde analizine tabi tutulan sorulardan, ayırt ediciliği .40 ve üzerinde olan sorular teste alınmıştır. Şekil 5 ve Şekil 6'da ayırt edicilik indeksi yüksek olduğu için teste dahil edilen sorular örnek olarak verilmiştir.

6. 1. Başla
 2. Sayac, Sayı, Toplam ve Ortalama değişkenlerini tanımla
 3. Toplam=0, Sayac=0
 4.?....
 5. Eğer Sayı= -1 ise 9. satıra git
 6. Toplam=Toplam + Sayı
 7. Sayac=Sayac + 1
 8. 4. satıra git
 9. Ortalama=Toplam/Sayac
 10. Ortalama değişkeninin değerini yazdır
 11. Dur
- "-1 değeri girilinceye kadar klavyeden girilen sayıların ortalamasını bulup yazdıran" yukarıdaki algoritmada boş bırakılan 4. satıra aşağıdakilerden hangisinde gelmelidir?
- a) Sayı değerini gir
 - b) Sayı=Sayı+l
 - c) Sayı=l
 - d) Sayı=-1
 - e) Sayac=Sayı+Sayac

Şekil 5: Teste Dahil Edilen Soru Örneği ($r_{jx}= .63$)

14. $A=2$, $B=4$, $K=3$, $L=5$ ise aşağıdaki işlemlerden hangisinin sonucu “Yanlıştır”?
- $K^2 \geq 9$
 - Not ($B^2 \leq A$)
 - $A^2=B$ Or $K^2=L$
 - $K \text{ Mod } A=0$ And $B \text{ Mod } L=0$
 - Not ($B \text{ Mod } A=1$ And $K^2 < 9$)

Şekil 6: Teste Dahil Edilen Soru Örneği (rjx= .50)

Ayırt edicilik indeksi .40’ın altında kalan ayırt etme gücü orta ve zayıf olan 13 soru ise testten çıkarılmıştır. Şekil 7 ve Şekil 8’de ayırt edicilik indeksi düşük olduğu için testten çıkarılan sorular örnek olarak sunulmuştur.

16. Simgeler veya sembollerle algoritmanın görsel olarak ifade edilmiş şekline ne ad verilir?
- Karar yapısı
 - Akış şeması
 - Görsel programlama
 - Diyagram
 - Çizelge

Şekil 7: Testten Çıkarılan Soru Örneği (rjx= .19)

40. `#include main () { int x = 5; int y = 2; if (x-y>0 || x%y<0){ printf (" %d", x+y);} else { printf (" %d", x*y);} } Yukarıdaki program kod parçası ekrana aşağıdakilerden hangisini yazdırır?`
- a) 2 b) 3 c) 5 d) 7 e) 10

Şekil 8: Testten Çıkarılan Soru Örneği (rjx= .25)

Testten çıkarılan 13 soru ile akademik başarı testi, konu-soru sayısı yönünden homojen dağılım göstermemiştir. Bunun üzerine konu bazında kapsam geçerliliğini sağlamak amacıyla tekrar uzman değerlendirmesi ile ayırt etme gücü yüksek olmasına rağmen 7 soru daha testten çıkartılmıştır. Testten çıkarılan sorular ile birlikte akademik başarı testinin 20 sorudan oluşan son hali ortaya çıkmıştır (Ek-5). Tablo 6’da bu testi oluşturan soruların madde ayırt edicilik indeksleri verilmiştir.

Son hali 20 sorudan oluşan ve nicel verilerin toplanması amacıyla ön ve son test olarak kullanılan testin güvenilirliğini kontrol etmek için cronbach alpha (α) değeri hesaplanmıştır ($\alpha= .777$). Cronbach alpha değeri .80 - 1.00 arasında çıkan ölçekler yüksek derecede güvenilir, .60 - .80 arasında çıkan ölçekler oldukça güvenilir, .40 - .60 arasında çıkan ölçekler düşük güvenilir olarak kabul edilir. Cronbach alpha değeri .40’ın altında kalan ölçekler ise güvenilir olmadığı için kullanılmaması gerekir (Ergün,

1995; Özdamar, 2004). Buradan hareketle araştırmada kullanılan akademik başarı testinin oldukça güvenilir olduğu söylenebilir.

Tablo 6: Akademik Başarı Testi (Öntest-Sontest) Madde Ayırt Edicilik İndeksi

Öntest-Sontest	Madde Ayırt Edicilik İndeksi (r _{jx})
Soru 1	.44
Soru 2	.44
Soru 3	.44
Soru 4	.69
Soru 5	.63
Soru 6	.63
Soru 7	.44
Soru 8	.44
Soru 9	.50
Soru 10	.56
Soru 11	.44
Soru 12	.50
Soru 13	.44
Soru 14	.50
Soru 15	.50
Soru 16	.44
Soru 17	.44
Soru 18	.56
Soru 19	.44
Soru 20	.44

3.3.2 Görüş Formu

Öğrencilerin akademik başarılarını yorumlamak için deney grubu öğrencilerinden nitel veriler toplanmıştır. Nitel verilerin toplanmasında programlama öğretiminde B³ etkinliklerinin kullanılmasına dair görüş formu kullanılmıştır (Ek-7). Öğrencilerin etkinliklerde neler yaşadıklarını, deneyimlerini ve olumlu-olumsuz görüşlerini öğrenmek için öncelikle bir soru havuzu oluşturulmuştur. Sonrasında görüş formunun kapsam geçerliliğinin sağlanması için dört alan uzmanı, bir eğitim uzmanı ve bir dil

uzmanının deęerlendirmesiyle bu forma son hali verilmiřtir. Etkinlikler tamamlandıktan sonra deney grubu öğrencilerinden bu formu yazılı olarak yanıtlamaları istenmiřtir. Görüş formunda yer alan sorulara dair örnekler ařaęıda belirtilmiřtir.

“Programlama öğretiminde bilgisayarsız bilgisayar bilimi etkinliklerinin uygulanması konuya dikkatinizi çekmeye yardımcı oldu mu? Sizde merak uyandırdı mı? Neden?”

“Bilgisayarsız bilgisayar bilimi etkinliklerinin programlama öğrenmenize katkısını nasıl deęerlendirirsiniz? Bu etkinliklerin size en çok fayda sağladıęı noktalar nelerdir?”

“Programlama öğretiminde bilgisayarsız bilgisayar bilimi etkinliklerinin kullanılması hakkındaki önerileriniz nelerdir?”

3.4 Veri Analizi

Bu bölümde nicel veri analizi ve nitel veri analizi başlıkları altında ilk olarak nicel verilerin analizine yer verilmiř, ardından nitel verilerin analizinin gerçekleştirilmesine deęinilmiřtir.

3.4.1 Nicel Veri Analizi

Nicel verilerin analizinde gerekli olan işlemler SPSS 22.0 paket programı kullanılarak yapılmıřtır. Nicel verilerin çözümlenmesinde hem birden fazla grupta tekrarlı ölçüm yapılması, hem de gruplar arası farkın deęerlendirilmesi sebebiyle karma varyans analizi (mixed ANOVA) testinden yararlanılmıřtır. Elde edilen bulgular bilimsel çerçevede içerisinde deęerlendirilerek arařtırmanın amacına uygun olarak yorumlanmıřtır. Bu analizde deney ve kontrol gruplarını tanımlayan grup deęiřkeni bağımsız deęiřken olarak yer alırken akademik başarı öntest ve sontest puanları ise bağımlı deęiřken olarak kullanılmıřtır.

Analiz gerçekleştirilmeden önce karma varyans analizinin gerektirdięi varsayımlar test edilmiřtir. Bu varsayımlar arasında bağımlı deęiřkenin sürekli olması, bağımsız deęiřkenin en az iki kategori içeren kategorik bir deęiřken olmasının yanında her bir grup için en az iki ölçümün yapılmıř olması gerekmektedir. Varsayımlar arasında yer

alan diđer bir unsur ise uç deđerlerin anlamlı bir fark yaratacak kadar fazla olmamasıdır. Bađımlı deđiřkenin her bir grup ile ilgili ölçümlerinin normal dađılım göstermesi varsayımlar arasında mutlaka test edilmesi gereken bir durumdur. Normallik gibi bađımlı deđiřken ölçümlerine dair varyansların gruplar içinde homojen dađılması beklenmektedir. Grup içi ve gruplar arası oluřan ölçüm farklara ait varyanslarının küreselliklerinin eřitliđi test edilmesi gereken son varsayımdır.

3.4.2 Nitel Veri Analizi

Görüş formunu öğrencilerin el yazılarıyla doldurmaları istenmiştir. Toplanan görüş formları Creswell (2014) tarafından betimlendiđi řekliyle gerçekleştirilmiştir. İlk olarak öğrenciler tarafından doldurulan görüş formları analize hazır hale getirilmiştir. Daha sonra iki öğrenciye ait görüş formlarından yola çıkılarak kategori ve temalar belirlenmiştir. Belirlenen kodlar ve temalar ışığında tüm görüş formları analiz edilmiştir. Nitel veri analizi öz yinelemeli (iterative) bir süreç olduğundan dolayı kodlar ve temalar tüm süreç boyunca gerekli hallerde güncellenmiştir.

3.5 Süreç

Bu çalışma 6 hafta sürmüřtür. Her hafta deney grubu programlama öğretiminde B³ etkinliklerini kullanırken, aynı konuyla ilgili kontrol grubu öğrencileri geleneksel yöntemle programlama öğretimine tabi tutulmuřtur. Ařađıda kontrol grubunda kullanılan geleneksel yöntem açıklanmış ve deney grubunda uygulanan etkinliklere yer verilmiştir.

3.5.1 Kontrol Grubunda Uygulanan Geleneksel Yöntem

Tüm haftalarda deney grubundaki öğrenciler ařađıda belirtilen B³ etkinliklerine katılırken, aynı konular kontrol grubunda geleneksel yöntem uygulanarak işlenmiştir. Geleneksel yöntemden kasıt dersi veren öğretim elemanının konuyu sunuř yöntemiyle anlatması, öğrencilerin de derse dinleyici ve izleyici olarak katılması řeklinde sürmüřtür.

3.5.2 Deney Grubunda Uygulanan B³ Etkinlikleri

Deney grubunda uygulanan etkinliklere ařađıda yer verilmiştir.

3.5.2.1 Hafta 1

Uygulamanın ilk haftasında kontrol grubuyla geleneksel yöntemle “İkilik Düzende Sayılar” konusu işlenirken, deney grubuyla csunplugged.org adresinde Creative Commons lisansı ile sunulan “Noktaları Say: İkilik Düzende Sayılar” etkinliği yapılmıştır. Creative Commons (CC), sağladığı özgür yasal araçlar ile bilginin ve yaratıcılığın paylaşımına, tekrar kullanımına imkan veren kar amacı gütmeyen, herkesin kullanımına açık ve ücretsiz bir lisans türüdür (CC, 2018). Etkinlikte ilk olarak noktalardan oluşan büyük çalışma kağıtları sınıf önünde dizilen öğrencilere verilerek istenilen sayıları birlikte üretmeleri istenmiş, sonrasında küçük çalışma kağıtları bütün gruba dağıtılarak her öğrenci bireysel olarak bu sayıları üretmiştir. Son olarak öğrencilere verilen alfabe kılavuzuna göre ikilik düzende verilen şifreli mesajın içeriği öğrenciler tarafından çözülmüştür. Etkinliğe ait örnek bir fotoğraf Şekil 9’da verilmiştir.



Şekil 9: 1. Hafta Etkinliğine Ait Örnek Bir Fotoğraf

3.5.2.2 Hafta 2

Uygulamanın ikinci haftasında kontrol grubuyla geleneksel yöntemle “Algoritma” konusu işlenirken, deney grubuyla code.org adresinde CC lisansı ile sunulan “Gerçek Hayat Algoritmaları: Kağıt Uçaklar” etkinliği yapılmıştır. Bu etkinlikle öğrencilerin etkinlikte verilen talimatlara uygun olarak algoritma geliştirmeleri hedeflenmekte, geliştirdikleri algoritmayı kağıt uçaklar yaparak gerçek hayattaki etkinliklerle algoritma kavramını ilişkilendirmeleri beklenmektedir. Öğrencilere ilk etapta her gün

rutin olarak yaptıkları işleri soru olarak yöneltip cevapları tahtaya sırasıyla yazmaları istenmiştir. Bu cevaplardan hareketle öğrencilerin her gün yaptığı şeyler için algoritma oluşturmanın mümkün olduğu fikrini kavramaları sağlanmıştır. Sonrasında kâğıt uçak hazırlama aşamalarını gösteren çalışma kağıtları öğrencilere dağıtılarak, algoritmasının hazırlanması ve kâğıt uçağın yapılarak uçurulması gerçekleştirilmiştir. Etkinliğe ait örnek bir fotoğraf Şekil 10’da verilmiştir.



Şekil 10: 2. Hafta Etkinliğine Ait Örnek Bir Fotoğraf

3.5.2.3 Hafta 3

Uygulamanın üçüncü haftasında kontrol grubuyla geleneksel yöntemle “Akış Şeması” konusu işlenirken, deney grubuyla eba.gov.tr’de yer alan “Akışı Değiştiriyorum” etkinliğine yer vermiştir. Bu etkinlikle öğrencilerin algoritma ile akış şemasının farkını ayırt etmesi ve bir algoritma için akış şeması çizmesi beklenmektedir. Uygulamada, öğrencilere akış şeması ile ilgili bilgiler verilmesinin ardından, *sabah rutini* ve *kek tarifi* algoritmalarını gösteren çalışma kağıtlarını kullanarak akış şemaları oluşturmaları istenmiştir. Öğrenciler tarafından oluşturulan akış şemalarının değerlendirilmesinin ardından, aynı etkinliğe ait *robotun rotası* adlı çalışma kağıdına geçilmiştir. Bu çalışmada da araştırmacı tarafından verilen komutlara göre robotun izleyeceği rotanın akış şeması çıkarılmıştır. Etkinliğe ait örnek bir fotoğraf Şekil 11’de verilmiştir.



Şekil 11: 3. Hafta Etkinliğine Ait Örnek Bir Fotoğraf

3.5.2.4 Hafta 4

Uygulamanın dördüncü haftasında kontrol grubuyla geleneksel yöntemle “*Sabitler ve Değişkenler*” konusu işlenirken, deney grubuyla code.org adresinde yayınlanan “*Zarftaki Değişkenler*” etkinliği yapılmıştır. Bu etkinlikle öğrencilere sabit ve değişken kavramları açıklandıktan sonra ilk olarak *zarftaki değişkenler* çalışma kağıtları dağıtılıp cümleleri doğru değişken değerleri ile doldurmaları istenmiştir. Daha sonra öğrenciler ikiyeşerli gruplara ayrılmış, bu gruplara *robot değişkenler* çalışma yaprakları ile küçük zarflar dağıtılmıştır. Her grup tasarladıkları robotun bilgilerini ilgili değişken zarfına yazmışlardır. Sonrasında bu zarflar toplanıp değişken değerleri sınıfla beraber yorumlanmıştır. Etkinliğe ait örnek bir fotoğraf Şekil 12’de verilmiştir.



Şekil 12: 4. Hafta Etkinliğine Ait Örnek Bir Fotoğraf

3.5.2.5 Hafta 5

Uygulamanın beşinci haftasında kontrol grubuyla geleneksel yöntemle “Operatörler” konusu işlenirken, deney grubuyla eba.gov.tr adresinde bulunan “Mantıklı Düşünüyorum” etkinliği yapılmıştır. Bu etkinlikle öğrencilerin problem çözümünde kullanılacak operatörlere örnek vermesi, verilen bir problemde ifade ve eşitlikleri kullanarak çözüm üretmesi hedeflenmektedir. Bu hedeflere yönelik ilk olarak aritmetiksel operatörlerle ilgili bilgilendirmeye yer verilmiş, öğrencilerden tahtaya yazılan sayılar arasındaki boşlukları anlamlı bir şekilde operatörlerle doldurmaları istenmiştir. Mantıksal operatörler hakkındaki bilgilendirmenin ardından da öğrencilerin “köprüler” isimli mantıksal operatör etkinliğini bireysel olarak yapmaları istenmiştir. Son olarak öğrenciler beşerli gruplara ayrılmış, 16 sorudan oluşan “doğanın renkleri” adlı mantıksal operatör etkinliğini en hızlı ve doğru şekilde tamamlayarak diğer grupları yenmeye çalışmıştır. Etkinliğe ait örnek bir fotoğraf Şekil 13’te verilmiştir.



Şekil 13: 5. Hafta Etkinliğine Ait Örnek Bir Fotoğraf

3.5.2.6 Hafta 6

Uygulamanın son haftasında kontrol grubuyla geleneksel yöntemle “Döngüler” konusu işlenirken, deney grubuyla code.org adresinde yayınlanan “Döngü Eğlencesi” etkinliği yapılmıştır. Bu etkinlikle öğrencilerin döngü mantığını kavramaları hedeflenmektedir. Döngü konusu ile ilgili bilgiler verildikten sonra öğrenciler ikişerli eşleştirilip etkinlikteki oyunu karşılıklı oynamaları sağlanmıştır. Etkinliğe ait örnek bir fotoğraf Şekil 14’te verilmiştir.



Şekil 14: 6. Hafta Etkinliğine Ait Örnek Bir Fotoğraf

DÖRDÜNCÜ BÖLÜM

IV. BULGULAR

Bu bölümde araştırma sorularına yanıt bulabilmek amacıyla araştırmaya katılan öğrencilerden toplanan nicel ve nitel verilerin analiz edilmesiyle ulaşılan bulgulara yer verilmiştir.

4.1 Nicel Verilere Ait Bulgular

4.1.1 Tanımlayıcı İstatistik Veriler

Tablo 7 çalışmanın deneysel bölümüne ait tanımlayıcı istatistik verileri sunmaktadır. Deney ve kontrol gruplarına ait öntest ve sontest sonuçlarına dair ortalama, medyan, standart sapma, minimum ve maksimum değerler ve çarpıklık değerleri (Skewness ve Kurtosis değerleri) bu tabloda yer almaktadır.

Buna göre deney grubunun önteste ilişkin ortalaması 39.25 iken, sonteste ilişkin ortalaması 69.07 olmuştur. Öntest ve sonteste ait medyan sırasıyla 40.00 ve 70.00 olarak gerçekleşmiştir. Öntestte minimum ve maksimum değerler 15.00 ve 70.00 iken, sontestte ise 40.00 ile 95.00 olarak ölçülmüştür.

Kontrol grubunun öntestte ortalaması 39.81 iken, sonteste ilişkin ortalaması 57.59 olmuştur. Öntest medyanı 40.00, sontestte medyanı ise 60.00 olarak ölçülmüştür. Öntestteki minimum ve maksimum değerler deney grubundaki değer ile aynı olup 15.00 ve 70.00 olarak saptanmıştır. Sontestte ise 25.00 ve 90.00 olarak gerçekleşmiştir.

Tablo 7: Tanımlayıcı İstatistik

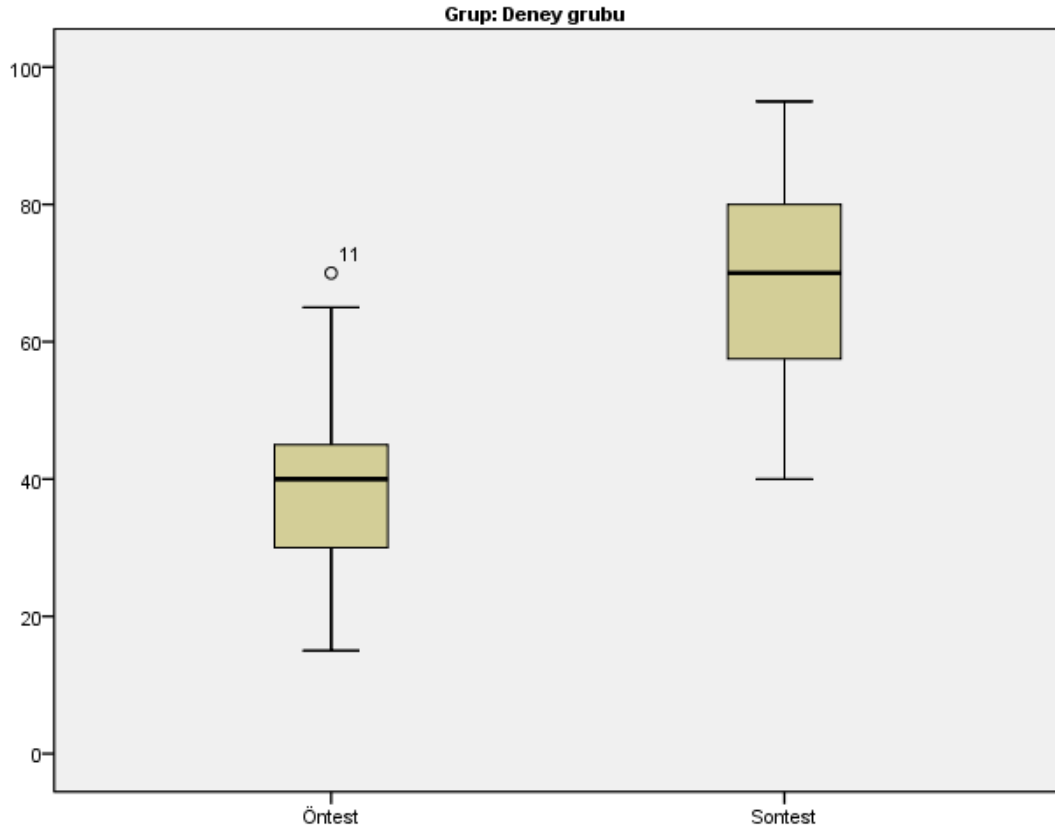
Grup			Statistic	Std. Error
Deney grubu	Öntest	Ortalama	39,25926	2,569597
		Median	40,00000	
		Varyans	178,276	
		Std. Deviation	13,352017	
		Minimum	15,000	
		Maximum	70,000	
		Skewness	,472	,448
	Kurtosis	,087	,872	
	Sontest	Ortalama	69,0741	2,94824
		Median	70,0000	
		Varyans	234,687	
		Std. Deviation	15,31948	
		Minimum	40,00	
		Maximum	95,00	
Skewness		-,227	,448	
Kurtosis	-,843	,872		
Kontrol grubu	Öntest	Ortalama	39,81481	2,701713
		Median	40,00000	
		Varyans	197,080	
		Std. Deviation	14,038510	
		Minimum	15,000	
		Maximum	70,000	
		Skewness	,514	,448
	Kurtosis	-,008	,872	
	Sontest	Ortalama	57,5926	3,58426
		Median	60,0000	
		Varyans	346,866	
		Std. Deviation	18,62434	
		Minimum	25,00	
		Maximum	90,00	
Skewness		,011	,448	
Kurtosis	-,843	,872		

4.1.2 Karma Varyans Analizi Varsayımları

Karma varyans analizi sonuçlarının raporlanmasından önce analizin yapılabilmesi için gerekli olan ön koşullar test edilmiştir. Buna göre ilk ön koşul bağımlı değişkenin sürekli bir değişken olmasıdır. Bağımlı değişken için kullanılan akademik başarı testi

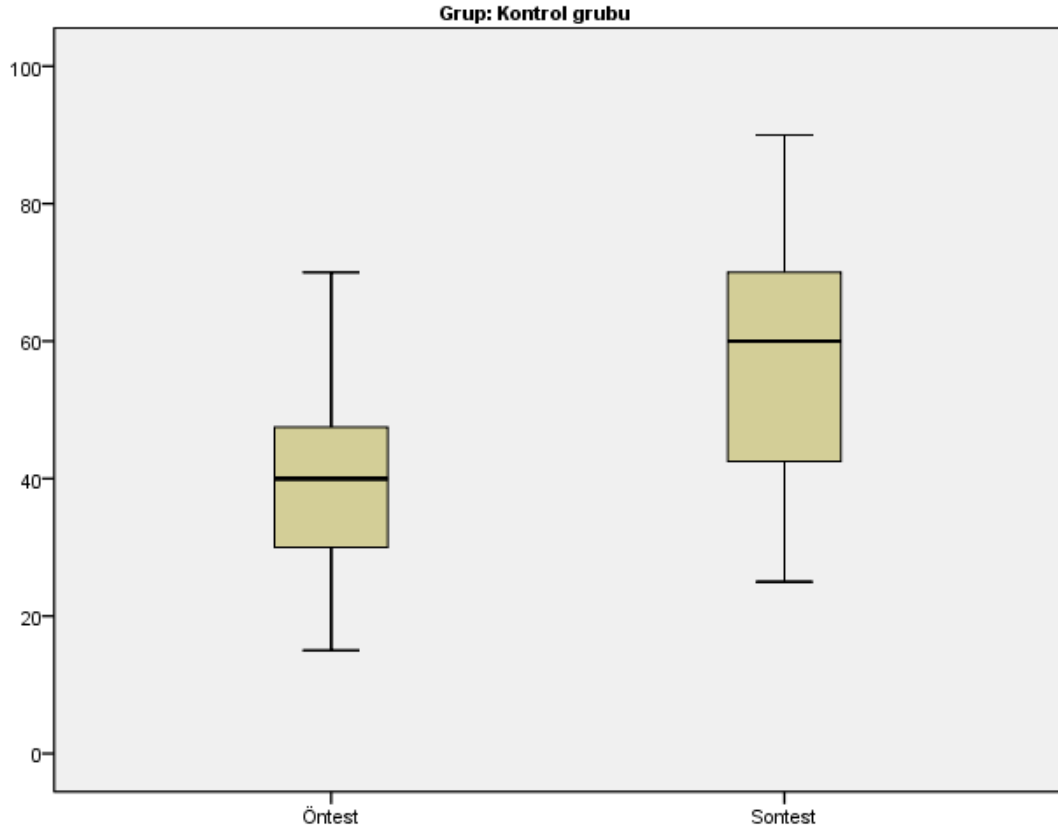
0 ile 100 arasında bir puan ile öğrenciyi değerlendiren bir testtir. Dolayısıyla sürekli bir değişken yapısı ortaya koyduğu için bu varsayım karşılanmıştır. İkinci varsayım ise bağımsız değişkenin en az iki kategori içermesidir. Çalışmada bağımsız değişken deneye katılanların grupları üzerinden oluşturulmuş bir değişkendir. İçerisinde deney ve kontrol olmak üzere iki kategori barındırmaktadır. Bu nedenle bağımsız değişkenin en az iki kategori içermesi varsayımı da sağlanmıştır. Ayrıca her bir grubun öntest ve sontest olmak üzere en az iki ölçüme sahip olması koşulu da yerine getirilmiştir.

Şekil 15, deney grubuna ait öntest ve sontest sonuçlarına ait bilgiler sunmaktadır. Şekilde görüldüğü üzere öntest sonucunda sadece bir uç ölçüm gerçekleşmiştir. Bu ölçümde sontestte ise herhangi bir uç değere rastlanmamıştır.



Şekil 15: Deney Grubuna Ait Box Plot Grafiği

Kontrol grubuna ait olan ölçümlerdeki uç değerler ise Şekil 16'da sunulmuş olan Box Plot ile test edilmiştir. Hem öntest hem de sontest ölçümlerinde herhangi bir uç değer olmadığı tespit edilmiştir. Bu koşullar altında ölçümlerin sonuçları etkileyecek düzeyde uç değerlere sahip olmadığı varsayılmıştır.



Şekil 16: Kontrol Grubuna Ait Box Plot Grafiği

Karma Varyans analizinin diğer bir varsayımı da her bir grup için elde edilen öntest ve sontest puanlarının normal dağılım göstermesidir. Bunun için ilk olarak Kolomogrov-Smirnov ile Shapiro-Wilk değerleri hesaplanmış ve anlamlı olup olmadıkları kontrol edilmiştir. Tablo 8’de gösterildiği üzere Kolomogrov-Smirnov testinin anlamlılık düzeyleri .069 ile .20 arasında bulunmuş ve tüm bu değerlerin .05’ten büyük olduğu tespit edilmiştir. Shapiro-Wilk testi de benzer şekilde tüm ölçümler için .33 ile .55 arasında değerler üretmiş ve bu değerlerin de .05’ten büyük olduğu görülmüştür. Kolomogrov-Smirnov ve Shapiro-Wilk testlerinin anlamlı sonuçlar üretmediği için tüm verinin normal dağılım gösterdiği varsayılmıştır. Ayrıca Tablo 7’de yer alan ve verilerin çarpıklık ve basıklık değerlerini ifade eden Skewness and Kurtosis değerlerinin de -1.5 ile +1.5 arasında olduğu gözlemlenmiştir. Bu da verinin normal dağılım gösterdiğine dair başka bir işarettir.

Tablo 8: Normallik Testi

Grup		Kolmogorov-Smirnov			Shapiro-Wilk		
		Statistic	df	Sig.	Statistic	df	Sig.
Deney grubu	Öntest	,145	27	,155	,967	27	,516
	Sontest	,132	27	,200	,961	27	,399
Kontrol grubu	Öntest	,161	27	,069	,958	27	,338
	Sontest	,107	27	,200	,968	27	,546

Varyansın homojenliği ile ilgili varsayım ise Levene testi ile kontrol edilmiştir. Tablo 9 Levene testi ile ilgili sonuçları sunmaktadır. Buna göre hem öntest hem de sontest ölçümleri içinde varyansın eşit dağılmadığına dair herhangi bir değer saptanmamıştır. Bir diğer deyişle Levene'nin testine dair anlamlılık düzeyleri .84 ve .29 olarak gerçekleşmiş ve anlamlı düzeyde homojenliği bozan bir kanıt bulunamamıştır.

Tablo 9: Levene'nin Hata Varyanslarının Eşitliği Testi

	F	df1	df2	Sig.
Öntest	,043	1	52	,837
Sontest	1,114	1	52	,296

Grup içi ve gruplar arası farkların varyansının eşit olması yani küresellik varsayımı ilgili analizi gerçekleştirilmeden önce son olarak test edilen varsayımdır. Bunun için Mauchly'nin küresellik testi sonuçları hesaplanmıştır (bkz. Tablo 10). Çalışmada sadece iki grup ve iki ölçüm olduğu için anlamlılık değeri hesaplanamamıştır. Dolayısıyla verinin küresellik varsayımını ihlal edecek herhangi bir durum oluşmamıştır. Tüm varsayımların test edilmesiyle karma varyans analizinin (Mixed ANOVA) uygulanması için herhangi bir varsayım ihlali tespit edilememiştir.

Tablo 10: Mauchly'nin Küresellik Testi

Within Subjects Effect	Mauchly's W	Approx. Chi-Square	df	Sig.
On_Test_Son_Test_Basari	1,000	,000	0	.

4.1.3 Karma Varyans Analizi Sonuçları

Tablo 11: Varyans Analizi Sonuçları

Source	Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared	Observed Power ^a	
On_Test_Son_Test_Basari	Sphericity Assumed	15289,120	1	15289,120	219,600	,000	,809	1,000
	Greenhouse-Geisser	15289,120	1,000	15289,120	219,600	,000	,809	1,000
	Huynh-Feldt	15289,120	1,000	15289,120	219,600	,000	,809	1,000
	Lower-bound	15289,120	1,000	15289,120	219,600	,000	,809	1,000
On_Test_Son_Test_Basari * Grup	Sphericity Assumed	978,009	1	978,009	14,047	,000	,213	,957
	Greenhouse-Geisser	978,009	1,000	978,009	14,047	,000	,213	,957
	Huynh-Feldt	978,009	1,000	978,009	14,047	,000	,213	,957
	Lower-bound	978,009	1,000	978,009	14,047	,000	,213	,957
Error(On_Test_Son_Test_Basari)	Sphericity Assumed	3620,370	52	69,623				
	Greenhouse-Geisser	3620,370	52,000	69,623				
	Huynh-Feldt	3620,370	52,000	69,623				
	Lower-bound	3620,370	52,000	69,623				

Tablo 11, karma varyans analizine ait sonuçları göstermektedir. Buna göre deney ve kontrol grubu akademik başarı testine göre gelişim göstermişlerdir. Tüm katılımcıları kapsayacak şekilde bakıldığında katılımcıları sontest puanları anlamlı düzeyde öntest puanlarından fazladır ($F(1,52) = 219.60, p < \alpha = .05$). Tüm katılımcıların süreç boyunca gelişim gösterdikleri tespit edilmekle beraber, öntest ile sontest arasında hangi grubun daha çok gelişim gösterdiği de ayrıca incelenmiştir. Buna göre deney grubu, kontrol grubuna göre anlamlı düzeyde daha yüksek başarı puanları elde etmiştir ($F(1,52) = 14.05, p < \alpha = .05$). Sonuç olarak öntest sonuçlarına göre eşleştirilerek ve eşleştirmeden sonra deney ve kontrol grubuna rastgele atanarak oluşturulmuş gruplardan deney grubu kontrol grubuna göre daha başarılı bulunmuştur. B³ etkinlikleri sonrasında öğrencilerin geleneksel öğretime göre programlamaya yönelik akademik başarıları daha yüksek oranda gelişmiştir.

4.2 Nitel Verilere Ait Bulgular

Altı hafta süren etkinlikler sonrasında deney grubundaki öğrencilere B³ etkinliklerinin kullanılmasına dair hazırlanan görüş formları dağıtılmış ve öğrencilerden bu formları doldurmaları istenmiştir (n=27). Öğrencilerden toplanan formların analizi sonucunda B³ etkinliklerinin olumlu katkıları, sınıf yapısı, öğrenci önerileri ve olumsuz görüşler olmak üzere dört tema ortaya çıkmış ve temalara ait kategoriler ve bulgular bu bölümde tablo ve öğrenci cevap örnekleriyle sunulmuştur.

4.2.1 B³ Etkinliklerinin Olumlu Katkıları

Öğrenci görüşlerinin analizi sonucu 13 ayrı kategori altında olumlu katkılar ortaya çıkarılmıştır. Bu katkılar Tablo 12’de listelenmiştir.

Tablo 12: Olumlu Katkıları

Kategori	f (kişi)	f (tekrar)
Etkileşim	27	64
Motivasyon/İlgi artışı	22	33
Uzun süreli hatırlama	21	37
Basitleştirme	20	24
Programlama öğrenimine destek	14	29
Aktif katılım/Kendini ifade etme fırsatı	12	13
Etkililik	10	14
Önyargıların azalması	8	12
Örneklendirme	8	12
Somatlaştırma	6	6
Eksiklikleri tamamlama/Hataları azaltma	3	7
Başarı artışı	3	3
Eşitlik	1	4

Görüş formu ile toplanan verilere bakıldığında en yüksek frekansa sahip kategorinin etkileşim olduğu görülmektedir (n=27). Ayrıca öğrenci-öğrenci etkileşimi ve öğrenci-öğretmen etkileşimi olmak üzere bu kategorinin iki alt boyutu mevcuttur. Öğrencilerin tamamı öğrenci-öğretmen etkileşiminden bahsederken, aynı zamanda 23 tanesi öğrenci-öğrenci etkileşimine de değinmiştir. Öğrenci-öğrenci etkileşiminin B³ etkinlikleri ile olumlu yönde geliştiğini belirten öğrencilerden öğrenci-14 özellikle sınıfındaki diğer arkadaşları ile daha iyi kaynaştığını ifade etmiş ve birbirinin fikirlerinden yararlanarak daha başarılı olduğunu söylemiştir. Ayrıca öğrenci-20 benzer şekilde görüşlerini aşağıdaki şekilde açıklamıştır.

“Arkadaşarımla olan iletişimimi arttırdı. Sınıf olarak bağlarımız güçlendi. Ortak çalışmayı öğrendik ve beraber kazanmayı kaybetmeyi öğrendik.”

Etkileşim kategorisinin ikinci boyutu olan öğrenci-öğretmen etkileşimiyle ilgili olarak öğrenci-25 etkinlikler sırasında öğretmen ile daha çok konuşma fırsatı olduğunu ve bu sayede kendini daha iyi ifade etmeye başladığını belirtmiştir. Öğrenci-21 ise; “Bu yöntem ile öğretmenimizle olan etkileşimimiz olumlu bir yönde ilerledi ve bu sayede birbirimizi daha iyi tanıdık” şeklinde öğrenci-öğretmen etkileşimine dikkat çekmiştir.

Öne çıkan diğer olumlu katkı ise motivasyon ve ilgi artışı kategorisidir. Bu kategoride 22 öğrenci, programlama öğretiminde B³ etkinliklerin olumlu katkısı bulunduğunu belirtmektedir. Öğrenci-2, programlama derslerinin zor geçtiğini ifade ederken, B³ etkinliklerinin zor görünen bu konuları bile kolaylaştırdığını ve bunun da motivasyonu olumlu etkilediğini aşağıdaki cümlelerle ifade etmiştir.

“Derste anlatılanlar bana genelde çok zor geliyordu. Bu etkinliklerden sonra yapamadığım her şeyi gayret içinde başarabildiğimi gördüm. Çok zor olarak görünse de aslında ikisi de aynı noktaya çıkıyor. Zor olan her şey aslında kolay olarak aktarılabilirmiş.”

Aynı zamanda öğrenci-4 ; “B³ birçok konuya merak duymamı sağladı, çünkü konuları eğlenerek öğrenmek daha çok aklımda kalmasına sebep oluyor.” ifadesi ile bu durumu açıklamıştır.

Uzun süreli hatırlama da öğrenciler tarafından B³ etkinliklerinin en önemli olumlu katkılarından biri olarak görülmüştür. Öğrencilerden 21 tanesi B³ etkinliklerinin programlama öğretiminde uzun süreli hatırlama konusunda etkili olduğu ifade etmiş, bu cevap öğrenciler tarafından 37 kez tekrar edilmiştir. Öğrenci-3 oyun içerikli etkinliklerin hatırlamaya yardımcı olduğunu söylerken, öğrenci-22 ise uzun süreli hatırlamaya olan katkıyı aşağıdaki şekilde ifade etmiştir.

“Ben görsel hafızaya sahip olduğumdan kolay unutmuyorum. Bu uygulamada görsel ve uygulamalı içerikler olduğu için hatırlama konusunda bana yardımcı olacaktır.”

Programlama öğretiminin B³ yoluyla öğretimi için 20 öğrenci geleneksel yöntemlere göre öğrenmeyi basitleştirdiğini belirtmektedir. Öğrenci-23 ve öğrenci-24 etkinlik sayesinde programlama mantığını öğrendiklerini ve programlama ortamına daha kolay aktarabildiklerini belirtmişlerdir. Öğrenci-7 de bu durumu aşağıdaki şekilde ifade etmiştir.

“Programlamada her şey iç içe olduğu için insanın aklı karışıyor. Bu etkinlik sayesinde daha iyi programlamanın mantığını anlıyorsun. Yani etkinlikler bana ve arkadaşlarıma çok şey kattı.”

Ayrıca B³ öğretiminin programlama öğretimine destek olarak kullanılması da öğrenciler tarafından verilen olumlu katkılar arasındadır (n=14). Programlama mantığı ve kod yazma bu kategorinin iki alt kategorisini oluşturmaktadır. Öğrencilerden 12 tanesi B³ etkinliklerinin programlama mantığını öğrenmede olumlu katkısı olduğunu belirtirken, iki öğrenci kod yazmada B³ etkinliğinin olumlu katkısını belirtmiştir. Öğrenci-3 ve öğrenci-9 B³ etkinliklerinin programlama öğretimine destek olarak kullanılmasının programlama mantığını anlamalarında kendilerine olumlu katkı sağladığını belirtmişlerdir. Öğrenci-27 de bilgisayarla programlamaya geçmeden B³ etkinliklerinin yapılmasının programlama mantığını anlamasına yardımcı olduğunu ifade etmiştir. Öğrenci-1 ise etkileşim yönünden daha zengin olduğu için geleneksel kod öğretimine destek olarak kullanılmasını başarılı olarak değerlendirmiştir.

Araştırmaya katılan öğrencilerden 12'si B³ etkinliklerinin kendilerini derse aktif katılıma yönelttiği ve kendini ifade etme yönünden etkili olduğunu belirtmiştir. Öğrenci-2 her aktivitede konuşma fırsatı bulmasının derse karşı ilgisini arttırdığını ifade etmiştir. Öğrenci-13 B³ etkinliklerinin eğlenceli ve akılda kalıcı olmasının aktif olarak derse katılmasını sağladığını, “Geleneksel öğretime göre daha eğlenceli, daha zihinde kalıcı olduğu için etkinliklere sıkılmadan hevesli bir şekilde katıldım.” şeklinde ifade etmiştir.

B³ etkinliklerinin olumlu katkılarından biri olan etkililik kategorisinde öğrenciler tarafından dikkat çekilmiştir (n=10). Öğrenci-13 etkinliklerin hafızada kalıcı olmasından dolayı, programlama dersinin B³ etkinlikleri kullanılarak işlenmesinin etkili olduğunu belirtmiştir. Öğrenci-26 da B³ etkinliklerinin etkililiği hakkındaki görüşünü aşağıdaki şekilde ifade etmiştir.

“Programlama dersi bilgisayarsız bilgisayar bilimi etkinlikleri kullanılarak işlenirse daha etkili olur. Çünkü yapılan aktiviteler öğrenmemizde daha etkili oldu. Eğlenceli bir şekilde aklımıza kazandı. Geleneksel yöntemin aksine sadece anlatımın olmadığı herkesin söz hakkı olarak konuşabildiği bir yöntem. Bu nedenle daha etkili olduğunu düşünüyorum.”

Öğrenci cevaplarından ortaya çıkan bir diğer olumlu kategori ise önyargıların azalmasıdır (n=8). Öğrenci-24 daha önce programlama konusunda kendisine güvenmediğini ancak etkinlikler sayesinde zorlanmadan programlama mantığını

anlayabildiğini söylemiştir. Öğrenci-15 ise bu konuda kendini aşağıdaki şekilde ifade etmiştir.

“Bilgisayarsız bilgisayar bilimi etkinlikleri düşüncemin değişmesine sebep oldu. Etkinlikten önce programlamanın sadece sanal ortamda makinede olduğunu, gerçek hayata çok yansımadığını düşünürken şu anda hayatla iç içe olduğunu farkına vardım. Bu da programlamayı daha kolay anlamamı sağladı.”

Tablo 12 incelendiğinde 8 öğrenci, B³ etkinliklerinin programlama etkinliklerine bir diğer olumlu katkısının örneklendirme imkanı sağlaması olduğunu dile getirmiştir. Öğrenci-19; “Günlük hayattan örneklerle başladığımız için kavramlar aklımda daha kolay yer etti, bu sayede hiç unutacağımı düşünmüyorum.” şeklinde kendini ifade ederek B³ etkinliklerdeki gerçek hayat örneklendirmelerinin uzun süreli hatırlamasını sağlayacağını belirtmiştir. Öğrenci-27 de etkinliklerde günlük hayattan örneklendirme yapılarak uygulama yapıldığı için programlama mantığını öğrenmeye katkı sağladığını belirtmiştir.

Altı öğrenci B³ etkinliklerinin somutlaştırmayı mümkün kıldığını ifade etmiştir. Bu öğrencilerden öğrenci-1 matematiksel yani soyut içeriklerin B³ etkinlikleri ile somutlaştırılmasının program yazarken kendisine olumlu bir katkı sağladığını söylemiştir. Bunu “Matematiksel içeriklerin örneklerle ve çalışmalarla desteklenmesi program yazarken faydalı oluyor.” şeklinde kendi sözleriyle ifade etmiştir.

B³ ile programlama öğretimine katılan öğrencilerden 3’ü etkinliklerin eksiklikleri tamamlayarak hatalarını azaltmada fayda sağladığını belirtmiştir. Öğrenci-14 bu etkinlikler sayesinde daha önce anlayamadığı bazı programlama konularını daha iyi anlayarak eksikliklerini tamamladığını söylemiştir. Benzer şekilde öğrenci-26 da programlama kavramlarında yaşadığı zorlukları bu etkinlik sayesinde düzelttiğini belirtmiştir. Öğrenci-9 ise “Program yazarken yaptığım hataları biraz da olsa azalttı.” ifadesi ile B³ etkinliklerinin hatayı azaltma etkisine değinmiştir.

Tablo 12’ye bakıldığında öğrencilerden 3’ünün B³ etkinliklerinin programlama öğretiminde kullanılmasının başarı artışını sağladığını ifade ettiği görülmektedir. Öğrenci-16 etkinlik sonrasında yapılan son testte, önteste göre daha başarılı olduğunu

belirtmiştir. Öğrenci-15 ise “Bu etkinliklerin programlama sınavındaki başarıma olumlu etkisi oldu.” şeklindeki ifadesiyle B³ etkinliklerin başarıyı etkileyebileceğine dikkat çekmiştir.

Bu kategoride son olarak bir öğrenci tarafından ifade edilen kategori de eşitliklerdir. Öğrenci-2 daha önce bilgisayar kullanmadığını ifade ederek B³ etkinlikleri sayesinde derse aktif katılım sağlayarak arkadaşlarıyla eşit konuma gelebildiğini ifade etmiştir.

Yukarıda verilen ifadelerden yola çıkarak B³ etkinliklerinin olumlu katkılarına en fazla frekansa sahip kategorilerin etkileşim, motivasyon/ilgi artışı, basitleştirme ve uzun süreli hatırlamaya yönelik olduğu söylenebilir. B³ ile programlama öğretimine ilişkin olumlu katkıların yanında sınıf yapısı temasına ilişkin kategoriler de mevcuttur.

4.2.2 B³ Etkinliklerinde Sınıf Yapısı

Araştırmaya katılan öğrencilerin cevaplarından B³ etkinliklerinde sınıf yapısı temasına ait kategoriler belirlenmiştir. Bu kategoriler aşağıda Tablo 13’te sunulmuştur.

Tablo 13: Sınıf Yapısı

Kategori	f (kişi)	f (tekrar)
Grup çalışması	9	9
Rekabet	4	4

B³ etkinlikleri ile programlama öğretimine katılan öğrencilerden 9’u etkinliklerin grup çalışmasına olanak tanıdığını ifade etmiştir. Öğrenci-6 grup etkinliklerinde sınıf arasındaki etkileşiminin artmasının herkese olumlu etki sağladığına değinmiştir. Buradan B³ etkinliklerindeki grup çalışmasının, en önemli olumlu katılardan biri olan etkileşimi ortaya çıkardığı söylenebilir. Ayrıca öğrenci-23 grup çalışmasını aşağıdaki şekilde ifade etmiştir.

“Etkinliklerde genelde gruplar halinde çalıştık. Bu da bizi birbirimize ısındırdı, birbirimize yardımcı da olduk. Bilen bilmeyene öğretti yani kısaca güzeldi.”

Sınıf yapısı temasına ait rekabet kategorisini de 4 öğrenci dile getirmiştir. Örneğin öğrenci-4 etkinliklerin bazılarının yarışma şeklinde yapılmasının kendisini derse daha çok motive ettiğini ve dersi sevdiğini ifade etmiştir. B³ etkinliklerinin rekabet

yapısının, olumlu katkılarından biri olan motivasyon ve ilgi artışına etkisi olduğu bu örnekte görülebilir. Öğrenci-18, “Herkes bir yarış içinde hissetti, aynı zamanda birbirine yardımcı olmaya çalıştı.” şeklindeki ifadesiyle etkinliklerdeki rekabet yapısının arkadaşları arasındaki etkileşimi de arttırdığına dikkat çekmiştir.

Öğrencilerin B³ etkinliklerinin grup çalışması ve rekabete olanak tanıdığını belirtmesinin yanında bu etkinliklere ilişkin önerileri de bulunmaktadır. Bu öneriler B³ öğrenci önerileri temasında kategorilere ayrılmıştır.

4.2.3 B³'e Yönelik Öğrenci Önerileri

Araştırmaya katılan öğrenciler aynı zamanda B³ etkinliklerinin kullanımına ait birtakım önerilerde bulunmuşlardır. Bu kategoriler aşağıda Tablo 14’te sunulmuştur.

Tablo 14: Öneriler

Kategori	f (kişi)	f (tekrar)
Temel programlama eğitiminde kullanım	9	10
Programlama öğrenimine destek olarak kullanım	7	9
Etkinlik sürelerinin/sayılarının/konularının artırılması	7	7
Farklı alanlarda kullanım	2	2

Tablo 14’e göre programlama öğretiminde B³ eğitime katılan 9 öğrenci B³’ün temel programlama eğitimine destek olarak kullanılması gerektiğini dile getirmiştir. Örneğin öğrenci-19 B³ etkinliklerinin basitleştirme özelliğinden dolayı öğrencilerin daha iyi anlayabileceğini düşünerekten temel programlama eğitiminde kullanılmasını önermiştir. Öğrenci-27 de B³ etkinliklerinin programlama mantığını anlamadaki olumlu katkısına vurgu yaparak, temel programlama eğitimindeki kullanılmasını önerisini aşağıdaki şekilde dile getirmiştir.

“Bence ilk başta bilgisayarsız bilim yöntemi kullanılmalı, daha sonra bilgisayar kullanımına geçilerek programlama yapılmalı. Çünkü programlama mantığını anlamada bilgisayarsız etkinliklerin faydası var.”

B³ etkinliklerinin programlama öğretimine destek olarak kullanılmasına yönelik görüşler 7 öğrenci tarafından verilmiştir. Öğrenci-1 başarıyı arttırmada etkili olacağını düşündüğü için, geleneksel yönteme ek olarak yardımcı bir ders olarak B³

etkinliklerinin verilmesi gerektiğini belirtmiştir. Öğrenci-15 de benzer şekilde geleneksel yöntem ek olarak B³ etkinliklerinin kullanılması gerektiğini düşünmektedir. Öğrenci-15 bunu aşağıdaki şekilde ifade etmiştir.

“B³ etkinlikleriyle programlamanın mantığını kavramak daha basit bir hal alıyor. Geleneksel yöntem kullanılmaya devam edilmeli ama ek olarak bilgisayarlı bilgisayar biliminin de kullanılması gerektiğini düşünüyorum.”

B³ etkinliklerinin gerçekleştirilirken etkinlik sürelerinin, sayılarının ya da konularının artırılması da öğrencilerden gelen öneriler arasındadır (n=7). Öğrenci-25 etkinlik sayısının artırılmasını gerektiğini, öğrenci-26 da etkinliklere ayrılan sürelerin artırılması gerektiğini önermiştir. Öğrenci-13 etkinlikteki konu sayılarının artırılmasının iyi olabileceğini belirtmiştir. B³ eğitiminin eğlenceli olduğuna dikkat çekerek daha fazla ders saatinde olmasını öneren öğrenci-5 bunu aşağıdaki şekilde ifade etmiştir.

“Bilgisayarsız bilgisayar eğitiminin daha fazla ders saatinde olmasını isterim. Çünkü eğlenerek programlama öğreniyoruz.”

Aynı zamanda araştırmaya katılan iki öğrenci de B³ etkinliklerinin farklı ders alanlarında da kullanılması gerektiğini belirtmektedir. Öğrenci-2 B³ etkinliklerine zevk alarak katıldığını ifade ederek, basitleştirici özelliğine dikkat çekip buna benzer etkinliklerin diğer derslerde de kullanılmasını önermektedir. Ayrıca öğrenci-2'nin ve farklı alanlarda kullanılmasını öneren diğer öğrenci olan öğrenci-24'ün görüş formundaki yanıtları incelendiğinde B³ etkinliklerini etkileşimli, motivasyonu arttıran, basitleştirici, aktif katılımı sağlayan, uzun süreli hatırlamaya yardımcı olan ve etkili bir öğrenmeye katkı sağlayan olumlu katkıları ifade ettiği görülmektedir. İlaveten öğrenci-24, öğrenci-2'den farklı olarak ayrıca B³ etkinliklerinin programlama öğrenimine destek, önyargıların azalması ve somutlaştırma gibi olumlu katkıları belirtmiştir. Her iki öğrencide B³ etkinliklerinin birçok olumlu katkı yaptığını düşünerek bu tarz etkinliklerin diğer derslerde de kullanılmasını önermişlerdir.

Öğrencilerin önerilerden en fazla frekansa sahip olan kategorinin B³ etkinliklerinin temel programlama eğitiminde kullanımı, programlama öğrenimine destek olarak kullanımı ve etkinlik sürelerinin/sayılarının/konularının artırılması olduğu

söylenbilir. Az sayıda da olsa öğrencilerin B³ etkinliklerinin programlama öğretiminde kullanılmasına ilişkin olumsuz görüşleri de mevcuttur.

4.2.4 B³ Etkinliklerine Ait Olumsuz Görüşler

Araştırmaya katılan öğrencilerin görüş formuna verdikleri cevaplardan yola çıkarak olumsuz görüşler teması oluşturulmuştur. Bu temaya ait kategoriler aşağıda Tablo 15’te verilmiştir.

Tablo 15: Olumsuz Görüşler

Kategori	f (kişi)	f (tekrar)
Transfer zorluğu	4	7
İçerik ile ilgili sınırlılık	1	1
Süre kısıtlılığı	1	1

Programlama öğretiminde B³ etkinliklerinin kullanılmasına ait olumsuz görüşlerin yer aldığı Tablo 15 incelendiğinde en yüksek frekansa ait kategorinin transfer zorluğu olduğu görülmektedir (n=4). Transfer zorluğu ile ifade edilmek istenen, etkinliklerde anlatılan konu/kavramların gerçek programlama ortamına aktarılması sırasında yaşanan zorluklardır. Örneğin öğrenci-4 yaşadığı zorluğu aşağıdaki şekilde ifade etmiştir.

“Döngüler konusunda biraz zorlandım. Bilgisayara nasıl aktaracağımı tam olarak kavrayamadım. Döngülerle ilgili program yazmaya çalışırken halen zorluk yaşıyorum.”

BEŞİNCİ BÖLÜM

V. SONUÇ, TARTIŞMA ve ÖNERİLER

5.1 Sonuç ve Tartışma

Bu araştırma B³ etkinliklerinin programlama öğretiminde kullanılmasının öğrencilerin akademik başarılarına etkisini araştırmak amacıyla yapılmıştır. Ayrıca öğrencilerin etkinliklerde neler yaşadıklarını, deneyimlerini, olumlu ve olumsuz görüşlerini öğrenmek amaçlanmıştır. B³ eğitimiyle ilgili gerçekleştirilen çalışmaların birçoğu K-12 eğitim düzeyinde yer aldığı (Rodriguez, Rader ve Camp, 2017; Taub, Ben-Ari ve Armoni, 2009) için, söz konusu eğitimin üniversite düzeyindeki öğrencilerin akademik başarısına fayda sağlayıp sağlamayacağını araştırılması önem arz ediyordu. Ayrıca alanyazında öğrencilerin programlamayı zor olarak algıladıkları (Aşkar ve Davenport, 2009) ve son yıllarda programlamaya karşı istek ve motivasyonlarının düştüğü (Cassel ve diğerleri, 2007) şeklinde çalışmalar vardı. B³'ün öğrencilerin programlamaya karşı olan zor algısına, düşen istek ve motivasyonuna olumlu etki yapıp yapmayacağı araştırmanın bir diğer önemini oluşturmaktaydı. Bu nedenle üniversite düzeyinde programlama öğretimine tabi tutulan 54 öğrenciyle araştırma yapılarak iki önemli soruya yanıt aranmaya çalışılmıştır. Bunlar B³ etkinliklerinin programlamaya yönelik akademik başarı üzerinde etkisi olup olmadığı ve etkinliklere katılan öğrencilerin uygulamaya yönelik görüşleridir.

Araştırma sürecinde öncelikle geçerlik-güvenirlik çalışması yapılarak öğrencilerin programlama başarılarını ölçecek akademik başarı testi geliştirilmiştir. Bu test araştırmaya katılmaya gönüllü 54 öğrencinin tamamına öntest olarak uygulanmış ve öğrenciler başarılarına göre sıralanarak ikili olarak eşleştirilmiştir. Daha sonra eşleştirilen ikililerden her biri deney ve kontrol gruplarına rastgele atanmıştır. Bu işlemin sonunda 27 öğrenci deney grubunda, 27 öğrenci de kontrol grubunda yer almıştır. Altı hafta süren araştırmada, deney grubunda yer alan öğrenciler programlama öğretiminde B³ etkinliklerine katılırken, kontrol grubu öğrencileri aynı konuyla ilgili geleneksel yöntemle programlama öğretimine tabi tutulmuştur.

Sonrasında her iki gruptan akademik başarı testi (sontest) ile nicel veriler toplanmıştır. Ayrıca deney grubundaki öğrencilerden görüş formu ile nitel veriler elde edilmiştir. Nitel araştırma sonucunda elde edilen bulgular nicel verilerin açıklanabilmesi için kullanılmıştır.

Araştırma bulguları incelendiği zaman hem deney grubunun, hem de kontrol grubunun öntest başarı puanlarıyla sontest başarı puanları arasında pozitif yönde anlamlı fark görülmüştür. Araştırmamızın önemli sorularından biri olan deney ve kontrol grupları sontestleri arasında deney grubu lehine anlamlı fark olduğu gözlenmiştir. Buradan hareketle B³ etkinlikleri kullanılarak yapılan programlama öğretiminin, geleneksel yöntemin kullanıldığı programlama öğretimine göre öğrencilerin akademik başarısına katkı sağladığı söylenebilir.

B³ etkinliklerinin akademik başarıyı arttırmasına etki eden faktörlerin değerlendirilmesinde, görüş formlarından elde edilen bulguların yardımcı olacağı düşünülmektedir. Öğrenciler B³ etkinliklerinin birçok olumlu yönüne dikkat çekmiştir. Etkinliklere katılan deney grubundaki öğrencilerin tamamı B³ etkinliklerinin yüksek etkileşime sahip olduğuna dikkat çekmiştir. Öğrenciler bu etkinlik sayesinde hem birbirleriyle hem de öğretmenleriyle etkileşim içerisinde olduklarını belirtmişlerdir. Etkileşiminin öğrenme üzerine etkisine yönelik Erden (1987), öğrenciler arasındaki etkileşimin önemli rol oynadığını ifade etmiştir. Silbar ise 2002 yılında gerçekleştirdiği araştırmasında öğrencilerin derste etkileşim halinde olmasının dersin daha iyi öğrenilmesini sağladığını belirtmiştir. Bloom (1984) da öğretmen-öğrenci etkileşiminin önemi üzerinde durmuş ve etkili bir eğitim için bunun şart olduğuna değinmiştir. Benzer şekilde Demir'in 2009 yılında yaptığı bir çalışmada öğretmen-öğrenci etkileşiminin akademik başarıya olumlu etki sağladığı da araştırma sonuçları arasındadır. Bu çalışmalardan yola çıkarak B³ etkinliklerinin etkileşimli olmasının öğrencilerin akademik başarılarına olumlu katkı sağlayabileceği söylenebilir.

Rodriguez, Rader ve Camp (2016) öğrenci etkileşimini arttırmak amacıyla B³ etkinliklerini kullanmış, araştırma sonucunda öğrencilerin bilgisayar bilimlerine olan ilgi ve motivasyonlarını arttırdığını öne sürmüştür. Bu çalışmada da öğrencilerin B³ etkinliklerine yönelik görüşlerinde en fazla değindiği ikinci faktör, programlama dersine olan ilgi ve motivasyonlarını arttırmasıdır. B³'ün üniversite seviyesi

öğrencilerinin de programlama dersine karşı olan ilgi ve motivasyonlarını arttırması dikkat çeken bir sonuçtur. Wigfield ve Wentzel (2007) öğrencilerin öğrenmeye yönelik motivasyonlarındaki değişimin doğrudan ya da dolaylı olarak başarılarını etkilediğini ifade etmiştir. Cassel ve diğerleri (2007) de programlama öğretimindeki güçlüklerden biri olarak öğrencilerin programlamaya karşı istek ve motivasyonlarının azalmasını öne sürmüştür. Buna karşın B³ etkinliklerinin öğrencilerin motivasyonları ve derse olan ilgilerini arttırmasından dolayı akademik başarılarına etki edebileceği ifade edilebilir. Kukul ve Karataş'ın 2016 yılında Bilişim Teknolojileri öğretmen adaylarıyla B³ eğitimi kullanılmasına yönelik yapmış olduğu araştırmada, etkinliklerin hem öğrenci motivasyonlarını arttırdığını hem de uzun süreli hatırlamalarına yardımcı olduğu ifade edilmiştir. Bu araştırmaya paralel olarak, uzun süreli hatırlama da B³ etkinliklerinin öğrenciler tarafından dile getirilen olumlu katkılarından biri olarak göze çarpmaktadır.

Öğrencilerin birçoğu B³ etkinliklerinin kendilerini ifade etme fırsatı tanıdığını ve derse aktif katılım sağlamalarına yardımcı olduğunu belirtmiştir. Cortina (2015)'nin B³ etkinlikleriyle öğrencilerin fiziksel olarak problemin içerisinde çözümün parçası olduğunu belirtmesi de B³ etkinliklerinin aktif katılıma imkan sağladığı şeklinde yorumlanabilir. Alanyazın incelendiğinde, öğrencilerin derse katılımının akademik başarıyı arttırdığına yönelik çalışmalar karşımıza çıkmaktadır (Adıyaman, 2008; Furrer ve Skinner, 2003). Etkinliklerin öğrencileri derse aktif katılmaya yönlmesi akademik başarı artışını sağlayan faktörlerden biri olarak gösterilebilir.

B³ etkinlikleriyle öğrenciler programlama ile ilgili kavramları kolay anlayabildiklerini ifade etmiş ve etkinlikler sayesinde öğrenmelerinin kolaylaştığını belirtmiştir. Alanyazında programlamanın öğrenciler tarafından zor olarak algılandığı çalışmalardan (Aşkar ve Davenport, 2009) hareketle B³'ün kolaylaştırıcı bir özelliği olduğu söylenebilir. Ayrıca öğrencilerin bir kısmı etkinliklerin, somutlaştırma ve örneklendirme özellikleri sayesinde programlama öğrenimine destek olarak etkili bir teknik olduğuna da değinmiştir. Bazı öğrenciler ise bu etkinlikler sayesinde programlamaya karşı olan önyargılarının azaldığını dile getirmiştir. Bilgisayar ve/veya programlama tecrübesi olmayan ya da az olan öğrenciler bu etkinlik ile diğer arkadaşlarıyla aynı seviyeye gelebildiklerini ifade etmiştir. Buradan B³ etkinliklerinin sadece öğrenmeye değil öğrenmeye hazırlamaya da etkisi olduğu görülebilir.

Olumlu katkıların yanı sıra öğrenci cevaplarında B³ etkinliklerinin gerçekleştiği sınıfın yapısıyla ilgili görüşler yer almaktadır. Öğrenciler kimi zaman bireysel, kimi zaman ise grup çalışmalarında yer aldıklarını belirtmiştir. Grup çalışması sayesinde öğrenciler, hem arkadaşlarıyla etkileşim içerisinde bulduklarını hem de rekabet yapısı ile derse olan ilgilerini arttırdığını dile getirmiştir. Bu araştırmaya paralel şekilde, Cortina (2015)'da B³ etkinliklerinin grup çalışmasına imkan tanıdığını ve öğrencilerin arkadaşlarıyla birlikte çalışıp, fikir paylaşarak çözümler tasarladıklarını vurgulamıştır. Johnson, Johnson ve Smith'e (2007) göre, öğrenciler grup halinde çalıştıklarında öğrenmelerini daha üst seviyelere taşımaktadır. Ayrıca, alanyazında grupla çalışmanın bireysel çalışmaya kıyasla akademik başarıyı pozitif yönde etkilediğini belirten birçok araştırma da yer almaktadır (Law, 2011; Nam ve Zellner, 2011). Buradan da B³ etkinliklerinin grup çalışmasını sağlama özelliğinin akademik başarı üzerinde etki sağlayabileceği söylenebilir.

B³ etkinliklerinde yer alan öğrencilerden birkaçı etkinliklerde yer alan konu ve kavramları anladığını ancak öğrendiklerini bilgisayar ortamına aktarırken zorluk yaşadığını söylemiştir. Öğrencilerin etkinlik esnasındaki dönütleri ile araştırma sonunda toplanan nitel verilerinden bilgisayarsız bilgisayar bilimini eğlenceli ve faydalı buldukları görülmüştür. Öğrencilerden B³'ün temel programlama eğitiminde kullanılması, programlama öğretimine destek olarak verilmesi ve hatta farklı derslerde de benzer etkinliklerin yer almasına dair öneriler gelmiştir. Ayrıca alanyazın incelendiğinde B³'ün ilköğretim ve ortaöğretim düzeyinde kullanıldığı araştırmalar yer alsa da, üniversite düzeyinde kullanımına yönelik araştırmaların azlığı dikkat çekmektedir. Bu araştırmayla da B³'ün üniversite düzeyinde programlama öğretiminde kullanılmasının, öğrencilerin akademik başarıları üzerinde geleneksel programlama öğretimine göre olumlu etkiye sahip olduğu sonucuna ulaşılabilir.

5.2 Öneriler

Bu bölümde araştırmanın sonuçlarına dayanarak programlama öğretiminde B³ etkinliklerinin kullanılmasına yönelik önerilere yer verilmiştir.

- Araştırmaya katılan öğrencilerden birçoğu B³ etkinliklerinin temel programlama kavramlarının öğretiminde etkili olduğunu ve programlama öğretimine destek olarak verilmesi gerektiğini belirtmektedir. Ayrıca

arařtırmada elde edilen bulgulara gre B³'n đrencilerin akademik bařarılarına olumlu etki yaptığı sylenbilir. Bu sebeple niversite dzeyinde temel programlama đretiminde B³'n kullanılması nerilebilir.

- Alanyazında karřımıza çıkan đrencilerin programlama derslerine karřı zellikle zor olduđunu dřndkleri iin dřen istek ve motivasyonlarının B³ etkinlikleriyle arttırılabileceđi grlmřtr. Eđitimciler derslerini ilgi ekici hale getirmek ve đrenci motivasyonlarını arttırabilmek iin B³ etkinliklerinden yararlanabilirler.
- Arařtırma sonuları incelendiđinde, đrencilerden bazıları B³'n somutlařtırma zelliđi sayesinde programlama đretiminin daha etkili olduđunu dile getirmiřtir. Bu nedenle B³ etkinliklerinin tasarlanması sırasında gerek dnyadan yansımalar kullanılmasına zen gsterilmelidir.
- Arařtırmaya katılan birkaç đrenci B³ ile đrendiđi programlama bilgilerini bilgisayar ortamına aktarırken zorlandıklarını ifade etmiřtir. Bu sonutan hareketle đrenilen bilgileri bilgisayar ortamına aktarmayı kolaylařtıracak farklı etkinlikler geliřtirilebilir.
- B³ etkinlikleri ile eđitimciler okullarının bulunduđu kořullara uygun bir biimde kendi etkinliklerini tasarlayabilirler. Bylece bulunulan fiziksel kořul ve materyallere uygun bir biimde đrenme gerekleřtirilebilir.
- Programlama eđitimi verecek eđitmenler B³ konusunda bilinlendirilmeli, bu etkinliklerin en az diđer bilgisayarlı etkinlikler kadar etkili olabileceđi konusunda haberdar edilmelidir.
- B³ etkinliklerinin ođu programlama eđitimine ynelik olduđu grlmektedir. Programlamanın dıřında bilgisayar ađları, veri tabanı, grafik, insan bilgisayar etkileřimi gibi diđer bilgisayar bilimi alanlarındaki etkinlik sayıları arttırılması faydalı olabilir. Bu alanlarda da B³ etkinliklerinin kullanılmasının đrenci bařarısına etkisinin deđerlendirilmesine ynelik arařtırmalar gerekleřtirilebilir.

- B³ etkinlikleri ile farklı ön bilgilere sahip öğrencilerin de aynı etkinlikler üzerinde çalışması mümkün olabilir. Bu sayede öğrencilerin öğrenmelerini eşit koşullarda gerçekleştirmesi sağlanacaktır. Özellikle öğrenci grubunun programlama giriş seviyesi arasında farklılıklar varsa eğitimciler B³ etkinliklerinden yararlanabilirler.
- B³'ün öğrenmeye etkisini araştıracak deneysel çalışmalar artırılabilir. Öğrencilerin B³ için bahsetmiş olduğu olumlu katkıların nasıl ve neden ortaya çıktığı derinlemesine araştırılabilir.
- Eğitimde B³'ün kullanılmasının üst düzey düşünme becerilerine (bilgi işlemsel düşünme) etkisi konusunda deneysel araştırmalar gerçekleştirilebilir.

KAYNAKÇA

- Akçay, A. (2015). Programlama becerisi öz yeterliliğinin problem çözme ve sorgulama becerileri bağlamında incelenmesi. *Necmettin Erbakan Üniversitesi, Eğitim Bilimleri Enstitüsü, Yüksek Lisans Tezi*.
- Akkaya, E. (2017). Eğitim teknolojisi ve kullanımı ile ilgili teorik bulgular. *Suleyman Demirel University Bulletin Publishes*, 53.
- Akpınar, Y. ve Altun, A. (2014). Bilgi toplumu okullarında programlama eğitimi gereksinimi. *İlköğretim Online*, 13(1), 3.
- Arabacıoğlu, T., Bülbül, H. İ. ve Filiz, A. (2007). Bilgisayar programlama öğretiminde yeni bir yaklaşım. *Akademik Bilişim*.
- Aşkar, P. ve Davenport, D. (2009). An investigation of factors related to self-efficacy for java programming among engineering students. *The Turkish Online Journal of Technology, Knowledge and Society*, 7(1), 37-51.
- Barut, E., Tuğtekin, U. ve Kuzu, A. (2016). Programlama eğitiminin bilgi işlemsel düşünme becerileri bağlamında incelenmesi. *In President of the Symposium* (P. 210).
- Bell, T., Alexander, J., Freeman, I. ve Grimley, M. (2009). Computer science unplugged: School students doing real computing without computers. *The New Zealand Journal of Applied Computing and Information Technology*, 13(1), 20-29.
- Bell, T., Rosamond, F. ve Casey, N. (2012). Computer science unplugged and related projects in math and computer science popularization. *In The Multivariate Algorithmic Revolution and Beyond* (pp. 398-456). Springer, Berlin, Heidelberg.
- Bell, T., Witten, I. H. ve Fellows, M. (2015). CS Unplugged: An enrichment and extension programme for primary-aged students.
- Bennedsen, J. ve Caspersen, M.E. (2008). Teaching of programming. *In J. Bennedsen, M. E. Caspersen, M. Kölling (Eds.), Reflections on the Teaching of Programming Methods and Implementations* (pp.6-16). Springer-Verlag Berlin Heidelberg.
- Bloom, B.S. (1984). The 2-sigma problem: The search for methods of group instruction as effective a on-to-one tutoring. *Educational Researcher* 13 (6),4-16.
- Booth, S. (2001). Learning computer science and engineering in context. *Computer Science Education*, 11(3), 169-188.
- Bower, M. ve Falkner, K. (2015). Computational thinking, the notional machine, pre-service teachers, and research opportunities. *In Proceedings of the 17th Australasian Computing Education Conference (ACE 2015)*, 27, 30.
- Byrne, P. ve Lyons, G. (2001). The effect of student attributes on success in programming. *In ACM SIGCSE Bulletin*. Vol. 33, No. 3, (pp. 49-52). ACM.

- Cassel, L. B., McGettrick, A., Guzdial, M. ve Roberts, E. (2007). The current crisis in computing: what are the real issues?. *In ACM SIGCSE Bulletin*, Vol. 39, No. 1, pp. 329-330, ACM.
- CC (2018). Creative commons nedir? <http://creativecommons.org.tr/hakkinda/> adresinden 20.12.2018 tarihinde erişilmiştir.
- Clancy, M. (2004). Misconceptions and attitudes that interfere with learning to program. *Computer Science Education Research*, 85-100.
- Code (2018). CS fundamentals unplugged. <https://code.org/curriculum/unplugged> adresinden 20.12.2018 tarihinde erişilmiştir.
- Cortina, T. J. (2015). Reaching a broader population of students through unplugged activities. *Communications of the ACM*, 58(3), 25-27.
- Coşar, M. (2013). Problem temelli öğrenme ortamında bilgisayar programlama çalışmalarının akademik başarı, eleştirel düşünme eğilimi ve bilgisayara yönelik tutuma etkileri. *Gazi Üniversitesi, Eğitim Bilimleri Enstitüsü, Doktora Tezi*, Ankara.
- Creswell, J. W. (2014). *Research design: Qualitative, quantitative, and mixed methods approaches*. Thousand Oaks, CA: SAGE Publications.
- Csunplugged (2018). Computer science without a computer. <https://csunplugged.org/en/> adresinden 20 Aralık 2018 tarihinde erişilmiştir.
- Çatlak, Ş., Tekdal, M. ve Baz, F.Ç. (2015). Scratch yazılımı ile programlama öğretiminin durumu: Bir doküman inceleme çalışması. *Journal of Instructional Technologies & Teacher Education*, 4(3).
- Çetin, İ. (2013). Visualization: A tool for enhancing students' concept images of basic objectoriented concepts. *Computer Science Education*, 23(1), 1-23.
- Çetin, İ. ve Toluk Uçar, Z. (2017). Bilgi-işlemsel düşünme tanımı ve kapsamı. *Y. Gülbahar (Ed.), Bilgi İşlemsel Düşünmeden Programlamaya* (41-74). Pegem Yayıncılık, Ankara.
- Demir, C. E. (2009). Factors influencing the academic achievement of the Turkish urban poor. *International Journal of Educational Development*, 29, 17-29.
- Demirel, Ö. (2002). *Planlamadan Değerlendirmeye Öğretme Sanatı*. PegemA, Ankara.
- Demirel, Ö. ve Altun, E. (2017). *Öğretim Teknolojileri ve Materyal Tasarımı*. Pegem Atıf İndeksi, 1-304.
- Demirer, V. ve Sak, N. (2016). Dünyada ve Türkiye'de programlama eğitimi ve yeni yaklaşımlar. *Eğitimde Kuram Ve Uygulama*, 12(3), 521-546.
- Denning, P. J. (2009). The profession of it beyond computational thinking. *Communications Of The Acm*, 52(6). DOI:10.1145/1516046.1516054.
- Dinçer, A. (2018). 6. sınıf öğrencilerine scratch ve kodu game lab programlama dillerinin öğretiminde öğrencilerin tutum, öz yeterlilik ve akademik başarılarının karşılaştırılması. *Dokuz Eylül Üniversitesi Eğitim Bilimleri Enstitüsü, Yüksek Lisans Tezi*.

- Eba (2018). Bilgisayarsız etkinlikler. <http://www.eba.gov.tr/kod/bilgisayarsiz-etkinlikler> adresinden 20.12.2018 tarihinde erişilmiştir.
- Erden, M. (1987). Öğrenciler arasındaki işbirliğine dayalı öğretim teknikleri. *Eğitim ve Bilim Dergisi*, 12:68, 57-60. <http://egitimvebilim.ted.org.tr/index.php/EB/article/viewFile/5967/2092> adresinden 28 Nisan 2019 tarihinde erişilmiştir.
- Ergün, M. (1995). *Bilimsel Araştırmalarda Bilgisayarla İstatistik Uygulamaları*. Ankara: Ocak Yayınları.
- Ersoy, H., Madran, R. O. ve Gülbahar, Y. (2011). Programlama dilleri öğretimine bir model önerisi: Robot programlama. *Akademik Bilişim 2011 Konferansı*.
- EuropeanSchoolnet (2015). Computing our future. http://fcl.eun.org/documents/10180/14689/Computing+our+future_final.pdf/746e36b1-e1a6-4bf1-8105-ea27c0d2bbe0 adresinden 16 Kasım 2018 tarihinde ulaşılmıştır.
- Eurydice, (2004). Key data on information and communication technology in schools in europe. *European Commission, Brussels*.
- Fox, R. W. ve Farmer, M. E. (2011). The effect of computer programming education on the reasoning skills of high school students. In H. R. Arabnia, V. A. Clinsy, ve L. Deligiannidis (Eds.), *Proceedings of the International Conference on Frontiers in Education: Computer Science and Computer Engineering. (FECS'11)* (pp. 187–193). USA: CSREA Press.
- Fraenkel, J. R., Wallen, N. E. ve Hyun, H. H. (2012). Internal validity. *How to Design and Evaluate Research in Education*. New York: McGraw-Hill, 166-83.
- Furrer, C. ve Skinner, E. (2003). Sense of relatedness as a factor in children's academic engagement and performance. *Journal of Educational Psychology*, 95(1), 148.
- Futschek, G. (2006). Algorithmic thinking: the key for understanding computer science. In *International Conference on Informatics in Secondary Schools- Evolution and Perspectives* (pp. 159-168). Springer, Berlin, Heidelberg.
- Futschek, G. ve Moschits, J. (2010). Developing algorithmic thinking by inventing and playing algorithms. https://publik.tuwien.ac.at/files/PubDat_187461.pdf adresinden 15.11.2018 tarihinde erişilmiştir.
- Giannakos, M. N., Jaccheri, L. ve Proto, R. (2013). Teaching computer science toyoung children through creativity: Lessons learned from the case of Norway. In *Proceedings of the 3rd Computer Science Education Research Conference on Computer Science Education Research*, Heerlen.
- Gomes, A. ve Mendes, A. J. (2007). Learning to program-difficulties and solutions. In *International Conference on Engineering Education-ICEE* (Vol. 2007).
- Göncü, A., Çetin, İ. ve Top, E. (2018). Öğretmen adaylarının kodlama eğitimine yönelik görüşleri: Bir durum çalışması. *Mehmet Akif Ersoy Üniversitesi Eğitim Fakültesi Dergisi*, (48), 85-110.
- Gülbahar, Y. ve Kalelioğlu, F. (2018). Bilişim teknolojileri ve bilgisayar bilimi: Öğretim programı güncelleme süreci. *Milli Eğitim Dergisi*, 47(217), 5-23.
- Gülbahar, Y., Kalelioğlu, F. ve Karataş, E. (2017). *Ortaöğretim Bilgisayar Bilimi Ders Kitabı -Kur I*. Milli Eğitim Bakanlığı.

- Gülmez, I. (2009). Programlama öğretiminde görselleştirme araçlarının kullanımının öğrenci başarı ve motivasyonuna etkisi. *Marmara Üniversitesi, Eğitim Bilimleri Enstitüsü, Yüksek Lisans Tezi*, İstanbul.
- Gültekin, K. (2006). Çoklu ortamın bilgisayar programlama başarısı üzerine etkisi. *Hacettepe Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi*, Ankara.
- 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. <https://doi.org/10.1016/j.compedu.2009.10.020>.
- Heintz, F., Mannila, L. ve Färnqvist, T. (2016). A review of models for introducing computational thinking, computer science and computing in K-12 education. *In Frontiers in Education Conference (FIE)*, 2016 IEEE (pp. 1-9). IEEE.
- Helminen, J. ve Malmi, L. (2010). Jype-a program visualization and programming exercise tool for Python. *In Proceedings of the 5th International Symposium on Software Visualization* (pp. 153-162). ACM.
- Hermans, F. ve Aivaloglou, E. (2017). To scratch or not to scratch?: A controlled experiment comparing plugged first and unplugged first programming lessons. *In Proceedings of the 12th Workshop on Primary and Secondary Computing Education*, 49-56, ACM.
- ISTE (2018). ISTE standards for students. <https://www.iste.org/standards/for-students> adresinden 16.11. 2018 tarihinde erişilmiştir.
- Jiang, S. ve Wong, G. K. (2018). Are children more motivated with plugged or unplugged approach to computational thinking?. *In Proceedings of the 49th ACM Technical Symposium on Computer Science Education*, 1094-1094, ACM.
- Johnson, D. W., Johnson, R. T. ve Smith, K. (2007). The state of cooperative learning in postsecondary and professional settings. *Educational Psychology Review*, 19(1), 15–29.
- Kafai, Y., Burke, Q. ve Resnick, M. (2014). Connected code: Why children need to learn programming.
- Kalelioğlu, F. (2015). A new way of teaching programming skills to K-12 students: Code.org. *Computers in Human Behavior*, 52, 200-210.
- Kalelioğlu, F. ve Keskinçilic, F. (2017). Bilgisayar bilimi eğitimi için öğretim yöntemleri. *Pegem Atf İndeksi*, 155-182.
- Kesici, T. ve Kocabaş, Z. (2007). *Bilgisayar 2 Ders Kitabı (2. Baskı)*. Ankara: Semih Ofset.
- Knuth, D. E. (1985). Dynamic huffman coding. *Journal of Algorithms*, 6(2), 163-180.
- Ko, P. (2013). A longitudinal study of the effects of a high school robotics and computational thinking class on academic achievement (WIP). Paper presented at the *IEEE Frontiers in Education Conference (FIE)*, Oklahoma City, OK.
- Kubica, M. ve Radoszewski, J. (2010). Algorithms without programming. *Olympiads in Informatics*, 4.

- Kukul, V. ve Karataş, S. (2016). Bilgisayar kullanmadan bilgisayar bilimi eğitimi: Öğretmen adaylarının görüşleri. *10th International Computer and Instructional Technologies Symposium (ICITS)*, Rize.
- Lambert, L. ve Guiffre, H. (2009). Computer science outreach in an elementary school. *Journal of Computing Sciences in colleges*, 24(3), 118-124.
- Law, Y. (2011). The effects of cooperative learning on enhancing Hong Kong fifth graders' achievement goals, autonomous motivation and reading proficiency. *Journal of Research in Reading*, 34(4), 402-425.
- Lin, J. M. C., Yen, L. Y., Yang, M. C. ve Chen, C. F. (2005). Teaching computer programming in elementary schools: A pilot study. *In National educational computing conference*.
- Lye, S. Y. ve Koh, J. H. L. (2014). Review on teaching and learning of computational thinking through programming: What is next for K-12? *Computers in Human Behavior*, 41, 51-61.
- Mano, C., Allan, V. ve Cooley, D. (2010). Effective in-class activities for middle school outreach programs. *In Frontiers in Education Conference (FIE)*, 2010 IEEE (pp. F2E-1). IEEE.
- Milková, E. ve Hulkova, A. (2013). Algorithmic and logical thinking development: base of programming skills. *WSEAS Transactions on Computers*, 12(2), 41-51.
- Nam, C.W., ve Zellner, R.D. (2011). The relative effects of positive interdependence and group processing on student achievement and attitude in online cooperative learning. *Computers & Education*, 56, 680-688.
- Nam, D., Kim, Y. ve Lee, T. (2010). The effects of scaffolding-based courseware for the scratch programming learning on student problem solving skill. *Proceedings of the 18th International Conference on Computers in Education*.
- Özdamar, K. (2004). *Paket Programlar İle İstatistiksel Veri Analizi*. Eskişehir: Kaan Kitabevi.
- Özmen, B. ve Altun, A. (2014). Üniversite öğrencilerinin programlama deneyimleri: Güçlükler ve engeller. *Turkish Online Journal of Qualitative Inquiry*, 5(3), 1-27.
- Parlak, B. (2017). Dijital çağda eğitim: Olanaklar ve uygulamalar üzerine bir analiz. *Suleyman Demirel University Journal of Faculty of Economics & Administrative Sciences*, 22.
- Partovi, H. (2013). Computer programming education needed. *USA Today*. <https://www.usatoday.com/story/opinion/2013/02/26/computer-programming-coding-education/1947551/> adresinden 30.12.2018 tarihinde erişilmiştir.
- Psycharis, S. ve Kallia, M. (2017). The effects of computer programming on high school students' reasoning skills and mathematical self-efficacy and problem solving. *Instructional Science*, 45(5), 583-602.
- Rodriguez, B., Rader, C. ve Camp, T. (2016). Using student performance to assess CS unplugged activities in a classroom environment. *In Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education* (pp. 95-100). ACM.

- Saygıner, Ş. ve Tüzün, H. (2017). İlköğretim düzeyinde programlama eğitimi: Yurt dışı ve yurt içi perspektifinden bir bakış. *Akademik Bilişim Konferansı*.
- Sayın, Z. ve Seferoğlu, S. S. (2016). Yeni bir 21. yüzyıl becerisi olarak kodlama eğitimi ve kodlamanın eğitim politikalarına etkisi. *Akademik Bilişim Konferansı*, 3-5.
- Selby, C. ve Woollard, J. (2013). Computational thinking: The developing definition. *University of Southampton (E-prints)* 6pp. https://eprints.soton.ac.uk/356481/1/Selby_Woollard_bg_soton_eprints.pdf adresinden 7 Kasım 2018 tarihinde erişilmiştir.
- Silbar, R. R. (2002). Web delivery of interactive laboratories: Comparison of three authoring tools. *Computer in Science and Engineering*, WhistleSoft Inc.
- Stephenson, C. (2001). Knowing what to teach-computing in high schools panel. *The Computer Science and Information Technology Symposium Issues and Trends in High School Computing*, Illinois, Chicago.
- Şimşek, İ. (2018). Dünyada programlama öğretimi. *Y. Gülbahar ve H. Karal (Eds.), Kuramdan Uygulamaya Programlama Öğretimi*. Pegem Yayınları, 38-129.
- Taub, R., Ben-Ari, M. ve Armoni, M. (2009). The effect of CS unplugged on middle-school students' views of CS. *ACM SIGCSE Bulletin*, 41(3), 99-103.
- Taylor, M., Harlow, A. ve Forret, M. (2010). Using a computer programming environment and an interactive whiteboard to investigate some mathematical thinking. *Procedia-Social and Behavioral Sciences*, 8, 561-570.
- Tebliğler Dergisi (1997). Bilgisayar dersi öğretim programı. *Talim Terbiye Kurumu Başkanlığı*, <http://tebligler.meb.gov.tr/index.php/tuem-sayilar/viewcategory/61-1997?limitstart=0&order=date&dir=asc> adresinden 10 Aralık 2018 tarihinde erişilmiştir.
- Tebliğler Dergisi (2007). Bilişim teknolojileri dersi öğretim programı. *Talim Terbiye Kurumu Başkanlığı*, <http://tebligler.meb.gov.tr/index.php/tuem-sayilar/viewcategory/71-2007> adresinden 10 Aralık 2018 tarihinde erişilmiştir.
- Tebliğler Dergisi (2013). BTY dersi öğretim programı. *Talim Terbiye Kurumu Başkanlığı*, <http://tebligler.meb.gov.tr/index.php/tuem-sayilar/viewcategory/80> adresinden 10 Aralık 2018 tarihinde erişilmiştir.
- Tebliğler Dergisi (2017). BTY dersi öğretim programı. *Talim Terbiye Kurumu Başkanlığı*, <http://tebligler.meb.gov.tr/index.php/tuem-sayilar/viewcategory/85-2017> adresinden 10 Aralık 2018 tarihinde erişilmiştir.
- Tekin, H. (2000). *Eğitimde Ölçme ve Değerlendirme*. Ankara: Yargı Kitabevi.
- Weinberg, A. E. (2013). Computational thinking: An investigation of the existing scholarship and research. <http://people.cs.vt.edu/~kafura/CS6604/Papers/CT-Existing-Scholarship-Research-Dissertation.pdf> adresinden 7 Kasım 2018 tarihinde ulaşılmıştır.
- Weintrop, D. ve Wilensky, U. (2015). To block or not to block, that is the question: Students' perceptins of blocks-based programming. Paper presented at the *14th Annual IDC Conference*, Boston, MA.

- Wigfield, A. ve Wentzel, K.R. (2007). Introduction to motivation at school: interventions that work. *Educational Psychologist*, 42(4), 191-196.
- Wing, J. M. (2006). Computational thinking. *Communications of The ACM*, 49(3), 33-35.
- Yıldırım, A. ve Şimşek, H. (2006). *Sosyal Bilimlerde Nitel Araştırma Yöntemleri*. Seçkin Yayıncılık. S.227.
- Yıldız, M. ve Kaya, Z. (2013). Meslek liselerindeki programlama temelleri dersi programının değerlendirilmesi. *Journal of Research in Education and Teaching*, 2(2), 359-368.
- Yükseltürk, E. ve Altıok, S. (2015). Bilişim teknolojileri öğretmen adaylarının bilgisayar programlama öğretimine yönelik görüşleri. *Amasya Üniversitesi Eğitim Fakültesi Dergisi*, 4(1), 50-65.



EKLER

Ek-1: Etik Kurul İzni



T.C.
ONDOKUZ MAYIS ÜNİVERSİTESİ
SOSYAL VE BEŞERİ BİLİMLER ETİK KURUL KARARLARI

KARAR TARİHİ	TOPLANTI SAYISI	KARAR SAYISI
02.11.2018	9	2018 / 282

KARAR NO: 2018 - 282
Üniversitemiz Eğitim Bilimleri Enstitüsü yüksek lisans öğrencisi Ercan ÇİMŞİR'in Dr. Öğrt. Üyesi Polat ŞENDURUR danışmanlığında "Programlama Öğretiminde Bilgisayarsız Bilgisayar Bilimi Etkinliklerinin Öğrencilerin Akademik Başarılarına Etkisi ve Öğrenci Görüşleri" isimli yüksek lisans tezine ilişkin anket çalışması okunarak görüşüldü.

Üniversitemiz Eğitim Bilimleri Enstitüsü yüksek lisans öğrencisi Ercan ÇİMŞİR'in Dr. Öğrt. Üyesi Polat ŞENDURUR danışmanlığında "Programlama Öğretiminde Bilgisayarsız Bilgisayar Bilimi Etkinliklerinin Öğrencilerin Akademik Başarılarına Etkisi ve Öğrenci Görüşleri" isimli yüksek lisans tezine ilişkin anket çalışmasının kabulüne oybirliği ile karar verildi.

Ek-2: Kurum İzni



T.C.
GİRESUN ÜNİVERSİTESİ REKTÖRLÜĞÜ
Teknik Bilimler Meslek Yüksekokulu
Bilgisayar Teknolojileri Bölüm Başkanlığı

Sayı : 82551395-100-E.8420
Konu : Araştırma İzni

12.02.2019

Sayın Öğr. Gör. Ercan ÇİMSİR

İlgi : 12.02.2019 tarihli ve 82551395-100-E.8275 sayılı yazımız.

Lisansüstü eğitiminizde kullanılmak üzere yapmayı planladığımız araştırma için bölümümüz 1. sınıf öğrencilerinden veri toplamanızda herhangi bir sakınca yoktur.

Gereğini ve bilgilerinize rica ederim.

 e-imzalıdır.
Öğr. Gör. Ferhat BÜYÜKKALKAN
Bölüm Başkanı

Ek-3: Kişisel Bilgi Formu

KİŞİSEL BİLGİ FORMU

Yönerge: Katılacağınız bu araştırma, programlama öğretiminde bilgisayarsız bilgisayar bilimi etkinliklerinin kullanımının öğrenci başarısına etkisi konusunda Dr. Öğr. Üyesi Polat ŞENDURUR danışmanlığında Öğr. Gör. Ercan ÇİMŞİR tarafından yapılmaktadır. Araştırma ile ilgili vereceğiniz bilgiler kesinlikle gizli tutulacak, bu formla alınacak veriler yalnızca çalışma grubunu betimlemek amacıyla kullanılacaktır. Tüm veriler güvenli bir şekilde saklanacak ve katılımcılar izin vermedikleri sürece, sadece araştırmayı yürütenlerce ulaşılabilecektir. Sözlü ya da yazılı olarak hiç bir katılımcının ismi kullanılmayacak, isim yerine araştırmacı tarafından verilen kodlar kullanılacaktır. İstedığınız takdirde araştırma sonuçlarının birer kopyası tarafınıza verilecektir. Katılımınız için teşekkür ederiz.

1. Cinsiyetiniz : Kız Erkek
2. Yaşınız:
3. Daha önce programlama öğretimine katıldınız mı? Evet Hayır
4. Sizce programlama öğretiminde bilgisayarsız bilgisayar bilimi etkinliklerinin kullanılması başarınızı arttırır mı? Evet Hayır
5. Daha önce programlama öğretiminde bilgisayarsız bilgisayar bilimi etkinliklerine katıldınız mı, katıldıysanız hangi uygulamaları yaptınız?.....

Ek-4: Akademik Başarı Testi (40 soru)

PROGRAMLAMA SORULARI

1. Makine dili için aşağıdaki seçeneklerde belirtilen ifadelerden hangisi doğrudur?

- a) Yalnızca 0 ve 1 değerlerden oluşur.
- b) Programlama dillerinin sınıflandırılmasında orta düzeyde bulunan dillerdir.
- c) Bilgisayar tarafından anlaşılabilmesi için derleyiciler ve yorumlayıcılar bulunur.
- d) Bilgisayar tarafından anlaşılabilmesi için çeviriciler bulunur.
- e) İnsanlar tarafından en rahat anlaşılabilir seviyededir.

2. Aşağıdakilerden hangisi yüksek seviye programlama dillerinden değildir?

- a) Assembly
- b) C
- c) C++
- d) Python
- e) Java

3. Aşağıdakilerden hangisinde bir program komutunun çalıştırılması esnasında gerçekleşen adımlar doğru sırada verilmiştir?

- a) Çözme-Bulma-Çalıştırma
- b) Çözme-Çalıştırma-Bulma
- c) Bulma-Çalıştırma-Çözme
- d) Bulma-Çözme-Çalıştırma
- e) Çalıştırma-Çözme-Bulma

4. Tarihsel gelişimine göre programlama dillerinin seviyeleri hangi seçenekte doğru olarak verilmiştir?

- a) Makine Seviyesi-Orta Seviye-Yüksek Seviye
- b) Makine Seviyesi-Çevirici Seviyesi-Okuyucu Seviyesi
- c) Makine Seviyesi-Çevirici Seviyesi-Yüksek Seviye
- d) İlk Seviye-Orta Seviye-Yüksek Seviye
- e) Makine Seviyesi-Donanım Seviyesi-Yazılım Seviyesi

5. Bilgisayar hafızasında 2 byte alana en fazla kaç bitlik bir dizilim yazılabilir?

- a) 0
- b) 2
- c) 8
- d) 16
- e) 1024

6. 10'luk tabandaki 212 sayının 2'lik tabandaki karşılığı aşağıdakilerden hangisidir?

- a) 00101011
- b) 01100110
- c) 10110111
- d) 11010100
- e) 11010101

7. $(10110011)_2 = (\dots? \dots)_{10}$ dönüşümünün sonucu aşağıdakilerden hangisinde doğru olarak verilmiştir?

- a) 5
- b) 16
- c) 32
- d) 51
- e) 179

8. Yazılımcıların belirli bir algoritmayı ifade etmek için kullandığı, yapılacak işlemi bilgisayara anlatmaya yarayan standart gösterimlere ne ad verilir?

- a) Algoritma
- b) Yazılım
- c) Programlama Dili
- d) Merkezi İşlem Birimi
- e) İşletim Sistemi

9. I. BYTE

II. ASCII

III. EBCDIC

IV. UNICODE

Yukarıdakilerden hangisi ya da hangileri bilgisayar kodlama sistemine örnek verilebilir?

- a) Yalnız I
- b) I ve II
- c) I, II ve III
- d) II, III ve IV
- e) I, III ve IV

10. Algoritma ile ilgili aşağıdakilerden hangisi yanlıştır?

- a) Mantıksal sıralama gerekli
- b) Sonsuz olmalı
- c) Net olmalı
- d) Sınırları belli olmalı
- e) Programlamadan önce hazırlanmalı

11. Aşağıdakilerden hangisi veri tiplerinden değildir?

- a) float
- b) long
- c) int
- d) byte
- e) string

12. 1. Başla

2. A ve B değişkenlerini tanımla

3. Birinci sayı; A değerlerini gir

4. İkinci sayı; B değerlerini gir

5.?....

6. A değişkeninin değerini yazdır

7. Eğer $A < B$ ise 5. satıra git

8. Dur

"Klavyeden girilen birinci sayı ile ikinci sayı arasında kalan sayıları yazdıran" yukarıdaki algoritmada boş bırakılan 5. satıra aşağıdakilerden hangisi gelmelidir?

- a) $A=A+A$
- b) $A=A+B$
- c) $A+1=A$
- d) $A=A+1$
- e) $A=A-1$

13. 1. Başla

2. Sayac, Sayı, Toplam ve Ortalama değişkenlerini tanımla

3. Toplam=0, Sayac=0

4.?....

5. Eğer Sayı=-1 ise 9. satıra git

6. Toplam=Toplam + Sayı

7. Sayac=Sayac + 1

8. 4. satıra git

9. Ortalama=Toplam/Sayac

10. Ortalama değişkeninin değerini yazdır

11. Dur

"-1 deęeri girilinceye kadar klavyeden girilen sayıların ortalamasını bulup yazdıran" yukarıdaki algoritmada boş bırakılan 4. satıra aşağıdakilerden hangisinde gelmelidir?

- a) Sayi deęerini gir
- b) Sayi=Sayi+1
- c) Sayi=1
- d) Sayi=-1
- e) Sayac=Sayi+Sayac

14. 1. Başla

2. $X=0$, $Y=0$

3. Oku A

4. Giriş:

5. $X=X+1$

6. $Y=Y*X$

7. Eğer $X=A$ ise Y'yi yaz değilse 3. satıra git

8. Dur

Algoritmada A deęişkenine 3 atandığını düşündüğünüzde verilen ifadelerden doğru olanı aşağıdakilerden hangisinde belirtilmiştir?

- a) Y 3 olur
- b) Y 18 olur
- c) Y hiçbir deęer alamaz
- d) Y 0 olur
- e) Algoritmada yazım hatası vardır

15. 1. sayac=0, toplam=0

2. a deęerini gir.

3. Eğer $a < 0$ A2'ye git.

4. toplam=toplam+a

5. sayac=sayac+1

6.

7. Toplam deęerini yaz.

8. Bitir.

Yukardaki algoritmanın klavyeden girilen 5 adet pozitif sayının toplamını ekrana yazdırması için boş bırakılan satıra aşağıdaki seçeneklerden hangisi yazılmalıdır?

- a) Eğer sayac<6 A2'ye git
- b) Eğer sayac<5 A2'ye git
- c) Eğer sayac<5 A4'e git
- d) Eğer sayac>5 A4'e git
- e) Eğer sayac>5 A6'ya git

16. Simge veya sembollerle algoritmanın görsel olarak ifade edilmiş şekline ne ad verilir?

- a) Karar yapısı
- b) Akış şeması
- c) Görsel programlama
- d) Diyagram
- e) Çizelge

17. I. Karar yapısı

II. Tekrarlı yapı

III. Sıralı yapı


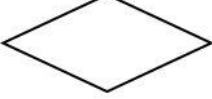

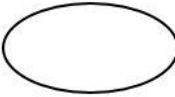

Yukarıdakilerden hangisi akış şemalarında kullanılan mantıksal yapılardandır?

- a) Yalnız I
- b) I, II
- c) I, III
- d) II, III
- e) I, II, III

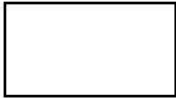
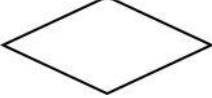

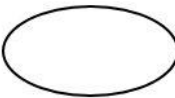

18. Akış şemaları için aşağıdakilerden hangisi yanlıştır?

- a) Her bir şeklin kendine özel anlamının bulunduğu şekillerdir.
- b) Mantıksal bir sıralama içerisinde sözle ifade edilen adımların simge veya şekil kullanılarak gösterilmesidir.
- c) Geçişler, oklar kullanılarak veya bağlantı sembolleriyle yapılmaktadır.
- d) Kullanılan semboller ve şekiller, uluslararası düzeyde farklı anlama sahiptir.
- e) Kod yazımı için ilk hazırlıkların yapılması için oluşturulur.

19. Aşağıdakilerden hangisi akış şemalarında veri veya bilgi girişini sağlar?

- a) 
- b) 
- c) 
- d) 
- e) 

20. Aşağıdaki akış şeması şekillerinden hangisi hesaplama ve atama işlemlerinin yapıldığı şekildir?

- a) 
- b) 
- c) 
- d) 
- e) 

21. Aşağıda verilen operatörlerden hangisi diğerlerinden farklı bir kategoriye aittir?

- a) <> b) + c) - d) / e) Mod

22. $(4+3)*4-6/2*3-(4*2)$ işleminin sonucu aşağıdakilerden hangisidir?

- a) 10 b) 11 c) 15 d) 22 e) 33

23. Aşağıda verilen değişkenlere atanan değerleri dikkate alarak doğru denklem sonucunun hangi seçenekte gösterildiğini bulunuz?

A=5, B=4 ise

$$X=A^2+5*(B-2)$$

X=?

- a) 5 b) 15 c) 30 d) 35 e) 60

24. Aşağıdaki kullanımlardan hangisiyle önce arttırma işlemi yapıp daha sonra atama işlemi yapılır?

- a) ++a b) a++ c) +a d) a+ e) Hepsi

25. Aşağıdaki değişken isimlerinden hangisi hatalıdır?

- a) ogrenciNo
- b) OgrenciAd
- c) ogrenci_soyad
- d) ogrenci sinif
- e) Hepsi

26. Aşağıda verilen değişkenlere atanan değerleri dikkate alarak denklemle ilgili olarak seçeneklerde verilen ifadelerden doğru olanı bulunuz?

A=2, B=8, X=3

B Mod A=0 And X²>=9

- a) B Mod A 0'a eşit değildir.
- b) X²>=9 ifadesi doğrudur.
- c) İşlem sonucu yanlıştır.
- d) B Mod A 1'e eşittir.
- e) And operatörünün sol tarafı yanlış sağ tarafı doğrudur.

27. A=2, B=4, K=3, L=5 ise aşağıdaki işlemlerden hangisinin sonucu "Yanlıştır"?

- a) K²>=9
- b) Not (B²<=A)
- c) A²=B Or K*2=L
- d) K Mod A=0 And B Mod L=0
- e) Not (B Mod A=1 And K²<9)

28. Aşağıdakilerden hangisi ilişkisel operatörler arasında yer almaz?

- a) ||
- b) <=
- c) !=
- d) >
- e) ==

29. Doğru ve yanlış (true ve false) olmak üzere sadece iki adet değer tutabilen boolean tipi değişkenlerin birbiri ile değerlendirilmesini sağlayan operatör türü aşağıdakilerden hangisidir?

- a) Aritmetik
- b) Mantıksal
- c) İlişkisel
- d) Atama
- e) Bit

30. 5 <= 3, 5 == 3 ve 5 != 3 ifadeleri hangi değerlerle sonuçlanır?

- a) false,false,true
- b) false,false,false
- c) false,true,true
- d) true,true,false
- e) true,false,false

31. Mantıksal operatörlerden biri olan AND operatörü için aşağıdakilerden hangisi doğrudur?

- a) Tek ifade ile kullanılabilir.
- b) En az üç ifade arasında kullanılmalıdır.
- c) İki ifadeden biri doğru biri yanlışsa sonuç doğru olur.
- d) İki ifadeden ikisi de yanlışsa sonuç doğru olur.
- e) İki ifadeden ikisi de doğruysa sonuç doğru olur.

32. İki değerin karşılaştırılmasında kullanılan operatör türü aşağıdakilerden hangisidir?

- a) Aritmetik Operatörleri
- b) Mantıksal Operatörleri
- c) İlişkisel Operatörleri
- d) Bit Operatörleri
- e) Atama Operatörleri

33. A=001110101 ve B=10001100 ise C = A | B işleminde C'nin değeri nedir?

- a) 10111101
- b) 00000100
- c) 01000110
- d) 10100001
- e) 00001101

34. Aşağıdakilerden hangisi bir döngü deyimi değildir?

- a) IF.....ELSE
- b) FOR
- c) FOREACH
- d) DO.....WHILE
- e) Hiçbiri

35. Aşağıda verilen for döngüsü tanımlamalarından hangisinde döngü sonsuz bir döngüye girer?

- a) for (int i=0; i<100; i++)
- b) for (int i=0; i<100; i--)
- c) for (int i=100; i>0; i--)
- d) for (int i=0; i<=100; i=i+5)
- e) for (int i=10; i>0; i--)

36. Aşağıda belirtilen anahtar kelimelerden hangisi, kategori olarak diğerlerinden farklı bir gruba aittir?

- a) For
- b) Double
- c) Break
- d) While
- e) Switch

37. İç-içe döngüler ile ilgili aşağıda verilen ifadelerden hangisi doğrudur?

- a) Bir if koşulu içerisinde do-while döngüsü kullanamayız.
- b) Hem do-while hem de while döngüsünü iç-içe kullanamayız.
- c) İç-içe kullanacağımız döngü sayılarında herhangi bir kısıtlama söz konusu değildir.
- d) İç-içe en fazla 2 adet döngü kullanabiliriz.
- e) Hepsi

38. For sayi as integer = -5 to -1 döngüsü çalıştırıldığında kaç kez döngü işlemi gerçekleşir?

- a) 1 kez
- b) 4 kez
- c) 5 kez
- d) 6 kez
- e) Hiç çalışmaz.

39. #include main () { int x=5; int y=2; while(x<8){ y=x+y; x++;} printf (" %d", y); } Yukardaki program kod parçası ekrana aşağıdakilerden hangisini yazdırır?

- a) 7
- b) 8
- c) 13
- d) 18
- e) 20

40. #include main () { int x = 5; int y = 2; if (x-y>0 || x%y<0){ printf (" %d", x+y);} else { printf (" %d", x*y);} } Yukarıdaki program kod parçası ekrana aşağıdakilerden hangisini yazdırır?

- a) 2
- b) 3
- c) 5
- d) 7
- e) 10

Ek-5: Akademik Başarı Testi (20 soru, Öntest, Sontest)

Bu formla toplanacak veriler isimleriniz saklı tutularak yalnızca araştırmada kullanılacak olup ders notunuza kesinlikle etki etmeyecektir.

AKADEMİK BAŞARI TESTİ

1. Makine dili için aşağıdaki seçeneklerde belirtilen ifadelerden hangisi doğrudur?

- a) Yalnızca 0 ve 1 değerlerden oluşur.
- b) Programlama dillerinin sınıflandırılmasında orta düzeyde bulunan dillerdir.
- c) Bilgisayar tarafından anlaşılabilmesi için derleyiciler ve yorumlayıcılar bulunur.
- d) Bilgisayar tarafından anlaşılabilmesi için çeviriciler bulunur.
- e) İnsanlar tarafından en rahat anlaşılabilir seviyededir.

2. Bilgisayar hafızasında 2 byte alana en fazla kaç bitlik bir dizilim yazılabilir?

- a) 0
- b) 2
- c) 8
- d) 16
- e) 1024

3. $(10110011)_2 = (\dots? \dots)_{10}$ dönüşümünün sonucu aşağıdakilerden hangisinde doğru olarak verilmiştir?

- a) 5
- b) 16
- c) 32
- d) 51
- e) 179

4. Algoritma ile ilgili aşağıdakilerden hangisi yanlıştır?

- a) Mantıksal sıralama gerekli
- b) Sonsuz olmalı
- c) Net olmalı
- d) Sınırları belli olmalı
- e) Programlamadan önce hazırlanmalı

5. 1. Başla

- 2. A ve B değişkenlerini tanımla
- 3. Birinci sayı; A değerlerini gir
- 4. İkinci sayı; B değerlerini gir
- 5.?....
- 6. A değişkeninin değerini yazdır
- 7. Eğer $A < B$ ise 5. satıra git
- 8. Dur

"Klavyeden girilen birinci sayı ile ikinci sayı arasında kalan sayıları yazdıran" yukarıdaki algoritmada boş bırakılan 5. satıra aşağıdakilerden hangisi gelmelidir?

- a) $A=A+A$
- b) $A=A+B$
- c) $A+1=A$
- d) $A=A+1$
- e) $A=A-1$

6. 1. Başla

- 2. Sayac, Sayı, Toplam ve Ortalama değişkenlerini tanımla
- 3. Toplam=0, Sayac=0
- 4.?....
- 5. Eğer Sayı=-1 ise 9. satıra git
- 6. Toplam=Toplam + Sayı
- 7. Sayac=Sayac + 1
- 8. 4. satıra git
- 9. Ortalama=Toplam/Sayac
- 10. Ortalama değişkeninin değerini yazdır
- 11. Dur

"-1 değeri girilinceye kadar klavyeden girilen sayıların ortalamasını bulup yazdıran" yukarıdaki algoritmada boş bırakılan 4. satıra aşağıdakilerden hangisinde gelmelidir?

- a) Sayı değerini gir
- b) Sayı=Sayı+1
- c) Sayı=1
- d) Sayı=-1
- e) Sayı=Sayı+Sayı

7. 1. Başla
2. X=0, Y=0
3. Oku A
4. Giriş:
5. X=X+1
6. Y=Y*X
7. Eğer X=A ise Y'yi yaz değilse 3. satıra git
8. Dur

Algoritmada A değişkenine 3 atandığını düşündüğünüzde verilen ifadelerden doğru olan aşağıdakilerden hangisinde belirtilmiştir?

- a) Y 3 olur
- b) Y 18 olur
- c) Y hiçbir değer alamaz
- d) Y 0 olur
- e) Algoritmada yazım hatası vardır

8. I. Karar yapısı
- II. Tekrarlı yapı
- III. Sıralı yapı


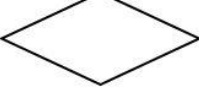
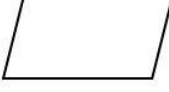

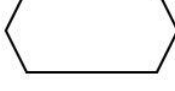
Yukarıdakilerden hangisi akış şemalarında kullanılan mantıksal yapılardandır?

- a) Yalnız I
- b) I, II
- c) I, III
- d) II, III
- e) I, II, III

9. Akış şemaları için aşağıdakilerden hangisi yanlıştır?

- a) Her bir şeklin kendine özel anlamının bulunduğu şekillerdir.
- b) Mantıksal bir sıralama içerisinde sözle ifade edilen adımların simge veya şekil kullanılarak gösterilmesidir.
- c) Geçişler, oklar kullanılarak veya bağlantı sembolleriyle yapılmaktadır.
- d) Kullanılan semboller ve şekiller, uluslararası düzeyde farklı anlama sahiptir.
- e) Kod yazımı için ilk hazırlıkların yapılması için oluşturulur.

10. Aşağıdakilerden hangisi akış şemalarında veri veya bilgi girişini sağlar?

- a) 
- b) 
- c) 
- d) 
- e) 

11. Aşağıda verilen operatörlerden hangisi diğerlerinden farklı bir kategoriye aittir?

- a) <>
- b) +
- c) -
- d) /
- e) Mod

12. Aşağıda verilen değişkenlere atanan değerleri dikkate alarak doğru denklem sonucunun hangi seçenekte gösterildiğini bulunuz?

A=5, B=4 ise

$$X=A^2+5*(B-2)$$

X=?

- a) 5 b) 15 c) 30 d) 35 e) 60

13. Aşağıda verilen değişkenlere atanan değerleri dikkate alarak denklemlerle ilgili olarak seçeneklerde verilen ifadelerden doğru olanı bulunuz?

$$A=2, B=8, X=3$$

$$B \text{ Mod } A=0 \text{ And } X^2 \geq 9$$

- a) B Mod A 0'a eşit değildir.
b) $X^2 \geq 9$ ifadesi doğrudur.
c) İşlem sonucu yanlıştır.
d) B Mod A 1'e eşittir.
e) And operatörünün sol tarafı yanlış sağ tarafı doğrudur.

14. A=2, B=4, K=3, L=5 ise aşağıdaki işlemlerden hangisinin sonucu "Yanlıştır"?

- a) $K^2 \geq 9$
b) Not ($B^2 \leq A$)
c) $A^2=B$ Or $K^2=L$
d) $K \text{ Mod } A=0$ And $B \text{ Mod } L=0$
e) Not ($B \text{ Mod } A=1$ And $K^2 < 9$)

15. Doğru ve yanlış (true ve false) olmak üzere sadece iki adet değer tutabilen boolean tipi değişkenlerin birbiri ile değerlendirilmesini sağlayan operatör türü aşağıdakilerden hangisidir?

- a) Aritmetik b) Mantıksal c) İlişkisel d) Atama e) Bit

16. Mantıksal operatörlerden biri olan AND operatörü için aşağıdakilerden hangisi doğrudur?

- a) Tek ifade ile kullanılabilir.
b) En az üç ifade arasında kullanılmalıdır.
c) İki ifadeden biri doğru biri yanlışsa sonuç doğru olur.
d) İki ifadeden ikisi de yanlışsa sonuç doğru olur.
e) İki ifadeden ikisi de doğruysa sonuç doğru olur.

17. Aşağıda belirtilen anahtar kelimelerden hangisi, kategori olarak diğerlerinden farklı bir gruba aittir?

- a) For b) Double c) Break d) While e) Switch

18. İç-içe döngüler ile ilgili aşağıda verilen ifadelerden hangisi doğrudur?

- a) Bir if koşulu içerisinde do-while döngüsü kullanamayız.
b) Hem do-while hem de while döngüsünü iç-içe kullanamayız.
c) İç-içe kullanacağımız döngü sayılarında herhangi bir kısıtlama söz konusu değildir.
d) İç-içe en fazla 2 adet döngü kullanabiliriz.
e) Hepsi

19. For sayı as integer = -5 to -1 döngüsü çalıştırıldığında kaç kez döngü işlemi gerçekleşir?

- a) 1 kez b) 4 kez c) 5 kez d) 6 kez e) Hiç çalışmaz.

20. #include main () { int x=5; int y=2; while(x<8){ y=x+y; x++;} printf (" %d", y); } Yukardaki program kod parçası ekrana aşağıdakilerden hangisini yazdırır?

- a) 7 b) 8 c) 13 d) 18 e) 20

Ek-6: Görüş Formu

Programlama Öğretiminde Bilgisayarsız Bilgisayar Bilimi Etkinliklerine Katılan Öğrenciler İçin Görüş Formu

Yönerge: Katılacağınız bu araştırma, programlama öğretiminde bilgisayarsız bilgisayar bilimi etkinliklerinin öğrencilerin akademik başarılarına etkisinin ve öğrenci görüşlerinin değerlendirilmesi konusunda Dr. Öğr. Üyesi Polat ŞENDURUR danışmanlığında Öğr. Gör. Ercan ÇİMŞİR tarafından yapılmaktadır. Araştırma ile ilgili vereceğiniz bilgiler kesinlikle gizli tutulacaktır. Tüm veriler güvenli bir şekilde saklanacak ve katılımcılar izin vermedikleri sürece, sadece araştırmayı yürütenlerce ulaşılabilecektir. Sözlü ya da yazılı olarak hiç bir katılımcının ismi kullanılmayacak, isim yerine araştırmacı tarafından verilen kodlar kullanılacaktır. İstedığınız takdirde araştırma sonuçlarının birer kopyası tarafınıza verilecektir. Sizden uygulamaya katıldığınız araştırmaya yönelik sorulara ait cevaplarınızı yazmanız beklenmektedir. Katılımınız için teşekkür ederiz.

1. Programlama öğretiminde bilgisayarsız bilgisayar bilimi etkinliklerinin uygulanması konuya dikkatinizi çekmeye yardımcı oldu mu? Sizde merak uyandırdı mı? Neden?

2. Programlama öğretiminde bilgisayarsız bilgisayar bilimi etkinlikleri sırasında arkadaşlarınızla olan etkileşiminiz hakkında ne düşünüyorsunuz?

3. Programlama öğretiminde bilgisayarsız bilgisayar bilimi etkinlikleri sırasında öğretmeninizle olan etkileşiminiz hakkında ne düşünüyorsunuz?

4. Bilgisayarsız bilgisayar bilimi etkinlikleri sırasında öğrendiklerinizi gerçek programlama ortamına aktarırken neler yaşadınız?

5. Bilgisayarsız bilgisayar bilimi etkinlikleri ile programlama kavramları arasındaki ilişkiyi kavrama konusunda zorluk yaşadınız mı? Örnek verebilir misiniz?

6. Öğrendiğiniz bilgisayar bilimi ve programlama kavramlarını dikkate aldığınızda, ileride bu kavramları hatırlama ve kullanma açısından bilgisayarsız bilgisayar bilimi etkinliklerinin ne gibi etkileri olacağını düşünüyorsunuz?

7. Bilgisayarsız bilgisayar bilimi etkinliklerinin programlama öğrenmenize katkısını nasıl değerlendirirsiniz? Bu etkinliklerin size en çok fayda sağladığı noktalar nelerdir?

8. Sizce programlama dersi bilgisayarsız bilgisayar bilimi etkinlikleri kullanılarak işlenirse geleneksel yönteme kıyasla daha etkili olur mu? Neden?

9. Programlama öğretiminde bilgisayarsız bilgisayar bilimi etkinliklerinin kullanılması hakkındaki önerileriniz nelerdir?

10. Bu sorular dışında eklemek istediğiniz bir nokta var mı? Varsa belirtiniz.