



**ÜSTÜN YETENEKLİ ÖĞRENCİLERİN SANAL VE
FİZİKSEL ROBOTİK KODLAMA
PERFORMANSLARININ, ÖZ-YETERLİKLERİNİN
VE KAYGILARININ İNCELENMESİ**

Nazmi SARIGÜL

Yüksek Lisans Tezi

**Bilgisayar ve Öğretim Teknolojileri Eğitimi Ana Bilim
Dalı**

2026

(Her hakkı saklıdır.)

T.C.
ATATÜRK ÜNİVERSİTESİ
EĞİTİM BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ EĞİTİMİ ANA BİLİM DALI
BİLGİSAYAR VE ÖĞRETİM TEKNOLOJİLERİ EĞİTİMİ BİLİM DALI

**ÜSTÜN YETENEKLİ ÖĞRENCİLERİN SANAL VE FİZİKSEL ROBOTİK
KODLAMA PERFORMANSLARININ, ÖZ-YETERLİKLERİNİN VE
KAYGILARININ İNCELENMESİ**

(Examining Gifted Students' Virtual and Physical Robotics Coding Performance, Self-Efficacy, and Anxiety)

YÜKSEK LİSANS TEZİ

Nazmi SARIGÜL

Danışman: Doç. Dr. Fatma Burcu TOPU

Erzurum
Ocak, 2026



T.C.
ATATÜRK ÜNİVERSİTESİ
Eğitim Bilimleri Enstitüsü

KABUL VE ONAY TUTANAĞI

Doç.Dr. Fatma Burcu TOPU danışmanlığında Nazmi SARIGÜL tarafından hazırlanan “**ÜSTÜN YETENEKLİ ÖĞRENCİLERİN SANAL VE FİZİKSEL ROBOTİK KODLAMA PERFORMANSLARININ, ÖZ-YETERLİKLERİNİN VE KAYGILARININ İNCELENMESİ**” başlıklı çalışma 23/01/2026 tarihinde aşağıdaki jüri tarafından Bilgisayar ve Öğretim Teknolojileri Eğitimi - Anabilim Dalı Bilgisayar ve Öğretim Teknolojileri Eğitimi Bilim Dalı'nda **Yüksek Lisans** tezi olarak oy birliği ile kabul edilmiştir.

Jüri Başkanı:	Prof.Dr. ÜNAL ÇAKIROĞLU <i>TRABZON ÜNİVERSİTESİ</i>	e-imza
Jüri Üyesi (Danışman):	Doç.Dr. Fatma Burcu TOPU <i>ATATÜRK ÜNİVERSİTESİ</i>	e-imza
Jüri Üyesi:	Dr. Öğr. Üyesi Ömer ARPACIK <i>ATATÜRK ÜNİVERSİTESİ</i>	e-imza

Bu tezin Atatürk Üniversitesi Lisansüstü Eğitim ve Öğretim Yönetmeliği'nin ilgili maddelerinde belirtilen şartları yerine getirdiğini onaylarım.

23/01/2026
Prof.Dr. Refik DİLBER
Enstitü Müdürü
e-imza



ETİK VE BİLDİRİM SAYFASI

Yüksek Lisans Tezi olarak sunduğum “Üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlama performanslarının, öz-yeterliklerinin ve kaygılarının incelenmesi” başlıklı çalışmanın tarafımdan bilimsel etik ilkelere uyularak yazıldığını ve yararlandığım eserleri kaynakçada gösterdiğimi beyan ederim.

... / ... / 20..

Aslı ıslak imzalıdır

Nazmi SARIGÜL

☐ Tezle ilgili patent başvurusu yapılması / patent alma sürecinin devam etmesi sebebiyle Enstitü Yönetim Kurulunun/....../.... tarih ve sayılı kararı ile teze erişim 2 (iki) yıl süreyle engellenmiştir.

☐ Enstitü Yönetim Kurulunun/....../.... tarih ve sayılı kararı ile teze erişim 6 (altı) ay süreyle engellenmiştir.

TEŞEKKÜR

Bu tez çalışmasının ortaya çıkmasında desteklerini esirgemeyen tüm kişi ve kurumlara en içten teşekkürlerimi sunarım. Öncelikle, tez sürecinin her aşamasında bilgisi, yönlendirmesi ve yapıcı katkılarıyla bana rehberlik eden değerli danışmanım Doç. Dr. Fatma Burcu TOPU hocama teşekkür ederim. Akademik bakış açımın gelişmesinde büyük pay sahibi olmuştur.

Araştırmamın uygulanma sürecinde destek sağlayan Tatvan Bilim ve Sanat Merkezi yönetimine, meslektaşlarıma ve derslere katılarak çalışmanın yürütülmesine katkı sunan tüm üstün yetenekli öğrencilerime teşekkür ederim. Onların ilgisi, emeği ve katılımı bu çalışmanın gerçekleşmesini mümkün kılmıştır. Bana her zaman sabır, anlayış ve motivasyon sağlayan aileme minnettarlığımı ifade etmek isterim. Akademik yolculuğum boyunca verdikleri destek, bu çalışmayı tamamlamamda en büyük güç kaynaklarımdan biri olmuştur. Son olarak, araştırma sürecinde görüşlerini paylaşarak katkı sunan tüm akademisyenlere ve arkadaşlarıma teşekkür ederim.

Bu çalışma, tüm bu değerli desteklerin bir ürünü olarak ortaya çıkmıştır.

Nazmi SARIGÜL

ÖZ

YÜKSEK LİSANS TEZİ

ÜSTÜN YETENEKLİ ÖĞRENCİLERİN SANAL VE FİZİKSEL ROBOTİK KODLAMA PERFORMANSLARININ, ÖZ-YETERLİKLERİNİN VE KAYGILARININ İNCELENMESİ

Nazmi SARIGÜL

Ocak 2026, 189 Sayfa

Amaç: Bu çalışmada Bilim ve Sanat Merkezinde öğrenim gören üstün yetenekli 6. ve 7. sınıf öğrencilerinin 10 haftalık sanal ve fiziksel robotik kodlama performansları, meşguliyetleri, Python programlama dili başarıları ve bilgilerinin kalıcılığı (hatırlanması), öz-yeterlikleri ve kaygıları araştırılmıştır. Ayrıca her iki gruptaki öğrencilerin robotik kodlama başarıları, öz-yeterlikleri ve kaygıları arasındaki ilişkiler incelenmiş, öğrencilerin sanal ve fiziksel robotik kodlama ortamlarındaki Python programlama dili öğrenme deneyimlerine ilişkin görüşleri ortaya çıkarılmıştır.

Yöntem: Bu çalışmada karma araştırma desenlerinden çeşitleme kullanılmıştır. Çalışma grubu, sanal (VEXcode-VR) ve fiziksel (Raspberry-Pi) robotik kodlama gruplarında yer alan ortaokul seviyesindeki toplam 20 üstün yetenekli öğrencisinden oluşmuştur. 10 haftalık uygulama sürecinde; veri toplama aracı olarak haftalık robotik kodlama hedef-kazanım değerlendirme kontrol listeleri ve meşguliyet gözlem formu, programlama başarı testi, öz-yeterlik ölçeği, kaygı ölçeği ve yarı yapılandırılmış görüşme rehberi kullanılmıştır.

Bulgular: Her iki gruptaki öğrencilerin 10 haftalık uygulama süresince robotik kodlama etkinliklerine aktif katıldıkları, çoğunlukla olumlu deneyimler yaşadıkları, haftalık performanslarının ve tüm meşguliyet boyutlarının (duyuşsal, bilişsel, davranışsal, temsili ve sosyal) benzer seviyede ve yüksek düzeyde olduğu belirlenmiştir. Her iki grupta da Python programlama dili başarıları ön testten son teste ve kalıcılık testine anlamlı biçimde artış göstermiştir. Ancak hem son test, hem kalıcılık testi puanları açısından gruplar arasında sanal robotik kodlama grubu lehine anlamlı düzeyde daha fazla artış görülmüştür. Uygulama sonunda her iki grubun programlama öz-yeterlik ortalamalarında yükselme ve programlama kaygısı ortalamalarında azalma görülse de bu değişimin gruplar arası anlamlı düzeyde olmadığı belirlenmiştir. Ayrıca fiziksel robotik kodlama grubunda başarı ile kaygı arasında güçlü ve negatif bir ilişki ortaya çıkmıştır. Öğrenci deneyimleri; öğrenme çıktısı, bilişsel süreçler, duyuşsal tepkiler, davranışsal süreçler, temsili/aracı katılım ve sosyal etkileşim boyutlarında kategorilendirilmiş, her iki grupta da bu boyutlardaki temalara ilişkin olumlu görüşlerin çoğunlukta olduğu belirlenmiştir.

Sonuçlar: Her iki grupta da her ne kadar olumlu bulgular ortaya çıksa da, sonuçlar fiziki araç ihtiyacı olmadan tekrar tekrar kullanılabilme, erişim kolaylığı ve fırsat eşitliği sunma gibi avantajları sayesinde sanal robotik kodlama araçlarının fiziksel robotik kodlama araçlarına kıyasla; üstün yetenekli öğrencilerin programlama dili öğrenme sürecinde daha fazla kullanılabileceğini göstermektedir.

Anahtar Kelimeler: sanal robotik kodlama, fiziksel robotik kodlama, Python programlama, öğrenme kalıcılığı, üstün yetenekli öğrenciler

ABSTRACT

MASTER'S THESIS

EXAMINING GIFTED STUDENTS' VIRTUAL AND PHYSICAL ROBOTICS CODING PERFORMANCE, SELF-EFFICACY, AND ANXIETY

Nazmi SARIGÜL

January 2026, 189 Pages

Purpose: This study investigated the virtual and physical robotics coding performance, engagement, Python programming language achievement and retention (recall), self-efficacy, and anxiety of gifted 6th and 7th-grade students studying at a Science and Art Center (BİLSEM) over a 10-week period. Furthermore, the relationships between robotics coding achievement, self-efficacy, and anxiety in both groups were examined, and students' views on their Python programming language learning experiences in virtual and physical robotics coding environments were revealed.

Method: In this study, the triangulation was conducted as a mixed research design. The study group consisted of a total of 20 gifted secondary school students participating in virtual (VEXcode-VR) and physical (Raspberry-Pi) robotics coding groups. During the 10-week implementation period, weekly robotics coding goal-acquisitions assessment checklists and engagement observation forms, the programming achievement test, the self-efficacy scale, the anxiety scale, and the semi-structured interview guide were used as data collection tools.

Findings: It was determined that students in both groups actively participated in robotics coding activities during the 10-week intervention period, mostly had positive experiences, and their weekly performance and all engagement dimensions (affective, cognitive, behavioral, agentic, and social) were at similar and high levels. In both groups, Python programming language success increased significantly from the pre-test to the post-test and retention test. However, in terms of both post-test and retention test scores, a significantly greater increase was observed between groups in favor of the virtual robotics coding group. Although both groups showed an increase in programming self-efficacy average and a decrease in programming anxiety average at the end of the intervention, this change was not statistically significant between the groups. Furthermore, a strong and negative correlation was found between achievement and anxiety in the physical robotics coding group. Student experiences were categorized into learning outcomes, cognitive processes, affective responses, behavioral processes, agentic participation, and social interaction dimensions. In both groups, positive opinions regarding themes in these dimensions were predominant.

Conclusions: Although positive findings emerged in both groups, the results indicate that virtual robotic coding tools, due to advantages such as reusability without the need for physical tools, ease of access, and equal opportunity, can be used more frequently by gifted students in their programming language learning process compared to physical robotic coding tools.

Keywords: virtual robotic coding, physical robotic coding, Python programming, learning retention, gifted students

İÇİNDEKİLER

KABUL VE ONAY TUTANAĞI.....	i
ETİK VE BİLDİRİM SAYFASI.....	ii
TEŞEKKÜR	iii
ÖZ.....	iv
ABSTRACT	v
İÇİNDEKİLER.....	viii
TABLolar DİZİNİ.....	viii
ŞEKİLLER DİZİNİ	x
BİRİNCİ BÖLÜM.....	1
Giriş	1
Araştırmanın Amacı	3
Araştırmanın Önemi ve Gerekçesi	4
Sınırlılıklar	8
Varsayımlar	9
Terim ve Tanımlar.....	9
İKİNCİ BÖLÜM	10
Kuramsal Çerçeve ve İlgili Araştırmalar	10
Eğitsel Robotik Kodlama Uygulamaları ve Programlama Öğretimi	10
Robotik Kodlama ve Programlama Başarısı	13
Robotik Kodlama ve Programlama Öz-Yeterliği.....	14
Robotik Kodlama ve Programlama Kaygısı	15
Eğitimde Robotik Kodlama Uygulamaları ile İlgili Araştırmalar.....	16
ÜÇÜNCÜ BÖLÜM.....	22
Yöntem	22
Araştırma Deseni.....	22
Çalışma Grubu	23
Veri Toplama Araçları	24
Programlama Başarı Testi	24
Programlama Öz-Yeterlik Ölçeği	36
Programlama Kaygı Ölçeği.....	36
Haftalık Robotik Kodlama Kazanım Değerlendirme Kontrol Listeleri	37

Haftalık Robotik Kodlama Meşguliyet Gözlem Formu.....	37
Görüşme Rehberi	38
Uygulama Süreci.....	38
Uygulama Öncesi.....	39
Uygulama Sırası.....	44
Uygulama Sonrası	44
Veri Analizi.....	45
Nicel Veri Analizi	46
Nitel Veri Analizi.....	47
Araştırmacının Rolü.....	47
Çalışmanın Geçerlik ve Güvenirliği.....	48
DÖRDÜNCÜ BÖLÜM.....	50
Bulgular	50
Üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlama performansları arasında anlamlı bir fark var mıdır?	50
Üstün yetenekli öğrencilerin haftalık sanal ve fiziksel robotik kodlama kazanımları arasında anlamlı bir fark var mıdır?	50
Üstün yetenekli öğrencilerin haftalık sanal ve fiziksel robotik kodlama meşguliyetleri arasında anlamlı bir fark var mıdır?	51
Grupların Duyuşsal Meşguliyetleri Arasındaki Fark	52
Grupların Bilişsel Meşguliyetleri Arasındaki Fark	54
Grupların Davranışsal Meşguliyetleri Arasındaki Fark	55
Grupların Temsili Meşguliyetleri Arasındaki Fark.....	57
Grupların Sosyal Meşguliyetleri Arasındaki Fark	58
Grupların Genel Meşguliyetleri Arasındaki Fark	60
Üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlamadaki Python programlama dili başarıları arasında anlamlı bir fark var mıdır?.....	62
Grup İçi Python Programlama Dili Başarıları Arasındaki Fark.....	62
Grupların Python Programlama Dili Başarıları Arasındaki Fark.....	63
Üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlamadaki Python programlama dili bilgilerinin kalıcılığı (hatırlanması) arasında anlamlı bir fark var mıdır?	64
Grup İçi Python Programlama Dili Bilgilerinin Kalıcılığı (Hatırlanması) Arasındaki Fark	64
Grupların Python Programlama Dili Bilgilerinin Kalıcılığı (Hatırlanması) Arasındaki Fark	65

Üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlama öz-yeterlikleri arasında anlamlı bir fark var mıdır?	65
Grup İçi Robotik Kodlama Öz-Yeterlikleri Arasındaki Fark	65
Grupların Robotik Kodlama Öz-Yeterlikleri Arasındaki Fark	66
Üstün Yetenekli Öğrencilerin Sanal Ve Fiziksel Robotik Kodlama Kaygıları Arasında Anlamlı Bir Fark Var Mıdır?	67
Grup İçi Robotik Kodlama Kaygıları Arasındaki Fark.....	67
Grupların Robotik Kodlama Kaygıları Arasındaki Fark.....	67
Üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlama performansları, öz-yeterlikleri ve kaygıları arasında anlamlı bir ilişki var mıdır?	68
Üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlama sürecindeki Python programlama dili öğrenme deneyimlerine ilişkin görüşleri nelerdir?	69
Öğrenme Çıktısı	71
Bilişsel Süreçler	74
Duyuşsal Tepkiler	76
Davranışsal Süreçler.....	79
Temsili/Aracı Katılım	80
Sosyal Etkileşim.....	82
BEŞİNCİ BÖLÜM	84
Tartışma ve Sonuç	84
Üstün Yetenekli Öğrencilerin Sanal ve Fiziksel Robotik Kodlama Performansları.....	84
Üstün Yetenekli Öğrencilerin Haftalık Sanal ve Fiziksel Robotik Kodlama Kazanımları	84
Üstün Yetenekli Öğrencilerin Haftalık Sanal ve Fiziksel Robotik Kodlama Meşguliyetleri	85
Üstün Yetenekli Öğrencilerin Sanal ve Fiziksel Robotik Kodlamadaki Python Programlama Dili Başarıları	87
Üstün Yetenekli Öğrencilerin Sanal ve Fiziksel Robotik Kodlamadaki Python Programlama Dili Bilgilerinin Kalıcılığı (Hatırlanması)	88
Üstün Yetenekli Öğrencilerin Sanal ve Fiziksel Robotik Kodlama Öz-Yeterlikleri	90
Üstün Yetenekli Öğrencilerin Sanal ve Fiziksel Robotik Kodlama Kaygıları	91
Üstün Yetenekli Öğrencilerin Sanal ve Fiziksel Robotik Kodlama Performansları, Öz-Yeterlikleri ve Kaygıları Arasındaki İlişki.....	93
Üstün Yetenekli Öğrencilerin Sanal ve Fiziksel Robotik Kodlama Sürecindeki Python Programlama Dili Öğrenme Deneyimleri	94

Öneriler	103
Bilim ve Sanat Merkezi Öğretmenlerine/ Uygulayıcılara Yönelik Öneriler	103
Milli Eğitim Bakanlığı Yetkililerine Yönelik Öneriler	103
Araştırmacılara Yönelik Öneriler	104
KAYNAKÇA	105
EKLER	117
EK-1. Sanal ve Fiziksel Robotik Kodlama Grupları için Programlama Başarı Testleri (Örnek Sorular)	117
EK-2. Programlama Öz-Yeterlik Ölçeği (Örnek Maddeler)	123
EK-3. Programlama Kaygı Ölçeği (Örnek Maddeler)	124
EK-4. Haftalık Robotik Kodlama Kazanım Değerlendirme Kontrol Listeleri (Örnek Maddeler)	125
EK-5. Haftalık Robotik Kodlama Meşguliyet Gözlem Formu (Örnek Maddeler)	127
EK-6. Görüşme Rehberi (Örnek Sorular)	128
EK-7. Raspberry-Pi Etkinlik Planları	129
EK-8. VEXcode-VR Etkinlik Planları	145
EK-9. Sanal ve Fiziksel Robotik Kodlama Gruplarının Uygulama Görüntüleri	165
EK-10. Araştırma Uygulama İzni	167
EK-11. Eğitsel Robotik Kodlama Çalışmaları	168
ÖZ GEÇMİŞ	175

TABLÖLAR DİZİNİ

Tablo 1. <i>Veri Toplama Araçlarına ve Cinsiyete göre Her İki Gruptaki Öğrenci Dağılımı</i>	24
Tablo 2. <i>Sanal ve Fiziksel Robotik Kodlama Başarı Testlerinin Uzman Görüşü Uyuşma Oranı, Kapsam Geçerlik Oranı ve İndeksi.....</i>	26
Tablo 3. <i>Python Programlama Dili Kazanımlarına göre Sanal ve Fiziksel Robotik Kodlama Başarı Testlerindeki Soru Dağılımı</i>	29
Tablo 4. <i>Pilot Uygulamalara Katılan Öğrencilerin Demografik Özellikleri</i>	30
Tablo 5. <i>Pilot Uygulama-I Verilerine İlişkin Sanal ve Fiziksel Robotik Programlama Başarı Testlerinin Madde Analizleri</i>	31
Tablo 6. <i>Pilot Uygulama-II Verilerine İlişkin Sanal ve Fiziksel Robotik Programlama Başarı Testlerinin Madde Analizleri</i>	33
Tablo 7. <i>Pilot Uygulama I ve II için Kuder-Richardson (KR-20) Güvenirlilik Değerleri.....</i>	34
Tablo 8. <i>Bloom Taksonomisi Bilişsel Süreç Boyutlarına Göre Başarı Testi Madde Dağılımı</i>	35
Tablo 9. <i>10 Haftalık Uygulama Takvimine Göre Sanal ve Fiziksel Robotik Kodlama Problem Durumları, Ortak Konular ve Kazanımlar.....</i>	40
Tablo 10. <i>Araştırma Sorularına Göre Kullanılan Veri Toplama Araçları ve Analiz Teknikleri</i>	46
Tablo 11. <i>Çalışmada Alınan Geçerlik ve Güvenirlilik Önlemleri</i>	49
Tablo 12. <i>Meşguliyet Alt Boyutlarına Göre Grupların Haftalık Ortalamaları</i>	51
Tablo 13. <i>Grupların Haftalık ve 10 Haftalık Duyuşsal Meşguliyetleri Arasındaki Fark.....</i>	53
Tablo 14. <i>Grupların Haftalık ve 10 Haftalık Bilişsel Meşguliyetleri Arasındaki Fark.....</i>	55
Tablo 15. <i>Grupların Haftalık ve 10 Haftalık Davranışsal Meşguliyetleri Arasındaki Fark ...</i>	56
Tablo 16. <i>Grupların Haftalık ve 10 Haftalık Temsili Meşguliyetleri Arasındaki Fark.....</i>	58
Tablo 17. <i>Grupların Haftalık ve 10 Haftalık Sosyal Meşguliyetleri Arasındaki Fark</i>	59
Tablo 18. <i>Grupların Haftalık ve 10 Haftalık Genel Meşguliyetleri Arasındaki Fark</i>	61
Tablo 19. <i>Ölçüm Zamanına Göre Grupların Robotik Kodlama Başarı Ortalamaları.....</i>	62
Tablo 20. <i>Grup İçİ Başarı Ön Test ve Son Teste İlişkin Wilcoxon İşaretli Sıralar Testi Sonuçları</i>	63
Tablo 21. <i>Gruplar Arası Başarı Ön Test ve Son Teste İlişkin Mann-Whitney U Sonuçları</i>	63
Tablo 22. <i>Grup İçİ Bilginin Kalıcılığına (Hatırlanmasına) İlişkin Wilcoxon İşaretli Sıralar Testi Sonuçları</i>	64

Tablo 23. <i>Gruplar Arası Bilginin Kalıcılığına (Hatırlanmasına) İlişkin Mann-Whitney U Sonuçları</i>	65
Tablo 24. <i>Grup İçi Öz-Yeterlik Ön Test ve Son Teste İlişkin Wilcoxon İşaretli Sıralar Testi Sonuçları</i>	66
Tablo 25. <i>Gruplar Arası Öz-Yeterlik Ön Test ve Son Teste İlişkin Mann-Whitney U Sonuçları</i>	66
Tablo 26. <i>Grup İçi Kaygı Ön Test ve Son Teste İlişkin Wilcoxon İşaretli Sıralar Testi Sonuçları</i>	67
Tablo 27. <i>Gruplar Arası Kaygı Ön Test ve Son Teste İlişkin Mann-Whitney U Sonuçları</i>	68
Tablo 28. <i>Gruplara Göre Değişkenler Arası İlişkiye Yönelik Spearman Sıra Farkları Korelasyon Test Sonuçları</i>	68
Tablo 29. <i>Üstün Yetenekli Öğrencilerinin Sanal ve Fiziksel Robotik Kodlama Sürecindeki Deneyimleri</i>	69

ŞEKİLLER DİZİNİ

Şekil 1. Programlama Kaygısını Etkileyen Faktörler	16
Şekil 2. 2014-2025 Yılları Arasında Eğitimde Robotik Uygulamaların Yıllara Göre Dağılımı	17
Şekil 3. Eğitimde Robotik Uygulamalarda Yaygın Olarak İncelenen Değişkenler	18
Şekil 4. Araştırmalarda Öğretimi Gerçekleştirilen Programlama Dilleri	18
Şekil 5. Araştırmalarda Kullanılan Robotik Araçlar	19
Şekil 6. Çalışma Kapsamı	21
Şekil 7. Çalışmada Temel Alınan Araştırma Deseni	22
Şekil 8. Başarı Testi Geliştirme Süreci	24
Şekil 9. Uygulama Süreci	45
Şekil 10. Grupların Haftalık Kazanım Değerlendirme Ortalamaları	50
Şekil 11. Grupların 10 Haftalık Meşguliyet Düzeyleri	52
Şekil 12. Grupların Haftalık ve 10 Haftalık Duyuşsal Meşguliyet Ortalamaları	53
Şekil 13. Grupların Haftalık ve 10 Haftalık Bilişsel Meşguliyet Ortalamaları	54
Şekil 14. Grupların Haftalık ve 10 Haftalık Davranışsal Meşguliyet Ortalamaları	56
Şekil 15. Grupların Haftalık ve 10 Haftalık Temsili Meşguliyet Ortalamaları	57
Şekil 16. Grupların Haftalık ve 10 Haftalık Sosyal Meşguliyet Ortalamaları	59
Şekil 17. Grupların Haftalık ve 10 Haftalık Genel Meşguliyet Ortalamaları	60

BİRİNCİ BÖLÜM

Giriş

Bireylerin hızla değişen ve gelişen hayat koşullarında yaşamdan beklentileri de değişmektedir. Öğrencilerin bu dünyaya ayak uydurmaları için yeni yeterliklere sahip olmaları ve çeşitli 21. yy becerilerini kazanmaları, kaçınılmaz bir ihtiyaç olarak görülmektedir (ISTE, 2016). Bu ihtiyaçlar eğitim sistemini dinamik tutmak için teknoloji destekli öğrenme-öğretme sürecinde birtakım değişiklikleri gerekli kılmıştır (AASL, 2018; ISTE, 2016; OECD, 2017; Partnership for 21st Century Skills, 2018). Bu beceriler arasında; öğrencilerin öğrenme süreçlerinde eğitsel robotikler gibi teknolojik araçları kullanarak edindiği disiplinler arası bilgi ve becerilere sahip olmayı gerektiren dijital yetkinlikler de yer almaktadır (Trilling & Fadel, 2009). Bu yetkinlikleri kazandırmayı hedefleyen Türkiye Yüzyılı Maarif Modeli (MEB, 2025) de, eğitim süreçlerini zenginleştirmek üzere disiplinler arası yaklaşımların kullanılması gerektiği benimsenerek hazırlanmıştır.

Endüstri 4.0 kavramıyla birlikte nesnelerin interneti, bulut bilişim, yapay zekâ, otonom araçlar ve robotik uygulamalar gibi aslında toplumun yabancı olduğu farklı kavramlar da hayatımıza girmeye başlamış ve bu kavramlar ile programlama becerisi de gün geçtikçe önem kazanmıştır. Günümüzde programlama becerisi sadece bilişim ve yazılım alanlarında değil eğitim, sağlık, güvenlik, eğlence ve iletişim gibi birçok alanda kullanılan teknolojinin temelinde de bulunmaktadır (Hangün vd., 2022). Eğitim-öğretim sistemleri de doğası gereği insanları yakından ilgilendiren bu gelişmeleri görmezden gelemeyerek yeni çalışmaların tohumunun atılmasına öncülük etmiştir (Kroustalli & Xinogalos, 2021). Temel seviyede de olsa yazılım geliştirebilmeyi amaçlayan kodlama ya da programlama becerisi, 21. yüzyılda eğitim kurumlarının önem verdiği, öğrencilerin çağa ayak uydurması adına sahip olunması gereken beceriler arasında gösterilmektedir (Mészárosová, 2015).

Milli Eğitim Bakanlığının (MEB) hazırladığı Bilişim Teknolojileri ve Yazılım Dersi Öğretim Programında (MEB, 2018), ortaokul düzeyindeki öğrencilerin bilgisayar bilimine ilişkin temel seviyede teknik bilgi edinmeleri ve beceri kazanmaları önemsenmektedir. Bu kapsamda akıl yürütme süreçlerini takip ederek algoritma kurabilmeleri, problem çözme, bilgi işlemsel düşünme becerilerini geliştirmeleri, problemleri çözmek için uygun programlama yaklaşımlarını seçip uygulayabilmeleri ve programlama dillerinden en az birini etkin şekilde kullanabilmeleri hedeflenmektedir. Ayrıca Millî Eğitim Bakanlığı hazırladığı Bilgisayar Bilimleri Dersi Öğretim Programında (MEB, 2023), metin tabanlı programlama dili öğretimini

somutlařtırmak iin eřitli ğretim yntemleri ve uygulama yazılımları kullanılarak ortağretim dzeyindeki ğrencilerin kodlama becerilerini geliřtirmeyi nemsenmektedir. Bu kapsamda ğrencilerin grece sade ve daha az sz dizimine sahip olan metin tabanlı Python programlama dilini (Mészárosová, 2015; Tran vd., 2024) temel seviyede kullanabilmeleri hedeflenmektedir.

Programlama dili ğretiminde programlama kaygısı, bilgisayar programlama ğreniminin nndeki nemli engellerden biridir. Dolayısıyla ğrencilerin kaygılarını azaltmak iin, kaygıyı artıran etkenleri bilerek buna zm retmek nemli bir konu olmaktadır (Glbahar vd., 2019). Programlama eğitiminde yeniliki ğretim yntemlerinin ve aralarının kullanılması, ğrencilerin meřguliyetini artırarak srece aktif katılmalarını ve programlama eğitiminin daha verimli olmasını saėlamaktadır (Ykseltrk & Altıok, 2015). ğrencilerin programlama kaygılarını azaltmak ve programlama z-yeterliklerini geliřtirmek amacıyla sanal (rnek: VEXcode VR) ve fiziksel (rnek: Raspberry Pi) robotik kodlama araları kullanılabilmektedir. Bu eğitsel robotik aralar; bir takım farklı grevleri yerine getirmek iin uyumlu paralar ile tasarlanmış programlanabilir, otonom olarak alışan aralar olarak tanımlanmaktadır (Uslu vd., 2022).

Eğitsel robotikler, ğrenenlerin blok, metinsel veya hibrit programlama dillerini ve aralarını kullanarak kendi tasarladıkları makineleri programlayabilmesi olarak da tanımlanabilir (Zhong & Li, 2019). Eğitsel robotikler bu ynyle birok eğitim kurumunda yeniliki bir ğrenme yntemi olarak tanıtılmakta, ğrencilerin ğrenme srecindeki meřguliyetini artırmalarına, st dzey dřnme becerilerinin yanında programlama yeteneklerini geliřtirmelerine ve bireylerin daha karmařık ve spesifik sorunları zmelerine yardımcı olmaktadır (Atmatzidou & Demetriadi, 2016). Eğitsel robotikler ile programlama ğretimi; robotik kodlama becerisinin yanı sıra iř birliėi, problem zme, bilgi iřlemsel dřnme, algoritmik dřnme ve yaratıcılık gibi birok becerinin gelişimini desteklemektedir (Hamiti vd., 2021; Kk & Snmez, 2023; Zhang vd., 2023). Bu robotik aralarla yapılan etkinlikler, katılımcıların programlamaya karřı tutumlarını ve programlama z-yeterliklerini artırmakta (Erol, 2020), programlama ğrenmelerini kolaylařtırmakta ve zevkli kılmaktadır (Sucu & akıroėlu, 2022). Bylece eğitsel robotiklerin programlama eğitiminde kullanımı; ğrencilerin ilgisini ve motivasyonunu artırarak ğrenme ortamıyla meřguliyetini yoėunlařtırmaya, daha verimli ğretimler gerekleřmesine de fırsat vermektedir (Zhong & Xia, 2020).

Bu alışmanın arařtırmacısı, Tatvan Bilim ve Sanat Merkezinde Teknoloji ve Tasarım ğretmeni olarak grev yapmakta ve stn yetenekli bireylerden oluřan ortaokul ğrencilerine

yönelik robotik kodlama atölyelerinde programlama ve robotik eğitimleri vermektedir. Derslerde genellikle Arduino robotik eğitim setleri kullanılmakta olup, bu setler aracılığıyla öğrencilere C++ tabanlı ve blok tabanlı programlama öğretimi gerçekleştirilmektedir. Ancak C++ tabanlı programlama dili, özellikle ilkokul ve ortaokul düzeyindeki öğrenciler için yazım kuralları ve kullanım karmaşıklığı açısından çeşitli güçlükler doğurmaktadır. Bu nedenle, temel düzeyde metin tabanlı bir programlama dili öğrenme sürecinde, daha sade sözdizimine sahip ve gelecekte robotik kodlama etkinliklerinde kullanım potansiyeli yüksek olan Python programlama dilinin öğretiminin, öğrencilerin programlama becerilerini geliştirmede daha uygun olacağı Milli Eğitim Bakanlığı tarafından öngörülmüştür. Bu çalışmada da üstün yetenekli öğrencilerin Python programlama dili performanslarını artırmaya, öz-yeterliğini geliştirmeye, kaygılarını azaltmaya yönelik olarak sanal ve fiziksel robotik etkinliklere yer verilmiştir.

Araştırmanın Amacı

Bu çalışmanın amacı; Bilim ve Sanat Merkezinde öğrenim gören üstün yetenekli 6. ve 7. sınıf öğrencilerinin sanal ve fiziksel robotik kodlama performanslarını, öz-yeterliklerini ve kaygılarını araştırmaktır. Kodlama performansı kapsamında; bu iki farklı robotik kodlama ortamının üstün yetenekli öğrencilerin Python programlama dili başarılarına ve programlama dili bilgilerinin kalıcılığına (hatırlanmasına) etkisinin yanı sıra, haftalık kazanımlara göre öğrenme süreçlerindeki performansları ve meşguliyetleri de incelenmiştir. Aynı zamanda üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlama performansları, öz-yeterlikleri ve kaygıları arasında anlamlı bir ilişki olup olmadığı da belirlenmiştir. Ayrıca öğrencilerin sanal ve fiziksel robotik kodlama ortamlarındaki Python programlama dili öğrenme deneyimlerine ilişkin görüşleri karşılaştırılarak aralarındaki benzerlik ve farklılıklar ortaya çıkarılmıştır.

Araştırmanın amacı çerçevesinde cevap aranan araştırma soruları aşağıdaki gibidir;

- 1) Üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlama performansları arasında anlamlı bir fark var mıdır?
 - a. Üstün yetenekli öğrencilerin haftalık sanal ve fiziksel robotik kodlama kazanımları arasında anlamlı bir fark var mıdır?
 - b. Üstün yetenekli öğrencilerin haftalık sanal ve fiziksel robotik kodlama meşguliyetleri arasında anlamlı bir fark var mıdır?
 - c. Üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlamadaki Python programlama dili başarıları arasında anlamlı bir fark var mıdır?

- d. Üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlamadaki Python programlama dili bilgilerinin kalıcılığı (hatırlanması) arasında anlamlı bir fark var mıdır?
- 2) Üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlama öz-yeterlikleri arasında anlamlı bir fark var mıdır?
- 3) Üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlama kaygıları arasında anlamlı bir fark var mıdır?
- 4) Üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlama performansları, öz-yeterlikleri ve kaygıları arasında anlamlı bir ilişki var mıdır?
- 5) Üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlama sürecindeki Python programlama dili öğrenme deneyimlerine ilişkin görüşleri nelerdir?

Araştırmanın Önemi ve Gerekçesi

Günümüz eğitim teknolojisi gelişen ve değişen dünyaya uyum sağlamak ve sistem içerisine daha nitelikli bireyler yetiştirmek için, tüm dünyada ve ülkemizde son zamanlarda programlama eğitiminin hızla yaygınlaşması önemli bir durum olarak karşımıza çıkmaktadır. 21. yüzyıl öğrenci becerilerinin kazandırılması için programlama eğitiminin ne kadar gerekli olduğu yapılan araştırmalar ile ortaya konulmaktadır.

Bilgi işlem cihazları ile aynı dili konuşabilmek, onları yöneten ya da onlar tarafından yönetilen bir birey olmanın temel farkını ortaya koyacaktır. Bu bağlamda, öğrencilerin teknoloji ve kodlama okuryazarlığına sahip üreten ve teknoloji yöneten bir gelecek oluşturmada problem çözme, algoritma tasarlama ve kodlama becerileri temelli bir programlama dersi almaları önemlidir.

Ülkemizde programlama eğitimi 2012 yılından sonra önem kazanmış ve ortaokuldan itibaren yeni öğretim programlarına alınmaya başlamıştır. Milli Eğitim Bakanlığı (MEB) ortaokullardaki Bilişim Teknolojileri ve Yazılım Dersi ile öğrencilerin çağın gereksinimlerine uygun 21. yy becerilerini kazanmalarını hedeflemektedir. Bu kapsamda öğrencilerin bilişim teknolojilerini etkili ve amacına uygun biçimde kullanmaları, bilgisayar bilimine ilişkin teknik bilgi kazanmaları, proje-ürün geliştirme süreçlerini kullanarak problem çözme ve bilgi işlemsel düşünme becerilerini geliştirmeleri, akıl yürütme süreçlerini takip etmeleri, algoritma geliştirmeleri ve görsel tabanlı programlama araçlarından en az birini temel düzeyde kullanabilmeleri öne çıkan bu beceriler arasındadır (MEB, 2018). Bu beceriler ile MEB, öğrencilerin yalnızca teknolojiyi kullanabilen bireyler olmalarını değil, aynı zamanda analitik düşünebilen, problem çözme sürecinde bilimsel yöntemleri uygulayabilen ve bilişim

teknolojilerini yaratıcı biçimde kullanarak yenilikçi ürünler geliştirebilen bireyler olarak yetişmeyi amaçlamaktadır.

MEB ortaöğretim Bilgisayar Bilimi dersi ile ise, öğrencilerin temel programlama becerilerini, metin tabanlı bir programlama diline uyarlayarak geliştirmelerini hedeflemektedir. Son yıllarda büyük ölçüde önem kazanan ve kullanım alanı gün geçtikçe genişleyen Python programlama dili bunlardan biridir. Bu kapsamda MEB tarafından Python programlama dili eğitime yönelik hedef kazanımlar belirlenmiştir. Böylece öğrencilerin sorgulama, aktif katılım, özgün düşünme ve pratiğe dayalı öğrenme yoluyla içsel motivasyonlarını artırarak programlama özgüvenlerini ve öz-yeterliklerini geliştirmeyi amaçlamaktadır (MEB, 2023).

Python programlama dilinin açık kaynaklı yapısı ve görece yalın bir sözdizimi, başlangıç düzeyindeki öğrencilerin dili bilişsel olarak aşırı yüklenmeden deneyimlemelerini kolaylaştırmaktadır. Öğrenciler temel komut ve yapıtaşlarını kavradıkça, Python'un sunduğu esneklik sayesinde programlamanın daha karmaşık bileşenlerine (ör. modüler yapı, veri yapıları, problem çözme stratejileri) yönelerek bilgi ve becerilerini kademeli olarak derinleştirme eğilimi göstermektedirler (Tiryaki & Balaman, 2021). Python programlama dilinin daha kolay öğrenildiğini, profesyonel bir programlama dili olduğunu ve günlük yaşamda kariyer hedeflerinde daha çok işe yarayacağını düşünen, buna bağlı olarak Python programlama dilini öğrenmeye yönelik daha güçlü bir motivasyon hissettiğini dile getiren öğrenciler olduğu alanyazında belirtilmiştir (Tran vd., 2024). Bu durum, Python'un gerçek dünya problemlerini çözmek için uygulanmasıyla da desteklenmekte ve böylece bu programlama dilinin uygulanabilirliği artmaktadır. Bu nedenle çalışmada Python programlama diline yönelik hedef kazanımlar temel alınmıştır.

Öğrenciler programlama dili eğitimi sürecinde öğrendiklerini performansa dönüştürmekte oldukça güçlük çekmektedirler (Özdiç & Altun, 2014). Yapılan araştırmalar incelendiğinde öğrenciler, programlama dersinin zor olduğunu düşünmekte ve programlama derslerinde birçok işlemin ve kavramın soyut kaldığını belirtmektedirler (Yurdakök, 2022). Bununla birlikte öğrendikleri bilgileri uygulamakta zorluk yaşadıkları ifade edilmektedir (Askar & Davenport, 2009). Bu durum programlama dili eğitimi sürecinde öğrenciler üzerinde kaygı, motivasyon eksikliği ve başaramam düşüncesi gibi bir takım olumsuz duyguları ortaya çıkartmaktadır. Yapılan araştırmalarda, programlama dili öğretiminde robotik araç kullanımının ortaya çıkabilecek bu olumsuz duyguları en aza indirdiği görülmektedir (Güteryüz, 2019; Kasalak, 2017; Tisza vd., 2022; Yıldırım, 2020; Yurdakök, 2022; Zhao vd., 2024).

Bu arařtırmada biri sanal diğeri fiziksel ortamda kullanılan iki farklı robotik kodlama aracı ile programlama öğretime gerekleřtirilmiřtir. Bu kapsamda sanal robotik kodlama ortamı olarak VEXcode-VR, fiziksel robotik kodlama ortamı olarak ise Raspberry-Pi tercih edilmiřtir. VEXcode-VR, öğrencilerin herhangi bir fiziksel donanıma ihtiya olmadan web tabanlı bir platform üzerinden, güvenli ve erişilebilir bir öğrenme alanı sunarak Python programlama dili ile sanal robotları kodlamalarına imkân tanıyan bir öğrenme aracıdır (VEX Robotics, 2024). Bu platform, öğrencilerin programlama sürecinde hataları anında görebilmelerine olanak sağlayarak, algoritmik düşünme, hata ayıklama ve problem çözme becerilerini geliřtirmeye katkıda bulunmaktadır. Fiziksel ortamda kullanılan Raspberry-Pi ise mikrodenetleyici tabanlı bir mini bilgisayar olup, sensör, motor ve diğeri elektronik bileřenlerin Python diliyle doğrudan programlanmasına imkân tanır (Raspberry-Pi, 2024). Bu sayede öğrenciler, fiziksel dünyadaki sistemlerle etkileřim kurarak kodladıkları programların gerek dünyadaki karřılığın gözlemleyerek öz-yeterlik düzeylerini güçlendirme, soyut kodlama becerilerini somut ürünlere dönüřtürerek uygulama temelli öğrenme deneyimlerini artırma ve gerek dünya uygulamaları geliřtirme fırsatı elde etmektedirler (Yıldırım, 2020; Güleriyüz, 2019). Birbirinden farklı özellikleri olan bu sanal ve fiziksel robotik kodlama ortamlarının, programlama hedef-kazanımlarına ulaşma, performans sergileme, öz-yeterlik geliřtirme ve kaygıyı düşürme potansiyelleri açısından karřılařtırılması alıřmaya değeri katmaktadır.

Eğitsel robotikler aracılığıyla öğretime gerekleřtirilen yazım kuralları, söz dizimi farklı olan birçok programlama dili bulunsu da, Python programlama dilinin bahsi geen avantajları sayesinde hem sanal ve hem de fiziksel robotik kodlama ortamlarında kullanımını yaygınlařmaktadır. Bu alıřmada da eřitli avantajlarının yanı sıra, sanal VEXcode-VR ve fiziksel Raspberry-Pi robotik kodlama ortamlarının ikisinde de ortak olarak kullanılabildiğeri için Python programlama dili öğretime gerekleřtirilmiřtir. Bu iki farklı robotik kodlama ortamının programlama dili kazanımlarını edindirme ve bu kazanımların kalıcılığın sağlama açısından ne derece etkili olduğun karřılařtırma, alanyazında rastlanmayan bir arařtırma olması nedeniyle özgünlük taşımaktadır.

Biri sanal diğeri fiziksel farklı iki robotik araç ile gerekleřtirilen Python programlama dili öğrenme sürecindeki öğrenci meřguliyetinin duyuřsal, biliřsel, davranıřsal, temsili ve sosyal alt boyutlarda incelenmesi, bu değeriřkenin öğrenme içeriğeri, öğretim materyali/aracı, öğrenme ortamı gibi eřitli etkenlerden ne derece etkilediğeri öğrenmek açısından önemli görölmektedir. Bu nedenle bu alıřmada Python programlama dili öğretime süresince öğrencilerin meřguliyet durumlarının sadece bir defaya mahsus değeri her hafta, öğretim etkinliğinden hemen sonra ölçölmesiyle, programlama dili öğretiminde etkinlik erevesinde

robotik kodlama meşguliyetlerinin tespiti de amaçlanmaktadır. Böylece sanal ve fiziksel robotik kodlama araçları ile programlama dili öğretimi çerçevesinde, ortaokul seviyesindeki öğrencilerin robotik kodlama meşguliyetleri üzerine yapılan az sayıdaki karşılaştırma araştırmalarına güç katacağı düşünülmektedir.

Bilim ve Sanat Merkezlerinde (BİLSEM) öğrenim gören öğrenciler, akranlarına göre bilişsel, yaratıcılık ve problem çözme becerileri açısından belirgin farklılıklar göstermektedir. Bu öğrenciler, yüksek düzeyde merak, özgün düşünme ve karmaşık problemlere yönelik çözüm üretme eğilimiyle tanınmaktadır (Kaya, 2013). BİLSEM öğrencilerinin öğrenme hızlarının yüksek olması, geleneksel öğretim ortamlarında sıkılmalarına ve potansiyellerini tam olarak kullanamamalarına yol açabilmektedir (Yaman, 2020). BİLSEM ortamları, öğrencilerin bu bireysel farklılıklarını gözeterek yeteneklerini destekleyen farklılaştırılmış öğretim uygulamaları sunmakta ve öğrencilerin öz-yeterlik, yaratıcılık ve özyönetim becerilerini geliştirmeyi hedeflemektedir (Oruç & Çağır, 2022). Yapılan çalışmalar, BİLSEM öğrencilerinin uygulama temelli etkinliklerde daha yüksek motivasyon ve katılım sergilediklerini, özellikle robotik, kodlama ve proje tabanlı öğrenme süreçlerinde üretkenliklerinin belirgin biçimde arttığını ortaya koymaktadır (Akbüker vd., 2019; Şahin, 2014). Bu özellikler, üstün yetenekli öğrencilerin eğitiminde bilişsel olduğu kadar duyuşsal, davranışsal ve sosyal gelişimin de bütüncül biçimde ele alınması gerektiğini göstermektedir.

Millî Eğitim Bakanlığı tarafından yayımlanan Bilim ve Sanat Merkezleri (BİLSEM) Yönergesi doğrultusunda, genel yetenek alanında tanılanarak sınavı kazanan tüm öğrenciler, bireysel farklılıklarını geliştirmeye yönelik olarak “Robotik Kodlama Atölyesi” dersi almaktadır. Yönerge kapsamında, öğrencilere yönelik robotik kodlama dersleri bireyselleştirilmiş eğitim planları hazırlanarak yürütülmektedir. Bu süreçte, grup içi öğrenci düzeyleri ve sınıf kademeleri birbirinden bağımsız olarak ele alınmakta; öğrenciler, grup sayıları ve atölye kapasitesi dikkate alınarak karma gruplar hâlinde robotik kodlama eğitimine katılmaktadır. Robotik Kodlama Atölyesi öğretim programında, farklı robotik araçlar ve programlama dilleri kullanılarak öğrencilerin algoritmik düşünme, problem çözme, öz-yeterlik ve üretkenlik becerilerinin geliştirilmesi amaçlanmaktadır (MEB, 2021) .

Eğitim müfredatımızda ortaokul seviyesinde metin tabanlı programlama eğitimi verilmemektedir. Sadece görsel tabanlı ortamlarda yapılan, algoritma ve temel programlama eğitimi verilmektedir. Ancak bu çalışmada hedef kitle, araştırmanın yapıldığı Bilim ve Sanat Merkezindeki üstün yetenekli ortaokul öğrencileridir. Bu bağlamda, biri sanal ve biri fiziksel ortamdaki farklı iki robotik aracı kullanılarak gerçekleştirilen 10’ar haftalık Python programlama dili öğretimi sürecinde, üstün yetenekli ortaokul öğrencilerinin robotik kodlama

meşguliyetleri çeşitli alt boyutlarda haftalık olarak ele alınmıştır. Bunun yanı sıra, öğrencilerin programlama başarıları ve programlama bilgilerinin kalıcılığı, programlama öz-yeterlikleri ve kaygıları da incelenerek duyuşsal, bilişsel, davranışsal ve sosyal boyutlarda anlamlı bir gelişim gösterip göstermedikleri karşılaştırmalı olarak sunulmuştur. Araştırma sonuçlarının Bilim ve Sanat Merkezindeki üstün yetenekli ortaokul öğrencilerine yönelik öğretim programını iyileştirme bakımından müfredat geliştiricilere ve alan yazına katkıda bulunacağı öngörülmektedir. Ortaokul seviyesinde sanal ve fiziksel iki farklı robotik kodlama aracıyla yapılan metin tabanlı programlama eğitimi araştırmalarının da azlığı göz önünde bulundurulduğunda, bu çalışmanın 10 hafta gibi uzun soluklu uygulama süreciyle yeni çalışmalara yol gösterici olacağı düşünülmektedir.

Sınırlılıklar

Bu çalışmanın sınırlılıkları aşağıdaki şekilde sıralanmıştır:

- Çalışma, robotik kodlama dersi olan ortaokul seviyesindeki üstün yetenekli öğrencilerin bulunduğu yalnızca bir Bilim ve Sanat Merkezinde gerçekleştirildiğinden, sanal ve fiziksel robotik kodlama grupları 10’ar öğrenciyle sınırlıdır.
- Her iki gruptaki üstün yetenekli öğrenci sayısının az olması sonuçların genellenebilirliğini sınırlandırmaktadır.
- Uygulama boyunca sanal robotik kodlama grubunda yalnızca VEXcode-VR, fiziksel robotik kodlama grubunda ise yalnızca Raspberry-Pi aracı kullanılmıştır.
- Çalışma grubundaki üstün yetenekli öğrencilerin hiçbiri daha önceden eğitimi gerçekleştirilen robotik araçlar ve programlama dilinde herhangi bir eğitim almamışlardır.
- Her iki gruptaki uygulama, robotik kodlama araçlarıyla metin tabanlı programlama eğitimi sürecine yönelik belirlenen ortak kazanımları gerçekleştirmek amacıyla haftada 2 saat olmak üzere toplamda 10 hafta sürmüştür.
- Araştırmayı yapan uygulayıcı aynı zamanda her iki gruptaki eğitimi gerçekleştiren eğitmeni olduğundan, bu durum öğrencilerin öğrenme sürecini olumlu/olumsuz etkilemiş olabilir.
- Araştırmada elde edilen sonuçlar, yalnızca sanal ve fiziksel robotik araçlarla programlama eğitimi yapılan üstün yetenekli ortaokul seviyesindeki öğrenci gruplarının robotik kodlama meşguliyetlerini, programlama başarılarını, öz-yeterliklerini, kaygılarını ve uygulama sürecine yönelik deneyimlerini yansıtmaktadır.

Varsayımlar

Bu çalışmadaki temel varsayımlar aşağıdaki şekilde sıralanmıştır:

- Sanal ve fiziksel robotik kodlama gruplarındaki tüm öğrenciler 10 haftalık etkinliklere istekli ve gönüllü olarak katılmışlardır.
- Python programlama dili hedef-kazanımları temel alınarak hazırlanan haftalık sanal ve fiziksel robotik kodlama etkinlikleri, üstün yetenekli ortaokul öğrencilerinin bilişsel gelişim seviyelerine uygundur.
- Uygulama öncesi, sırası ve sonrasında öğrenciler veri toplama araçlarına rastgele cevaplar vermemişler, soruları cevaplarırken gerçek duygu ve düşüncelerini dile getirmişlerdir.
- Nicel ve nitel tüm verilerin analizi tarafsız ve doğru şekilde gerçekleştirilmiştir.

Terim ve Tanımlar

VEXcode-VR Sanal Robotik Kodlama Aracı: VEX Robotics tarafından geliştirilen, robotik ve programlama becerilerini sanal ortamda öğretmeyi amaçlayan bir eğitim platformudur. Fiziksel bir robota ihtiyaç olmadan, web tarayıcısı üzerinden erişilebilen bu platform sayesinde öğrenciler sanal robotları kodlayarak çeşitli görevleri yerine getirebilirler. VEXcode-VR arayüzü ile görsel programlama yapılabilen kod bloklarının yanı sıra Python gibi çeşitli metin tabanlı programlama dilleri de kullanılabilir. Bu nedenle başlangıç seviyesinden ileri düzey programlama bilen öğrencilere kadar herkes için uygun bir sanal robotik kodlama aracıdır (VEX Robotics, 2024).

Raspberry-Pi Fiziksel Robotik Kodlama Aracı: Raspberry-Pi cihazları kullanarak robotik projelerin geliştirilmesine yönelik bir platformdur. GPIO (General Purpose Input Output) pinlerini kullanarak motorları, sensörleri ve diğer elektronik bileşenleri kontrol etmek için yazılım geliştirmeyi içerir. Raspberry-Pi, düşük maliyetli, küçük bir bilgisayar olmasına rağmen güçlü işlem kapasitesi sunar ve çeşitli fiziksel robotik uygulamalar için ideal bir temel oluşturur (Raspberry-Pi, 2024).

Python Programlama Dili: Temel felsefesi, programcıların karmaşık fikirleri sade ve anlaşılır biçimde ifade edebilmesini sağlamak olan Python, kod okunabilirliği yüksek, öğrenmesi kolay ve güçlü bir programlama dilidir. Bu nedenle Python, hem başlangıç seviyesindeki hem de ileri düzey programcılar için uygun bir programlama dili olarak kullanılmaktadır. Bu özellikleri ile Python sanal ve fiziksel çeşitli robotik kodlama araçlarında kullanılabilen bir programlama dili haline gelerek geniş kitlelerce desteklenmiştir (Rossum, 1995).

İKİNCİ BÖLÜM

Kuramsal Çerçeve ve İlgili Araştırmalar

Eğitsel Robotik Kodlama Uygulamaları ve Programlama Öğretimi

Günümüz eğitim modelinde ve eğitim programlarının hazırlamasında önem arz eden Uluslararası Eğitim Teknolojileri Derneği [The International Society for Technology in Education] (ISTE) standartlarında, programlama eğitiminin öneminden sıkça bahsedilmektedir. ISTE standartları birçok disiplinde bilgisayar okuryazarlığı olgusunu ön plana çıkarmaktadır. Özellikle son yıllarda tüm okul kademelerinde, ISTE (2016) standartlarına göre bilgi toplumu ile aynı seviyede kalabilmek için 21.yüzyıl öğrenen özelliklerine sahip olunması gerekmektedir. Bu standartlara göre öğrenenlerden beklenen temel beceriler; eleştirel düşünme ile bilgi oluşturucu, yaratıcı ürünler üreten, öğrenme yaşantısını geliştirmek için dijital araçları verimli kullanan, faydalı çözümler üreterek sorunları tanımlayan, dijital ortamları kullanarak iletişim kuran, bilgi ve becerisini kullanarak iş birliği yapan, paylaşan ve zenginleştiren bireyler olmalarıdır.

Millî Eğitim Bakanlığı (MEB) da öğretim programlarında ortaöğretim bilgisayar okuryazarlığı kapsamında 21. yy becerilerini içine alan üst düzey düşünme becerilerinin geliştirilmesini sağlayacak önemli düzenlemeler yapmıştır (MEB, 2023). MEB tarafından geliştirilmesi amaçlanan üst düzey düşünme becerilerinden bir de bu becerileri kapsayan programlama/kodlama becerisidir. Bu nedenle temelde programlama öğretiminde birçok teknik ve uygulama yazılımları kullanarak öğrenen bireylerin programlama dilleri ve araçları aracılığıyla kodlama becerisi kazanma arzusunu artırıcı ve programlama öğretimi somutlaştırıcı çalışmalar yapılmaktadır (MEB, 2016). Programlama yapmak, teknik bir beceri olmanın ötesinde, aynı zamanda bilimsel problem çözme yöntemini de içinde barındırır. Programlama öğrenmek, öğrencilerin mantıksal düşünme, yaratıcılık ve karmaşık problemlerle başa çıkma becerilerini geliştirerek öğrenme yeteneklerini ve bilgi işlemsel düşünme becerilerinin gelişmesine olanak sağlar (Wang vd., 2014).

Öğrenciler için kodlama eğitimi analitik ve eleştirel düşünme becerilerinin geliştirilmesinde önemli katkılar sunmaktadır ve günümüzde bu amaç için kullanımı oldukça yaygınlaşmıştır. Aynı zamanda muhakeme becerilerinin artırılması, problem çözme becerilerinin geliştirilmesi ve tasarım odaklı fikirlerin geliştirilmesi eğitim alanında kodlama odaklı çalışmaların merkezinde yer almaktadır (Hamiti vd., 2021). Python da,

programlama/kodlama becerisi kazandırma amacıyla kullanılan metin tabanlı programlama dillerinden biridir.

Python programlama dilinin temel felsefesi, kod okunabilirliğine önem vererek, programcıların karmaşık fikirleri sade ve anlaşılır biçimde ifade edebilmesini sağlamaktır. Özellikle Python kodlarını yorumlamak ve öğrenmek diğer dillere göre daha kolaydır diğer dillerde bulunan noktalama işareti zorunlulukları, parantezler veya kurallar programlamaya yeni başlayan kullanıcı için zaman zaman zorluklar çıkarmaktadır. Ancak Python’da bu tür zorunluluklar olmadığı gibi yapısı itibarıyla diğer dillere göre daha sadedir. Bu nedenle Python, öğrenmesi kolay ve güçlü bir programlama dilidir (Rossum, 1995). Python’un kullanımı keyifli, öğretimi kolay ve geniş bir topluluk tarafından desteklenen bir dil olması, onu hem başlangıç seviyesindeki hem de ileri düzey kullanıcılar için uygun bir araç haline getirmektedir. Bu kapsamda özellikle lise düzeyindeki Bilgisayar Bilimleri dersi kapsamında Python programlama dili öğretimi gerçekleştirilmektedir (MEB, 2020).

Programlama öğretiminin kalitesi, öğrencilerin bilginin pasif alıcıları olmak yerine aktif olarak bilgiyi kendi işleyen ve analiz eden bunun sonucunda özgün sentezlemeler yaparak sonuçlar ortaya koyabilen öğretim, yapılandırmacı yaklaşımlar kullanılarak geliştirilir. Bu nedenle, öğretici materyal ne kadar spesifik ve pratik olursa, öğrenme sürecindeki sonuç da o kadar iyi olur (Alvarez & Larranaga, 2016). Yenilikçi öğretim uygulamalarından biri olan eğitsel robotiklerin programlama eğitiminde kullanımı, öğrencilerin ilgisini ve motivasyonunu artırarak daha verimli öğretim gerçekleşmesini sağlama potansiyeliyle karşımıza çıkmaktadır. Eğitsel robotikler ile programlama eğitiminde sensörler, LED’ler, servo motorlar gibi programlara bilir mikro işlemciler ve mikrodenetleyicilerin yardımıyla problem çözme, algoritma oluşturma, kod yazma gibi programlama bileşenleri bir araya getirilir (Sarı vd., 2022).

Öğrenciler, programlama eğitiminde bilgisayar ekranında yazdıkları kodların sonuçlarını yalnızca bilgisayar ekranından görebilirler, fakat eğitsel robotik ile yapılan programlama eğitiminde geleneksel programlama eğitiminden farklı olarak kodların çalıştırdığı, hareket ettirdiği sensör sistemleri ile algıladığı bir çıktı görebilirler. Bu da programlama eğitiminin somutlaştırılarak küçük yaş gruplarında bile başarılı ve verimli bir eğitim ortamı olanağı sunar (Fatsa & Turan, 2022). Eğitsel robotik uygulamalar, öğrencilerin kendilerini pasif konumdan aktif öğrenenler konumuna dönüştürmelerine, çevreleriyle iş birliği yaparak yeni bilgiler oluşturmalarına ve bir araştırmacı gibi davranarak temel zihinsel becerileri geliştirmelerine yardımcı olur (Atmatzidou & Demetriadi, 2016). Robotik kit kullanılan bir araştırmada, etkinliğin kod yazma gibi soyut ve karmaşık işlemleri somutlaştırarak öğrencilerin öğrenme potansiyellerine katkı sağladığı belirtilmektedir (Gündoğdu, 2020). Bir başka

arařtırmada, robotik eęitimlere katılan öęrencilerin öęrenme deneyimlerinde aktif rol alarak bir hedef doęrultusunda robotlar tasarlayarak ve kodlayarak programlamanın sonucunu da geręekęi bir řekilde üç boyutlu olarak görebildięi vurgulanmaktadır (Uęuz, 2019).

Sucu ve akıroęlu (2022) eęitsel robotiklerin programlama öęretiminde, anaokulundan üniversiteye kadar önemli bir potansiyele sahip olduęunu vurgulamaktadırlar. Eęitsel robotiklerin tüm eęitim kademelerinde kullanılabiliyor olması, önemli avantajlar arasındadır. Hangün vd. (2022) eęitsel robotiklerin amaçlar doęrultusunda kodlanmasını, öęrencilerin program yazmaya ilgi duymasını saęlayan, öęrencilerin yazılım geliřtirme maharetlerinin gelişmesine imkân tanıyan önemli bir teknoloji olarak tanımlamaktadır. Eęitsel robotikler ile kodlama yapma, erken çocukluk döneminden itibaren öęrencilerin programlama ve algoritma yeteneęi gibi STEM ile ilgili biliřsel yeteneklerini geliřtirmek için gelişimsel olarak uygun araçlardır (Yang vd., 2022). Bu nedenle bu alıřmada, Bilim ve Sanat Merkezleri'ndeki üstün yetenekli ortaokul öęlencilerine Python programlama dili öęretimi için eęitsel robotik uygulamalardan faydalanılmıřtır.

Son yıllarda farklı sanal ve fiziksel robotik uygulamalar, eřitli programlama dillerini öęretmek amacıyla yaygın biimde kullanılmaktadır. Bu uygulamalar benzer hedeflere hizmet etse de, sahip oldukları teknik özellikler ve öęrenme ortamları aısından farklılık gösterebilmektedir. Sanal robotik araçlar arasında VEXcode-VR ve Tinkercad Circuits gibi platformlar yer almakta olup, öęrencilerin evrim ii ortamlarda kodlama ve problem özme becerilerini geliřtirmelerine olanak tanımaktadır. Buna karřılık, fiziksel robotik araçlar olarak Raspberry-Pi, LEGO Mindstorms, mBot ve Arduino setleri gibi donanıma dayalı sistemler kullanılmakta, bu araçlar öęrencilere gerek donanım üzerinde deneyim kazanma ve somut üretim süreçlerini gözlemleme fırsatı sunmaktadır (Kılı, 2022; Ru & Fok, 2022).

Her iki tür robotik uygulama da programlama öęretiminde etkili olmakla birlikte, öęrenme deneyimi ve etkileřim biimi bakımından birbirinden farklılıklar gösterebilmektedir. Rukangu vd. (2025), sanal ve fiziksel robotik araçların farklı yönlerden motive edici olduęunu; fiziksel robotiklerin faydalılık kategorisinde daha güçlü olduęunu, sanal robotiklerin ise ilgi ekicilik kategorisinde daha güçlü olduęu ifade etmiřtir. Bu alıřmada da ise; sanal robotik kodlama ortamı olarak VEXcode-VR, fiziksel robotik kodlama ortamı olarak ise Raspberry-Pi tercih edilmiřtir. VEXcode-VR, öęrencilerin herhangi bir fiziksel donanıma ihtiya olmadan web tabanlı bir platform üzerinden, güvenli ve erişilebilir bir öęrenme alanı sunarak Python programlama dili ile sanal robotları kodlamalarına imkân tanıyan bir öęrenme aracıdır (VEX Robotics, 2024). Fiziksel ortamda kullanılan Raspberry-Pi ise mikrodenetleyici tabanlı bir mini bilgisayar olup, sensör, motor ve dięer elektronik bileřenlerin Python diliyle doğrudan

programlanmasına imkân tanır (Raspberry Pi, 2024). Bu kapsamda, bu çalışmada birbirinden farklı özellikleri olan bu iki robotik ortamdan sanal VEXcode-VR ile bir grup, fiziksel Raspberry-Pi ile diğer grup Python programlama dili temelinde robotik kodlama yapmış, her iki gruptaki öğrencilerin robotik kodlama performansları, meşguliyetleri ve deneyimleri karşılaştırılmıştır.

Millî Eğitim Bakanlığı tarafından yayımlanan Bilim ve Sanat Merkezleri (BİLSEM) Yönergesi doğrultusunda, genel yetenek alanında tanılanarak sınavı kazanan tüm öğrenciler, bireysel farklılıklarını geliştirmeye yönelik olarak “Robotik Kodlama Atölyesi” dersi almaktadır. Yönerge kapsamında, öğrencilere yönelik robotik kodlama dersleri bireyselleştirilmiş eğitim planları hazırlanarak yürütülmektedir. Bu süreçte, grup içi öğrenci düzeyleri ve sınıf kademeleri birbirinden bağımsız olarak ele alınmakta; öğrenciler, grup sayıları ve atölye kapasitesi dikkate alınarak karma gruplar hâlinde robotik kodlama eğitimine katılmaktadır. Robotik Kodlama Atölyesi öğretim programında, farklı robotik araçlar ve programlama dilleri kullanılarak öğrencilerin algoritmik düşünme, problem çözme ve üretkenlik becerilerinin geliştirilmesi amaçlanmaktadır (MEB, 2021).

Robotik Kodlama Atölyesi derslerinde genellikle Arduino robotik setleri ile öğrencilere blok tabanlı kodlamanın yanı sıra C++ programlama dili öğretilmektedir. Ancak özellikle ortaokul düzeyindeki öğrenciler için C++ programlama dili, yazım kuralları ve kullanım karmaşıklığı nedeniyle bazı zorluklar doğurmaktadır. Bu nedenle, temel düzeyde metin tabanlı bir programlama dili öğrenme sürecinde, daha sade sözdizimine sahip ve gelecekte robotik kodlama etkinliklerinde kullanım potansiyeli yüksek olan Python programlama dilinin öğretiminin, öğrencilerin programlama becerilerini geliştirmede daha uygun olacağı Milli Eğitim Bakanlığı tarafından öngörülmüştür (MEB, 2021). Bu çalışmada da üstün yetenekli öğrencilerin Python programlama dili performanslarını ve meşguliyetlerini artırmaya, öz-yeterliliğini geliştirmeye, kaygılarını azaltmaya yönelik olarak sanal ve fiziksel robotik kodlama etkinlikleri yapılmıştır.

Robotik Kodlama ve Programlama Başarısı

Programlama eğitimi, ilköğretimden ortaöğretim seviyesine kadar her kademede önemli eğitim faaliyetlerinden biri olarak karşımıza çıkmaktadır (Çalışkan, 2020). Programlama öğretimi süreci, öğrencilerin algılayıp uygulaması zor görevleri barındıran etkinlikler içermektedir (Koorse vd., 2015). Bu nedenle farklı üst düzey düşünme becerilerinin kullanımını gerektirir (Durak vd., 2018). Pek çok farklı disiplinde olduğu gibi soyut içerik, programlama eğitiminde de öğrenme hedeflerinin öğrenenlerin zihninde belirsizleşmesine neden olabilmektedir (Kert vd., 2020). Programlama dili öğretiminde ise, kavramlar öğrenciler

iin soyut kalmakta ve ğrenciler ğrendikleri kavramları somutlaştırmada sıkıntı yaşamaktadırlar (Armoni vd., 2015; Ersoy vd., 2011). Bu kapsamda ğrencilerin kodlama etkinliklerine isteyerek ve motive bir şekilde katılmalarını saėlayan, problem özme ve bilgi-işlemsel düşünme becerilerini kullanmalarını kolaylaştıran blok tabanlı ve robotik kodlama ğretimine yönelik etkinlikler tercih edilmektedir (Kasalak, 2017).

Programlama ğretiminde robotik teknolojiler aracılığı ile soyut kavramlar somutlaştırılabilir ve bireyler üst düzey düşünme becerilerini daha kolay ve kalıcı bir şekilde geliştirebilirler (Numanoėlu & Kesar, 2017). Özellikle ilkokul ve ortaokul seviyesinde robotik araçların soyut ğrenme becerilerinin geliştirilmesinde, programlama yapma becerisi kazanılmasında önemli katkıları olabilmektedir (Alvarez & Larranaga, 2016; avaş & Huyugüzel-avaş, 2004; Ersoy vd., 2011). Literatürdeki ilgili araştırmalar incelendiėinde; programlama ğretiminde robotik kitlerin kullanımının programlama başarısını arttırdığını gösteren benzer sonuçların ortaya ıktığı birçok araştırma bulunmaktadır (Erayta, 2019; Gündoėdu, 2020; Uėuz, 2019; Yıldırım, 2020). Bu nedenle alıřmada biri sanal diėeri fiziksel robotik kodlama ortamında Python programlama dili eėitimi gerçekleştirilmiř ve üstün yetenekli ğrencilerin bu iki farklı robotik ortamdaki programlama dili başarılarının yanı sıra programlama bilgilerinin kalıcılığı (hatırlanması) da karşılaştırılmıřtır.

Robotik Kodlama ve Programlama Öz-Yeterliėi

Öz-yeterlik bireylerin birtakım görevleri yerine getirme konusunda kendilerine olan inanları olarak tanımlanır. Bireylerin belirli bir konudaki başarıya ilişkin inanları başarılarında ok önemli bir rol oynamaktadır (Doėan & Korkmaz, 2025). Diėer bir ifadeyle öz-yeterlik, bireyin belirli bir görevi başarılı bir şekilde yerine getirmede gösterdiėi performansı ve gayreti ile doğrudan ilişkilidir (Kasalak, 2017). Bireylerin öz-yeterlik algısı ile incelenen ilgili görevin başarıyla tamamlanması arasında anlamlı ve pozitif bir ilişkinin olduėunu alan yazında vurgulanmıřtır (Stajkovic & Luthans, 1998). Düşük bireysel öz yeterlik algısı kişisel başarısızlığı ortaya ıkartmaktadır (Askar & Davenport, 2009). Bu nedenle programlama öz-yeterliėi de, programlama görevinin başarıyla tamamlanması iin önemli bir deėişken olarak görölmektedir (Kayhan vd., 2024; Tsai vd., 2019).

Literatürde eėitsel robotik araçlarla yapılan blok tabanlı veya metin tabanlı programlama/kodlama eėitiminin, eřitli sınıf düzeylerindeki ğrencilerin programlama öz-yeterliklerini ve bilgi işlemsel düşünme becerilerini geliştirmelerine katkı saėladığı, sonraki dönemlerde programlama derslerinin ierikleri ile algoritmik düşünme becerileri arasında bir baė oluşturabilmeleri bakımından önemli olduėu belirtilmiřtir (Doėan & Korkmaz, 2022; Kasalak 2017; Yurdakök, 2022). Bu raėmen alanyazındaki eřitli sistematik analiz

alışmalarına bakıldığında (Hangn vd., 2022; Talan, 2020); programlama ğretiminde z-yeterlik deėiřkeninin incelendiėi arařtırmalarda blok tabanlı kodlamanın yaygın olarak kullanıldėı, ancak metin temelli programlamaya iliřkin arařtırmaların daha az olduėu grlmřtr. Bunun yanında z-yeterlik algısı ile ilgili yapılmıř alışmaların diėer deėiřkenlere oranla daha az olduėu gzlemlenmiřtir. Bu nedenle bu alışmada ortaokul dzeyinde sanal ve fiziksel eėitsel robotiklerin kullanıldėı metin tabanlı programlama eėitiminde stn yetenekli ğrencilerin z-yeterlik puanları karřılařtırılmıřtır.

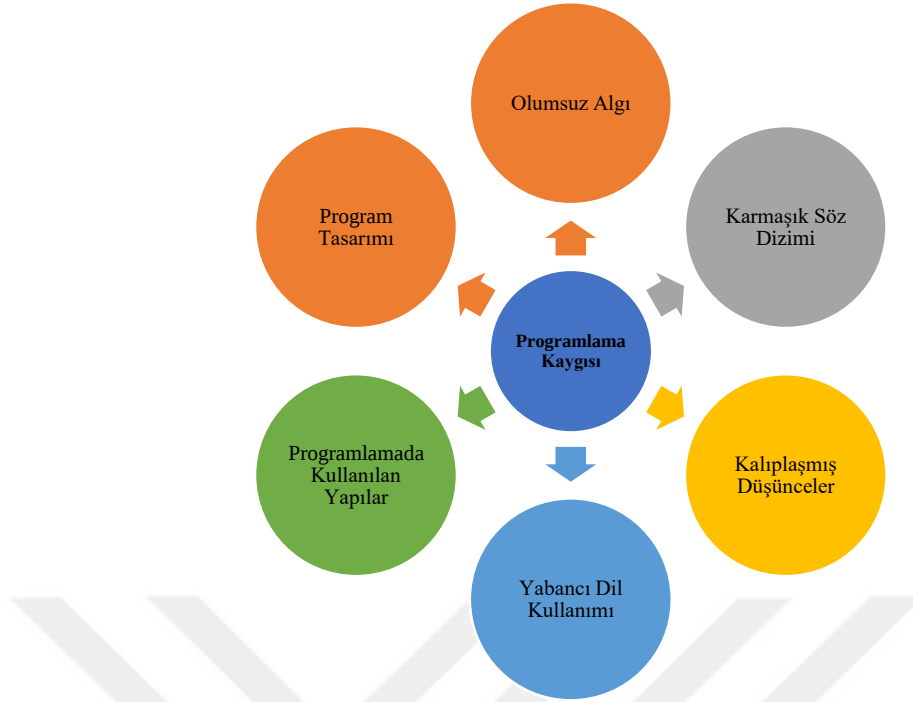
Robotik Kodlama ve Programlama Kaygısı

Kaygı bireyin evresel etkenlere baėlı olarak bulunduėu ortamda duygusal anlamda huzursuz ve endiřeli hissetme halidir (Kula & Sara, 2016). avuş ve Gnbatar (2008) kaygının, gnlk yařantımızda bireyi bazen drtleyerek yaratıcı, yapıcı ve olumlu davranıřlar gstermesini saėlarken bu davranıřların tam aksi bir durum olarak olumlu davranıřların bastırılmasına neden olduėunu aklamaktadır. Wang vd. (2014) kaygıyı gerginlik, endiře ve kaygıyla belirginleřen hoř olmayan bir duygusal durum olarak tanımlamıřtır. Bilgisayar kaygısının bir alt bileřeni olan programlama kaygısı (kendisi de genel bir kaygı biimidir), bireylerin bilgisayar programlamayı ğrenme becerilerini yanlıř deėerlendirdiklerinde ortaya ıkan kaygı trdr (Connolly vd., 2007). Bu durum teknolojinin geliřip, yeni teknolojik ortamların var olması ile bireylerin yeni tip kaygılarının ortaya ıkmasına neden olmuřtur.

zellikle biliřim teknolojilerindeki geliřmeyle ve hayatımızın her alanında kullanımının yaygınlařmasıyla birlikte bilgisayar destekli uygulamaların artıřı “bilgisayar kaygısı” kavramını da beraberinde getirmiřtir. Alan yazında bilgisayar kaygısının tanımları arařtırıldığında, bireyin bilgisayar kullanırken ya da bilgisayar kullanma ihtimali olduėunda korku ve endiře hissetmesi grlmektedir (Namlu & Ceyhan, 2003). Bilgisayar destekli bir ğretim srecinde bireyler bilgisayar kaygısı ile birlikte programlama kaygısı da yařamaktadır, programlama kaygısı; kodlama eėitiminde eėitmenlerin ve ğrencilerin karřılařtıėı bireysel zorluklardan biri olarak nmze ıkmaktadır (Ozmen & Altun, 2014).

Programlama ğretimiyle ilgili alışmalarda programlama kaygısına neden olan faktrler olduėu gze arpmaktadır. Programlama kaygısının; programlamaya karřı algı, programlama dillerindeki karmařık sz dizimi, programlamaya karřı oluřan kalıplařmıř dřnceler, programlamada yabancı bir dil kullanımı, programlamaya zg kullanılan yapılar ve program tasarımı gibi nedenlerden kaynaklanabileceėi alanyazında ortaya ıkmıřtır (nal, 2019). Bu etmenlere baėlı olarak programlama kaygısını etkileyen bařlıklar řekil 1’de gsterilmiřtir.

Şekil 1. Programlama Kaygısını Etkileyen Faktörler



Eğitsel robotik kodlama etkinlikleri öğrenci motivasyonuna katkı sağlayarak bu programlama kaygılarını azaltabilir (Çınar & Tüzün, 2020). Alvarez ve Larranaga (2016) kodlamanın robotik ile birleştirilmesiyle öğrencilerin ders kavramlarını daha iyi anlamalarına yardımcı olduğunu ve algılarının daha da açık olduğunu kanıtlamıştır. Aynı çalışmada öğretmenlerin algılarına göre de, robotik araçlarla yapılan kodlama etkinliklerinde öğrenci motivasyonunda artış olduğu ve programlama kaygı düzeyi düşük olduğu için bu durumun sınıf atmosferini pozitif yönde etkilediği görülmüştür. Ancak eğitsel robotikler ile programlama öğretimine yönelik çalışmalar incelendiğinde, programlama kaygısı ile ilgili araştırmaların düşük sayılarda kaldığı görülmektedir (Talan, 2020; Yolcu & Demirel, 2017). Bu nedenle bu çalışmada sanal ve fiziksel robotiklerle yapılan Python programlama dili öğretiminde üstün yetenekli ortaokul öğrencilerinin programlama kaygı düzeyleri karşılaştırılmıştır.

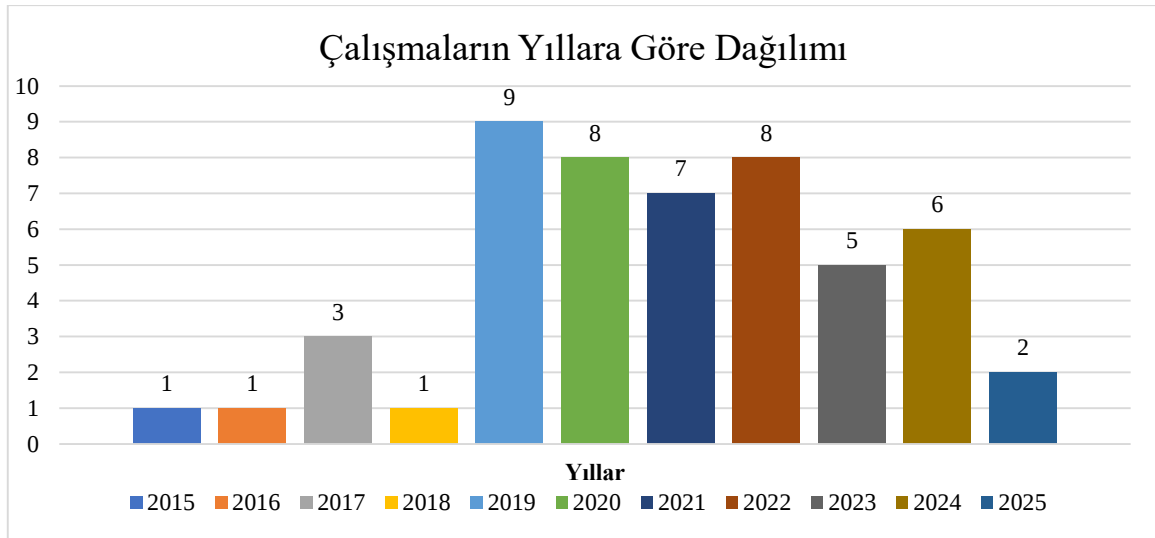
Eğitimde Robotik Kodlama Uygulamaları ile İlgili Araştırmalar

Teknolojik ürünlerin daha kolay ulaşılabilir olmasıyla birlikte eğitsel robotikler oldukça yaygınlaşmış (Hangün vd., 2022), yaptığı bir araştırmada farklı özelliklere sahip çeşitli robotik araçlar, eğitsel robotik uygulamalarda kullanılmaya başlamışlardır (Fatsa & Turan, 2022). Robotik araçlar ile blok tabanlı kodlamanın yanı sıra C/C++, Java ve Python gibi popüler metin tabanlı programlama dilleri de kullanılarak etkinlikler yapılabilmektedir. Öğrenciler kodların çalıştırdığı sensör sistemleri ile algılanan hareket sonucunu eğitsel robotik araçlar üzerinde somut çıktı olarak görebilirler. Ancak eğitsel robotiklerin elektronik cihazlar olması sebebiyle bozulma riskleri, maliyeti ve taşınma zorluğu gibi sınırlılıklar eğitsel robotiklerin eksiklikleri

olarak ortaya çıkmaktadır (Tselegkaridis & Sapounidis, 2021). Eğitimde bilişim teknolojileri, fen, mühendislik ve matematik alanları başta olmak üzere farklı disiplinlerde robotik sistemlerin kullanıldığı ve 21.yy öğrenci becerilerini geliştirmeye katkısının olduğu görülmektedir (Hamiti vd., 2021).

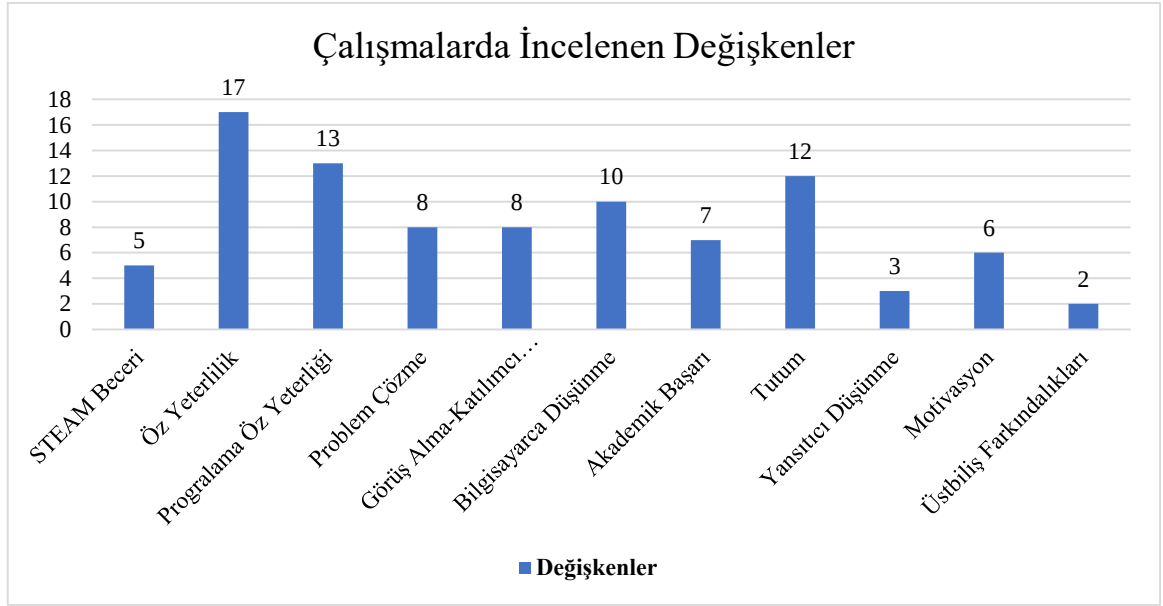
Bu çalışmada yapılan literatür taramasında 2014-2025 yılları arasında eğitimde robotik kullanımıyla ilgili 250 çalışma incelenmiş bunların içinden 50 çalışmanın eğitimde robotik kullanımı ile ilgili olduğu tespit edilmiştir. İncelenen bu çalışmaların; amaçları, kullanılan robotik setler, örneklem düzeyleri ve sayıları, araştırma yöntemleri, veri toplama araçları ve sonuçları EK-11’de tablo halinde sunulmuştur. İncelenen bu çalışmaların yıllara göre dağılımı Şekil 2’de gösterilmektedir.

Şekil 2. 2014-2025 Yılları Arasında Eğitimde Robotik Uygulamaların Yıllara Göre Dağılımı



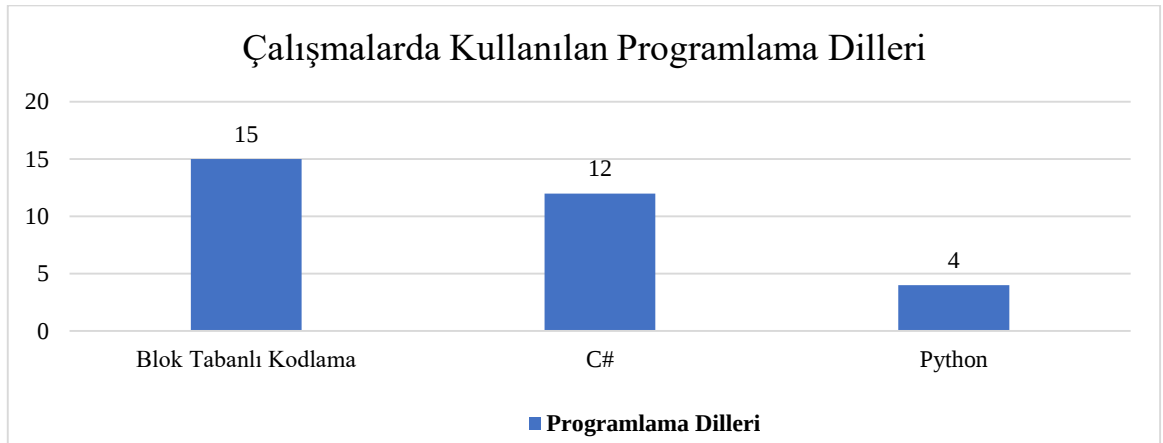
Şekil 2 incelendiğinde, robotik kodlama alanında metin tabanlı programlama öğretiminin yapıldığı çalışmaların 2019 yılından itibaren artmaya başladığı görülmektedir. Bu çalışmada da iki farklı robotik araç ile metin tabanlı programlama öğretiminde öğrencilerin programlama başarısının, programlama eğitiminde kaygı seviyelerinin, programlama öz-yeterlik ve meşguliyet seviyelerini kıyaslamak amacıyla kullanılmıştır. Yapılan literatür taramalarında robotik uygulamalarda yaygın olarak incelenen değişkenler Şekil 3’te sunulmuştur.

Şekil 3. Eğitimde Robotik Uygulamalarda Yaygın Olarak İncelenen Değişkenler



Şekil 3'e göre çalışmalarda daha çok programlama öz-yeterliği ve bilgisayarca düşünme becerisi (Avcı, 2021; Wu & Hwang, 2014), tutum (Alvarez & Larranaga, 2016; Mayerova, 2012) değişkenlerinin incelendiği görülmektedir. Bu çalışmada ise, alanyazındaki gibi öğrencilerin robotik kodlama sürecindeki programlama başarıları ve öz-yeterlikleri incelense de programlama kaygıları ve bu süreçteki meşguliyet seviyeleri de ele alındığından, bu çalışma diğer araştırmalardan farklılık göstermektedir. Diğer taraftan alanyazındaki araştırmalarda kullanılan ve öğretimi gerçekleştirilen programlama dilleri örneklem düzeylerine göre farklılık gösterse de en çok kullanılan programlama dilleri Şekil 4'te sunulmuştur.

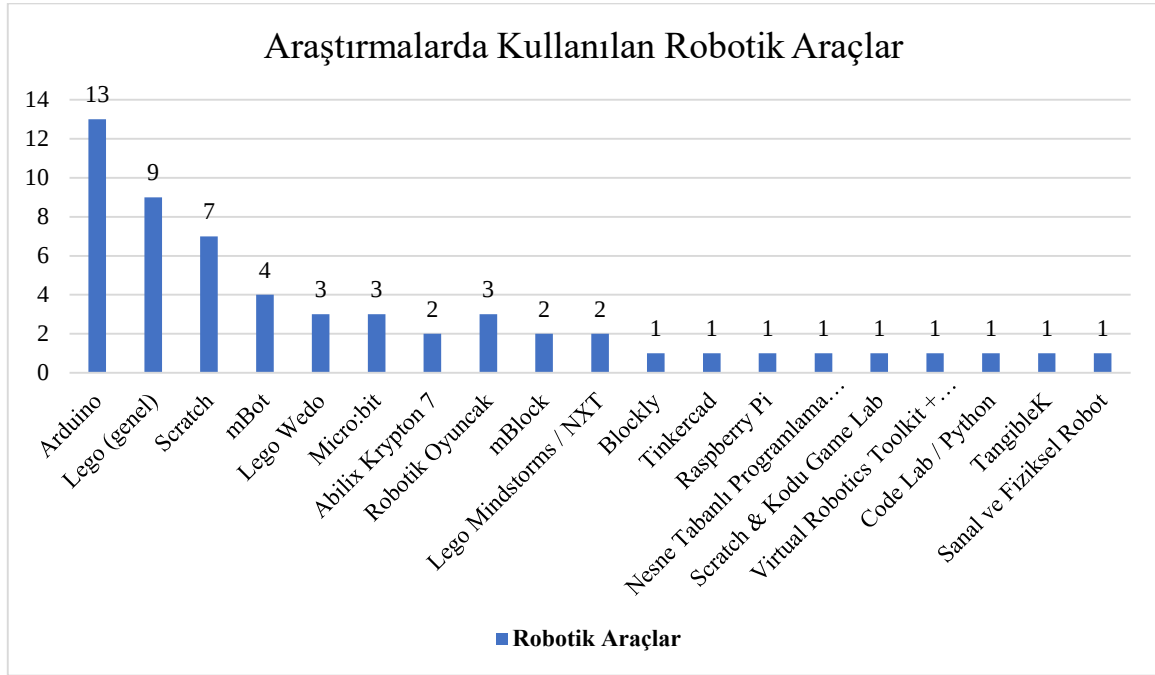
Şekil 4. Araştırmalarda Öğretimi Gerçekleştirilen Programlama Dilleri



Şekil 4'e bakıldığında çoğu araştırmada blok tabanlı kodlama yapıldığı, ancak metin tabanlı programlama dillerinin daha az kullanıldığı görülmüştür. Ayrıca çeşitli avantajları olsa da Python programlama dilinin çok az araştırmada kullanıldığı saptanmıştır. Bu nedenle bu çalışmada iki farklı robotik kodlama ortamında da kullanılabilen Python programlama dili

öğretimi temel alınmıştır. Diğer taraftan alanyazındaki çalışmalarda hangi robotik araçların tercih edildiği de incelenmiş ve Şekil 5'te sunulmuştur.

Şekil 5. Araştırmalarda Kullanılan Robotik Araçlar



Şekil 5'te görüldüğü üzere maliyetinin daha ucuz olması ve daha kolay ulaşılabilir olması nedeniyle Arduino setlerinin en sık kullanılan robotik araçlar olduğu, ikinci sırada Lego setlerinin tercih edildiği görülmüştür. Ayrıca çoğunlukla fiziksel robotik araçların tercih edildiği, sanal robotik araçlarla yapılan çalışmaların çok az olduğu saptanmıştır. Bu çalışmada ise çeşitli avantajları ve özellikleri dikkate alınarak, biri sanal (VEXcode-VR), diğeri fiziksel (Raspberry-Pi) iki farklı robotik kodlama ortamında Python programlama dili öğretimi yapılmış, öğrencilerin robotik kodlama performansları, başarıları ve meşguliyetlerinin yanı sıra programlama öz-yeterlikleri ve kaygıları da karşılaştırılmıştır. Çalışma bu yönüyle diğer araştırmalardan ayrılmaktadır.

Alanyazında ortaokul seviyesinde yapılan çalışmalar incelediğinde, özellikle üstün yetenekli öğrencilere yönelik çalışmaların az olduğu görülmektedir (Atmatzidou & Demetriadi, 2016). Bu çalışmada ise, Bilim ve Sanat Merkezinde bilişim teknolojileri alanında robotik kodlama atölyesine devam eden ortaokul seviyesindeki üstün yetenekli öğrencilere ilk kez sanal ve fiziksel robotik kodlama araçları kullanılarak metin tabanlı programlamanın mantığını ve alt yapısını oluşturan temel robotik kodlama eğitimi verilmiştir. Çalışma bu yönüyle diğer araştırmalardan farklılık göstermektedir.

İncelenen çalışmalarda, araştırma yöntemi seçiminde karma araştırmaların, nicel ve nitel araştırmalara göre daha fazla kullanıldığı görülmüştür (Hangün vd., 2022). Veri toplama

aracı olarak en çok gözlem formlarının kullanıldığı, bunu akademik başarı testlerinin ve görüşme formlarının izlediği görülmüştür (Yumbul & Bayraktar, 2022). Bu çalışmada ise, hem nicel hem nitel araştırmaların güçlerini bileştiren karma araştırma desenlerinden Çeşitleme yapılmış; nicel ve nitel olarak farklı veri toplama araçları kullanılarak veri çeşitlemesiyle incelenen değişkenlere ait bulguların geçerliği ve güvenilirliği artırılmaya çalışılmıştır.

İncelenen çalışmalarda eğitsel robotiklerle programlama öğretiminde birçok değişkenin ölçüldüğü ve çeşitli sonuçlar elde edildiği görülmüştür. Pekçok çalışmada eğitsel robotiklerin ortaokul öğrencilerinin robotik kodlama başarılarını artırdığı (Eraytaç, 2019; Gandy vd., 2010) ve bilgi işlemsel düşünme becerilerini geliştirdiği (Kert vd., 2020; Ru & Fok, 2022; Zhong & Li, 2019) saptanmıştır. Bazı çalışmalarda eğitsel robotiklerin öğrencilerin aktif olarak öğrenme sürecine katılmalarını sağladığı ve etkin kod yazma becerisi kazandırdığı (Yıldırım, 2020) ve problem çözme becerisine (Çınar & Tüzün, 2020; Konyaoğlu, 2019), yaratıcı ve yansıtıcı düşünme becerilerine (Okuyucu, 2019) katkı sağladığı belirlenmiştir. Eğitsel robotiklerin işbirlikli çalışmaya imkan sunduğu (Kök, 2019), bireysel çalışmalarda proje yönetimine de katkısının olduğu (Yang vd., 2022), gerçek yaşam deneyimlerini öğrencilere yaşatma, öğrencilerin farklı disiplinleri bir arada kullanarak öğrenmelerine yardımcı olma gibi faydalarının olduğunu belirten çalışmalar da mevcuttur (Çayır, 2010). Ayrıca bazı çalışmalarda eğitsel robotiklerin öğrencilerin programlama motivasyonlarına ve programlamaya karşı algılarına olumlu katkıda bulunduğu da görülmüştür (Coşkunserçe, 2023; Strawhacker & Bers, 2019).

Yapılan araştırmalarda her ne kadar farklı sonuçlar ortaya çıkmış olsa da robotik kodlama ile programlama öğretiminin 21.yy becerilerine katkı sağladığı görülmektedir. Bu sebeple bu çalışmada da ortaokul seviyesinde (6. ve 7. sınıf) üstün yetenekli öğrencilere robotik araçlar ile Python programlama dili öğretimi yapılmıştır. Bu çalışmada diğer araştırmalardan farklı olarak; nicel ve nitel veri toplama araçları kullanılarak üstün yetenekli öğrencilerin biri sanal robotik araç olan VEXcode-VR diğeri fiziksel robotik araç olan Raspbberry-Pi kullanarak programlama dili öğrenimi sürecinde; robotik kodlama performanslarının yanı sıra programlama başarıları, öz-yeterlikleri ve kaygıları da karşılaştırılmıştır. Ayrıca her iki gruptaki öğrencilerin robotik kodlama meşguliyetleri duyuşsal, bilişsel, davranışsal, temsili, sosyal boyutlarda incelenmiş ve öğrenci deneyimleri de detaylı olarak ortaya konmuştur. Buna göre çalışmanın robotik kodlama araçlarıyla programlama öğretimi alandaki yeni araştırmalara veri kaynağı sağlayacağı ve yön vereceği düşünülmektedir.

Üstün yetenekli ortaokul öğrencilerinin katılımıyla sanal ve fiziksel robotik kodlama araçları kullanılarak yapılan Python programlama dili eğitiminin çeşitli değişkenlere etkilerinin

nicel ve nitel olarak bütüncül bir şekilde incelendiği bu çalışmanın kapsamı Şekil 6'da özetlenmiştir.

Şekil 6. Çalışma Kapsamı



ÜÇÜNCÜ BÖLÜM

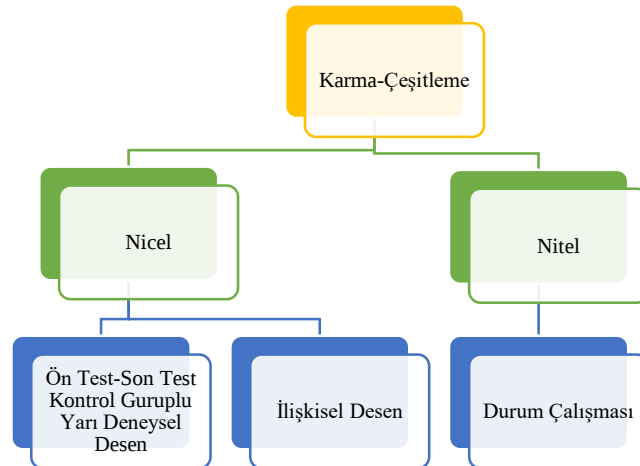
Yöntem

Araştırma Deseni

Çalışmada karma araştırma desenlerinden “Çeşitleme” kullanılmıştır. Böylece, nicel ve nitel veri toplama ve analiz teknikleri birlikte kullanılarak araştırma soruları daha kapsamlı bir şekilde cevaplanmıştır (Creswell, 2010). Bu noktada her iki veri türüne farklı bakış açıları ile bakarak, araştırma problemi hem nicel hem de nitel olarak yorumlanmıştır (Denzin, 2009). Bu araştırma deseni kullanılarak araştırmanın geçerlik ve güvenirliği artırılmaya çalışılmıştır (Johnson & Onwuegbuzie, 2004).

Çalışmanın nicel bölümünde sanal robotik kodlama aracı olan VEXcode-VR ve fiziksel robotik kodlama aracı olan Raspberry-Pi kullanılarak öğretim yapılan grupların seçkisiz atandığı ön test-son test kontrol gruplu yarı deneysel desen kullanılmıştır (Büyüköztürk vd., 2024). Böylece üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlama performansları (kazanımları, meşguliyetleri, başarıları, kalıcılıkları), öz-yeterlikleri ve kaygıları karşılaştırılmıştır. Ayrıca ilişkisel desen kullanılarak öğrencilerin sanal ve fiziksel robotik kodlama başarıları, öz-yeterlikleri ve kaygıları arasındaki ilişkinin anlamlılık düzeyi saptanmıştır (McMillan & Schummacher, 2010). Çalışmanın nitel bölümünde ise, sanal ve fiziksel robotik kodlama gruplarının haftalık kodlama performanslarının, ve robotik kodlama sürecindeki Python programlama dili öğrenme deneyimlerine ilişkin görüşlerinin bütüncül olarak ele alınması ve derinlemesine incelenmesi için durum çalışması yapılmıştır (Şimşek & Yıldırım, 2021). Çalışmada temel alınan araştırma deseni Şekil 7’deki gibidir.

Şekil 7. Çalışmada Temel Alınan Araştırma Deseni



Çalışma Grubu

Çalışma grubu Bitlis ili Tatvan ilçesine bağlı Tatvan Bilim ve Sanat Merkezinde (BİLSEM) 2024-2025 eğitim-öğretim yılında öğrenim gören üstün yetenekli 6. ve 7. sınıf öğrencilerinden oluşmaktadır. Türkiye’de Bilim ve Sanat Merkezlerinde 6. ve 7. sınıf öğrencilerine Bireysel Yetenekleri Fark Ettirme (BYF) Programı uygulanmaktadır. Bu programda farklı sınıf düzeylerindeki öğrencilere atölye dersi adı altında robotik kodlama eğitimi verilerek metin tabanlı bir programlama dili öğretilmektedir (MEB, 2021). Ancak metin tabanlı bir programlama dilini içeren robotik kodlama etkinliklerine katılacak üstün yetenekli öğrencilerin, daha önce blok tabanlı bir kodlama eğitimi almış olmalarının yanı sıra, herhangi bir metin tabanlı programlama dili eğitimi almamış ve sanal/fiziksel robotik kodlama etkinliklerine katılmamış olmaları gerekmektedir. Bu nedenle bu araştırmadaki çalışma grubu; amaçlı örnekleme yöntemi kullanılarak belirlenmiş olup (Büyüköztürk vd., 2024), daha önce blok tabanlı bir kodlama eğitimi almış ve sanal/fiziksel robotik kodlama aracı ile yapılan metin tabanlı Python programlama dili eğitimine ilk kez katılacak olan üstün yetenekli toplam 20 ortaokul öğrencisinden oluşmuştur.

Sanal ve fiziksel robotik kodlama grupları için Python programlama dili öğrenmeye yönelik haftalık kazanımların temel alındığı ve birbirine paralel olarak hazırlanan 10 haftalık etkinlik planlarından birinde VEXcode-VR sanal robotik kodlama uygulaması ile diğerinde ise Raspberry-Pi fiziksel robotik kodlama aracı ile Python programlama dili eğitimi verilmiştir. Her iki gruptaki her bir öğrencinin ilgili robotik aracını bireysel olarak kullanabilmesi için bilgisayar laboratuvarının teknik altyapısı uygun hale getirilmiştir. Sanal robotik kodlama grubundaki öğrenciler için internet alt yapısı olan masaüstü bilgisayarlarda üç boyutlu VEXcode-VR robotik uygulamasının çalışması sağlanmıştır. Fiziksel robotik kodlama grubundaki öğrenciler için ise Raspberry-Pi robotik aracına ait elektronik parçaların ve bilgisayar bluetooth bağlantısının sorunsuz çalışması sağlanmıştır. Haftalık sanal veya fiziksel robotik kodlama etkinliklerinde sunulan problem senaryolarının çözümü için gruptaki öğrencilerin uygun Python kodlarını yazmaları beklenmiştir. Ayrıca 10 haftalık tüm uygulama sonunda, sanal ve fiziksel robotik kodlama gruplarından rastgele seçilen toplam 7 öğrenciyle yüz yüze ve birebir görüşme yapılmıştır. Veri toplama araçlarına ve cinsiyete göre her iki gruptaki öğrenci sayıları Tablo 1’de gösterilmiştir.

Tablo 1. *Veri Toplama Araçlarına ve Cinsiyete göre Her İki Gruptaki Öğrenci Dağılımı*

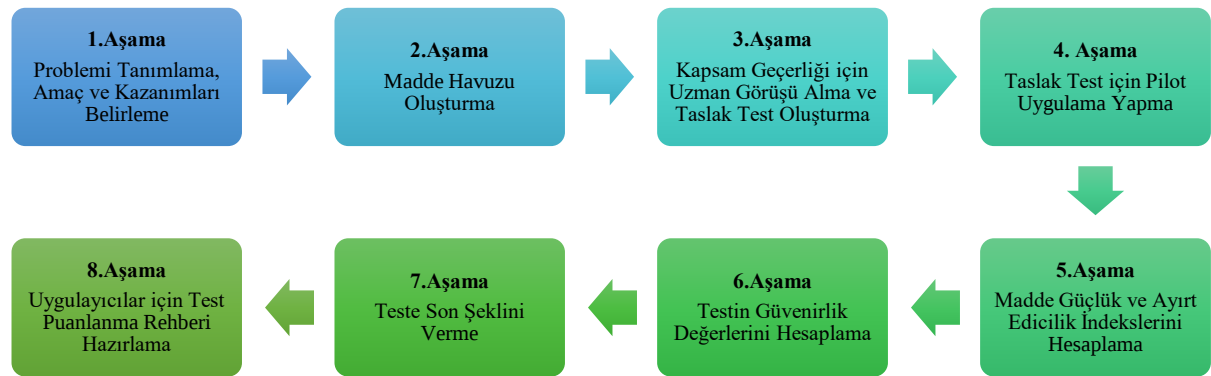
	Sanal Robotik Kodlama (VEXcode-VR)			Fiziksel Robotik Kodlama (Raspberry-Pi)			Genel Toplam
	Kız	Erkek	Toplam	Kız	Erkek	Toplam	
Ön Test-Son Test-Kalıcılık Testi Uygulanan Tüm Örneklem	2	8	10	4	6	10	20
Görüşme Yapılan Örneklem	1	3	4	1	2	3	7

Veri Toplama Araçları

Programlama Başarı Testi

Alanyazın incelendiğinde, VEXcode-VR (sanal) veya Raspberry-Pi (fiziksel) robotik kodlama araçları ile Python programlama dili öğrenme seviyesini belirlemek için geçerlik ve güvenirlik analizleri yapılmış olan başarı testlerine ulaşılamamıştır. Bu durum bu yönde özgün başarı testleri geliştirme sürecini de beraberinde getirmiştir. Sanal veya fiziksel robotik kodlama gruplarındaki üstün yetenekli öğrencilerin Python programlama dili öğrenme seviyelerini tespit etmek için, 10 haftalık etkinliklerdeki kazanımların temel alındığı, birbirine paralel ve denk iki başarı testi geliştirilmiştir. Başarı testlerinin birinde VEXcode-VR sanal robotik kodlama aracı ile diğerinde ise Raspberry-Pi fiziksel robotik kodlama aracı ile Python programlama dili eğitimine yönelik sorular yer almıştır. Her ne kadar sanal ve fiziksel robotik kodlama başarı testleri birbirine paralel ve denk hazırlanmış olsa da, kullanılan VEXcode-VR (sanal) ve Raspberry-Pi (fiziksel) robotik kodlama ortamlarının farklı özellikleri olduğu için, her bir başarı testinde ilgili robotik kodlama aracını tanımaya yönelik sorular da sorulmuştur. Başarı testlerinin geliştirilme süreci (Büyüköztürk vd., 2023) Şekil 8’de özetlenmiştir.

Şekil 8. Başarı Testi Geliştirme Süreci



Şekil 8’de sunulan başarı testi geliştirme sürecinin her bir aşamasında yapılanlar devam eden bölümde detaylı olarak açıklanmıştır.

1. Aşama: Problemi Tanımlama, Amaç ve Kazanımları Belirleme. Programlama eğitimi kazanımları belirlenirken; BİLSEM 6. ve 7. sınıf öğrencileri için uygulanan Bireysel Yetenekleri Fark Ettirme (BYF) programı kapsamındaki atölye dersinde yer alan robotik kodlama eğitimi kazanımlarının (MEB, 2016, 2021) yanı sıra, MEB altıncı sınıf Bilişim Teknolojileri ve Yazılım dersi kitabındaki problem çözme ve programlama kazanımları (MEB, 2018) ve MEB ortaöğretim Bilgisayar Bilimi Dersi Kur-1 kitabındaki Python programlama dili kazanımları (MEB, 2020) da incelenmiştir. Ayrıca literatürde yer alan Python programlama dili öğretimi amacıyla geliştirilmiş başarı testlerindeki sorular da incelenmiştir (Ünal, 2019; Unal & Topu, 2022). Buna göre, “*Robotik Programlama Ortamını ve Programlama Araçlarını Tanıma, Değişkenler ve Sabitler, Döngüler ve Tekrarlı Yapılar, Fonksiyonları Uygulama, Koşullu İfadeler, Listeler ve Dizi Tipleri*” ünite-konu başlıklarında yer alan kazanımlar dikkate alınarak VEXcode-VR sanal robotik kodlama aracı ve Raspberry-Pi fiziksel robotik kodlama aracına yönelik birbirine paralel ve denk olan iki ayrı Python programlama dili başarı testinin oluşturulmasına karar verilmiştir.

2. Aşama: Madde Havuzu Oluşturma. Sanal ve fiziksel robotik kodlama gruplarına ait iki ayrı Python programlama dili başarı testi için, aynı kazanıma ait soru numaralarının aynı sırada yer aldığı, çoktan seçmeli ve beş seçenekli, birbirine paralel ve denk olan her biri 32’şer soruluk iki ayrı madde havuzu oluşturulmuştur. Bu iki ayrı madde havuzundaki sorular; kapsam geçerliği, her iki testteki soruların birbirine denkliği, öğrenci seviyesine uygunluğu ve Türkçe dil kullanımı açısından robotik kodlama eğitimi veren 5 BİLSEM uzmanı ve 1 öğretim teknolojileri uzmanı tarafından ayrı ayrı değerlendirilmiştir.

3. Aşama: Kapsam Geçerliği için Uzman Görüşü Alma ve Taslak Formu Oluşturma. Konu alanı uzmanlarının testlerin kapsam geçerliliğine ilişkin görüşleri ilk olarak uyuşma oranları hesaplanarak değerlendirilmiştir. Uyuşma oranı .80 veya daha büyük olan maddelerin yüksek oranda kapsam geçerliliğine sahip olduğu söylenebilir (Yurdagül & Bayrak, 2012; Johnson & Onwuegbuzie, 2004). Ayrıca Lawshe (1975) tekniği olarak da bilinen bir teknik ile test maddelerine ilişkin Kapsam Geçerlik Oranları (KGO) hesaplanmıştır. KGO’lar, uzmanların testlerde bulunan her bir maddeye yönelik görüşleri toplanarak aşağıdaki formülün uygulanmasıyla elde edilmektedir (Yurdagül, 2005).

$\frac{N_G - \frac{N}{2}}{\frac{N}{2}}$ NG: Maddeye ilişkin “Uygun” görüşünü belirten uzman sayısı N: Toplam uzman sayısı

KGO değerinin -1.0 (uygun değil) ile +1.0 (uygun) arasında olduğu, pozitif KGO için $\alpha=0.05$ anlamlılık seviyesindeki minimum değerleri (kapsam geçerlilik ölçütleri için kritik değerler alanyazında tablolatırılmıştır (Veneziano & Hooper, 1997; Greene vd., 1989). Bahsi geçen tabloya göre, bu çalışmada 6 uzmandan görüş alındığı için minimum kritik değer 0.99'dur. Uzman sayısı ile ilişkili olarak minimum KGO değerleri, maddenin istatistiksel olarak anlamlılığını ifade etmektedir (Yurdağül, 2005). Lawshe (1975)'a göre, her bir maddeye ait KGO değeri, bahsi geçen bu kritik değere eşit (bu çalışma için 0.99) veya büyükse o madde, test için uygun madde olarak kabul edilmektedir. Test için uygun olduğu belirlenen bu maddelerin tümüne ait KGO değerlerinin ortalaması alınarak da testin Kapsam Geçerlik İndeksi (KGİ) belirlenmektedir. KGO değeri KGİ değerine eşit veya büyük olan maddeler için testin kapsam geçerliğinin sağlandığı söylenebilir. Buna göre, her iki testin madde havuzundaki her bir soruya ilişkin uzman değerlendirmesine göre uyuma oranı ve KGO değerleri Tablo 2'de sunulmuştur. Tablo satırları, testte kalan maddeler için (Yeşil), testten çıkarılan maddeler için (Turuncu) ile vurgulanmıştır.

Tablo 2. Sanal ve Fiziksel Robotik Kodlama Başarı Testlerinin Uzman Görüşü Uyuma Oranı, Kapsam Geçerlik Oranı ve İndeksi

Madde No (Her iki test için)	Sanal Robotik Kodlama Başarı Testi (VEXcode-VR)					Fiziksel Robotik Kodlama Başarı Testi (Raspberry-Pi)				
	Uygun	Uygun Değil	Düzeltilme Önerisi (Gerekli ise)	Uyuma Oranı (%)	KGO	Uygun	Uygun Değil	Düzeltilme Önerisi (Gerekli ise)	Uyuma Oranı (%)	KGO
Madde1	6	0		100	1	5	1	1 uzman "pass" seçeneğinin kullanılmasını uygun görmediği için soruda düzeltme yapılmıştır.	84	.66
Madde2	6	0		100	1	6	0		100	1
Madde3	0	6	6 uzman sorunun kazanım dışı olduğunu belirttiği için soru testten çıkartılmıştır.	.00	-1	0	6	6 uzman sorunun kazanım dışı olduğunu belirttiği için soru testten çıkartılmıştır.	.00	-1
Madde4	1	5	5 uzman soru ve şıkların uyumsuz olduğunu belirttiği için soru testten çıkartılmıştır.	.16	-.66	1	5	5 uzman soru ve şıkların uyumsuz olduğunu belirttiği için soru testten çıkartılmıştır.	.16	-.66
Madde5	5	1	1 uzman soru kökünün sadeleştirilmesi gerektiğini belirttiği için soruda düzeltme yapılmıştır.	84	.66	6	0		100	1

Tablo 2 (Devamı)

Madde6	6	0		100	1	6	0		100	1
Madde7	6	0		100	1	5	1	1 uzman soru kökünün sadeleştirilmesi gerektiğini belirttiği için soruda düzeltme yapılmıştır.	84	.66
Madde8	5	1	1 uzman "pass" seçeneğinin kullanılması uygun görmediği için soruda düzeltme yapılmıştır.	84	.66	6	0		100	1
Madde9	5	1	1 uzman soru kökünün sadeleştirilmesi gerektiğini belirttiği için soruda düzeltme yapılmıştır.	84	.66	6	0		100	1
Madde10	6	0		100	1	6	0		100	1
Madde11	6	0		100	1	5	1	1 uzman soru kökünün sadeleştirilmesi gerektiğini belirttiği için soruda düzeltme yapılmıştır.	84	.66
Madde12	6	0		100	1	6	0		100	1
Madde13	6	0		100	1	6	0		100	1
Madde14	1	5	5 uzman sorunun kapsam dışı olduğunu belirttiği için soru testten çıkartılmıştır.	.16	-.66	1	5	5 uzman sorunun kapsam dışı olduğu belirttiği için soru testten çıkartılmıştır.	.16	-.66
Madde15	1	5	5 uzman soru ve şıkların uyumsuz olduğunu belirttiği için soru testten çıkartılmıştır.	.16	-.66	6	0		100	1
Madde16	6	0		100	1	1	5	5 uzman soru ve şıkların uyumsuz olduğunu belirttiği için soru testten çıkartılmıştır.	.16	-.66
Madde17	6	0		100	1	6	0		100	1
Madde18	2	4	4 uzman sorunun öğrenci seviyesinin üstünde olduğunu belirttiği için soru testten çıkartılmıştır.	.25	-.33	1	5	5 uzman sorunun kapsam dışı olduğu belirttiği için soru testten çıkartılmıştır.	.16	-.66

Tablo 2 (Devamı)

Madde19	6	0		100	1	6	0		100	1
Madde20	5	1	1 uzman soru kökünün sadeleştirilmesi gerektiğini belirttiği için soruda düzeltme yapılmıştır.	84	.66	5	1	1 uzman soru kökünün sadeleştirilmesi gerektiğini belirttiği için soruda düzeltme yapılmıştır.	84	.66
Madde21	6	0		100	1	6	0		100	1
Madde22	6	0		100	1	6	0		100	1
Madde23	6	0		100	1	6	0		100	1
Madde24	5	1	1 uzman şıklardaki çeldiricilerin birbirine yakın olduğunu belirttiği için çeldiricilerde düzeltme yapılmıştır.	84	.66	5	1	1 uzman şıklardaki çeldiricilerin birbirine yakın olduğunu belirttiği için çeldiricilerde düzeltme yapılmıştır.	84	.66
Madde25	5	1	1 uzman soru kökünde düzeltme yapılması gerektiğini belirttiği için soru kökü değiştirilmiştir.	84	.66	5	1	1 uzman soru kökünde düzeltme yapılması gerektiğini belirttiği için soru kökü değiştirilmiştir.	84	.66
Madde26	6	1		100	1	6	1		100	1
Madde27	6	0		100	1	2	4	4 uzman sorunun öğrenci seviyesinin üstünde olduğunu belirttiği için soru testten çıkartılmıştır.	.25	-.33
Madde28	6	0		100	1	6	0		100	1
Madde29	2	4	4 uzman sorunun öğrenci seviyesinin üstünde olduğunu belirttiği için soru testten çıkartılmıştır.	.25	-.33	6	0		100	1
Madde30	6	0		100	1	6	0		100	1
Madde31	2	4	4 uzman sorunun kapsam dışı olduğu belirttiği için soru testten çıkartılmıştır.	.25	-.33	2	4	4 uzman sorunun kapsam dışı olduğu belirttiği için soru testten çıkartılmıştır.	.25	-.33
Madde32	6	0		100	1	6	0		100	1

Tablo 2’de görüldüğü üzere, 32’şer maddelik soru havuzundan oluşan her iki test için uygun olmayan maddeler testlerden çıkarılmış, uzmanların uyuşma oranı .80 ve üzeri olan maddeler testlerde kalmıştır. Sanal ve fiziksel robotik kodlama başarı testlerinde kalan bu maddeler arasından ise uzman önerilerine göre gerekli düzeltmeler yapılan maddelerin, tekrar

bu uzmanlara gösterilerek testler için uygunlukları doğrulanmıştır. Böylece her iki testte 6 uzman tarafından da görüş birliğiyle testte kalması uygun görülen 25'er maddenin her birinin KGO değeri 1 olarak hesaplanmıştır. Buna bağlı olarak bu 25'er maddenin KGO ortalamalarına göre her iki başarı testinin KGİ değeri de 1 olarak hesaplanmış ve kapsam geçerliği yeterli düzeyde olan 25'er soruluk birbirine paralel ve denk iki ayrı başarı testi elde edilmiştir. Ayrıca bu testlerdeki tüm soruların anlaşılabilirliğini denetlemek için; uygulama öncesinde taslak başarı testi Türkçe dil uzmanına gösterilmiş ve daha önce robotik kodlama eğitimi alan 1 kız ve 1 erkek BİLSEM öğrencisine sorular okutulmuştur.

Son hali 25'er sorudan oluşan ve soru numaraları yeniden düzenlenen bu iki başarı testinde yer alan bazı sorular, üniteye-konuya uygun olan hedef-kazanımın yanı sıra birden fazla kazanımı da ölçerken bazı sorular ise tek bir hedef-kazanımı ölçmektedir. Buna göre Tablo 3'te testlerdeki her bir sorunun hedef-kazanımlara göre dağılımı sunulmuştur.

Tablo 3. *Python Programlama Dili Kazanımlarına göre Sanal ve Fiziksel Robotik Kodlama Başarı Testlerindeki Soru Dağılımı*

Ünite-Konu	Kazanım	Soru Numarası (sanal ve fiziksel robotik kodlama testlerindeki soru numaraları aynı sıradadır)
Robotik Programlama Ortamını ve Programlama Araçlarını Tanıma	Kodlama yapacağı (sanal ve fiziksel) aracı tanır.	1,2,3
Değişkenler ve Sabitler	Öğretimi gerçekleştirilecek programlama dilinde değişkenleri ve sabitleri uygun şekilde kullanarak programlar geliştirir.	3,4,5,6,7
Döngüler Tekrarlı Yapılar	Öğretimi gerçekleştirilecek programlama dilinde tekrarlı yapıları kullanarak programlar yazar.	8,9,10,11,12
Fonksiyonlar Uygulama	Öğretimi gerçekleştirilecek programlama dilinde parametre alan, değer döndüren, fonksiyon içeren programlar geliştirir.	13,14,15,16,17
Koşullu İfadeler	Öğretimi gerçekleştirilecek programlama dilinde kontrol yapılarını kullanarak programlar geliştirir.	18,19,20,21,22
Listeler Dizi Tipleri	Öğretimi gerçekleştirilecek programlama dilinde dizi ve listelere tipinde veri tanımlama işlemi yapar.	23,24,25

4. Aşama: Pilot Uygulama. Birbirine paralel ve denk olarak hazırlanan taslak halindeki 25'er soruluk sanal ve fiziksel robotik kodlama başarı testlerinin pilot uygulamasını yapmak amacıyla daha önce VEXcode-VR veya Raspberry-Pi ile Python programlama dili dersi almış

öğrenci grupları araştırılmıştır. Ancak araştırmada uygulanacak bu robotik araçlar ile yapılacak Python programlama dili öğretiminin MEB’den izin alınan ortaokul düzeyindeki eğitim kurumlarında daha önce hiç uygulanmamış olduğu belirlenmiştir. Bu nedenle sanal ve fiziksel robotik kodlama başarı testlerinin pilot uygulaması iki aşamada yapılmıştır (Pilot Uygulama I ve II). Bu iki pilot uygulama, üç farklı eğitim kurumundaki ve iki farklı sınıf düzeyindeki katılımcılarla gerçekleştirilmiştir. Her iki pilot uygulama esnasında da her sınıfta bulunan öğrenci sayısı dikkate alınarak sınıfın yarısına rastgele olacak şekilde VEXcode-VR başarı testi, diğer yarısına Raspberry-Pi başarı testi dağıtılmıştır. Öğrencilerin testi doldurmaları için bir ders saati (40 dk) süresi verilmiştir.

Pilot Uygulama-I, Bitlis Tatvan’daki MEB’e bağlı bir devlet ortaokulunda Bilişim Teknolojileri derslerinde algoritma ve blok tabanlı programlama dersi almış 7. sınıfta öğrenim gören 171 öğrenciyle yapılmıştır. Pilot Uygulama-II ise, 11. sınıf öğrencilerine uygulanmıştır. Bu kapsamdaki uygulama, MEB’e bağlı bir devlet Anadolu Lisesinin Bilgisayar Bilimi dersinde Python programlama dili temelleri eğitimi almış 136 öğrenci ve bir BİLSEM eğitim kurumunun robotik kodlama etkinliklerindeki herhangi bir programlama dili eğitimine katılmış 77 öğrenci olmak üzere toplam 213 öğrenciyle yapılmıştır. Pilot I ve II uygulamalarına katılan tüm öğrencilerin demografik bilgileri Tablo 4’te sunulmuştur.

Tablo 4. *Pilot Uygulamalara Katılan Öğrencilerin Demografik Özellikleri*

	Eğitim Kurumu Türü	Sınıf Düzeyi	Programlama Eğitimi Seviyesi	Kız	Erkek	Toplam
<i>Pilot Uygulama-I</i>	Bitlis Tatvan Devlet Ortaokulu	7. Sınıf	Bilişim Teknolojileri derslerinde algoritma ve blok tabanlı programlama dersi almış	92	79	171
<i>Pilot Uygulama-II</i>	Bitlis Tatvan Devlet Anadolu Lisesi	11. Sınıf	Bilgisayar Bilimi dersinde Python programlama dili temelleri eğitimi almış	91	45	136
	Bitlis Bilim ve Sanat Merkezi (BİLSEM)	11. Sınıf	Robotik kodlama etkinliklerindeki herhangi bir programlama dili eğitimine katılmış	25	22	77
Toplam				208	146	354

Tablo 4’teki katılımcı profiliyle yapılan her iki pilot uygulamadan elde edilen verilere ilişkin başarı testi geçerlik ve güvenirlik analizi sonuçları devam eden aşamalarda detaylarıyla açıklanmıştır.

5. Aşama: Madde Güçlük ve Ayırt Edicilik Analizleri: Sanal veya fiziksel robotik kodlama tabanlı Python programlama diline yönelik başarı testlerinin kapsam geçerliğini

belirlemek için madde analizleri yapılmış, madde güçlük indeksi (MGİ) ve madde ayırt edicilik indeksi (MAİ) belirlenmiştir. Araştırmanın hedef kitlesi olan asıl uygulamanın yapılacağı BİLSEM ortaokul öğrencileriyle aynı sınıf düzeyinde olan bir ortaokuldaki 7. sınıf öğrencilerine uygulanan Pilot Uygulama-I verileri kullanılarak alt ve üst gruplar %27 grup ortalamaları madde analizi ile madde güçlük ve madde ayırt edicilik indeksleri hesaplanmıştır (Büyüköztürk vd., 2023). Sonuçlar Tablo 5’te sunulmuştur.

Tablo 5. *Pilot Uygulama-I Verilerine İlişkin Sanal ve Fiziksel Robotik Programlama Başarı Testlerinin Madde Analizleri*

Pilot Uygulama-I Madde No (Her iki test için)	Sanal Robotik Kodlama Başarı Testi (VEXcode-VR)			Fiziksel Robotik Kodlama Başarı Testi (Raspberry-Pi)		
	MGİ (pj)	MAİ (D)	Açıklama	MGİ (pj)	MAİ (D)	Açıklama
1.Madde	0.41	0.25	Orta güçlükte ve düşük düzeyde ayırt edici madde	0.28	0.13	Zor ve çok düşük düzeyde ayırt edici madde
2.Madde	0.34	0.25	Orta güçlükte ve düşük düzeyde ayırt edici madde	0.26	0.26	Zor ve düşük düzeyde ayırt edici madde
3.Madde	0.36	0.07	Orta güçlükte ve ters yönde çalışmış madde çok düşük düzeyde ayırt edici madde	0.39	0.26	Orta güçlükte ve düşük düzeyde ayırt edici madde
4.Madde	0.13	0.04	Zor ve ters yönde çalışmış madde çok düşük düzeyde ayırt edici madde	0.15	0.26	Zor ve düşük düzeyde ayırt edici madde
5.Madde	0.25	0.43	Zor ve yüksek düzeyde ayırt edici madde	0.13	0.04	Zor ve çok düşük düzeyde ayırt edici madde
6.Madde	0.16	0.32	Zor ve orta düzeyde ayırt edici madde	0.28	0.17	Zor ve çok düşük düzeyde ayırt edici madde
7.Madde	0.32	0.36	Orta güçlükte ve orta düzeyde ayırt edici madde	0.28	0.13	Zor ve çok düşük düzeyde ayırt edici madde
8.Madde	0.36	0.64	Orta güçlükte ve yüksek düzeyde ayırt edici madde	0.28	0.40	Zor ve orta düzeyde ayırt edici madde
9.Madde	0.20	0.32	Zor ve orta düzeyde ayırt edici madde	0.28	0.40	Zor ve yüksek düzeyde ayırt edici madde
10.Madde	0.34	0.39	Orta güçlükte ve orta düzeyde ayırt edici madde	0.33	0.30	Orta güçlükte ve orta düzeyde ayırt edici madde
11.Madde	0.21	0.36	Zor ve orta düzeyde ayırt edici madde	0.28	0.22	Zor ve düşük düzeyde ayırt edici madde
12.Madde	0.27	0.32	Zor ve orta düzeyde ayırt edici madde	0.33	0.40	Orta güçlükte ve orta düzeyde ayırt edici madde
13.Madde	0.41	0.46	Orta güçlükte ve yüksek düzeyde ayırt edici madde	0.35	0.26	Orta güçlükte ve düşük düzeyde ayırt edici madde
14.Madde	0.14	0.07	Zor ve ters yönde çalışmış çok düşük düzeyde ayırt edici madde	0.41	0.48	Orta güçlükte ve yüksek düzeyde ayırt edici madde
15.Madde	0.41	0.61	Orta güçlükte ve yüksek düzeyde ayırt edici madde	0.46	0.65	Orta güçlükte ve yüksek düzeyde ayırt edici madde

Tablo 5 (Devamı)

16.Madde	0.11	0.07	Zor ve çok düşük düzeyde ayırt edici madde	0.22	0.26	Zor ve düşük düzeyde ayırt edici madde
17.Madde	0.27	0.46	Zor ve yüksek düzeyde ayırt edici madde	0.04	0.00	Zor ve çok düşük düzeyde ayırt edici madde
18.Madde	0.34	0.39	Orta güçlükte ve orta düzeyde ayırt edici madde	0.09	0.09	Zor ve çok düşük düzeyde ayırt edici madde
19.Madde	0.45	0.61	Orta güçlükte ve yüksek düzeyde ayırt edici madde	0.22	0.26	Zor güçlükte ve düşük düzeyde ayırt edici madde
20.Madde	0.18	0.00	Zor ve çok düşük düzeyde ayırt edici madde	0.28	0.13	Zor ve çok düşük düzeyde ayırt edici madde
21.Madde	0.45	0.46	Orta güçlükte ve yüksek düzeyde ayırt edici madde	0.22	0.26	Zor ve düşük düzeyde ayırt edici madde
22.Madde	0.23	0.18	Zor ve çok düşük düzeyde ayırt edici madde	0.20	0.30	Zor ve orta düşük düzeyde ayırt edici madde
23.Madde	0.27	0.11	Zor ve çok düşük düzeyde ayırt edici madde	0.54	0.40	Kolay güçlükte ve yüksek düzeyde ayırt edici madde
24.Madde	0.20	0.25	Zor ve düşük düzeyde ayırt edici madde	0.24	0.40	Zor ve yüksek düzeyde ayırt edici madde
25.Madde	0.45	0.39	Orta güçlükte ve orta düzeyde ayırt edici madde	0.26	0.26	Zor güçlükte ve orta düzeyde ayırt edici madde

Tablo 5 incelendiğinde sanal ve fiziksel robotik kodlama başarı testlerindeki bazı soruların zorluk seviyelerinin yüksek ve ayırt edicilik düzeylerinin düşük olduğu görülmektedir (Büyüköztürk vd., 2023). Ancak 7. sınıfta öğrenim gören öğrencilerin Bilişim Teknolojileri derslerinde algoritma ve blok tabanlı programlama dersi almış olsalar bile, daha önce VEXcode-VR ve Raspberry-Pi robotik kodlama araçlarıyla etkinlik yapmadığı ve herhangi bir metin tabanlı programlama dili eğitimi almadığı düşünüldüğünde; bu sonucun ortaya çıkması doğal bir durumdur. Bu nedenle sanal ve fiziksel robotik kodlama tabanlı Python programlama diline yönelik başarı testlerinin kapsam geçerliğini belirlemek için iki farklı eğitim kurumunun 11. sınıfındaki toplam 213 öğrenciyle Pilot Uygulama-II yapılmıştır. Bu öğrenciler araştırmanın hedef kitlesi olan asıl uygulamanın yapılacağı BİLSEM ortaokul öğrencileriyle aynı sınıf düzeyinde değildir. Ancak 136 öğrencinin Anadolu Lisesi Bilgisayar Bilimi Dersinde Python programlama dili eğitimi aldığı, 77 öğrencinin ise bir BİLSEM eğitim kurumunun robotik kodlama etkinliklerindeki herhangi bir programlama dili eğitime katıldığı için bu öğrencilere başarı testleri uygulanmıştır. Pilot Uygulama-II verileri kullanılarak alt ve üst gruplar %27 grup ortalamaları madde analizi ile madde güçlük ve madde ayırt edicilik indeksleri hesaplanmıştır. Sonuçlar Tablo 6’da sunulmuştur.

Tablo 6. Pilot Uygulama-II Verilerine İlişkin Sanal ve Fiziksel Robotik Programlama Başarı Testlerinin Madde Analizleri

Pilot Uygulama-II Madde No (Her iki test için)	Sanal Robotik Kodlama Başarı Testi (VEXcode-VR)			Fiziksel Robotik Kodlama Başarı Testi (Raspberry-Pi)		
	MGİ (pj)	MAİ (D)	Açıklama	MGİ (pj)	MAİ (D)	Açıklama
1.Madde	0.52	0.54	Kolay güçlükte ve yüksek düzeyde ayırt edici madde	0.50	0.58	Kolay ve yüksek düzeyde ayırt edici madde
2.Madde	0.54	0.75	Kolay güçlükte ve yüksek düzeyde ayırt edici madde	0.54	0.67	Kolay ve yüksek düzeyde ayırt edici madde
3.Madde	0.50	0.75	Kolay güçlükte ve yüksek düzeyde ayırt edici madde	0.56	0.71	Kolay ve yüksek düzeyde ayırt edici madde
4.Madde	0.52	0.63	Kolay güçlükte ve yüksek düzeyde ayırt edici madde	0.50	0.83	Kolay ve yüksek düzeyde ayırt edici madde
5.Madde	0.54	0.50	Kolay güçlükte ve yüksek düzeyde ayırt edici madde	0.56	0.79	Kolay ve yüksek düzeyde ayırt edici madde
6.Madde	0.52	0.88	Kolay güçlükte ve yüksek düzeyde ayırt edici madde	0.56	0.63	Kolay ve yüksek düzeyde ayırt edici madde
7.Madde	0.52	0.54	Kolay güçlükte ve yüksek düzeyde ayırt edici madde	0.42	0.58	Orta Güçlükte ve yüksek düzeyde ayırt edici madde
8.Madde	0.54	0.75	Kolay güçlükte ve yüksek düzeyde ayırt edici madde	0.67	0.67	Kolay ve yüksek düzeyde ayırt edici madde
9.Madde	0.56	0.63	Kolay güçlükte ve yüksek düzeyde ayırt edici madde	0.67	0.42	Kolay ve yüksek düzeyde ayırt edici madde
10.Madde	0.63	0.58	Kolay güçlükte ve yüksek düzeyde ayırt edici madde	0.56	0.71	Kolay ve yüksek düzeyde ayırt edici madde
11.Madde	0.65	0.54	Kolay güçlükte ve yüksek düzeyde ayırt edici madde	0.58	0.83	Kolay ve yüksek düzeyde ayırt edici madde
12.Madde	0.63	0.58	Kolay güçlükte ve yüksek düzeyde ayırt edici madde	0.46	0.75	Orta Güçlükte ve yüksek düzeyde ayırt edici madde
13.Madde	0.58	0.58	Kolay güçlükte ve yüksek düzeyde ayırt edici madde	0.69	0.46	Kolay ve yüksek düzeyde ayırt edici madde
14.Madde	0.48	0.79	Orta ve yüksek düzeyde ayırt edici madde	0.48	0.63	Orta güçlükte ve yüksek düzeyde ayırt edici madde
15.Madde	0.69	0.29	Kolay güçlükte ve düşük düzeyde ayırt edici madde	0.65	0.38	Kolay güçlükte ve orta düzeyde ayırt edici madde
16.Madde	0.60	0.63	Kolay güçlükte ve yüksek düzeyde ayırt edici madde	0.52	0.63	Kolay ve yüksek düzeyde ayırt edici madde
17.Madde	0.56	0.54	Kolay güçlükte ve yüksek düzeyde ayırt edici madde	0.35	0.54	Zor güçlükte ve yüksek düzeyde ayırt edici madde
18.Madde	0.75	0.25	Çok Kolay güçlükte ve düşük düzeyde ayırt edici madde	0.50	0.75	Kolay ve yüksek düzeyde ayırt edici madde
19.Madde	0.71	0.33	Çok Kolay güçlükte ve orta düzeyde ayırt edici madde	0.48	0.79	Zor güçlükte ve yüksek düzeyde ayırt edici madde
20.Madde	0.52	0.88	Kolay güçlükte ve yüksek düzeyde ayırt edici madde	0.63	0.42	Kolay ve yüksek düzeyde ayırt edici madde
21.Madde	0.54	0.58	Kolay güçlükte ve yüksek düzeyde ayırt edici madde	0.50	0.49	Orta güçlükte ve yüksek düzeyde ayırt edici madde
22.Madde	0.54	0.67	Kolay güçlükte ve yüksek düzeyde ayırt edici madde	0.44	0.63	Zor ve yüksek düzeyde ayırt edici madde
23.Madde	0.49	0.75	Orta ve yüksek düzeyde ayırt edici madde	0.56	0.54	Kolay ve yüksek düzeyde ayırt edici madde
24.Madde	0.48	0.63	Orta güçlükte ve yüksek düzeyde ayırt edici madde	0.50	0.83	Kolay ve yüksek düzeyde ayırt edici madde
25.Madde	0.63	0.17	Kolay güçlükte ve düşük düzeyde ayırt edici madde	0.31	0.46	Zor güçlükte ve yüksek düzeyde ayırt edici madde

Tablo 6 incelendiğinde sanal ve fiziksel robotik kodlama başarı testlerindeki soruların yarısından fazlasının kolay veya orta güçlükte ve ayırt edici madde olduğu görülmektedir (Büyüköztürk vd., 2023). Buna göre önceki pilot uygulamada ortaya çıkan olumsuz sonuçların bu pilot uygulamada olumluya döndüğü söylenebilir. Testin uygulandığı öğrencilerin metin tabanlı bir programlama dili eğitimi almış veya robotik kodlama etkinliklerinde herhangi bir programlama dili eğitimine katılmış olmalarının soruların madde güçlüğü ve ayırt ediciliğini olumlu yönde etkilediği düşünülmektedir.

6. Aşama: Güvenirlik Analizleri. Programlama başarı testlerinin güvenilirliğini tespit etmek amacıyla sorular arası iç tutarlık analizi yapılmıştır. Pilot Uygulama I ve II verileri için ayrı ayrı Kuder-Richardson (KR-20) güvenirlilik katsayısı hesaplanmış ve Tablo 7’de sunulmuştur.

Tablo 7. *Pilot Uygulama I ve II için Kuder-Richardson (KR-20) Güvenirlilik Değerleri*

	Sanal Robotik Kodlama Başarı Testi (VEXcode-VR)	Fiziksel Robotik Kodlama Başarı Testi (Raspberry-Pi)
Pilot Uygulama-I	.68<.70	.41<.70
Pilot Uygulama-II	.88>.70	.88>.70

Tablo 7 incelendiğinde, Pilot Uygulama-I için her iki başarı testinin güvenirlilik katsayısı düşük olsa da ($KR-20_{\text{sanal}} = .68 < .70$; $KR-20_{\text{fiziksel}} = .41 < .70$), Pilot Uygulama-II için bu değerlerin yüksek olduğu görülmektedir ($KR-20_{\text{sanal}} = .88 > .70$; $KR-20_{\text{fiziksel}} = .88 > .70$). Bu sonuçlar, hem sanal hem de fiziksel robotik kodlama başarı testlerinin güvenilir birer ölçme aracı olduğunu kanıtlamaktadır.

7. Aşama: Birbirine Paralel ve Denk İki Ayrı Başarı Testinin Son Hali. Geçerlik ve güvenirlilik analizlerinden sonra her biri çoktan seçmeli ve beş seçenekli 25’er sorudan oluşan sanal ve fiziksel robotik kodlama ile Python programlama diline yönelik hazırlanan iki ayrı başarı testinin geçerli ve güvenilir birer ölçme aracı olduğu belirlenmiştir. Son hali verilen başarı testlerindeki her bir maddenin yenilenen Bloom Taksonomisindeki (Anderson vd., 2001) bilişsel süreç boyutlarına (Hatırlama, Anlama, Uygulama, Çözümleme, Değerlendirme, Yaratma) göre dağılımı Tablo 8’de sunulmuştur.

Tablo 8. Bloom Taksonomisi Bilişsel Süreç Boyutlarına Göre Başarı Testi Madde Dağılımı

Konu/ Öğrenme Alanı	Kazanım	Madde No (Her iki test için)	Hatırlama	Anlama	Uygulama	Çözümleme
Programlama Araçlarını Tanıma	Kodlama yapacağı (sanal ve fiziksel) aracı tanır.	Madde1	✓			
Programlama Araçlarını Tanıma	Kodlama yapacağı (sanal ve fiziksel) aracı tanır.	Madde2	✓			
Programlama Araçlarını Tanıma	Kodlama yapacağı (sanal ve fiziksel) aracı tanır.	Madde3	✓			
Değişkenler ve Sabitler	Öğretimi gerçekleştirilecek programlama dilinde değişkenleri ve sabitleri uygun şekilde kullanarak programlar geliştirir.	Madde4		✓		
Değişkenler ve Sabitler	Öğretimi gerçekleştirilecek programlama dilinde değişkenleri ve sabitleri uygun şekilde kullanarak programlar geliştirir.	Madde5	✓			
Değişkenler ve Sabitler	Öğretimi gerçekleştirilecek programlama dilinde değişkenleri ve sabitleri uygun şekilde kullanarak programlar geliştirir.	Madde6		✓		
Değişkenler ve Sabitler	Öğretimi gerçekleştirilecek programlama dilinde değişkenleri ve sabitleri uygun şekilde kullanarak programlar geliştirir.	Madde7		✓		
Döngüler ve Tekrarlı Yapılar	Öğretimi programlama dilinde tekrarlı yapıları kullanarak programlar yazar.	Madde8				
Döngüler ve Tekrarlı Yapılar	Öğretimi programlama dilinde tekrarlı yapıları kullanarak programlar yazar.	Madde9		✓		
Döngüler ve Tekrarlı Yapılar	Öğretimi programlama dilinde tekrarlı yapıları kullanarak programlar yazar.	Madde10			✓	
Döngüler ve Tekrarlı Yapılar	Öğretimi programlama dilinde tekrarlı yapıları kullanarak programlar yazar.	Madde11		✓		
Döngüler ve Tekrarlı Yapılar	Öğretimi programlama dilinde tekrarlı yapıları kullanarak programlar yazar.	Madde12			✓	
Fonksiyonları Uygulama	Öğretimi programlama dilinde parametre alan değer döndüren fonksiyon içeren programlar geliştirir.	Madde13			✓	
Fonksiyonları Uygulama	Öğretimi programlama dilinde parametre alan değer döndüren fonksiyon içeren programlar geliştirir.	Madde14			✓	
Fonksiyonları Uygulama	Öğretimi programlama dilinde parametre alan değer döndüren fonksiyon içeren programlar geliştirir.	Madde15				✓
Fonksiyonları Uygulama	Öğretimi programlama dilinde parametre alan değer döndüren fonksiyon içeren programlar geliştirir.	Madde16				✓
Fonksiyonları Uygulama	Öğretimi programlama dilinde parametre alan değer döndüren fonksiyon içeren programlar geliştirir.	Madde17			✓	
Koşullu İfadeler	Öğretimi programlama dilinde kontrol yapılarını kullanarak programlar geliştirir.	Madde18		✓		
Koşullu İfadeler	Öğretimi programlama dilinde kontrol yapılarını kullanarak programlar geliştirir.	Madde19		✓		
Koşullu İfadeler	Öğretimi programlama dilinde kontrol yapılarını kullanarak programlar geliştirir.	Madde20				✓
Koşullu İfadeler	Öğretimi programlama dilinde kontrol yapılarını kullanarak programlar geliştirir.	Madde21			✓	
Koşullu İfadeler	Öğretimi programlama dilinde kontrol yapılarını kullanarak programlar geliştirir.	Madde22				✓

Tablo 8 (Devamı)

Listeler ve Dizi Tipleri	Öğretimi gerçekleştirilecek programlama dilinde dizi ve listelere tipinde veri tanımlama işlemi yapar	Madde23	✓
Listeler ve Dizi Tipleri	Öğretimi gerçekleştirilecek programlama dilinde dizi ve listelere tipinde veri tanımlama işlemi yapar	Madde24	✓
Listeler ve Dizi Tipleri	Öğretimi gerçekleştirilecek programlama dilinde dizi ve listelere tipinde veri tanımlama işlemi yapar	Madde25	✓

8. Aşama: Uygulayıcılar için Test Puanlanma Rehberi. EK-1’de bazı örnek maddeleri sunulan birbirine paralel ve denk 25’er soruluk başarı testlerinden biri sanal robotik kodlama aracı olan VEXcode-VR ile diğeri ise fiziksel robotik kodlama aracı olan Raspberry-Pi ile Python programlama dili öğretimi amacıyla geliştirilmiştir. Son hali verilen başarı testleri, bu çalışmadaki ilgili gruplara ön test, son test olarak yapılmış ve tüm uygulama süreci bittikten 4 hafta sonra ise bilginin kalıcılık (hatırlanma) düzeyini belirlemek için kullanılmıştır.

Testlerdeki her sorunun 5 şıkkı ve 4 puanlık tek doğru cevabı vardır. Yanlış cevaplanan sorularda eksi puan alınmaz veya yanlış cevaplar doğru cevabı götürmez. Testten alınacak en yüksek puan 100’dür. 25 soruluk her test için uygulama süresi 1 ders saatidir (40 dk). Testler sanal veya fiziksel robotik kodlamaya yönelik Python programlama dili eğitimi veren BİLSEM eğitim kurumları dahil ortaokul ve liselerde ön test, son test olarak ve uygulamadan 4-5 hafta sonra bilginin kalıcılık (hatırlanma) düzeyini belirlemek üzere kullanılabilir.

Programlama Öz-Yeterlik Ölçeği

Programlamaya ilişkin öz-yeterlik ölçeği Kukul vd. (2017) tarafından ortaokul öğrencilerinin programlama öz-yeterliklerini belirlemek için geliştirilmiştir. EK-2’de bazı örnek maddeleri sunulan ve “Kesinlikle katılmıyorum”, “Katılmıyorum”, “Kararsızım”, “Katılıyorum”, “Kesinlikle katılıyorum” seçenekleri olan 5’li likert tipi ölçek 31 maddeden oluşmaktadır. Geçerlik ve güvenirlik analizleri için 233 ortaokul öğrencisine uygulanan tek faktörlü ölçeğin madde yükleri 0.47 ile 0.71 arasında değişmekte olup, varyans açıklama oranı %41.15 olarak bulunmuştur. İç tutarlılık analizinde, Cronbach alfa değeri 0.95 ve eşdeğer yarılar yöntemi sonucu 0.96 olarak belirlenmiştir. Yapı geçerliği için açımlayıcı ve doğrulayıcı faktör analizi uygulanan ölçeğin geçerli ve güvenilir bir veri toplama aracı olduğu ortaya konmuştur. Ölçek, bu çalışmada sanal ve fiziksel robotik kodlama etkinlikleri öncesinde ve sonrasında olmak üzere iki kez uygulanmıştır.

Programlama Kaygı Ölçeği

Cheung (1990) tarafından geliştirilen ve “Hiçbir zaman doğru değil”, “Nadiren doğru”, “Bazen doğru”, “Sıklıkla doğru”, “Her zaman doğru” şeklinde ifade edilen 5’li likert tipi programlama kaygı ölçeği 19 maddeden oluşmaktadır. Ölçeğin Demir (2015) tarafından Türkçe’ye

uyarlanmasında üçü İngilizce uzmanı, üçü Türkçe uzmanı görev almış, geçerlik ve güvenirlik analizleri yapılmıştır. 116 kişilik gruba uygulanan ölçeğin iç tutarlık analizinde, Cronbach alfa değeri 0.963 olarak belirlenmiştir. Geçerlik analizinde ise, Kaiser-Meyer-Olkin değeri 0.911, Bartlett'in Sphericity testi sonucu anlamlı bulunmuştur ($p < .001$). Ölçeğin bütün maddelerinin madde-toplam korelasyonları 0.48-0.89 aralığında değişiklik gösterdiği saptanmıştır. Buna göre, programlama kaygı ölçeği veri toplama aracında bulunan maddelerin güvenirlik kat sayılarının yüksek olduğu, anket sorularının tutarlı davranışları tespit etmeye yönelik olduğu ve ayırt edici bir özelliğe sahip olduğu görülmektedir. EK-3'te bazı örnek maddeleri sunulan ölçek, bu çalışmada sanal ve fiziksel robotik kodlama etkinlikleri öncesinde ve sonrasında olmak üzere iki kez uygulanmıştır.

Haftalık Robotik Kodlama Kazanım Değerlendirme Kontrol Listeleri

Öğrencilerin haftalık sanal ve fiziksel robotik kodlama etkinliklerindeki birbirine paralel ve denk kazanımları elde edip etmediklerini belirlemek için her etkinlik sonunda uygulanmak üzere araştırmacı tarafından “Evet” ve “Hayır” seçeneklerinden oluşan kazanım değerlendirme sorularını içeren 10 haftalık kontrol listeleri hazırlanmıştır. Haftalık kontrol listeleri VEXcode-VR sanal ve Raspberry-Pi fiziksel robotik kodlama gruplarına yönelik olacak şekilde ayrı ayrı oluşturulmuştur. Kontrol listelerindeki maddeler, bu çalışmadaki sanal ve fiziksel robotik kodlama başarı testlerinin “1.Aşama: Problemi Tanımlama, Amaç ve Kazanımları Belirleme” sürecinde temel alınan “Robotik Programlama Ortamını ve Programlama Araçlarını Tanıma, Değişkenler ve Sabitler, Döngüler ve Tekrarlı Yapılar, Fonksiyonları Uygulama, Koşullu İfadeler, Listeler ve Dizi Tipleri” ünite-konu başlıklarında yer alan kazanımlar (MEB, 2016, 2018, 2020, 2021) dikkate alınarak belirlenmiştir. Soruların soruluş biçimi veya içeriğinde benzeşen veya örtüşen türden verilerin bulunmamasına dikkat edilmiştir. Robotik kodlama eğitimi veren 5 BİLSEM uzmanı ve 1 öğretim teknolojileri uzmanı tarafından incelenen kazanım değerlendirme kontrol listelerine son hali verilmiştir. Bu ölçme aracının amacı öğrencilerin yanlış, eksik ya da öğrenilememiş kavramları ders sonrasında ortaya çıkarmaktır. Bu kontrol listeleri araştırmacıya, öğrencilerin eksik veya yanlış öğrenmelerini gözden geçirerek bir sonraki dersin ilk dakikalarında bilgiyi tekrar etme, düzeltme imkânı da sağlamıştır. Haftalık kontrol listelerinde yer alan bazı örnek maddeler EK-4'te sunulmuştur.

Haftalık Robotik Kodlama Meşguliyet Gözlem Formu

Sanal veya fiziksel robotik kodlama aracını kullanan gruptaki öğrencilerin, haftalık kazanımların temel alındığı problem durumunu anlatan senaryolara yönelik çözüm üretme sürecinde tamamlamaları gereken görevlerdeki duyuşsal, bilişsel, davranışsal, temsili ve sosyal

meşguliyetlerini gözlemlemek için yarı yapılandırılmış gözlem formu kullanılmıştır. Her iki robotik kodlama grubunda da haftalık etkinlikler sonunda kullanılan bu meşguliyet gözlem formu, robotik kodlamada öğrenci meşguliyetine ilişkin alan yazındaki çalışmalar (Fredricks vd., 2005; Reeve, 2012) ve tez danışmanının bu alandaki deneyimine dayalı olarak oluşturulmuştur. “Gözlendi”, “Kısmen Gözlendi”, “Gözlenmedi” şeklindeki değerlendirme ile öğrencilerin görevi tamamlama durumu belirlenmiştir. Meşguliyetin her bir alt boyutundaki gözlem maddelerinin robotik kodlama meşguliyeti ile ilgili olup olmadığı, öğretim teknolojileri alanındaki robotik kodlama uzmanı bir öğretim üyesine incelenmiştir. Haftalık gözlem formunda yer alan bazı örnek maddeler EK-5’te sunulmuştur.

Görüşme Rehberi

Sanal veya fiziksel robotik kodlama etkinliklerine katılan öğrencilerin öğrenme süreci boyunca yaşadıkları olumlu ve olumsuz deneyimleri ortaya çıkarmak için tez danışmanı ile araştırmacı tarafından hazırlanan yarı yapılandırılmış görüşme rehberi EK-6’da sunulmuştur. Görüşme rehberinde, her iki gruptaki öğrencilerin 10 haftalık etkinliklerdeki kazanımları edinmek için sergiledikleri performans ve meşguliyetin yanı sıra, Python programlama dili öz-yeterliklerine ve programlama kaygılarına yönelik görüşlerini detaylı olarak incelemeye yönelik sorular yer almıştır. Araştırma amacına uygunluğu, anlaşılabilirliği ve dil denetimi için sorular; bilişim teknolojileri öğretmeni, öğretim teknolojileri alanındaki öğretim üyesi ile Türk dili ve edebiyatı öğretmenine incelenmiştir. Ayrıca örneklem dışındaki 1 erkek ve 1 kız öğrenciye sorular okutularak gerekli düzeltmeler yapılmıştır. Uygulama sürecinin tamamlanmasının ardından sanal ve fiziksel robotik kodlama gruplarındaki gönüllü kız ve erkek öğrencilerle birebir ve yüz yüze görüşmeler gerçekleştirilmiştir. Öğrencilerin sorulara daha samimi cevaplar verebilmeleri için görüşmeler ders saatleri dışında yapılmış, katılımcıların da izni ile ses kaydı da alınmıştır.

Uygulama Süreci

Uygulama, 2024-2025 eğitim-öğretim yılı güz döneminde Bitlis ili Tatvan ilçesindeki Bilim ve Sanat Merkezine yönelik Bireysel Yetenekleri Fark Ettirme (BYF) öğretim programında yer alan Robotik Kodlama Atölye dersinde gerçekleştirilmiştir. Bu dersteki sanal VEXcode-VR veya fiziksel Raspberry-Pi robotik kodlama araçlarının kullanıldığı iki farklı gruptan her birine rastgele atanan üstün yetenekli öğrencilerden oluşan 6. ve 7. sınıflar ile Python programlama dili öğrenmeye yönelik etkinlikler yapılmıştır. Hedef kitlenin yer aldığı BİLSEM kurumunda uygulamanın resmi olarak yapılabilmesi için MEB’den alınan araştırma ve uygulama izin belgeleri EK-10’da sunulmuştur. Devam eden bölümde, dersin öğretmeni olan araştırmacı tarafından bizzat yürütülen uygulama öncesi, sırası ve sonrasındaki tüm süreç detaylı olarak açıklanmıştır.

Uygulama Öncesi

Uygulama öncesinde sanal VEXcode-VR ve fiziksel Raspberry-Pi robotik kodlama araçlarının kullanıldığı iki farklı gruba üstün yetenekli öğrencilerden oluşan 6. ve 7. sınıflar rastgele atanmıştır. Sanal ve fiziksel robotik kodlama grupları için Python programlama dili öğretimine yönelik 10 haftalık etkinlik planları hazırlanmıştır. Bu etkinlik planlarında hem BİLSEM Robotik Kodlama Atölye dersi kazanımları, hem Bilişim Teknolojileri ve Yazılım dersi problem çözme ve programlama kazanımları, hem de MEB Bilgisayar Bilimi dersi Python programlama dili öğretim programındaki kazanımlar temel alınmıştır. Bu kapsamda sanal VEXcode-VR ve fiziksel Raspberry-Pi robotik kodlama araçları kullanılarak öğretilen Python programlama dili için haftalık hedef kazanımları ve göstergeleri oluşturulmuştur. Böylece her hafta için birbirine paralel kazanımları olan ve içerik olarak birbirine denk olan iki ayrı etkinlik planı hazırlanmıştır. Her iki grup için ayrı ayrı hazırlanan 10'ar haftalık etkinlik planları EK-7 (Fiziksel)'da ve EK-8 (Sanal) sunulmuştur. Öz-yeterlik ve kaygı değişkenleri için geçerliği ve güvenirliği test edilmiş uygun veri toplama araçları seçilmiştir. Görüşme rehberinin yanı sıra, sanal veya fiziksel robotik kodlama ile Python programlama dili öğrenme sürecine yönelik olarak haftalık kazanım değerlendirme kontrol listeleri ve meşguliyet gözlem formları oluşturulmuştur. Ayrıca 10 haftalık pilot uygulama ile her iki grup için geliştirilen başarı testlerinin geçerlik-güvenirlik analizleri yapılmıştır.

Uygulamanın yapıldığı atölyenin teknik alt yapısı kontrol edilerek etkinliklerde aksaklık yaşanmaması için gerekli önlemler alınmıştır. Bu kapsamda, bilgisayar ve internet altyapısı güncellenmiş, her öğrencinin bireysel olarak kullanabileceği sayıda sanal VEXcode-VR ve fiziksel Raspberry-Pi robotik kodlama araçları temin edilmiştir. 2 haftalık ders süreci boyunca, sanal ve fiziksel robotik kodlama gruplarına uygulama sürecinin nasıl ilerleyeceği anlatılmış, öğretim süresince kullanılacak olan robotik kodlama araçları tanıtılmıştır. Uygulamanın hemen öncesindeki 1 haftalık süreçte, her iki gruba da başarı testi ile öz-yeterlik ve kaygı ölçekleri ön test olarak uygulanmıştır.

Tatvan Bilim ve Sanat merkezi robotik kodlama atölyesi teknik altyapısı robotik kodlama öğretim aracı kullanımı için uygun hale getirildiği için uygulama takvimi robotik kodlama atölyesi ders programına göre düzenlenmiştir. 10 haftalık etkinlik sırasına göre her iki grup için ortak olan robotik kodlama konuları ve kazanımlarının yanı sıra, haftalık etkinliklerde sanal ve fiziksel robotik kodlama gruplarına özgü problem durumlarının yer aldığı uygulama takvimi Tablo 9'da özetlenmiştir.

Tablo 9. 10 Haftalık Uygulama Takvimine Göre Sanal ve Fiziksel Robotik Kodlama Problem Durumları, Ortak Konular ve Kazanımlar

Her İki Grup İçin Ortak				Sanal Robotik Kodlama (VEXcode-VR)	Fiziksel Robotik Kodlama (Raspberry-Pi)
Ay	Hafta	Konu	Kazanım	Etkinlikteki Problem Durumu	Etkinlikteki Problem Durumu
Ekim	1.hafta	Kod editör ekranı ve robotik araç tanıtımı	Kodlama yapacağı sanal veya fiziksel robotik aracı tanır. Kod editöründe temel işlevleri yapar.	▪ VEXcode-VR sanal 3B robotik araç bileşenlerini ve çevrimiçi kod editör arayüzünü tanıma	▪ Raspberry-Pi fiziksel robotik araç bileşenlerini ve bilgisayardaki kod editör arayüzünü tanıma
	2.hafta	Kod editör ekranı ve robotik araç tanıtımı	Kodlama yapacağı sanal veya fiziksel robotik aracı tanır. Kod editöründe temel işlevleri yapar.	▪ VEXcode-VR sanal robotik aracının sensor ve motor bağlantılarını tanıma ▪ Kodlama amacına uygun sanal robotu seçme ▪ Kod blokları ve Python programlama dilini karşılaştırma ▪ Temel hareket ve yönlendirme komutlarını kullanma (sağ, sol, ileri, geri)	▪ Raspberry-Pi fiziksel robotik aracının sensor ve motor bağlantılarını tanıma ▪ GPIO (genel amaçlı giriş çıkış) pinlerinin kullanımını ve kart üzerindeki yerlerini gösterme ▪ Python programlama dilini nasıl kullanacağını gösterme ▪ Raspberry-Pi robotik kartının bilgisayarda kod editör programına bağlantısını yapma
	3.hafta	Değişkenler Sabit Değerler Karışık Türlü İfadeler	Öğretimi gerçekleştirilecek programlama dilinde değişkenleri ve sabitleri uygun şekilde kullanarak programlar geliştirir.	▪ VEXcode-VR sanal robotik aracının hareketini tasarlayarak belirli sayıda manevra yaptıрма ▪ VEXcode-VR değişkenini tanımlama kuralını bilme ▪ VEXcode-VR robotuna eşkenar çokgenler çizdirme işlemini değiken tanımlayarak yapma ▪ Robotu 12 kez eşit mesafelerde manevralar yaptırarak 12 kenarlı eşkenar bir çokgen çizme ▪ VEXcode-VR robotunun kalem özelliğini kullanma	▪ Birden fazla Ledin süreli yanıp sönmesi konusunda gözümüzü açıp kapama örneği verme ▪ Her bir ledin açık ya da kapalı kalma süresi ve bu sürenin ayarlanması için kodlar kullanma ▪ Trafik ışıkları örneği verme ve benzer bir uygulama gerçekleştirme ▪ LED bağlantısının nasıl yapılması gerektiğini anlatma ▪ Her bir LED için tek tek değişken tanımlama ve değişken tiplerini açıklama ▪ Kullanılan led sayısı kadar değişken tanımlanarak ledlerin kısa süreli yanıp sönmelerini sağlayan komutlarını veme, Python kodlarını yazma

Tablo 9 (Devamı)

Kasım	4.hafta	Döngüler While döngüsü Belirli ve belirsiz döngüler Döngüden çıkma komutları	Belirlenen programlama dilinde tekrarlı yapıları kullanarak programlar geliştirir.	▪ VEXcode-VR robotunun tampon sensörünün çalışma prensibini anlama ▪ Sensörün True/False sinyalleri gönderdiğini bilme ve Boolean değişkenlerini kullanma ▪ While not Komutu ile sensör tetiklendiğinde robotun hareketini kodlama ▪ left_bumper.pressed ve right_bumper.pressed komutları ile robotu sağa ve sola yönlendirme	▪ Televizyon, Çamaşır Makinesi vb. elektronik cihazlarda LED diyotun bulunduğunu farketme ▪ Işıklı uyarı sistemlerinde kullanılabileceği durumlara örnekler verme ▪ Etkinlikte led diyotu sürekli ve belirli aralıklarda yakıp söndürerek while döngüsünü uygulama ▪ Led Diyot ve direnç bağlantı elemanları devre kurulumu yapma ▪ While True komutunu kullanma
	5.hafta	Döngüler Döngü yapıları For döngüsü İç içe döngüler	Öğretimi gerçekleştirilecek programlama dilinde tekrarlı yapıları kullanarak programlar geliştirir.	▪ VEXcode-VR for döngüsünü nasıl yazması gerektiğini öğrenme ▪ Robotu belirli sayılarda hareket ettirerek hedef noktaya doğru bir şekilde park etme (döngü komutları ile kare çizerek) ▪ Robot hareketlerini robotun kalem işlevini aktifleştirerek robotun gittiği yolu çizdirme ve bunun için kalem komutunu kullanma ▪ For value in range() ve pen komutunu kullanma ▪ Wait() komutu ile robotu bekletme	▪ Işıklı uyarı sistemlerini tanıma ▪ Işık kaynağının istenilen sayıda yanıp sönmeye gerektiğinde nasıl bir kodlama yapılabileceğini anlama ▪ Döngü kullanmadan led ışık kaynağını beş kez yakıp söndürme işlemi yapma ▪ Aynı işlemi For döngüsü kullanarak yapma for i in range(0,50): ▪ Bu sayede ne kadar az kod yazdığını fark etme
	6.hafta	Fonksiyonlar Global değişkenler Fonksiyon sıralaması belirleme Fonksiyon örnekleri	Öğretimi gerçekleştirilecek programlama dilinde parametre alan değer döndüren fonksiyon içeren programlar geliştirir.	▪ VEXcode-VR robotunun daima yapması gereken birtakım hareketlerin olduğu ve bunlar için program içerisinde sürekli kod tekrarlarının yapılmasına gerek olmadığı, bunun için fonksiyon oluşturulabileceğini farketme ▪ Örnek olarak sağa ve sola dönme fonksiyonlarını anlama ▪ Robotun sağa ve sola 90 derece dönüşler yaparak belirli bir hedefe park etmesi görevini yerine getirme	▪ Robotların hareketini sağlayan elektronik devre elemanı, elektrik motorları ile ilgili bilgilendirme yapma ▪ Robota hareket kabiliyeti sağlayan en önemli motorların sağa sola ileri geri hareketlerinin programlama ile kolay bir şekilde yapılabileceğini anlama ▪ Servo motorun Raspberry-Pi ye bağlantısını yapma ve servo motora farklı hareketler yaptırmak için her bir harekete özgü farklı bir fonksiyon oluşturulabileceğini anlama ▪ servo motora açı yaptıracak hareket için def

Tablo 9 (Devamı)

7.Hafta	Koşul İfadeleri İlişkisel Operatörler İf ifadesi İf/Else ifadesi İç içe koşul yapıları	Öğretimi gerçekleştirilecek programlama dilinde kontrol yapılarını kullanarak programlar geliştirir.	▪ def onevent_saga_donus_0(): def onevent_sola_donus_0(): fonksiyon komutlarını kullanarak Python ile kodlamasını yapma ▪ VEXcode-VR robotun renk sensörünü kullanarak robotu disk labirentinde doğru yolda hedefe ulaştırma ▪ VEXcode-VR disk labirenti haritasında farklı renkteki diskleri tanıyarak robotu ilerletmek için göz-renk sensörünü kullanma ▪ Bunu yapmak için göz-renk sensörüne uygun koşul komutlarını kullanarak robotu kodlama ▪ Kodlamalarda If front_eye.detecs(GREEN): komutunu ile robotun renk algılayıcı sensörünü kullanma	aciAyarla(aci): x=(1/180)*aci + 1 duty=x*5 pwm.ChangeDutyCycle(duty) fonksiyon komutlarını kullanarak Python ile kodlamasını yapma ▪ Robotları dışarıdan müdahale ile nasıl kontrol edileceğini anlama, böylece herhangi bir sistemin nasıl açılıp kapatılacağını farketme ▪ Robotik devreye dışarıdan bir müdahale ile led ışığı açıp kapatma işlemi için buton devre elemanının devreye bağlama ▪ Butona basıldığında ledin yanması tekrar basıldığında led ışığın sönmesini sağlayacak koşullu bir program yazma ▪ Elektronik devreye buton bağlama ▪ Bu buton sistemine özgü if buton_durum == 0: GPIO.output(LED,GPIO.HIGH) else: GPIO.output(LED,GPIO.LOW) Python kodlarını kullanma
8.Hafta	Listeler Liste kavramı Değerler arası gezinme Liste üyeliği	Öğretimi gerçekleştirilecek programlama dilinde dizi tipinde veri tanımlar.	▪ Akıllı cihazların otonom bir şekilde kullanımını anlama ▪ VEXcode-VR robotunun farklı hızlarda belirli bir mesafede ilerlemesi için neler yapılması gerektiğini tartışma ▪ Robotun 10 farklı hızda belirli bir hedefe doğru ilerletilmesini sağlama ▪ hiz_listesi = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100] komutu kullanılarak robotun ilk hızdan son hıza kadar takip edilerek hızlanmasını izleme	▪ Farklı renkteki ledleri Raspberry-Pi nin beş farklı GPIO pinine bağlayarak GPIO pinlerini tek tek kodlamak yerine her led 1 saniye yanacak şekilde Python dilinde Liste (Dizi) komutlarını kullanarak kodlama ▪ pinler=[40,38,36,37,35] komutunu kullanarak 5 farklı ledi Liste içinde ve farklı zamanlamalarda açıp kapatacak kodları yazma

Tablo 9 (Devamı)

Aralık	9.Hafta	Proje Gerçekleştirme	Önceki haftalarda uygulanan programlama yapılarının tümünü kullanarak proje hazırlama	▪ VEXcode-VR robotik ortamında farklı amaçlar için kullanılan haritaların olduğunu, her haritanın farklı algoritmalar ile kodlanabileceğini farketme ▪ Duvar labirent projesindeki görevleri tamamlamak için uygun komutları içeren Python kodlarını yazma ▪ Bunun için Duvar labirent alanında robotun farklı dört istasyona uğrayarak istasyonlarda bekleme yapmadan son istasyona ulaşmasını sağlayan kodları yazma ▪ Bu noktada her istasyona gitmek için daha önceki derslerde öğrenilen farklı algoritmalar ve kodlamalar yapma ve robotun uygun sensörlerini kullanma	▪ Raspberry-Pi ile akıllı ev sistemi kurma projesindeki görevleri tamamlamak için uygun komutları içeren Python kodlarını yazma ▪ Bu noktada kurmak için daha önceki derslerde öğrenilen farklı algoritmalar ve kodlamalar yapma ve robotun uygun sensörlerini kullanma (en az bir adet sensör ve bir adet buzzer kullanma) ▪ Tüm sistemi bir adet Raspberry-Pi robotik kartı üzerinde Python programlama dili ile kodlama
	10.Hafta	Proje Gerçekleştirme	Önceki haftalarda uygulanan programlama yapılarının tümünü kullanarak proje hazırlama	▪ Duvar labirent projesindeki robotta bulunan tampon sensörü ilk istasyon için kullanılacak sensör olarak belirleme ▪ Robotun her istasyonda 3 saniye beklemesini sağlayan kodu yazma ▪ Robotu farklı hızlarda kullanarak son istasyona en hızlı şekilde ulaştıracak kodları yazma ▪ Son istasyonda robotun kırmızı-siyah damalı zemini görünce durması için renk sensörünü kullanma	▪ Tüm sistemi bir adet Raspberry-Pi robotik kartı üzerinde Python programlama dili ile kodlama ▪ Kurulacak olan akıllı ev sisteminde en az 4 adet led kullanma ▪ Kurulacak olan akıllı ev sisteminde en az 1 adet servo motor kullanma ▪ Tüm sistemi bir adet Raspberry-Pi robotik kartı üzerinde Python programlama dili ile kodlama

Uygulama Sırası

Sanal ve fiziksel robotik kodlama gruplarına uygulanan haftalık etkinlikler, ders programına göre haftanın 1 günü (günde 2 ders saati) olmak üzere 10 hafta devam etmiş ve uygulama süreci her bir grup için toplam 20 saatte tamamlanmıştır. Bu kapsamda sanal robotik kodlama grubundaki öğrenciler bilgisayar üzerinde çevrimiçi çalışan VEXcode-VR simülasyon robot temelli 3B ortamını, fiziksel robotik kodlama grubundaki öğrenciler ise bilgisayar ile bluetooth bağlantı kurarak çalışan Raspberry-Pi setlerini kullanmışlardır.

Etkinlik saatlerinde ilk önce öğretmen tarafından haftanın uygulaması hakkında temel bilgiler verilmiş; daha sonra gruplara göre öğrencilerin, sanal veya fiziksel robotik kodlama etkinliklerindeki haftalık kazanımları dikkate alan problem durumuna ilişkin görevleri yerine getirirken, Python programlama dilini temel alarak problemin çözümünü kodlamaları beklenmiştir. Her gruptaki öğrenciler kendi kodlama etkinliklerini ilgili robotik kodlama öğretim aracını kullanarak bireysel olarak tamamlamışlardır. Uygulama için alınan MEB resmi araştırma iznine uygun olarak veri analizinde kullanılmak üzere, haftalık etkinliklerde öğrenci yüzleri gizlenip yalnızca uygulamaya yönelik fotoğraflar çekilmiş, ekran görüntüleri ve video kayıtları alınmıştır. Sanal ve fiziksel robotik kodlama gruplarına ait uygulama görüntülerinden örnek fotoğraflar EK-9’da gösterilmiştir.

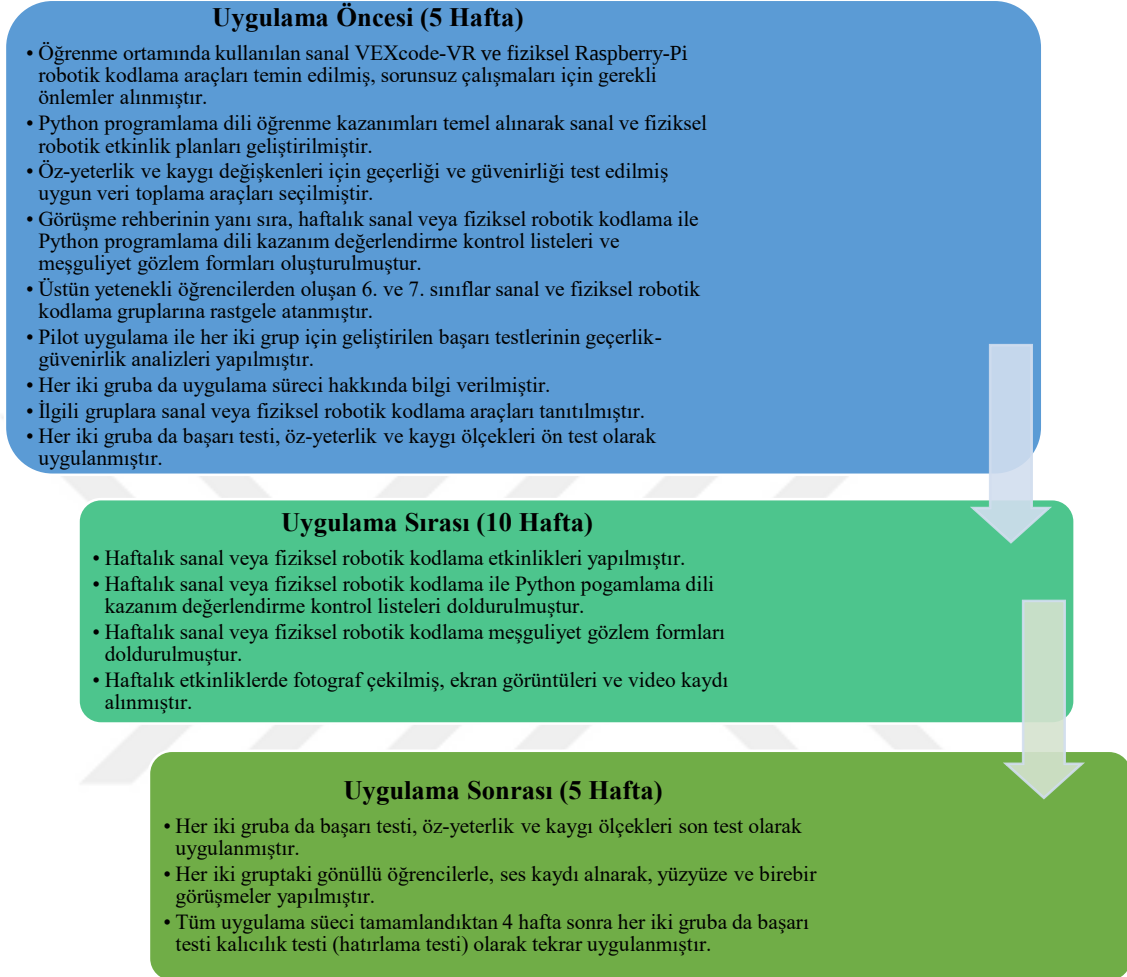
Uygulama süresince bilgisayarlarda oluşabilecek aksaklıklara ve robotik kodlama araçlarında oluşabilecek arızalara karşı yedek bilgisayarlar ve robotik araçları bulundurulmuştur. Her iki grupta da dersler aynı zamanlarda ve ders saatlerinde, aynı konu ve kazanımlar gözetilerek işlenmiştir. Ayrıca her haftaki etkinlik sonunda (2. ders saatinin sonunda), ders öğretmeni olan araştırmacı tarafından, gruptaki her bir öğrenci için ayrı ayrı “*Robotik Kodlama Kazanım Değerlendirme Kontrol Listesi*” ve “*Robotik Kodlama Meşguliyet Gözlem Formu*” doldurulmuştur.

Uygulama Sonrası

Tüm uygulama sürecinin bittiği 10. haftanın sonunda her iki gruba da başarı testi ile öz-yeterlik ve kaygı ölçekleri son test olarak uygulanmıştır. Sanal ve fiziksel robotik kodlama etkinliklerine katılan üstün yetenekli öğrenciler arasından gönüllü olanlarla “*Görüşme Rehberi*” temel alınarak yüz yüze ve birebir görüşmeler yapılmıştır. Bu görüşmelerle, her iki gruptaki öğrencilerin 10 haftalık etkinliklerdeki robotik kodlama kazanımlarını edinme sürecine yönelik deneyimleri, sergiledikleri performans ve meşguliyetin yanı sıra, Python programlama dili öz-yeterliklerine ve programlama kaygılarına yönelik görüşleri de detaylı olarak sorgulanmıştır. Görüşmeler sırasında veri kaybını önlemek için ses kaydı alınmıştır.

Ayrıca son testler uygulandıktan 4 hafta sonra, bilginin kalıcılık (hatırlanma) düzeyini belirlemeye yönelik olarak her iki gruptaki öğrencilere başarı testi bir kez daha uygulanmıştır. Tüm uygulama sürecinde yapılanlar Şekil 9’da özetlenmiştir.

Şekil 9. Uygulama Süreci



Veri Analizi

Araştırma süreci boyunca araştırmacı tarafından toplanan nicel ve nitel verilerin analizinde farklı veri analiz teknikleri kullanılmıştır. Yapılan nicel ve nitel analizler öğretim teknolojileri ve robotik kodlama alanında uzman bir öğretim üyesi tarafından kontrol edilmiştir. Ortaya çıkan sonuçlar, araştırma sorularına göre bulgular bölümünde tablo ve grafiklerle desteklenerek raporlaştırılmış, kritik noktaları ayrıntılı olarak yorumlanmıştır. Araştırma sorularına ve veri türüne göre, kullanılan veri toplama araçları ve veri analiz teknikleri Tablo 10’da özetlenmiştir.

Tablo 10. Araştırma Sorularına Göre Kullanılan Veri Toplama Araçları ve Analiz Teknikleri

Araştırma Soruları	Veri Toplama Araçları	Veri Analiz Teknikleri
1a	Kazanım Kontrol Listeleri	Betimsel analiz/ Ortalama, Standart Sapma Kestirimsel Analiz/ Gruplar Arası Mann-Whitney U Testi
1b	Meşguliyet Gözlem Formları	Betimsel analiz/ Ortalama, Standart Sapma Kestirimsel Analiz/ Gruplar Arası Mann-Whitney U Testi
1c	Programlama Başarı Testi	Betimsel Analiz/ Ortalama, Standart Sapma Kestirimsel Analiz/ Grup İçi Wilcoxon İşaretli Sıralar Testi Kestirimsel Analiz/ Gruplar Arası Mann-Whitney U Testi
1d	Programlama Kalıcılık Testi	Betimsel Analiz/ Ortalama, Standart Sapma Kestirimsel Analiz/ Grup İçi Wilcoxon İşaretli Sıralar Testi Kestirimsel Analiz/ Gruplar Arası Mann-Whitney U Testi
2	Programlama Öz-Yeterlik Ölçeği	Betimsel Analiz/ Ortalama, Standart Sapma Kestirimsel Analiz/ Grup İçi Wilcoxon İşaretli Sıralar Testi Kestirimsel Analiz/ Gruplar Arası Mann-Whitney U Testi
3	Programlama Kaygı Ölçeği	Betimsel Analiz/ Ortalama, Standart Sapma Kestirimsel Analiz/ Grup İçi Wilcoxon İşaretli Sıralar Testi Kestirimsel Analiz/ Gruplar Arası Mann-Whitney U Testi
4	Programlama Başarı Testi Programlama Öz-Yeterlik Ölçeği Programlama Kaygı Ölçeği	Korelasyon Analizi/ Spearman Sıra Farkları Korelasyon Testi
5	Görüşme Rehberi	İçerik Analizi/ Tema ve Kodların Frekansları, Gruplardaki Öğrenci İfadelerinden Alıntılar

Nicel Veri Analizi

Nicel veriler SPSS 20 programına aktarılarak istatistiksel test koşulları incelenmiştir. Bu bağlamda öncelikle gruplardaki her bir öğrenciye ID numarası verilerek, her bir değişkenin aynı öğrenciye ait olan ön test ve son test ve kalıcılık testi verileri eşleştirilmiştir. Bu verilerin betimsel özelliklerini ortaya koyan ortalama, standart sapma gibi istatistiki hesaplamalar yapılmış, normallik ve homojenlik testleri uygulanmıştır. Ancak bu çalışmada, her iki gruptaki öğrenci sayısı az olduğu için her bir değişken açısından grupların ön test, son test ve kalıcılık testine ait veri setleri normallik varsayımlarını sağlayamamıştır (Büyüköztürk vd., 2023; Çelik

vd., 2020). Bu nedenle parametrik olmayan istatistiksel testlerden grup içi karşılaştırmalar için Wilcoxon İşaretli Sıralar Testi, gruplar arası karşılaştırmalar için Mann-Whitney U testi yapılmıştır. Değişkenler arası ilişkileri belirlemek için ise Spearman Sıra Farkları Korelasyon Testi kullanılmıştır (Field, 2013).

Nitel Veri Analizi

Sanal ve fiziksel robotik kodlama gruplarındaki gönüllü öğrencilerle yapılan yüz yüze ve birebir görüşmelerin ses kayıtlarına ID numarası verilmiştir. Bu ses kayıtları metne dönüştürüldükten sonra QDA Miner Lite programına aktararak içerik analizi yapılmıştır. İçerik analizi; ortaya çıkan tema ve kodların frekanslarına göre bu araştırmanın değişkenleri çerçevesinde birbiriyle ilişkisini detaylı inceleyip bir bütün olarak ele almayı ve yorumlamayı kolaylaştırmıştır (Fraenkel vd., 2012; McMillan & Schummacher, 2010). Bu kapsamda ortaya çıkan tema ve kodları desteklemek için her iki gruptaki öğrenci ifadelerinden alıntılar da yapılmıştır (Şimşek & Yıldırım, 2021).

Araştırmacının Rolü

Bu çalışmada, sanal ve fiziksel robotik kodlama gruplarındaki etkinlikleri yürüten ve aynı zamanda bu dersin öğretmeni olan araştırmacı, 20 haftalık tüm araştırma sürecinin sağlıklı ve hatasız koordine edilmesi açısından uygulama öncesinde, esnasında ve sonrasında çeşitli görevler üstlenmiştir. Uygulama başlamadan önce; öğretimde kullanılan materyalleri, sanal ve fiziksel robotik kodlama araçlarını ve etkinlik planlarını hazırlamış, hazır veri toplama araçlarını elde etmiş, kullanılan diğer veri toplama araçlarını geliştirmiş, pilot uygulamaları yapmıştır. Uygulama esnasında; dersin öğretmeni olarak sürecin ve robotik kodlama öğretim araçlarının tanıtımını yapmış, her iki gruptaki öğretim sürecini 10 hafta boyunca bizzat kendisi yönetmiştir. Uygulama yapıldıktan sonra veri toplama sürecini tamamlamış, nicel ve nitel veri analizlerini gerçekleştirmiş ve ortaya çıkan sonuçları araştırma soruları çerçevesinde raporlaştırmıştır. Süreçte karşılaşılan küçük sorunlar araştırmacı tarafından çözülmüştür. Araştırmacı, etkinliklerde bulunan içeriklerin örneklem gruplarına aktarılırken, her iki grup için de aynı süre verilmesine dikkat etmiştir. Sanal ve fiziksel robotik kodlama etkinliklerinin öğrencilerin sınıf düzeyine ve bilgi-beceri-yeteneklerine uygun olarak gerçekleştirmelerine, öğretimin Python programlama dili çerçevesinde yapılmasına özen göstermiştir. Öğretim teknolojileri ve robotik kodlama alanında uzman+ olan danışman ise; tüm sürecin etik kurallara uygun şekilde ilerlemesi, gerekli izinlerin alınması için yürütücü görevi üstlenmiş, araştırmacıyla koordinasyon içinde çalışarak, uygulama öncesi, esnası ve sonrasındaki her bir adımını takip etmiş ve süreç boyunca uzman olarak görüşünü de iletmiştir.

Çalışmanın Geçerlik ve Güvenirliği

Yapılan araştırmanın literatüre katkı sunması ve doğru sonuçlar vermesi için çalışmanın geçerli ve güvenilir olması önemli bir unsurdur. Bu nedenle çalışmanın kalitesinin artırılması için birtakım önlemler alınmasının gerekliliği belirtilmektedir (Büyüköztürk vd., 2023; McMillan & Schummacher, 2010; Şencan, 2005). Karma araştırma desenlerinden Çeşitlemenin temel alındığı bu çalışmada hem nicel hem nitel açıdan geçerlik ve güvenilirlik önlemleri birlikte kullanılmıştır. Bu önlemler sayesinde çalışmanın inandırıcılığı, genellenebilirliği, tutarlılığı artırılmaya çalışılmıştır (Fraenkel vd., 2012).

Araştırmanın iç güvenirliliğini artırabilmek için literatür taraması yapılmıştır. Dış güvenirliliği artırmak için ise veri toplama araçları, elde edilen veriler ve veri analizleri öğretim teknolojileri alanında uzman ve çalışmanın danışmanlığını yürüten öğretim üyesi tarafından takip edilmiş ve incelenmiştir. Çalışmanın iç geçerliliğini artırmak için; farklı veri toplama araçları kullanılmış, geliştirilen veri toplama araçlarına alan yazın taraması yapıldıktan ve uzman görüşü alındıktan sonra son hali verilmiş, pilot uygulamaları yapılmış, geçerlik ve güvenirliliği test edilmiş hazır veri toplama araçları da kullanılmıştır. Ayrıca nitel veri toplama sürecinde ses ve video kaydı yapılarak veri kaybının önüne geçilmiştir. Veri analizi sürecinde uzman görüşüne başvurulmuştur. Çalışmanın dış geçerliliğini sağlamak için; her iki grup için ders planı ve öğretimde kullanılacak etkinlik planları ayrıntı bir şekilde geliştirilmiş, nicel ve nitel veri analizinde yapılanlar detaylandırılmış, veri toplama araçlarının geçerlik ve güvenirlilikleri hesaplanmıştır. Toplanan verilerin sentezi araştırmacı, akranlar ve danışman öğretim üyesi tarafından kontrol edilmiştir.

Çalışmanın geçerliği ve güvenirliliğinin yüksek olması için veri toplama araçlarına çeşitli istatistiksel metotlar ile testler uygulanmış, maddelerin değişkenleri temsil etme miktarı ve hata ortaya çıkma ihtimali en düşük seviyeye indirilmeye çalışılmıştır. Çalışmada, kullanılan hazır veri toplama araçlarının geçerlik ve güvenirliliği testinden geçmiş araçlar olmasına özen gösterilmiştir. Araştırmacı tarafından geliştirilen veri toplama araçlarının ise kapsam geçerliliğini ortaya koymak için, ilgili literatür incelenmiş, 5 Bilişim Teknolojileri öğretmeni ve 1 Öğretim Teknolojileri uzmanından görüşler alınmış, dönütler doğrultusunda gerekli düzeltmeler yapılmıştır. Veri toplama araçlarının güvenirliliğini belirlemek için ise başarı testlerinin Kuder-Richardson (KR-20) değeri, öz-yeterlik ve kaygı ölçeklerinin Cronbach Alpha güvenirlilik katsayıları hesaplanmıştır. Çalışmada alınan geçerlik ve güvenirlilik önlemleri Tablo 11’de gösterilmiştir.

Tablo 11. Çalışmada Alınan Geçerlik ve Güvenirlik Önlemleri

Strateji	Önlem
<i>Yöntemi Detaylandırma</i>	Araştırma sorularının tüm boyutları ile cevaplanabilmesi için seçilen karma araştırma yönteminin nasıl uygulandığı detaylı olarak açıklanmıştır.
<i>Örneklemi Seçme</i>	Çalışma amacına uygun örneklemin nasıl seçildiği, gruplara nasıl atandığı, gruplardaki öğrencilerin sayısı ve demografik özellikleri detaylı açıklanmıştır.
<i>Öğretim Araçlarını Seçme</i>	Çalışmada hangi sanal ve fiziksel robotik kodlama araçlarının niçin kullanılacağına öğretim teknolojileri ve robotik kodlama alanında uzman olan danışman ile birlikte karar verilmiştir.
<i>Veri Çeşitliliği Sağlama</i>	Değişkenlere ilişkin nicel ve nitel veri toplama araçları kullanılarak veri çeşitliliği sağlanmıştır.
<i>Uygun Veri Toplama Araçlarını Seçme</i>	Hazır veri toplama araçları seçilirken literatür taraması yapılarak ve uzman görüşü alınarak karar verilmiş, madde sayısına dikkat edilmiş, öğrenci seviyesine ve araştırma kapsamına uygunluğu kontrol edilmiştir.
<i>Veri Toplama Araçlarını Geliştirme ve Geçerlik Analizleri</i>	Sanal ve fiziksel robotik kodlama için geliştirilen iki ayrı başarı testinin birbirine paralel ve denk olup olmadığı, BT öğretmeni olan 2 yüksek lisans öğrencisi, 5 BİLSEM uzmanı ve 1 öğretim teknolojileri uzmanı tarafından kontrol edilmiş, soruların kapsam geçerliği hesaplanmış ve madde analizleri yapılmıştır.
<i>Uzman Görüşü</i>	Sanal ve fiziksel kodlama grupları için hazırlanan haftalık etkinlik planları, kazanım kontrol listeleri, meşguliyet gözlem formları ve görüşme rehberi BT öğretmeni olan 2 yüksek lisans öğrencisi ve 1 öğretim teknolojileri uzmanı tarafından kontrol edilmiştir.
<i>Akan Değerlendirme</i>	Başarı testlerindeki ve görüşme rehberindeki soruların anlaşılabilirliğini test etmek için örnekleme aynı sınıf düzeyinde ve örneklem dışındaki 1 kız 1 erkek öğrenciye sorular okutulmuştur.
<i>Veri Toplama Planı</i>	Hangi veri toplama araçlarının ne zaman ve nasıl kullanılacağı açıklanmıştır.
<i>Pilot Uygulama</i>	Sanal ve fiziksel robotik kodlama için geliştirilen iki ayrı başarı testinin uygunluğunu test etmek için pilot uygulama I ve II yapılmıştır.
<i>Asıl Uygulama</i>	Asıl uygulama öncesi, esnası ve sonrasında yapılanlar adım adım açıklanmıştır.
<i>Araştırmacı Rolü</i>	Araştırmacının çalışma sürecindeki yetkileri ve sınırları belirtilmiştir.
<i>Sınırlılıklar ve Varsayımlar</i>	Çalışmadaki sınırlılıklar ve temel varsayımlar madde madde sıralanmıştır.
<i>Grupların Denkliği</i>	Örneklemdaki üstün yetenekli öğrenci gruplarının birbirine denk olup olmadığı kontrol edilmiştir.
<i>Etkinlikleri Planlama ve Uygulama</i>	Sanal ve fiziksel robotik kodlama grupları için hazırlanan haftalık etkinlik planlarının kazanımlara ve örneklem seviyesine uygunluğu kontrol edilmiş, gruplara ne zaman ve nasıl uygulanacağı planlanmış ve 10 haftalık uzun sürece yayılarak haftalık olarak uygulanmıştır.
<i>Güvenirlik Analizleri</i>	Veri toplama araçlarının güvenilirliğini belirlerken, iki ayrı başarı testi için Kuder-Richardson (KR-20), öz-yeterlik ve kaygı ölçekleri için ise Cronbach Alpha değerleri hesaplanmıştır.
<i>Uygulama Süreciyle İlgili Bilgilendirme</i>	Araştırmaya başlamadan önce çalışmaya katılacak gruplara uygulama süreci hakkında detaylı bilgilendirme yapılmıştır.
<i>Veri Toplama Yönergesi</i>	Veri toplama araçlarının nasıl doldurulacağı yönergelerle açıklanmıştır.
<i>Uygulama Ortamı ve Süresi</i>	Uygulama ve veri toplama sürecinin, tüm gruplar için aynı atölye ortamı ve aynı zaman diliminde gerçekleştirilmesine özen gösterilmiştir.
<i>Veri Kaybını Önleme</i>	Her iki gruptaki haftalık etkinliklerde ve görüşmelerde alınan ses ve video kayıtları Google Drive bulut ortamında yedeklenmiş, kayıtlar yazı diline aktarılırken ses ve videolar birbiri ile karşılaştırılarak incelenmiştir.
<i>Veri Analizlerinin Tutarlılığı</i>	Çalışmadaki nicel ve nitel verilerin analizi ve sentezi için uzman, akran ve danışman görüşüne başvurulmuştur.

DÖRDÜNCÜ BÖLÜM

Bulgular

Bu bölümde, üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlama performansları, programlama öz-yeterlikleri ve programlama kaygıları ile ilgili bulgular araştırma soruları çerçevesinde alt başlıklar halinde sunulmuştur.

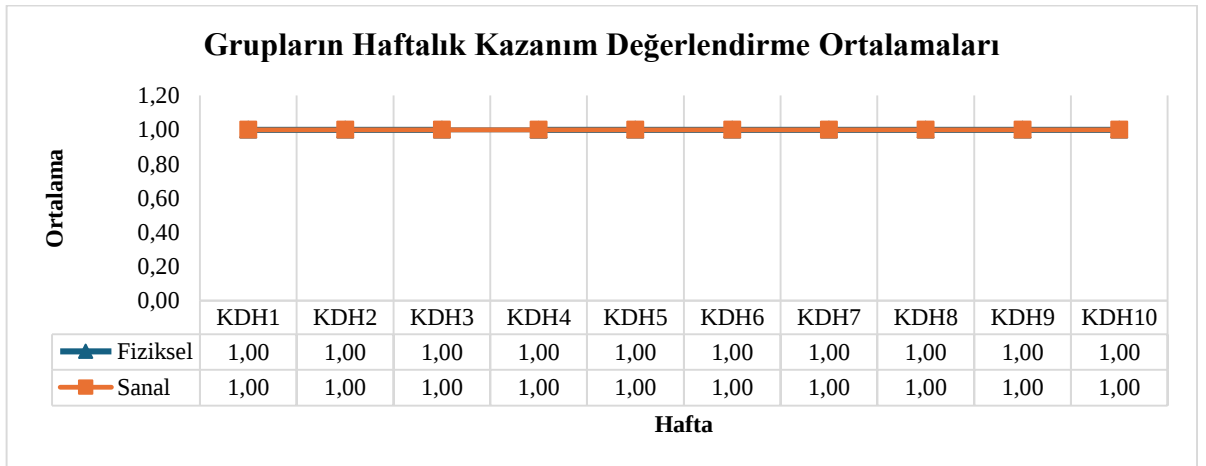
Üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlama performansları arasında anlamlı bir fark var mıdır?

Çalışmada kodlama performansı kapsamında; sanal (VEXcode-VR) ile fiziksel (Raspberry-Pi) robotik kodlama ortamlarındaki etkinliklerin üstün yetenekli öğrencilerin Python programlama dili başarılarına ve programlama dili bilgilerinin kalıcılığına (hatırlanmasına) etkisi incelenmiştir. Bunun yanı sıra, öğrencilerin haftalık kazanımlara göre bu iki farklı ortamda gerçekleşen öğrenme süreçlerinde edindikleri robotik kodlama kazanımları ve meşguliyetleri de karşılaştırılmıştır.

Üstün yetenekli öğrencilerin haftalık sanal ve fiziksel robotik kodlama kazanımları arasında anlamlı bir fark var mıdır?

Çalışmada 10 haftalık uygulama sürecine katılan sanal ve fiziksel robotik kodlama gruplarının haftalık kazanım değerlendirme ortalamaları Şekil 10'da karşılaştırmalı olarak gösterilmiştir.

Şekil 10. Grupların Haftalık Kazanım Değerlendirme Ortalamaları



Şekil 10'da görüldüğü üzere, her iki gruptaki öğrenciler haftalık kazanımlarla ilgili tüm görevleri yerine getirmişlerdir. Buna göre her iki grubun puanlarının haftalar ilerlese bile

istikrarlı seyrettiği, gruplarda genel olarak üst düzeyde öğrenim gerçekleştiği söylenebilir. Bu sonuç, sanal ve fiziksel ortamda yürütülen robotik kodlama ile Python programlama dili eğitiminin öğrencilerin kazanımları gerçekleştirmeleri açısından aynı oranda etkili olduğunu düşündürmektedir.

Üstün yetenekli öğrencilerin haftalık sanal ve fiziksel robotik kodlama meşguliyetleri arasında anlamlı bir fark var mıdır?

Çalışmada sanal ve fiziksel robotik kodlama gruplarının 10 haftalık uygulama sürecindeki haftalık genel meşguliyet ortalamalarının yanı sıra duyuşsal, bilişsel, davranışsal, temsili ve sosyal meşguliyet ortalamaları da Tablo 12’de sunulmuştur.

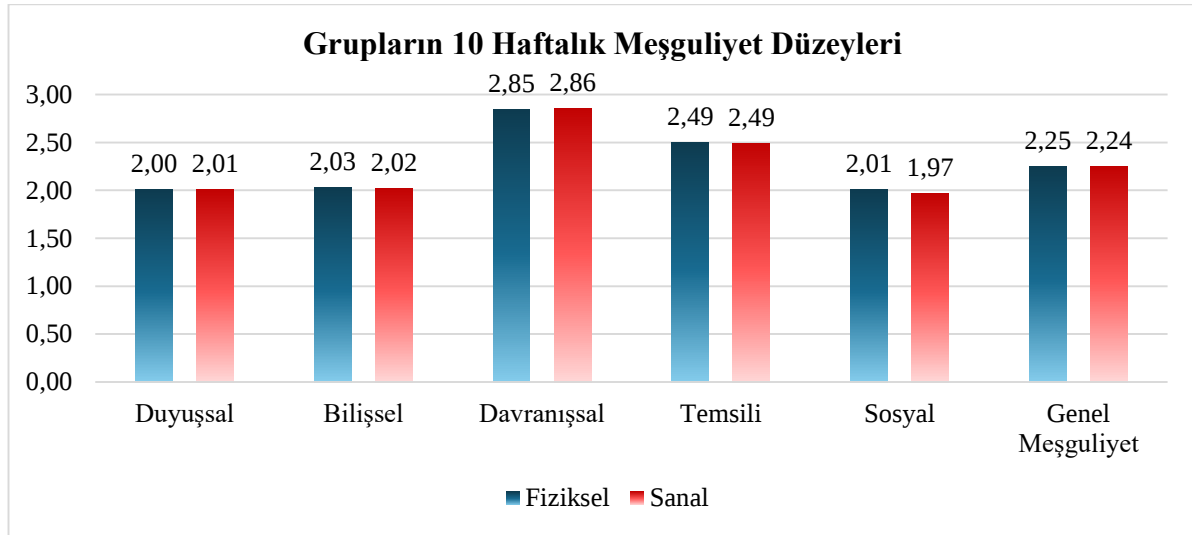
Tablo 12. Meşguliyet Alt Boyutlarına Göre Grupların Haftalık Ortalamaları

Grup	Ölçüm Zamanı	Duyuşsal		Bilişsel		Davranışsal		Temsili		Sosyal	Genel Meşguliyet		
	Haftalar	M	SD	M	SD	M	SD	M	SD	M	SD	M	SD
Fiziksel (N=10)	Hafta1	2.09	.121	2.28	.278	2.87	.044	2.76	.227	1.98	.333	2.38	.165
	Hafta2	2.02	.126	2.05	.127	2.80	.102	2.40	.211	1.78	.175	2.20	.065
	Hafta3	2.01	.091	2.05	.118	2.77	.077	2.42	.148	1.88	.193	2.22	.057
	Hafta4	1.98	.079	1.88	.162	2.83	.115	2.42	.257	1.78	.148	2.15	.065
	Hafta5	1.99	.093	1.84	.135	2.86	.068	2.32	.286	1.92	.103	2.16	.082
	Hafta6	1.98	.128	1.95	.172	2.89	.059	2.52	.193	2.20	.189	2.26	.083
	Hafta7	1.96	.156	2.00	.141	2.84	.107	2.30	.501	2.08	.270	2.22	.116
	Hafta8	1.98	.115	2.02	.079	2.78	.124	2.36	.310	2.10	.216	2.22	.073
	Hafta9	2.05	.088	2.14	.097	2.86	.068	2.68	.235	2.16	.207	2.34	.053
	Hafta10	2.00	.103	2.06	.084	2.96	.068	2.76	.227	2.22	.239	2.35	.042
	10 Haftalık	2.01	.048	2.03	.057	2.85	.032	2.49	.102	2.01	.128	2.25	.030
Sanal (N=10)	Hafta1	2.19	.190	2.34	.158	2.90	.097	2.58	.305	1.90	.194	2.39	.100
	Hafta2	2.03	.080	1.98	.123	2.66	.215	2.44	.158	1.78	.114	2.16	.072
	Hafta3	1.94	.178	1.94	.126	2.84	.143	2.40	.211	1.90	.170	2.18	.119
	Hafta4	2.14	.151	2.01	.160	2.85	.106	2.48	.253	1.80	.249	2.24	.067
	Hafta5	1.93	.169	1.93	.116	2.83	.114	2.40	.189	1.94	.165	2.18	.082
	Hafta6	1.94	.223	1.86	.126	2.87	.127	2.24	.295	2.00	.267	2.15	.112
	Hafta7	1.95	.148	2.02	.123	2.89	.091	2.46	.366	2.06	.284	2.25	.116
	Hafta8	1.95	.119	2.01	.110	2.90	.068	2.46	.389	2.02	.290	2.24	.113
	Hafta9	2.03	.143	2.07	.106	2.94	.072	2.80	.249	2.10	.170	2.34	.071
	Hafta10	2.00	.157	2.04	.117	2.89	.091	2.64	.295	2.18	.220	2.31	.077
	10 Haftalık	2.01	.093	2.02	.063	2.86	.057	2.49	.171	1.97	.119	2.25	.062

Tablo 12 incelendiğinde, sanal ve fiziksel robotik kodlama gruplarının haftalık genel meşguliyetin yanı sıra alt meşguliyet boyutlarındaki ortalamalarının da 2-3 aralığında (2=kısmen gözlendi, 3=gözlendi) ve birbirine yakın seviyede olduğu gözlemlenmiştir. Her iki grupta da davranışsal ve sosyal meşguliyet ortalamalarında haftalar ilerledikçe artış eğilimi

görülmüştür. Bu durum, meşguliyet gözlem formunun ölçeklendirme aralığı göz önüne alındığında, her iki gruptaki öğrencilerin haftalar ilerlese bile robotik kodlama etkinlikleriyle yüksek düzeyde meşgul olmaya devam ettiklerini, uygulama sürecinde aktif katılım sergilediklerini ve sınıf içi etkileşimlerinin zamanla arttığını göstermektedir. Şekil 11’de ise meşguliyet boyutlarındaki ortalamalara göre grupların 10 haftalık meşguliyet düzeyleri gösterilmiştir.

Şekil 11. Grupların 10 Haftalık Meşguliyet Düzeyleri

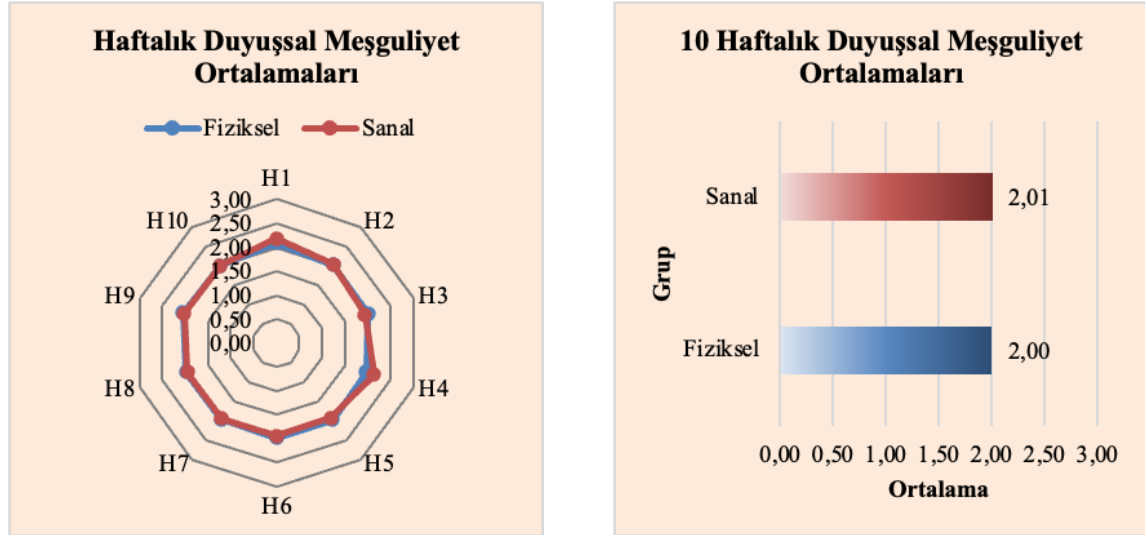


Şekil 11 incelendiğinde, sanal ve fiziksel robotik kodlama gruplarının 10 haftalık duyuşsal, bilişsel, davranışsal, temsili, sosyal ve genel meşguliyet düzeyleri arasında benzerlik olduğu görülmektedir. Her iki grupta da en yüksek meşguliyet düzeyi davranışsal boyutta gözlemlenmiştir. Temsili boyutta da her iki grup aynı ortalamaya sahiptir. Bu bulgular, sanal ve fiziksel ortamların öğrencilerin robotik kodlama meşguliyetleri üzerindeki etkisinin büyük ölçüde benzer olduğunu göstermektedir. Özellikle davranışsal meşguliyet boyutundaki yüksek skorlar, her iki ortamda da öğrencilerin öğrenme sürecine aktif olarak katıldığını ortaya koymaktadır.

Grupların Duyuşsal Meşguliyetleri Arasındaki Fark

Sanal ve fiziksel robotik kodlama gruplarındaki öğrencilerin haftalık ve 10 haftalık duyuşsal meşguliyet ortalamaları Şekil 12’de sunulmuştur.

Şekil 12. Grupların Haftalık ve 10 Haftalık Duyuşsal Meşguliyet Ortalamaları



Şekil 12 incelendiğinde, sanal ve fiziksel robotik kodlama gruplarının haftalık ve 10 haftalık duyuşsal meşguliyet ortalamalarının birbirine yakın seviyede ve 2-3 aralığında (2=kısmen gözlemlendi, 3=gözlemlendi) olduğu gözlemlenmiştir. Buna göre her iki gruptaki öğrencilerin duyuşsal meşguliyet eğilimlerinin süreç boyunca istikrarlı bir şekilde devam ettiği söylenebilir.

Çalışmada haftalık ve 10 haftalık duyuşsal meşguliyet ortalamaları açısından sanal ve fiziksel robotik kodlama grupları arasında anlamlı fark olup olmadığını belirlemek için Mann-Whitney U testi kullanılmış ve test sonuçları Tablo 13'te verilmiştir.

Tablo 13. Grupların Haftalık ve 10 Haftalık Duyuşsal Meşguliyetleri Arasındaki Fark

Ölçüm Zamanı	Grup	N	M	SD	Mean Rank	Sum of Ranks	Mann-Whitney U	z	p
Hafta1	Fiziksel	10	2.09	.121	9.10	91.00	36.000	-	.262
	Sanal	10	2.19	.190	11.90	119.00		1.121	
Hafta2	Fiziksel	10	2.02	.126	10.55	105.50	49.500	-.041	.967
	Sanal	10	2.03	.080	10.45	104.50			
Hafta3	Fiziksel	10	2.01	.091	12.00	120.00	35.000	-	.212
	Sanal	10	1.94	.178	9.00	90.00		1.247	
Hafta4	Fiziksel	10	1.98	.079	7.30	73.00	18.000	-	.004*
	Sanal	10	2.14	.151	13.70	137.00		2.846	
Hafta5	Fiziksel	10	1.99	.093	11.45	114.50	40.500	-	.304
	Sanal	10	1.93	.169	9.55	95.50		1.028	
Hafta6	Fiziksel	10	1.98	.128	10.65	106.50	48.500	-.117	.907
	Sanal	10	1.94	.223	10.35	103.50			
Hafta7	Fiziksel	10	1.96	.156	10.75	107.50	47.500	-.209	.835
	Sanal	10	1.95	.148	10.25	102.50			
Hafta8	Fiziksel	10	1.98	.115	10.90	109.00	46.000	-.356	.722
	Sanal	10	1.95	.119	10.10	101.00			
Hafta9	Fiziksel	10	2.05	.088	10.75	107.50	47.500	-.209	.835
	Sanal	10	2.03	.143	10.25	102.50			
Hafta10	Fiziksel	10	2.00	.103	11.20	112.00	43.000	-.653	.514
	Sanal	10	2.00	.157	9.80	98.00			
10 Haftalık	Fiziksel	10	2.01	.048	10.10	101.00	46.000	-.304	.761
	Sanal	10	2.01	.093	10.90	109.00			

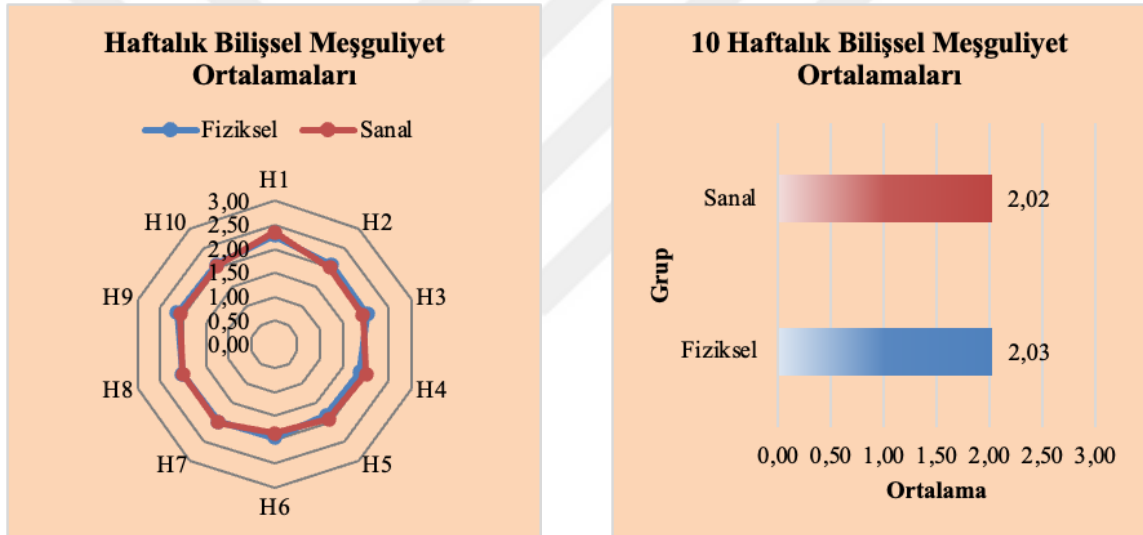
*p<.05

Tablo 13 incelendiğinde, grupların 10 haftalık duyuşsal meşguliyetleri arasında istatistiksel olarak anlamlı bir fark bulunmamıştır ($U=46.000$, $z=-0.304$, $p=.761>.05$). Haftalık ortalamalar açısından ise; gruplar arasında yalnızca “Döngüler” kazanımına yönelik etkinliklerin yapıldığı 4. haftada sanal robotik kodlama grubu lehine istatistiksel olarak anlamlı bir fark bulunmuş, diğer haftalarda anlamlı bir fark saptanmamıştır. Bu durum öğrencilerin “Döngüler” gibi kapsamlı ve çok adımlı kodlama işlemlerinin başlangıcı olan 4. haftada, Raspberry-Pi ile yapılan fiziksel robotik kodlama etkinliğine duyuşsal olarak alışmakta biraz zorlandığını göstermektedir.

Grupların Bilişsel Meşguliyetleri Arasındaki Fark

Sanal ve fiziksel robotik kodlama gruplarındaki öğrencilerin haftalık ve 10 haftalık bilişsel meşguliyet ortalamaları Şekil 13’te sunulmuştur.

Şekil 13. Grupların Haftalık ve 10 Haftalık Bilişsel Meşguliyet Ortalamaları



Şekil 13 incelendiğinde, sanal ve fiziksel robotik kodlama gruplarının haftalık ve 10 haftalık bilişsel meşguliyet ortalamalarının birbirine yakın seviyede ve 2-3 aralığında (2=kısmen gözlendi, 3=gözlendi) olduğu gözlemlenmiştir. Buna göre her iki gruptaki öğrencilerin bilişsel meşguliyet seviyelerinde süreç boyunca belirgin bir ayrışma yaşanmadığı ve istikrarlı bir şekilde devam ettiği söylenebilir.

Çalışmada haftalık ve 10 haftalık bilişsel meşguliyet ortalamaları açısından sanal ve fiziksel robotik kodlama grupları arasında anlamlı fark olup olmadığını belirlemek için Mann-Whitney U testi kullanılmış ve test sonuçları Tablo 14’te verilmiştir.

Tablo 14. Grupların Haftalık ve 10 Haftalık Bilişsel Meşguliyetleri Arasındaki Fark

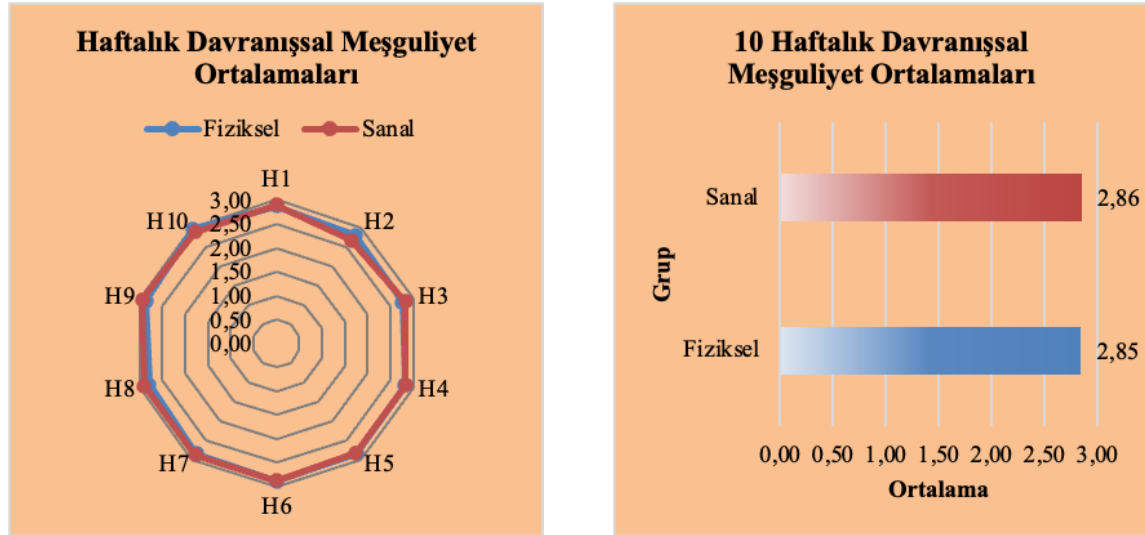
Ölçüm Zamanı	Grup	N	M	SD	Mean Rank	Sum of Ranks	Mann-Whitney U	z	p
Hafta1	Fiziksel	10	2.28	.278	9.50	95.00	40.000	-0.763	.446
	Sanal	10	2.34	.158	11.50	115.00			
Hafta2	Fiziksel	10	2.05	.127	12.15	121.50	33.500	-1.281	.200
	Sanal	10	1.98	.123	8.85	88.50			
Hafta3	Fiziksel	10	2.05	.118	12.95	129.50	25.500	-1.912	.056
	Sanal	10	1.94	.126	8.05	80.50			
Hafta4	Fiziksel	10	1.88	.162	8.45	84.50	29.500	-1.593	.111
	Sanal	10	2.01	.160	12.55	125.50			
Hafta5	Fiziksel	10	1.84	.135	8.85	88.50	33.500	-1.286	.199
	Sanal	10	1.93	.116	12.15	121.50			
Hafta6	Fiziksel	10	1.95	.172	12.40	124.00	31.000	-1.495	.135
	Sanal	10	1.86	.126	8.60	86.00			
Hafta7	Fiziksel	10	2.00	.141	9.85	98.50	43.500	-.511	.609
	Sanal	10	2.02	.123	11.15	111.50			
Hafta8	Fiziksel	10	2.02	.079	10.90	109.00	46.000	-.318	.751
	Sanal	10	2.01	.110	10.10	101.00			
Hafta9	Fiziksel	10	2.14	.097	12.30	123.00	32.000	-1.421	.155
	Sanal	10	2.07	.106	8.70	87.00			
Hafta10	Fiziksel	10	2.06	.084	11.00	110.00	45.000	-.396	.692
	Sanal	10	2.04	.117	10.00	100.00			
10 Haftalık	Fiziksel	10	2.03	.057	11.00	110.00	45.000	-.379	.704
	Sanal	10	2.02	.063	10.00	100.00			

Tablo 14 incelendiğinde, grupların haftalık ve 10 haftalık bilişsel meşguliyetleri arasında istatistiksel olarak anlamlı bir fark bulunmamıştır. Bu durum, hem sanal (VEXcode-VR) hem de fiziksel (Raspberry-Pi) robotik kodlama grubundaki üstün yetenekli öğrencilerin Python programlama dili öğrenme sürecindeki bilişsel meşguliyetlerinin benzer oranda gerçekleştiğine işaret etmektedir.

Grupların Davranışsal Meşguliyetleri Arasındaki Fark

Sanal ve fiziksel robotik kodlama gruplarındaki öğrencilerin haftalık ve 10 haftalık davranışsal meşguliyet ortalamaları Şekil 14'te sunulmuştur.

Şekil 14. Grupların Haftalık ve 10 Haftalık Davranışsal Meşguliyet Ortalamaları



Şekil 14 incelendiğinde, sanal ve fiziksel robotik kodlama gruplarının haftalık ve 10 haftalık davranışsal meşguliyet ortalamalarının birbirine yakın değerlerde ve 3 seviyesine (3=gözlendi) çok yakın olduğu gözlemlenmiştir. Buna göre her iki gruptaki öğrencilerin davranışsal meşguliyet eğilimlerinin süreç boyunca istikrarlı bir şekilde yüksek seviyede devam ettiği söylenebilir.

Çalışmada haftalık ve 10 haftalık davranışsal meşguliyet ortalamaları açısından sanal ve fiziksel robotik kodlama grupları arasında anlamlı fark olup olmadığını belirlemek için Mann-Whitney U testi kullanılmış ve test sonuçları Tablo 15’te verilmiştir.

Tablo 15. Grupların Haftalık ve 10 Haftalık Davranışsal Meşguliyetleri Arasındaki Fark

Ölçüm Zamanı	Grup	N	M	SD	Mean Rank	Sum of Ranks	Mann-Whitney U	z	p
Hafta1	Fiziksel	10	2.87	.044	9.45	94.50	39.500	-0.990	.322
	Sanal	10	2.90	.097	11.55	115.50			
Hafta2	Fiziksel	10	2.80	.102	12.65	126.50	28.500	-1.718	.086
	Sanal	10	2.66	.215	8.35	83.50			
Hafta3	Fiziksel	10	2.77	.077	8.70	87.00	32.000	-1.458	.145
	Sanal	10	2.84	.143	12.30	123.00			
Hafta4	Fiziksel	10	2.83	.115	9.80	98.00	43.000	-0.601	.548
	Sanal	10	2.85	.106	11.20	112.00			
Hafta5	Fiziksel	10	2.86	.068	11.00	110.00	45.000	-0.497	.619
	Sanal	10	2.83	.114	10.00	100.00			
Hafta6	Fiziksel	10	2.89	.059	10.70	107.00	48.000	-.168	.866
	Sanal	10	2.87	.127	10.30	103.00			
Hafta7	Fiziksel	10	2.84	.107	9.35	93.50	38.500	-.965	.335
	Sanal	10	2.89	.091	11.65	116.50			
Hafta8	Fiziksel	10	2.78	.124	7.55	75.50	20.500	-2.378	.017*
	Sanal	10	2.90	.068	13.45	134.50			
Hafta9	Fiziksel	10	2.86	.068	7.80	78.00	23.000	-2.368	.018*
	Sanal	10	2.94	.072	13.20	132.00			
Hafta10	Fiziksel	10	2.96	.068	12.65	126.50	28.500	-1.834	.067
	Sanal	10	2.89	.091	8.35	83.50			
10 Haftalık	Fiziksel	10	2.85	.032	9.05	90.50	35.500	-1.102	.270
	Sanal	10	2.86	.057	11.95	119.50			

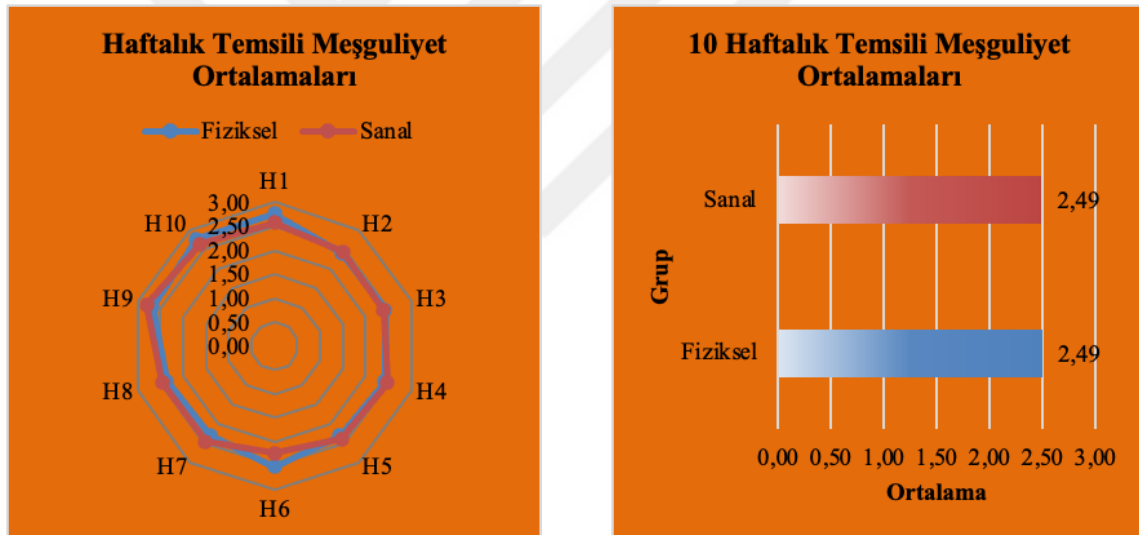
*p<.05

Tablo 15 incelendiğinde, grupların 10 haftalık davranışsal meşguliyetleri arasında istatistiksel olarak anlamlı bir fark bulunmamıştır ($U=35.50$, $z=-1.102$, $p=.270>.05$). Haftalık ortalamalar açısından ise; gruplar arasında yalnızca 8. ve 9. haftalarda sanal robotik kodlama grubu lehine davranışsal meşguliyete yönelik istatistiksel olarak anlamlı bir fark saptanmıştır. Bu durum, 10 haftalık uygulama sürecinin sonuna doğru yapılan 8. haftadaki “Listeler” ile 9. haftadaki “Proje Gerçekleştirme-Özgün Çözüm Bulma, Hata Ayıklama” kazanımlarının temel alındığı kapsamlı ve çok adımlı kodlama işlemlerini içeren etkinliklerde, sanal (VEXcode-VR) robotik kodlama grubundaki öğrencilerin davranışsal meşguliyetlerinin anlamlı seviyede daha yüksek olduğunu ve daha fazla aktif katılım sergilediğini göstermektedir.

Grupların Temsili Meşguliyetleri Arasındaki Fark

Sanal ve fiziksel robotik kodlama gruplarındaki öğrencilerin haftalık ve 10 haftalık temsili meşguliyet ortalamaları Şekil 15’te sunulmuştur.

Şekil 15. Grupların Haftalık ve 10 Haftalık Temsili Meşguliyet Ortalamaları



Şekil 15. incelendiğinde, sanal ve fiziksel robotik kodlama gruplarının haftalık ve 10 haftalık temsili meşguliyet ortalamalarının 2-3 aralığında (2=kısmen gözlendi, 3=gözlendi) ve birbirine yakın değerlerde olduğu gözlemlenmiştir. Buna göre her iki gruptaki öğrencilerin temsili meşguliyet eğilimlerinin süreç boyunca birbirine denk ve istikrarlı bir şekilde devam ettiği söylenebilir.

Çalışmada haftalık ve 10 haftalık temsili meşguliyet ortalamaları açısından sanal ve fiziksel robotik kodlama grupları arasında anlamlı fark olup olmadığını belirlemek için Mann-Whitney U testi kullanılmış ve test sonuçları Tablo 16’da verilmiştir.

Tablo 16. Grupların Haftalık ve 10 Haftalık Temsili Meşguliyetleri Arasındaki Fark

Ölçüm Zamanı	Grup	N	M	SD	Mean Rank	Sum of Ranks	Mann-Whitney U	z	p
Hafta1	Fiziksel	10	2.76	.227	12.40	124.00	31.000	-	.133
	Sanal	10	2.58	.305	8.60	86.00		1.501	
Hafta2	Fiziksel	10	2.40	.211	10.05	100.50	45.500	-	.689
	Sanal	10	2.44	.158	10.95	109.50		1.501	
Hafta3	Fiziksel	10	2.42	.148	11.05	110.50	44.500	-	.660
	Sanal	10	2.40	.211	9.95	99.50		0.439	
Hafta4	Fiziksel	10	2.42	.257	9.70	97.00	42.000	-	.533
	Sanal	10	2.48	.253	11.30	113.00		0.624	
Hafta5	Fiziksel	10	2.32	.286	9.70	97.00	42.000	-	.531
	Sanal	10	2.40	.189	11.30	113.00		0.626	
Hafta6	Fiziksel	10	2.52	.193	13.30	133.00	22.000	-	.026*
	Sanal	10	2.24	.295	7.70	77.00		2.231	
Hafta7	Fiziksel	10	2.30	.501	9.25	92.50	37.500	-	.338
	Sanal	10	2.46	.366	11.75	117.50		0.959	
Hafta8	Fiziksel	10	2.36	.310	9.30	93.00	38.000	-	.353
	Sanal	10	2.46	.389	11.70	117.00		0.928	
Hafta9	Fiziksel	10	2.68	.235	8.90	89.00	34.000	-	.209
	Sanal	10	2.80	.249	12.10	121.00		1.257	
Hafta10	Fiziksel	10	2.76	.227	11.70	117.00	38.000	-	.348
	Sanal	10	2.64	.295	9.30	93.00		0.938	
10 Haftalık	Fiziksel	10	2.49	.102	9.85	98.50	43.500	-	.620
	Sanal	10	2.49	.171	11.15	111.50		0.496	

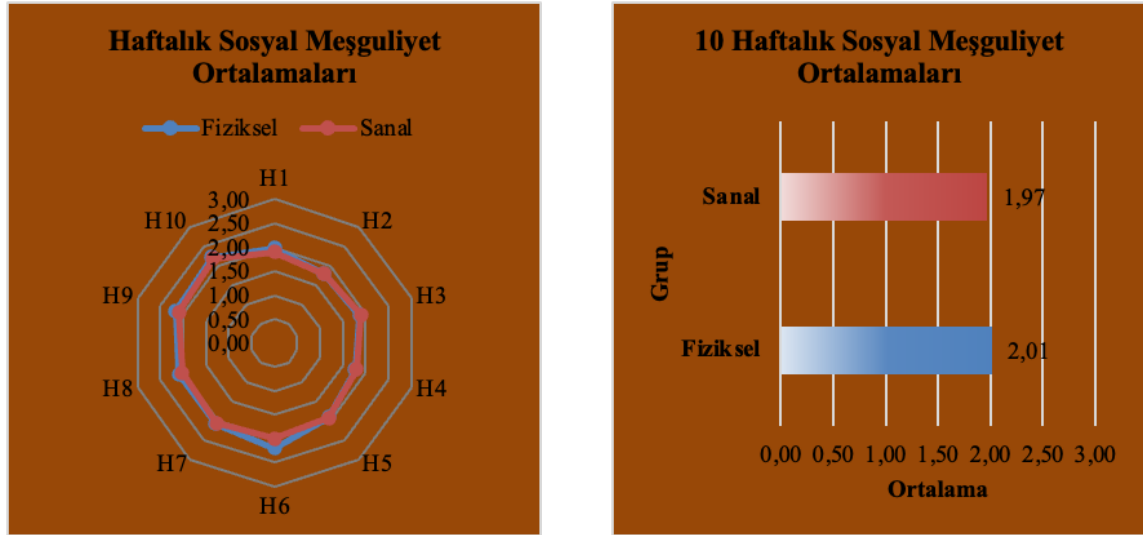
*p<.05

Tablo 16 incelendiğinde, grupların 10 haftalık temsili meşguliyetleri arasında istatistiksel olarak anlamlı bir fark bulunmamıştır ($U=43.500$, $z=-0.496$, $p=.620>.05$). Haftalık ortalamalar açısından ise; yalnızca “*Fonksiyonlar*” kazanımına yönelik etkinliklerin yapıldığı 6. haftada gruplar arasında fiziksel robotik kodlama grubu lehine istatistiksel olarak anlamlı bir fark bulunmuş, diğer haftalarda anlamlı bir fark saptanmamıştır. Bu durum öğrencilerin “*Fonksiyonlar*” gibi iç içe kodlama işlem adımlarının olduğu 6. haftada, Raspberry-Pi ile yapılan fiziksel robotik kodlama etkinliğinde kendi duygu ve düşüncelerini daha fazla dile getirerek temsili meşguliyet sergilediklerini göstermektedir.

Grupların Sosyal Meşguliyetleri Arasındaki Fark

Sanal ve fiziksel robotik kodlama gruplarındaki öğrencilerin haftalık ve 10 haftalık sosyal meşguliyet ortalamaları Şekil 16’da sunulmuştur.

Şekil 16. Grupların Haftalık ve 10 Haftalık Sosyal Meşguliyet Ortalamaları



Şekil 16 incelendiğinde hem sanal hem fiziksel sosyal meşguliyet ortalamalarının haftalar ilerledikçe 2 seviyesine (2=kısmen gözlendi) doğru artma eğiliminde olduğu saptanmıştır. 10 haftalık ortalamalar açısından ise, fiziksel robotik kodlama grubunun sosyal meşguliyet ortalamasının daha yüksek ve 2 seviyesinde olduğu gözlemlenmiştir. Buna göre, her ne kadar her iki grubun sosyal meşguliyet seviyeleri haftalık olarak artma eğilimi gösterecek olsa da, 10 haftalık süreç boyunca fiziksel robotik kodlama grubundaki öğrencilerin sosyal etkileşimlerinin sanal robotik kodlama grubundakilere kıyasla daha sık gerçekleştiği söylenebilir.

Çalışmada haftalık ve 10 haftalık sosyal meşguliyet ortalamaları açısından sanal ve fiziksel robotik kodlama grupları arasında anlamlı fark olup olmadığını belirlemek için Mann-Whitney U testi kullanılmış ve test sonuçları Tablo 17’de verilmiştir.

Tablo 17. Grupların Haftalık ve 10 Haftalık Sosyal Meşguliyetleri Arasındaki Fark

Ölçüm Zamanı	Grup	N	M	SD	Mean Rank	Sum of Ranks	Mann-Whitney U	z	p
Hafta1	Fiziksel	10	1.98	.333	11.30	113.00	42.000	-0.638	.523
	Sanal	10	1.90	.194	9.70	97.00			
Hafta2	Fiziksel	10	1.78	.175	10.80	108.00	47.000	-.267	.789
	Sanal	10	1.78	.114	10.20	102.00			
Hafta3	Fiziksel	10	1.88	.193	10.25	102.50	47.500	-0.200	.841
	Sanal	10	1.90	.170	10.75	107.50			
Hafta4	Fiziksel	10	1.78	.148	10.25	102.50	47.500	-0.197	.844
	Sanal	10	1.80	.249	10.75	107.50			
Hafta5	Fiziksel	10	1.92	.103	9.90	99.00	44.000	-0.521	.603
	Sanal	10	1.94	.165	11.10	111.00			
Hafta6	Fiziksel	10	2.20	.189	13.05	130.50	24.500	-2.004	.045*
	Sanal	10	2.00	.267	7.95	79.50			
Hafta7	Fiziksel	10	2.08	.270	11.00	110.00	45.000	-.387	.698
	Sanal	10	2.06	.284	10.00	100.00			
Hafta8	Fiziksel	10	2.10	.216	11.45	114.50	40.500	-.740	.459
	Sanal	10	2.02	.290	9.55	95.50			

Tablo 17 (Devamı)

Hafta9	Fiziksel	10	2.16	.207	11.50	115.00	40.000	-.818	.413
	Sanal	10	2.10	.170	9.50	95.00			
Hafta10	Fiziksel	10	2.22	.239	10.80	108.00	47.000	-.236	.814
	Sanal	10	2.18	.220	10.20	102.00			
10 Haftalık	Fiziksel	10	2.01	.128	11.60	116.00	39.000	-.833	.405
	Sanal	10	1.97	.119	9.40	94.00			

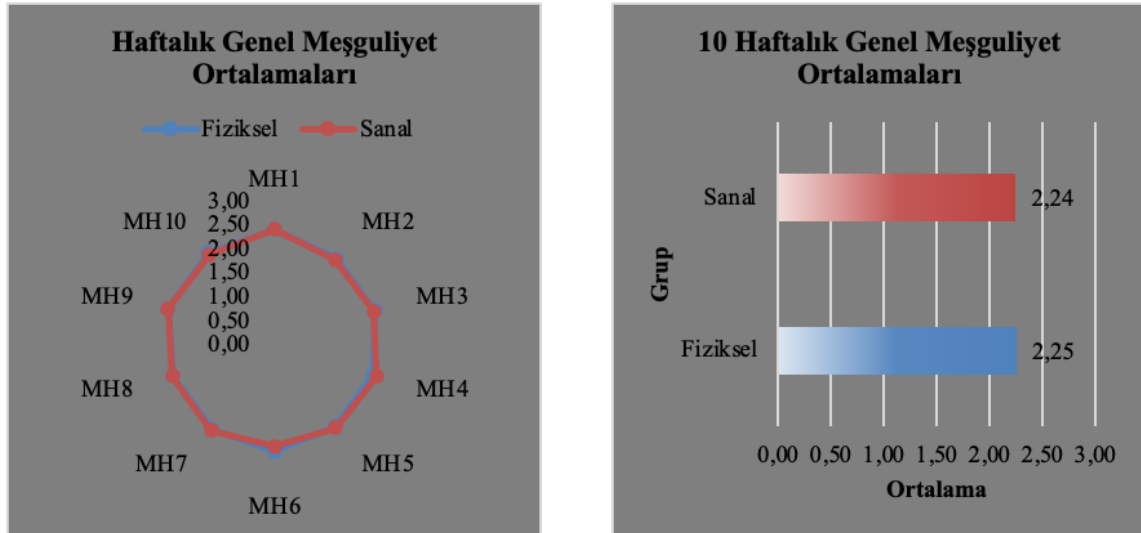
*p<.05

Tablo 17 incelendiğinde, grupların 10 haftalık sosyal meşguliyetleri arasında istatistiksel olarak anlamlı bir fark bulunmamıştır ($U=39.000$, $z=-0.833$, $p=.405>.05$). Bu bulgu, sanal ve fiziksel robotik kodlama gruplarındaki öğrencilerin sosyal etkileşimi dengeli biçimde sürdürdüğünü düşündürmektedir. Haftalık ortalamalar açısından ise; yalnızca “Fonksiyonlar” kazanımına yönelik etkinliklerin yapıldığı 6. haftada gruplar arasında fiziksel robotik kodlama grubu lehine istatistiksel olarak anlamlı bir fark saptanmıştır. Bu durum öğrencilerin “Fonksiyonlar” gibi iç içe kodlama işlem adımlarının olduğu 6. haftada, Raspberry-Pi ile yapılan fiziksel robotik kodlama etkinliğinde daha fazla sosyal etkileşim içinde olduklarını göstermektedir.

Grupların Genel Meşguliyetleri Arasındaki Fark

Sanal ve fiziksel robotik kodlama gruplarındaki öğrencilerin haftalık ve 10 haftalık genel meşguliyet ortalamaları Şekil 17’de sunulmuştur.

Şekil 17. Grupların Haftalık ve 10 Haftalık Genel Meşguliyet Ortalamaları



Şekil 17 incelendiğinde, sanal ve fiziksel robotik kodlama gruplarının haftalık ve 10 haftalık genel meşguliyet ortalamalarının birbirine yakın değerlerde ve 2-3 aralığında (2=kısmen gözlemlendi, 3=gözlemlendi) olduğu gözlemlenmiştir. Buna göre her iki gruptaki öğrencilerin genel meşguliyet eğilimlerinin süreç boyunca büyük ölçüde paralellik gösterdiği söylenebilir. Çalışmada haftalık ve 10 haftalık genel meşguliyet ortalamaları açısından sanal ve

fiziksel robotik kodlama grupları arasında anlamlı fark olup olmadığını belirlemek için Mann-Whitney U testi kullanılmış ve test sonuçları Tablo 18’de verilmiştir.

Tablo 18. *Grupların Haftalık ve 10 Haftalık Genel Meşguliyetleri Arasındaki Fark*

Ölçüm Zamanı	Grup	N	M	SD	Mean Rank	Sum of Ranks	Mann-Whitney U	z	p
Hafta1	Fiziksel	10	2.38	.165	10.25	102.50	47.500	-	.849
	Sanal	10	2.39	.100	10.75	107.50		0.190	
Hafta2	Fiziksel	10	2.20	.065	12.10	121.00	34.000	-	.222
	Sanal	10	2.16	.072	8.90	89.00		1.221	
Hafta3	Fiziksel	10	2.22	.057	11.60	116.00	39.000	-	.401
	Sanal	10	2.18	.119	9.40	94.00		0.839	
Hafta4	Fiziksel	10	2.15	.065	7.25	72.50	17.500	-	.013*
	Sanal	10	2.24	.067	13.75	137.50		2.481	
Hafta5	Fiziksel	10	2.16	.082	9.50	95.00	40.000	-	.445
	Sanal	10	2.18	.082	11.50	115.00		0.764	
Hafta6	Fiziksel	10	2.26	.083	13.25	132.50	22.500	-	.035*
	Sanal	10	2.15	.112	7.75	77.50		2.107	
Hafta7	Fiziksel	10	2.22	.116	9.75	97.50	42.500	-	.569
	Sanal	10	2.25	.116	11.25	112.50		-570	
Hafta8	Fiziksel	10	2.22	.073	9.80	98.00	43.000	-	.592
	Sanal	10	2.24	.113	11.20	112.00		-535	
Hafta9	Fiziksel	10	2.34	.053	10.40	104.00	49.000	-	.938
	Sanal	10	2.34	.071	10.60	106.00		-078	
Hafta10	Fiziksel	10	2.35	.042	12.25	122.50	32.500	-	.177
	Sanal	10	2.31	.077	8.75	87.50		1.349	
10 Haftalık	Fiziksel	10	2.25	.030	10.55	105.50	49.500	-	.969
	Sanal	10	2.25	.062	10.45	104.50		-038	

*p<.05

Tablo 18 incelendiğinde, grupların 10 haftalık genel meşguliyetleri arasında istatistiksel olarak anlamlı bir fark bulunmamıştır ($U=49.500$, $z=-0.038$, $p=.969>.05$). Bu bulgu hem sanal hem de fiziksel robotik kodlama ortamının üstün yetenekli öğrencilerin Python programlama dili öğrenme sürecindeki genel meşguliyetlerini destekleyici olduğunu göstermektedir. Haftalık genel meşguliyete yönelik ortalamalar açısından ise; gruplar arasında “Döngüler” kazanımına yönelik etkinliklerin yapıldığı 4. haftada sanal robotik kodlama grubu lehine istatistiksel olarak anlamlı bir fark bulunmuştur. Bu durum “Döngüler” gibi kapsamlı ve çok adımlı kodlama işlemlerinin başlangıcı olan 4. haftada, öğrencilerin VEXcode-VR ile yapılan sanal robotik kodlama etkinliğiyle daha fazla meşgul olduğunu göstermektedir. “Fonksiyonlar” kazanımına yönelik etkinliklerin yapıldığı 6. haftada ise, fiziksel robotik kodlama grubu lehine istatistiksel olarak anlamlı bir fark saptanmıştır. Bu durum öğrencilerin “Fonksiyonlar” gibi iç içe kodlama işlem adımlarının olduğu 6. haftada, Raspberry-Pi ile yapılan fiziksel robotik kodlama etkinliğiyle daha fazla meşgul olduğunu göstermektedir.

Üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlamadaki Python programlama dili başarıları arasında anlamlı bir fark var mıdır?

Sanal (VEXcode-VR) ile fiziksel (Raspberry-Pi) robotik kodlama ortamlarındaki etkinliklere katılan üstün yetenekli öğrencilerin Python programlama dili ön test ve son test başarılarının yanı sıra programlama dili bilgilerinin kalıcılığı (hatırlanması) da incelenmiştir. Ölçüm zamanına göre grupların robotik kodlamada Python programlama dili başarı ortalamaları Tablo 19’da sunulmuştur.

Tablo 19. Ölçüm Zamanına Göre Grupların Robotik Kodlama Başarı Ortalamaları

Grup	Ölçüm Zamanı	N	M (Mean)	SD (Std. Deviation)
Fiziksel	Ön Test	10	28.80	6.197
	Son Test	10	61.20	8.854
	Kalıcılık Testi	10	63.20	6.746
Sanal	Ön Test	10	34.40	10.013
	Son Test	10	75.60	11.067
	Kalıcılık Testi	10	74.00	12.543

Tablo 19 incelendiğinde, her iki grubun da son test başarı ortalamaları, ön test başarı ortalamalarına göre artış göstermiştir. Ancak bu atışın sanal robotik kodlama grubunda daha yüksek olduğu görülmüştür. Buna göre her iki gruptaki öğrencilerin Python programlama dili başarılarının attığı söylenebilir. Diğer taraftan, fiziksel robotik kodlama grubunun kalıcılık testi ortalaması son testten daha yüksektir. Sanal robotik kodlama grubunda ise kalıcılık testi ortalaması son teste yakın değerlerde olsa da düşüş göstermiştir. Yine de sanal robotik kodlama grubunun kalıcılık testi ortalamasının fiziksel robotik kodlama grubundan yüksek olduğu belirlenmiştir. Buna göre sanal robotik kodlama grubunda Python programlama diline yönelik bilginin kalıcılık (hatırlanma) düzeyinin fiziksel robotik kodlama grubuna göre daha yüksek değerlerde olduğu söylenebilir.

Grup İçi Python Programlama Dili Başarıları Arasındaki Fark

Her iki grubun, ön test ve son teste ilişkin grup içi Python programlama dili başarı ortalamaları arasında fark olup olmadığını belirlemek için Wilcoxon İşaretli Sıralar Testi kullanılmıştır. Elde edilen bulgular Tablo 20’de sunulmuştur.

Tablo 20. Grup İçi Başarı Ön Test ve Son Teste İlişkin Wilcoxon İşaretli Sıralar Testi Sonuçları

Grup		N	Mean Rank (Sıra Ortalaması)	Sum of Ranks (Sıra Toplamı)	z	p	η^2 (Etki Büyüklüğü)
Fiziksel	Son Test- Ön Test	Negatif Sıralar	0	.00	-2.809	.005*	-.89
		Pozitif Sıralar	10	55.00			
		Fark olmayan	0				
		Toplam	10				
Sanal	Son Test- Ön Test	Negatif Sıralar	0	.00	-2.807	.005*	-.89
		Pozitif Sıralar	10	55.00			
		Fark olmayan	0				
		Toplam	10				

*Pozitif sıralara dayalı ($p < .05$)

Tablo 20 incelendiğinde; her iki grupta da uygulama öncesi ve sonrası başarı ortalamaları arasında son test lehine yüksek etkiye sahip anlamlı fark olduğu belirlenmiştir ($p < .05$, $\eta^2 = -.89$). Bu durum, hem sanal ve hem fiziksel robotik kodlama araçlarıyla yapılan etkinliklerde öğrencilerin Python programlama dilini öğrenme seviyelerinin anlamlı ve güçlü bir şekilde gelişmeye devam ettiğini göstermektedir.

Grupların Python Programlama Dili Başarıları Arasındaki Fark

Sanal ve fiziksel robotik kodlama gruplarının ön test ve son teste ilişkin Python programlama dili başarı ortalamaları açısından gruplar arası fark olup olmadığını belirlemek için Mann-Whitney U testi kullanılmıştır. Elde edilen bulgular Tablo 21’de sunulmuştur.

Tablo 21. Gruplar Arası Başarı Ön Test ve Son Teste İlişkin Mann-Whitney U Sonuçları

Ölçüm Zamanı	Grup	N	M	SD	Mean Rank	Sum of Ranks	Mann-Whitney U	z	p	η^2
Ön Test	Fiziksel	10	28.80	6.197	8.65	86.50	31.500	-1.423	.155	.32
	Sanal	10	34.40	10.013	12.35	123.50				
Son Test	Fiziksel	10	61.20	8.854	7.25	72.50	17.500	-2.485	.013*	.56
	Sanal	10	75.60	11.067	13.75	137.50				

* $p < .05$

Tablo 21’e göre, sanal ve fiziksel robotik kodlama gruplarının ön test başarı ortalamaları arasında istatistiksel olarak anlamlı bir fark bulunmamıştır ($p > .05$). Bu bulgu, her iki gruba uygulanan robotik kodlama öğretim süreci öncesinde bu iki grubun da benzer düzeyde Python programlama dili bilgisine sahip olduğunu göstermektedir.

Diğer taraftan sanal ve fiziksel robotik kodlama gruplarının son test başarı ortalamaları arasında sanal robotik kodlama grubu lehine istatistiksel olarak anlamlı seviyede ve yüksek etkiye sahip fark olduğu belirlenmiştir ($p<.05$, $\eta^2=-.56$). Bu durum, üstün yetenekli öğrencilerin Python programlama dili öğrenme sürecinde, sanal robotik kodlama aracının fiziksel robotik kodlama aracına göre daha etkili olduğunu göstermektedir.

Üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlamadaki Python programlama dili bilgilerinin kalıcılığı (hatırlanması) arasında anlamlı bir fark var mıdır?

Sanal ve fiziksel robotik kodlama gruplarında 10 haftalık etkinlik süreci bittikten ve son testler uygulandıktan 4 hafta sonra, her iki gruptaki öğrencilerin Python programlama dili bilgilerini hatırlama seviyesini belirlemek için kalıcılık testi uygulanmıştır.

Grup İçi Python Programlama Dili Bilgilerinin Kalıcılığı (Hatırlanması) Arasındaki Fark

Her iki grubun, grup içi Python programlama dili ön test ve kalıcılık testi ile son test ve kalıcılık testi ortalamaları arasında fark olup olmadığını belirlemek için Wilcoxon İşaretli Sıralar Testi kullanılmıştır. Elde edilen bulgular Tablo 22’de sunulmuştur.

Tablo 22. *Grup İçi Bilginin Kalıcılığına (Hatırlanmasına) İlişkin Wilcoxon İşaretli Sıralar Testi Sonuçları*

Grup		N	Mean Rank	Sum of Ranks	z	p	η^2
Fiziksel	Negatif Sıralar	2	6.00	12.00	-1.268	.205	-.40
	Kalıcılık Testi-Son Test						
	Pozitif Sıralar	7	4.71	33.00			
	Fark olmayan	1					
	Toplam	10					
	Negatif Sıralar	0	.00	.00	-2.814	.005*	-.89
	Kalıcılık Testi-Ön Test						
	Pozitif Sıralar	10	5.50	55			
	Fark olmayan	0					
Sanal	Toplam	10			-.962	.336	-.30
	Negatif Sıralar	3	3.67	11.00			
	Kalıcılık Testi-Son Test						
	Pozitif Sıralar	2	2.00	4.00			
	Fark olmayan	5					
	Toplam	10			-2.812	.005*	-.89
	Negatif Sıralar	0	.00	.00			
	Kalıcılık Testi-Ön Test						
	Pozitif Sıralar	10	5.50	55			
	Fark olmayan	0					
	Toplam	10					

*Pozitif sıralara dayalı ($p<.05$)

Tablo 22 incelendiğinde; her iki grupta da ön test ve kalıcılık testi ortalamaları arasında kalıcılık testi lehine yüksek etkiye sahip anlamlı fark olduğu belirlenmiştir ($p<.05$, $\eta^2=-.89$). Bu durum, hem sanal ve hem fiziksel robotik kodlama araçlarıyla yapılan etkinliklerde öğrencilerin Python programlama dilini bilgilerinin anlamlı ve güçlü bir seviyede kalıcı olduğunu göstermektedir.

Diğer taraftan her iki grupta da son test ve kalıcılık testi ortalamaları arasında istatistiksel olarak anlamlı bir fark bulunmamıştır ($p>.05$). Bu durum, hem sanal ve hem fiziksel robotik kodlama araçlarıyla yapılan etkinliklerde üstün yetenekli öğrencilerin edindiği Python programlama dili bilgilerini, tüm uygulama bitiminden 4 hafta sonra bile benzer seviyede hatırladıklarını göstermektedir.

Grupların Python Programlama Dili Bilgilerinin Kalıcılığı (Hatırlanması) Arasındaki Fark

Sanal ve fiziksel robotik kodlama gruplarının kalıcılık testine ilişkin Python programlama dili başarı ortalamaları açısından gruplar arası fark olup olmadığını belirlemek için Mann-Whitney U testi kullanılmıştır. Elde edilen bulgular Tablo 23'te sunulmuştur.

Tablo 23. *Gruplar Arası Bilginin Kalıcılığına (Hatırlanmasına) İlişkin Mann-Whitney U Sonuçları*

Ölçüm Zamanı	Grup	N	M	SD	Mean Rank	Sum of Ranks	Mann-Whitney U	z	p	η^2
Kalıcılık Testi	Fiziksel	10	63.20	6.745	7.75	77.50	22.500	-2.103	.035*	-.47
	Sanal	10	74.00	12.543	13.25	132.50				

* $p<.05$

Tablo 23'e göre, sanal ve fiziksel robotik kodlama gruplarının kalıcılık testi başarı ortalamaları arasında sanal robotik kodlama grubu lehine istatistiksel olarak anlamlı ve orta seviyede etkiye sahip bir fark olduğu belirlenmiştir ($p<.05$, $\eta^2=-.47$). Bu durum, sanal robotik kodlama aracıyla yapılan eğitimde fiziksel robotik kodlama aracıyla yapılandırılan göre üstün yetenekli öğrencilerin Python programlama dili bilgilerinin daha kalıcı olduğunu, sanal robotik kodlama aracının bilginin hatırlanmasında daha etkili olduğunu göstermektedir.

Üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlama öz-yeterlikleri arasında anlamlı bir fark var mıdır?

Grup İçi Robotik Kodlama Öz-Yeterlikleri Arasındaki Fark

Her iki grubun, grup içi ön test ve son test robotik kodlama öz-yeterlik ortalamaları arasında fark olup olmadığını belirlemek için Wilcoxon İşaretli Sıralar Testi kullanılmıştır. Elde edilen bulgular Tablo 24'te sunulmuştur.

Tablo 24. Grup İçi Öz-Yeterlik Ön Test ve Son Teste İlişkin Wilcoxon İşaretli Sıralar Testi Sonuçları

Grup		N	Mean Rank	Sum of Ranks	z	p	η^2
Fiziksel	Son Test-Ön Test	Negatif Sıralar	2	1.50	-2.310	.021*	-.73
		Pozitif Sıralar	7	6.00			
		Fark olmayan	1				
		Toplam	10				
Sanal	Son Test-Ön Test	Negatif Sıralar	5	5.80	-.153	.878	-.05
		Pozitif Sıralar	5	5.20			
		Fark olmayan	0				
		Toplam	10				

*Pozitif sıralara dayalı ($p < .05$)

Tablo 24 incelendiğinde; fiziksel robotik kodlama grubunun uygulama öncesi ve sonrası robotik kodlama öz-yeterlik ortalamaları arasında son test lehine yüksek etkiye sahip anlamlı fark olduğu belirlenmiştir ($p < .05$, $\eta^2 = -.73$). Bu durum, fiziksel robotik kodlama araçlarıyla yapılan etkinliklerin öğrencilerin Python programlama dilini öğrenmeye yönelik öz-yeterlik seviyeleri üzerinde anlamlı ve güçlü bir etkisi olduğunu göstermektedir.

Diğer taraftan sanal robotik kodlama grubunun uygulama öncesi ve sonrası robotik kodlama öz-yeterlik ortalamaları arasında istatistiksel olarak anlamlı bir fark bulunmamıştır ($p > .05$). Bu durum, sanal robotik kodlama araçlarıyla yapılan etkinliklerin öğrencilerin Python programlama dilini öğrenmeye yönelik öz-yeterlik seviyeleri üzerinde güçlü bir etkisi olmadığını göstermektedir.

Grupların Robotik Kodlama Öz-Yeterlikleri Arasındaki Fark

Sanal ve fiziksel robotik kodlama gruplarının ön test ve son test öz-yeterlik ortalamaları açısından gruplar arası fark olup olmadığını belirlemek için Mann-Whitney U Testi kullanılmıştır. Elde edilen bulgular Tablo 25’te sunulmuştur.

Tablo 25. Gruplar Arası Öz-Yeterlik Ön Test ve Son Teste İlişkin Mann-Whitney U Sonuçları

Ölçüm Zamanı	Grup	N	M	SD	Mean Rank	Sum of Ranks	Mann-Whitney U	z	p
Ön Test	Fiziksel	10	3.22	1.083	9.45	94.50	39.500	-.795	.427
	Sanal	10	3.70	.65	11.55	115.50			
Son Test	Fiziksel	10	3.74	.795	10.85	108.50	46.500	-.265	.791
	Sanal	10	3.67	.812	10.15	101.50			

* $p < .05$

Tablo 25’e göre, sanal ve fiziksel robotik kodlama gruplarının öz-yeterlik ön test ortalamaları arasında istatistiksel olarak anlamlı bir fark bulunmamıştır ($p > .05$). Bu bulgu, her iki gruba uygulanan robotik kodlama öğretim süreci öncesinde bu iki gruptaki üstün yetenekli

öğrencilerin Python programlama diline yönelik benzer düzeyde öz-yeterliğe sahip olduğunu göstermektedir.

Diğer taraftan sanal ve fiziksel robotik kodlama gruplarının öz-yeterlik son test ortalamaları arasında da istatistiksel olarak anlamlı bir fark bulunmamıştır ($p>.05$). Bu durum, üstün yetenekli öğrencilerin Python programlama dili öğrenme sürecinde sanal veya fiziksel robotik kodlama aracıyla 10 haftalık etkinlik yapmalarının öz-yeterlik seviyelerinde anlamlı bir değişime neden olmadığını göstermektedir.

Üstün Yetenekli Öğrencilerin Sanal Ve Fiziksel Robotik Kodlama Kaygıları Arasında Anlamlı Bir Fark Var Mıdır?

Grup İçi Robotik Kodlama Kaygıları Arasındaki Fark

Her iki grubun, grup içi ön test ve son test robotik kodlama kaygı ortalamaları arasında fark olup olmadığını belirlemek için Wilcoxon İşaretli Sıralar Testi kullanılmıştır. Elde edilen bulgular Tablo 26’da sunulmuştur.

Tablo 26. Grup İçi Kaygı Ön Test ve Son Teste İlişkin Wilcoxon İşaretli Sıralar Testi Sonuçları

Grup		N	Mean Rank	Sum of Ranks	z	p
Fiziksel	Negatif Sıralar	6	4.67	28.00	-.652	.514
	Pozitif Sıralar	3	5.67	17.00		
	Fark olmayan	1				
	Toplam	10				
Sanal	Negatif Sıralar	6	4.50	27.00	-1.262	.207
	Pozitif Sıralar	2	4.50	9.00		
	Fark olmayan	2				
	Toplam	10				

*Pozitif sıralara dayalı ($p<.05$)

Tablo 26’ya göre, her iki grupta da ön test ve son test ortalamaları arasında istatistiksel olarak anlamlı bir fark bulunmamıştır ($p>.05$). Bu durum, hem sanal ve hem fiziksel robotik kodlama araçlarıyla yapılan etkinliklerde üstün yetenekli öğrencilerin Python programlama dili öğrenmeye yönelik kaygılarında anlamlı bir değişim olmadığını göstermektedir.

Grupların Robotik Kodlama Kaygıları Arasındaki Fark

Sanal ve fiziksel robotik kodlama gruplarının ön test ve son test kaygı ortalamaları açısından gruplar arası fark olup olmadığını belirlemek için Mann-Whitney U Testi kullanılmıştır. Elde edilen bulgular Tablo 27’de sunulmuştur.

Tablo 27. Gruplar Arası Kaygı Ön Test ve Son Teste İlişkin Mann-Whitney U Sonuçları

Ölçüm Zamanı	Grup	N	M	SD	Mean Rank	Sum of Ranks	Mann- Whitney U	z	p
Ön Test	Fiziksel	10	1.30	.722	12.00	120.00	35.000	-1.136	.256
	Sanal	10	.94	.609	9.00	90.00			
Son Test	Fiziksel	10	1.21	.808	12.40	124.00	31.000	-1.438	.150
	Sanal	10	.72	.396	8.60	86.00			

*p<.05

Tablo 27'ye göre, sanal ve fiziksel robotik kodlama gruplarının kaygı ön test ortalamaları arasında istatistiksel olarak anlamlı bir fark bulunmamıştır ($p>.05$). Bu bulgu, her iki gruba uygulanan robotik kodlama öğretim süreci öncesinde bu iki gruptaki üstün yetenekli öğrencilerin Python programlama diline yönelik benzer düzeyde kaygıya sahip olduğunu göstermektedir.

Diğer taraftan sanal ve fiziksel robotik kodlama gruplarının kaygı son test ortalamaları arasında da istatistiksel olarak anlamlı bir fark bulunmamıştır ($p>.05$). Bu durum, her ne kadar her iki grupta da son test kaygı ortalamaları ön teste göre düşüş gösterse de, üstün yetenekli öğrencilerin Python programlama dili öğrenme sürecinde sanal veya fiziksel robotik kodlama aracıyla 10 haftalık etkinlik yapmalarının kaygı seviyelerinde anlamlı bir değişime ve düşüşe neden olmadığını göstermektedir.

Üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlama performansları, öz-yeterlikleri ve kaygıları arasında anlamlı bir ilişki var mıdır?

Üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlama başarıları, öz-yeterlikleri ve kaygıları arasında anlamlı bir ilişki olup olmadığını belirlemek için Spearman Sıra Farkları Korelasyon Test kullanılmıştır. Elde edilen bulgular Tablo 28'de sunulmuştur.

Tablo 28. Gruplara Göre Değişkenler Arası İlişkiye Yönelik Spearman Sıra Farkları Korelasyon Test Sonuçları

Grup		Başarı	Öz-Yeterlik	Kaygı
Fiziksel	Başarı	1		
	Öz-Yeterlik	.400	1	
	Kaygı	-.884**	-.491	1
Sanal	Başarı	1		
	Öz-Yeterlik	.395	1	
	Kaygı	-.466	-.213	1

p**<.01

Tablo 28'e göre, fiziksel robotik kodlama grubunda başarı ile kaygı arasında güçlü ve negatif yönlü, istatistiksel olarak anlamlı bir ilişki bulunmuştur ($r=-.884$, $p<.01$). Bu bulgu, fiziksel robotik kodlama aracıyla etkinlik yapan üstün yetenekli öğrencilerin Python programlama diline yönelik başarılarının arttıkça kaygılarının önemli seviyede azaldığını göstermektedir.

Üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlama sürecindeki Python programlama dili öğrenme deneyimlerine ilişkin görüşleri nelerdir?

Üstün yetenekli öğrencilerinin sanal (VEXcode-VR) ve fiziksel (Raspberry-Pi) robotik kodlama etkinliklerindeki öğrenme deneyimlerine ilişkin görüşlerini belirlemek için içerik analizi yapılmıştır. Elde edilen nitel bulgular; “*Öğrenme Çıktısı, Bilişsel Süreçler, Duyuşsal Tepkiler, Davranışsal Süreçler, Temsili/Aracı Katılım, Sosyal Etkileşim*” kategorilerinde toplanmıştır. Bu kategorilerdeki çeşitli temalar altında pek çok kod oluşturulmuştur. Bu kodlara ait frekans değerleri; sanal (VEXcode-VR) ve fiziksel (Raspberry-Pi) robotik kodlama gruplarına ayrılarak Tablo 29’da sunulmuştur. Böylece, hem sanal hem de fiziksel robotik kodlama grubundaki üstün yetenekli öğrencilerin bu etkinliklere ilişkin deneyimleri karşılaştırmalı olarak görülebilmektedir.

Tablo 29. *Üstün Yetenekli Öğrencilerinin Sanal ve Fiziksel Robotik Kodlama Sürecindeki Deneyimleri*

Tema	Kodlar	Fiziksel (Raspberry-Pi)	Sanal (VEXcode-VR)
		f	f
Öğrenme Çıktısı	Python programlama dilini öğrenme	6	9
	Robotik kodlama mantığını anlama	5	5
	Robotik kodlamayı işlevsel ve yaşantısal bağlamda öğrenme	2	1
	Robot bileşenlerini kodlayarak deneyimleme	7	5
	Robotik kodlamayı eğlenerek öğrenme	6	1
	Robotik alanına yönelik mesleki eğilim geliştirme	3	4
	Robotik bilgilerini gelecekteki mesleğinde kullanabileceğini fark etme	4	2
	Yeni bir araç kullanarak robotik kodlama deneyimi kazanma	3	5
	Robotik kodlama becerisini geliştirme	0	1
	Robotik kodlama yapabileceğini kanıtlama	0	3
	Gerçekçi etkinlikler-görevler sayesinde robotik kodlamayı öğrenme	2	0
	Robotik kodlama hatalarından ders çıkarmayı öğrenme	0	2

Tablo 29 (Devamı)

Öğrenmede Yaşanan Zorluklar	Hatalı kodu bulmakta zorlanma	1	5
	Robotik araç seçimi ve bağlantıları kurmada güçlük yaşama	2	1
	Karmaşık robotik kodlama görevlerini tamamlamakta zorlanma	1	1
	Kod yazarken zorlanma	2	2
	Kod yapılarında karışıklık yaşama	1	2
	Kod yapılarını unutma	3	3
	Fonksiyonları kullanmada zorlanma	2	1
	Kodlama sürecinde sözdizimi hatası yapma	6	9
	Kodlama sırasında Türkçe karakter hatası yapma	0	6
Bilişsel Süreçler	Algoritmik akış kurmada zorlanma	2	1
	Zorlu kodlama görevlerinde kararlılık gösterme	3	3
	Robotik problemi çözeceğine/başaracağına inanma	2	3
	Robotik probleme alternatif çözüm yolları geliştirmeyi başarma	0	1
	Deneme-yanılma yoluyla başaracağına inanma	1	3
	Robotik kodlama hatalarını düzeltebileceğine inanma	0	1
	Robotik kodlama yeterliğine yönelik özgüven geliştirme	3	1
	Hedefe yönelik robotik kodlama yapacağına inanma	1	1
	Robotik kodlama konusunda kendini yeterli hissetme	3	4
	Robotik kodlama konusunda kendini yetersiz hissetme	0	3
	Robotik kodlama alanında kendini geliştireceğine inanma	2	1
	Günlük yaşamdaki robotik uygulamalarla ilişkilendirebilme	6	10
	Gerçek hayat için robotik projeler üretebileceğini fark etme	2	6
Duyuşsal Tepkiler	Ne öğreneceğini merak etme	2	5
	İlk İzlenim		
	Zorlanacağını düşünme	1	2
	Yapabileceğine ilişkin tereddüt etme	1	1
	Karmaşık olduğunu düşünme	0	2
	Kodun hatalı çalışma ihtimalinden korkma	1	0
	Hata yapınca endişelenme	1	3
	Robotik problemi çözememe kaygısı yaşama	1	1
	Kaygı Duyma/ Duymama		
	Başaramayacağını düşünme/kendine güvenmeme	2	1
	Karmaşık/Zor kodlarla karşılaşınca endişelenme	0	1
	Genel olarak rahat olma, kaygı duymama	6	4
	Tekrar tekrar denemekten korkmama	0	1
	Sıkılma		
	Hatayı bulup düzeltirken sıkılma	2	0
	Üzülme		
	Başarısız denemelerde olumsuz duygulara kapılıp üzülme	0	2
	Akran başarısı karşısında olumsuz duygusal zorlanma	0	1

Tablo 29 (Devamı)

Eğlenme/Keyif Alma	Robotik kodlama etkinliklerini yaparken eğlenme	4	2	
	Robotik kodlama problemlerini çözmekten keyif alma	5	3	
	Robotik kod doğru çalışınca mutlu olma	2	4	
	Hatalı kodu düzeltince çok mutlu olma	0	2	
	Zor kodlama sürecini tamamlamaktan memnuniyet duyma	1	1	
	Yeni bir robotik araç ile yeni bilgiler öğrendiği için mutlu olma	3	3	
	Hazır robotik sistemle çalışma kolaylığından memnun olma	1	1	
	Yeni robotik bilgileri keşfetme heyecanı yaşama	3	3	
	Zor görev karşısında öğrenme isteğiyle heyecanlanma	1	5	
Davranışsal Süreçler	Kod Yazma/Robotu Çalıştırma	Robotik aracın doğru çalışmasını sağlayan kodu elde edene kadar test yapma	2	2
		Hatalı kodu bulmakta zorluk yaşama	2	5
		Hatalı kodu buluncaya kadar tekrar tekrar deneme yapma	4	6
	Uygun Robotu Seçme/Oluşturma	Uygun robotik aracı kolayca seçme/bağlantıyı oluşturma	2	0
Temsili/Aracı Katılım	Sorumluluğunun Farkında Olma	Sorumluluk bilinciyle etkin katılım sağlama	0	1
	Yardıma İhtiyacı Olduğunu Söyleme	Öğretmene soru sorma	3	1
	Duygu/Düşüncesi ni Dile Getirme	Olumlu duygularını ifade etme	3	6
		Olumsuz duyguları ifade etme	0	1
Sosyal	Yardım İsteme	Öğretmenden yardım isteme	4	5
		Arkadaşlarından yardım isteme	1	1
	Akran Baskısı	Akran baskısı nedeniyle duygusal zorlanma	0	1

Tablo 29'daki nitel bulgular, üstün yetenekli öğrencilerin yalnızca sanal ve fiziksel robotik kodlama etkinliklerindeki öğrenme çıktılarına yönelik görüşlerini değil, aynı zamanda bu süreçteki duygu, düşünce ve davranışlarını da detaylı olarak gözler önüne sermektedir. Devam eden bölümde öğrenci deneyimleri; “Öğrenme Çıktısı, Bilişsel Süreçler, Duyuşsal Tepkiler, Davranışsal Süreçler, Temsili/Aracı Katılım, Sosyal Etkileşim” kategorileri altındaki temalarda yer alan kodlara göre sanal (VEXcode-VR) ve fiziksel (Raspberry-Pi) robotik kodlama gruplarındaki öğrenci ifadelerinden alıntılarla desteklenmiş ve karşılaştırmalı bir şekilde yorumlanmıştır.

Öğrenme Çıktısı

Bu kategoride ortaya çıkan “Eğitsel Katkılar” ve “Öğrenmede Yaşanan Zorluklar” temaları altında çeşitli kodlar belirlenmiştir. Bu kodlar, sanal veya fiziksel robotik kodlama

aracıyla Python programlama dili öğrenme sürecine katılan üstün yetenekli öğrencilerin edindikleri kazanım ve deneyimlerin yanı sıra karşılaştıkları zorlukları da ortaya koymaktadır.

Eğitsel Katkılar: Her iki grupta öğrencilerin Python programlama dilini öğrenme kazanımı edindikleri ve bu yeni araçlarla robot bileşenlerini kodlayarak deneyimleme fırsatı buldukları görülmüştür. Ayrıca bu etkinlikler sayesinde her iki gruptaki bazı öğrencilerin, edindikleri bu robotik bilgilerini gelecekteki mesleklerinde kullanabileceklerini fark ettiği, hatta robotik alanına yönelik mesleki eğilim geliştirdiği belirlenmiştir. Sanal robotik kodlama grubunda, bu etkinlikler sayesinde robotik kodlama mantığını anladığını ve yaptığı robotik kodlama hatalarından ders çıkarmayı öğrendiğini dile getiren öğrenciler fiziksel robotik kodlama grubuna göre çoğunluktadır. Fiziksel robotik kodlama grubunda ise, gerçekçi görev ve senaryolar sayesinde robotik kodlamayı daha kolay öğrendiklerini belirten öğrenciler sanal gruba göre çoğunluktadır. Buna göre her iki gruptaki öğrencilerin bu öğrenme sürecinde eğitsel açıdan çeşitli kazanımlar ve deneyimler elde ettikleri söylenebilir. Bu temadaki kodlara ilişkin öğrenci ifadelerinden alıntılar şunlardır:

“İlk haftada kodların nasıl çalıştığını öğrendik, listeler ve sürekli tekrarlar kodlarını öğrendik. Mesela For kodunu yani [kodu] kaç kez tekrarlayacağını öğrendik. Bi de mesela robotun hareket etme ve renk sensörlerini çalıştırma mantığını öğreniyorsun. Robotu hareket ettiriyorsun, renk sensörünü çalıştırıyorsun, bu çok güzel, renk sensörüyle harfler yazabiliyorduk (Sanal, Ö3, Erkek).”

“Hocam, sensörleri öğrendim. Hocam yazdığım kodların robota nasıl bir etkisi olduğunu robotun hareketlerinde görüyorsun (Sanal, Ö4, Erkek).”

“Mesela While True var onu öğrendik. O da döngü, kodu sürekli yazmak yerine döngü yapmak için kullanılıyor. Akıllı ev sistemi gibi şeyler yaptık. Mesela LED karanlıkta, ışık olmadığında ışık yakıyordu, ışık olduğunda ışığı söndürüyordu. Mesela ben daha önceden hiç bilmiyordum servo motorunu, bunları da öğrendim (Fiziksel, Ö2, Erkek).”

“Şey hocam, While döngüsüyle For döngüsünü öğrendim. Yazdığım kodu çalıştırabilmek, kurduğum devreyi çalıştırabilmek falan bilmediğim şeyler öğendim. Led yakmayı, motor çalıştırmayı da öğrendim hocam (Fiziksel, Ö3, Kız).”

“Ledler'in artı eksisine göre bağlanmayı öğrendik. Aklıma gelen, mesela tuşa basılırsa lambayı yak gibi cümleler kullandığımızda bunu gerçek hayatta da kullandığımızı fark ettim, sonra bununla ilgili projeler yapıp bunu da gerçek hayatımıza geçirebiliriz onu fark ettim (Fiziksel, Ö1, Erkek).”

“İlk derse geldiğimde çok hata yapmışım. Ama bazen hatalarım yeni sonuçlar elde etmeme, yeni bilgiler öğrenmeme yardımcı oluyor. Örneğin def main fonksiyonunu her kodlamanın başına yazmamız gerektiğini, if koşulunu falan, tekrarlamaları, sonra hocam if-else koşulunu öğrendim. Robotun altıgen çizirken ki ilerleme mesafesini filan robotik kodlama dersi sayesinde öğrendik (Sanal, Ö1, Kız).”

“Hocam ben birinci önceliğim cerrah olmayı düşünüyorum. Cerrahide robotik sistemler, sensörler, x-ray'ler falan işimize yarıyor. Yani cerrahide kullanılacak bilgisayar sistemleriyle alakalı bana katkı sağladığını düşünüyorum (Sanal, Ö2, Erkek).”

“...ben gelecekte yazılım mühendisi olmak istiyorum, yazılım mühendisi ile robotik kodlamanın çok alakası olduğunu düşünüyorum. Bu da bana çok yardım edecek, gerçek ortamda yapacağım bir denemenin bana daha çok katkısının olduğunu düşünüyorum (Fiziksel, Ö1, Erkek).”

Öğrenmede Yaşanan Zorluklar: Her ne kadar hem sanal hem de fiziksel robotik kodlama grubundaki az sayıda öğrencinin robotik kodlama görevlerini tamamlamakta ve kod yazmakta zorlandığı görülsede, öğrencilerin çoğunlukla kod yapılarını unuttuğu saptanmıştır. Sanal robotik kodlama grubundaki öğrencilerin kodlama sürecinde sıklıkla söz dizimi hatası yaptıkları ön plana çıkmıştır. Ayrıca bu grupta Türkçe karakter kullanma ve hatalı kodu bulmakta zorlanma durumuyla daha çok karşılaşmıştır. Fiziksel robotik kodlama grubunda ise; uygun robotik aracı seçmekte veya bağlantıyı kurmakta zorlanan (örneğin LED bağlama gibi işlemler) öğrencilerin daha çok olduğu gözlemlenmiştir. Ayrıca bu grupta fonksiyonları kullanırken zorlanma durumuyla daha çok karşılaşmıştır. Bu bulgular, her iki gruptaki öğrencilerin ilk kez bu çalışma kapsamında Python programlama dilini kullanarak robotik kodlama yapmaya başlaması nedeniyle, gerçekçi ve dijital bir aracı programlama alışkanlıklarının yetersizliğine işaret edebilir. Bu temadaki kodlara ilişkin öğrenci ifadelerinden alıntılar şunlardır:

“Hatalarımı ararken uzun zaman geçiyordu en zorlandığın konu hataları bulmaktı. Yani boşluğu yazmadığımızda hata çıkıyor. Sadece bazen iki noktayı yanlış yazıyordum ya da boşlukları yanlış yazıyordum. Bazen de sadece kodların yerlerini karıştırıyordum. O kadar (Sanal, Ö2, Erkek).”

“Hataları bulmak zordu o yüzden en baştan tekrardan bakıyordum satır satır o zor oluyordu. Hocam mesela kodu doğru yazmıştım ama boşluğu bırakmadığımda hata çıkıyordu. Yazım hatası yaptığımı düşünüyordum ya da boşluk bırakmayı unutuyordum. (Sanal, Ö3, Erkek).”

“Sevmediğim bir şey ise mesela yazarken yanlış çıkıyor, hata çıkıyor. Hatayı bulmak uzun zaman alıyor 10 saat onu bulmaya çalışıyordum (Fiziksel, Ö2, Erkek).”

“Hocam tabii de Türkçe karakterler herhalde hocam, bazen hatayla Türkçe karakterler kullanabiliyordum hocam, Onları çözmek zor oluyordu. (Sanal, Ö4, Erkek).”

“Ama bazen gerçekten VEXcode-VR program bağlantı hatası verebiliyor. Ayrıca kod yazarken de iki kere üç kere zorlandım. Fonksiyon yazarken bir tab ötede yapıyoruz ben boşlukları nedense sürekli atlıyordum bazen bir tabı biraz atlıyordum veya yazım hatalarını yapıyordum (Sanal, Ö1, Kız).”

“Kablolar çok karışık oluyor. Robotik parçaları kurmak, birleştirmek daha çok zor... Mesela hocam kodları yazmak hepsi alt alta ya, işte onda biraz sıkıntı çıkıyor. Kolay geldi bana ama kod yazarken bazen unutabiliyordum (Fiziksel, Ö2, Erkek).”

“Hocam unuttuğum kodlar oluyordu. Kod yazarken okuduğum kodun devamında ne geleceğini çok düşündüm (Sanal, Ö4, Erkek).”

“Hocam kodları yazmak işte onda biraz sıkıntı çıkıyor (Fiziksel, Ö2, Erkek).”

“Hocam kodları yazmak zor oldu. for yanlış olursa program çalışmıyor biliyordum. Kodlama yazımında sıralamayı da bazen karıştırıyordum (Fiziksel, Ö3, Kız).”

“Fonksiyonlar bazen zor olabiliyor, o nedenle fonksiyonları kullanmakta zorlandığım kısımlar oldu (Fiziksel, Ö1, Erkek).”

Bilişsel Süreçler

Bu kategoride ortaya çıkan, “Öz-Yeterlik ve Problem Çözme Durumu”, “Gerçek Hayatla İlişkilendirme” temaları altında çeşitli kodlar belirlenmiştir. Bu kodlar, üstün yetenekli öğrencilerin sanal veya fiziksel ortamlarda yapılan etkinliklerdeki robotik kodlamaya yönelik problem senaryolarına çözüm üretme durumlarının, görevleri tamamlama kararlılıklarının yanı sıra, etkinlikleri gerçek hayatla ilişkilendirebilme durumlarını, başarıma inançlarını ve özgüvenlerini ortaya koymaktadır.

Öz-Yeterlik ve Problem Çözme: Her iki grupta da aynı sayıda öğrenci, zorlu kodlama görevlerinde kararlılık gösterdiğini dile getirmiştir. Bu bulgu, öğrencilerin problem çözme sürecine aktif şekilde katıldığını ve çaba göstermeye istekli olduklarını ortaya koymaktadır. Ayrıca robotik problemi çözeceğine/başaracağına inanan ve robotik kodlama konusunda kendini yeterli hisseden öğrenci sayısı da her iki grupta birbirine yakın seviyededir. Ancak robotik kodlama yeterliğine yönelik özgüven geliştirdiğini net olarak dile getiren öğrenci sayısı fiziksel robotik kodlama grubunda daha yüksek düzeydedir. Bu durum, her iki gruptaki öğrencilerin genel olarak robotik kodlamaya yönelik olumlu bir öz-yeterlik algısına sahip olduklarını göstermekle birlikte, fiziksel robotik kodlama grubunda daha belirgin seviyede olduğunu ortaya koymaktadır.

Diğer taraftan hem problem çözümünü deneme-yanılma yoluyla başaracağına inanan hem de kendini robotik kodlama konusunda yeterli hissetmeyen öğrenci sayısı sanal robotik kodlama grubunda daha fazladır. Aynı grupta ortaya çıkan bu iki zıt bulgu, öğrencilerin ilk kez sanal bir ortamda robotik kodlama etkinliklerine katılım göstermelerinin bir sonucu olabilir. Robotik probleme çözüm bulma konusunda alternatif yollar üretme ve hata ayıklama yapabileceğine inanma durumu ise yalnızca sanal robotik kodlama grubunda ve düşük düzeyde dile getirilmiştir. Bu durum, öğrencilerin bu becerileri gerçekleştirmede veya ifade etmede sınırlı kaldıklarını, bazı bilişsel stratejileri (ör. hata ayıklama, alternatif çözüm üretme) kullanma konusunda daha fazla destek ve deneyime ihtiyaç duyduklarını göstermektedir. Bu temadaki kodlara ilişkin öğrenci ifadelerinden alıntılar şunlardır:

“Ben zaten zihnimde canlandırmaya çalışıyordum ama fazla da zor olmuyordu (Sanal, Ö2, Erkek).”

“Çok şey yapabildiğin bu etkinlikler, başarı hissini veriyor (Sanal, Ö1, Kız).”

“Zor oluyor hatayı bulmak istiyorsun. Yani bir şeyler seni zorlayınca daha da üstüne gitmek istiyorsun. Onunla daha fazla uğraşıyorum (Fiziksel, Ö2, Erkek).”

“Hocam, hata olunca yine baştan farklı bakıp, farklı çözüm buluyordum (Sanal, Ö4, Erkek).”

“Hocam başlarda kötüydüm ama sonrasında hallettim. Çünkü kodu yazıp buradan tekrar tekrar yapınca deneme yanılmayla çözdüm. Son derslerde böyle yaptım (Sanal, Ö2, Erkek).”

“Mesela kendim de istesem program yapabilirim artık. Kendimi yeterli hissediyorum, çünkü şimdi bunun çok daha kolay olduğunu öğrendim robotik kodlamada (Fiziksel, Ö1, Erkek).”

“Robotu görüp, yani yazdığın kodun doğru olduğunu görünce yapabildiğini görüyorsun. Son zamanlarda yazdığımız kodlarda daha iyi olmaya başladım. O nedenle hocam robotik kodlamada kendimi yeterli hissediyorum bence (Sanal, Ö2, Erkek).”

“Yeterli hissediyorum. Mesela kendim de istesem program yapabilirim artık (Fiziksel, Ö1, Erkek).”

“Ben zaten kodu yazarken yazımı geliştirdim, mesela genellikle tamamlıyorum o kodları. O nedenle kendimi yeterli hissediyorum (Fiziksel, Ö2, Erkek).”

“Hocam ben robotik kodlamada kendimi fazla yeterli hissetmiyorum. Çünkü sadece bu derste temelini attım. Daha çok ilerletirsem daha güzel sonuçlar elde edeceğimi düşünüyorum (Sanal, Ö1, Kız).”

“Son zamanlarda yazdığımız kodlara daha iyi olmaya başladım. Hatta sizin yaptığınız testlere de ilk baştakine göre çok az yanlışım çıktı (Sanal, Ö2, Erkek).”

Gerçek Hayatla İlişkilendirme: Her iki grupta da etkinlikleri günlük yaşamdaki robotik uygulamalarla ilişkilendiren ve gerçek hayat için robotik projeler üretebileceğini fark eden öğrenciler mevcuttur. Bu bulgu, öğrencilerin robotik etkinliklerde kazandıkları bilgileri günlük yaşam bağlamına transfer edebildiklerini göstermektedir. Ancak bu oran fiziksel robotik kodlama grubuna göre sanal robotik kodlama grubunda oldukça yüksektir. Bu durum ise, sanal ve fiziksel ortamlarda kullanılan robotik kodlama araçlarının teknik özelliklerinden kaynaklanmış olabileceği gibi, sanal robotik kodlama etkinliklerinin öğrencilerin bilgiyi gerçek yaşamla ilişkilendirmelerine daha fazla katkı sunduğunu ortaya koymaktadır. Bu temadaki kodlara ilişkin öğrenci ifadelerinden alıntılar şunlardır:

“Mesela araba kazalarını önlemek için kullanılıyor sensörler ya da araçların bir şeye çarpmasını sensör önüyor (Sanal, Ö2, Erkek).”

“Hocam mesela biz elimizi kullanarak yemek yiyoruz, robotta kadlayarak yazı yazabiliyor ya da resim çizebiliyor. Hocam mesela ben bir kod yazdığımda o istediğim şekilde ilerleyebiliyor, böylece çizdiğim şekillerin iç açılarının kaç derece olduğunu da öğretiyor (Sanal, Ö3, Erkek).”

“Mesela diyelim, bir arabada sensör oluyor o araba kullanırken sorun çıkardı sensör hemen bize bildiriyor (Sanal, Ö4, Erkek).”

“Raspberry-Pi robotuna sensörler falan bağladık. Düşünün işte hocam mesela sensörler günlük hayatta arabalarda çok kullanılıyor kazaları önlemek için, ya da çoğu farklı işte bu akıllı araçlar ev sistemleri kullanılıyor. Bu akıllı ev sistemini kurmuştuk işte o evlerin geliştirilmesi çok önemli mesela (Fiziksel, Ö2, Erkek).”

“Mesela eğer yangın dumanı gelirse Raspberry-Pi bunu algılar ve robottan ses çıkar. Bu sayede de bunun yangın olduğunu anlamamızı sağlıyor (Fiziksel, Ö3, Kız).”

Duyuşsal Tepkiler

Bu kategoride ortaya çıkan, “İlk İzlenim”, “Kaygı Duyma/Duymama”, “Sıkılma”, “Üzülme”, “Eğlenme/Keyif Alma”, “Mutlu Olma”, “Heyecanlanma” temaları altında çeşitli kodlar belirlenmiştir. Bu kodlar, üstün yetenekli öğrencilerin sanal veya fiziksel robotik kodlama araçlarıyla karşılaştıklarında hangi duygular içinde olduklarını, bu araçlarla katıldıkları Python programlama dili öğrenme sürecindeki kaygı durumlarını, olumlu/olumsuz duygusal tepkilerini ortaya koymaktadır.

İlk İzlenim: Sanal robotik kodlama grubundaki öğrencilerin bu etkinliklerle ilk karşılaştıklarında fiziksel robotik kodlama grubundakilere göre ne öğreneceklerini daha fazla merak etmelerine karşın, yalnızca sanal robotik kodlama grubundaki bazı öğrencilerin ilk izlenim olarak bu etkinliklerin karmaşık olduğunu düşünmeleri şaşırtıcıdır. Bu durum, özellikle sanal robotik kodlama aracının öğrencilerin ilk kez katıldıkları bu etkinliklere karşı daha yüksek bir merak duygusu taşımalarını sağladığını, ancak bu ortamdaki robotik kodlama etkinliklerini başlangıçta zihinsel olarak zorlayıcı bulmalarına neden olduğunu ortaya koymaktadır. Bu temadaki kodlara ilişkin öğrenci ifadelerinden alıntılar şunlardır:

“Hocam ilk gördüğümde acaba biz bu robotla ne yapacağız diye düşündüm robotun üstündeki kalemi el sandım belki o elle bir şeyler tutup bırakırsınız diye düşünmüştüm (Sanal, Ö2, Erkek).”

“Hocam, ben tabii ki çok merak ettim dedim ki, acaba biz bu uygulamada ne yapacağız? (Sanal, Ö4, Erkek).”

“Hocam ilk başta yapabilir miyim diye birazcık merak ettim (Fiziksel, Ö1, Erkek).”

“Hocam çok karmaşık gibi duruyordu. Ben ilk geldiğimde gerçekten yapamayacağımı, zor olduğunu düşünmüştüm (Sanal, Ö1, Kız).”

Kaygı Duyma/Duymama: Sanal robotik kodlama grubunda hata yapınca endişeye kapılan öğrenciler çoğunlukta iken, fiziksel robotik kodlama grubunda ise; her ne kadar başaramayacağını düşünen/kendine güvenmeyen öğrenciler olsa da, etkinliklerde genel olarak rahat olan, kaygı duymayan öğrenciler çoğunluktadır. Bu bulgu, özellikle fiziksel robotik kodlama etkinliklerinin sanal robotik kodlama etkinliklerine göre öğrencilerin sürece daha rahat yaklaşmalarını ve önemli bir kısmının kaygı hissetmeden etkinliklere katılmalarını sağladığını göstermektedir. Buna karşın sanal robotik kodlama etkinliklerinin öğrencilerin hata yapma sonrası daha fazla endişe yaşamalarına neden olduğunu da ortaya koymaktadır. Bu temadaki kodlara ilişkin öğrenci ifadelerinden alıntılar şunlardır:

“Hatalar! Bu hataları bulurken endişelendim (Sanal, Ö1, Kız).”

“Yanlış yaptığımda biraz daha tedirginleşiyorum (Sanal, Ö3, Erkek).”

“Kodumda hata çıkabileceğinden korkuyorum (Fiziksel, Ö1, Erkek).”

“Ara sıra [başarabileceğime dair] güvensizlik hissediyordum hocam (Sanal, Ö1, Kız).”

“Hocam, ilk geldiğimde böyle acaba yapabilir miyim diye tereddüdüm oldu (Fiziksel, Ö3, Kız).”

“Şimdi Raspberry-Pi’de kodlama yaparken hiç özgüvensiz olmuyorum, her zaman kendimi rahat hissediyorum, yani öyle çok rahat da değil ama öğretmenlerimin ve arkadaşlarımdan kızmasından da korkmuyorum (Fiziksel, Ö1, Erkek).”

“Ben genellikle kod yazarken veya Raspberry-Pi ile uğraşırken rahat oluyorum. Hiç kendimi germiyorum (Fiziksel, Ö2, Erkek).”

“Kendimi rahat hissediyorum genel olarak öyleyim, çünkü eğer anlamasam bile sıkıntı olmaz (Fiziksel, Ö3, Kız).”

“Ben kodları çok severek yazdım (Sanal, Ö1, Kız).”

“Hocam açıkçası kendimi rahat hissediyorum, çünkü yanlış yapınca yanlış bulacağımı ve sizin yardım edeceğinizi biliyorum (Sanal, Ö2, Erkek).”

Sıkılma: Fiziksel robotik kodlama grubundaki bazı öğrenciler hataları bulup düzeltme sürecinde sıkıldığını belirtmiştir. Buna karşılık, sanal robotik kodlama grubundaki öğrenciler bu duruma ilişkin herhangi bir olumsuz duygu bildirmemiştir. Bu bulgu, fiziksel ortamda yapılan robotik kodlama etkinliklerindeki hata düzeltme sürecinin bazı öğrenciler için motivasyon kaybına neden olabileceğini, sanal ortamda yapılan robotik kodlama etkinliklerinde ise böyle bir duygu durumunun yaşanma ihtimalinin daha düşük olduğunu göstermektedir. Bu temadaki kodlara ilişkin öğrenci ifadelerinden alıntılar şunlardır:

“Kodumda hata çıktığında o hatayı bulmak uzun sürebiliyor, sıkılıyorum (Fiziksel, Ö1, Erkek).”

“Sevmediğim bir şey ise mesela yazarken yanlış çıkıyor, hata çıkıyor. 10 saat onu bulmaya çalışıyordum, çok sıkıcı (Fiziksel, Ö2, Erkek).”

Üzülme: Fiziksel robotik kodlama grubundaki öğrencilerden hiçbiri başarısız denemelerde veya akran başarısı karşısında olumsuz duygular yaşadığını belirtmemiştir. Buna karşılık, sanal robotik kodlama grubundaki öğrencilerden bazıları başarısız denemelerde olumsuz duygular yaşadığını, akran başarısı karşısında duygusal olarak zorlandığını ifade etmiştir. Bu bulgular, sanal ortamda gerçekleştirilen robotik kodlama etkinliklerinde bazı öğrencilerin başarısızlık veya akran karşılaştırmalarına daha duyarlı olabileceğini göstermektedir. Fiziksel grupta bu tür olumsuz duyguların hiç raporlanmamış olması, yüz yüze ortamın destekleyici etkisi veya duygusal zorlanmaların daha az fark edilmesi ile açıklanabilir. Bu temadaki kodlara ilişkin öğrenci ifadelerinden alıntılar şunlardır:

“Ben programda hata yapınca biraz üzülüüyordum (Sanal, Ö1, Kız).”

“Mutsuz olduğum zamanlar, doğru programı yazdığımı sanıp programın yanlış olması, o sırada diğer arkadaşlarımdan doğru olması (Sanal, Ö3, Erkek).”

Eğlenme/Keyif Alma: Fiziksel robotik kodlama grubunda, etkinlikleri yaparken eğlendiğini, robotik problemleri çözmekten keyif aldığını belirten öğrenciler, sanal robotik kodlama grubuna göre çoğunluktadır. Bu sonuç, fiziksel ortamdaki etkinliklerin Python programlama dili öğrenme sürecinde öğrencilerin olumlu duygularını daha çok desteklediğini göstermektedir. Bu durum, yüz yüze etkileşimlerin ve fiziksel materyallerle çalışma fırsatlarının, öğrencilerin duygusal katılımını artırabileceğini düşündürmektedir. Bu temadaki kodlara ilişkin öğrenci ifadelerinden alıntılar şunlardır:

“Çok güzel. Raspberry-Pi eğlenceli geldi (Fiziksel, Ö1, Erkek).”

“Bunu çok sevdim. Güzel bir şey robotik kodlama. Ben de yapmaya devam etmek istiyorum yani (Fiziksel, Ö3, Kız).”

“Kodlama yaparken zevk aldım, çünkü hem artık kodlama yazabildiğimi kendime kanıtlamış oldum (Fiziksel, Ö1, Erkek).”

“Hocam bence robotik kodlama eğlenceli (Sanal, Ö1, Kız).”

Mutlu Olma: Sanal robotik kodlama grubunda, robotik kod doğru çalışınca mutlu olduğunu dile getiren öğrenciler fiziksel robotik kodlama grubuna göre çoğunluktadır. Ancak yalnızca sanal robotik kodlama grubundaki bazı öğrencilerin, hatalı kodu düzeltince çok mutlu olduğunu dile getirmesi, özellikle sanal ortamda yapılan robotik kodlama etkinliklerinin öğrencilerin başarıya ulaşma durumunda olumlu duygular yaşamalarına daha fazla katkı sağladığını göstermektedir. Diğer taraftan, her iki grupta da aynı sayıda öğrenci, yeni bir robotik araç ile yeni bilgiler öğrendiği için mutlu olduğunu dile getirmişti. Bu durum öğrencilerin yalnızca sanal veya fiziksel ortamdaki robotik aracın çalışmasıyla değil, sürece dair öğrenmelerle de içsel doyum yaşadığını ortaya koymaktadır. Katılımcı profilinin üstün yetenekli öğrencilerden oluşması da bu durumu sağlamış olabilir. Az sayıda da olsa her iki grupta, zor kodlama sürecini tamamlamaktan ve hazır robotik sistemle çalışma kolaylığından duyduğu memnuniyeti dile getiren öğrenciler olmuştur. Bu sonuç, öğrencilerin zorlayıcı süreçlerden çok doğrudan başarıya ulaştıkları anlara odaklandıklarını düşündürülebilir. Bu temadaki kodlara ilişkin öğrenci ifadelerinden alıntılar şunlardır:

“Yaptığım programın başarılı olması beni mutlu ediyordu (Sanal, Ö3, Erkek).”

“Hocam her zaman kodlar beni mutlu etti. Sorunları çözmenin çok faydalı olduğunu, çok güzel bir konu olduğunu anladım (Sanal, Ö1, Kız).”

“O kodu çalıştırdığımda bana mutluluk veriyor (Fiziksel, Ö1, Erkek).”

“En çok beni mutlu ediyordu ledi yakmak bile (Fiziksel, Ö3, Kız).”

“Ama eğer çalışmazsa hatalarımı bulup düzeltiyorum bu sefer daha da mutlu oluyorum (Sanal, Ö1, Kız).”

“Yeni bir araç gördüğüm için daha da mutlu oldum. Onun nasıl çalıştığını ve çalışma biçimini öğrendim. Bunu öğrendiğim için de sevindim (Fiziksel, Ö2, Erkek).”

“Robotu ilk gördüğümde düzenli bir biçimi vardı çok güzel görünüyordu (Sanal, Ö4, Erkek).”

Heyecanlanma: Sanal robotik kodlama grubunda, zor bir görevle karşılaştıklarında öğrenmeye yönelik heyecan duyduklarını ifade eden öğrenciler, fiziksel robotik kodlama grubuna göre çoğunluktadır. Bu durum, sanal ortamda gerçekleştirilen robotik kodlama etkinliklerinin öğrencilerde merak ve öğrenme isteği uyandırmada daha etkili olduğunu göstermektedir. Diğer taraftan her iki grupta da aynı sayıda öğrenci yeni robotik bilgileri keşfetme sürecinden duydukları heyecanı dile getirmiştir. Bu sonuç ister sanal ister fiziksel ortamda olsun, yeni bir robotik araçla Python programlama dilini keşfetme sürecinin öğrencilerde heyecan uyandırdığını göstermektedir. Bu temadaki kodlara ilişkin öğrenci ifadelerinden alıntılar şunlardır:

“Hocam bu yeni robotla kodlamayı öğrenmek için çok heyecanlıydım (Sanal, Ö3, Erkek).”

“Aslında yeni şeyler öğreneceğim için heyecanlandım. Bir de kod daha zor olduğunda daha da heyecanlanıyorum. Heyecanlanmak olumlu etkiliyor beni, çünkü heyecanlanınca daha da yapmak istiyorum (Fiziksel, Ö2, Erkek).”

“Nasıl bulurum diye düşünüyorum ama hocam o heyecanla düşünüyorum (Sanal, Ö2, Erkek).”

“Kodlar çok güzel ama karmaşık gözüküyordu. Yani kodları görünce şaşırmıştım. Bundan sonra kodları öğrenince daha da heyecanlanmaya başladım (Sanal, Ö3, Erkek).”

Davranışsal Süreçler

Bu kategoride ortaya çıkan, “Kod Yazma/Robotu Çalıştırma”, “Uygun Robotu Seçme/Oluşturma” temaları altında çeşitli kodlar belirlenmiştir. Bu kodlar, üstün yetenekli öğrencilerin sanal veya fiziksel robotik araçlarla problem durumuna çözüm üreten kodlama görevlerini yerine getirirken sergiledikleri davranışları ortaya koymaktadır.

Kod Yazma/Robotu Çalıştırma: Her iki grupta da aynı sayıda öğrenci, robotik aracın doğru çalışmasını sağlayan kodu elde edene kadar test yaptığını dile getirmiştir. Bu sonuç ister sanal ister fiziksel ortamda robotik kodlama etkinliğine katılsın, her iki gruptaki öğrencilerin de problem çözme sürecinde kararlılık gösterdiklerine işaret etmekte, diğer bir ifadeyle doğru sonuca ulaşıp görevi tamamlama konusundaki kararlı davranışlarını gözler önüne sermektedir. Bu durum katılımcı profilinin üstün yetenekli öğrencilerden oluşmasından kaynaklanmış olabilir. Buna karşın, sanal robotik kodlama grubunda, hatalı kodu bulmakta zorluk yaşayan öğrenciler fiziksel robotik kodlama grubuna göre çoğunluktadır. Bu bulgu, sanal ortamda robotik kodlama yapan öğrencilerin hata ayıklama sürecinde daha fazla güçlük yaşadıklarını düşündürülebilir. Bu temadaki kodlara ilişkin öğrenci ifadelerinden alıntılar şunlardır:

“Genellikle kodda hatalar çıkıyordu. Onları da kendim bulmaya çalışıyordum (Fiziksel, Ö2, Erkek).”

“Ben de deneyerekten doğru sonucu buluyordum (Fiziksel, Ö3, Kız).”

“Boşluk ya da yazım hatası varsa düzeltiyordum (Sanal, Ö3, Erkek).”

“Hocam kodu çalıştırdığımda aşağıda hatanın nerede yaptığını gösteriyordu, onu düzeltiyordum (Sanal, Ö3, Erkek).”

“Bu robotta bazen, hatanız var diyor ama bazen söylemiyor. Çünkü o kodda bir hata yoktur. Sadece robotun gideceği mesafesini yanlış yapıyorsun (Sanal, Ö1, Kız).”

“Sadece bazen iki noktayı yanlış yazıyordum ya da boşlukları yanlış yazıyordum, o hatayı bulmak çok zor oluyordu (Sanal, Ö2, Erkek).”

“Hocam, herhalde en zoru Türkçe karakter hatası yapmak, çünkü onları çözmek zor oluyordu (Sanal, Ö4, Erkek).”

“Kodumda hata çıktığında o hatayı bulmak uzun sürebiliyor (Fiziksel, Ö1, Erkek).”

Uygun Robotu Seçme/Oluşturma: Fiziksel robotik kodlama grubundaki bazı öğrenciler uygun robotik aracı kolayca seçtiğini/bağlantıyı oluşturduğunu ifade etmiştir. Buna karşın sanal robotik kodlama grubundaki öğrenciler bu doğrultuda herhangi bir beyanda bulunmamıştır. Bu bulgu, fiziksel ortamda robotik araçlarla doğrudan etkileşim kuran öğrencilerin, uygun aracı seçme ve bağlantı kurma süreçlerini daha somut ve anlaşılır biçimde deneyimlediklerini göstermektedir. Diğer yandan, sanal robotik kodlama grubu katılımcılarının bu süreçte dair ifade belirtmemesi, çevrim içi ortamda robotik araçlarla fiziksel temasın olmamasının, uygun robotu belirleme ve bağlantı kurma süreçlerinin fark edilmesini zorlaştırabileceğini düşündürmektedir. Bu durum, uygulamalı robotik etkinliklerde fiziksel ortamın tercih edilmesinin, öğrencilerin araç seçimi ve bağlantı kurma gibi teknik becerileri geliştirmeleri açısından destekleyici bir rol oynayabileceğini ortaya koymaktadır. Bu temadaki kodlara ilişkin öğrenci ifadelerinden alıntılar şunlardır:

“Ben robotik kurmakta zorlanmadım, doğru parçaları, devreleri kolayca bir araya getirdim (Fiziksel, Ö3, Kız).”

“Raspberry-Pi eğlenceliydi, çok zorlanmadım kabloları uygun şekilde bağlarken (Fiziksel, Ö1, Erkek).”

Temsili/Aracı Katılım

Bu kategoride ortaya çıkan, “Sorumluluğunun Fakında Olma”, “Yardıma İhtiyacı Olduğunu Söyleme”, “Duygu/Düşüncesini Dile Getirme” temaları altında çeşitli kodlar belirlenmiştir. Bu kodlar, üstün yetenekli öğrencilerin sanal veya fiziksel robotik kodlama görevlerini yerine getirirken sorumluluk bilinciyle etkin katılım sağlayıp sağlamama, yardıma ihtiyacı olup olmama durumlarına ilişkin duygu ve düşüncelerini ne derece dile getirdiklerini ortaya koymaktadır.

Sorumluluğunun Farkında Olma: Her ne kadar robotik kodlama sürecine katılan her iki gruptaki öğrenciler, nicel bulgulara ortaya çıktığı üzere, tüm görevleri tamamlama performansı sergilemiş olsalar da, yalnızca sanal robotik kodlama grubundaki bir öğrencinin sorumluluk bilinciyle etkin katılım sağladığını dile getirmesi şaşırtıcıdır. Fiziksel robotik kodlama grubunda ise bu kod hiç gözlenmemiştir. Bu durum, sanal ortamda görev bilinciyle hareket ederek bireysel katılım sergileyen öğrencilerin sınırlı da olsa varlık gösterdiğini ortaya koymaktadır. Söz konusu bulgu ister sanal ister fiziksel ortamda olsun, öğrencilerin robotik etkinliklerde sorumluluk alma davranışlarını doğrudan dile getirme eğiliminde olmadıklarını ve bu becerinin görünür kılınması için daha fazla yapılandırılmış yönlendirme ve görev paylaşımına ihtiyaç duyulabileceğini düşündürmektedir. Bu temadaki kodlara ilişkin öğrenci ifadelerinden alıntılar şunlardır:

“Kötü düşüncelerimden uzak kaldım, etkinlikte istenileni düşündüm, ona uygun şekilde doğru sonuca ulaşmak için odaklandım (Sanal, Ö1, Kız).”

Yardıma İhtiyacı Olduğunu Söyleme: Fiziksel robotik kodlama grubunda etkinlikleri doğru şekilde tamamlamak için yardıma ihtiyacı olduğunda öğretmene çekinmeden soru sorduğunu dile getiren öğrenci sayısı, sanal robotik kodlama grubundakilere göre çoğunluktadır. Her iki gruptaki robotik etkinliklerde de öğretmenin öğrencilerle aynı laboratuvar ortamında bulunduğu ve öğrencilerin öğretmene doğrudan erişiminin sağlandığı düşünüldüğünde bu bulgu şaşırtıcıdır. Bu durum, fiziksel ortamdaki robotik araçla gerçekleştirilen somut/dokunmatik etkileşimin, öğrencilerin bulundukları öğrenme çevresinin gerçekliğinin farkında olmalarını sağlayarak, yardım isteme davranışlarını daha rahat sergilediklerini göstermektedir. Diğer yandan, sanal ortamdaki robotik aracı kullanırken öğrencilerin bu çevrimiçi ve 3B ortamın içine dalıp (immersion), bulundukları öğrenme çevresinin gerçekliğini unutmalarına neden olarak, yardım isteme davranışlarını daha sınırlı düzeyde sergilediklerini düşündürmektedir. Bu temadaki kodlara ilişkin öğrenci ifadelerinden alıntılar şunlardır:

“Benim kodlarda bazen yanlışlık çıkıyordu. Onu nasıl çözeceğimi bilmiyordum ama size sorup onu da halledebiliyordum (Fiziksel, Ö1, Erkek).”

“Bulamayınca da zaten size soruyordum. Mesela akıllı ev sisteminde size de sordum (Fiziksel, Ö2, Erkek).”

“Bir kere size sormuştum, siz bana yanlışımı gösterdiniz (Sanal, Ö1, Kız).”

Duygu/Düşüncelerini Dile Getirme: Sanal robotik kodlama grubunda robotik kodlama süreciyle ilgili olumlu duygularını ifade eden öğrencilerin sayısı fiziksel robotik kodlama grubundakilere göre çoğunluktadır. Bu bulgu, her iki gruptaki öğrencilerin de olumlu duygu/düşüncelerini ifade etmeye yatkın olduğunu, özellikle sanal robotik kodlama grubunda öğrencilerin bu tür duygu/düşünceleri dile getirmekte daha belirgin bir şekilde öne çıktığını

göstermektedir. Öte yandan, olumsuz duygu/düşünce ifadesi yalnızca sanal robotik kodlama grubunda bir kez gözlemlenmiştir. Olumsuz duygu/düşünce paylaşımının her iki grupta da düşük düzeyde kalması, öğrencilerin negatif duygu/düşünceleri dile getirme konusunda çekingen olabileceklerine ya da bu duygu/düşüncelerin yeterince fark edilmemiş olabileceğine işaret etmektedir. Bu çerçevede gerek sanal gerekse fiziksel robotik kodlama gibi problem çözme temelli öğrenme ortamlarında öğrencilerin duygu/düşüncelerini açıkça ifade etmelerini destekleyici iletişim ortamlarının oluşturulması; duygu/düşüncelerine yönelik farkındalık ve düzenleme becerilerinin gelişimi açısından önemli olabilir. Bu temadaki kodlara ilişkin öğrenci ifadelerinden alıntılar şunlardır:

“Sizinle daha çok kendine güveniyor insan. Sizin yardımınızla bu tür etkinliklerde kendime güvenmemeyi atlattım. Süpersiniz (Sanal, Ö1, Kız).”

“Gittikçe aramıza bir bağ oluştu hocam. VEXcode-VR ile aramıza güzel bir bağ oluştu (Sanal, Ö4, Erkek).”

“...Yeni bir araç gördüğüm için daha da mutlu oldum. Bunu öğrendiğim için de sevindim (Fiziksel, Ö2, Erkek).”

“Hocam etkinliği sevdiğimde, bence bana robotik kodlama konusunda çok fazla katkı sağladı (Fiziksel, Ö3, Kız).”

“Robot üstüme geliyor gibi geldi (Sanal, Ö1, Kız).”

Sosyal Etkileşim

Bu kategoride ortaya çıkan, “Yardım İsteme”, “Akran Baskısı” temaları altında çeşitli kodlar belirlenmiştir. Bu kodlar, üstün yetenekli öğrencilerin sanal veya fiziksel robotik kodlama görevlerini yerine getirirken öğrenme ortamındaki diğer paydaşlarla sosyal etkileşime açık olup olmama durumlarını ortaya koymaktadır.

Yardım İsteme: Sanal robotik kodlama grubunda öğretmenden yardım istediğini dile getiren öğrenci sayısı fiziksel robotik kodlama grubundakilere göre çoğunluktadır. Bu bulgu, sanal robotik kodlama grubu katılımcılarının yardım ihtiyacını özellikle öğretmene yöneltme eğiliminde olduklarını göstermektedir. Bu durum, çevrim içi ortamlarda öğrencilerin bireysel olarak karşılaştıkları sorunlarda doğrudan öğretmene yönelmeyi tercih ettiklerini ve belirsizlik karşısında öğretici desteğine daha fazla ihtiyaç duyduklarını düşündürmektedir. Fiziksel robotik kodlama grubu katılımcılarının ise öğretmenden yardım isteme davranışının az olması, öğretmene doğrudan ulaşmanın olağan bir süreç olarak algılanmasından dolayı ifade edilmeye gerek görülmemesiyle açıklanabilir. Diğer taraftan her iki grupta da az sayıda öğrenci arkadaşlarından yardım istediğini dile getirmiştir. Bu durum, her iki grupta da az sayıda öğrencinin daha fazla akran etkileşimi üzerinden çözüm arama eğiliminde olduğunu, diğer bir

ifadeyle her iki gruptaki pek çok öğrencinin çok da fazla akran desteği arayışında olmadığını göstermektedir. Bu temadaki kodlara ilişkin öğrenci ifadelerinden alıntılar şunlardır:

“Öğretmenim size soruyorum. Çünkü yapamayınca hatayı bulacağımı veya sizin yardım edeceğinizi biliyorum (Sanal, Ö2, Erkek).”

“... neyse ki sizinle beraber ne yapmam gerektiğini tek tek öğrendim (Sanal, Ö1, Kız).”

“Bazı kodlarda yanlışlık çıkıyordu. Onu nasıl çözeceğimi bilmiyordum ama öğretmenimin yardımıyla onu da halledebiliyordum (Fiziksel, Ö1, Erkek).”

“Arkadaşlarımızdan da yardım aldık (Sanal, Ö3, Erkek).”

Akran Baskısı: Sanal robotik kodlama grubunda yalnızca bir öğrenci akran baskısı nedeniyle duygusal zorlanma yaşadığını dile getirmiş, fiziksel robotik kodlama grubunda ise böyle bir bulguya rastlanmamıştır. Bu bulgu, sanal ortamdaki robotik etkinliklerde az da olsa akran baskısının öğrenciler üzerinde duygusal etkiler yaratabileceğini göstermektedir. Fiziksel robotik kodlama grubunda bu tür bir duygusal zorlanmanın ifade edilmemesi ise, öğrencilerin yüz yüze ortamda sosyal baskıyı daha kolay yönetebildiklerini ya da bu baskının daha az hissedilir düzeyde yaşandığını düşündürülebilir. Bununla birlikte, iki grup arasındaki bu düşük frekans farkı dikkat çekicidir ve akran baskısının etkilerinin her iki ortamda da açıkça ifade edilmediğini göstermektedir. Bu durum, öğrencilerin duygusal zorlanmalarını paylaşma konusunda çekingen davranabileceklerini ya da bu tür deneyimlerin farkında olmadan içselleştirilebileceğini düşündürmektedir. Bu temadaki kodlara ilişkin öğrenci ifadelerinden alıntılar şunlardır:

“...yapamadığımda, arkadaşlarım bana dik dik baktığında nedense beni hep üzdü (Sanal, Ö1, Kız).”

BEŞİNCİ BÖLÜM

Tartışma ve Sonuç

Bu çalışmada; bilim ve sanat merkezinde, bireysel yetenekleri fark ettirme programında öğrenim gören 6. ve 7. sınıftaki üstün yetenekli öğrencilerin sanal (VEXcode-VR) ve fiziksel (Raspberry-Pi) robotik kodlama performansları, öz-yeterlilikleri ve kaygıları araştırılmıştır. Aynı zamanda sanal ve fiziksel robotik kodlama grupları için, bu değişkenler arasında anlamlı bir ilişki olup olmadığı da belirlenmiştir. Robotik kodlama performansı kapsamında ise; bu iki robotik kodlama ortamının öğrencilerin Python programlama dili başarılarına ve programlama dili bilgilerinin kalıcılığına (hatırlanmasına) etkisinin yanı sıra, haftalık kazanımlara göre öğrenme süreçlerindeki performansları ve meşguliyetleri de incelenmiştir. Ayrıca öğrencilerin sanal ve fiziksel robotik kodlama ortamlarındaki Python programlama dili öğrenme deneyimlerine ilişkin görüşleri karşılaştırılarak aralarındaki benzerlik ve farklılıklar ortaya çıkarılmıştır. Bu bölümde bulgular araştırma soruları çerçevesinde literatüre dayandırılarak yorumlanmış, çıkarımlar yapılmış ve gelecek çalışmalar için öneriler sunulmuştur.

Üstün Yetenekli Öğrencilerin Sanal ve Fiziksel Robotik Kodlama Performansları

Üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlama performansları kapsamında; grupların haftalık kazanımları ve meşguliyetlerinin yanı sıra, Python programlama dili başarıları ve bilgilerinin kalıcılığı (hatırlanması) ile ilgili ortaya çıkan sonuçlar tartışılmıştır.

Üstün Yetenekli Öğrencilerin Haftalık Sanal ve Fiziksel Robotik Kodlama Kazanımları

Her iki öğrenme ortamında da öğrencilerin haftalık Python programlama dili kazanım puanlarının yüksek olduğu ve 10 haftalık süreç boyunca istikrarlı bir düzeyde devam ettiği görülmüştür. Bu sonuç, sanal ve fiziksel ortamda yürütülen robotik kodlama etkinliklerinin üstün yetenekli öğrencilerin Python programlama dili kazanımlarını benzer düzeyde desteklediğini göstermektedir. Nitel bulgularda ortaya çıkan olumlu öğrenme çıktılarının her iki grupta da yüksek frekansta olması bu sonuçları desteklemektedir. Literatürde de benzer biçimde, öğrenme ortamından bağımsız olarak öğrencilerin bilişsel kapasitelerinin ve problem çözme becerilerinin gelişiminde robotik tabanlı programlama etkinliklerinin etkili olduğu belirtilmektedir (Eraytaç, 2019). Ayrıca, öğrencilerin haftalık kazanım puanlarının zaman içinde aynı seviyede seyretmesi, 10 haftalık uygulama boyunca öğrenme düzeylerinin korunduğunu göstermektedir. Bu durum, üstün yetenekli öğrencilerin soyut düşünme ve

algoritmik mantığı kavrama kapasitelerinin, farklı öğrenme ortamlarında benzer biçimde aktif hale geldiğini düşündürmektedir (Çubukçu & Tosuntaş, 2018).

Üstün Yetenekli Öğrencilerin Haftalık Sanal ve Fiziksel Robotik Kodlama Meşguliyetleri

Hem sanal hem de fiziksel robotik kodlama ortamlarının üstün yetenekli öğrencilerin meşguliyet düzeylerini büyük ölçüde benzer biçimde desteklediği görülmüştür. Grupların haftalık genel meşguliyet puanlarının yanı sıra alt meşguliyet boyutlarındaki ortalamaların 2-3 aralığında (2 = kısmen gözlendi, 3 = gözlendi) seyretmesi, öğrencilerin süreç boyunca aktif katılım gösterdiklerine işaret etmektedir. Davranışsal meşguliyetin en yüksek ortalama sahip oluşu, öğrencilerin öğrenme sürecine aktif olarak katılım sağladıklarının önemli bir göstergesidir. 10 haftalık süreçte sanal ve fiziksel robotik kodlama gruplarının duyuşsal, bilişsel, davranışsal, temsili, sosyal ve genel meşguliyet ortalamalarının paralel düzeyde devam etmesi ve gruplar arası anlamlı fark olmaması, her iki öğrenme ortamının da öğrencilerin ilgisini, katılımını ve bilişsel çabasını sürdürülebilir şekilde desteklediğini (Ru Qu & Fok, 2022; Sullivan & Bers, 2018), öğrenme memnuniyetini artırdığını göstermektedir (Zhong vd., 2020). Meşguliyetin alt boyutlarına ilişkin her iki grupta ortaya çıkan olumlu nitel bulgular da bu sonuçları desteklemektedir. Bu durum sanal ortamların da fiziksel uygulamalar kadar etkili bir öğrenme deneyimi sağlayabileceğini ortaya koymakta, sanal öğrenme ortamlarının özellikle üstün yetenekli öğrenciler için güvenilir bir alternatif öğrenme modeli olabileceğini göstermektedir (Erol vd., 2024).

Üstün yetenekli öğrencilerin sanal ve fiziksel robotik kodlama ortamlarındaki meşguliyet alt boyutlarına ilişkin haftalık ortalamalar incelendiğinde ise; “Döngüler” kazanımına yönelik etkinliklerin yapıldığı 4. haftada hem duyuşsal meşguliyet hem de genel meşguliyet açısından sanal robotik kodlama grubu lehine istatistiksel olarak anlamlı bir fark bulunmuştur. Bu durum, üstün yetenekli öğrencilerin çok adımlı kodlama süreçlerinin başlangıcında Raspberry-Pi ile yapılan fiziksel robotik kodlama etkinliğine duygusal uyum sağlamak zorlanabildiklerini düşündürmektedir. Nitel bulgulardaki duyuşsal tepkiler arasında yer alan “Zor görev karşısında öğrenme isteğiyle heyecanlanma” ifadesinin sanal robotik kodlama grubunda daha yüksek frekansa sahip oluşu bu sonucu desteklemektedir. Bu bulgular, duyuşsal meşguliyetin haftalık değişimlerinde etkinlik türü ve ortam özelliklerinin belirleyici rol oynayabileceğini göstermektedir. Zhao vd. (2024), fiziksel robotik kodlama ortamlarında ilk adaptasyon sürecinin öğrencilerin kaygı düzeylerini artırabileceğini belirtmiştir. Öte yandan, VEXcode-VR gibi sanal robotik bir ortamın görsellik ve geri bildirim açısından zengin içerikleri öğrencilerin ilgisini ve motivasyonunu artırmış olabilir (Sullivan & Bers, 2018).

Benzer şekilde, Rukangu vd. (2025) sanal ortamların “ilgi” bileşenini güçlendirdiğini vurgulamışlardır.

Her iki grubun davranışsal meşguliyet düzeyleri açısından 8. ve 9. haftalarda sanal robotik kodlama grubu lehine anlamlı farklar görülmüştür. Bu haftalarda yapılan “Listeler” ve “Proje Gerçekleştirme-Hata Ayıklama” etkinliklerinin kapsamlı, karmaşık ve çok adımlı kodlama işlemlerini içermesi, Raspberry-Pi fiziksel robotik aracıyla kodlama yapan üstün yetenekli öğrencilerin davranışsal meşguliyetlerinin sanal robotik kodlama grubuna göre daha düşük seviyede kalmasına neden olmuş olabilir. Nitel bulgulardaki davranışsal süreçler arasında yer alan “Hatalı kodu buluncaya kadar tekrar tekrar deneme yapma” ifadesinin sanal robotik kodlama grubunda daha yüksek frekansa sahip oluşu bu sonucu desteklemektedir. Nitekim Zhao vd. (2024), sanal robotik araçların özellikle zorlu görevlerde katılımı artırıcı rol oynayabileceğini belirtmiştir. Fredricks vd. (2004) ve Reeve (2012) de davranışsal meşguliyetin görev niteliği ile ilişkili olduğunu vurgulamışlardır.

Her iki grubun hem temsili ve sosyal meşguliyet hem de genel meşguliyet düzeyleri açısından “Fonksiyonlar” kazanımına yönelik etkinliklerin yapıldığı 6. haftada gruplar arasında fiziksel robotik kodlama grubu lehine istatistiksel olarak anlamlı bir fark saptanmıştır. Bu durum iç içe kodlama işlem adımlarının olduğu bu haftada, Raspberry-Pi ile yapılan fiziksel robotik kodlama etkinliğine katılan üstün yetenekli öğrencilerin sanal robotik kodlama grubuna kıyasla, kendi duygu ve düşüncelerini daha fazla dile getirerek temsili meşguliyet sergilediklerini ve daha fazla sosyal etkileşim içinde olduklarını göstermektedir. Zhao vd. (2024), fiziksel ortamların somut etkileşim vesilesiyle motivasyonu artırabileceğini, bu sayede öğrencilerin kendi duygu ve düşüncelerini daha rahat dile getirebileceğini belirtmiştir. Ayrıca bu sonuçlar, fiziksel ortamın belirli dönemlerde daha yoğun sosyal etkileşim fırsatları sunduğunu göstermektedir. Yüz yüze etkileşimin aidiyet duygusunu artırdığı (Aldridge & Fraser, 2016; Belt vd., 2021), fiziksel etkileşimin ve sosyal ortamın motivasyonu güçlendirdiği (Kahu, 2011) literatürde de vurgulanmaktadır. Ancak nitel bulgulardaki temsili/aracı katılım ve sosyal etkileşim kategorilerinde bu duruma zıt bulgular ortaya çıkmıştır. Fiziksel robotik kodlama grubundaki öğrenciler, sınıf ortamında etkinliklere katıldıkları için, bu süreçte kendi duygu ve düşüncelerini dile getirmelerini ve sosyal etkileşimi, olağan bir ders akışı olarak algılamış ve bu yöndeki görüşlerini ifade etme ihtiyacı duymamış olabilirler. Bu sonuç dikkate alınarak, yeni çalışmalarda öğrencilerin temsili ve sosyal meşguliyet boyutlarındaki görüşlerini daha net ortaya koymalarını sağlamak için, öğrencilere sınıf ortamında fiziksel bir araçla robotik kodlama yaptıklarını daha fazla vurgulayan sorular sorulması gerekebilir.

Üstün Yetenekli Öğrencilerin Sanal ve Fiziksel Robotik Kodlamadaki Python Programlama Dili Başarıları

Sanal ve fiziksel robotik kodlama gruplarındaki üstün yetenekli öğrencilerin 10 haftalık öğretim süreci sonundaki Python programlama dili son test başarı ortalamaları, ön test başarı ortalamalarına kıyasla anlamlı biçimde ve yüksek etkiye sahip seviyede artış göstermiştir. Bu durum hem sanal ve hem fiziksel robotik kodlama araçlarıyla yapılan etkinliklerde öğrencilerin Python programlama dilini öğrenme seviyelerinin anlamlı ve güçlü bir şekilde gelişmeye devam ettiğini göstermektedir. Yüksek etki büyüklükleri, her iki öğrenme ortamının da güçlü bir öğretimsel etkiye sahip olduğunu göstermektedir. Nitel bulgulara ortaya çıkan eğitsel katkıların her iki grupta da yüksek frekansta olması bu sonuçları desteklemektedir. Bu sonuçlar, robotik kodlama ve programlama öğretiminde hem sanal hem de fiziksel ortamların etkili olduğunu ortaya koyan önceki araştırmalarla tutarlıdır (Alimisis, 2013). Cam ve Kilicer (2022), LEGO Mindstorms EV3 robotik kodlama aracının sanal versiyonu ile fiziksel robot setlerini karşılaştırmış ve her iki ortamın da (sanal ve fiziksel) kodlama becerilerinde anlamlı artış sağladığını, birbirine benzer düzeyde etkili olduğunu belirlemiştir. Yurdakök ve Kalelioğlu (2023), fiziksel programlama aracı olan Micro:bit kullanarak 6 hafta süreyle Python öğretimi yapmış, ön test ve son testler arasında anlamlı fark olduğunu, öğrencilerin Python programlama ile ilgili değişkenler, koşullar, döngüler ve fonksiyonlar gibi kavramları öğrenmede önemli kazanımlar sağladığını göstermiştir.

Her iki grubun ön test ve son test ortalamaları dikkate alındığında, sanal robotik kodlama grubundaki artışın, fiziksel robotik kodlama grubundakine kıyasla daha yüksek olduğu belirlenmiştir. Bu durum sanal robotik kodlama ortamının bilgi düzeyini geliştirmede ve öğrenmenin sürdürülebilirliğini sağlamada etkili olduğunu düşündürmektedir. Sanal ortamların donanım ve kontrol gibi kullanım kolaylığı sunan unsurlar açısından öğrencilere tekrar etme, görselleştirme imkânı sağlayarak kodlama becerilerini geliştirdiği alanyazında bildirilmiştir (Sucu & Çakıroğlu, 2022). Bakırcı (2019), sanal programlama aracı ile geleneksel öğretim ortamını kıyasladığında, robotik kodlama eğitiminin başarıyı artırmada, programlama ve algoritma mantığını kazandırmada etkili olduğunu belirlemiştir. Sonuç olarak, esnek yapısı sayesinde sanal öğrenme ortamları öğrencilerin kendi hızlarında ilerleyebilmesine, öğrenme çıktılarının daha da güçlenmesine katkı sağlayabilir.

Sanal ve fiziksel robotik kodlama gruplarının son test başarı ortalamaları arasında sanal robotik kodlama grubu lehine anlamlı seviyede ve yüksek etkiye sahip fark olduğu belirlenmiştir. Bu bulgular, sanal robotik kodlama ortamlarının fiziksel robotik kodlama ortamlarına kıyasla üstün yetenekli öğrencilerin Python programlama dili öğrenme süreçlerinde daha etkili olduğunu göstermektedir. Fiziksel robotik uygulamaların ise özellikle somut

deneyim, iş birliği ve takım çalışmasını desteklediği bilinmektedir (Bers, 2020). Ancak çalışmada elde edilen bulgular, sanal ortamın yüksek başarıya daha fazla katkı sunduğunu göstermektedir. Bunun nedeni, üstün yetenekli öğrencilerin hızlı öğrenme eğilimleri ve bireyselleştirilmiş öğrenme ortamlarından daha fazla fayda sağlamaları olabilir (Subotnik vd., 2011). Ayrıca alanyazında, sanal ortamda sağlanan esnek öğrenme süreçlerinin öğrencilerin öğrenme motivasyonunu ve akademik katılımını artırdığı belirtilmiştir (Aldridge & Fraser, 2016).

Sanal öğrenme ortamlarının, öğrencilere bireysel hızda ilerleme, tekrar yapabilme ve çoklu etkileşim imkânları sunarak öğrenmeyi desteklediği özellikle etkileşimli içeriklerin ve öğrenci merkezli etkinliklerin, öğrencilerin programlama becerilerini geliştirmede önemli rol oynadığı (Korkmaz & Altun, 2014) önceki araştırmalarda vurgulanmıştır. Nitekim yapılan araştırmalarda, sanal robotik kodlama uygulamalarının öğrencilerin problem çözme, algoritmik düşünme ve bilişsel esneklik becerilerini geliştirdiği de belirtilmektedir (Altıok & Üçgöl, 2024). Sonuç olarak, sanal robotik kodlama ortamlarının üstün yetenekli öğrencilerin programlama başarısını artırmada güçlü bir alternatif olduğu söylenebilir. Nitel bulgulara ortaya çıkan olumlu öğrenme çıktıları arasındaki “Python programlama dilini öğrenme” ile ilgili öğrenci ifadelerine ait frekansların sanal robotik kodlama grubunda daha yüksek olması bu sonuçları desteklemektedir.

Üstün Yetenekli Öğrencilerin Sanal ve Fiziksel Robotik Kodlamadaki Python Programlama Dili Bilgilerinin Kalıcılığı (Hatırlanması)

Her iki grupta da ön test ve kalıcılık testi ortalamaları arasında kalıcılık testi lehine yüksek etkiye sahip anlamlı fark olduğu belirlenmiştir. Bu durum hem sanal hem de fiziksel robotik kodlama ortamlarının öğrenme sürecinde bilgi kalıcılığı açısından güçlü sonuçlar verdiğini göstermektedir. Etki büyüklüklerinin her iki grup için de yüksek olması, Python programlama eğitiminin kalıcı öğrenmeler sağladığını göstermektedir. Fiziksel materyal kullanımı, doğrudan öğretmen geribildirim ve sosyal etkileşim gibi öğeler fiziksel ortamda daha belirgin olabilmekte ve öğrencilerin öğrendiklerini unutma olasılığını azaltabilmektedir. Bu durum öğrencilerin derin öğrenmesini sağlayabilmekte ve kavramların daha kalıcı olmasına büyük destek sunabilmektedir (Selçuk, 2019). Bunun yanında tekrar-odaklı ve etkileşimli sanal öğrenme materyallerinin başarı açısından fiziksel materyaller ile rekabet edebilecek seviyede olduğu da alanyazında dile getirilmiştir (Fatsa & Turan, 2022). Bu çalışmada ortaya çıkan sonuçlar da, önceki araştırmalarda vurgulanan “robotik tabanlı programlama eğitiminin öğrenme motivasyonunu ve uzun vadeli bilişsel kazanımları artırdığı” bulgularıyla örtüşmektedir (Rosenberg & Zviel-Girshin, 2024).

Her iki grupta da son test ve kalıcılık testi ortalamaları arasında istatistiksel olarak anlamlı bir fark bulunmamıştır. Bu durum, hem sanal ve hem fiziksel robotik kodlama araçlarıyla yapılan etkinliklerde üstün yetenekli öğrencilerin edindiği Python programlama dili bilgilerini, tüm uygulama bitiminden 4 hafta sonra bile benzer seviyede hatırladıklarını göstermektedir. Ayrıca her iki gruptaki öğrencilerin son testteki performanslarını koruduklarını, yani öğrenmenin büyük ölçüde sürdürüldüğünü de ortaya koymaktadır. Bu bulgular, literatürde robotik ve programlama eğitimlerinin uzun dönemli öğrenme üzerinde olumlu etkileri olduğunu ortaya koyan çalışmalarla tutarlıdır (Alimisis, 2013). Alanyazında hem sanal hem de fiziksel robotik kodlama ortamlarının yalnızca kısa vadede başarıyı artırmakla kalmayıp uzun vadede da öğrenmenin kalıcılığını desteklediğini ortaya koyan çalışmalar da mevcuttur (Fernando & Gonzalez, 2024). Bu sonuçlar, uygulanan 10 haftalık etkinliklerdeki pekiştirme unsurlarının kalıcı öğrenme izlerini güçlendirmiş olduğuna da işaret ettiğinden, öğrenmenin zaman içine dağıtılarak yapılan aralıklı tekrarların, bilginin geri çağırmasını ve uzun dönem kalıcılığını desteklediğini göstermektedir (Selcuk vd., 2024; Stasolla vd., 2025).

Sanal ve fiziksel robotik kodlama gruplarının kalıcılık testi başarı ortalamaları arasında sanal robotik kodlama grubu lehine anlamlı ve orta seviyede etkiye sahip bir fark olduğu belirlenmiştir. Bu durum, sanal robotik kodlama aracıyla yapılan eğitimde fiziksel robotik kodlama aracıyla yapıldığına göre üstün yetenekli öğrencilerin Python programlama dili bilgilerinin daha kalıcı olduğunu, sanal robotik kodlama aracının bilginin hatırlanmasında daha etkili olduğunu göstermektedir. Sanal ortamlar, öğrencilere risk almadan deneme yapma, anında geribildirim alma ve çeşitli senaryoları kolaylıkla simüle etme fırsatı sunarak öğrenme sürecini zenginleştirmektedir (Bers, 2020). Sanal robotik kodlama uygulamalarının bu etkileşimli yapısının, öğrenenlerin hata yapma ve geri bildirim alma süreçlerini kolaylaştırarak bilişsel yükü azaltabileceğini ve problem çözme becerilerini geliştirebileceğini de düşündürmektedir (Altıok & Üçgül, 2024). Nitel bulgularda ortaya çıkan olumlu öğrenme çıktıları arasındaki “Robotik kodlama mantığını anlama” ile ilgili öğrenci ifadelerine ait frekansların sanal robotik kodlama grubunda daha yüksek olması da bu sonuçları desteklemektedir. Bu bulgular, dijitalleşen dünyada sanal ortamlar gibi esnek, etkileşimli ve erişilebilir araçların robotik kodlama eğitimi süreçlerine daha fazla entegre edilmesi gerektiğini düşündürmektedir. Gelecek araştırmalarda, çeşitli sanal ve fiziksel robotik kodlama araçlarıyla, farklı programlama dilleri ve uzun süreli uygulamalarda benzer karşılaştırmalar yapılarak bu sonuçların genellenebilirliği test edilebilir.

Üstün Yetenekli Öğrencilerin Sanal ve Fiziksel Robotik Kodlama Öz-Yeterlikleri

Fiziksel robotik kodlama grubunun uygulama öncesi ve sonrası robotik kodlama öz-yeterlik ortalamaları arasında son test lehine yüksek etkiye sahip anlamlı bir fark olduğu belirlenmiştir. Bu durum, fiziksel robotik kodlama araçlarıyla yapılan etkinliklerin öğrencilerin Python programlama dilini öğrenmeye yönelik öz-yeterlik seviyeleri üzerinde anlamlı ve güçlü bir etkisi olduğunu ortaya koymaktadır. Fiziksel robotik uygulamalarda öğrenciler, robotları doğrudan kontrol ederek problem çözme sürecine aktif biçimde katılır ve hata düzeltme süreçlerinde gerçek zamanlı geribildirim alır. Bu deneyimler, öz-yeterlik algısının artmasını destekleyen otantik başarı yaşantıları sağlamış olabilir (Zhang & Zhu, 2022). Nitel bulgularda ortaya çıkan “Robotik kodlama yeterliğine yönelik öz güven geliştirme” ile ilgili öğrenci ifadelerine ait frekansların fiziksel robotik kodlama grubunda daha yüksek olması da bu sonuçları desteklemektedir.

Sanal robotik kodlama grubunda ise ön test ile son test öz-yeterlik puanları arasında anlamlı bir fark gözlenmemiştir. Bu durum sanal robotik kodlama araçlarıyla yapılan etkinliklerin öğrencilerin Python programlama dilini öğrenmeye yönelik öz-yeterlik seviyeleri üzerinde anlamlı bir değişim oluşturmadığını göstermektedir. Sanal robotik ortamların öz-yeterlik üzerinde anlamlı bir değişim yaratmaması, literatürde belirtilen sınırlılıklarla örtüşmektedir. Sanal ortamlar hızlı geri bildirim ve kişisel hızda öğrenme avantajı sağlasa da duygusal bağ ve fiziksel deneyim eksikliği öz-yeterlik oluşumunu olumsuz etkileyebilir (Liang & Du, 2025). Benzer şekilde Usher vd. (2025), sanal uygulamalarda modelleme ve sosyal etkileşim etmenlerinin yeterince güçlü olmamasının öz-yeterlik gelişimini zayıflatıldığını belirtmiştir. Nitel bulgularda ortaya çıkan “Robotik kodlama alanında kendini geliştireceğine inanma” ile ilgili öğrenci ifadelerine ait frekansların sanal robotik kodlama grubunda daha düşük olması da bu sonuçları desteklemektedir.

Gruplar arası karşılaştırmalarda ise, sanal ve fiziksel robotik kodlama gruplarının öz-yeterlik son test ortalamalarının yüksek olduğu belirlenmiş, ancak gruplar arasında istatistiksel olarak anlamlı bir fark bulunmamıştır. Bu durum, üstün yetenekli öğrencilerin Python programlama dili öğrenme sürecinde sanal veya fiziksel robotik kodlama aracıyla uzun süre etkinlik yapmalarının öz-yeterlik seviyelerinde anlamlı bir değişime neden olmadığını, grupların 10 haftalık uygulama sonrası öz-yeterlik düzeylerinin birbirine oldukça benzediğini ortaya koymuştur. Özellikle her iki grubun da üstün yetenekli öğrencilerden oluşması, başlangıçtan itibaren yüksek öz-yeterlik düzeyi nedeniyle anlamlı bir farkın ortaya çıkmamasını açıklayabilir (Aksoy, 2014; Alabbasi vd., 2022; Bandura, 1997). Ayrıca öz-yeterlik algısının güçlenmesi, bireylerin başarı deneyimleri yaşamaları, modelleme ve akran gözlemleri

yapmaları, sosyal destek almaları ve olumlu geri bildirimlerle pekiştirilmeleriyle mümkün olabilir (Schunk & DiBenedetto, 2020). Bu çalışmadaki her iki grupta da öz-yeterlik son test ortalamalarının yüksek çıkması, üstün yetenekli öğrencilerin 10 haftalık etkinliklerde bu görüşü destekleyecek deneyimler yaşadıklarını ortaya koymaktadır.

Bulgular, sanal (VEXcode-VR) ve fiziksel (Raspberry-Pi) robotik kodlama ortamlarının herhangi birinin diğerine kıyasla öz-yeterlik algısında belirgin bir avantaj veya üstünlük sağlamadığını göstermektedir. De la Hoz vd. (2024) robotik uygulamaların katılımcıların özgüvenini artırdığını, ancak ortam türünün (sanal veya fiziksel) bu artış üzerinde anlamlı fark oluşturmadığını bildirmiştir. Benzer biçimde bir meta-analiz çalışmasının sonuçları da, sanal ve fiziksel robotik ortamlarının kullanıcı algısı, sosyal varlık ve tutum değişkenleri üzerinde benzer etkiler yarattığını, farklılıkların daha çok görev karmaşıklığı veya kullanıcı deneyim düzeyiyle ilişkili olduğunu göstermiştir (Mulwijik & Lazonder, 2023). Zhang ve Zhu (2022) robotik etkinliklerin öğrencilerin STEM alanlarındaki güven duygularını ve problem çözme becerilerini geliştirdiğini, böylece öz-yeterlikte anlamlı artışlar sağladığını belirtmiştir. Nitel bulgularda ortaya çıkan “Zorlu kodlama görevlerinde kararlılık gösterme” ile ilgili sanal ve fiziksel robotik kodlama gruplarındaki öğrenci ifadelerinin aynı frekansa sahip olması da bu sonuçları desteklemektedir.

Öz-yeterlik açısından gruplar arası fark çıkmamasının sebebi, kullanılan sanal (VEXcode-VR) ve fiziksel (Raspberry-Pi) robotik kodlama araçlarının özellikleri de olabilir. Nitekim sanal ortamda yürütülen çalışmaların öz-yeterlik artışında daha sınırlı etkiler gösterdiğine dair çalışmalar da mevcuttur (Yang vd., 2022). Bu nedenle bu sonuçların genellenebilmesi için ileriki araştırmalarda, bu çalışmada kullanılan robotik kodlama araçlarından farklı sanal ve fiziksel robotik kodlama araçlarıyla yapılan uzun süreli benzer uygulamalarda üstün yetenekli öğrencilerin öz-yeterlik düzeyleri karşılaştırılabilir.

Üstün Yetenekli Öğrencilerin Sanal ve Fiziksel Robotik Kodlama Kaygıları

Her iki grupta da kaygı son test ortalaması ön test ortalamasına göre azalmış olsada, grup içi ön test ve son test ortalamaları arasında istatistiksel olarak anlamlı bir fark bulunmamıştır. Bu bulgular hem sanal ve hem fiziksel robotik kodlama araçlarıyla yapılan 10 haftalık etkinliklerin öğrencilerin kaygılarını olumlu yönde etkilese de, her iki gruptaki üstün yetenekli öğrencilerin Python programlama dili öğrenmeye yönelik kaygılarında anlamlı bir değişim sağlamadığını göstermektedir. Fan vd. (2025) yaptığı çalışmada, öğrencilerin kaygı düzeylerinde azalma eğilimi bulunduğunu, ancak bu azalmanın her durumda istatistiksel olarak anlamlı olmadığını bildirmiştir. Öğrencilerin duyuşsal tepkilerine yönelik nitel bulgularda yer

alan “Robotik problemi çözememe kaygısı yaşama” ile ilgili her iki gruptaki öğrenci ifadelerinin düşük seviyede aynı frekansa sahip oluşu da bu sonuçları desteklemektedir.

Sanal ve fiziksel robotik kodlama gruplarının kaygı son test ortalamaları arasında da istatistiksel olarak anlamlı bir fark bulunmamıştır. Bu durum, her ne kadar her iki grupta da son test kaygı ortalamaları ön teste göre düşüş gösterecek, üstün yetenekli öğrencilerin Python programlama dili öğrenme sürecinde sanal veya fiziksel robotik kodlama aracıyla 10 haftalık etkinlik yapmalarının kaygı seviyelerinde anlamlı bir değişime ve düşüşe neden olmadığını ortaya koymaktadır. Alanyazında da benzer sonuçlara ulaşan çalışmalar mevcuttur. Fan vd. (2025) sanal ve fiziksel robotik ortamlar arasında kaygı düzeyleri açısından belirgin bir fark saptamamıştır. Yang (2025), robot ile kodlama ve tablet uygulama ile kodlama karşılaştırması yaptığı çalışmada, her iki grubun bilişsel düşünme becerilerinde gelişim sağladığını, ancak sosyal-duygusal beceriler ve duygusal düzenleme açısından anlamlı fark bulunmadığını belirlemiştir. Bu durum, farklı öğrenme ortamlarının öğrencilerin kaygı düzeylerini benzer biçimde etkilediğini desteklemektedir. Hsu vd. (2022) de eş programlama temelli robotik etkinliklerin öğrencilerin dil öğrenme kaygısını azalttığını, problem çözme ve sosyal etkileşim becerilerini geliştirdiğini göstermiştir. Ayrıca çevrimdışı ve çevrimiçi robotik uygulamalarını karşılaştıran Madariaga vd. (2023), öğrencilerin motivasyon ve katılım düzeylerinde ortam türüne göre anlamlı fark olmadığını; her iki ortamın da benzer düzeyde ilgi ve öğrenme isteği yarattığını bildirmiştir. Zhong (2023), sanal ve fiziksel robotik ortamların öğrencilerin tutum ve becerilerinde anlamlı fark yaratmadığını vurgulamıştır. Öğrencilerin duyuşsal tepkilerine yönelik nitel bulgulara yer alan “Yeni robotik bilgileri keşfetme heyecanı yaşama” ile ilgili her iki gruptaki öğrenci ifadelerinin aynı frekansa sahip oluşu da bu sonuçları desteklemektedir.

Her iki grubun da genel olarak düşük düzeyde kaygı göstermesi, öğrencilerin robotik kodlama sürecine olumsuz duygular olmadan katılabildiklerini göstermektedir. Ayrıca hem sanal hem de fiziksel robotik kodlama gruplarındaki katılımcıların üstün yetenekli öğrencilerden oluşması da bu olumlu sonuçların ortaya çıkmasını sağlamış olabilir. Olpak ve Karamete (2024) tarafından yürütülen araştırmada, üstün yetenekli öğrencilerin öz-yeterlik algıları ile kaygı düzeyleri arasında anlamlı bir ilişki olduğu ortaya konulmuştur. Çalışma bulguları, öz-yeterlik düzeyi yüksek olan öğrencilerin kaygı düzeylerinin daha düşük olduğunu göstermektedir. Benzer şekilde, bu araştırmada da öz-yeterlik algısı yüksek olan öğrenci gruplarının programlama kaygısı düzeylerinin orantılı olarak daha düşük olduğu belirlenmiştir. Zhong (2023) da öğrencilerin duygusal destek ve hata yönetimi deneyimlerinin robotik kodlama kaygısı üzerindeki belirleyici faktörler olduğunu vurgulamıştır. Öğrencilerin duyuşsal tepkilerine yönelik nitel bulgulara yer alan “Yeni bir robotik araç ile yeni bilgiler öğrendiği

için mutlu olma” ve “Zor kodlama sürecini tamamlamaktan memnuniyet duyma” ile ilgili her iki gruptaki öğrenci ifadelerinin aynı frekanslara sahip oluşu da bu sonuçları desteklemektedir.

Her ne kadar son testler açısından gruplar arası anlamlı bir fark olmasa da, sanal robotik kodlama grubunun kaygı son test ortalamasının fiziksel robotik kodlama grubundan daha düşük olduğu belirlenmiştir. Bu sonuç, bu tür platformların özellikle robotik kodlama ile bir programlama dili öğrenmeye yeni başlayan öğrenciler için daha destekleyici bir öğrenme atmosferi sağlayabileceğini göstermektedir. Alanyazında da benzer sonuçlara ulaşan çalışmalar mevcuttur. Alsoliman (2022), sanal robotik ortamlarda öğrencilerin hata yapma korkularının azaldığını, bunun da psikolojik olarak daha güvenli bir öğrenme alanı oluşturduğunu vurgulamıştır. Benzer biçimde Kolil vd. (2020) sanal deney platformlarının öğrencilerin öz-yeterlik algılarını artırdığını ve deneysel kaygıyı azalttığını belirtmiştir. Sanal robotik kodlama grubundaki öğrencilerin nitel bulgulara yer alan duyuşsal tepkilerine yönelik “Tekrar tekrar denemekten korkmama” ve “Zor görev karşısında öğrenme isteğiyle heyecanlanma” ile ilgili ifadelerinin fiziksel robotik kodlama grubundaki öğrencilerden daha yüksek frekanslara sahip oluşu da bu sonuçları desteklemektedir.

Üstün Yetenekli Öğrencilerin Sanal ve Fiziksel Robotik Kodlama Performansları, Öz-Yeterlikleri ve Kaygıları Arasındaki İlişki

Fiziksel robotik kodlama grubunda Python programlama dili başarısı ile programlama kaygısı arasında güçlü, negatif yönlü ve istatistiksel olarak anlamlı bir ilişki bulunmuştur. Bu bulgu, fiziksel robotik kodlama aracıyla etkinlik yapan üstün yetenekli öğrencilerin Python programlama diline yönelik başarı arttıkça kaygılarının önemli seviyede azaldığını göstermektedir. Ayrıca, bu grupta başarı ile öz-yeterlik arasında pozitif yönlü ve orta düzeyli, öz-yeterlik ve kaygı arasında ise negatif yönlü ve orta düzeyli bir ilişki görülmüş, ancak bu ilişkilerin istatistiksel olarak anlamlı olmadığı tespit edilmiştir. Literatürde programlama dili öğrenimi bağlamlarında sıkça araştırmalara konu edilen “yüksek başarı = daha düşük kaygı” modelini destekleyen çalışmalar mevcuttur (Demir, 2015; Hsu & Hwang, 2023; Olpak & Karamete, 2024; Olipas vd., 2021).

Sanal robotik kodlama grubunda ise, Python programlama dili başarısı ile programlama kaygısı arasında negatif yönlü ve orta düzeyli, başarısı ile öz-yeterlik ve öz-yeterlik ile kaygı arasında pozitif yönlü ve zayıf düzeyli ilişkiler görülmüş, ancak bu ilişkilerin istatistiksel olarak anlamlı olmadığı tespit edilmiştir. Alanyazında bu yönde çok az çalışmaya erişilebilmiştir (Hsu & Hwang, 2023). Bu nedenle fiziksel robotik kodlamanın yanı sıra özellikle sanal robotik kodlamada bu değişkenler arası ilişkileri inceleyen yeni araştırmalara ihtiyaç olduğu söylenebilir.

Üstün Yetenekli Öğrencilerin Sanal ve Fiziksel Robotik Kodlama Sürecindeki Python Programlama Dili Öğrenme Deneyimleri

Nitel bulgular; “Öğrenme Çıktıları”, “Bilişsel Süreçler”, “Duyuşsal Tepkiler”, “Davranışsal Eğilimler”, “Temsili/Aracı Katılımlar” ve “Sosyal Etkileşimler” kategorilerinde toplanmıştır. Sonuçlar, sanal (VEXcode-VR) ve fiziksel (Raspberry-Pi) robotik kodlama gruplarındaki üstün yetenekli öğrencilerin 10 haftalık etkinlikler sonunda Python programlama dili öğrenmeye ilişkin deneyimlerinin; bu kategoriler açısından bazı durumlarda benzer ve bazı durumlarda farklılaşan nitelikte etkiler yarattığını ortaya koymaktadır.

Öğrenme Çıktıları: Her iki grupta öğrencilerin Python programlama dilini öğrenme kazanımı edindikleri ve bu yeni araçlarla robot bileşenlerini kodlayarak deneyimleme fırsatı buldukları görülmüştür. Ayrıca bu etkinlikler sayesinde her iki gruptaki bazı öğrencilerin, edindikleri bu robotik bilgilerini gelecekteki mesleklerinde kullanabileceklerini fark ettiği, hatta robotik alanına yönelik mesleki eğilim geliştirdiği belirlenmiştir. Bu kazanımlar, literatürdeki robotik uygulamaların öğrencilerin bilişsel ve teknik yetkinliklerini artırdığı yönündeki bulgularla tutarlıdır (Çiftçi & Bildiren, 2020; Usher vd., 2025). Sanal robotik kodlama grubunda, bu etkinlikler sayesinde robotik kodlama mantığını anladığını ve yaptığı robotik kodlama hatalarından ders çıkarmayı öğrendiğini dile getiren öğrenciler fiziksel robotik kodlama grubuna göre çoğunluktadır. Fiziksel robotik kodlama grubunda ise, gerçekçi görev ve senaryolar sayesinde robotik kodlamayı daha kolay öğrendiklerini belirten öğrenciler sanal robotik kodlama grubuna göre çoğunluktadır. Buna göre her iki gruptaki öğrencilerin bu öğrenme sürecinde eğitsel açıdan çeşitli kazanımlar ve deneyimler elde ettikleri söylenebilir (Gürkez, 2021).

Her ne kadar hem sanal hem de fiziksel gruptaki az sayıda öğrencinin robotik kodlama görevlerini tamamlamakta ve kod yazmakta zorlandığı görülse de öğrencilerin çoğunlukla kod yapılarını unuttuğu saptanmıştır. Bazı araştırmalar, bu süreçte öğrencilerin bilişsel yük dengesinin ve görev karmaşıklığının dikkate alınmasının kritik bir tasarım unsuru olarak öne çıktığı için robotik uygulamaların özellikle karmaşık görevlerde öğrenme sürecini zorlaştırabileceğini vurgulamaktadır (Chen vd., 2020). Sanal robotik kodlama grubundaki öğrencilerin kodlama sürecinde sıklıkla söz dizimi hatası yaptıkları ön plana çıkmıştır. Ayrıca bu grupta Türkçe karakter kullanma ve hatalı kodu bulmakta zorlanma durumuyla daha çok karşılaşmıştır. Başlangıç seviyesindeki programlama öğrencilerinin sıklıkla söz dizimi ve mantık hatalarıyla karşılaştığını ve bu hataların giderilmesi için hata ayıklama becerilerinin geliştirilmesinin kritik olduğunu belirten çalışmalar bu bulguyu desteklemektedir (Zhang & Jalal, 2019). Fiziksel robotik kodlama grubunda ise; uygun robotik aracı seçmekte veya bağlantıyı kurmakta zorlanan (örneğin LED bağlama gibi işlemler) öğrencilerin daha çok

olduğu gözlemlenmiştir. Ayrıca bu grupta fonksiyonları kullanırken zorlanma durumuyla daha çok karşılaşmıştır. Bu bulgu, fiziksel robotik çalışmalarında donanım bağlantısı, sensör entegrasyonu ve programlama senkronizasyonunun başarıyı etkileyen temel faktörler olduğunu vurgulayan araştırmalarla paralellik göstermektedir (Chalmers, 2018). Tüm bu bulgular, her iki gruptaki öğrencilerin ilk kez bu çalışma kapsamında Python programlama dilini kullanarak robotik kodlama yapmaya başlaması nedeniyle, gerçekçi ve dijital bir aracı programlama alışkanlıklarının yetersizliğine işaret edebilir.

Bilişsel Süreçler: Her iki grupta da aynı sayıda öğrenci, zorlu kodlama görevlerinde kararlılık gösterdiğini dile getirmiştir. Bu bulgu, öğrencilerin problem çözme sürecine aktif şekilde katıldığını ve çaba göstermeye istekli olduklarını ortaya koymaktadır. Alanyazında da robotik etkinliklerin bilişsel katılımı artırdığını teyit eden (Sullivan & Bers, 2018), öğrencilerin motivasyon, katılım ve öz-yeterlik düzeylerini artırdığını belirtilen (Zamora vd., 2025) çalışmalar mevcuttur. Ayrıca bu çalışmada robotik problemi çözeceğine/başaracağına inanan ve robotik kodlama konusunda kendini yeterli hisseden öğrenci sayısı da her iki grupta birbirine yakın seviyededir. Alanyazında robotik kodlamanın, öğrencilerin öz-yeterlik, yaratıcılık problem çözme ve bilişsel strateji kullanımı gibi üst düzey düşünme süreçlerini geliştirdiği vurgulanmaktadır (Benitti, 2012). Ancak robotik kodlama öz-yeterliğine yönelik öz güven geliştirdiğini net olarak dile getiren öğrenci sayısı fiziksel robotik kodlama grubunda daha yüksek düzeydedir. Bu durum, her iki gruptaki öğrencilerin genel olarak robotik kodlamaya yönelik olumlu bir öz-yeterlik algısına sahip olduklarını göstermekle birlikte, fiziksel robotik kodlama grubunda daha belirgin seviyede olduğunu ortaya koymaktadır. Alanyazında da fiziksel etkileşimin, öğrencilerin öz-yeterlik algısını daha güçlü bir şekilde desteklediği (Kılıçkiran vd., 2020), fiziksel robotiklerin öğrencilerin problem çözme becerilerini ve motivasyonlarını artırdığı (Alimisis, 2013) dile getirilmiştir.

Diğer taraftan hem problem çözümünü deneme-yanılma yoluyla başaracağına inanan hem de kendini robotik kodlama konusunda yeterli hissetmeyen öğrenci sayısı sanal robotik kodlama grubunda daha fazladır. Aynı grupta ortaya çıkan bu iki zıt bulgu, öğrencilerin ilk kez sanal bir ortamda robotik kodlama etkinliklerine katılım göstermelerinin bir sonucu olabilir. Bu durum sanal ortamlarda deneyim eksikliğinin bilişsel strateji geliştirmeyi sınırlayabileceğini göstermiştir (Alsoliman, 2022). Robotik probleme çözüm bulma konusunda alternatif yollar üretme ve hata ayıklama yapabileceğine inanma durumu ise düşük düzeyde de olsa yalnızca sanal robotik kodlama grubunda dile getirilmiştir. Bu durum, öğrencilerin bu becerileri gerçekleştirmede veya ifade etmede sınırlı kaldıklarını, bazı bilişsel stratejileri (ör. hata ayıklama, alternatif çözüm üretme) kullanma konusunda daha fazla destek ve deneyime ihtiyaç

duyduklarını göstermektedir. Sanal ortam hata yapma korkusunu azaltarak öğrencilerin deneme-yanılma yoluyla bilişsel risk alma eğilimini artırmış (Chao, 2016) ve soyut kavramları görselleştirerek kavramsal anlamayı kolaylaştırmış olabilir (Wu & Su, 2021).

Her iki grupta da etkinlikleri günlük yaşamdaki robotik uygulamalarla ilişkilendiren ve gerçek hayat için robotik projeler üretebileceğini fark eden öğrenciler mevcuttur. Bu bulgu, öğrencilerin robotik etkinliklerde kazandıkları bilgileri günlük yaşam bağlamına transfer edebildiklerini göstermektedir. Alanyazında bu sonucu destekleyen araştırmalara rastlanmıştır (Cam & Kızılcı, 2022). Ancak bu oran fiziksel robotik kodlama grubuna göre sanal robotik kodlama grubunda oldukça yüksek seviyededir. Bu durum ise, sanal ve fiziksel ortamlarda kullanılan robotik kodlama araçlarının teknik özelliklerinden kaynaklanmış olabileceği gibi, sanal robotik kodlama etkinliklerinin öğrencilerin bilgiyi gerçek yaşamla ilişkilendirmelerine daha fazla katkı sunduğunu ortaya koymaktadır. Benzer şekilde Gegenfurtner vd. (2014) da, sanal ortamdaki simülasyonların gerçek yaşam senaryolarını modelleme kapasitesi sayesinde bilgiyi gerçek yaşama transfer etmeyi kolaylaştırabileceğini dile getirmiştir.

Duyuşsal Tepkiler: Sanal robotik kodlama grubundaki öğrenciler bu etkinliklerle ilk karşılaştıklarında fiziksel robotik kodlama grubundakilere göre ne öğreneceklerini daha fazla merak etmelerine karşın, yalnızca sanal robotik kodlama grubundaki bazı öğrencilerin ilk izlenim olarak bu etkinliklerin karmaşık olduğunu düşünmeleri şaşırtıcıdır. Bu durum, özellikle sanal robotik kodlama aracının öğrencilerin ilk kez katıldıkları için bu etkinliklere karşı daha yüksek bir merak duygusu taşımalarını sağladığını, ancak bu ortamdaki robotik kodlama etkinliklerini başlangıçta zihinsel olarak zorlayıcı bulmalarına neden olduğunu ortaya koymaktadır. Sanal ortamların öğrenciler üzerinde bu yönde etkileri olduğunu belirten çalışmalara rastlanmıştır (Miguel-Alonso vd., 2023).

Sanal robotik kodlama grubunda hata yapınca endişeye kapılan öğrenciler çoğunlukta iken, fiziksel robotik kodlama grubunda ise; her ne kadar başaramayacağını düşünen/kendine güvenmeyen öğrenciler olsa da, etkinliklerde genel olarak rahat olan, kaygı duymayan öğrenciler çoğunluktadır. Bu bulgu, özellikle fiziksel robotik kodlama etkinliklerinin sanal robotik kodlama etkinliklerine göre öğrencilerin sürece daha rahat yaklaşmalarını ve önemli bir kısmının kaygı hissetmeden etkinliklere katılmalarını sağladığını göstermektedir. Buna karşın sanal robotik kodlama etkinliklerinin öğrencilerin hata yapma sonrası daha fazla endişe yaşamalarına neden olduğunu da ortaya koymaktadır. Sanal ortamların öğrenciler üzerinde bu yönde etkileri olduğunu belirten çalışma alanyazında mevcuttur (Cam & Kilicer, 2022).

Fiziksel robotik kodlama grubundaki bazı öğrenciler hataları bulup düzeltme sürecinde sıkıldığını belirtmiştir. Buna karşılık, sanal robotik kodlama grubundaki öğrenciler bu duruma

ilişkin herhangi bir olumsuz duygu bildirmemiştir. Bu bulgu, fiziksel ortamda yapılan robotik kodlama etkinliklerindeki hata düzeltme sürecinin bazı öğrenciler için motivasyon kaybına neden olabileceğini, sanal ortamda yapılan robotik kodlama etkinliklerinde ise böyle bir duygu durumunun yaşanma ihtimalinin daha düşük olduğunu göstermektedir. Fiziksel robotik kodlama etkinliklerinin öğrenciler üzerinde bu yönde etkileri olduğunu belirten çalışmaya alanyazında ulaşılmıştır (Uğuz, 2019).

Fiziksel robotik kodlama grubundaki öğrencilerden hiçbiri, başarısız denemelerde veya akran başarısı karşısında olumsuz duygular yaşadığını belirtmemiştir. Buna karşılık, sanal robotik kodlama grubundaki öğrencilerden bazıları başarısız denemelerde olumsuz duygular yaşadığını, akran başarısı karşısında duygusal olarak zorlandığını ifade etmiştir. Sanal ortamların öğrenciler üzerinde bu yönde etkileri olduğunu ortaya koyan çalışma alanyazında mevcuttur (Stephan vd., 2019). Bu bulgular, sanal ortamda gerçekleştirilen robotik kodlama etkinliklerinde bazı öğrencilerin başarısızlık veya akran karşılaştırmalarına daha duyarlı olabileceğini göstermektedir. Fiziksel grupta bu tür olumsuz duyguların hiç raporlanmamış olması, yüz yüze ortamın destekleyici etkisi veya duygusal zorlanmaların daha az fark edilmesi ile açıklanabilir.

Fiziksel robotik kodlama grubunda, etkinlikleri yaparken eğlendiğini, robotik problemleri çözmekten keyif aldığını belirten öğrenciler, sanal robotik kodlama grubuna göre çoğunluktadır. Bu sonuç, fiziksel ortamdaki etkinliklerin Python programlama dili öğrenme sürecinde öğrencilerin olumlu duygularını daha çok desteklediğini göstermektedir. Bu durum, yüz yüze etkileşimlerin ve fiziksel materyallerle çalışma fırsatlarının, öğrencilerin duygusal katılımını artırabileceğini düşündürmektedir (Yurdakök, 2022).

Sanal robotik kodlama grubunda, robotik kod doğru çalışınca mutlu olduğunu dile getiren öğrenciler fiziksel robotik kodlama grubuna göre çoğunluktadır. Ancak yalnızca sanal robotik kodlama grubundaki bazı öğrencilerin, hatalı kodu düzeltince çok mutlu olduğunu dile getirmesi, özellikle sanal ortamda yapılan robotik kodlama etkinliklerinin öğrencilerin başarıya ulaşma durumunda olumlu duygular yaşamalarına daha fazla katkı sağladığını göstermektedir. Diğer taraftan, her iki grupta da aynı sayıda öğrenci, yeni bir robotik araç ile yeni bilgiler öğrendiği için mutlu olduğunu dile getirmiştir. Bu durum öğrencilerin yalnızca sanal veya fiziksel ortamdaki robotik aracın çalışmasıyla değil, sürece dair öğrenmelerle de içsel doyum yaşadığını ortaya koymaktadır. Az sayıda da olsa her iki grupta, zor kodlama sürecini tamamlamaktan ve hazır robot sistemiyle çalışma kolaylığından duyduğu memnuniyeti dile getiren öğrenciler olmuştur. Bu sonuç, öğrencilerin zorlayıcı süreçlerden çok doğrudan başarıya ulaştıkları anlara odaklandıklarını düşündürülebilir (Olipas vd., 2021).

Sanal robotik kodlama grubunda, zor bir görevle karşılaştıklarında öğrenmeye yönelik heyecan duyduklarını ifade eden öğrenciler, fiziksel robotik kodlama grubuna göre çoğunluktadır. Bu durum, sanal ortamda gerçekleştirilen robotik kodlama etkinliklerinin öğrencilerde merak ve öğrenme isteği uyandırmada daha etkili olduğunu göstermektedir. Diğer taraftan her iki grupta da aynı sayıda öğrenci yeni robotik bilgileri keşfetme sürecinden duydukları heyecanı dile getirmiştir. Bu sonuç ister sanal ister fiziksel ortamda olsun, yeni bir robotik araçla Python programlama dilini keşfetme sürecinin öğrencilerde heyecan uyandırdığını göstermektedir. Bu durum, robotun pedagojik davranışlarının ve duygusal geri bildirim kapasitesinin öğrenci motivasyonu üzerindeki etkisini vurgulayan literatürle uyumludur (Chen vd., 2020).

Davranışsal Eğilimler: Her iki grupta da aynı sayıda öğrenci, robotik aracın doğru çalışmasını sağlayan kodu elde edene kadar test yaptığını dile getirmiştir. Bu sonuç ister sanal ister fiziksel ortamda robotik kodlama etkinliğine katılsın, her iki gruptaki öğrencilerin de problem çözme sürecinde kararlılık gösterdiklerine işaret etmekte, diğer bir ifadeyle doğru sonuca ulaşıp görevi tamamlama konusundaki kararlı davranışlarını gözler önüne sermektedir. Bu durum, katılımcı profilinin üstün yetenekli öğrencilerden oluşmasından kaynaklanmış olabilir (Kılıçkiran vd., 2020). Buna karşın, sanal robotik kodlama grubunda, hatalı kodu bulmakta zorluk yaşayan öğrenciler fiziksel robotik kodlama grubuna göre çoğunluktadır. Bu bulgu, sanal ortamda robotik kodlama yapan öğrencilerin hata ayıklama sürecinde daha fazla güçlük yaşadıklarını düşündürülebilir. Alanyazında bu durumu ifade eden çalışmalar mevcuttur (Gong vd., 2024; Zhang & Jalal, 2019).

Fiziksel robotik kodlama grubundaki bazı öğrenciler uygun robotik aracı kolayca seçtiğini/bağlantıyı oluşturduğunu ifade etmiştir. Buna karşın sanal robotik kodlama grubundaki öğrenciler bu doğrultuda herhangi bir beyanda bulunmamıştır. Bu bulgu, fiziksel ortamda robotik araçlarla doğrudan etkileşim kuran öğrencilerin, uygun aracı seçme ve bağlantı kurma süreçlerini daha somut ve anlaşılır biçimde deneyimlediklerini göstermektedir. Fiziksel robotik araç ile doğrudan temasın, öğrencilerin donanım bileşenlerini daha bilinçli bir şekilde seçmelerini ve sistem farkındalığı geliştirmelerini sağladığı alanyazında dile getirilmiştir (Chalmers, 2018; Kılıçkiran vd., 2020). Diğer yandan, sanal grup katılımcılarının bu sürece dair ifade belirtmemesi, çevrim içi ortamda araçlarla fiziksel temasın olmamasının, uygun robotu belirleme ve bağlantı kurma süreçlerinin fark edilmesini zorlaştırabileceğini düşündürmektedir. Bu durum, uygulamalı robotik etkinliklerde fiziksel ortamın tercih edilmesinin, öğrencilerin araç seçimi ve bağlantı kurma gibi teknik becerileri geliştirmeleri açısından destekleyici bir rol oynayabileceğini ortaya koymaktadır. Sanal arayüzlerin otomatik bağlantı gibi özellikleri bu

farkındalığı sınırlandırabilmektedir (Berland & Wilensky, 2015). Fiziksel ortamların sistem farkındalığını güçlendirirken sanal ortamların soyutlama becerilerini öne çıkardığı alanyazında da vurgulanmaktadır (Zhong, 2023).

Temsili/Aracı Katılımlar: Her ne kadar robotik kodlama sürecine katılan her iki gruptaki öğrenciler, nicel bulgularda ortaya çıktığı üzere, tüm görevleri tamamlama performansı sergilemiş olsalar da, yalnızca sanal robotik kodlama grubundaki bir öğrencinin sorumluluk bilinciyle etkin katılım sağladığını dile getirmesi şaşırtıcıdır. Fiziksel robotik kodlama grubunda ise bu kod hiç gözlenmemiştir. Bu durum, sanal ortamda görev bilinciyle hareket ederek bireysel katılım sergileyen öğrencilerin sınırlı da olsa varlık gösterdiğini ortaya koymaktadır. Söz konusu bulgu, ister sanal ister fiziksel ortamda olsun, öğrencilerin robotik etkinliklerde sorumluluk alma davranışlarını doğrudan dile getirme eğiliminde olmadıklarını ve bu becerinin görünür kılınması için daha fazla yapılandırılmış yönlendirme ve görev paylaşımına ihtiyaç duyulabileceğini düşündürmektedir (Fung vd., 2025; Verner vd., 2022).

Fiziksel robotik kodlama grubunda etkinlikleri doğru şekilde tamamlamak için yardıma ihtiyacı olduğunda öğretmene çekinmeden soru sorduğunu dile getiren öğrenci sayısı, sanal robotik kodlama grubundakilere göre çoğunluktadır. Her iki gruptaki robotik etkinliklerde de öğretmenin öğrencilerle aynı laboratuvar ortamında bulunduğu ve öğrencilerin öğretmene doğrudan erişiminin sağlandığı düşünüldüğünde bu bulgu şaşırtıcıdır. Bu durum, fiziksel ortamdaki robotik araçla gerçekleştirilen somut/dokunmatik etkileşimin, öğrencilerin bulundukları öğrenme çevresinin gerçekliğinin farkında olmalarını sağlayarak, yardım isteme davranışlarını daha rahat sergilediklerini göstermektedir. Diğer yandan, sanal ortamdaki robotik aracı kullanırken öğrencilerin bu çevrimiçi ve 3B ortamın içine dalıp (immersion), bulundukları öğrenme çevresinin gerçekliğini unutmalarına neden olarak, yardım isteme davranışlarını daha sınırlı düzeyde sergilediklerini düşündürmektedir (Alsoliman, 2022; Berland & Wilensky, 2015; Rukangu vd., 2025).

Sanal robotik kodlama grubunda robotik kodlama süreciyle ilgili olumlu duygularını ifade eden öğrencilerin sayısı fiziksel robotik kodlama grubundakilere göre çoğunluktadır. Bu bulgu, her iki gruptaki öğrencilerin de olumlu duygu/düşüncelerini ifade etmeye yatkın olduğunu, özellikle sanal robotik kodlama grubunda öğrencilerin bu tür duygu/düşünceleri dile getirmekte daha belirgin bir şekilde öne çıktığını göstermektedir. Öte yandan, olumsuz duygu/düşünce ifadesi yalnızca sanal robotik kodlama grubunda bir kez gözlemlenmiştir. Olumsuz duygu/düşünce paylaşımının her iki grupta da düşük düzeyde kalması, öğrencilerin negatif duygu/düşünceleri dile getirme konusunda çekingen olabileceklerine ya da bu duygu/düşüncelerin yeterince fark edilmemiş olabileceğine işaret etmektedir. Bu çerçevede,

gerek sanal gerekse fiziksel robotik kodlama gibi problem çözme temelli öğrenme ortamlarında öğrencilerin duygu/düşüncelerini açıkça ifade etmelerini destekleyici iletişim ortamlarının oluşturulması; duygu/düşüncelerine yönelik farkındalık ve düzenleme becerilerinin gelişimi açısından önemli olabilir (Ağır, 2022; Ahmadzadeh & Moghadam, 2024).

Sosyal Etkileşimler: Sanal robotik kodlama grubunda öğretmenden yardım istediğini dile getiren öğrenci sayısı fiziksel robotik kodlama grubundakilere göre çoğunluktadır. Bu bulgu, sanal grup katılımcılarının yardım ihtiyacını özellikle öğretmene yöneltme eğiliminde olduklarını göstermektedir. Bu durum, çevrim içi ortamlarda öğrencilerin bireysel olarak karşılaştıkları sorunlarda doğrudan öğretmene yönelmeyi tercih ettiklerini ve belirsizlik karşısında öğretici desteğine daha fazla ihtiyaç duyduklarını düşündürmektedir. Alanyazında bu durumu destekleyen çalışmalar mevcuttur (Flowers, 2024). Fiziksel robotik kodlama grubu katılımcılarının ise öğretmenden yardım isteme davranışının az olması, öğretmene doğrudan ulaşmanın olağan bir süreç olarak algılanmasından dolayı ifade edilmeye gerek görülmemesiyle açıklanabilir. Alanyazında bu durumun yüz yüze etkileşimin doğallığı ve bu talebin söze dökülme gereksiniminin az olmasıyla açıklayan çalışmalara rastlanmıştır (Topping vd., 2021). Diğer taraftan her iki grupta da az sayıda öğrenci arkadaşlarından yardım istediğini dile getirmiştir. Bu durum, her iki grupta da az sayıda öğrencinin daha fazla akran etkileşimi üzerinden çözüm arama eğiliminde olduğunu, diğer bir ifadeyle her iki gruptaki pek çok öğrencinin çok da fazla akran desteği arayışında olmadığını göstermektedir. Alanyazında sanal ortamlarda öğrencilerin yardım isteme davranışını daha az sergilediğine dair bulgular da mevcuttur (Schulz vd., 2023). Ayrıca sanal ortamlarda akran etkileşimi için gerekli araçlar olmadığında akan desteğinin düşebileceği de alanyazında vurgulanmıştır (Topping vd., 2021).

Sanal robotik kodlama grubunda yalnızca bir öğrenci akran baskısı nedeniyle duygusal zorlanma yaşadığını dile getirmiş, fiziksel robotik kodlama grubunda ise böyle bir bulguya rastlanmamıştır. Bu bulgu, sanal ortamdaki robotik etkinliklerde az da olsa akran baskısının öğrenciler üzerinde duygusal etkiler yaratabileceğini göstermektedir. Alanyazında sanal ortamların sosyal karşılaştırma ve akran baskısı gibi duygusal zorluklar da yaratabildiği belirtilmiştir (Nuryana vd., 2023). Fiziksel robotik kodlama grubunda bu tür bir duygusal zorlanmanın ifade edilmemesi ise, öğrencilerin yüz yüze ortamda sosyal baskıyı daha kolay yönetebildiklerini ya da bu baskının daha az hissedilir düzeyde yaşandığını düşündürebilir. Bununla birlikte, iki grup arasındaki bu düşük frekans farkı dikkat çekicidir ve akran baskısının etkilerinin her iki ortamda da açıkça ifade edilmediğini göstermektedir. Bu durum, öğrencilerin duygusal zorlanmalarını paylaşma konusunda çekingen davranabileceklerini ya da bu tür deneyimlerin farkında olmadan içselleştirilebileceğini düşündürmektedir. Öğrencilerin

olumsuz duyguları paylaşmadaki çekingenliği, öğretmenlerin bu tür sinyalleri fark etmesi için düzenli duygu taramaları gibi yöntemlerin önemini ortaya koymaktadır (Stephan vd., 2019; Zhong, 2023).

Çalışmada ortaya çıkan tüm sonuçlar değerlendirildiğinde; sanal veya fiziksel robotik kodlama aracıyla uzun sürece yayılarak yapılan tüm haftalardaki etkinliklerde, her iki gruptaki üstün yetenekli öğrencilerin kazanımlara ilişkin görevleri yerine getirdikleri, öğrenme süreciyle duyuşsal, bilişsel, davranışsal, temsili/aracı, sosyal tüm boyutlarda yüksek düzeyde meşgul oldukları, Python programlama dili başarılarının arttığı ve bu başarıyı devam ettiren bilginin kalıcılığının sağlandığı, programlama öz-yeterliklerinin yükseldiği, programlama kaygılarının ise düştüğü saptanmıştır. Çalışmada her iki gruptaki öğrencilerin bilişsel, duyuşsal, davranışsal, temsili ve sosyal kazanımlarının birlikte değerlendirilmesi de, robotik kodlama eğitiminin bütüncül etkilerini daha kapsamlı biçimde gözler önüne sermektedir.

Bu çalışmada kullanılan haftalık robotik kodlama etkinlik planları; MEB Python programlama dili ve BİLSEM robotik kodlama hedef kazanımlarının dikkate alınarak sanal olarak VEXcode-VR ve fiziksel olarak Raspberry-Pi robotik kodlama araçlarına uygun hazırlanıp uygulanmış ve tüm süreç araştırmacı olan öğretmen tarafından kontrollü bir şekilde yönetilmiştir. Bu durumun çalışmadaki olumlu sonuçlara erişmede etkili olduğu söylenebilir.

Fiziksel robotik araçlar, öğrencilerin somut ve dokunsal deneyimler yoluyla programlama kavramlarını öğrenmelerine olanak tanınması açısından önemli pedagojik fırsatlar sunmaktadır. Bununla birlikte, bu araçların yüksek maliyetleri, bakım ve onarım gereksinimleri, donanım arızaları, sınırlı sayıda öğrenciyle eş zamanlı kullanım zorluğu ve fiziksel mekâna bağımlılık gibi çeşitli dezavantajları bulunmaktadır. Bu durum, özellikle kaynakları sınırlı olan eğitim kurumlarında robotik tabanlı programlama öğretiminin sürdürülebilirliğini zorlaştırmaktadır (Berland & Wilensky, 2015). Sanal robotik ortamlar ise, öğrencilerin herhangi bir fiziksel donanıma ihtiyaç duymadan robotik sistemleri programlamalarına, algoritmalarını test etmelerine ve hata ayıklama süreçlerini risksiz bir şekilde deneyimlemelerine olanak tanımaktadır. Ayrıca bu ortamlar, internet erişimi olan her yerden kullanılabilmesi sayesinde erişilebilirlik ve fırsat eşitliği açısından önemli avantajlar sağlamaktadır (Alsoliman, 2022). Bu bağlamda, sanal robotik araçlar ve simülasyon tabanlı robotik ortamlar, fiziksel robotik araçlara erişimde yaşanan bu sınırlılıkları büyük ölçüde ortadan kaldıran etkili bir alternatif olarak öne çıkmaktadır.

Literatürde yer alan çalışmalar, sanal robotik araçların programlama öğretiminde akademik başarı, bilgi işlemsel düşünme becerileri ve öğrenci katılımı açısından fiziksel robotik araçlara benzer düzeyde etkili olabildiğini göstermektedir. Sanal robotik araçların bir diğer

önemli avantajı, donanım kaynaklı zaman kayıplarını ortadan kaldırmasıdır. Fiziksel robotik etkinliklerde karşılaşılan bağlantı sorunları, sensör arızaları veya montaj hataları, öğrencilerin programlama sürecinden uzaklaşmasına neden olabilmektedir. Sanal ortamlarda ise öğrenciler doğrudan kodlama ve algoritmik düşünme süreçlerine odaklanabilmekte, bu durum da öğrenme sürecinin bilişsel verimliliğini artırmaktadır (Zhong, 2023). Sonuç olarak, fiziksel robotik araçların maliyet, erişim ve teknik sınırlılıkları göz önünde bulundurulduğunda, sanal robotik araçların programlama dili öğretiminde etkili, ekonomik ve erişilebilir bir alternatif sunduğu söylenebilir. Özellikle başlangıç düzeyindeki öğrenciler için sanal robotik ortamlar, programlama mantığını kavrama, hata ayıklama becerisi geliştirme ve özgüven kazanma açısından önemli bir pedagojik araç olarak değerlendirilmektedir. Bu nedenle, robotik tabanlı programlama öğretiminde sanal ve fiziksel robotik araçların tamamlayıcı ya da gereksinime dayalı alternatif yaklaşımlar olarak ele alınması, öğrenme ortamlarının niteliğini artırabilir.

Bu çalışma; ele alınan değişkenlere göre üstün yetenekli ortaokul öğrencileriyle yapılan sanal (VEXcode-VR) ve fiziksel (Raspberry-Pi) robotik kodlama etkinliklerini nicel ve nitel olarak karşılaştırması yönüyle özgündür. Bu kapsamda araştırma, her iki gruptaki öğrencilerin haftalık olarak edindikleri kazanımları ortaya koyması, meşguliyeti alt boyutlarıyla değerlendirmesi, Python programlama dili başarısının yanı sıra bu başarıya ilişkin bilginin kalıcılığını (hatırlanmasını) da sorgulaması, programlama öz-yeterliliğini ve kaygıyı da ele alması ve tüm bu değişkenler arası ilişkileri incelemesi açısından BİLSEM öğretmenlerine ve MEB yetkililerine nicel ve nitel somut bulgular ortaya koymaktadır. Bunun yanında yapılan alanyazın taramalarında her iki robotik ortam için hazırlanmış geçerliği ve güvenirliği yapılmış başarı testine rastlanmamıştır. Bu araştırma için Python programlama dilini temel alan, sanal ve fiziksel robotik kodlama ortamları için ortak kazanımları içeren geçerliği ve güvenirliği test edilerek birbirine paralel ve denk olacak şekilde iki ayrı başarı testinin geliştirilmesi ve alanyazına kazandırılması da bu araştırmanın diğer özgün yanı olarak gösterilebilir. Ancak her iki gruptaki etkinliklerin az sayıda öğrenciyle gerçekleştirilmiş olması, ortaya çıkan sonuçların genellenmesini sorgulatabileceği için, ileriki çalışmalarda daha geniş öğrenci gruplarıyla araştırmalar yürütülebilir ve bu durum test edilerek daha kapsamlı sonuçlar elde edilebilir. Ayrıca bu çalışmada olduğu gibi sanal ve fiziksel robotik etkinliklerin öğrenci özellikleri, içerik ve kazanım, uygulanma şekli ve bağlam açısından etkilerini birlikte değerlendirmenin gelecek araştırmalara yol gösterici olduğu düşünülmektedir.

Öneriler

Araştırmada elde edilen nicel ve nitel bulgular birlikte değerlendirildiğinde, sanal (VEXcode-VR) ve fiziksel (Raspberry-Pi) robotik kodlama eğitimi almış üstün yetenekli ortaokul öğrencilerinin Python programlama dili öğrenme süreçlerinde güçlü ve tamamlayıcı pedagojik artılarının ortaya çıktığı görülmektedir. Fakat bu eğitsel katkıların etkili ve sürdürülebilir biçimde kullanılabilmesi, öğretim tasarımı, uygulama koşulları, eğitim politikası düzeyinde düzenlemeler, araştırma yaklaşımları ve öğrenme ortamlarının tasarımı özellikleriyle yakından ilişkilidir. Bu doğrultuda aşağıda yer alan öneriler, farklı paydaş grupları temel alınarak sunulmuştur.

Bilim ve Sanat Merkezi Öğretmenlerine/ Uygulayıcılara Yönelik Öneriler

- Üstün yetenekli ortaokul öğrencilerinin Python programlama dilini temel alan sanal ve fiziksel robotik kodlama kazanımlarını edinmeleri için; BİLSEM atölyelerinde uygulamaya konan ve MEB (2020, 2021, 2023) tarafından hazırlanmış öğretim programlarının dikkate alınarak haftalık etkinliklerin detaylı bir şekilde hazırlanması önerilir.
- Üstün yetenekli ortaokul öğrencilerinin sanal ve fiziksel robotik kodlama sürecindeki duyuşsal, bilişsel, davranışsal, temsili ve sosyal meşguliyet boyutlarındaki değişimlerin haftalık olarak düzenli bir şekilde gözlenmesi önerilir.
- Üstün yetenekli ortaokul öğrencilerinin bilişsel özellikleri göz önünde bulundurularak, sanal ve fiziksel robotik kodlamada edindikleri bilgi, beceri ve kazanımların kalıcılığını sağlamak için, etkinliklerin uzun sürece yayılması önerilir.
- Hem sanal hem de fiziksel robotik kodlama sürecinde üstün yetenekli ortaokul öğrencilerinin yardım isteme, sosyal etkileşim, olumlu/olumsuz duygu ve düşüncelerini dile getirme gibi davranışlarını destekleyici bir iletişim iklimi oluşturulması önerilir.

Milli Eğitim Bakanlığı Yetkililerine Yönelik Öneriler

- BİLSEM öğretmenlerinin sanal ve fiziksel robotik kodlama öğretiminde yalnızca ürün odaklı değerlendirme değil, haftalık süreç odaklı değerlendirme anlayışını da benimsemeleri için teşvik edilmeleri önerilir.
- Fiziksel robotik setler için bakım, onarım ve teknik destek mekanizmalarının güçlendirilmesi, alt yapı eksikliklerinin tamamlanması; sanal robotik platformlar için kurumsal erişim, lisanslama ve güvenli kullanım yönergelerinin standartlaştırılması önerilir.

- Sanal robotik kodlama ortamlarının, donanım maliyetleri ve erişim sınırlılıklarını azaltma potansiyeli nedeniyle fırsat eşitliği ve sürdürülebilirlik politikaları kapsamında yaygınlaştırılması, internet alt yapısının iyileştirilmesi önerilir.
- Hem sanal hem de fiziksel robotik kodlama araçlarıyla kullanılabilen, diğer programlama dillerine göre çeşitli avantajları bulunan Python programlama dilini temel alan robotik kodlama eğitimlerinin BİLSEM kurumlarında yaygınlaştırılması önerilir.

Araştırmacılara Yönelik Öneriler

- Bu çalışma, sanal ve fiziksel robotik kodlama etkinliklerine katılan az sayıdaki üstün yetenekli ortaokul öğrencisiyle gerçekleştirilmiştir. Araştırmanın genellenebilirliğini artırmak için daha fazla öğrencinin katıldığı karşılaştırmalı uygulamaların yapılması önerilir.
- Sanal ve fiziksel robotik araçların üstün yetenekli öğrencilerin robotik kodlama performanslarını, duyuşsal, bilişsel, davranışsal, temsili/aracı ve sosyal meşguliyetlerini artırdığını; bilginin kalıcılığını sağladığını, programlama öz-yeterliliğini geliştirdiğini bunun yanı sıra programlama kaygısını düşürdüğünü doğrulamak için karşılaştırmalı yeni araştırmaların yapılması önerilir.
- Üstün yetenekli ortaokul öğrencilerinin bilişsel özellikleri göz önünde bulundurularak, sanal ve fiziksel robotik kodlama etkinliklerinin uzun sürece yayılması ve bu süreçte edindikleri kazanımların sürekliliğini sağlamak için, Python programlama dili bilgisi kalıcılığının (bilginin hatırlanması) tespit edilmesi önerilir.
- Farklı özelliklere sahip robotik araçlarla kullanılabilen ve diğerlerine göre çeşitli avantajları bulunan Python programlama dilinin üstün yetenekli öğrencilere sanal veya fiziksel robotik kodlama araçlarıyla öğretime odaklanan karşılaştırmalı araştırma sayısının artırılması önerilir.
- Sanal ve fiziksel robotik kodlama sürecini 21. yüzyıl öğrenen özellikleri temelindeki çeşitli değişkenler açısından inceleyen, nicel ve nitel verilerle desteklenmiş karma araştırma sayısının artırılması önerilir.
- Çalışmada sanal ve fiziksel robotik kodlamaya yönelik geliştirilen iki ayrı başarı testinin ve haftalık süreç değerlendirme araçlarının geçerliğinin ve güvenilirliğinin farklı örneklemelerde yeniden sınanması önerilir.

KAYNAKÇA

- AASL. (2018, 06 02). *American Association of School Librarians*. AASL: <https://www.ala.org/aasl> adresinden alındı.
- Ağır, A. (2022). Investigation of the relationships between the attitudes of secondary schoolstudents towards robotic activities and the levels of self-regulation,metacognitive awareness. *Journal of Educational Technology& Online Learning*, 5(4), 963-980.
- Ahmadzadeh, L., & Moghadam, K. D. (2024). Exploring the impact of robotics-based learning on students' emotional intelligence and academic achievement: A semi-experimental study. *Interdisciplinary Journal of Virtual Learning in Medical Sciences*, 15(4), 355-368.
- Akbüker, B. A., Erdik, E., Güney, H., Çimşitoğlu, G., & Akbüker, C. (2019). Bilim ve sanat merkezlerinde özel yetenekli öğrencilerin sorunlarının değerlendirilmesinde bir yöntem önerisi “özel Yetenekli öğrenci çalıştayı. *Üstün Zekâlılar Eğitimi ve Yaratıcılık Dergisi*, 6(1), 22-39.
- Akbıyık, N. (2019). *Programlama eğitiminde Arduino mikro denetleyici uygulamaları kullanımının lise öğrencilerinin programlama öz-yeterlikleri ve problem çözme becerileri üzerine etkisi* (Tez No. 592578) [Yüksek Lisans Tezi,Gazi Üniversitesi-Ankara].Yüksek Öğretim Kurumu Ulusal Tez Merkezi.
- Aksoy, Ö. (2014). Üstün zekalı ve normal öğrencilerin algılanan akademik öz-yeterliklerinin karşılaştırılması. *Üstün Zekâlılar Eğitimi ve Yaratıcılık Dergisi*, 1(1), 1-6.
- Alabbasi, A. M., Karwowski, M., & Ayoub, A. E. (2022). Self-efficacy in gifted and non-gifted students: A multilevel meta-analysis. *American Psychological Association 6th edition*, 6(2), 1-31.
- Aldridge, J. M., & Fraser, B. J. (2016). Teachers' views of their school climate and its relationship with teacher self-efficacy and job satisfaction. *Learning Environments Research*, 19(1), 291-307.
- Alimisis, D. (2013). Educational robotics: Open questions and new challenges. *Themes in Science & Technology Education*, 6(1), 63-71.
- Alsoliman, B. S. (2022). Virtual robotics in education: The experience of eighth grade students in STEM. *Frontiers in Education* 7(1), 1-12.
- Altıok, S., & Üçgül, M. (2024). Effects of robotic coding on computational thinking skills of secondary school students. *Canadian Journal of Learning and Tecnology*, 50(2), 1-24.
- Alvarez, A., & Larranaga, M. (2016). Experiences incorporating Lego Mindstorms robots in the basic programming syllabus: lessons learned. *Journal of Intelligent & Robotic Systems*, 81(1), 117-129.
- Anderson, L., Krathwohl, D., Airasian, P., Cruikshank, K., Mayer, R., Pintrich, P., . . . Wittrock, M. (2001). *Taxonomy for learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives*. NY: Addison Wesley Longman.
- Armoni, M., Meerbaum-Salant, O., & Ben-Ari, M. (2015). From Scratch to “Real” Programming. *ACM Transactions on Computing Education (TOCE)*, 14(4), 1-25.

- Askar, P., & Davenport, D. (2009). An investigation of factors related to self-efficacy for Java programming among engineering students. *The Turkish Online Journal of Educational Technology*, 8(1), 26-32.
- Atmatzidou, S., & Demetriadi, S. (2016). Advancing students' computational thinking skills through educational robotics: A study on age and gender relevant differences. *Robotics and Autonomous Systems*, 75, 661-670.
- Avcı, E. (2021). *STEM eğitime uygun tasarlanmış robotik kodlama etkinliklerinin üstün yetenekli öğrencilerin robotik ve kodlamaya karşı tutumuna etkisinin belirlenmesi* (Tez No: 666214). [Yüksek Lisans Tezi, İnönü Üniversitesi-Malatya]. Yüksek Öğretim Kurulu Ulusal Tez Merkezi.
- Büyüköztürk, Ş., Çakmak, E. K., Akgün, Ö. E., Karadeniz, Ş., & Demirel, F. (2023). *Eğitimde bilimsel araştırma yöntemleri*. Pagem Akademi.
- Büyüköztürk, Ş., Köklü, N., & Çokluk, Ö. (2024). "Sosyal bilimler için istatisti" k. Pagem Akademi.
- Balon, B., & Simić, M. (2019). Using Raspberry-Pi computers in education. *Mipro*, 20(24), 671-676.
- Bandura, A. (1997). *Self-efficacy: The exercise of control*. W H Freeman/Times Books/ Henry Holt & Co.
- Bakırcı, F. (2019). *Blok tabanlı programlama aracının 6. sınıf öğrencilerinin programlama başarıları algoritma geliştirme öz-yeterlikleri ve güdülemelerine etkisi* (Tez No: 585158). [Yüksek Lisans Tezi, Sakarya Üniversitesi-Sakarya]. Yüksek Öğretim Kurulu Ulusal Tez Merkezi.
- Belt, E. S., Borup, J., West, R. E., & Lowenthal, P. R. (2021). Faculty perceptions of using synchronous video-based communication technology. *Online Learning*, 25(4), 74-103.
- Benitti, F. B. (2012). Exploring the educational potential of robotics in schools: A systematic review. *Computers & Education*, 58(1), 978-988.
- Berland, M., & Wilensky, U. (2015). Comparing virtual and physical robotics environments for supporting complex systems and computational thinking. *Journal of Science Education and Technology*, 24(1), 628-647.
- Bers, M. U. (2020). *Programming and computational thinking in the early childhood classroom*. Routledge.
- Cam, E., & Kıyıcı, M. (2022). The impact of robotics assisted programming education on academic success, problem solving skills and motivation. *Journal of Educational Technology & Online Learning*, 5(1), 47-65.
- Cam, E., & Kilicer, K. (2022). The effect of virtual reality assisted robotics coding teaching on spatial visualization and coding skills. *Journal of Theory and Practice in Education*, 18(2), 68-84.
- Chalmers, C. (2018). Robotics and computational thinking in primary school. *International Journal of Child-Computer Interaction*, 17(1), 93-100.
- Chao, P.-Y. (2016). Exploring students' computational practice, design and performance of problem-solving through a visual programming environment. *Computers & Education*, 95(1), 202-215.
- Chen, H., Park, H. W., & Breazeal, C. (2020). Teaching and learning with children: Impact of reciprocal peer learning with a social robot on children's learning and emotive engagement. *Computers & Education*, 150(1), 1-19.

- Cheung, K. (1990). *Meaningful measurement in the classroom using the rash model: Some exemplars*. 13 Aralık 2017 tarihinde ERIC veritabanından alınmıştır (ED 326544). <https://files.eric.ed.gov/fulltext/ED326544.pdf>.
- Coşkunserçe, O. (2023). Comparing the use of blockbased and robot programming in introductory programming education: Effects on perceptions of programming self-efficacy. *Wiley Periodicals LLC*, 31(1), 1234–1255.
- Connolly, C., Murphy, E. M., & Moore, S. (2007). Second chance learners, supporting adults learning computer programming. *International Conference on Engineering Education*, 3(7), 1-5.
- Creswell, W. (2010). *Karma yöntem araştırmalarına giriş (Çev:Mustafa Sözbilir)*. Pegem Akademi.
- Çınar, M., & Tüzün, H. (2020). Comparison of object-oriented and robot programming activities: The effects of programming modality on student achievement, abstraction, problem solving, and motivation. *Journal of Computer Assisted Learning*, 37(2), 370-386.
- Çalışkan, E. (2020). The effects of robotics programming on secondary school students' problem-solving skills. *World Journal on Educational Technology: Current Issues*, 12(4), 217-230.
- Çavaş, B., & Huyugüzel-Çavaş, P. (2004,11-13 Şubat). *Teknoloji tabanlı öğrenme: Robotics club*[Yazılı Bildiri]. Akademik Bilişim Konferansları,İzmir.
- Çavuş, H., & Günbatar, M. S. (2008). Bilgisayar kaygı ölçeğinin Türkçeye uyarlama çalışması. *Gazi Eğitim Fakültesi Dergisi*, 28(3), 147-163.
- Çayır, E. (2010). *Lego-Logo ile desteklenmiş öğrenme ortamının bilimsel süreç becerisi ve benlik algısı üzerine etkisinin belirlenmesi (Tez No: 265835)*. [Yüksek Lisans Tezi, Sakarya Üniversitesi-Sakarya]. Yüksek Öğretim Kurulu Ulusal Tez Merkezi.
- Çelik, H., Baykal, N. B., & Memur, H. N. (2020). Nitel veri analizi ve temel ilkeleri. *Eğitimde Nitel Araştırmalar Dergisi*, 8(1), 379-406.
- Çiftçi, S., & Bildiren, A. (2020). The effect of coding courses on the cognitive abilities andproblem-solving skills of preschool children. *Computer Science Education*, 30(1), 2-21.
- Çubukçu, Z., & Tosuntaş, Ş. B. (2018). Üstün yetenekli/zekâlı öğrencilerin eğitiminde teknolojinin yeri. *Elektronik Eğitim Bilimleri Dergisi*, 7(13), 1-13.
- De la Hoz, A., Melo, L., Canada, F., & Cubero, J. (2024). Educational robotics for science and mathematics teaching: Analysis of pre-service teachers' perceptions and self-confidence. *Heliyon*, 10(1), 1-18.
- Demir, F. (2015). *Programlama öğretiminde eğitsel programlama dilinin farklı kullanımlarının programlama başarısı ve kaygısına etkisi (Tez No: 429631)*. [Doktora Tezi, Atatürk Üniversitesi-Erzurum]. Yüksek Öğretim Kurulu Ulusal tez Merkezi.
- Denzin, N. K. (2009). *The Research Act: A theoretical introduction to sociological methods*. Routledge.
- Dinçer, A. (2018). *6.sınıf öğrencilerine Scratch ve Kodu Game Lab programlama dillerinin öğretiminde öğrencilerin tutum, öz yeterlilik ve akademik başarılarının karşılaştırılması (Tez No: 512965)*. [Yüksek Lisans Tezi,Dokuz Eylül Üniversitesi-İzmir].Yüksek Öğretim Kurumu Ulusal Tez Merkezi.

- Doğan, M., & Korkmaz, Ö. (2025). Proje tabanlı eğitsel robot etkinliklerinin ortaokul öğrencilerin programlamaya dönük öz yeterliliklerine ve bilgisayarca düşünme becerilerine etkisi. *Türkiye Sosyal Araştırmalar Dergisi*, 29(2), 68-592.
- Durak, H. Y., Yılmaz, F. G., & Yılmaz, R. (2018). Robot tasarımı etkinliklerinin programlama öğretiminde kullanılmasıyla ilgili ortaokul öğrencilerinin görüşlerinin incelenmesi. *Ege Eğitim Teknolojileri Dergisi*, 2(2), 32- 43.
- Eraytaç, Ö. F. (2019). *Robotik kodlama eğitiminde blok tabanlı kodlama yönteminin ortaokul öğrencilerinin akademik başarısına etkisi* (Tez No: 562699). [Yüksek Lisans Tezi, Çukurova Üniversitesi-Adana]. Yüksek Öğretim Kurulu Ulusal Tez Merkezi.
- Ergin, H. (2019). *Programlama dersinde proje kullanımının öğrencilerin bilgi işlemsel düşünme becerilerine ve programlama öz yeterlilik inancına etkisi* (Tez No: 561297). [Yüksek Lisans Tezi, Ege Üniversitesi-İzmir]. Yüksek Öğretim Kurumu Ulusal Tez Merkezi.
- Erol, O. (2020). How do students' attitudes towards programming and self-efficacy in programming change in the robotic programming process? *International Journal of Progressive Education*, 16(4), 13-26.
- Erol, O., Aköz, O., & Güngör, R. (2024). Examination of relationship between secondary school students' robotic coding attitudes and STEM career interests. *International Journal of Technology in Education*, 7(4), 814-828.
- Ersoy, H., Madran, R. O., & Gülbahar, Y. (2011). Programlama dilleri öğretimine bir model önerisi: Robot programlama. *XIII. Akademik Bilişim Konferansı Bildirileri*. Malatya: İnönü Üniversitesi.
- Fan, G., Liu, D., Zhang, R., & Pan, L. (2025). The impact of AI-assisted pair programming on student motivation, programming anxiety, collaborative learning, and programming performance: A comparative study with traditional pair programming and individual approaches. *International Journal of STEM Education*, 12(16), 1-17.
- Fatsa, Ö. F., & Turan, Z. (2022). A comparative analysis of educational robotics kits. *Instructional Technology and Lifelong Learning*, 3(2), 144-175.
- Fernando, G., & Gonzalez, P. (2024). A hybrid physical-virtual educational robotic arm. *ASEE ASEE Computers in Education Journal*, 14(1), 1-19.
- Field, A. (2013). *Discovering statistics using IBM SPSS statistics* (4th ed.). SAGE Publications.
- Flowers, L. O. (2024). Virtual laboratories in STEM. *International Journal of Science and Research Archive*, 12(02), 1573–1576.
- Fraenkel, J. R., Wallen, N. E., & Hyun, H. H. (2012). *How to design and evaluate research in education*. McGraw-Hill.
- Fredricks, J. A., Blumenfeld, P. C., & Paris, A. H. (2004). School engagement: Potential of the concept, state of the evidence. *Review of Educational Research*, 74(1), 59-109.
- Fredricks, J. A., Blumenfeld, P., Friedel, J., & Paris, A. (2005). School engagement. *What do children need to flourish? Conceptualizing and measuring indicators of positive development*, 305-321.
- Fung, K. Y., Fung, K. C., Lui, T. L., Sin, K. F., Lee, L. H., Qu, H., & Song, S. (2025). Exploring the impact of robot interaction on learning engagement: a comparative study of two multi-modal robots. *Smart Learning Environments*, 12(12), 1-25.

- Gülbahar, Y., Kert, S. B., & Kalelioğlu, F. (2019). Bilgi işlemsel düşünme becerisine yönelik öz yeterlik algısı ölçeği: geçerlik ve güvenirlik çalışması . *Turkish Journal of Computer and Mathematics Education* , 10(1), 1-29.
- Gülyüz, B. G. (2019). Ortaokul öğrencilerinin ders içi robotik kodlama etkinliklerinin blok tabanlı programlamaya ilişkin öz yeterlilik algısına etkisi ve robotik kodlama hakkındaki görüşleri (Tez No: 591569). [Yüksek Lisans Tezi,Uludağ Üniversitesi-Bursa].Yüksek Öğretim Kurumu Ulusal Tez Merkezi.
- Gündoğdu, B. (2020). Meslek lisesi öğrencilerine Lego robotikle algoritma öğretiminin bilgisayarca düşünme, bilişsel yük ve başarıya etkisi (Tez No: 626701). [Yüksek Lisans Tezi,Atatürk Üniversitesi-Erzurum].Yüksek Öğretim Kurulu Ulusal Tez Merkezi.
- Güngördü, N. (2021). Bilişim teknolojileri öğretmenlerinin genel özyeterlikleri ile uzaktan eğitimde kodlama etkinliklerindeki özyeterlik algılarının incelenmesi (Tez No: 665887). [Yüksek Lisans Tezi,Mimar Sinan Güzel Sanatlar Üniversitesi-İstanbul].Yüksek Öğretim Kurulu Ulusal Tez Merkezi.
- Gürkez, Ş. (2021). Ortaokul öğrencilerinin robotik kodlama eğitiminin üst biliş beceri farkındalığı ve öğrenmeye yönelik sorumlulukları üzerine etkisi: Abilix Krypton 7 Örneği (Tez No: 658845). [Yüksek Lisans Tezi,Necmettin Erbakan Üniversitesi-Konya].Yüksek öğretim Kurulu Ulusal Tez Merkezi.
- Gandy, E. A., Bradley, S., Brookes, D. A., & Allen, N. R. (2010). The use of LEGO Mindstorms NXT robots in the teaching of introductory java programming to undergraduate students. *Innovation in Teaching and Learning in Information and Computer Sciences*, 9(1), 2-9.
- Gegenfurtner, A., Quesada-Pallarès, C., & Knogler, M. (2014). Digital simulation-based training: A meta-analysis. *British Journal of Educational Technology*, 45(6).
- Gong, X., Xu, W., Qiao, A., & Li, Z. (2024). Unravelling the computational thinking and spatial thinking development: An exploration of a virtual robot programming environment. *Journal of Computer Assisted Learning*, 41(1), 1-16.
- Greene, J. C., Caracelli, V. J., & Graham, W. F. (1989). Toward a conceptual framework for mixed-method evaluation designs. *Educational Evaluation and Policy Analysis*, 11(3), 255-274.
- Hamiti, M., Kosarenko, N. N., Şabanov, G. A., Tantsura, T. A., & Şennikova, N. V. (2021). Opinions of computer and Instructional technologies students about the robotic coding course. *International Journal of Emerging Technologies in Learning*, 16(19), 206–218.
- Hangün, M. E., Kalinkara, Y., Bayer, H., & Tekin, A. (2022). Eğitimde robotik kullanımına yönelik araştırmaların incelenmesi: Bir içerik analizi çalışması. *Erzincan Üniversitesi Eğitim Fakültesi Dergisi*, 24(3), 558-578.
- Hsu, T.-C., & Hwang, G.-J. (2023). Interaction of visual interface and academic levels with young students' anxiety, playfulness, and enjoyment in programming for robot control. *Universal Access in the Information Society*, 22(1), 213-225.
- Hsu, T.-C., Chang, C., Wu, L.-K., & Looi, C.-K. (2022). Effects of a pair programming educational robot-based approach on students' interdisciplinary learning of computational thinking and language learning. *Frontiers in Psychology*, 13(1), 1-15.
- Ünal, A. (2019). Sosyal bilimler lisesi öğrencilerine blok tabanlı proprogramlama öğretimini kaygı, bilişsel yüklerine ve akademik başarılarına etkisi (Tez No: 551146). [Yüksek Lisans Tezi,Atatürk Üniversitesi-Erzurum].Yüksek Öğretim Kurulu Ulusal Tez Merkezi.

- ISTE. (2016, 06 02). *ISTE standarts for students*. ISTE: <https://iste.org/standards> adresinden alındı.
- Johnson, R. B., & Onwuegbuzie, A. J. (2004). Mixed methods research: A research paradigm whose time has come. *Educational Researcher*, 33(7), 14-26.
- Kök, A. B. (2019). *Beşinci sınıf öğrencilerinin grup çalışması ile robotik kodlama deneyimlerinin incelenmesi* (Tez No: 580116). [Yüksek Lisans Tezi, Afyon Kocatepe Üniversitesi-Afyon].Yüksek Öğretim Kurumu Ulusal Tez Merkezi.
- Küçük, D. K., & Sönmez, I. (2023). Kodlama eğitiminde eğitsel robot kullanımının özel yetenekli öğrencilerin bilgi işlemsel düşünme becerilerine etkisi. *Ahmet Keleşoğlu Eğitim Fakültesi Dergisi (AKEF) Dergisi*, 5(3), 1536-1555.
- Kılıç, S. (2022). Robotik programlamanın ön lisans öğrencilerinin bilgi işlemsel düşünme becerisi gelişimine etkisi. *Afyon Kocatepe Üniversitesi Sosyal Bilimler Dergisi*, 24(2), 480-494.
- Kılıçkiran, H., Korkmaz, Ö., & Çakır, R. (2020). Robotik kodlama eğitiminin üstün yetenekli öğrencilere katkısı. *Turkish Journal of Primary Education*, 5(1), 1-15.
- Kahu, E. R. (2011). Framing student engagement in higher education. *Studies in Higher Education*, 38(5), 758-773.
- Kasalak, İ. (2017). *Robotik kodlama etkinliklerinin ortaokul öğrencilerinin kodlamaya ilişkin öz-yeterlik algılarına etkisi ve etkinliklere ilişkin öğrenci yaşantıları* (Tez No: 454911). [Yüksek Lisans Tezi, Hacettepe Üniversitesi-Ankara].Yükseköğretim Kululu Ulusal Tez Merkezi.
- Kaya, N. G. (2013). Üstün yetenekli öğrencilerin eğitimi ve BİLSEM'ler. *Erzincan Üniversitesi Eğitim Fakültesi Dergisi*, 15(1), 115-122.
- Kayhan, O., Korkmaz, Ö., & Çakır, R. (2024). How do computational thinking and logical and math thinking skills predict programming self-efficacy? *Computers in the Schools*, 41(1), 51-73.
- Kert, S. B., Erkoç, M. F., & Yeni, S. (2020). The effect of robotics on six graders' academic achievement, computational thinking skills and conceptual knowledge levels. *Thinking Skills and Creativity*, 38(1), 1-11.
- Koca, S., & Çakır, R. (2022). Effect of educational robotic applications on students' cognitive outcomes. *Behaviour & Information Technology*, 41(15), 3329-3345.
- Kolil, V. K., Muthupalani, S., & Achuthan, K. (2020). Virtual experimental platforms in chemistry laboratory education and its impact on experimental self-efficacy. *International Journal of Educational Technology in Higher Education*, 17(1), 1-22.
- Konyaoğlu, C. (2019). *Robotik kodlama eğitiminin ortaokul öğrencilerinin problem çözme becerilerine etkileri ve öğrencilerin robotik kodlama etkinliklerine ilişkin görüşleri* (Tez No: 587923). [Yüksek Lisans Tezi, Bolu Abant İzzet Baysal Üniversitesi-Bolu].Yüksek Öğretim Kurumu Ulusal Tez Merkezi.
- Koorsse, M., Cilliers, C., & Calitz, A. (2015). Programming assistance tools to support the learning of IT programming in South African secondary schools. *Computers & Education*, 2(1), 162-78.
- Korkmaz, Ö., & Altun, H. (2014). A validity and reliability study of the attitude scale of computer programming learning. *International Journal of Education (MIJE)*, 4(1), 30-43.

- Kroustalli, C., & Xinogalos, S. (2021). Studying the effects of teaching programming to lower secondary school students with a serious game: A case study with Python and CodeCombat. *Education and Information Technologies*, 26(1), 6069–6095.
- Kukul, V., Gökçearslan, Ş., & Günbatır, M. S. (2017). Computer programming self-efficacy scale for secondary school students: Development, validity and reliability. *Journal of Baltic Science Education*, 16(2), 216-227.
- Kula, K. Ş., & Saraç, T. (2016). Üniversite öğrencilerinin gelecek kaygısı. *Mustafa Kemal Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, 13(33), 227-242.
- Lawshe, C. (1975). A quantitative approach to content validity. *Personnel Psychology*, s. 563–575.
- Liang, C., & Du, L. (2025). The impact of educational robotics, virtual, and unplugged coding on EFL learners' problem-solving, computational thinking, and coding skills. *Journal of Educational Computing Research*, 63(1), 1689-1716.
- Madariaga, L., Allendes, C., Nussbaum, M., Barrios, G., & Acevedo, N. (2023). Offline and online user experience of gamified robotics for introducing computational thinking: Comparing engagement, game mechanics and coding motivation. *Computers & Education*, 193(1), 1-24.
- Mayerova, K. (2012). Pilot activities: LEGO WeDo at primary school. *Proceedings of 3rd International Workshop Teaching Robotics, Teaching with Robotics Integrating Robotics in School Curriculum* (s. 32-39). Bratislava: Comenius University in Bratislava, Faculty of Mathematics.
- McMillan, J. H., & Schumacher, S. (2010). *Research in education: Evidence-based inquiry*. London Pearson.
- MEB. (2016). *STEAM Eğitim Raporu*. T.C. Millî Eğitim Bakanlığı, Yenilik ve Eğitim Teknolojileri Genel Müdürlüğü. Ankara: Millî Eğitim Bakanlığı.
- MEB. (2018). *T.C. Milli Eğitim Bakanlığı Bilişim Teknolojileri ve Yazılım Dersi Öğretim Programı*. Ankara: Milli Eğitim Bakanlığı.
- MEB. (2020). *Herkes İçin Python Dili Eğitimi*. T.C. Milli Eğitim Bakanlığı, Öğretmen Yetiştirme ve Geliştirme Genel Müdürlüğü. Ankara: Milli Eğitim Bakanlığı.
- MEB. (2021). *T.C. Milli Eğitim Bakanlığı Bilsem Yönergesi*. MEB Özel Eğitim ve Rehberlik Hizmetleri Genel Müdürlüğü, Bilim ve Sanat Merkezleri Yönergesi Genel Müdürlüğü. Ankara: Milli Eğitim Bakanlığı.
- MEB. (2023). *T.C. Milli Eğitim Bakanlığı Ortaöğretim Programı*. Ankara: Milli Eğitim Bakanlığı.
- MEB. (2025). *T.C. Milli Eğitim Bakanlığı Türkiye Yüzyılı Maarif Modeli*. Ankara: Millî Eğitim Bakanlığı.
- Mészárosóvá, E. (2015). Is Python an appropriate programming language for teaching programming in secondary schools. *2015 ICTE*, 4(2), 5-14.
- Miguel-Alonso, I., Rodriguez-Garcia, B., Checa, D., & Bustillo, A. (2023). Countering the novelty effect: A tutorial for immersive virtual reality learning environments. *Applied Sciences*, 13(1), 1-20.
- Muilwijk, S. E., & Lazonder, A. W. (2023). Learning from physical and virtual investigation: A meta-analysis of conceptual knowledge acquisition. *Frontiers in Education*, 8(1), 1-9.

- Namlu, A. G., & Ceyhan, E. (2003). Bilgisayar kaygısı: Öğretmen adayları üzerinde çok yönlü bir inceleme. *Kuram ve Uygulamada Eğitim Bilimleri*, 1(3), 401-432.
- Numanoğlu, M., & Hafize, K. (2017). Programlama öğretiminde robot kullanımı - Mbot örneği. *Eğitim Fakültesi Dergisi*, 6(2), 497-515.
- Nuryana, Z., Xu, W., Kurniawan, L., Sutanti, N., Makruf, S. A., & Nurcahyati, I. (2023). Student stress and mental health during online learning: Potential for post-COVID-19 school curriculum development. *Comprehensive Psychoneuroendocrinology*, 14(1), 1-8.
- OECD. (2017, 02 06). *OECD (Ekonomik İşbirliği ve Kalkınma Örgütü)*. OECD: <https://www.oecd.org/> adresinden alındı.
- Okuyucu, M. O. (2019). *Robotik kodlama eğitiminin lise öğrencilerinin üstbiliş ve yansıtıcı düşünme düzeyleri üzerindeki etkisinin incelenmesi* (Tez No: 573636). [Yüksek Lisans Tezi,Erzincan Binali Yıldırım Üniversitesi-Erzincan].Yüksek Öğretim Kurulu Ulusal Tez Merkezi.
- Olipas, C. N., Leona, R. F., Villegas, A. C., Cunanan Jr., A. I., & Javate, C. L. (2021). The academic performance and the computer programming anxiety of BSIT students: A basis for Instructional strategy Improvement. *International Journal of Advanced Engineering, Management and Science*, 7(6), 1-5.
- Olpak, T., & Karamete, A. (2024). The relationship between gifted students' self-efficacy perceptions of computational thinking skills and information technologies self-efficacy perceptions. *Necatibey Faculty of Education Electronic Journal of Science and Mathematics Education*, 18(1), 148-174.
- Oruç, Ş., & Çağır, S. (2022). Bilim ve sanat merkezleri üzerine yapılan akademik çalışmaların değerlendirilmesi. *Türkiye Eğitim Dergisi*, 7(2), 398-412.
- Ozmen, B., & Altun, A. (2014). Üniversite öğrencilerinin programlama deneyimleri: Güçlükler ve engeller. *Turkish Online Journal of Qualitative Inquiry*, 5(3), 9-27.
- Özdinç, F., & Altun, A. (2014). Factors effecting information technology teacher trainees' programming. *Elementary Education Online*, 13(4), 1531-1541.
- Raspberry Pi. (2024). *Raspberry Pi*. <https://www.raspberrypi.com/> adresinden alındı.
- Reeve, J. (2012). A Self-determination theory perspective on student engagement. *Handbook of Research on Student Engagement*, 1(1), 149-172.
- Rosenberg, N., & Zvi-Girshin, R. (2024). Enhancing early STEM engagement: The impact of inquiry-based robotics projects on first-grade students' problem-solving self-efficacy and collaborative attitudes. *Education Sciences*, 15(10), 1-22.
- Rossum, V. G. (1995). *Python tutorial*. Centrum voor Wiskunde en Informatica.
- Ru Qu, J., & Fok, P. K. (2022). Cultivating students' computational thinking through student-robot interactions in robotics education. *International Journal of Technology and Design Education*, 32(1), 1983-2002.
- Rukangu, A., Morelock, J., Johnsen, K., & Moyaki, D. (2025). Virtual and physical robots in engineering education: A study on motivation and learning with augmented reality. *ASEE Computers in Education*, 13(4), 1-21.
- Sarı, U., Pektaş, H. M., Şen, Ö. F., & Çelik, H. (2022). Algorithmic thinking development through physical computing activities with Arduino in STEM education. *Education and Information Technologies*, 27(1), 6669-6689.

- Schulz, S., M. McLaren, B., & Pinkwart, N. (2023). Towards a tutoring system to support robotics activities in classrooms – Two Wizard-of-Oz studies. *International Journal of Artificial Intelligence in Education*, 33(1), 688–719.
- Schunk, D. H., & DiBenedetto, M. K. (2020). Motivation and social cognitive theory. *Contemporary Educational Psychology*, 60(1), 361-476.
- Selçuk, N. A. (2019). *Eğitsel robotik uygulamalarının ortaokul öğrencilerinin ders motivasyonları, robotik tutumları ve başarıları açısından incelenmesi* (Tez No: 557781). [Yüksek Lisans Tezi, İstanbul Üniversitesi-İstanbul]. Yüksek Öğretim Kurulu Ulusal Tez Merkezi.
- Selcuk, N. A., Kucuk, S., & Sisman, B. (2024). Does really educational robotics improve secondary school students' course motivation, achievement and attitude? *Education and Information Technologies*, 29(1), 23753–23780.
- Partnership for 21st Century Skills. (2018). *Partnership for 21st Century Skills*. <https://www.p21.org/>.
- Stajkovic, A., & Luthans, F. (1998). Self-efficacy and work-related performance: A meta-analysis. *Psychological Bulletin*, 124(2), 240-261.
- Stasolla, F., Curcio, E., Borgese, A., Passaro, A., Gioia, M. D., Zullo, A., & Martini, E. (2025). Educational robotics and game-based interventions for overcoming dyscalculia: A pilot study. *Computers MDPI*, 14(1), 1-23.
- Stephan, M., Stefan, M., & Gläser-Zikuda, M. (2019). Students' achievement emotions and online learning in teacher education. *Frontiers in Education*, 4(1), 1-12.
- Strawhacker, A., & Bers, M. U. (2019). What they learn when they learn coding: investigating cognitive domains and computer programming knowledge in young children. *Educational Technology Research and Development*, 67(1), 541–575.
- Subotnik, R. F., Paula, O.-K., Worrell, F. C., & Subotnik, R. F. (2011). Rethinking giftedness and gifted education: A proposed direction forward based on psychological science. *Psychological Science in the Public Interest*, 12(1), s. 3-54.
- Sucu, F., & Çakıroğlu, Ü. (2022). Robotik çevrimiçi öğretilir mi?: Pandemi sırasında robotik eğitim süreçlerindeki değişimler. *Uludağ Üniversitesi Eğitim Fakültesi Dergisi*, 35(3), 532-559.
- Sullivan, A., & Bers, M. U. (2018). Dancing robots: Integrating art, music, and robotics in Singapore's early childhood centers. *International Journal of Technology and Design Education*, 28(1), 325-346.
- Şahin, Ç. Ç. (2014). Bilim ve sanat merkezi öğrencilerinin bu kurumlara ilişkin görüşlerinin incelenmesi. *Hasan Ali Yücel Eğitim Fakültesi Dergisi*, 11(1), 101-117.
- Şencan, H. (2005). *Sosyal ve davranışsal ölçümlerde güvenilirlik ve geçerlilik*. Seçkin Yayıncılık.
- Şimşek, H., & Yıldırım, A. (2021). *Nitel Araştırma yöntemleri*. Seçkin Yayıncılık.
- Talan, T. (2020). Eğitsel robotik uygulamaları üzerine yapılan çalışmaların incelenmesi. *Yaşadıkça Eğitim*, 34(20), 503-522.
- Tiryaki, S. H., & Balaman, F. (2021). The opinions of students about Scratch, Arduino and Python using from open source code softwares. *Journal of Computer and Education Research*, 9(18), 831-852.

- Tisza, G., Sharma, K., Papavlasopoulou, S., Markopoulos, P., & Giannakos, M. (2022). Understanding fun in learning to code: A multi-modal data approach. *IDC '22: Proceedings of the 21st Annual ACM Interaction Design and Children Conference*, 14(1), 274-287.
- Topping, K. J., Douglas, W., Robertson, D., & Ferguson, N. (2021). The effectiveness of online and blended learning from Schools. *University of Dundee*, 1(1), 1-151.
- Tran, H. V., Dinh, T. ., Anh, T. L., & Quoc, V. H. (2024). What factors most influence self-learning Python programming for 10th grade students? A case study in Viet Nam. *American Journal of Science Education Research*, 643(1), 1-11.
- Trilling, B., & Fadel, C. (2009). *21st Century Skills*. John Wiley & Sons.
- Tsai, M.-J., Wang, C.-Y., & Hsu, P.-F. (2019). Developing the computer programming self-efficacy scale for computer literacy education. *Journal of Educational Computing Research*, 56(8), 1345-1360.
- Tselegkaridis, S., & Sapounidis, T. (2021). Simulators in educational robotics: A review. *MDPI Education Sciences*, 11(11), 1-12.
- Unal, A., & Topu, F. B. (2022). Programlama eğitiminde metin-tabanlı ve hibrit-tabanlı ortamları kullanan lise öğrencilerinin deneyimlerine ilişkin karşılaştırmalı bir durum çalışması. *Kafkas Eğitim Araştırmaları Dergisi*, 9(1), 492-512.
- Usher, M., Reznik, N., Bronshtein, G., & Kohen-Vacs, D. (2025). Bridging virtual and physical: Exploring students computational thinking and creativity in robot-guided vs. simulation-based Learning. *The Journal of Learning Analytics*, 12(2), 35-49.
- Uslu, A., Yavuz, G. Ö., & Koçak, U. Y. (2022). A systematic review study on educational robotics and robots. *Interactive Learning Environments*, 31(9), 5874–5898.
- Uysal, H. (2024). *Ters yüz sınıf modeliyle fiziksel programlamanın çevrimiçi bağıllık ve programlama öz-yeterlilik algısı açısından incelenmesi (Tez No: 844517)*. [Yüksek Lisans Tezi, Gazi Üniversitesi-Ankara]. Yüksek Öğretim Kurulu Ulusal Tez Merkezi.
- Uğuz, H. (2019). *Lego robotikle programlamanın ortaokul öğrencilerinin problem çözme becerilerine ve başarılarına etkisi (Tez No: 584836)*. [Yüksek Lisans Tezi, Atatürk Üniversitesi-Erzurum]. Yüksek Öğretim Kurulu ulusal Tez Merkezi.
- Veneziano, L., & Hooper, J. (1997). A method for quantifying content validity of health-related questionnaires. *American Journal of Health Behavior*, 21(1), 67-70.
- Verner, I. M., Perez, H., & Lavi, R. (2022). Characteristics of student engagement in high-school robotics courses. *International Journal of Technology and Design Education*, 32(1), 2129-2150.
- VEX Robotics. (2024). *VEXcode Home*. <https://www.vexrobotics.com/vexcode> adresinden alındı.
- Voštinár, P., & Knežník, J. (2020). Education with BBC micro:bit. *iJOE*, 16(14), 81-94.
- Wang, H.-Y., Huang, I., & Hwang, G.-J. (2014). Effects of an integrated Scratch and project-based learning approach on the learning achievements of gifted students in computer courses. *2014 IIAI 3rd International Conference on Advanced Applied Informatics*, 1(1), 382-387.
- Wu, S.-Y., & Hwang, W.-Y. (2014). A case study of collaboration with multi-robots and its effect on children's interaction. *Interactive Learning Environments*, 22(4), 429–443.

- Wu, S.-Y., & Su, Y.-S. (2021). Visual programming environments and computational thinking performance of fifth- and sixth-grade students. *Journal of Educational Computing Research*, 59(6), 1075-1092.
- Yükseltürk, E., & Altıok, S. (2015). Bilişim Tteknolojileri öğretmen adaylarının bilgisayar programlama öğretimine yönelik görüşleri. *Amasya Üniversitesi Eğitim Fakültesi Dergisi*, 4(1), 50-65.
- Yıldırım, T. S. (2020). *Robotik kodlama öğretiminde probleme dayalı öğrenme yaklaşımının başarı, pozitif duygu ve bilgi işlemsel düşünmeye etkisi* (Tez No: 623962). [Yüksek Lisans Tezi, Çanakkale Onsekiz Mart Üniversitesi-Çanakkale]. Yüksek Öğretim Kurulu Ulusal Tez Merkezi.
- Yıldız, T., & Seferoğlu, S. S. (2021). The effect of robotic programming on coding attitude and computational thinking skills toward self-efficacy perception. *Journal of Learning and Teaching in Digital*, 6(2), 101-116.
- Yaman, S. (2020). *Bilim ve sanat merkezlerinde öğrencilere sunulan eğitim etkinliklerinin yönetimine ilişkin paydaş görüşleri* (Tez No: 640457). [Yüksek Lisans Tezi, Marmara Üniversitesi-İstanbul]. Yüksek Öğretim Kurulu Ulusal Tez Merkezi.
- Yang, W. (2025). Coding with robots or tablets? Effects of technology-enhanced embodied learning on preschoolers' computational thinking and social-emotional competence. *Journal of Educational Computing Research*, 62(4), 718-740.
- Yang, W., Ng, D. T., & Gao, H. (2022). Robot programming versus block play in early childhood education: Effects on computational thinking, sequencing ability, and self-regulation. *British Journal of Educational Technology*, 53(1), 1817-1841.
- Yolcu, V., & Demirer, V. (2017). A Review on the studies about the use of robotic technologies in education. *SDU International Journal of Educational Studies*, 4(2), 127-139 .
- Yumbul, E., & Bayraktar, Ş. (2022). A systematic analysis of research on robotic applications conducted at primary school level in Turkey. *Electronic Journal of Education Sciences*, 11(22), 378-400.
- Yurdagül, H. (2005, 28-30 Eylül). *Ölçek geliştirme çalışmalarında kapsam geçerliliği için kapsam geçerlik indekslerinin kullanılması*. XIV. Ulusal Eğitim Bilimleri Kongresi. Denizli.
- Yurdagül, H., & Bayrak, F. (2012). Ölçek geliştirme çalışmalarında kapsam geçerlilik ölçüleri: Kapsam geçerlilik indeksi ve kappa istatistiğinin karşılaştırılması. *Hacettepe Üniversitesi Eğitim Fakültesi Dergisi*, 2(1), 264-271.
- Yurdakök, E. A. (2022). *Fiziksel programlama aracı destekli programlama öğretiminin ortaokul öğrencilerinin bilgi işlemsel düşünme becerilerine ve özyeterlik algılarına etkisi* (Tez No: 751517). [Yüksek Lisans Tezi, Başkent Üniversitesi-Ankara]. Yüksek Öğretim Kurulu Ulusal Tez Merkezi.
- Yurdakök, E. A., & Kalelioğlu, F. (2023). The effect of teaching physical programming on computational thinking skills and self-efficacy perceptions towards computational thinking. *Journal of Educational Computing Research*, 62(3), 565-595.
- Zamora, P., Lozada, A., Buele, J., & Avilés-Castillo, F. (2025). Robotics in higher education and its impact on digital learning. *Frontiers in Computer Science*, 7(1), 1-6.
- Zhang, L., & Jalal, N. (2019). A systematic review of learning computational thinking through Scratch in K-9. *Computers & Education*, 141(1), 1-25.

- Zhang, X., Chen, Y., Li, D., Hu, L., Hwang, G.-J., & Tu, Y.-F. (2023). Engaging young students in effective robotics education: An embodied learning-based computer programming approach. *Journal of Educational Computing Research*, 62(2), 1-27.
- Zhang, Y., & Zhu, Y. (2022). Effects of educational robotics on the creativity and problem-solving skills of K-12 students: A meta-analysis. *Educational Studies*, 50(6), 1539-1557.
- Zhao, J.-H., Yang, Q.-F., Lian, L.-W., & Wu, X.-Y. (2024). Impact of pre-knowledge and engagement in robot-supported collaborative learning through using the ICAPB model. *Computers & Education*, 217(1), 1-18.
- Zhong, B. (2023). An exploration of combining virtual and physical robots in robotics education. *Interactive Learning Environments*, 31(1), 370-382.
- Zhong, B., & Li, T. (2019). Can pair learning improve students' troubleshooting performance in robotics education? *Journal of Educational Computing Research*, 58(1), 220-248.
- Zhong, B., & Xia, L. (2020). A Systematic review on exploring the potential of educational robotics in mathematics education. *International Journal of Science and Mathematics Education*, 18(1), 769-101.
- Zhong, B., Zheng, J., & Zhan, Z. (2023). An exploration of combining virtual and physical robots in robotics education. *Interactive Learning Environments*, 31(1), 370-382.

EKLER

EK-1. Sanal ve Fiziksel Robotik Kodlama Grupları için Programlama Başarı Testleri (Örnek Sorular)

Öğrenci Adı-Soyadı:

Cinsiyeti:

Sınıf Düzeyi:

Grubu:

Kız

☐

Erkek

☐

6. Sınıf

☐

7. Sınıf

☐

Sanal Robotik Kodlama

☐

Fiziksel Robotik Kodlama

☐

Sanal Robotik Kodlama Başarı Testi Soruları (VEXcode-VR)	Fiziksel Robotik Kodlama Başarı Testi Soruları (Raspberry-Pi)
Soru 1-) Bir VEXcode-VR robotunda tampon sensörünün temel işlevi nedir? A) Nesneler veya duvarlarla fiziksel temasın algılanması B) Nesnelerin renginin algılanması C) Bir nesneye olan mesafenin ölçülmesi D) Dokunduğu malzemenin türünü belirlemek E) Robotun hızının hesaplanması	Soru 1-) Raspberry-Pi'nin GPIO pinleri hangi amaçla kullanılır? A) Bilgisayarı şarj etmek B) Harici cihazlarla etkileşim sağlamak C) Yazılım yüklemek D) Ses dosyalarını oynatmak E) İnternet bağlantısı kurmak
Ekran Çıktısı:	
Cevap: A) Nesnelerle veya duvarlarla fiziksel teması algılamak	Cevap: B Harici cihazlarla etkileşim sağlamak
Soru 2-) VEXcode-VR robotunda nerede, kaç tane tampon sensörü vardır? A-) Bir adet, önde B-) Dört, her köşede bir tane C-) Arkada her iki tarafta birer tane olmak üzere iki D-) Ön tarafta her iki tarafta birer tane olmak üzere iki E-) Bir, arkada	Soru 2-) Raspberry-Pi'de kaç adet GPIO pini bulunmaktadır? A) 38 B) 40 C) 42 D) 35 E) 42
Ekran Çıktısı:	
Cevap: D. Ön tarafta her iki tarafta birer tane olmak üzere iki adet	Cevap: B 40
Soru 3-)	Soru 3-)
Ekran Çıktısı:	
Cevap:	Cevap:

Soru 4-)	Soru 4-)
Ekran Çıktısı:	
Cevap:	Cevap:
Soru 5-)	Soru 5-)
Ekran Çıktısı:	
Cevap:	Cevap:
Soru 6-)	Soru 6-)
Ekran Çıktısı:	
Cevap:	Cevap:
Soru 7-) Python programlama dilindeki kodun sonundaki vr_thread(when_started1) komutunun VEXcode-VR'daki işlevi nedir? A) when_started1 fonksiyonunu sonlandırmak B) when_started1 fonksiyonunu başlatmak C) when_started1 fonksiyonunu duraklatmak D) when_started1 fonksiyonunu tekrar etmek E) when_started1 fonksiyonunu değiştirmek	yesilled= LED(17)Soru 7-) Yukarıdaki Python kod satırının Raspberry-Pi'deki işlevi nedir? A) GPIO pin 17'yi analog giriş olarak ayarlar. B) GPIO pin 17'yi dijital giriş olarak ayarlar. C) GPIO pin 17'yi dijital çıkış olarak ayarlar. D) GPIO pin 17'yi analog çıkış olarak ayarlar. E) GPIO pin 17'yi PWM çıkış olarak ayarlar.
Ekran Çıktısı:	
Cevap: B when_started1 fonksiyonunu başlatmak	Cevap: C GPIO pin 17'yi dijital çıkış olarak ayarlar.
Soru 8-)	Soru 8-)
Ekran Çıktısı:	
Cevap:	Cevap:
<pre>def main(): while not right_bumper.pressed(): drivetrain.turn_for(RIGHT, 90, DEGREES) wait(5,MSEC) drivetrain.stop()</pre> Soru 9-) Yukarıdaki Python kodu VEXcode-VR robotunun ana işlevine dahil edildiğinde ne yapar?	<pre>kirmiziled.off()</pre> Soru 9-) Yukarıdaki Python kodu Raspberry-Pi'de çalıştırıldığında LED'in ne yapmasını sağlar? A) Kırmızı LED'i yakar. B) Kırmızı LED'i söndürür. C) Kırmızı LED'in parlaklığını artırır. D) Kırmızı LED'in parlaklığını azaltır. E) Kırmızı LED'in rengini değiştirir.

A) Sol tampona basılana kadar robotu geriye doğru hareket ettirir B) Robotu hemen durdurur C) Robotu sabit bir mesafe boyunca ileri doğru hareket ettirir D) Sağ tampona basıldığında robot sağa 90 derece döner E) Robotu rastgele bir yönde hareket ettirir	
Ekran Çıktısı:	
Cevap: D) Sağ tampona basıldığında robot sağa 90 derece döner	Cevap: B Kırmızı LED'i söndürür.
Soru 10-)	Soru 10-)
Ekran Çıktısı:	
Cevap:	Cevap:
Soru 11-)	Soru 11-)
Ekran Çıktısı:	
Cevap:	Cevap:
Soru 12)	Soru 12-)
Ekran Çıktısı:	
Cevap:	Cevap:
Soru 13-)	Soru 13-)
Ekran Çıktısı:	
Cevap:	Cevap:
Soru 14-)	Soru 14-)
Ekran Çıktısı:	
Cevap:	Cevap:
<pre>def onevent_sola_donus_0(): drivetrain.turn_for(LEFT, 90, DEGREES)</pre> Soru 15-) Yukarıdaki Python kodu VEXcode-VR robotunun hangi hareketi yapmasını sağlar? A) Robotu 90 derece sağa döndürür B) Robotu 180 derece döndürür	<pre>def aciAyarla(aci): x=(1/180)*aci + 1 duty=x*5 pwm.ChangeDutyCycle(duty)</pre> Soru 15-) Yukarıdaki Python fonksiyonu Raspberry-Pi'de hangi değeri döndürür? A) Açık

C) Robotu ileri hareket ettirir D) Robotu 90 derece sola döndürür E) Robotu durdurur	B) Voltaj C) Akım D) Hız E) Güç
Ekran Çıktısı:	
Cevap: D Robotu 90 derece sola döndürür	Cevap: A aç
Soru 16-)	Soru 16-)
Ekran Çıktısı:	
Cevap:	Cevap:
<pre>def onevent_saga_donus_0(): global saga_donus, sola_donus drivetrain.turn_for(RIGHT, 90, DEGREES)</pre> Soru 17-) Yukarıdaki Python kodunun çıktısı nedir? A-) Sola Dönüş için fonksiyon oluşturulmuştur B-) Sağa Dönüş için fonksiyon oluşturulmuştur C-) Robotu sürekli sağa döndürecektir D-) Robot sürekli ilerlemek için fonksiyon oluşturmuştur. E-) Robot sürekli mesafe sensörü nesne gördüğünde dönüş yapacaktır.	Soru 17-) Bir Python fonksiyonundaki kodun çalışmasını durdurmak için hangi ifade kullanılır? A) halt B) end C) break D) return E) stop
Ekran Çıktısı:	
Cevap: B. Sağa Dönüş için fonksiyon oluşturulmuştur	Cevap: D return
<pre>if göz_sensörü == "yeşil": robot.sola_dön(90) else: robot.saga_dön(90)</pre> Soru 18-) Yukarıdaki Python kodunun işlevi nedir? A) Göz sensörü yeşilse sola, değilse sağa döner B) Göz sensörü mavi gördüğünde sağa döner C) Göz sensörü her durumda sağa döner D) Göz sensörü her durumda sola döner E) Robot durur	Soru 18-) Python programlama dilinde kontrol yapıları kullanarak LED'in butona basıldığında yanmasını sağlamak için hangi yapı(kod) kullanılır? A) Döngü (Loop) B) Fonksiyon (Function) C) Koşul (If-Else) D) Sınıf (Class) E) Modül (Module)
Ekran Çıktısı:	
Cevap: A Göz sensörü yeşilse sola, değilse sağa döner	Cevap: C Koşul (If-Else)

Soru 19-)	Soru 19-)
Ekran Çıktısı:	
Cevap:	Cevap:
Soru 20-)	Soru 20-)
Ekran Çıktısı:	
Cevap:	Cevap:
Soru 21-)	Soru 21-)
Ekran Çıktısı:	
Cevap:	Cevap:
<pre>def when_started1(): while True: if front_distance.get_distance(MM) < 200: drivetrain.turn(RIGHT) else: drivetrain.drive(FORWARD) vr_thread(when_started1)</pre> Soru 22) Yukarıdaki Python kodu çalıştırıldığında VEXcode-VR robotunun hangi sensörüne komut gönderilir? A) Ön Mesafe Sensörü B) Ön Göz Sensörü C) Sol Tampon Dokunma Sensörü D) Sağ Tampon Dokunma Sensörü E) Aşağı Göz Sensörü	<pre>if buton_durum == 0: GPIO.output(LED1, GPIO.HIGH) else: GPIO.output(LED, GPIO.LOW)</pre> Soru 22-) Yukarıdaki Python kodunda LED'in ışık yayması için hangi koşulun sağlanması gerekir? A) if buton_durum == 1: B) if buton_durum != 0: C) if buton_durum == 0: D) if buton_durum > 0: E) if buton_durum < 0:
Ekran Çıktısı:	
Cevap A Ön Mesafe Sensörü	Cevap : C if buton_durum == 0:
Soru 23-) Soru 23-)	Soru 23-)
Ekran Çıktısı:	
Cevap:	Cevap:
<pre>drivetrain.drive_for(FORWARD, hiz_listesi[2 - 1], MM)</pre> Soru 24-) Yukarıdaki Python kodu, VEXcode-VR'daki hız listesinin hangi öğesini kullanarak robotu hareket ettirir?	<pre>LED_pins = [17, 18, 27, 22, 23] for pin in LED_pins: GPIO.output(pin, GPIO.HIGH) sleep(1) GPIO.output(pin, GPIO.LOW)</pre>

A) hiz_listesi'nin 0. ögesi B) hiz_listesi'nin 1. ögesi C) hiz_listesi'nin 2. ögesi D) hiz_listesi'nin 3. ögesi E) hiz_listesi'nin son ögesi	Soru 24-) Yukarıdaki Python kodunda, Raspberry-Pi listesindeki her bir LED'i sırayla yakmak için hangi döngü kullanılır? A) while B) else C) if D) def E) try
Ekran Çıktısı:	
Dogru Cevap: B) hiz_listesi'nin 1. ögesi	Cevap : A while
liste_3 print(liste_4) Soru 25-)	Soru 25-)
Ekran Çıktısı:	
Cevap:	Cevap:

EK-2. Programlama Öz-Yeterlik Ölçeği (Örnek Maddeler)

Programlama Öz-Yeterlik Ölçeği*	
Sevgili Öğrenciler, Bu ölçekler bilimsel bir araştırma için geliştirilmiştir. Burada belirteceğiniz görüşler yalnızca bilimsel araştırma amacıyla kullanılacaktır. Cevaplarınızın kesinlikle gizli tutulacağından emin olunuz. Araştırmanın başarısı sizin vereceğiniz cevaplara bağlıdır. Bu nedenle içtenlikle ve gerçek görüşlerinizi belirtmeniz bizim için büyük önem taşımaktadır. Her maddeyi dikkatle okuyup belirtilen görüşün sizin görüşünüze ne derece uyduğunu ya da uymadığını belirleyiniz. Soruların doğru ya da yanlış cevapları yoktur. Lütfen tüm soruları yanıtlayınız ve her madde için bir seçeneği işaretleyiniz.	
Doç. Dr. Fatma Burcu TOPU Robotik Kodlama Öğretmeni Nazmi SARIGÜL	

Öğrenci Adı-Soyadı:

Cinsiyeti:

Sınıf Düzeyi:

Grubu:

Kız	☐	Erkek	☐
6. Sınıf	☐	7. Sınıf	☐
Sanal Robotik Kodlama	☐	Fiziksel Robotik Kodlama	☐

	Kesinlikle katılmıyorum	Katılmıyorum	Kararsızım	Katılıyorum
1 Karşılaştığım bir problemin, programlama problemi olup olmadığını anlayabilirim.				
2 Karmaşık programlama problemlerini, küçük alt problemlere ayırarak çözebilirim.				
3 Bir programlama probleminin çözülüp çözülemeyeceği hakkında yorum yapabilirim.				
4				
5				
6				
7				
8				
9				
10 Farklı çözüm adımlarından programlama probleminin çözümüne en uygun olanı seçerim.				
11 Çözüm adımlarından programlama probleminin çözümüne en uygun olanı seçerim.				
12 Programlama probleminin çözümü için bulduğum adımları arkadaşlarımla paylaşıyorum.				
13				
14				
15				
16				
17				
18				
19				
20 Programlamanın aşamalarını bilirim.				
21 Program kodlarının nerede yazılacağını bilirim.				
22 Oluşturduğum programı çalıştırabilirim.				
23				
24				
25				
26				
27				
28				
29				
30 Birden fazla yazılım projesi arasından verilen ölçütlere uygun olanı seçerim.				
31 Yazılım projesi fikrimi adım adım açıklayabilirim.				

EK-3. Programlama Kaygı Ölçeği (Örnek Maddeler)

Sevgili öğrenciler,

Bu ölçek programlama öğretim sürecinde yaşadığınız deneyimi belirlemek amacıyla hazırlanmıştır. Her soruyu dikkatli bir şekilde okuyup cevaplamanızı ve hiçbir soruyu cevapsız bırakmamanızı rica ederiz. Vermiş olduğunuz cevaplar sadece araştırma amaçlı kullanılacaktır. Katılımınız için teşekkür ederiz.

Doç. Dr. Fatma Burcu TOPU
Robotik Kodlama Öğretmeni Nazmi SARIGÜL

Öğrenci Adı-Soyadı:

Cinsiyeti:

Sınıf Düzeyi:

Grubu:

6. Sınıf

Kız ☐

7. Sınıf

Erkek ☐

Sanal Robotik Kodlama ☐

Fiziksel Robotik Kodlama ☐

Lütfen, programlama ile ilgili kendinizi nasıl hissettiğinizi en iyi açıklayan numarayı yuvarlak içine alarak tüm ifadeleri işaretleyiniz.

Hiçbir zaman doğru değil	0	Nadiren doğru	1	Bazen doğru	2	Sıkla doğru	3	Her zaman doğru	4
1.									
2.									
3.									
4.									
5.									
6.									
7.									
8.									
9.									
10.									
11.									
12.									
13.									
14.									
15.									
16.									
17.									
18.									
19.									

EK-4. Haftalık Robotik Kodlama Kazanım Değerlendirme Kontrol Listeleri (Örnek Maddeler)

Öğrenci Adı-Soyadı:

Cinsiyeti:

Sınıf Düzeyi:

Grubu:

(Hafta No) Kazanım(lar):

Sanal veya Fiziksel Robotik Etkinlik Adı:

6. Sınıf ☐ Kız ☐

Sanal Robotik Kodlama ☐

7. Sınıf ☐ Erkek ☐

Fiziksel Robotik Kodlama ☐

NOT: Aşağıdaki kontrol listeleri haftalık etkinliklerdeki robotik kodlama kazanımlarına göre öğretmen tarafından doldurulacaktır.

Sanal Robotik Kontrol Listeleri			Fiziksel Robotik Kontrol Listeleri		
1.Hafta			1.Hafta		
Sorular	Evet/Hayır		Sorular	Evet/Hayır	
VEXcode-VR ile neler yapılabileceğini özetledi.			Raspberry-Pi ile neler yapılabileceğini özetledi.		
			Raspberry-Pi'nin kullanım alanlarıyla, günlük hayatta kullandığı elektronik cihazları karşılaştırdı.		
VEXcode-VR Bolk kodlar ile Metin Kodları birbiri ile karşılaştırdı.					

2.Hafta			2.Hafta		
Sorular	Evet/Hayır		Sorular	Evet/Hayır	
			Raspberry-Pi üzerindeki GPIO (giriş-çıkış) pinlerini görevlerini bildi.		
Kodlayacağı robotun özellikleri bildi.					

3.Hafta			3.Hafta		
Sorular	Evet/Hayır		Sorular	Evet/Hayır	
VEXcode-VR da bekleme komutunu uygun biçimde kullandı.			Derleyicide yazdığı kodları Raspberry-Pi ye yükledi.		

4.Hafta			4.Hafta		
Sorular	Evet/Hayır		Sorular	Evet/Hayır	
			Derleyicide yazdığı kodları Raspberry-Pi ye yükledi.		
Döngü içerisinde yapılacak işlemleri doğru belirledi.			Döngü içerisinde yapılacak işlemleri doğru belirledi.		

5.Hafta			5.Hafta		
Sorular	Evet/Hayır		Sorular	Evet/Hayır	
			Raspberry-Pi ile Led uygulamasında farklı saniyelerde ledi yakıp-söndürdü.		

6.Hafta			6.Hafta		
Sorular	Evet/Hayır		Sorular	Evet/Hayır	
Disk labirenti algoritmasını doğru kurdu.			Derleyicide yazdığı kodları Raspberry-Pi ye yükledi.		

7.Hafta			7.Hafta		
---------	--	--	---------	--	--

Sorular	Evet/Hayır		Sorular	Evet/Hayır	
Farklı algoritma kurarak kurduđu algoritmaya uygun program yazdı.			İf-Else komutunu kullanım durumlarını açıkladı.		

8.Hafta			8.Hafta		
Sorular	Evet/Hayır		Sorular	Evet/Hayır	
			Raspberry-Pi ile 6 led bağlayarak Liste şeklinde gösterini uyguladı.		

9.Hafta ve 10 Hafta			9.Hafta ve 10 Hafta		
Sorular	Evet/Hayır		Sorular	Evet/Hayır	
Labirente sensörleri kullandı ve doğru bir şekilde kodladı.			Akıllı ev sisteminde sensör kullandı ve doğru kodladı.		
Robot 3 noktasına doğru bir şekilde ilerledi.			Akıllı ev sisteminde 4 adet led kullandı ve doğru kodladı.		



EK-5. Haftalık Robotik Kodlama Meşguliyet Gözlem Formu (Örnek Maddeler)

Öğrenci Adı-Soyadı:

Cinsiyeti: Kız ☐

Erkek ☐

Sınıf Düzeyi:

6. Sınıf ☐

7. Sınıf ☐

Grubu:

Sanal Robotik Kodlama ☐

Fiziksel Robotik Kodlama ☐

(Hafta No) Kazanım(lar):

Sanal veya Fiziksel Robotik Etkinlik Adı:

NOT: Aşağıdaki meşguliyet gözlem formu haftalık etkinliklerdeki robotik kodlama kazanımlarına göre öğretmen tarafından doldurulacaktır.

	Gözlenmedi (1)	Kısmen Gözlendi (2)	Gözlendi (3)	Açıklama
1. Robotik kodlama etkinliğini görünce meraklanma				
2. Robotik kodlama etkinliğine katılmak için sabırsızlanma				
3.				
4.				
5. Robotik kodlama etkinliğinde sıkılma				
6.				
7. Robotik kodlama etkinliğinde yanlış yaptığında sinirlenme				
8.				
9.				
10. Robotik kodlama etkinliğini yönergeleri takip ederek doğru-uygun şekilde tamamlamak için çabalama				
11. Robotik kodlama etkinliğindeki yönergeleri takip ederek etkinliği tamamlarken zorlanma				
12.				
13.				
14.				
15.				
16. Robotik kodlama etkinliğinde robotu oluştururken ve algoritmayı geliştirirken yaptığı hatayı fark etme				
17.				
18.				
19.				
20.				
21.				
22.				
23. Robotik kodlama etkinliğindeki doğru algoritma adımlarını uygulama				
24. Geliştirdiği robotu ve kurduğu algoritmayı test etme (robotu çalıştırma)				
25. Robotik kodlama etkinliğini tamamlamak için arkadaşlarından-öğretmeninden yardım isteme				
26.				
27.				
28.				
29.				
30. Robotik kodlama etkinliğine yönelik görüşlerini ve duygularını öğretmene söyleme				
31.				
32. Robotik kodlama etkinliğini tamamlamak için arkadaşlarıyla-öğretmeniyle iletişim kurma				
33.				
34. Robotik kodlama etkinliğiyle ilgili olmayan davranışlar sergileyerek arkadaşlarıyla diyalog kurmaya çalışma				
35. Robotik kodlama etkinliğinde arkadaşları arasında söz sahibi olmaya-lideri olmaya çalışma (kendi görüşünü kabul ettirmeye çalışma)				

EK-6. Görüşme Rehberi (Örnek Sorular)

Sevgili öğrenciler, “Robotik kodlama” dersi “Programlama eğitimi” içerisinde yer alan eğitsel robotik etkinlikleri ile gerçekleştirdiğimiz Sanal Robotik (Vex-Vr) ve Fiziksel Robotik (Raspberry-Pi) etkinlikleriyle ilgili sizlerle görüşmek istiyorum. Fikirleriniz Sanal Robotik (Vex-Vr) ve Fiziksel Robotik (Raspberry-Pi) uygulamalarıyla Python programlama dili öğretimi, iletişim, etkileşim ve uygulamada karşılaşılan sorunlarla ilgili faktörleri ortaya çıkarmak açısından faydalı olacaktır. Bu nedenle samimi ve içten cevaplarınız araştırmamıza katkı sunacaktır. Çalışmanın daha sağlıklı yürütmesi açısından izin vererseniz görüşmeyi kaydetmek istiyorum.

Görüşmeye başlamadan önce, görüşme süresince belirttiğiniz görüşlerin gizli olduğunu ve araştırma sonuçlarını yazarken kimliğiniz ile ilgili bilgilerin rapora kesinlikle yansıtılmayacağını belirtmek isterim. Görüşme sırasında yapılacak kayıtların yüksek lisans tez danışmanı ve/veya Tez İzleme Kurulu üyeleri tarafından kimliğiniz belirtilmeden dinlenebilecektir. Başlamadan önce, bu söylediklerim ile ilgili belirtmek istediğiniz bir düşünce ya da sormak istediğiniz bir soru var mı? Katılımınız için teşekkür ederim.

Robotik Kodlama Öğretmeni Nazmi Sarıgül

Doç. Dr. Fatma Burcu TOPU

Öğrenci Adı-Soyadı:

Cinsiyeti: Kız ☐ Erkek ☐
Sınıf Düzeyi: 6. Sınıf ☐ 7. Sınıf ☐
Grubu: Sanal Robotik Kodlama ☐ Fiziksel Robotik Kodlama ☐

GÖRÜŞME SORULARI

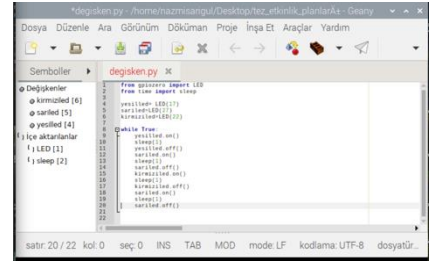
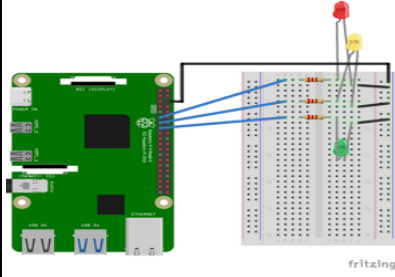
1. Sanal Robotik (Vex-Vr) ve Fiziksel Robotik (Raspberry-Pi) uygulamasının başından sonuna kadar (ilk haftadan son haftaya kadar) neler yaptığınızı açıklayınız mı?
2. Sanal Robotik (Vex-Vr) ve Fiziksel Robotik (Raspberry-Pi) ilk gördüğünde neler düşündün veya ne hissettin? Açıklayınız mı?
3. Sanal Robotik (Vex-Vr) ve Fiziksel Robotik (Raspberry-Pi) hangi yönleri seni memnun etti ya da etmedi? Örneklerle açıklayınız mı?
4. Sanal Robotik (Vex-Vr) ve Fiziksel Robotik (Raspberry-Pi) problem ve bulduğunuz çözüm yolları ile günlük hayatta yaşanabilecek problemler arasında bir ilişki kurabildin mi?
Evet ise nasıl bir ilişki kurduğunu açıklayınız mı?
Hayır ise neden kuramadığınızı açıklayınız mı?
5. VEXcode-VR ve Raspberry-Pi etkinlikleri sırasında kendinizi ne kadar rahat hissediyorsunuz? Açıklayınız mı?
6. Sanal Robotik (Vex-Vr) ve Fiziksel Robotik (Raspberry-Pi) etkinlikler esnasında karşılaştığınız en büyük sorun neydi ve sorunu nasıl hallettin?
7. Robotik kodlama dersinin, gelecekteki mesleki hedeflerinize katkı sağladığını düşünüyor musunuz? Neden?
8. Verilen görev/problem için çözüm yolu olacak programı oluştururken zihinsel olarak en zorlandığı şey neydi? Nedenini açıklayınız mı?
9. Etkinlikler sonunda kendinizi programlama konusunda yeterli hissediyor musunuz? Bu yeterliliğinizi nasıl değerlendirirsiniz?
10. Bilgisayarda yazdığınız kodları robot üzerinde denemek sana hangi açılarından katkı sağladığını/seni hangi yönlerden olumlu/olumsuz etkilediğini açıklayınız mı?

EK-7. Raspberry-Pi Etkinlik Planları

Etkinlik No	1								
Dersin Adı	ROBOTİK KODLAMA								
Sınıfı									
Tarih									
Modül Adı/ Etkinliklerin Adı	Mikroişlemci ve Mikro denetleyiciler / Raspberry-Pi nedir? Niçin Kullanılır.								
Önerilen Süre	40+40 dk.								
Kazanım:	1. Raspberry-Pi ile neler yapılabileceğini özetler. 2. Raspberry-Pi'nin kullanım alanlarıyla, günlük hayatta kullandığı elektronik cihazları karşılaştırır. 3. Günlük hayatta kendisine kolaylık sağlayacak elektronik bir devre tasarlar.								
Öğretme öğrenme süreci	• Raspberry-Pi elektronik kartın anlatımı. • Günlük hayatta kullandığı bir elektronik cihazın Raspberry-Pi ile nasıl yapılabileceği sorusu sorulur. 								
DEĞERLENDİRME	Etkinlik sonunda etkinliğe katılan her öğrenci için aşağıda bulunan kazanım Kontrol Listesini doldurulması tavsiye edilir. (Öğrencinin kazanımı gerçekleştirme durumuna göre Evet – Hayır bölümünü doldurunuz.) <table border="1"> <thead> <tr> <th>Kazanım Kontrol Listesi</th><th>Evet / Hayır</th></tr> </thead> <tbody> <tr> <td>Raspberry-Pi ile neler yapılabileceğini özetledi.</td><td></td></tr> <tr> <td>Raspberry-Pi'nin kullanım alanlarıyla, günlük hayatta kullandığı elektronik cihazları karşılaştırdı.</td><td></td></tr> <tr> <td>Günlük hayatta Raspberry-Pi nerelerde kullanılabileceğini örneklendirdi.</td><td></td></tr> </tbody> </table>	Kazanım Kontrol Listesi	Evet / Hayır	Raspberry-Pi ile neler yapılabileceğini özetledi.		Raspberry-Pi'nin kullanım alanlarıyla, günlük hayatta kullandığı elektronik cihazları karşılaştırdı.		Günlük hayatta Raspberry-Pi nerelerde kullanılabileceğini örneklendirdi.	
Kazanım Kontrol Listesi	Evet / Hayır								
Raspberry-Pi ile neler yapılabileceğini özetledi.									
Raspberry-Pi'nin kullanım alanlarıyla, günlük hayatta kullandığı elektronik cihazları karşılaştırdı.									
Günlük hayatta Raspberry-Pi nerelerde kullanılabileceğini örneklendirdi.									
HAZIRLAYANLAR									

Etkinlik No	2								
Dersin Adı	ROBOTİK KODLAMA								
Sınıfı									
Tarih									
Modül Adı/ Etkinliklerin Adı	Mikroişlemci ve Mikro denetleyiciler / Raspberry-Pi nedir? Niçin kullanılır.								
Önerilen Süre	40+40 dk.								
Kazanım:	1.Raspberry-Pi bilgisayarın kurulumunu yapabilir. 2.Raspberry-Pi üzerinde GPIO (genel amaçlı giriş çıkış)’ pinlerinin kullanımını ve kart üzerindeki yerlerini bilir.								
Öğretme öğrenme süreci	• Raspberry-Pi elektronik kartın kurulumunu yaparabilir. • Raspberry-Pi çıkış pinlerini tanır. • GPIO Çıkış pinlerini kullanımını anlar.								
DEĞERLENDİRME	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; padding: 5px;">Kazanım Kontrol Listesi</th><th style="text-align: center; padding: 5px;">Evet / Hayır</th></tr> </thead> <tbody> <tr> <td style="padding: 5px;">Raspberry-Pi de kaç GPIO bulunmaktadır.</td><td style="text-align: center; padding: 5px;"></td></tr> <tr> <td style="padding: 5px;">Raspberry-Pi’nin özelliklerini sıralar.</td><td style="text-align: center; padding: 5px;"></td></tr> <tr> <td style="padding: 5px;">Raspberry-Pi üzerinde ki GPIO (giriş-çıkış) pinlerini görevlerini bilir</td><td style="text-align: center; padding: 5px;"></td></tr> </tbody> </table>	Kazanım Kontrol Listesi	Evet / Hayır	Raspberry-Pi de kaç GPIO bulunmaktadır.		Raspberry-Pi’nin özelliklerini sıralar.		Raspberry-Pi üzerinde ki GPIO (giriş-çıkış) pinlerini görevlerini bilir	
Kazanım Kontrol Listesi	Evet / Hayır								
Raspberry-Pi de kaç GPIO bulunmaktadır.									
Raspberry-Pi’nin özelliklerini sıralar.									
Raspberry-Pi üzerinde ki GPIO (giriş-çıkış) pinlerini görevlerini bilir									
HAZIRLAYANLAR									

Etkinlik No	3
Dersin Adı	ROBOTİK KODLAMA
Sınıfı	
Tarih	
Modül Adı/ Etkinliklerin Adı	Mikroişlemci ve Mikro denetleyiciler / Raspberry-Pi ile led yakıp söndürerek değişken tanımlar.
Önerilen Süre	40+40 dk.
Kazanım:	1- LED diyotun kullanım alanlarına örnekler verir. 2- Belirlenen programlama dilinde değişkenleri ve sabitleri uygun şekilde kullanarak programlar geliştirir. 3- Raspberry-Pi derleyici kullanarak kod yüklemeyi test eder. 4- Raspberry-Pi ile 3 LED bağlantısını sağlayan devreyi kurar. 5- Programda değişkenleri uygun biçimde tanımlar. 6- Değişkeni programlama diline uygun şekilde tanımlar.
Öğretme öğrenme Süreci	Güdüleme aşamasında etkinliğin iyi anlaşılması ve dersin ilerlemesini sağlayabilmek adına öğrencilerin öncelikle dikkatlice dinlemesi, anlatım bittikten sonra öğretmenin söylediği zamanda uygulamaya geçilerek öğretilen adımları sırasıyla uygulaması istenir. Ardından öğrencilerle oyunun kuralları paylaşılır. <i>Şimdi sizlerle çok eğlenceli bir oyun oynayacağız. Ben başlayın dediğim anda hepiniz benimle birlikte iki gözünüzü birden kırpacaksınız. İki göz kırpma arasındaki zamanı ayarlamak için ben elimle masaya vuracağım. Değiştir dediğim anda önce bir kez önce bir gözünüzü ardından diğer gözünüzü olacak şekilde kırpacaksınız. Tekrar değiştir diyeceğim bu defa sağ gözünüz iki kere sol gözünüzü iki kere olacak şekilde kırpacağız. Bu döngüyü her değiştir dediğimde bu sıra ile tekrar edeceksiniz. Oyunu karıştırmadan, başarı ile tamamlayanlara göz kırpma emojişi takaca- gız”</i> denilerek öğrenciler oyun ve etkinlik için hazır hale getirilir. Oyun bitiminin ardından öğrencilere ödülleri verilir. <i>(Burada oyunu çeşitlendirmek ve buzzer atıf yapmak adına el çırpma hareketi de eklenebilir.)</i> <i>Oyundan sonra led diyotları gözümüze benzeterek led yakıp söndürmede değişkenin ne olduğu öğrencilere sorulur ve programlamada değişken kavramı anlatılır.</i> • Led diyotun kullanım alanlarına öğrenciler tarafından örnekler verilmesi istenir. • Elektronik devrelerde diyotun devreye nasıl bağlanması gerektiği anlatılır. • Fiziksel olarak diyotun özelliklerinden (+) Artı bacağın uzun bacak (-) Eksi Bacağın kısa bacak olduğu anlatılır. Fiziksel devre kurulurken bu bağlantıya dikkat edilmesi gerektiği söylenir. • LED Diyot Raspberry-Pi karta bağlanmadan önce öğrencilere ne yapmaları gerektiği sorulur. • LED diyotun 1 saniye aralıklarla yanıp sönmelerini sağlayacak kod yazılır.
Uygulama Çıktısı	



from gpiozero import LED

from time import sleep

yesilled= LED(17)

sariled=LED(27)

kirmiziled=LED(22)

while True:

 yesilled.on()

 sleep(1)

 yesilled.off()

 sariled.on()

 sleep(1)

 sariled.off()

 kirmiziled.on()

 sleep(1)

 kirmiziled.off()

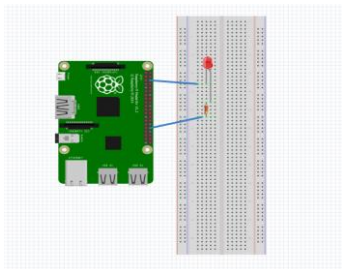
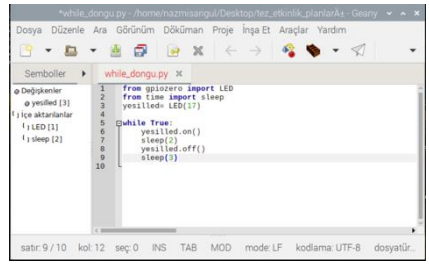
 sariled.on()

 sleep(1)

 sariled.off()

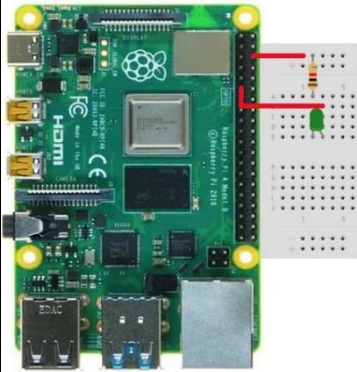
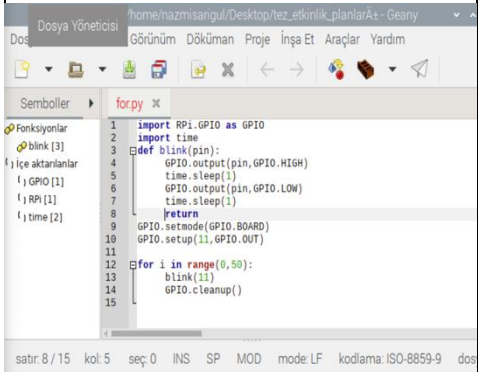
Kod Editör Ekran Görüntüsü

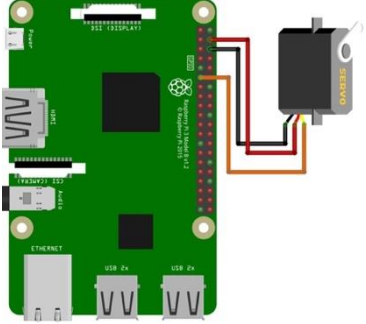
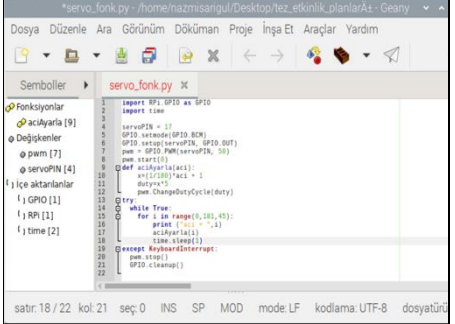
DEĞERLENDİRME	Etkinlik sonunda etkinliğe katılan her öğrenci için aşağıda bulunan kazanım Kontrol Listesini doldurulması tavsiye edilir. (Öğrencinin kazanımı gerçekleştirme durumuna göre Evet – Hayır bölümünü doldurunuz.)	
	Kazanım Kontrol Listesi	Evet / Hayır
	Raspberry-Pi ile LED bağlantısını sağlayan devreyi kurdu ve çalıştırdı.	
	Derleyicide yazdığı kodları Raspberry-Pi ye yükler.	
	Program yazımında değişkenleri uygun biçimde tanımlar.	
	Farklı değişkenleri aynı program içerisinde kullanır.	
HAZIRLAYANLAR		

Etkinlik No	4
Dersin Adı	ROBOTİK KODLAMA
Sınıfı	
Tarih	
Modül Adı/ Etkinliklerin Adı	Mikroişlemci ve Mikro denetleyiciler / Raspberry-Pi ile LED Yakıp Söndürme
Önerilen Süre	40+40 dk.
Kazanım:	1-LED diyotun kullanım alanlarına örnekler verir. 2-Belirlenen programlama dilinde tekrarlı yapıları kullanarak programlar geliştirir. 3-Raspberry-Pi derleyici kullanarak kod yüklemeyi test eder. 4-Raspberry-Pi ile 1 LED bağlantısını sağlayan devreyi kurar. 5-Sürekli döngü kavramını açıklar. 6-Sürekli döngü içerisinde yapılacak işlemleri doğru belirler.
Öğretme öğrenme Süreci	• Televizyon, Çamaşır Makinesi vb. elektronik cihazlarda LED diyotun bulunduğu söylenir. Işıklı uyarı sistemlerinde kullanılabileceği durumlara örnekler verilir. • Led diyotun kullanım alanlarına öğrenciler tarafından örnekler verilmesi istenir. • Elektronik devrelerde diyotun devreye nasıl bağlanması gerektiği anlatılır. • Fiziksel olarak diyotun özelliklerinden (+) Artı bacağın uzun bacak (-) Eksi Bacağın kısa bacak olduğu anlatılır. Fiziksel devre kurulurken bu bağlantıya dikkat edilmesi gerektiği söylenir. • LED Diyot Raspberry-Pi karta bağlanmadan önce öğrencilere ne yapmaları gerektiği sorulur. • LED diyotun 1 saniye aralıklarla yanıp sönmelerini sağlayacak kod yazılır. • Gerekli bağlantılar sağlandıktan sonra LED diyotun 1 saniye aralıklarla yanıp söneceği ve bunun sahte bir araba alarmı olabileceği öğrencilere söylenir. • Zamanlama durumuna göre ek olarak Ek 1 de verilen devre şeması ve kodu öğrenciler ile birlikte uygulanabilir.
Uygulama Çıktısı	 from gpiozero import LED from time import sleep kirmiziled= LED(17) while True:  Kod Editör Ekran Görüntüsü

	kirmiziled.on() sleep(2) kirmiziled.off() sleep(3)	
DEĞERLENDİRME	Etkinlik sonunda etkinliğe katılan her öğrenci için aşağıda bulunan kazanım Kontrol Listesini doldurulması tavsiye edilir. (Öğrencinin kazanımı gerçekleştirme durumuna göre Evet – Hayır bölümünü doldurunuz.)	
	Kazanım Kontrol Listesi	Evet / Hayır
	Raspberry-Pi ile LED bağlantısını sağlayan devreyi kurdu ve çalıştırdı.	
	Derleyicide yazdığı kodları Raspberry-Pi ye yükler.	
	Döngü içerisinde yapılacak işlemleri doğru belirler.	
	While True(sürekli döngü) kodunu doğru kullandı.	
HAZIRLAYANLAR		

Etkinlik No	5
Dersin Adı	ROBOTİK KODLAMA
Sınıfı	
Tarih	
Modül Adı/ Etkinliklerin Adı	Mikroişlemci ve Mikro denetleyiciler / Raspberry-Pi ile for döngüsü uygulaması
Önerilen Süre	40+40 dk.
Kazanım:	1- Günlük hayatta ledler kullanılarak uygulanan ışıklı uyarı sistemlerine örnekler verir. 2- Raspberry-Pi ile Led kullanarak ışıklı uyarı sistemi devresi kurar. 3- Işıklı uyarı sistemlerinin çalışma prensiplerini arkadaşları ile paylaşır. 4- For döngüsünü uygun biçimde programına yazar. 5- Belirlenen programlama dilinde tekrarlı yapıları kullanarak programlar geliştirir.
Öğretme öğrenme süreci	• Derse girişte dikkat çekmek için öğrencilere aşağıdaki soru sorularak başlanır. • Soru : Gördüğünüz fakat sesinizi ve hareketleriniz iletemediğiniz birisi ile nasıl iletişime geçersiniz? • Öğrencilerden gelen cevaplar hızlıca tahtaya yazılır ve verilen cevaplardan ışık ile iletişim konusunda gelen cevaplar üzerinde durularak ışıklı iletişim ile konusuna değinerek derse devam edilir. • Öğrencilere Raspberry-Pi Led otomatik yakıp söndürme uygulaması için devrenin kurulması istenir. • Gerekli bağlantılar sağlandıktan sonra buton ve LED devresi çalıştırılır. • Ek Olarak Etkinlik sonunda öğrencilere ek bir uygulama olarak farklı bir led bağlamaları istenir ve her bir ledin farklı saniyelerde yakıp söndürmeleri için kod yazmaları istenir.

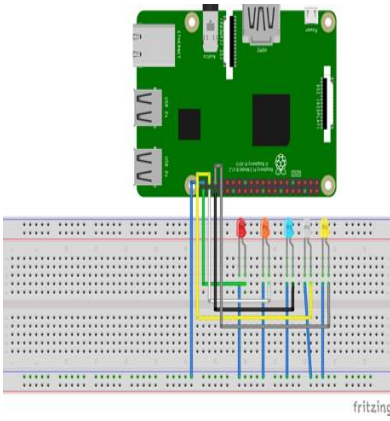
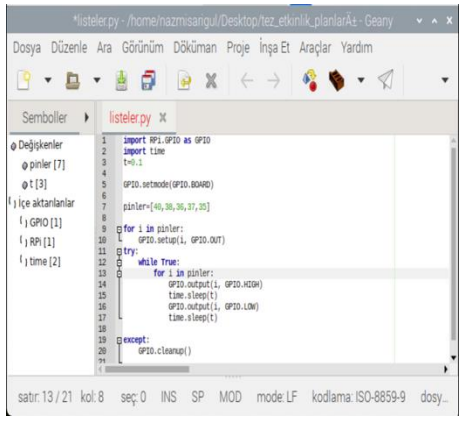
Uygulama Çıktısı												
	<pre>import RPi.GPIO as GPIO import time def blink(pin): GPIO.output(pin,GPIO.HIGH) time.sleep(1) GPIO.output(pin,GPIO.LOW) time.sleep(1) return GPIO.setmode(GPIO.BOARD) GPIO.setup(11,GPIO.OUT) for i in range(0,50): blink(11) GPIO.cleanup()</pre>	Kod Editör Ekran Görüntüsü										
DEĞERLENDİRME	- Etkinlik sonunda etkinliğe katılan her öğrenci için aşağıda bulunan kazanım Kontrol Listesini doldurulması tavsiye edilir. (Öğrencinin kazanımı gerçekleştirme durumuna göre Evet – Hayır bölümünü doldurunuz.) <table><tr><th>Kazanım Kontrol Listesi</th><th>Evet / Hayır</th></tr><tr><td>For döngüsünü tanımladı</td><td></td></tr><tr><td>For döngüsü yazımını doğru uyguladı.</td><td></td></tr><tr><td>Farklı yerlerde nasıl kullanımına örnekler veridi</td><td></td></tr><tr><td>Raspberry Pi ile Led uygulamasında farklı saniyelerde ledi yakıp-söndürdü</td><td></td></tr></table>		Kazanım Kontrol Listesi	Evet / Hayır	For döngüsünü tanımladı		For döngüsü yazımını doğru uyguladı.		Farklı yerlerde nasıl kullanımına örnekler veridi		Raspberry Pi ile Led uygulamasında farklı saniyelerde ledi yakıp-söndürdü	
Kazanım Kontrol Listesi	Evet / Hayır											
For döngüsünü tanımladı												
For döngüsü yazımını doğru uyguladı.												
Farklı yerlerde nasıl kullanımına örnekler veridi												
Raspberry Pi ile Led uygulamasında farklı saniyelerde ledi yakıp-söndürdü												
HAZIRLAYANLAR	ROBOTİK KODLAMA KOMİSYONU											

Etkinlik No	6
Dersin Adı	ROBOTİK KODLAMA
Sınıfı	
Tarih	
Modül Adı/ Etkinliklerin Adı	Mikroişlemci ve Mikro denetleyiciler / Raspberry-Pi ile LED Yakıp Söndürme
Önerilen Süre	40+40 dk.
Kazanım:	1. Servo motor kullanım alanlarını söyler. 2. Raspberry-Pi ile servo motor devresini kurar. 3. Belirlenen programlama dilinde parametre alan değer döndüren fonksiyon içeren programlar geliştirir. 4. Fonksiyon kavramını açıklar. 5. Python fonksiyon içerisinde yapılacak işlemleri doğru belirler.
Öğretme öğrenme Süreci	• Dikkat çekmek için sınıfın kapısı açılır ve öğretmen sınıfa girmeden kumanda ile yönlendirilebilen bir robot yönlendirilerek kapıdan içeri girmesi sağlanır ve öğretmen masasına doğru ilerler. Arkasından öğretmen girer ve robotu öğrencilere tanıtır. • Öğrencilerden robotu hareket ettiren şeyin ne olduğu sorulur ve nasıl hareket ettiği öğrencilerden cevaplanması istenir. • Cevaplar alındıktan sonra robotlarda elektrik motorları ile ilgili bilgiler verilir robotların hareketini sağlayan şeyin motorlar olduğunu ve robotlarda kullanılan elektrik motorlarından bahsedilir. • Robotlarda en çok kullanılan servo motorların özellikleri ve çalışma prensipleri anlatılır. • Öğrencilerden günlük hayatta kullanılan servo motor örnekleri öğrencilerden vermeleri istenir. • Raspberry-Pi ile gerekli bağlantılar sağlanıp devre kurulur ve çalıştırılır.
Uygulama Çıktısı	 <pre> import RPi.GPIO as GPIO import time servoPIN = 17 GPIO.setmode(GPIO.BCM) </pre>  import RPi.GPIO as GPIO import time servoPIN = 17 GPIO.setmode(GPIO.BCM) Kod Editör Ekran Görüntüsü

	<pre>GPIO.setup(servoPIN, GPIO.OUT) pwm = GPIO.PWM(servoPIN, 50) pwm.start(0) def aciAyarla(aci): x=(1/180)*aci + 1 duty=x*5 pwm.ChangeDutyCycle(duty) try: while True: for i in range(0,181,45): print ("aci = ",i) aciAyarla(i) time.sleep(1) except KeyboardInterrupt: pwm.stop() GPIO.cleanup()</pre>	
DEĞERLENDİRME	Etkinlik sonunda etkinliğe katılan her öğrenci için aşağıda bulunan kazanım Kontrol Listesini doldurulması tavsiye edilir. (Öğrencinin kazanımı gerçekleştirme durumuna göre Evet – Hayır bölümünü doldurunuz.)	
	Kazanım Kontrol Listesi	Evet / Hayır
	Raspberry-Pi ile LED bağlantısını sağlayan devreyi kurdu ve çalıştırdı.	
	Derleyicide yazdığı kodları Raspberry-Pi ye yükler.	
	Döngü içerisinde yapılacak işlemleri doğru belirler.	
	While True(sürekli döngü) kodunu doğru kullandı.	
HAZIRLAYANLAR		

Etkinlik No	7
Dersin Adı	ROBOTİK KODLAMA
Sınıfı	
Tarih	
Modül Adı/ Etkinliklerin Adı	Mikroişlemci ve Mikro denetleyiciler / Raspberry-Pi ile Buton LED Uygulaması
Önerilen Süre	40+40 dk.
Kazanım:	1-Butonu(Düğme) tanımlar. 2-Belirlenen programlama dilinde kontrol yapılarını kullanarak programlar geliştirir. 3-Butona tıkladıkça LED’ışık vermesi gerektiğini açıklar. 4-Pull-Down direncinin ne işe yaradığını açıklar. 5-Raspberry-Pi ile buton LED devresini kurar ve çalıştırır. 6-Koşul cümlesini ifade eder. 7-Koşul cümlesini istenen koşulu sınavacak şekilde düzenler
Öğretme öğrenme süreci	- Günlük hayatta bir elektrik devrelerini devresini dışarıdan nasıl kontrol ettiğimiz öğrencilere sorulur. Öğrencilerden gelen cevaplar doğrultusunda elektrik devrelerinde buton veya anahtarın ne işe yaradığı açıklaması yapılır. - Elektrik devrelerinde bilgisayar dili olan ikilik sayı sisteminden ve 1’ler ve 0’ların ne anlama geldiği konusunda açıklamalar yapılır. - Butonun ne işe yaradığı söylenir. - Butondan değer okunabilmesi için uçlarının Raspberry-Pi’nin hangi uçlarına bağlanması gerektiği sorusu sorulur. - Butona basıldığında LED diyotun yanması ve çekildiğinde LED diyotun sönmesi istenir. - Pull-down direnci, GPIO pinleri giriş olarak kullandığımızda sinyalin bozulmamasını sağlar. Buton basılı değilken dijital pinden okunan değer 0 V yani lojik LOW seviyesidir. Pull-down direnci, buton basılıp değer HIGH’a çekilmediği sürece bu pindeki gerilimin 0 V’ta sabit kalmasını sağlar. - Gerekli bağlantılar sağlandıktan sonra buton ve LED devresi çalıştırılır. - Ek Olarak Etkinlik sonunda öğrencilere ek bir uygulama olarak butona basıldığında kırmızı led butondan elimizin çekildiğinde sarı led yanacak şekilde kodlamaları istenir.

Uygulama Çıktısı	Kod Editör Ekran Görüntüsü <pre> import RPi.GPIO as GPIO from time import sleep GPIO.setwarnings(False) GPIO.setmode(GPIO.BCM) Buton = 24 LED = 23 GPIO.setup(Buton,GPIO.IN,pull_up_down=GPIO.PUD_UP) GPIO.setup(LED,GPIO.OUT) while True: buton_durum = GPIO.input(Buton) print(buton_durum) sleep(0.1) if buton_durum == 0: GPIO.output(LED,GPIO.HIGH) else: GPIO.output(LED,GPIO.LOW) </pre>										
DEĞERLENDİRME	Etkinlik sonunda etkinliğe katılan her öğrenci için aşağıda bulunan kazanım Kontrol Listesini doldurulması tavsiye edilir. (Öğrencinin kazanımı gerçekleştirme durumuna göre Evet – Hayır bölümünü doldurunuz.) <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 75%;">Kazanım Kontrol Listesi</th><th style="width: 25%;">Evet / Hayır</th></tr> </thead> <tbody> <tr> <td>Günlük hayatta örnek olarak televizyon kumandasında buton kullandığının farkına varır.</td><td></td></tr> <tr> <td>Raspberry-Pi ile buton LED devresini kurar ve çalıştırır.</td><td></td></tr> <tr> <td>Pull-Down direncinin ne işe yaradığını açıklar.</td><td></td></tr> <tr> <td>If-Else komutunu kullanım durumlarını açıklar</td><td></td></tr> </tbody> </table>	Kazanım Kontrol Listesi	Evet / Hayır	Günlük hayatta örnek olarak televizyon kumandasında buton kullandığının farkına varır.		Raspberry-Pi ile buton LED devresini kurar ve çalıştırır.		Pull-Down direncinin ne işe yaradığını açıklar.		If-Else komutunu kullanım durumlarını açıklar	
Kazanım Kontrol Listesi	Evet / Hayır										
Günlük hayatta örnek olarak televizyon kumandasında buton kullandığının farkına varır.											
Raspberry-Pi ile buton LED devresini kurar ve çalıştırır.											
Pull-Down direncinin ne işe yaradığını açıklar.											
If-Else komutunu kullanım durumlarını açıklar											
HAZIRLAYANLAR											

Etkinlik No	8
Dersin Adı	ROBOTİK KODLAMA
Sınıfı	
Tarih	
Modül Adı/ Etkinliklerin Adı	Mikroişlemci ve Mikro denetleyiciler / Raspberry-Pi kara şimşek uygulaması
Önerilen Süre	40+40 dk.
Kazanım:	1-Sarı, kırmızı ve yeşil renklerde LED diyotlar bulunduğunu söyler. 2-Günlük hayatta kullandığımız farklı renklerde ışıklarının nasıl çalıştığının farkına varır. 3-Raspberry-Pi ile 5 farklı led devresini kurar ve çalıştırır. 4- LED'leri yakacak şekilde programlar ve birden fazla LED'i bir arada kullanabilmeyi test eder. 5- Belirlenen programlama dilinde dizi tipinde veri tanımlar.
Öğretme öğrenme Süreci	• Derse başlarken akıllı tahtada bir trafik ışıkları sunusu açılır. • Öğrencilerin bu sunuyu dikkatlice izler ve ne gördükleri konusunda fikirlerini sınıfla paylaşması istenir. • Her Bir öğrenci fikrini sınıfla paylaştıktan sonra farklı renklerde led ışıkların olduğunu trafik ışıklarında da bu led ışıkların kullanıldığını ve bunun bir Raspberry-Pi aracılığıyla yapılacağını söylenir. • Farklı renkteki ledleri Raspberry-Pinin beş farklı GPIO pinine bağlayarak GPIO pinlerini tek tek kodlamak yerine python dilinde liste (Dizi) ile kodlayarak her led 1 saniye yanacak şekilde kodları yazılır. • Raspberry-Pi ile gerekli bağlantılar sağlanıp devre kurulur ve çalıştırılır.
Uygulama Çıktısı	Kod Editör Ekran Görüntüsü   <pre> import RPi.GPIO as GPIO import time t=0.1 GPIO.setmode(GPIO.BOARD) pinler=[40,38,36,37,35] while True: for i in pinler: GPIO.output(i, GPIO.HIGH) time.sleep(t) GPIO.output(i, GPIO.LOW) time.sleep(t) except: GPIO.cleanup() </pre>

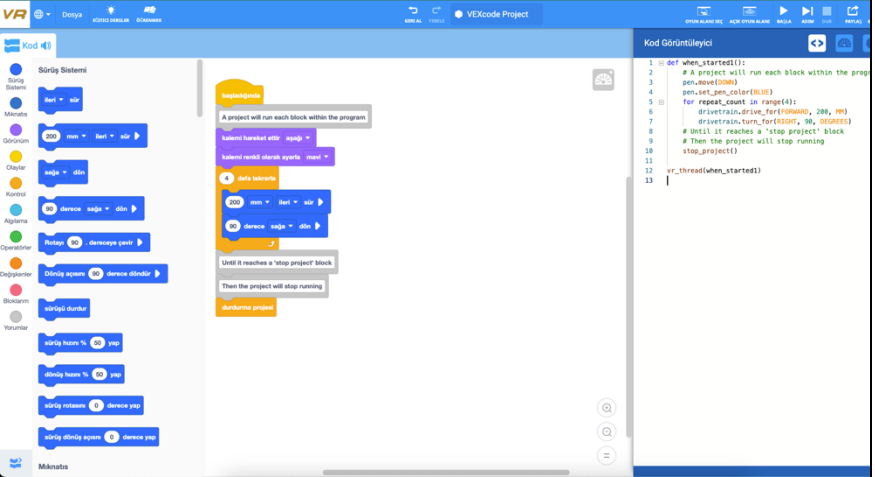
	<pre> for i in pinler: GPIO.setup(i, GPIO.OUT) try: while True: for i in pinler: GPIO.output(i, GPIO.HIGH) time.sleep(t) GPIO.output(i, GPIO.LOW) time.sleep(t) except: GPIO.cleanup() </pre>								
DEĞERLENDİRME	Etkinlik sonunda etkinliğe katılan her öğrenci için aşağıda bulunan kazanım Kontrol Listesini doldurulması tavsiye edilir. (Öğrencinin kazanımı gerçekleştirme durumuna göre Evet – Hayır bölümünü doldurunuz.) <table border="1"> <thead> <tr> <th>Kazanım Kontrol Listesi</th><th>Evet / Hayır</th></tr> </thead> <tbody> <tr> <td>Farklı renkte led ışıkların olduğunu söyledi</td><td></td></tr> <tr> <td>Liste (Dizi) kullanımının farklı kullanım durumlarını açıkladı</td><td></td></tr> <tr> <td>Raspberry-Pi ile 6 led bağlayarak Liste şeklinde gösterini uyguladı.</td><td></td></tr> </tbody> </table>	Kazanım Kontrol Listesi	Evet / Hayır	Farklı renkte led ışıkların olduğunu söyledi		Liste (Dizi) kullanımının farklı kullanım durumlarını açıkladı		Raspberry-Pi ile 6 led bağlayarak Liste şeklinde gösterini uyguladı.	
Kazanım Kontrol Listesi	Evet / Hayır								
Farklı renkte led ışıkların olduğunu söyledi									
Liste (Dizi) kullanımının farklı kullanım durumlarını açıkladı									
Raspberry-Pi ile 6 led bağlayarak Liste şeklinde gösterini uyguladı.									
HAZIRLAYANLAR									

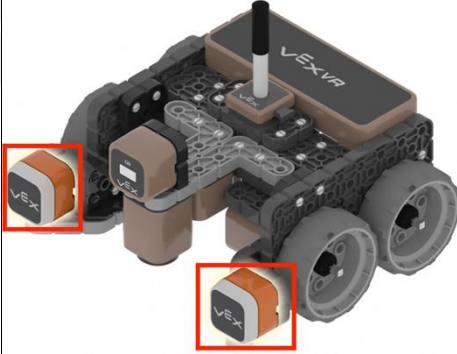
Etkinlik No	9 ve 10 Hafta
Dersin Adı	ROBOTİK KODLAMA-POJE ÇALIŞMASI
Sınıfı	
Tarih	
Modül Adı/ Etkinliklerin Adı	Mikroişlemci ve Mikro denetleyiciler / Raspberry-Pi Akıllı Ev Güvenlik Sistemi Uygulaması
Önerilen Süre	40+40+40+40 dk.
Kazanım:	1-Akıllı ev sistemleri hakkında araştırma yapar. 2-Arşatıma sonuçlarına göre akıllı ev sistemlerinde kullanılan güvenlik sistemlerini kategorik olarak ayıt eder. 3-Raspberry-Pi ile akıllı evlerde kullanılan bir güvenlik sistemi kurar. 4- Araştırma sonuçlarını doğru analiz eder 5- Proje hazırlığında zamanı doğru kullanır .
Öğretme öğrenme Süreci	• Derse girişte akıllı telefon akıllı saat akıllı ev gibi sistemlerden bahsedilerek başlanır. Öğrencilere bu sistemlere neden akıllı olarak nitelendirildiği sorusu yöneltilir ve öğrencilerden gelen cevaplar ile bir tartışma ortamı oluşturularak derse başlanır. • Soru-cevap kısmı bittikten sonra öğrencilere belirli bir süre verilerek akıllı ev sistemleri hakkında internet üzerinden bir ön araştırma yapmaları istenir. • Araştırma sonuçlarına göre öğrencilerin fikirleri tek tek her bir öğrenciye sorularak yanlış bilgiler dönüt verilerek düzeltilir. • Öğrencilere Akıllı evlerde bulunan güvenlik sistemleri ile ilgili bir proje uygulaması yapılacağı ifade edilir. • Projede daha önceki derslerde yapılan uygulamaların bir arada kullanıldığı bir sistem ortaya konulması gerektiği vurgulanarak proje şartları ve koşulları öğrencilere akıllı tahtaya yansıtılarak anlatılır.
Uygulama Çıktısı	PROJE KURALLARI - Kurulacak olan akıllı ev sisteminde en az bir adet sesör kullanılacak - Kurulacak olan akıllı ev sisteminde en az bir adet buzzer kullanılacak - Kurulacak olan akıllı ev sisteminde en az 4 adet led kullanılacak - Kurulacak olan akıllı ev sisteminde en az 1 adet servo motor kullanılacak - Tüm sistem bir adet Raspberry-Pi üzerine kurulacak ve pyton programlama dilinde kodlanarak çalıştırılacak
DEĞERLENDİRME	

EK-8. VEXcode-VR Etkinlik Planları

	1								
Dersin Adı	ROBOTİK KODLAMA								
Sınıfı									
Tarih									
Modül Adı/ Etkinliklerin Adı	VEXcode-VR Robot Editör Ekranını Tanıtımı. VEXcode-VR nedir? Niçin kullanılır? Robotik alanında kullanılan uygulama geliştirme ortamlarını amacına uygun olarak kullanır. (VEXcode-VR üyelik ve arayüz tanıtımı)								
Önerilen Süre	40+40 dk.								
Kazanım:	1. VEXcode-VR ile neler yapılabileceğini özetler. 2. Robotik alanında kullanılan uygulama geliştirme ortamlarını amacına uygun olarak kullanır. (VEXcode-VR üyelik ve arayüz tanıtımı)								
Öğretme öğrenme süreci	- VEXcode-VR Ara yüz tanıtımı araç kutusu kod ekranı Metin Tabanlı kod Ekranı tanıtımı Öğretmen derse girdiğinde öğrencilere “Aranızda daha önce hiç çevrimiçi(online) oyun oynayan var mı? - Peki aranızda daha önce hiç çevrimiçi program kullanan oldu mu? - Çevrimiçi kullandığınız programlara örnekler verebilir misiniz? - Simülasyon kelimesini daha önce duydunuz mu? Ne anlama gelmektedir? - Herhangi bir simülasyon programı kullandınız mı?” sorularının ardından verilen cevaplar hakkında konuşulur. Dikkat çekme sorularını takiben “Şimdi birlikte içinde birçok farklı özellik barındıran çevrimiçi bir simülasyon programı kullanma ne dersiniz?” sorusu sorularak öğrenciler ara yüz tanıtımı için güdülenir. Öğrencinin güdülenmesinin ardından etkinliğin iyi anlaşılması ve dersin ilerlemesini sağlayabilmek adına etkinlik etkileşimli tahtada gösterilirken öğrencilerin dikkatlice dinlemesi, anlatım bittikten sonra öğretmenin söylediği zamanda uygulamaya geçilerek öğretilen adımları sırasıyla simülasyon ortamında uygulaması istenir. Ayrıca bilgisayarlarını açarak internet tarayıcılarından arama motorlarına girmeleri istenir. Etkinliğe geçilir. Arama motoruna “Vex-Vr” anahtar kelimesi yazılarak öğrencilerden ilgili siteye girmeleri istenir.								
DEĞERLENDİRME	Etkinlik sonunda etkinliğe katılan her öğrenci için aşağıda bulunan kazanım Kontrol Listesini doldurulması tavsiye edilir. (Öğrencinin kazanımı gerçekleştirme durumuna göre Evet – Hayır bölümünü doldurunuz.) <table> <tr> <th>Kazanım Kontrol Listesi</th><th>Evet / Hayır</th></tr> <tr> <td>VEXcode-VR ile neler yapılabileceğini özetledi.</td><td></td></tr> <tr> <td>VEXcode-VR kullanım ve robot simüle ortamlarını karşılaştırdı.</td><td></td></tr> <tr> <td>VEXcode-VR Blok kodlar ile Metin Kodları birbiri ile karşılaştırdı.</td><td></td></tr> </table>	Kazanım Kontrol Listesi	Evet / Hayır	VEXcode-VR ile neler yapılabileceğini özetledi.		VEXcode-VR kullanım ve robot simüle ortamlarını karşılaştırdı.		VEXcode-VR Blok kodlar ile Metin Kodları birbiri ile karşılaştırdı.	
Kazanım Kontrol Listesi	Evet / Hayır								
VEXcode-VR ile neler yapılabileceğini özetledi.									
VEXcode-VR kullanım ve robot simüle ortamlarını karşılaştırdı.									
VEXcode-VR Blok kodlar ile Metin Kodları birbiri ile karşılaştırdı.									
HAZIRLAYANLAR									

Etkinlik No	2
Dersin Adı	ROBOTİK KODLAMA
Sınıfı	
Tarih	
Modül Adı/ Etkinliklerin Adı	VEXcode-VR Robot Kodlamaya Başlıyorum.
Önerilen Süre	40+40 dk.
Kazanım:	1. Tercih edilen robotu tanır. 2. VEXcode-VR simülasyon robotun özelliklerini sensor ve motor bağlantılarını tanır.
Öğretme öğrenme süreci	Derse “Sizlere çok ilginizi çekeceğinizi düşündüğüm birkaç video izleteceğim. Ardından videolarla ilgili görüşlerinizi alacağım.” denilerek başlanır. Ders öncesinde öğretmenin hazırlamış, olduğu VEXcode-VR ile yapılmış, proje örneklerini içeren videolar öğrencilere izletilir. VEXcode-VR ile ilgili videolar izlendikten sonra öğretmen öğrencilere “Sizlerde böyle programlayarak benzer tasarımlar yapmak ister misiniz?” sorusunu yöneltir. “Sizlerde robotik kodlama dersleriniz tamamlandığında robotik alanında benzer çalışmalara yapabilecekleri söylenir. Derse geçerken etkinliğin iyi anlaşılması, düzenli ve disiplinli ilerlemesini sağlayabilmek adına öğrencilerin etkinliği etkileşimli tahtada gösterilirken dikkatlice dinlemesi, anlatım bittikten sonra öğretmenin söylediği zamanda uygulamaya geçilerek öğretilen adımları sırasıyla simülasyon ortamında uygulaması istenir. Etkinliğe başlarken öğrencilerden videolarda izledikleri projelerin nasıl yapılmış olabileceğine dair görüşleri alınır. En fazla hangi projenin dikkatlerini çektiği sorulur. Görüşlerin alınmasının ardından VEXcode-VR ortamında robotların programlamak için iyi tanımamız gerektiği ve programlama dilinin de iyi kullanabilmek gerektiği söylenir. VEXcode-VR simülasyon ortamında kod bloklarının bulunduğu araç menüsü ve her bir menünün kendine has işlevlerinin olduğu anlatılır. Programlama yaparken blok kodlara karşılık gelen metin kodlama ekranının da açık olması gerektiği özellikle vurgulanır.

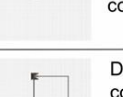
											
DEĞERLENDİRME	<table border="1"> <thead> <tr> <th>Kazanım Kontrol Listesi</th><th>Evet / Hayır</th></tr> </thead> <tbody> <tr> <td>Simülasyon ortamına giriş yapar.</td><td></td></tr> <tr> <td>Simülasyon ortamında araç menüsünde kod blokları ve metin dilini karşılaştırır.</td><td></td></tr> <tr> <td>Simülasyon ortamında kodlayacağı robotu seçer</td><td></td></tr> <tr> <td>Kodlayacağı robotun özellikleri bilir</td><td></td></tr> </tbody> </table>	Kazanım Kontrol Listesi	Evet / Hayır	Simülasyon ortamına giriş yapar.		Simülasyon ortamında araç menüsünde kod blokları ve metin dilini karşılaştırır.		Simülasyon ortamında kodlayacağı robotu seçer		Kodlayacağı robotun özellikleri bilir	
Kazanım Kontrol Listesi	Evet / Hayır										
Simülasyon ortamına giriş yapar.											
Simülasyon ortamında araç menüsünde kod blokları ve metin dilini karşılaştırır.											
Simülasyon ortamında kodlayacağı robotu seçer											
Kodlayacağı robotun özellikleri bilir											
HAZIRLAYANLAR											

Etkinlik No	3
Dersin Adı	ROBOTİK KODLAMA
Sınıfı	
Tarih	
Modül Adı/ Etkinliklerin Adı	VEXcode-VR ile robotu Hareket Ettirme.
Önerilen Süre	40+40 dk.
Kazanım:	1. Belirlenen programlama dilinde tekrarlı yapıları kullanarak programlar geliştirir. 2. VEXcode-VR yazdığı kodu test eder. 3. VEXcode-VR da robotu sürekli hareket edecek kodu yazar. 4. VEXcode-VR simülasyon ortamında hazırladığı kodunu yazılım geliştirme ortamında yazar. 5. Sürekli döngü kavramını açıklar. 6. Sürekli döngü içerisinde yapılacak işlemleri doğru belirler.
Öğretme öğrenme Süreci	• Sensörler robolar için çok önemli devre elemanlarıdır. Bir VR Robotunun gerçek anlamda bir robot olarak kabul edilebilmesi için çevresini algılayabilmesi ve onunla etkileşime girebilmesi gerekir. Bu genellikle Duyu → Düşün → Harekete Geç karar döngüsü olarak adlandırılır.  Tampon Sensörü Bu etkinlikte, fiziksel teması algılayabilen bir sensör olan Tampon Sensörünü kullanacağız. Tampon Sensörü, VR Robotuna bir nesne veya duvarla temas ettiğinde bunu algılamak için kullanılır. Bir VR Robot üzerinde karar vermek için Sol veya Sağ Tampon olmak üzere ön tarafta iki adet sensör bulunur. Tampona Basılan Komutlar Tampon Sensörü, VR Robotun bir nesneye veya duvara temas edip etmediğini belirlemek için kullanılır. Bu veriler daha sonra bir projede kullanılabilir ve böylece VR Robot, Tampon Sensörüne basıldığında karar verebilir. Tampon Sensörünün durumunu kontrol etmek için bir pythonda left_bumper.pressed ve right_bumper.pressed komutlarını kullanın . Left_bumper.pressed ve right_bumper.pressed , Bumper'a basıldığında True ve basılmadığında False sinyali gönderen Boolean raporör komutlarıdır. left_bumper.pressed ve right_bumper.pressed komutları tipik olarak while döngüsü gibi bir kontrol yapısıyla birlikte kullanılır , böylece VR Robotu bir karar vermek için Tampon Sensöründen gelen verileri kullanabilir. Yeni bir Metin projesinin parçası olarak otomatik olarak doldurulan Aktarma Organı komutunu değiştirmek için while komutunu çalışma alanına sürükleyin veya yazın. Döngünün içinde bir bekleme komutunun olduğuna dikkat edin. Bekleme komutunu kaldırmayın; bu, VR Robotun projeyi doğru şekilde yürütmesini sağlayacaktır.

	<pre> def main (): while condition: wait(5,MSEC) While komutundaki 'while'dan sonra not operatörünü yazın. def main (): while not condition: wait(5,MSEC) left_bumper.pressed komutunu while döngüsünün 'koşuluna' yazın . Döngü içindeki komutlar, koşul Doğru olmadığında, yani Sol Tampon Sensörüne basılmadığında çalıştırılır . def main (): while not left_bumper.pressed(): wait(5,MSEC) While döngüsünün içine, sürücü komutunu yazın. Döngünün amaçlandığı gibi çalışması için, Drive komutunun ve wait komutunun while döngüsü içinde doğru girintili olduğundan emin olun. Ekranda ki kılavuz çizgileri görsel yardım olarak kullanabilirsiniz. def main (): while not left_bumper.pressed(): drivetrain.drive(FORWARD) wait(5,MSEC) Durdurma komutunu while döngüsünün dışına yazın. Stop komutunun while döngüsüyle hizalandığından ve girintili olmadığından emin olun, böylece amaçlandığı gibi döngünün dışında çalışır. Ekranda ki klavuz çizgileri görsel yardım olarak kullanabilirsiniz. def main (): while not left_bumper.pressed(): drivetrain.drive(FORWARD) wait(5,MSEC) drivetrain.stop() Sürüş komutu, VR Robotun, left_bumper.pressed komutuyla Tampon Sensörünün durumunu kontrol ederken ileri doğru ilerlemeye devam etmesini sağlar . left_bumper.pressed komutu True olarak raporlandığında , VR Robot döngünün dışındaki bir sonraki komuta geçecek ve sürüşü durduracaktır </pre>						
Uygulama Çıktısı	<table border="1"> <tr> <td colspan="2" data-bbox="564 1682 1458 1783"></td> </tr> <tr> <td data-bbox="564 1783 1011 1823"></td><td data-bbox="1011 1783 1458 1823"></td></tr> <tr> <td colspan="2" data-bbox="564 1823 1458 1890"></td> </tr> </table>						

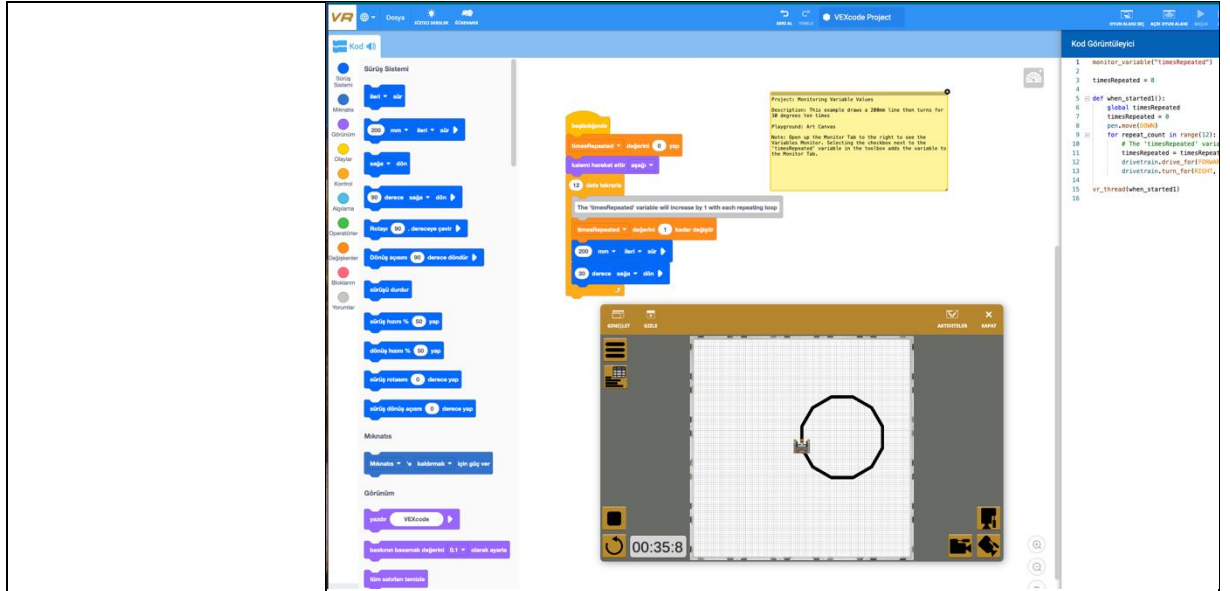
DEĞERLENDİRME	Etkinlik sonunda etkinliğe katılan her öğrenci için aşağıda bulunan kazanım Kontrol Listesini doldurulması tavsiye edilir. (Öğrencinin kazanımı gerçekleştirme durumuna göre Evet – Hayır bölümünü doldurunuz.)	
	Kazanım Kontrol Listesi	Evet / Hayır
	VEXcode-VR while döngüsünü uygun biçimde yazarak kullandı.	
	VEXcode-VR da bekleme komutunu uygun biçimde kullandı	
	Döngü içerisinde yapılacak işlemleri doğru belirler.	
	While True(sürekli döngü) not anahtar sözcüğünü doğru kullandı.	
HAZIRLAYANLAR		

Etkinlik No	4
Dersin Adı	ROBOTİK KODLAMA
Sınıfı	
Tarih	
Modül Adı/ Etkinliklerin Adı	VEXcode-VR ile robotu Hareket Ettirme.
Önerilen Süre	40+40 dk.
Kazanım:	1. Belirlenen programlama dilinde tekrarlı yapıları kullanarak programlar geliştirir. 2. VEXcode-VR yazdığı kodu test eder. 3. VEXcode-VR da robotu belirli sayıda hareket edecek kodu yazar. 4. VEXcode-VR simülasyon ortamında hazırladığı kodunu yazılım geliştirme ortamında yazar. 5. For döngü kavramını açıklar. 6. For döngü içerisinde yapılacak işlemleri doğru belirler.
Öğretme öğrenme Süreci	Bir etkinlikte for döngüsü kullandığında robotun nasıl belirli sayıda hareket ettireceğiz. Örneğin bu iki kod örneği VR Robotun aynı şekilde hareket etmesini sağlayacaktır. Ancak for döngüsüne sahip proje, bunu yapabilmek için iki motoru dört kez tekrarlar. Bir dizi komutu belirli sayıda tekrarlamak için for döngüsü kullanılır. For döngüsü, içindeki komut dizisi boyunca tekrarlanır (tekrarlamalar olarak da bilinir). Python for döngüsünde, 'value' değişkeni geçerli tekrarlamayı izlemek için kullanılır ve 'in' anahtar sözcüğü yineleme sayısını belirtmek için range işlevini kullanır. Range işlevi, döngünün kaç kez veya tekrarlanması gerektiğini tanımlar. Hareket istenilen sayıda tekrarlamak için range fonksiyonunun parametresini ayarlayın. Range işlevi temel olarak döngünün tamamlanabilmesi için neyi yinelemesi gerektiğini belirten geçici bir liste oluşturur. Bir karenin dört kenarını çizmek gibi tekrarlanan davranışlar durumunda range işlevi, VR Robotunun bu komutları dört kez yürütmesini sağlamak için geçici bir (0, 1, 2, 3) listesi kullanır. VR Robotun hareketi burada gösterildiği gibi menzil fonksiyonunun parametresine bağlı olarak değişecektir.

	<pre>def main(): pen.move(DOWN) for value in range(0): drivetrain.drive_for(FORWARD, 600, MM) drivetrain.turn_for(RIGHT, 90, DEGREES) wait(5, MSEC)</pre>		Drivetrain commands executed 0 times
	<pre>def main(): pen.move(DOWN) for value in range(1): drivetrain.drive_for(FORWARD, 600, MM) drivetrain.turn_for(RIGHT, 90, DEGREES) wait(5, MSEC)</pre>		Drivetrain commands executed 1 time
	<pre>def main(): pen.move(DOWN) for value in range(2): drivetrain.drive_for(FORWARD, 600, MM) drivetrain.turn_for(RIGHT, 90, DEGREES) wait(5, MSEC)</pre>		Drivetrain commands executed 2 times
	<pre>def main(): pen.move(DOWN) for value in range(3): drivetrain.drive_for(FORWARD, 600, MM) drivetrain.turn_for(RIGHT, 90, DEGREES) wait(5, MSEC)</pre>		Drivetrain commands executed 3 times
	<pre>def main(): pen.move(DOWN) for value in range(4): drivetrain.drive_for(FORWARD, 600, MM) drivetrain.turn_for(RIGHT, 90, DEGREES) wait(5, MSEC)</pre>		Drivetrain commands executed 4 times, completing all 4 sides of the square
	<pre>def main(): pen.move(DOWN) for value in range(5): drivetrain.drive_for(FORWARD, 600, MM) drivetrain.turn_for(RIGHT, 90, DEGREES) wait(5, MSEC)</pre>		Drivetrain commands executed 5 times, completing the square then tracing over the first side again
Range fonksiyonu dörtten beşe değiştiğinde karenin çizilen şeklinin değişmediğine dikkat edin. Bu nedenle, varsayılan for döngüsü kullanmak aynı kare şeklinin çizilmesini sağlar, ancak VR Robot hareketleri altı kez daha tekrarlamaya devam eder. VR Robot, for döngü başlığından son girintili komuta kadar komutlar arasında hareket edecektir. For döngüsü, VR Robotunun for döngüsü dışındaki bir sonraki komuta geçmeden önce döngü içindeki eylemleri belirtilen sayıda tekrarlaması gerektiğini belirtir. Girinti for döngüsünün içinde hangi komutların olduğunu ve tekrarlanması gerektiğini gösterir.			
Uygulama Çıktısı			
	<pre>def main(): pen.move(DOWN) for value in range(0): drivetrain.drive_for(FORWARD, 600, MM) drivetrain.turn_for(RIGHT, 90, DEGREES) wait(5, MSEC)</pre>		Drivetrain commands executed 0 times
	<pre>def main(): pen.move(DOWN) for value in range(1): drivetrain.drive_for(FORWARD, 600, MM) drivetrain.turn_for(RIGHT, 90, DEGREES) wait(5, MSEC)</pre>		Drivetrain commands executed 1 time
	<pre>def main(): pen.move(DOWN) for value in range(2): drivetrain.drive_for(FORWARD, 600, MM) drivetrain.turn_for(RIGHT, 90, DEGREES) wait(5, MSEC)</pre>		Drivetrain commands executed 2 times
	<pre>def main(): pen.move(DOWN) for value in range(3): drivetrain.drive_for(FORWARD, 600, MM) drivetrain.turn_for(RIGHT, 90, DEGREES) wait(5, MSEC)</pre>		Drivetrain commands executed 3 times
	<pre>def main(): pen.move(DOWN) for value in range(4): drivetrain.drive_for(FORWARD, 600, MM) drivetrain.turn_for(RIGHT, 90, DEGREES) wait(5, MSEC)</pre>		Drivetrain commands executed 4 times, completing all 4 sides of the square
	<pre>def main(): pen.move(DOWN) for value in range(5): drivetrain.drive_for(FORWARD, 600, MM) drivetrain.turn_for(RIGHT, 90, DEGREES) wait(5, MSEC)</pre>		Drivetrain commands executed 5 times, completing the square then tracing over the first side again

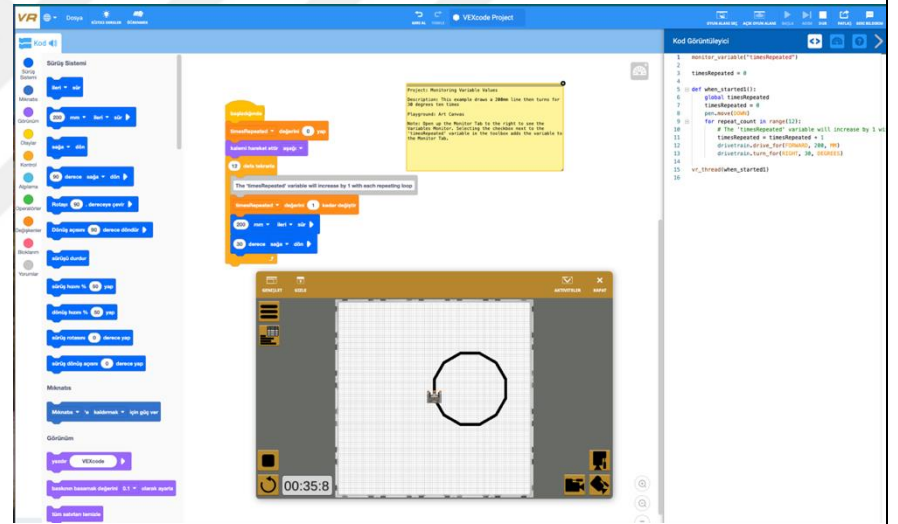
DEĞERLENDİRME	Etkinlik sonunda etkinliğe katılan her öğrenci için aşağıda bulunan kazanım Kontrol Listesini doldurulması tavsiye edilir. (Öğrencinin kazanımı gerçekleştirme durumuna göre Evet – Hayır bölümünü doldurunuz.)	
	Kazanım Kontrol Listesi	Evet / Hayır
	VEXcode-VR while for döngüsünü uygun biçimde yazarak kullandı.	
	VEXcode-VR da for döngüsüne bekleme komutu ekledi.	
	For Döngü içerisinde yapılacak işlemleri doğru belirler.	
HAZIRLAYANLAR		

Etkinlik No	5
Dersin Adı	ROBOTİK KODLAMA
Sınıfı	
Tarih	
Modül Adı/ Etkinliklerin Adı	VEXcode-VR ile robotu ile değişken tanımlama etkinliği.
Önerilen Süre	40+40 dk.
Kazanım:	1. Değişken kavramını tanımlar. 2. Değişkeni programlama diline uygun şekilde tanımlar. 3. Belirlenen programlama dilinde değişkenleri ve sabitleri uygun şekilde kullanarak programlar geliştirir. 4. VEXcode-VR robotunda çizgi çizme komutunu yazar. 5. VEXcode-VR robotun değişken tanımlayarak robota düzgün çokgen çizdirir.
Öğretme öğrenme Süreci	Etkinlikte: Değişken Değerlerin İzlenmesi Açıklama: etkinliğimizde robot 200 mm’lik uzaklıklar da eşit aralıklarda çizgi çizer ve ardından 12 kez 30 derece döner. Programın sonunda 12 kenarlı bir çokgen düzgün bir şekilde çizmiş olacaktır. <pre> timesRepeated = 0 def when_started1(): global timesRepeated timesRepeated = 0 pen.move(DOWN) for repeat_count in range(12): # The 'timesRepeated' variable will increase by 1 with each repeating loop timesRepeated = timesRepeated + 1 drivetrain.drive_for(FORWARD, 200, MM) drivetrain.turn_for(RIGHT, 30, DEGREES) vr_thread(when_started1) Oyun Alanı: Sanat Kanvası </pre>



Not: Değişkenler Monitörünü görmek için sağdaki Monitör Sekmesini açın. Araç kutusunda 'timesRepeated' değişkeninin yanındaki onay kutusunun seçilmesi değişkeni Monitör Sekmesine ekler.

Uygulama Çıktısı



DEĞERLENDİRME

- Etkinlik sonunda etkinliğe katılan her öğrenci için aşağıda bulunan kazanım Kontrol Listesini doldurulması tavsiye edilir. (Öğrencinin kazanımı gerçekleştirme durumuna göre Evet – Hayır bölümünü doldurunuz.)

Kazanım Kontrol Listesi

Evet / Hayır

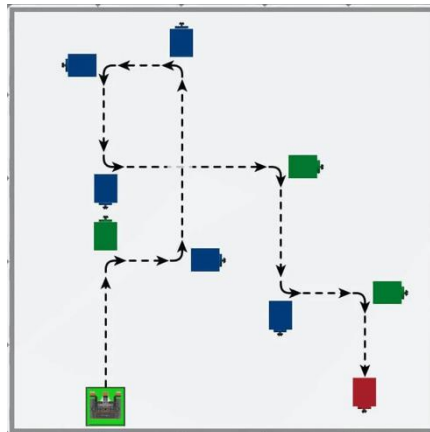
	Değişkeni doğru tanımladı.	
	Tanımladığı değişkeni programda doğru kullandı.	
	Programlama da değişken türlerini ayırt eder.	
	Farklı Değişkenler Tanımlar	

Bir if koşulunun çalışabilmesi için içinde bir komut olması gerekir, bu nedenle pass ifadesi eklenir. If ifadesine komut eklediğinizde, bunlar pass ifadesinin yerini alacaklardır .

```
def main():
    if condition:
        pass

def main():
    if front_eye.detects(GREEN):
        drivetrain.turn_for(RIGHT, 90, DEGREES)
```

Uygulama Çıktısı



DEĞERLENDİRME

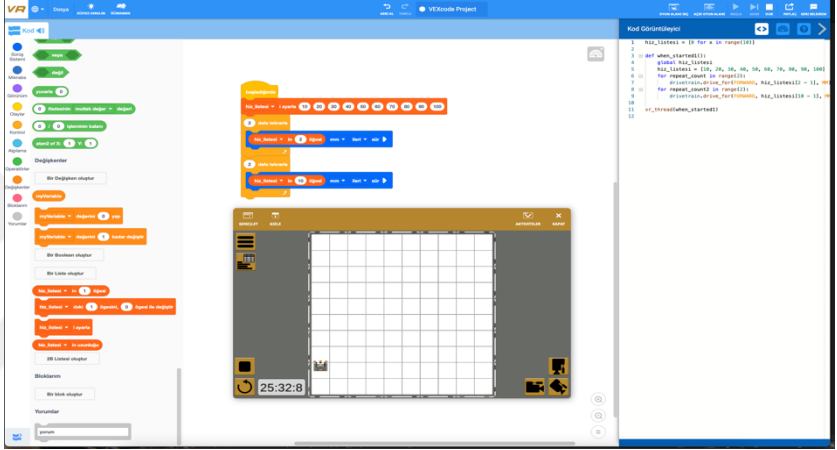
- Etkinlik sonunda etkinliğe katılan her öğrenci için aşağıda bulunan kazanım Kontrol Listesini doldurulması tavsiye edilir. (Öğrencinin kazanımı gerçekleştirme durumuna göre Evet – Hayır bölümünü doldurunuz.)

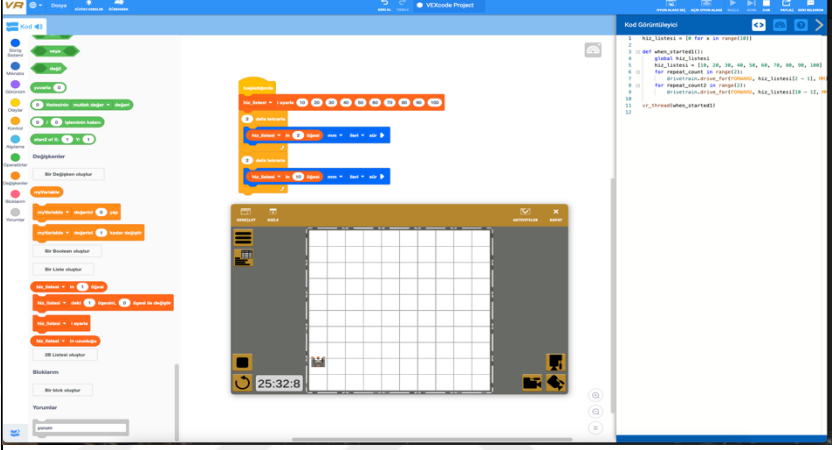
Kazanım Kontrol Listesi

Evet / Hayır

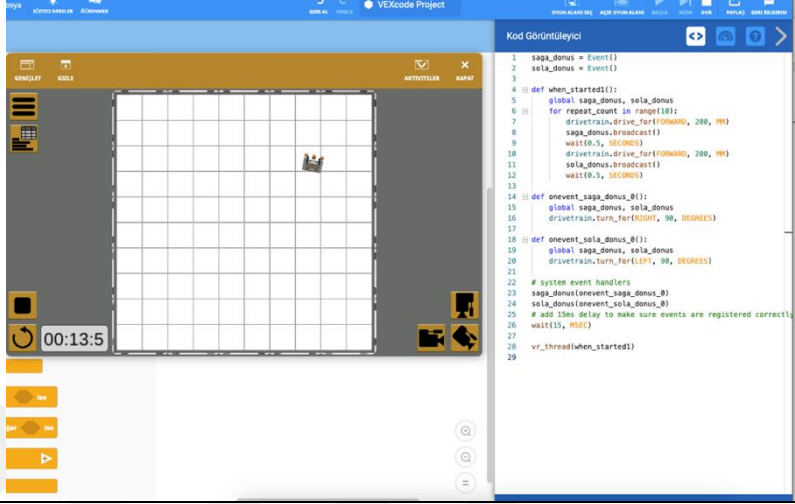
İf Komutunu doğru amaç ile kullandı.

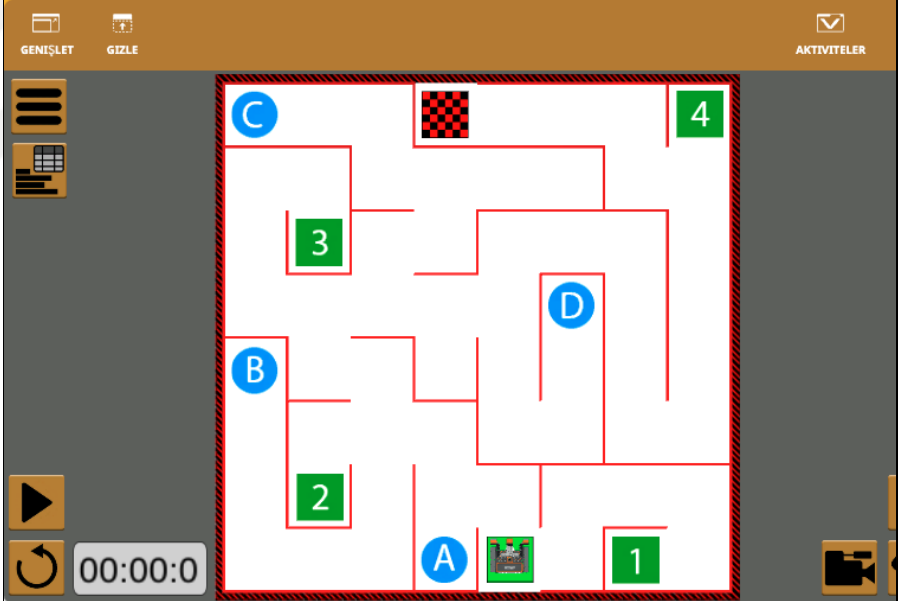
	Disk labirenti algoritmasını doğru kurdu	
	If komutunu yazımını doğru yazdı	
	Farklı algoritma kurarak kurduğu algorithmaya uygun program yazdı.	
HAZIRLAYANLAR		

Etkinlik No	7
Dersin Adı	ROBOTİK KODLAMA
Sınıfı	
Tarih	
Modül Adı/ Etkinliklerin Adı	VEXcode-VR ile robotu liste (Dizi) oluşturarak farklı hızlarda sürme etkinliği
Önerilen Süre	40+40 dk.
Kazanım:	1- VEXcode-VR robotun belirlediği hız dizisinde hareket ettirir. 2- VEXcode-VR robotun farklı hızlarda ilerletir. 3- Belirlenen programlama dilinde dizi tipinde veri tanımlar.
Öğretme öğrenme Süreci	 Derse Girişte Öğrencilere; Son yıllarda üretilen elektrikli ve elektronik cihazlara neden akıllı cihazlar olarak nitelendirildiği sorusu ile başlanır. Öğrencilerden gelen cevaplar bittikten sonra akıllı cihazları anlatan video akıllı tahtadan izletilerek makinaların ya da cihazların akıllı olma nedenleri öğrenciler ile paylaşılır. Ve etkinliğimizde VEXcode-VR robotumuzu kendi hızını otonom bir şekilde ayarlaya bilecek bir programlama yazacağız. İlk olarak robotumuzun 10 farklı hızda ilerlemesini sağlamanın programlama da dizi oluşturarak yapılacağı python dilinde liste (Dizi) oluşturma kuralları anlatılır ve konu le ilgili kodlar editöre aktarılır. <pre>hiz_listesi = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100]</pre> Hız listesinde 10 ar aralıklarla bir liste oluşturulur ve robotun İki farklı hızda ilerlemesi sağlanır. <pre>hiz_listesi = [0 for x in range(10)]</pre> <pre>def when_started1(): global hiz_listesi</pre>

	<pre> hiz_listesi = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100] for repeat_count in range(2): drivetrain.drive_for(FORWARD, hiz_listesi[2 - 1], MM) for repeat_count2 in range(2): drivetrain.drive_for(FORWARD, hiz_listesi[10 - 1], MM) vr_thread(when_started1) </pre> Kodlar yazıldıktan sonra çalıştırılır ve robotun hareketleri izlenir.		
Uygulama Çıktısı			
DEĞERLENDİRME	Etkinlik sonunda etkinliğe katılan her öğrenci için aşağıda bulunan kazanım Kontrol Listesini doldurulması tavsiye edilir. (Öğrencinin kazanımı gerçekleştirme durumuna göre Evet – Hayır bölümünü doldurunuz.)		
	Kazanım Kontrol Listesi	Evet / Hayır	
	VEXcode-VR robotunu programlamada dizi tanımlamayı doğru kullandı.		
	Dizi İçerisine elemanları doğru bir şekilde kodladı		
	Dizi elemanlarını uygun olarak robotu hareket ettirdi		
	Farklı algoritma kurarak kurduğu algorithmaya uygun program yazdı.		
HAZIRLAYANLAR			

Etkinlik No	8
Dersin Adı	ROBOTİK KODLAMA
Sınıfı	
Tarih	
Modül Adı/ Etkinliklerin Adı	VEXcode-VR ile robotu ile programlama dilinde fonksiyon tanımlama etkinliği
Önerilen Süre	40+40 dk.
Kazanım:	1- Fonksiyon kavramını açıklar. 2- Belirlenen programlama dilinde parametre alan değer döndüren fonksiyon içeren programlar geliştirir. 3- Python fonksiyon içerisinde yapılacak işlemleri doğru belirler. 4- VEXcode-VR Robota dönüş fonksiyonu yazar.
Öğretme öğrenme süreci	Derse girişte öğrencilere robotların her an yapması gereken birtakım hareketlerin olduğu ve bu hareketler için bir program yazılırken her zaman kod yazmanın zor zaman kaybı ve kod kalabalığına neden olacağı bu nedenle programlama dillerinde fonksiyon oluşturmanın programlamayı kolaylaştıracağı anlatılır. Python programlama dilinde fonksiyon oluşturmanın kuralları anlatılır İlk Fonksiyonda robotumuzu sağa 90 derece döndürmek için bir fonksiyon oluşturuyoruz. <pre>def onevent_saga_donus_0(): global saga_donus, sola_donus drivetrain.turn_for(RIGHT, 90, DEGREES)</pre> İkinci fonksiyonumuzda ise robotumuzu sola döndürmek için bir fonksiyon oluşturuyoruz. <pre>def onevent_sola_donus_0(): global saga_donus, sola_donus drivetrain.turn_for(LEFT, 90, DEGREES)</pre> Fonksiyonlarımızı oluşturduktan sonra robotumuzu artık oluşturduğumuz fonksiyonlar doğrultusunda hareketlerini istediğimi yönde değiştirebiliriz. Şimdi VEXcode-VR Robotuna Bulunduğu köşeden çaprazdaki köşeye sağ ve sol dönüş fonksiyonlarını kullanarak bir program oluşturalım. <pre>saga_donus = Event() sola_donus = Event() def when_started1(): global saga_donus, sola_donus for repeat_count in range(10): drivetrain.drive_for(FORWARD, 200, MM) saga_donus.broadcast() wait(0.5, SECONDS) drivetrain.drive_for(FORWARD, 200, MM) sola_donus.broadcast() wait(0.5, SECONDS) def onevent_saga_donus_0(): global saga_donus, sola_donus drivetrain.turn_for(RIGHT, 90, DEGREES) def onevent_sola_donus_0():</pre>

	<pre> global saga_donus, sola_donus drivetrain.turn_for(LEFT, 90, DEGREES) saga_donus(onevent_saga_donus_0) sola_donus(onevent_sola_donus_0) wait(15, MSEC) vr_thread(when_started1) </pre>								
Uygulama Çıktısı									
DEĞERLENDİRME	Etkinlik sonunda etkinliğe katılan her öğrenci için aşağıda bulunan kazanım Kontrol Listesini doldurulması tavsiye edilir. (Öğrencinin kazanımı gerçekleştirme durumuna göre Evet – Hayır bölümünü doldurunuz.) <table border="1"> <thead> <tr> <th>Kazanım Kontrol Listesi</th><th>Evet / Hayır</th></tr> </thead> <tbody> <tr> <td>Dönüş Fonksiyonlarını yazarak robotu programladı.</td><td></td></tr> <tr> <td>Fonksiyon farklı kullanım durumlarını açıkladı</td><td></td></tr> <tr> <td>Fonksiyon içeriğini doğru yazarak kodu çalıştırdı.</td><td></td></tr> </tbody> </table>	Kazanım Kontrol Listesi	Evet / Hayır	Dönüş Fonksiyonlarını yazarak robotu programladı.		Fonksiyon farklı kullanım durumlarını açıkladı		Fonksiyon içeriğini doğru yazarak kodu çalıştırdı.	
Kazanım Kontrol Listesi	Evet / Hayır								
Dönüş Fonksiyonlarını yazarak robotu programladı.									
Fonksiyon farklı kullanım durumlarını açıkladı									
Fonksiyon içeriğini doğru yazarak kodu çalıştırdı.									
HAZIRLAYANLAR	ROBOTİK KODLAMA KOMİSYONU								

Etkinlik No	9-10
Dersin Adı	ROBOTİK KODLAMA-POJE ÇALIŞMASI
Sınıfı	
Tarih	
Modül Adı/ Etkinliklerin Adı	VEXcode-VR Sümülasyon aracında Duvar Labirent Oyun Alanı Kullanımı Projesi
Önerilen Süre	40+40+40+40 dk.
Kazanım:	1-VEXcode-VR duvar labirent alanını seçer. 2-Duvar labirent oyun alanında A bölgesine ulaşmanın algoritmasını yazar. Algoritmada if koşulunu kullanır 3- Duvar labirent oyun alanında B bölgesine ulaşmanın algoritmasını yazar. Algoritmada for döngüsünü kullanır. 4- Duvar labirent oyun alanında C bölgesine ulaşmanın algoritmasını yazar. Algoritmada while döngüsünü kullanır 5- Duvar labirent oyun alanında D bölgesine ulaşmanın algoritmasını yazar.Algoritmasında Liste(Dizi) oluşturur 6- En az iki farklı hareket için kendi fonksiyonunu yazar. 7- Algoritmalarda VEXcode-VR robot üzerinde bulunan sensörleri kullanır. 8-
Öğretme öğrenme Süreci	
Uygulama Çıktısı	Derse girişte VEXcode -VR ücretsiz ulaşım sağladığı ve projeler yapabileceğimiz oyun alanları ile ilgili bilgiler verilir. Öğrencilere bu her oyun alanında farklı programlamaların yapılabileceği her oyun alanı için farklı algoritmalar oluşturulması gerekeceği koşunda bilgilendirmeler yapıldıktan sonra he bir öğrencinin kendi bilgisayarında Duvar labirent oyun alanı başlatılır ve bu oyun alanının bir kaç dakika öğrenciler tarafından analiz edilmesi istenir . Duvar labirenti oyun alanında 4 adet sayı ile gösterilen hedef ve 4 adet harf ile gösterilen hedef bulunmaktadır. Robotun her bir hedefe ulaşması için farklı algoritmalar yazılması gerektiği ve bu algoritmaların farklı programlamalar ile gerçekleşmesi gerektiği öğrencilere anlatılır.

	DUVAR LABİRENTİ OYUN ALANI KURALLARI - Oyun alanı algoritmaları oluştururken en az bir adet sensör kullanılacak - Robot pyton programlama dili kullanılarak yapılacaktır.
DEĞERLENDİRME	

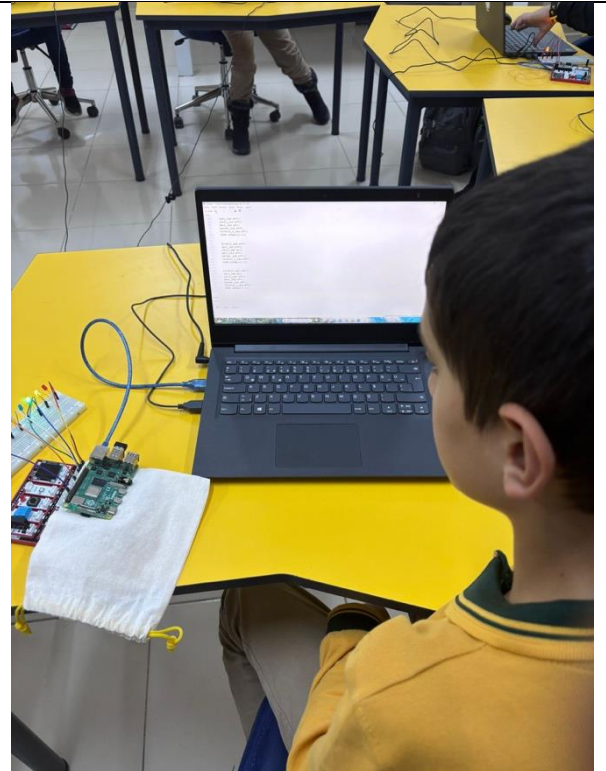
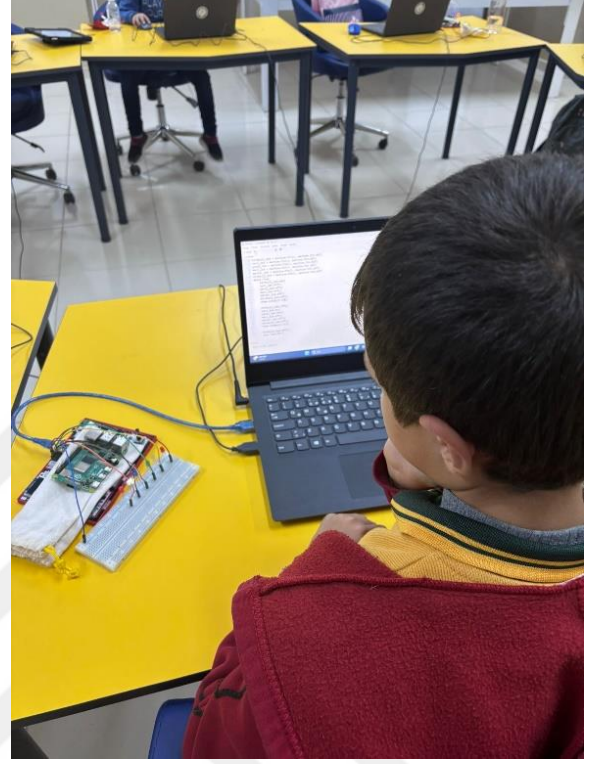
	• Etkinlik sonunda etkinliğe katılan her öğrenci için aşağıda bulunan kazanım Kontrol Listesini doldurulması tavsiye edilir. (Öğrencinin kazanımı gerçekleştirme durumuna göre Evet – Hayır bölümünü doldurunuz.) <table border="1"> <thead> <tr> <th>Kazanım Kontrol Listesi</th><th>Evet / Hayır</th></tr> </thead> <tbody> <tr> <td>Labirentte sensörleri kullandı ve doğru bir şekilde kodladı</td><td></td></tr> <tr> <td>Robot A noktasına doğru bir şekilde ilerledi</td><td></td></tr> <tr> <td>Robot 3 noktasına doğru bir şekilde ilerledi</td><td></td></tr> <tr> <td>Robot bitiş noktasına doğru bir şekilde ilerledi</td><td></td></tr> <tr> <td>Programlama da yaptığı hataları buldu ve düzeltti.</td><td></td></tr> </tbody> </table>	Kazanım Kontrol Listesi	Evet / Hayır	Labirentte sensörleri kullandı ve doğru bir şekilde kodladı		Robot A noktasına doğru bir şekilde ilerledi		Robot 3 noktasına doğru bir şekilde ilerledi		Robot bitiş noktasına doğru bir şekilde ilerledi		Programlama da yaptığı hataları buldu ve düzeltti.	
Kazanım Kontrol Listesi	Evet / Hayır												
Labirentte sensörleri kullandı ve doğru bir şekilde kodladı													
Robot A noktasına doğru bir şekilde ilerledi													
Robot 3 noktasına doğru bir şekilde ilerledi													
Robot bitiş noktasına doğru bir şekilde ilerledi													
Programlama da yaptığı hataları buldu ve düzeltti.													
HAZIRLAYANLAR	ROBOTİK KODLAMA KOMİSYONU												

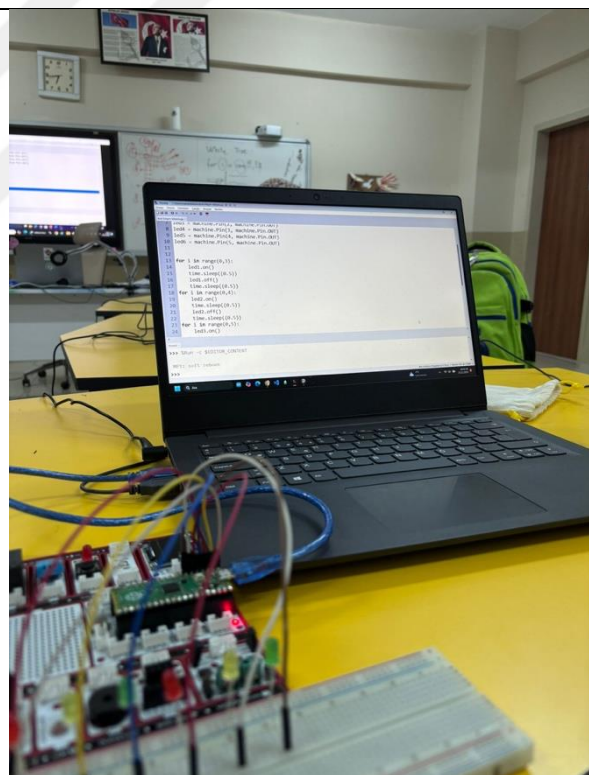
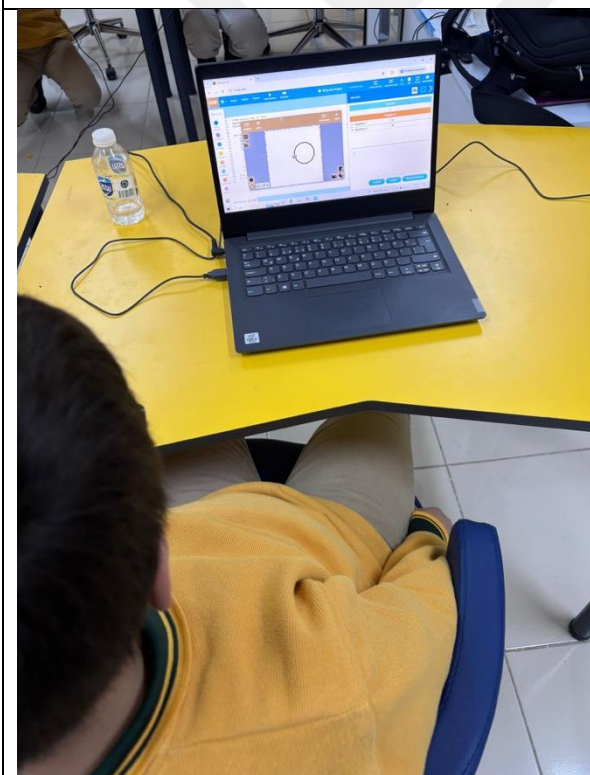
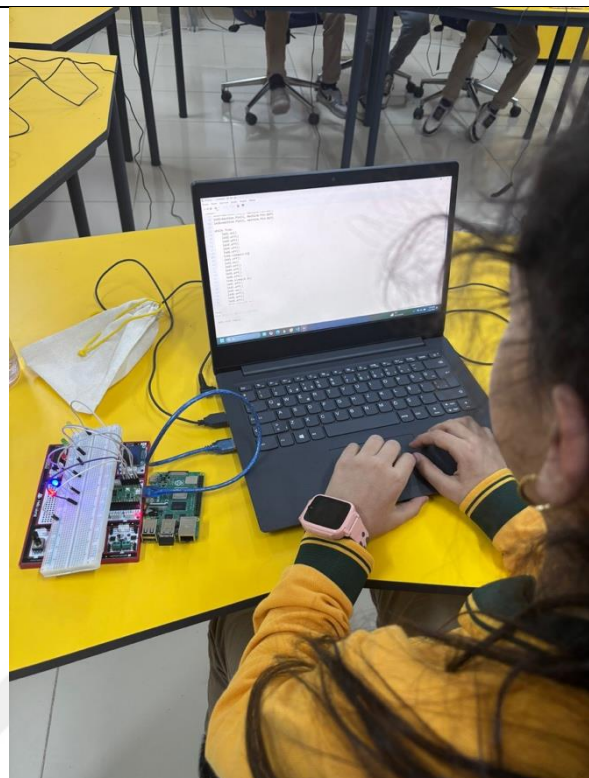
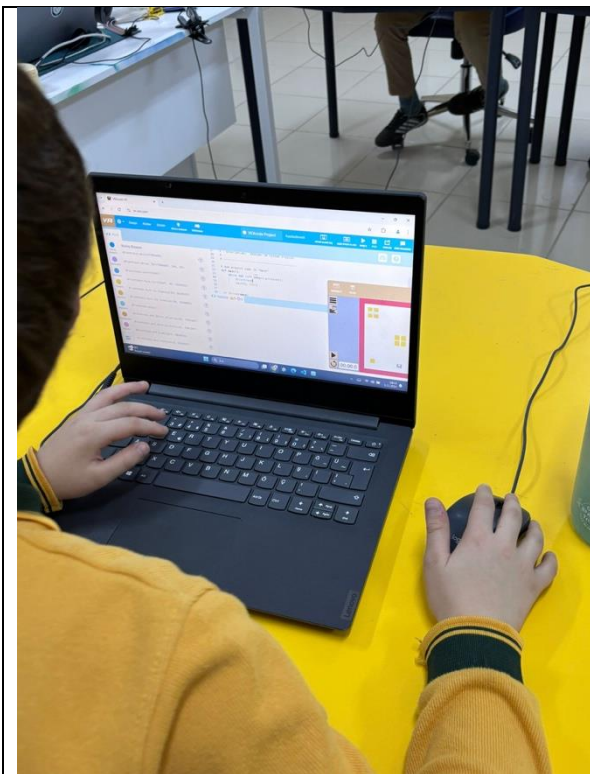
EK-9. Sanal ve Fiziksel Robotik Kodlama Gruplarının Uygulama Görüntüleri

Sanal Robotik Kodlama Grubu (VEXcode-VR)



Fiziksel Robotik Kodlama Grubu (Raspberry-Pi)





EK-10. Araştırma Uygulama İzni



Tatvan Bilim ve Sanat Merkezi Müdürlüğüne



Başvuru No: MEB.TT.2024.001373

Uygulama Yapılacak MEB Teşkilatının Kurum Kodu: 767373

T.C. Kimlik No: 30292544234

Adı Soyadı: FATMA BURCU TOPU

Araştırmacının Adı: Üstün Yetenekli Öğrencilerin Sanal ve Fiziksel Robotik Kodlama Performanslarının, Öz-Yeterliklerinin ve Kaygılarının İncelenmesi

Araştırmacının Niteliği: Yüksek Lisans Tezi

Araştırmacının Örneklem / Çalışma Grubu: Öğrenci

Uygulama Yapılacak MEB Teşkilatı: Tatvan Bilim ve Sanat Merkezi

Uygulama Yapılacak Birim: BİLSEM (Üstün veya Özel Yetenekliler)

Uygulama Yapılacak İl: BİTLİS

Veri Toplama Aracının Başlığı: Programlama Başarı Testi, Programlama Öz-Yeterlik Ölçeği, Programlama Kaygı Ölçeği, Haftalık Robotik Kodlama Kazanım Değerlendirme Kontrol Listeleri, Haftalık Robotik Kodlama Meşguliyet Gözlem Formu, Görüşme Rehberi

Araştırma Uygulama İzninin Kabul Tarihi: 11.09.2024

Araştırmacının Uygulama İzninin Bitiş Tarihi: 11.09.2025

Yukarıda kimliği yazılı araştırmacı "Araştırma Uygulama İzinleri Genelgesine (2024/41)" göre belirtilen kapsamda araştırmasını yapmayı taahhüt etmiştir. Araştırmacının bilgi ve belgelerinin uygunluğu kontrol edilmiş olup araştırma uygulama izni BİTLİS İl Millî Eğitim Müdürlüğü tarafından onaylanmıştır.

NOT: Okul/kurum yöneticileri tarafından "Araştırma Uygulama İzni" belgesinin ve veri toplama araçlarının (araçlardaki maddelerinin) modülde yer alan belge ve araçlarla aynı olduğu kontrol edilmelidir. Aynı olmadığı durumda araştırma uygulama izni verilmeyecektir.

* Başvuru detayını görüntülemek ve belgeyi doğrulamak için '<https://arastirmaizinleri.meb.gov.tr/belge-dogrula>' bağlantısını kullanınız.

- Araştırma Uygulama İzinleri Başvuru ve Değerlendirme Modülü -

EK-11. Eğitsel Robotik Kodlama Çalışmaları

Yazar (Yıl)	Amaç	Robotik Kodlama Aracı	Örneklem Düzeyi ve Sayısı	Değişkenler	Araştırma Deseni	Sonuç
Uysal (2024)	Araştırmadaki amaç; yazılım mühendisliği öğrencilerinin fiziksel programlama öğretiminin tasarlanması, ters yüz sınıf modeli yöntemiyle öğretim yapılması, bağlılık ve programlama öz-yeterlik algısı bağlamında değerlendirmektir.	Arduinio	64 lisans öğrencisi	Bağlılık ve programlama öz-yeterliği	Nicel	Arduino robotik etkinliklerin kodlamayı daha eğlenceli kılması, öğretimi somutlaştırması ve problem çözümünün becerilerine katkı sağlaması bireyin programlama kişinin öz-yeterlik algısını geliştirmede etkilidir.
Leonard vd. (2023)	Ortaokul 5.ve 6. Sınıftaki öğrencileri arasında bilişimsel düşünme (CT), bilişimsel katılım (CP) ve mekânsal akıl yürütmeyi geliştirme bağlamını incelemektedir.	Lego	22 ortaokul öğrencisi	STEAM beceri, Bilgisayarca Düşünme, Uzamsal Zekâ	Nitel	Sonuç olarak, öğrenci özerkliği, kendi kendine öğrenme ve iş birliğinin öğrencilerin bilişimsel düşünme (CT), bilişimsel katılım (CP) ve mekânsal akıl yürütme becerilerini geliştirdiği ortaya çıkmıştır.
Gürkez (2023)	Amaç robotik kodlamadaki bilişsel beceri farkındalığı ve öğrenme sorumlulukları üzerindeki etkisini incelemektir.	Abilix Krypton 7	40 ortaokul öğrencisi	Akademik Başarı, Üstbiliş Farkındalıkları, Öğrenci Görüşleri	Nicel	Verilen eğitim ile üst bilişsel düşünme becerileri ölçülmekte ve 21. yüzyıl becerileri olan algoritmik düşünme, hesaplamalı düşünme, eleştirel düşünme ve yaratıcı düşünmeye olumlu etkisi gözlemlenmiştir.
Ünal ve Topu (2022)	Çalışmada lise öğrencilerinin python programlama dili öğrenme sürecinde metin tabanlı programlama ortamı olarak python editörünü kullanan bir gurup ile hibrit tabanlı programlama ortamı olarak blockly kullanan diğer guruptaki deneyimlerin detaylı analizi yapılmıştır.	Blockly	19 lise öğrencisi	Programlama Öz-yeterliği, Öğrenci başarısı, deneyimleri ve Görüşleri	Karma	Her iki gruptaki öğrenciler de programlama dili öğreniminin önemini fark etmişlerdir. Her iki guruptaki öğrenciler de döngü yapıları, listeler ve fonksiyonları anlamakta, program hatalarını bulmada zorluk çekmişlerdir.
Kılıç (2022)	Araştırmada robotik kodlama ön lisans öğrencilerinin bilgi işlemsel düşünme becerisi gelişimine olan etkisi incelenmiştir.	Tincercad	32 önlisans öğrencisi	Bilgi işlemsel Düşünme becerileri, Öğrenci Görüşleri	Karma	Arduino etkinlikler robotik programlama dersinin ön lisans düzeyindeki öğrencilerin bilgi işlemsel düşünme becerilerini olumlu etkilemiştir. Günlük hayatlarında karşılarına çıkacak problemlere eleştirel bir yaklaşımla yaratıcı çözümler bulmuşlardır.
Qu ve Fok (2022)	Bu araştırmadaki amaç; öğrenci-robot etkileşimlerinin öğrencilerin bilişimsel düşünmeyi nasıl geliştirdiğini keşfetmektir.	Lego	32 ilkokul öğrencisi	Programlama başarısı ve görüş alma	Karma	Öğrenci Robot etkileşiminin (programlama-bilgisayar, gözlemsel araştırma ve katılımcı araştırma) öğrencilerin BT'sini geliştirmede önemli roller oynadığını gözlemlemiştir.
Sarı vd. (2022)	Çalışmada fen bilgisi öğretmen adaylarının STEM odaklı fiziksel hesaplama etkinliklerini deneyimlemeleri, algoritmik düşünme becerileri kazanmaları ve bu tür uygulamalara yönelik farkındalık yaratmaları amaçlanmaktadır.	Arduinio	24 üniversite öğrencisi	Bilgisayarca Düşünme, Öğrenci görüşleri, STEAM Becerisi	Karma	Algoritmik düşünme alt boyutları olarak belirtilen çözüm stratejilerini belirleme ve algoritma oluşturma becerilerinde artış olduğu sonucuna ulaşmıştır.

Yurdagök (2022)	Araştırmanın amacı, fiziksel kodlama aracı kullanarak metin tabanlı programlama öğretiminin ortaokul öğrencilerinin bilgi işlemsel düşünme becerilerine ve öz-yeterlik algılarına etkisini görmektir.	Micro-bit	13 ortaokul öğrencisi	Tutum ve programlama öz-yeterliği, Bilgisayarca Düşünme	Karma	Fiziksel programlama aracı ile yapılan python metin tabanlı programlama dili öğretiminde tüm öğrenciler severek ve üst düzey motivasyon ile eğitim sürecini geçirdikleri söylenebilir; odak gurup görüşmelerinde dersin devamını istemeleri bu yönde olmuştur.
Koca ve Çakır (2022)	LEGO tabanlı eğitim uygulamalarının öğrencilerin akademik motivasyonuna, problem çözmeye yönelik yansıtıcı düşünme becerisine, uzamsal görselleştirmeye ve zihinsel döndürmeye etkisini karşılaştırmalı olarak belirlemektir.	ScratchŞLeg o	20 ortaokul öğrencisi	Problem Çözme, Yansıtıcı Düşünme, Motivasyon, Uzamsal Düşünme	Karma	Bu çalışma sonucunda akademik motivasyonda azalma görülmüştür. Ayrıca bu çalışmanın sonuçları, bu tür robotik ve kodlama etkinliklerinin bu teknolojik çağda ortaöğretim programlarına entegre edilmesi gerektiğine dair kanıtlar sunmuştur.
Yüksel (2022)	Ortaokul öğrencilerini Çoklu zeka alanları ve blok tabanlı programlamaya yönelik Öz-yeterlik algılarının incelenmesi	Scratch	45 ortaokul öğrencisi	Öz-yeterlik algıları	Nicel	İlişkili zeka alanına sahip öğrenciler programlama konusunda daha başarılı olmuşlardır.
Hamiti vd. (2021)	Araştırmanın amacı;bilişim teknolojileri öğretmenliğinde son sınıfta okuyan öğrencilerin robotik programlama dersin hakkındaki düşüncelerini belirlemektir.	Kodlama İle İlgili Görüşler	42 üniversite öğrencisi	Katılımcı Görüşleri	Nitel	Öğretmen adaylarının eğitim ve teknolojiye ilişkin sonuçları dikkate alındığında görüşleri oldukça olumludur. Öğretmen adayları teknoloji ve eğitim kavramını “çağlara ayak uydurmak” olarak tanımlamışlardır.
Güngördü (2021)	Bilişim teknolojileri öğretmenlerinin uzaktan eğitim kullandıktan sonra genel öz-yeterlik ve kodlama etkinliklerindeki öz-yeterlik algıları arasında bir ilişki olup olmadığını incelemektir.	Blok Tabanlı Kodlama	58 BT öğretmeni	Öz-yeterlik	Nicel Betimsel Tarama	Öz-yeterlik algıları yaş, cinsiyet ve sınıf düzeyi gibi değişkenlere bağlı olarak değişkenlik göstermiştir.
Yang vd. (2021)	Çalışma, bir blok oyun programına karşı bir robot programlama müdahalesinin anaokulu öğrencilerinin BT, sıralama yeteneği ve öz düzenleme üzerindeki etkilerini araştırmıştır.	Robotik Oyuncak	101 anasınıfı öğrencisi	Programlamaya karşı tutum, Bilgisayarca Düşünme, Başarı	Karma	Blok kodlama ile robotik etkinliklerinden sonra öğrencilerin algoritma oluşturma ve öz-yeterlik seviyelerinde gelişmeler olduğu görülmüştür.
Gürkez (2021)	Bireylerin robotik kodlama eğitiminin, akademik başarı, üst biliş beceri farkındalığı ve öğrenmeye yönelik sorumlulukları üzerine etkisinin Abilix Krypton 7 aracıyla incelenmesi amaçlanmıştır.	Abilix Krypton 7	132 ilköğretim öğrencisi	Akademik başarı, STEAM Beceri, Tutum	Karma	Açık uçlu sorulara verilen cevaplara göre öğrencilerin çoğunluğunun robotik kodlama eğitiminde kendini geliştirmek istediği ve ilgi odağı olduğu için bu eğitimi aldıkları tespit edilmiştir.
Yıldız ve Seferoğlu (2021)	Araştırmanın amacı Lego Mindstorms EV3 robotik kit ile verilen programlama eğitiminin bireylerin programlamaya karşı tutumlarını ve bilişimsel düşünme becerileri öz-yeterlik algılarına etkisini araştırmaktır.	Lego	30 ortaokul öğrencisi	Programlama Öz-yeterliği, Tutum, Bilgisayarca Düşünme	Nicel	Araştırma bulgularında bireylerin hem kodlamaya karşı tutumları, bilişimsel düşünme becerisine yönelik öz-yeterlik algılarında olumlu yönde bir değişim olduğu tespit edilmiştir.
Çınar ve Tüzün (2020)	Nesne yönelimli ve robot programlama etkinliklerinin etkilerini karşılaştırmaktadır.	Lego, Arduino	81 lise öğrencisi	Başarı, problem çözme becerisi, Programlama Öz-yeterliği	Karma	Öğrencilerin programlamaları arasında orta düzeyde pozitif ve anlamlı ilişkiler bulunmuştur. Öğrencilerin problem çözme becerilerinde ve motivasyonlarında artış görülmüştür.

Zhong ve Li (2020)	Çalışmada: öğrencilerin karşılaştığı sorun giderme sorunlarının sayısı, sorun kategorisi, öğrencilerin sorun giderme sorunlarını çözüp çözmedikleri araştırılmıştır	Arduinio	38 lise öğrencisi	Programlama başarısı, Tutum Programlama Öz-yeterliliği	Nicel	Öğrencilerin robot programlama yapmayı öğrenmediği, ancak bu süreçte sorun giderme becerilerini geliştirmek için daha etkili ikili öğrenme stratejilerinin geliştirilmesi gerektiği belirlenmiştir.
Zhong vd. (2020)	Çalışmada robotik programlama dersinde farklı öğrenme modellerinin sanal ve fiziksel robotların (VPR) kullanılmasının etkileri incelenmiştir.	Sanal ve Fiziksel Robot	84 ilkokul öğrencisi	Programlama başarısı, Programlama Öz-yeterliliği, Problem Çözme	Karma	Öğrencilerin ön test ve son test yüksek düzeyde öğrenme katılımı olduğu, sanal ortamda ile fiziksel ortamda yapılan programlamadaki bilişsel yüklerinden anlamlı düzeyde farklı olduğu görülmüştür.
Kert vd. (2020)	Bu çalışma, eğitim ortaokulda programlamayı eğitim robotları yoluyla öğrenen bir öğrenci grubu ile blok tabanlı programlama ortamı aracılığıyla programlamayı öğrenen başka bir öğrenci grubu arasında başarı, programlama öz-yeterliliği, problem çözme, bilgisayarca düşünme açısından anlamlı bir fark olup olmadığını araştırmaktadır.	Scratch, LEGO	56 ortaokul öğrencisi	Başarı, Programlama Öz-yeterliliği, Problem Çözme, Bilgisayarca Düşünme	Karma	Hem blok tabanlı programlamanın hem de eğitim robotlarının bilgi işlemsel düşünme becerilerini geliştirmede etkili yollar olduğu, her iki grupta programlamadaki kavramsal gelişimin olduğu görülmüştür.
Voštinár ve Knežník (2020)	Araştırmının temel amacı, BBC micro:bit'i kullanarak öğrencileri programlama eğitimi konusunda motive etmektir.	Micro-bit	66 ortaokul öğrencisi	Tutum, Motivasyon	Nitel	Bu çalışma, BBC Micro:bit kullanımının öğrencilerin programlama becerilerini ve algoritmik düşüncelerini geliştirmek için mükemmel cihaz olduğunu, öğrencilerin bu dersleri sevdiklerini göstermiştir.
Yıldırım (2020)	Probleme dayalı öğrenme modeline göre hazırlanan robotik programlama öğretiminin bireylerin programlama becerisi başarıları, bilgi işlemsel düşünme öz-yeterlik algıları ve pozitif duygularına etkisi araştırılacaktır.	Mbot	178 ortaokul öğrencisi	Bilgisayarca Düşünme Başarı Programlama Öz-yeterlik algısı	Karma	Kodlama becerisi/başarı puanlarında artış görülmüştür. Probleme dayalı programlama eğitimi deney gurubunda öz-yeterlik becerisinin geliştiğini göstermiştir.
Gündoğdu (2020)	Araştırmada mesleki lisesinde öğrenim gören bireylerin lego robotik araçların kullanıldığı ve kullanılmadığı guruplara ayırarak algoritma öğretiminin öğrencilerin, bilgisayarca düşünme becerilerine, bilişsel yüklerine ve algoritma başarılarına etkisi incelenmiştir.	Lego WeDo	62 lise öğrencisi	Akademik başarı, Bilgisayarca Düşünme, Katılımcı Görüşleri	Karma	Lego robotikler ile eğitim verilen deney gurubunda değişkenlerin daha anlamlı bir farklılık gösterdiği tespit edilmiştir.
Balon ve Simić (2019)	Raspberry-Pi bilgisayarlarının lise ve yüksek öğrenimde kullanılmasına ilişkin düşünceler aratılmaktadır.	Raspberry-Pi	90 üniversite öğrencisi	STEAM becerisi, Öğrenci Görüşleri	Nitel	STEAM etkinliklerinin öğrencilere sorunları başarılı bir şekilde çözmek için örnekler yoluyla eğitim vermeye ve farklı disiplinleri birleştirmeye teşvik ettiği belirlenmiştir.
Ergin (2019)	Araştırmının amacı programlama öğretiminde proje tabanlı öğrenme yönteminin kullanılmasının, bireylerin programlama becerisine yönelik öz-yeterlik algılarının ve bilgi işlemsel düşünme becerilerine etkilerini incelemektir.	Nesne Tabanlı Programlama (NTP)	14 lise öğrencisi	Öz-yeterlik inançları ve bilgi işlemsel düşünme becerileri	Nicel Deneyssel	Araştırma sonucunda, proje tabanlı öğrenme yöntemi öncesi ve sonrası öğrencilerin programlamaya ilişkin öz-yeterlik algılarında pozitif bir artış olduğu görülmüştür. Fakat bireylerin bilgi işlemsel düşünme becerileri arasındaki farkın değişmediği ortaya çıkmıştır.
Eraytaç (2019)	Bilişim Teknolojileri dersinde robotik kodlama	Arduinio	130 ortaokul	Başarı	Nicel	Deney ve Kontrol guruplarının akademik başarı düzeyleri arttığı görülmüştür.

	eğitiminin blok tabanlı kodlama yöntemi ile verilmesinin akademik başarıya etkisini araştırmaktadır.		öğrencisi			Blok tabanlı (mBlock) eğitim alan öğrencilerin metin tabanlı (Arduino) eğitim alan öğrencilere göre son test puanları daha yüksek çıkmıştır.
Akbıyık (2019)	Arduino uygulamalarının öğrencilerin programlama eğitimindeki öz-yeterlikleri ve problem çözme becerileri üzerine etkisi incelenmiştir.	Arduino	30 lise öğrencisi	Öz-yeterlik ve Problem Çözme becerisi	Nicel	Araştırmada; bireylerin öz-yeterlikleri ve problem çözme becerileri arasında anlamlı bir fark bulunmuştur. Görüşmeler sonucunda bireylerin etkinliklerle ilgili olumlu görüşler bildirmişlerdir.
Kök (2019)	Araştırmanın amacı robot kitler kullanılarak ortaokul öğrencilerinin gruplara ayrılarak robotik öğrenimini incelemektir.	Lego	39 ortaokul öğrencisi	Programlama Öz-yeterlik algısı, Bilgisayarca Düşünme, Üst biliş Farkındalıkları Tutum	Nitel	Öğrencilerin derste daha fazla ilgi duydukları keyifli saatler geçirdikleri, merak duygularının uyandığı, mutlu oldukları, derse bir an önce başlamak istedikleri ve dersin hiç bitmemesini istedikleri görülmüştür. Lego ile robotik kodlamanın öğrencilerde içsel motivasyonlarını artırdığı tespit edilmiştir. Bunun yanında iş birliği halinde çalışarak çözümler üretmelerine, yaratıcılık ve problem çözme becerilerini desteklediği tespit edilmiştir.
Konyaoğlu (2019)	Robotik kodlama eğitimi alan bireylerin problem çözme becerilerini incelemek, robotik programlama ile ilgili görüşlerini ve memnuniyetlerini belirlemektir.	Mbot	26 ortaokul öğrencisi	Programlama Öz-yeterliği, Problem Çözme, Tutum	Karma	Öğrencilerin problem çözme becerilerinde pozitif yönde bir değişim olduğu görülmüştür. Öğrenciler, robotik kodlama eğitimi süresince kendi problemlerini çözmeye gayret göstermişlerdir.
Uğuz (2019)	Araştırmada lego robotikle ile programlama öğrenme sürecinde bireysel ve işbirlikli eğitimlerin bireylerin problem çözme becerilerine ve programlama başarılarına etkisi araştırılmıştır.	Lego WeDo	55 ortaokul öğrencisi	Poblemler çözme ve akademik başarı	Nicel	Bireysel ve işbirlikli olarak lego robotik programlama etkinliklerine katılan öğrencilerin tümünün uygulama sonunda problem çözme becerilerinde artış olduğu görülmüş, bireysel çalışan bireylerde problem çözme becerisinin işbirlikli gruptan daha fazla olduğu ortaya çıkmıştır.
Güleryüz (2019)	Ortaokul öğrencilerin blok tabanlı kodlamaya ilişkin öz-yeterlik becerileri arasında bir ilişki olup olmadığı incelenmiştir.	Arduino	122 ortaokul öğrencisi	Akademik başarı, Tutum, Programlama Öz-yeterliği	Nicel	Öğrencilerin blok temelli programlamaya ilişkin öz-yeterlik algısında pozitif yönde bir artış olduğu tespit edilmiştir.
Dinçer (2018)	Araştırmanın amacı Scratch ve Kodu Game Lab ile programlama öğretiminin 6. Sınıfta okuyan bireylerde akademik başarıları, tutumu ve öz-yeterlik algıları üzerine olan etkisini incelemektir.	Scratch, Kodu Game Lab	27 ortaokul öğrencisi	Tutum Öz-yeterlik ve akademik başarı	Karma	Scratch ve Kodu Game Lab ile programlama eğitimi alan öğrencilerin programlama dillerine karşı olan öz-yeterlik algılarının anlamlı derecede artış olduğu görülmüştür. Ayrıca Kodu Game Lab ile programlama eğitiminin tutumu daha çok artırdığı tespit edilmiştir.
Numanoğlu ve Keser (2017)	Programlama öğretiminde Mbot robot kitinin değerlendirilmesi	Mblock	55 ortaokul öğrencisi	Problem çözme becerisi, Programlama Öz-yeterliği, STEAM Beceri	Nitel	Mbot sisteminin düşük maliyetli kolay kurulumlu ve basit programlanabilmesinin, robotik eğitiminde kullanılabilirliğini artırdığı görülmüştür. Yapılandırmacı model ve temel öğretim stratejileri olan senaryo temelli, problem-temelli ya da proje-temelli öğrenme modelleri uygun bir materyal olarak tanımlanmıştır.
Çankaya vd. (2017)	Araştırmada robotlarla programla eğitimi alan bireylerin akademik başarılarının ve görüşlerinin belirlenmesi amaçlanmıştır.	Lego	9 ortaokul öğrencisi	Programlama Başarı ve Öğrenci Görüşleri	Karma	Bireylerin yaratıcı problem çözme becerileri ile verilen eğitim sonunda uygulanan performans değerlendirme puanları arasında orta düzeyde bir ilişki olduğu görülmüştür, bu nedenle yaratıcı problem çözme becerisi yüksek olan öğrencilerin programlama eğitiminde daha başarılı olacakları düşünülebilir.

Kasalak (2017)	Robotik kodlama eğitimi verilen ortaokul öğrencilerinin kodlama becerisine ilişkin öz-yeterlik algılarına etkisini ve etkinliklere ilişkin öğrenci deneyimleri araştırılmaktadır.	Scratch	329 ortaokul öğrencisi	Öz-yeterlik algısı ve öğrenci yaşantıları	Karma	Eğitim sonunda bireylerde hem basit kodlama hem de karmaşık kodlarda programlama eğitime yönelik öz-yeterlik algılarında pozitif yönde bir artış olduğu görülmüştür. Öğrencilerin eğitimi ilgi çekici bulunduğu tespit edilmiştir.
Okuyucu (2019)	Robotik kodlama eğitim etkinliklerinin, öğrencilerin yansıtıcı düşünme ve üstbiliş farkındalıklarına etkisini incelemek amacıyla yapılmıştır.	Ardunio	119 ortaokul öğrencisi	Yansıtıcı Düşünme Becerileri, Üst biliş Farkındalıkları	Nicel	Eğitimin sonunda lise öğrencilerinin üstbiliş farkındalıklarının yansıtıcı düşünme becerilerinde gelişimin olduğu söylenebilir.
Atmatzidou ve Demetriadis (2015)	Eğitsel robotik (ER) öğrenme etkinliği bağlamında öğrencilerin bilişimsel düşünme (BT) becerilerinin gelişimini araştırmaktadır	Lego	164 ilkokul öğrencisi	Bilgi İşlemsel Düşünme becerileri	Karma	Öğrenciler eğitim sonunda bilgi işlemsel düşünme becerisine aynı düzeyde ulaşmışlar ve becerileri önemli ölçüde artmıştır. Kızlar erkeklere nazaran daha fazla gayret sarf etmiştir.
Yurdagök ve Kalelioğlu (2024)	Fiziksel programlama aracıyla metin tabanlı programlama öğretiminin ortaokul öğrencilerinin hesaplamalı düşünme becerileri ve öz-yeterlik algıları üzerindeki etkisi incelenmiştir	micro:Bit	85 ortaokul öğrencisi	Hesaplamalı düşünme becerisine yönelik öz-yeterlik algısı	Karma	Öğrencilerin hesaplamalı düşünme becerisine yönelik öz-yeterlik algılarında anlamlı ve pozitif yönde bir fark bulunmuştur. Programlama eğitimi almanın bir sonucu olarak, öğrenciler gerekli programlama kavramlarını ve süreçlerini tatmin edici bir şekilde öğrenlerdir. Fiziksel bir programlama aracıyla Python programlamayı öğrenerek, öğrenciler yalnızca yazmak için gereken becerileri kazanmakla kalmamış; metinde incelenen değişkenler, kullanılan programlama dili ve hangi robotik aracın kullanıldığına yönelik bilgi de edinmişlerdir.
Yang (2024)	Kodlama robotları kullanan program Kodlama uygulamaları kullanan programın okul öncesi çocukların Hesaplamalı Düşünme (BT) ve Sosyal-Duygusal Yeterlik (BDT) gelişimlerini destekleme deki etkililiğini karşılaştırmaktır.	Robotik oyuncak	73 okul öncesi öğrencisi	Hesaplamalı Düşünme (BT) becerisi Sosyal-Duygusal Yeterlik (BDT)	Deneysel	Yaş, cinsiyet, aile, sosyo ekonomik durumu ve başlangıç BDT puanları kontrol edildikten sonra, Kodlama Robot Programına katılan çocukların Kodlama Uygulaması Programına katılanlardan daha yüksek BDT puanlarına sahip olduğu görülmüştür.
Özel (2023)	Araştırmada, Scratch Jr blok tabanlı programlama dili ile çalışan okul öncesi çocukların kodlama sürecindeki davranışlarının (ilgi, merak, yönerge takibi, keşfetme vb.) nitel olarak incelenmesi amaçlanmıştır.	Scratch	52 okul öncesi öğrencisi	Tutum	Nitel	Çoğu çocuk des süresini tam olarak kullanmış, hatta bazı çocuklar süreyi uzatmak istemiştir. Çoğunluk uygulamanın ayrıntılarını keşfederek ve gerçek hayatla ilişkilendirerek, ilk talimatlardan sürecin sonuna kadar merak duygusunu kaybetmeden süreci amaçlarına uygun bir şekilde verimli bir şekilde tamamlamıştır.
Selcuk vd. (2024)	Robot temelli programlama etkinliklerinin erken çocukluk dönemindeki çocukların Hesaplamalı düşünme (computational thinking) becerileri, Sosyal becerileri ve Öğrenme motivasyonu üzerindeki etkilerini araştırmaktır	Ardunio	112 ortaokul öğrencisi	Hesaplamalı düşünme Sosyal beceriler Öğrenme motivasyonu	Karma	Bulgular, öğrencilerin motivasyon düzeylerinin uygulama öncesi ve sonrası orta düzeyde olduğunu, ancak uygulama sonrasında ders motivasyon puanlarında göreceli bir düşüş olduğunu göstermiştir. Öğrenciler robotiğe karşı olumlu tutumlar sergilemiş ve iyi bir başarı düzeyine ulaşmışlardır. Ayrıca, motivasyonun robotiğe yönelik tutumlar üzerinde önemli bir etkisi olduğu gözlemlenmiştir.
Erol vd. (2024)	Bu çalışmanın amacı, ortaokul öğrencilerinin robotik kodlamaya yönelik tutumları ile STEM kariyer ilgileri arasındaki ilişkiyi incelemektir. Ayrıca öğrencilerin bu tutum ve ilgilerinin cinsiyet, robotik eğitimi alma durumu ve ebeveyn meslekleri gibi değişkenlere göre farklılaşıp farklılaşmadığı da araştırılmıştır	Robotik Oyuncak	213 ortaokul öğrencisi	Tutum	Karma	İster okul ortamında ister okul dışında yürütülsün, robotik kodlamaya yönelik tutumları geliştiren ve ilgiyi artıran tüm etkinliklerin, öğrencilerin gelecekte STEM ile ilgili bir kariyer arayışını kolaylaştıracağı belirlenmiştir.

Fernando ve Gonzalez (2025)	Bu çalışmanın amacı, robotik eğitiminde fiziksel ve sanal robotik kolların birlikte kullanıldığı hibrit bir öğrenme ortamı geliştirerek öğrencilerin motivasyonunu artırmak ve robot programlama becerilerini geliştirmektir.	Arduinio	35 üniversite öğrencisi	Motivasyonu performans Deneyimi memnuniyeti	Deneysel	Hibrit yaklaşımın kullanıldığı Robotik Giriş dersinde ilgili iki ödevin tamamlanma oranı yükselmiştir. Fiziksel kolun kullanımına ilişkin bir anket, öğrencilerin genel olarak ödevleri tamamlama konusunda motive olmalarına ve öğrenme deneyimlerini zenginleştirmelerine yardımcı olduğu konusunda hemfikir olduklarını göstermiştir.
Zhong vd. (2023)	Bu çalışmanın amacı, sanal (VR) ve fiziksel (PR) robotların birlikte (VPR) kullanıldığı hibrit bir öğretim stratejisinin ortaokul düzeyindeki öğrencilerin: öğrenme tutumları, programlama becerileri, mühendislik tasarım yeteneği, öğrenme katılımı ve bilişsel yükü üzerindeki etkilerini incelemektir	Sanal, Arduinio	84 lise öğrencisi	Öğrenme tutumları Programlama becerileri, Mühendislik tasarım yeteneği, Öğrenme katılımı Bilişsel yük	Deneysel	Öğrenme tutumu, programlama becerisi ve katılım açısından anlamlı fark bulunmamıştır. VPR grubu, mühendislik tasarım yeteneği bakımından anlamlı biçimde daha başarılı olmuş, ayrıca bilişsel yükleri daha düşük rapor edilmiştir. Bu bulgular, sanal ve fiziksel robotların birleşiminin özellikle karmaşık problem çözme ve üst düzey bilişsel beceriler açısından daha etkili olduğunu göstermiştir.
Alsoliman (2022)	Bu çalışmanın amacı, fiziksel robotların yerine sanal robotların kullanıldığı çevrim içi (virtual) öğrenme ortamında sekizinci sınıf öğrencilerinin STEM (Fen, Teknoloji, Mühendislik ve Matematik) alanlarındaki öğrenme deneyimlerini incelemektir.	Virtual Robotics Toolkit (VRT), LEGO Mindstorms	25 ortaokul öğrencisi	Öğrenme deneyimi Motivasyonu, Etkileşim biçimi, Memnuniyet düzeyi		Sonuç olarak, çalışma sanal robotiklerin fiziksel robotlara alternatif bir öğretim aracı olarak STEM öğrenmesinde pedagojik değer taşıdığı ve gelecekte hibrit öğretim ortamlarında önemli bir rol oynayabileceğini vurgulanmıştır.
Balaban vd. (2021)	Bu araştırmanın amacı, ortaöğretim öğrencilerinin açık kaynak kodlu yazılımlardan Scratch, Arduino ve Python'a yönelik görüşlerini incelemektir.	Scratch Arduino Python	9 lise öğrencisi	Öğrenci Görüşleri öğrenici görüşleri		Öğrenciler açık kaynak yazılımları özgürlük, erişilebilirlik ve öğrenme kolaylığı açısından avantajlı bulmuştur. Scratch en eğlenceli ve öğrenmesi kolay araç olarak görülmüştür. Python, “profesyonel ve kariyer için gerekli” bir dil olarak algılanmıştır. Arduino, “gerçek ürünler üretmeye imkân veren” uygulamalı bir araç olarak değerlendirilmiştir. Tüm yazılımlar, yaratıcılığı, problem çözme becerisini ve planlı düşünmeyi geliştirdiği gerekçesiyle olumlu bulunmuştur.
Çayır (2010)	Bu çalışmada lego robotik setler ile desteklenmiş öğrenme ortamının ilköğretim 8. sınıf öğrencilerinin bilimsel süreç becerisi, benlik algısı üzerindeki etkileri incelenmiştir.	Lego Mindstorm NXT	40 ortaokul öğrencisi	Bilimsel süreç becerisi, Benlik algısı		Araştırma sonuçlarına göre lego - logo ile desteklenmiş öğrenme ortamının öğrencilerin benlik algısı üzerinde olumlu etkiler oluşturduğunu göstermiştir.
Küçük ve Sönmez (2023)	Bu çalışmada kodlama eğitiminde eğitsel robot kullanımının, özel yetenekli öğrencilerin bilgi işlemsel düşünme becerisine etkisini incelemek amaçlanmıştır.	Mbot	110 ortaokul öğrencisi	Bilgi İşlemsel Düşünme Becerisi		Robotik kodlama eğitimi alan öğrencilerin bilgi işlemsel düşünme becerilerinde artış olduğu gözlemlenmiştir.
Balcı ve Kalkan (2024)	Bu araştırmanın amacı: Robotik kodlama dersinde kullanılan blok tabanlı kodlama etkinliklerinin, metin tabanlı kodlama etkinliklerine etkisi olup olmadığını incelemektir.	Code Lab Python	8 ortaokul öğrencisi	Programlama Başarısı		Bulguların sonucunda blok tabanlı kodlama eğitimi alan öğrencilerin metin tabanlı kodlama daha iyi kavradıkları ve öğrendikleri, blok tabanlı kodlamayı almayan öğrencilerin, metin tabanlı kodlama yaparken zorlandıkları sonucuna varılmıştır.

Doğan ve Korkmaz (2025)	Bu araştırmanın amacı, proje tabanlı eğitsel robot etkinliklerinin ortaokul öğrencilerinin bilgisayarca düşünme ve programlamaya dönük öz yeterliklerine etkisini belirlemektir.	Lego	24 ortaokul öğrencisi	Bilgisayarca düşünme, Programlama öz-yeterliği	Analizler sonucunda proje tabanlı eğitsel robot etkinliklerinin ortaokul öğrencilerinin bilgisayarca düşünme becerisine olumlu etkilerinin olduğunu ve programlamaya dönük özyeterlilik anlamı düzeyde katkısının olduğu sonucuna varılmıştır.
Tselegkaridis ve Sapounidis (2021)	Bu çalışmanın amacı, ilkokul düzeyinde programlama öğretiminde somut cihazlar (tangible devices) ve eğitim robotlarının etkisini incelemektir.	TangibleK	60 ilkokul öğrencisi	Programlama başarı Programlamaya yönelik ilgi düzeyi, Öz yeterlik algısı, Problem çözme becerileri, Tutum, Motivasyon	TangibleK kullanan öğrenciler, kontrol grubuna göre anlamlı düzeyde daha yüksek başarı, daha olumlu tutum ve daha fazla ilgi göstermiştir. Somut, dokunsal ve oyun temelli programlama ortamı, küçük yaş gruplarının bilişsel gelişimine uygun bulunmuştur.

ÖZ GEÇMİŞ

29.09.1984 yılında Şırnak İdil İlçesinde doğumdum. ilk ve ortaokul öğrenimini Bitlis Tatvan ilçesinde tamamladım. Lise öğrenimini Bitlis Ahlat İlçesinde Endüstri Meslek Lisesi Elektrik Bölümünü okudum. Lisans öğrenimini Gazi Üniversitesi Endüstriyel Sanatlar Eğitim Fakültesi Endüstriyel Teknoloji Öğretmenliği Bölümünde okudum. Bitlis Tatvan ilçesi Dumlupınar Ortaokuluna Teknoloji Tasarım Öğretmeni olarak atandım ve göreve başladım. 2021 yılından itibaren Tatvan Bilim ve Sanat Merkezinde Teknoloji ve Tasarım öğretmeni olarak görev yapmaktayım. Evli ve iki çocuk babasıyım.

