

T.C.
KASTAMONU ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
EĞİTİM BİLİMLERİ ANA BİLİM DALI
EĞİTİM PROGRAMLARI VE ÖĞRETİM



YÜKSEK LİSANS TEZİ

**İŞBİRLİKLİ ÖĞRENME YAKLAŞIMI İLE PROGRAMLAMA
ÖĞRETİMİNİN ALTINCI SINIF ÖĞRENCİLERİNİN
PROGRAMLAMA ÖZ-YETERLİKLERİNE, ÖZ-DÜZENLEME
BECERİLERİNE VE AKADEMİK BAŞARILARINA ETKİSİ**

MEHMET DERVİŞ

Danışman : Doç. Dr. Mustafa Öztürk AKCAOĞLU

Jüri Üyesi : Doç. Dr. İlkay AŞKIN TEKKOL

Jüri Üyesi : Doç. Dr. Emrullah YILMAZ

KASTAMONU-2025

TAAHHÜTNAME

Bu tezin tasarımı, hazırlanması, yürütülmesi, araştırmalarının yapılması ve bulgularının analizlerinde bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu; ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını, bilimsel etiğe uygun olarak kaynak gösterildiğini bildirir ve taahhüt ederim.

Mehmet DERVİŞ

ÖZET

YÜKSEK LİSANS TEZİ

İŞBİRLİKLİ ÖĞRENME YAKLAŞIMI İLE PROGRAMLAMA ÖĞRETİMİNİN ALTINCI SINIF ÖĞRENCİLERİNİN PROGRAMLAMA ÖZ-YETERLİKLERİNE, ÖZ-DÜZENLEME BECERİLERİNE VE AKADEMİK BAŞARILARINA ETKİSİ

MEHMET DERVİŞ

KASTAMONU ÜNİVERSİTESİ SOSYAL BİLİMLER ENSTİTÜSÜ

EĞİTİM BİLİMLERİ ANA BİLİM DALI

EĞİTİM PROGRAMLARI VE ÖĞRETİM

DANIŞMAN: DOÇ. DR. MUSTAFA ÖZTÜRK AKCAOĞLU

Programlama, 21. yüzyılda öğrencilerin kazanması gereken en önemli becerilerden biri haline gelmiştir. Programlama öğrencilere beceri olarak kazandırmak amacıyla eğitim programlarına hızla entegre edilmeye başlanmıştır. Bu nedenle özellikle son 10 yılda programlama öğretim yaklaşımları üzerine çalışmalar yapılmaktadır. Bu çalışma alanlarından biri olarak işbirlikli öğrenme yaklaşımı ile programlama öğretimi, öğrencilerin ekip çalışmaları ile programlama becerileri kazanması üzerine odaklanmıştır. Bu çalışmanın amacı, işbirlikli öğrenme teknikleriyle kurgulanmış programlama öğretiminin ortaokul öğrencilerinin akademik başarıları, programlama öz-yeterlikleri ve öz-düzenleme becerileri üzerindeki etkisini ortaya çıkarmaktır. Çalışma grubunu 2024-2025 eğitim yılında Kastamonu’da bulunan bir devlet okulunda 6.sınıf düzeyinde eğitim gören 43 ortaokul öğrencisi oluşturmuştur. Araştırmada ön-test ve son-test deney ve kontrol gruplu yarı deneysel desen kullanılmıştır. Bilişim Teknolojileri ve Yazılım dersinde problem çözme ve programlama konuları sekiz hafta boyunca, haftada iki saat deney grubuna işbirlikli öğrenme teknikleri ile kontrol grubuna ise öğretim programında yer alan yöntem ve teknikler ile öğretilmiştir. Katılımcılardan veri toplamak amacıyla akademik başarı testi, algılanan öz-düzenleme ölçeği ve programlama öz-yeterliği ölçeği kullanılmıştır. Toplanan veriler SPSS programı aracılığı ile Bağımlı Örneklem T-testi, Bağımsız Örneklem T-testi, Mann Whitney-U testi, Wilcoxon İşaretli Sıralar testi yapılarak analiz edilmiştir. Araştırma bulguları, işbirlikli öğrenme teknikleriyle programlama öğretimi yapılan deney grubu öğrencilerinin programlama öz-yeterliği ve öz-düzenleme beceri düzeyleri kontrol grubu öğrencilerine göre daha yüksek olduğunu ve deney grubu lehine anlamlı farklılık olduğunu ortaya koymuştur. Akademik başarı testi bulgularında ise son-test puanlarında deney grubunun puan ortalamasının kontrol grubuna göre yüksek olmasına karşın deney grubu lehine istatistiksel olarak anlamlı fark görülmemiştir. İşbirlikli öğrenme gruplarının heterojen yapısı sayesinde öğrenme gruplardaki tüm öğrencilerin etkileşim kurarak, yardımlaşarak ve birbirlerinin deneyimlerinden yararlanarak programlama görevlerini başarıyla yerine getirdikleri görülmüştür. Bu durum, öğrencilerin programlama öz-yeterlik düzeyleri ve öz-düzenleme becerileri düzeylerini etkilemiştir. Bu çalışmanın sonuçları doğrultusunda programlama öğretimi ortaokul Bilişim Teknolojileri ve Yazılım dersi öğretim programlarına, lisans düzeyindeki Bilişim Teknolojileri öğretmeni yetiştiren öğretim

programlarına ve okullarda görev yapan öğretmenlere yönelik hizmet içi eğitim programlarına yönelik önerilere yer verilmiştir.

ANAHTAR KELİMELEER: İşbirlikli Öğrenme, Programlama, Programlama Öz-yeterliğı, Öz-düzenleme

Nisan 2025, 129 Sayfa



ABSTRACT

MSC THESIS

THE EFFECT OF COLLABORATIVE LEARNING APPROACH ON 6TH GRADE STUDENTS' PROGRAMMING SELF-EFFICACY, SELF- REGULATION SKILLS, AND ACADEMIC ACHIEVEMENT IN PROGRAMMING EDUCATION

MEHMET DERVİŞ

KASTAMONU UNIVERSITY INSTITUTE OF SOCIAL SCIENCE

DEPARTMENT OF EDUCATIONAL SCIENCES

EDUCATION PROGRAMS AND TEACHING

SUPERVISOR: ASSOC. PROF. DR. MUSTAFA ÖZTÜRK AKCAOĞLU

Programming has become one of the most important skills that students should acquire in the 21st century. Countries have rapidly integrated programming into their curricula to equip students with programming skills. To this end, studies on teaching approaches regarding programming have been conducted, especially in the last decade. One of these areas of study is the collaborative learning approach in programming education, which focuses on students acquiring programming skills through teamwork. The purpose of this study is to investigate the effect of collaborative learning techniques on middle school students' academic achievement, programming self-efficacy, and self-regulation skills. The study group consisted of 43 middle school students in the 6th grade level at a public school in Kastamonu during the 2024-2025 academic year. A quasi-experimental design with pre-test and post-test control groups was used in the study. During the Information Technologies and Software course, problem-solving and programming topics were taught to the experimental group using collaborative learning techniques for eight weeks, two hours a week, while the control group was taught using the methods and techniques in the curriculum. An academic achievement test, perceived self-regulation scale, and programming self-efficacy scale were used to collect data from the participants. The collected data were analyzed using SPSS software through the Dependent Samples T-test, Independent Samples T-test, Mann Whitney-U test, and Wilcoxon Signed Ranks test. The research findings revealed that the students in the experimental group, who were taught programming using collaborative learning techniques, had higher levels of programming self-efficacy and self-regulation skills compared to the students in the control group, and there was a significant difference in favor of the experimental group. However, in the academic achievement test findings, although the experimental group's mean score was higher than the control group's, no statistically significant difference was found in favor of the experimental group. It was observed that the heterogeneous structure of the collaborative learning groups enabled all students in the learning groups to interact, help each other, and benefit from each other's experiences, successfully completing their programming tasks. This situation affected the students' programming self-efficacy levels and self-regulation skills levels. Based on the results of this study, recommendations were made for teaching programming in middle school Information Technologies and Software courses, teacher

training programs for Information Technologies teachers, and in-service training programs for teachers.

KEYWORDS: Collaborative Learning, Programming, Programming Self-Efficacy, Self-Regulation

April 2025, 129 Page



TEŞEKKÜR

Yüksek lisans tezimi yazdığım süreç boyunca elinden gelenin fazlasıyla bana destek olan, yol gösteren, bilgi ve kültürel olarak gelişmemi sağlayan Değerli Hocam, Sayın Doç. Dr. Mustafa Öztürk AKCAOĞLU'na bu çalışmadaki emekleri için teşekkür ederim.

Eğitim Programları ve Öğretim Anabilim Dalında öğrenim gördüğüm süre içinde bilgi ve kültürel olarak gelişmemi sağlayan ve öğrenim sürecinde bana yol gösteren tüm hocalarıma emekleri için ayrı ayrı teşekkür ederim.

Bu araştırmaya katılarak sınıf içi ve sınıf dışı öğrenme etkinliklerinde üstlerine düşen görevleri fazlasıyla yerine getiren ve özverili çalışan tüm öğrencilerime teşekkür ediyorum, tüm hayatları boyunca başarılar diliyorum.

Bu çalışma benim mesleki yeterliliklerimi artırmam ve kişisel gelişimimi sağlamam adına önemli bir çalışmadır. Bu çalışmayı, eğitim hayatım boyunca kültürel olarak gelişmemi sağlayan, güçlüklerle mücadele etmeyi öğretti ve her zaman çocuklarının arkasında duran kıymetli anneme ithaf ediyorum. Tezimi ortaya çıkarma sürecinde bana her zaman destek veren, motivasyonumu artıran kıymetli annem Arzu KAHYAOĞLU'na teşekkür ederim.

Mehmet DERVİŞ

Kastamonu, 2025

İÇİNDEKİLER

Sayfa

TEZ ONAYI	ii
TAAHHÜTNAME	iii
ÖZET.....	iv
ABSTRACT	vi
TEŞEKKÜR	viii
İÇİNDEKİLER	ix
ŞEKİLLER VE GÖRSELLER DİZİNİ.....	xii
TABLolar DİZİNİ	xiii
SİMGELER VE KISALTMALAR DİZİNİ	xiv
1. GİRİŞ.....	1
1.1 Problem Durumu	1
1.2 Araştırmanın Amacı	3
1.3 Araştırmanın Önemi	4
1.4 Araştırmanın Problemi	4
1.5 Sınırlılıklar.....	6
1.6 Sayıltılar/Varsayımlar	6
1.7 Tanımlar	6
2. KURAMSAL ÇERÇEVE	8
2.1 İşbirlikli Öğrenme	8
2.1.1 İşbirlikli Öğrenmenin Temelleri	8
2.1.2 İşbirlikli Öğrenmenin Tanımı	8
2.1.3 İşbirlikli Öğrenmenin Özellikleri.....	9
2.1.4 İşbirlikli ve Geleneksel Grupların Karşılaştırılması	11
2.1.5 İşbirlikli Öğrenme Teknikleri	12
2.2 Programlama	15
2.2.1 Programlamanın Tanımı	15
2.2.2 Programlama Eğitimi	16
2.2.3 Programlama ve Bilgi İşlemsel Düşünme	17
2.2.4 Programlama, Algoritma ve Algoritmik Düşünme.....	18
2.2.5 Programlama ve Üst Düzey Düşünme Becerileri	19
2.2.6 Dünya’da Programlama Eğitimi	20
2.2.7 Türkiye’de Programlama Eğitimi	22
2.2.8 Programlama Öğretim Yaklaşımları	23
2.2.8.1 Eşli programlama	25
2.2.8.2 Bilgisayarsız programlama.....	26
2.2.8.3 Fiziksel (Robotik) programlama	26
2.2.8.4 Metin tabanlı programlama	26
2.2.8.5 Blok tabanlı programlama.....	27
2.3 Öz-Yeterlik	28
2.3.1 Öz-Yeterliğin Tanımı ve Öz-Yeterlik İnancı.....	28
2.3.2 Programlama Öz-Yeterliği.....	29
2.4 Öz-Yeterlik ve Öz-Düzenleme İlişkisi	30
2.5 Öz-Düzenleme.....	30
2.5.1 Öz-Düzenlemenin Tanımı.....	30

2.5.2	Öz-Düzenleme Becerisi	31
2.5.3	Öz-Düzenleme Becerisi ve Programlama	32
2.6	Öz-Düzenleme ve Akademik Başarı İlişkisi	33
2.7	Akademik Başarı	33
2.8	İşbirlikli Öğrenme, Akademik Başarı, Öz-Yeterlik ve Öz-Düzenleme İlişkisi	34
2.9	İlgili Bilimsel Çalışmalar	34
2.9.1	Yurt İçinde Yapılan Çalışmalar	34
2.9.2	Yurt Dışında Yapılan Çalışmalar	39
3.	YÖNTEM	43
3.1	Araştırmanın Modeli	43
3.2	Araştırmanın Çalışma Grubu	44
3.3	Araştırma Verilerinin Toplanması	45
3.3.1	Algılanan Öz-Düzenleme Ölçeği	45
3.3.2	Programlama Öz-yeterliği Ölçeği	46
3.3.3	Akademik Başarı Testi	47
3.4	Araştırmanın Etik İzinleri	51
3.5	Uygulama Süreci	51
3.5.1	Dersin Genel Planı	52
3.5.2	Günlük Plan	54
3.5.3	Eğitim Materyalleri	56
3.6	Verilerin Analizi	59
3.7	Araştırmacının Rolü	62
3.8	Geçerlik ve Güvenirlik Çalışmaları	63
4.	BULGULAR	66
4.1	Çalışma Gruplarının Ön-Test Puanlarına Ait Betimsel İstatistik Bulguları	66
4.2	Çalışma Gruplarının Son-Test Puanlarına Ait Betimsel İstatistik Bulguları	67
4.3	Programlama Öz-Yeterliği Bulgularının Çalışma Gruplarına Göre Karşılaştırılması	68
4.4	Öz-Düzenleme Becerisi Bulgularının Çalışma Gruplarına Göre Karşılaştırılması	69
4.5	Akademik Başarı Testi Bulgularının Çalışma Gruplarına Göre Karşılaştırılması	70
4.6	Deney Grubu Ön-Test ve Son-Test Bulgularının Karşılaştırılması	71
4.7	Kontrol Grubu Ön-Test ve Son-Test Bulgularının Karşılaştırılması	74
5.	SONUÇ VE TARTIŞMA	76
5.1	Programlama Akademik Başarısına Yönelik Tartışma	76
5.2	Programlama Öz-Yeterliğine Yönelik Tartışmalar	78
5.3	Öz-Düzenleme Becerisine Yönelik Tartışmalar	80
6.	ÖNERİLER	83
6.1	Uygulamaya Yönelik Öneriler	83
6.2	Yeni Araştırmalara Yönelik Öneriler	84
KAYNAKLAR	86	
EKLER	104	
EK A	Ölçek Kullanım İzinleri	105
EK B	Resmi İzinler	106
EK C	Günlük Planlar	108

EK D Programlama Öz-Yeterliđi Ölçeđi	119
EK E Algılanan Öz-Düzenleme Ölçeđi.....	121
EK F Akademik Başarı Testi.....	122
EK G Uygulamaya Ait Görseller	129



ŞEKİLLER VE GÖRSELLER DİZİNİ

Sayfa

Şekil 2.1 Programlama eğitimi ve bilişsel beceriler arasındaki ilişki	19
Şekil 2.2 MTP ve BTP kodlarının karşılaştırılması	28
Şekil 3.1 Scratch platformu arayüzü	57



TABLolar DİZİNİ

Sayfa

Tablo 2.1 İşbirlikli gruplar ile geleneksel grupların karşılaştırılması	12
Tablo 2.2 Programlama öğretim yaklaşımları.....	24
Tablo 3.1 Araştırma deseni	44
Tablo 3.2 Katılımcıların gruplara göre dağılımı	44
Tablo 3.3 Belirtke tablosu	48
Tablo 3.4 Madde analizleri	49
Tablo 3.5 Ders planı.....	52
Tablo 3.6 Birinci haftaya ait örnek günlük plan	54
Tablo 3.7 Scratch kod blokları	58
Tablo 3.8 Akademik başarı testi normallik verileri	59
Tablo 3.9 Akademik başarı testi çarpıklık ve basıklık değerleri.....	60
Tablo 3.10 Programlama öz-yeterliği testi normallik verileri.....	60
Tablo 3.11 Programlama öz-yeterliği testi çarpıklık ve basıklık değerleri.....	60
Tablo 3.12 Öz-düzenleme testi normallik verileri	61
Tablo 3.13 Öz-düzenleme testi çarpıklık ve basıklık değerleri.....	61
Tablo 4.1 Çalışma grupları ön-test puan ortalamaları.....	66
Tablo 4.2 Çalışma gruplarının son-test puan ortalamaları	67
Tablo 4.3 Çalışma gruplarının programlama öz-yeterliği ön-testine ait bağımsız örneklem t-testi sonuçları	68
Tablo 4.4 Çalışma gruplarının programlama öz-yeterliği son-testine ait mann-whitney u sonuçları	69
Tablo 4.5 Algılanan öz-düzenleme ön-testinin çalışma gruplarına ait bağımsız örneklem t-testi sonuçları	69
Tablo 4.6 Algılanan öz-düzenleme ölçeği son-testinin çalışma gruplarına ait bağımsız örneklem t-testi sonuçları.....	70
Tablo 4.7 Çalışma gruplarına göre akademik başarı ön-testine ait bağımsız örneklem t-testi sonuçları	70
Tablo 4.8 Çalışma gruplarına göre akademik başarı son-testine ait mann-whitney u sonuçları	71
Tablo 4.9 Deney grubu akademik başarı wilcoxon işaretli sıralar testine ait betimsel istatistikler.....	72
Tablo 4.10 Deney grubu akademik başarı wilcoxon işaretli sıralar testine ait sonuçları	72
Tablo 4.11 Deney grubu programlama öz-yeterliği wilcoxon işaretli sıralar testine ait betimsel istatistikler	72
Tablo 4.12 Deney grubu programlama öz-yeterliği wilcoxon işaretli sıralar testine ait sonuçları.....	73
Tablo 4.13 Deney grubu öz-düzenleme ilişkili örneklem t-testine ait sonuçlar	73
Tablo 4.14 Kontrol grubu akademik başarı ilişkili örneklem t-testine ait sonuçlar	74
Tablo 4.15 Kontrol grubu Programlama öz-yeterliği ilişkili örneklem t-testine ait sonuçlar	74
Tablo 4.16 Kontrol grubu algılanan öz-düzenleme ilişkili örneklem t-testine ait sonuçlar	75

SİMGELER VE KISALTMALAR DİZİNİ

Kısaltmalar

AB	: Avrupa Birliği
ABT	: Akademik Başarı Testi
AÖÖ	: Algılanan Öz-düzenleme Ölçeği
BİD	: Bilgi işlemsel düşünme
BK	: Basıklık katsayısı
BTP	: Blok Tabanlı Programlama
BTY	: Bilişim Teknolojileri ve Yazılım
ÇK	: Çarpıklık katsayısı
K-12	: Okulöncesi, ilköğretim ve ortaöğretim kademesi
MEB	: Milli Eğitim Bakanlığı
MTP	: Metin Tabanlı Programlama
PÖÖ	: Programlama Öz-yeterlik Ölçeği
YY	: Yüzyıl

1. GİRİŞ

Bu bölümde problem durumu, araştırmanın amacı, araştırmanın önemi, araştırma problemi, sınırlılıklar, sayılı ve tanımlar yer almaktadır.

1.1 Problem Durumu

21. yüzyıl becerileri, bilgi ve teknoloji çağında bireylerin çağa ayak uydurabilmesi için kazanması gereken beceriler olarak ifade edilmektedir. 21. yüzyıl becerileri bireylerden aktif olma, eleştirel düşünme, iş birliği ve liderlik, girişimcilik ve inisiyatif alma, etkili sözlü ve yazılı iletişim, sorgulama ve problem çözme becerilerine sahip olmasını beklenmektedir (Kayhan vd., 2019). Ayrıca bu beceriler, kendini yenileyen değişime ayak uyduran bireylerin ön plana çıkmasını sağlamaktadır. Bireylerin girişimcilik ve inisiyatif alma becerisine sahip olmaları kendi özyönetim ve özdenetimlerini yapmalarını bunun sonucunda iş yaşamında aktif ve başarılı olmalarını sağlamaktadır (Göksün, 2016). Diğer yandan, 21. yüzyıl becerileri bireylerin zihin süreçlerinin aktif olmasını eleştirel düşünme, yaratıcı düşünme, problem çözme, iletişim ve iş birliği analitik düşünme becerilerine sahip olmasına odaklanmaktadır. Tüm bu beceriler göz önüne alındığında tüm becerilerin bireylere kazandırılması için programlama (kodlama) yeteneği ön plana çıkmaktadır. Sayın ve Seferoğlu (2016) programlamayı 21. yüzyılda kilit bir beceri olarak görmektedir. Çünkü bireyler programlama yapabilmek için problemi anlamalı, analiz etmeli ve yaratıcı düşünerek çözüm üretebilmelidir. Programlama becerilerinin günümüzdeki dijitalleşmiş toplumunu anlamak için rehber olduğunu, problem çözme, yaratıcılık ve mantıksal düşünme gibi 21. yüzyıl becerilerini geliştirebildiği bilinmektedir (Tuomi vd., 2017). Bireylerin programlama sürecinde 21. yüzyıl becerilerini aktif kullandığı görülmektedir (Yıldız vd., 2017). Bu doğrultuda World Economic Forum (2020, Akt. Zurnacı ve Turan, 2022), 21. yüzyıl becerileri arasından aktif öğrenme, problem çözme, eleştirel düşünme, teknoloji ve tasarım, programlama gibi bazı becerilerin 2025 yılına kadar daha fazla öne çıkacağını vurgulamaktadır. Eğitim kurumlarının işlevlerinden biri 21. yüzyılda toplumun ve işletmelerin istediği bilgi ve becerilere sahip bireyleri yetiştirmek olmuştur (Kayhan vd., 2019).

Programlama, bilgisayarda problem çözme için yapılan tüm süreçleri kapsayan bir ifadedir (Kalelioğlu, 2015; Yıldız, 2019). Son yıllarda programlama eğitimi üzerine yapılan araştırmalarda problem çözme becerisinin programlama ile ilişkili olduğu ve programlama eğitiminin öğrencilerin problem çözme becerisini geliştirdiği görülmektedir (Kukul ve Gökçearslan, 2014). 21. yüzyıl becerilerinden problem çözme becerisi hedefe ulaşmak için en kısa yolu kullanmaktır. Problem çözme becerisinin alt adımları problemi anlama, planı uygulama, plan yapma ve çözümleri değerlendirmedir (Özdişi ve Katrancı, 2020). Programlama öğrenmek öğrencilerin motivasyonunu artırır, dijital yetkinliklerini, analitik düşünme ve işbirlikli öğrenme becerilerini geliştirir (Kaya ve Alban, 2020). Bu amaçla ülkeler programlama eğitimine günümüzde çok önem vermektedir ve K-12 eğitim programlarına bilgisayar programlama dersini ekleme yoluna gitmişlerdir (Bayburt ve Eğin, 2021). Bazı Avrupa ülkeleri erken yaşlarda çocuklara programlama becerisini kazandırmak amacıyla programlama eğitimini eğitim programlarına eklemişlerdir. Programlamayı öğreten bu Avrupa ülkeleri; Litvanya, İngiltere, Avusturya, Çek Cumhuriyeti, Danimarka, Bulgaristan, Estonya, İrlanda, Fransa, İspanya, Slovakya, Malta, Polonya, Macaristan ve Portekiz'dir (Bacconi vd., 2016). AB ülkelerinin programlamaya önem vermesinin sebeplerinden birisi mantıksal düşünme ve problem çözme becerilerinin gelişmesini sağlamaktır (Sayın ve Seferoğlu, 2016). AB, çocuklarda erken yaşlardan başlayarak kodlama ve bilgi işlemsel düşünmeyi desteklemek için kodlama haftası (Code Week) hareketini başlatmıştır. Kodlama haftası etkinlikleri aracılığıyla öğretmenler kodlama deneyimleri paylaşılmakta aynı zamanda öğretmen ve öğrenciler kodlamaya teşvik edilmektedir (URL-1, 2019; Sayın, 2020). AB kodlama haftası etkinlikleriyle ön plana çıkarken ABD ise “kodlama saati” vb. kodlama etkinlikleriyle kodlama öğretimini desteklemektedir (Sayın ve Seferoğlu, 2016).

Programlama, Türkiye’de ortaokul seviyesinde BTY dersinde “problem çözme ve programlama” ünitesi kapsamında öğretilmeye başlanmaktadır. BTY dersi haftalık iki saat, 5. sınıflarda ve 6. sınıflarda zorunlu olarak öğretilmektedir. Problem çözme ve programlama ünitesi kapsamında öğrencilerin programlama becerilerini geliştirilmesi için gerekli dijital içeriklere yer verilmiştir (MEB, 2018b).

Programlama eğitiminde öğrencilerin başarısını artırılmasını sağlayan nedenlerden birisi kodlama öğretiminde kullanılan yöntem, teknik ve yaklaşımlardır (Demirören, 2016). Bilgisayar programlamasını temel seviyede öğrenmeye başlayan öğrenciler kodları düzenlerken ve programlama hatalarını bulurken zorluk yaşamalarından dolayı çocukların programlamayı öğrenebilmesi için scratch vb. blok tabanlı kodlama platformları kullanılmaktadır (Çal ve Can, 2019). Blok tabanlı programlama dilleri çocuklara yönelik yapboz parçaları şeklinde sürükle-bırak mantığıyla programlama yapmaya olanak verdiği için kullanım kolaylığı sunmaktadır (Sırakaya, 2018). Çocuklar programlama öğrenirken kendi bilgi ve deneyimlerinden yararlandığı gibi akranlarının deneyim ve bilgilerinden de yararlanmaktadır. Bu doğrultuda programlamanın grupla öğretimini kapsayan yaklaşımlar ortaya çıkmıştır (Demirören, 2016). Bu yaklaşım öğrencilerin programlamayı grupla öğrenmelerini sağlamak için işbirlikli öğrenmenin programlama öğretimine yansması eşli kodlama (programlama) yöntemidir (Tosuntaş vd., 2020). Eşli programlama öğretim yaklaşımının öğrencilerde bazı üst düzey düşünme becerilerini geliştirdiği bilinmektedir. Başlıca gelişen bu üst düzey beceriler, problem çözme ve kritik düşünmedir (Çal ve Can, 2019). Eşli programlama, öğrencilerin birlikte programlama etkinlikleri gerçekleştirmesini sağlarken aynı zamanda sosyal ve iletişim becerilerini geliştirmekte ve bilgi paylaşımını kolaylaştırmaktadır (Çal ve Can, 2019; Tosuntaş vd., 2020).

MEB, öğretim programlarına iş birliği ve iletişimi geliştirmeye yönelik; inisiyatif alma ve girişimcilik, sosyal ve vatandaşlık, kültürel farkındalık ve ifade becerilerini dahil etmiştir (Yalçın, 2018). Yine programlama becerilerini geliştirmek amacıyla MEB, BTY öğretim programında problem çözme, algoritma becerilerine ve bilgi işlemsel düşünme yer vermiştir (MEB, 2018b). Bu durumlar düşünüldüğünde işbirlikli öğrenme yaklaşımının programlama öğretimi üzerindeki etkileri ortaya çıkarılarak, programlama öğretim yaklaşımları literatürüne katkı sağlanacaktır.

1.2 Araştırmanın Amacı

Çalışmanın amacı işbirlikli öğrenme teknikleriyle kurgulanan programlama öğretimi etkinliklerinin altıncı sınıf öğrencilerinin programlama öz-yeterliklerine, öz-düzenleme becerilerine ve akademik başarılarına etkisini ortaya çıkarmaktır.

1.3 Araştırmanın Önemi

Bilgisayar programlama becerisi hızlı teknolojik gelişmelerin yanı sıra problem çözme, mantıksal düşünme ve yaratıcılık gibi diğer becerilerinin de gelişimini sağladığı için son yıllarda popülerlik kazanmıştır (Scherer vd., 2020). Bu doğrultuda bilgisayar bilimleri eğitimindeki araştırmalar programlama öğretmek için etkili yollar bulmaya çalışmıştır fakat araştırmalarda genel olarak sınırlı bir başarı elde etmiştir (Tuomi, 2017). Literatürdeki eksikleri tamamlayacak ve katkıda bulunacak yeni programlama öğretim yaklaşımlarının araştırması önemlidir. 21.yy. da iş birliğinin öne çıkan katılım biçimi olması nedeniyle bireylerin birlikte düşünme ve çalışması gerekliliği ortaya çıkmıştır. Bu durum bireysellikten ekip çalışmasına geçilmesine sebep olmuştur (Laal ve Ghodsi, 2012). Öğrencilerin hedeflerine ulaşabilmesi amacıyla birlikte çalışma ve birbirlerine destek olma gibi sosyal becerileri öne çıkaran öğrenme modeli olan işbirlikli öğrenmenin (Marr, 1997), programlama öğretiminde etkili olacağı ve 21.yy. gerekliliklerinden olan iş birliği becerisini kazandıracağı düşünüldüğünden önemli bir öğrenme yaklaşımı olarak görülmektedir. Bu çalışmanın önemi, altıncı sınıf öğrencilerinin işbirlikli öğretim tekniklerinin programlama öğretiminde başarısını ortaya çıkararak programlama öğretim yaklaşımları literatürüne katkıda bulunması olacaktır. Diğer yandan ise işbirlikli öğrenme ile kurgulanmış programlama öğretiminin öğrencilerin programlama öz-yeterlikleri, öz-düzenleme becerileri ve akademik başarıları üstündeki etkisi ortaya çıkarılabilecek ve literatüre katkı sağlayacaktır.

1.4 Araştırmanın Problemi

Bu çalışmanın problemini, ortaokul BTY dersi kapsamında işbirlikli öğrenme teknikleriyle öğretilen programlama konularının 6. sınıf öğrencilerinin programlama öz-yeterlikleri, öz-düzenleme becerileri ve akademik başarıları üzerinde etkisi var mıdır? sorusu oluşturmaktadır.

Bu amaç doğrultusunda yarı deneysel desende gerçekleştirilen araştırmada aşağıda belirtilen alt problemlere cevap aranmıştır:

Deney grubu öğrencileri ile kontrol grubu öğrencilerinin;

1. Programlama öz-yeterliği ön-test puanları arasında anlamlı bir fark var mıdır?
2. Programlama öz-yeterliği son-test puanları arasında anlamlı bir fark var mıdır?
3. Öz-düzenleme ön-test puanları arasında anlamlı bir fark var mıdır?
4. Öz-düzenleme son-test puanları arasında anlamlı bir fark var mıdır?
5. Akademik başarı ön-test puanları arasında anlamlı bir fark var mıdır?
6. Akademik başarı son-test puanları arasında anlamlı bir fark var mıdır?

Deney grubu öğrencilerinin;

7. Akademik başarı ön-test ve son-test puanları arasında anlamlı bir fark var mıdır?
8. Programlama öz-yeterliği ön-test ve son-test puanları arasında anlamlı bir fark var mıdır?
9. Öz-düzenleme ön-test ve son-test puanları arasında anlamlı bir fark var mıdır?

Kontrol grubu öğrencilerinin;

10. Akademik başarı ön-test ve son-test puanları arasında anlamlı bir fark var mıdır?
11. Programlama öz-yeterliği ön-test ve son-test puanları arasında anlamlı bir fark var mıdır?
12. Öz-düzenleme ön-test ve son-test puanları arasında anlamlı bir fark var mıdır?

1.5 Sınırlılıklar

1. Araştırma, 2024-2025 eğitim-öğretim yılında Kastamonu ili Taşköprü ilçesinde yer alan bir devlet ortaokulunun 6. sınıfında öğrenim gören 43 öğrenci ile sınırlıdır.
2. Araştırma Bilişim Teknolojileri ve Yazılım dersi 6. sınıf öğretim programındaki “problem çözme ve programlama” ünitesinin ardışık kazanımları ile sınırlıdır.
3. Araştırma 8 hafta süresince toplanan veriler ile sınırlıdır.
4. Araştırmada ulaşılan bulgular, kullanılan “Programlama Öz-yeterlik Ölçeği”, “Algılanan Öz-Düzenleme Ölçeği” ve “Akademik Başarı Testi” veri toplama araçlarıyla elde edilen verilerle sınırlıdır.
5. Araştırma programlama öz-yeterlikleri, öz-düzenleme becerileri ve akademik başarı değişkenleri ile sınırlıdır.

1.6 Sayıltılar/Varsayımlar

1. Araştırmada kullanılan ölçeklere ve başarı testine katılımcıların samimiyetle cevap verdikleri varsayılmıştır.
2. Deney grubundaki etkinliklere katılımcı öğrencilerin istekli ve ciddi olarak katıldıkları varsayılmıştır.
3. Deney ve kontrol grubunun yaş ve cinsiyet değişkenleri açısından denk olduğu varsayılmıştır.

1.7 Tanımlar

İşbirlikli Öğrenme: Yetenek, başarı düzeyleri gibi çeşitli değişkenler bakımından farklı özelliklere sahip bireylerin küçük gruplar kurarak öğrenme gerçekleştirdiği ve bir dizi teknikten oluşan öğrenme yaklaşımıdır (Slavin, 1987).

Programlama (Kodlama): Programlama, bir problemin çözümü için programlama dilleri kullanılarak bilgisayara komut vermek için kodlar yazmak, amaca uygun algoritma tasarlamak ve yazılan kodları derleyip çalışma işlemi olarak tanımlanmaktadır (Gökoğlu, 2017). Programlama ve kodlama aynı anlamda kullanılan ifadelerdir (Eskici vd., 2020).

Blok Tabanlı Programlama: Programlamaya yeni başlayanlar için görsel arayüze sahip sürükle-bırak mantığı ile kodla yazmayı sağlayan programlama aracıdır (Yıldız Durak, 2020).

Eşli Programlama: İşbirlikli öğrenme yaklaşımının programlama öğretimine yansmasıyla ortaya çıkan grupla programlama öğretimi yöntemidir (Tosuntaş vd., 2020).

Programlama Öz-yeterliği: Öğrencilerin programlamaya karşı kendine duyduğu inanç ve güvendir (Ekici ve Çınar, 2020).

Öz-düzenleme: Öğrencilerin kendi özelliklerinin farkında olması, kendi öğrenme hedeflerini belirlemesi, motivasyon ve davranışlarını düzenlemesi, kendilerini kontrol çabası olarak tanımlanmaktadır (Zimmerman, 1989).

2. KURAMSAL ÇERÇEVE

Bu bölümde çalışmanın kuramsal temelini oluşturan bilgiler ve çalışmayla ilişkili araştırmalara yer verilmiştir. Literatür taraması kapsamında sırasıyla işbirlikli öğrenme, programlama, öz-yeterlik, öz-düzenleme kavramlarına ait bilgilere yer verilmiştir.

2.1 İşbirlikli Öğrenme

2.1.1 İşbirlikli Öğrenmenin Temelleri

İngilizce adı “Collaborative Learning”, “Collective Learning”, “Learning Communities”, “Peer Learning”, “Reciproal Learning”, “Team Learning”, “Study Circles”, “Study Group”, “Peer Teaching” ve “Team Work” gibi terimlerle ifade edilen yaklaşım Türkçe’ye “işbirlikli öğrenme” veya “kubaşık öğrenme” olarak çevrilmiştir (Bayrakçeken vd., 2013, s. 2).

İşbirlikli öğrenmenin dayandığı temel, sosyal yapılandırmacılık teorisidir. Sosyal yapılandırmacılık teorisi, Lev Semyonovich Vygotsky’nin (1896-1934) fikirleri etrafında gelişmiştir (Li ve Lam, 2013). Vygotsky, insanların öğrenmelerini anlamaya çalışmış ve öğrenmelerin kültür, toplum, dil ve etkileşimin oluşturduğunu düşünmüştür. Vygotsky sosyal yapılandırmacılık üzerine görüşlerinde “iş birliği” ve “kültür” vurgusu yapmıştır. Bireylerin toplum içinde sosyal etkileşimleri ile dil, düşünme ve akıl yürütme gibi becerilerinin geliştiğini ileri sürmüştür. Bu yüzden ebeveyn, öğretmen ve akran gibi çocukların çevresindeki bireylerin öğrenme üzerinde aktif bir rolü olduğunu savunmaktadır (Li ve Lam, 2013).

2.1.2 İşbirlikli Öğrenmenin Tanımı

Slavin (1987)’e göre işbirlikli öğrenme, başarı düzeyleri ve yetenekleri bakımından karma öğrencilerden oluşan küçük öğrenme gruplarını kapsayan bir öğretim yöntemidir. Bayrakçeken vd. (2013)’ne göre işbirlikli öğrenmenin öğrenciyi merkeze alan öğrencinin aktif rol üstlendiği bir grupta öğrenme yaklaşımı olduğunu

söylemişlerdir. Efe vd. (2007) işbirlikli öğrenme, ortak bir amaç uğruna farklı yeteneklere sahip öğrencilerin küçük gruplar halinde çalışarak en yüksek düzeyde öğrenmelerini sağlayan bir yaklaşımdır. Aydın ve Atalay (2015)'a göre işbirlikli öğrenme, akademik bir görevi yerine getirmek amacıyla oluşturulan küçük grupları ve grupların başarısının ödüllendirildiği öğrenme yaklaşımıdır.

Yapılan tanımlar incelendiğinde işbirlikli öğrenmenin; öğrencilerin öğrenme süreçlerine aktif katılım sağladığı, bilgi ve beceri bakımından farklı özelliklere sahip öğrencilerin toplandığı heterojen yapıdaki küçük öğrenme gruplarıyla gerçekleştirilen ve bir dizi öğrenme tekniklerinden oluşan öğrenme modeli olduğu anlaşılmaktadır.

2.1.3 İşbirlikli Öğrenmenin Özellikleri

İki ile altı kişiden oluşan farklı yetenek, cinsiyet, bilgi seviyesi gibi farklı özelliklere dayanan heterojen öğrenme gruplarının ortak amaç için çalışması ve grup olarak ödüllendirilmesi işbirlikli öğrenmenin önemsendiği konulardır (Bayrakçı vd., 2013). İşbirlikli öğrenme grupları, öğrencileri gruplarındaki tüm bireylerin performansına göre ödüllendirerek sınıflardaki rekabeti veya bireyselliği azaltmaya çalışır. İşbirlikli öğrenmenin öğrencilerin akademik, bilişsel ve sosyal becerilerini geliştirme amacı taşıdığını belirtmektedir (Slavin ve Cooper, 1999). Öğrenci başarısının, işbirlikli öğrenme gruplarında geleneksel öğrenme gruplarına göre daha yüksek olduğu bilinmektedir. Eğlenceli bir öğrenme yaklaşımı olması ve sosyal becerileri geliştirmesi işbirlikli öğrenmenin ön plana çıkan özellikleridir. (Slavin, 1987). Ayrıca, işbirlikli öğrenme, biliş ve üstbiliş ile bağlantılıdır bu yüzden üst düzey düşünme becerilerinin gelişmesini sağlar. Grup üyelerinin kişiler arası sözlü fikir paylaşımlarının öğrencilerin eleştirel düşünme, zihinsel muhakeme ve bilişsel düşünmeye katkı sağladığı bilinmektedir. Bu doğrultuda işbirlikli öğrenme gruplarında öğrencilerin birbirleriyle farklı düşünceleri paylaşımlarının, onları yaratıcı düşünmeye teşvik ettiği düşünülmektedir (Holt, 1993).

İşbirlikli öğrenmenin diğer temel özellikleri aşağıda sıralanmıştır:

- Olumlu bağlılık (Johnson ve Johnson, 1999, s. 71),

- Bireysel sorumluluk (Johnson ve Johnson, 1999, s. 71),
- Yüz yüze etkileşim (Johnson ve Johnson, 1999, s. 71),
- Sosyal beceriler (Johnson ve Johnson, 1999, s. 71),
- Grup süreçleri (Johnson ve Johnson, 1999, s. 71),
- Ödüller (Bayrakçeken, vd., 2013, s. 5),
- Öğretmenin Rolü (Bayrakçeken vd., 2013, s. 4).

Olumlu bağlılık, grup üyelerinin birbirlerine karşı bağlı olduğu algısıdır. İşbirlikli öğrenme gruplarındaki öğrencilerin birinin başarılarının diğerinin başarısını da etkileyeceği bilincine sahip olması durumudur. İşbirlikli öğrenmede başarı öğrencilerin etkileşimiyle gerçekleşmektedir. Olumlu bağlılık oluşturmak için öğrencilere verilen materyaller, ortak ödüller, kaynaklar ve görevler kullanılabilir (Johnson ve Johnson, 1999).

Bireysel sorumluluk, gruptaki her bireyin bireysel performansının grubun başarısını etkilemesi durumu olarak ifade edilebilir. İşbirlikli öğrenmede amaçlardan birisi gruptaki bireyleri güçlü hale getirmektir. Bireylerin güçlenmesi için kendine düşen görev ve sorumlulukları yerine getirmelidir (Johnson ve Johnson, 1999).

Yüz yüze etkileşim, bireylerin birbirini destekleyerek ve başarıya teşvik ederek etkileşim kurmasıdır. Grup üyeleri arasındaki yüz yüze etkileşim arttıkça akrana karşı sorumluluk artar ve birbirlerinin öğrenmelerine yardımcı olmaktadır. İşbirlikli öğrenmede bireylerin oluşturduğu sosyal ortamlar sayesinde birbirlerine destek olması ve yol göstermesi grubunun birbiriyle etkileşimine bağlıdır (Johnson ve Johnson, 1999).

Sosyal beceriler; karar verme, liderlik, çatışma yönetimi, güven oluşturma, iletişim gibi becerilerden oluşmaktadır, bireylere bu becerilerin öğretilmesi gereklidir. İşbirliği çalışmaları kişilerarası ve küçük grup becerileri gerektirmektedir. Sosyal becerileri

zayıf öğrencilerin becerileri, işbirlikli öğrenme grupları sayesinde geliştirilebilir (Johnson ve Johnson, 1999).

Grup süreçleri, grup üyelerinin hedeflerine ne kadar ulaştıklarını ve etkili çalışma için ilişkilerini ne kadar sürdürdüklerini değerlendirmeleridir. Grubun işleyebilmesi için grup üyelerinin, grup işleme sürecinde bireyler arası ilişki sorunlarını tanımlaması ve çözmesi gerekir. Üyelerin, grup süreçlerinde hangi davranışları devam ettireceğine ve hangi davranışlarının yararlı olup olmadığına yönelik kararlar alması beklenmektedir (Johnson ve Johnson, 1999).

Ödüller, gruba verilen görevler yerine getirilince birinci gruba olmak üzere diğer gruplara da verilmektedir. Verilen ödüller grup üyelerini ortak bir amaç uğruna çalışmaya teşvik eder, olumlu bağlılığın oluşmasına ve yardımlaşmaya olanak sağlar. Ödüller öğrencilerin motive olmasını sağlayarak çalışmanın devamlılığına olanak verir. Grup ödülleri öğretmen veya öğrenciler tarafından gruptaki bireylerin özelliklerine uygun olarak verilir (Bayrakçı vd., 2013).

Öğretmenin rolü, işbirlikli öğrenme yaklaşımında öğrencilerle etkileşim kurarak onların bağımsız düşünmesine aracılık etmektir. Öğrencilerin kendi performanslarını ortaya çıkarması için öğretmenin önce kritik noktalara müdahale etmesi sonra kendini geriye çekerek öğrenme gruplarına verilen görevlerin üstesinden gelmesini sağlamalıdır (Yanpar Yelken ve Akay, 2019). Bu doğrultuda öğretmenin öğrencilere rehberlik yapması ve görevlerini kolaylaştırması beklenmektedir (Efe vd., 2007). Öğretmenin görevi sadece ders anlatmak değildir aynı zamanda öğrencilerin liderlik, paylaşma, empati, uzlaşma ve iletişim gibi sosyal becerilerinin gelişmesine yardımcı olmaktır (Bayrakçı vd., 2013).

2.1.4 İşbirlikli ve Geleneksel Grupların Karşılaştırılması

Okullarda geleneksel olarak uygulanan kümelenmiş grup çalışmalarını işbirlikli öğrenme gruplarından ayıran özellikler vardır. Geleneksel gruplarda öğrencilerin bireysel çalışması ön plana çıkmaktadır (Bayrakçı vd., 2013). Bu yüzden işbirlikli öğrenmenin geleneksel öğrenme gruplarına göre üstün yönleri vardır. Bu üstün yönler Tablo 2.1’de verilmiştir (Yıldız, 1999; Efe vd., 2007; Bayrakçı vd., 2013).

Tablo 2.1 İşbirlikli gruplar ile geleneksel grupların karşılaştırılması

İşbirlikli öğrenme grupları	Geleneksel öğrenme grupları
1. Öğrenme grupları heterojen yapıdadır. Üyeler farklı yetenekli ve kişilik özelliklerine sahiptir.	1. Öğrenme grupları homojen olabilir. Heterojen grup oluşturmaya dikkat edilmez.
2. Liderlik grup üyeleri arasında paylaşılmaktadır.	2. Grubu yönlendiren bir lider vardır.
3. Üyeler birbirinin öğrenmelerinden sorumludur.	3. Üyelerinin birbirinin öğrenmelerinden genellikle sorumlu olamamaktadır.
4. Grup üyeleri arasında iyi çalışma ilişkileri vardır. Grup tek bir ürün ve çalışma ortaya çıkarırlar.	4. Grup üyeleri bireysel çalışmalar yaparlar ve bireysel ürünler ortaya çıkar.
5. Grup üyeleri arasında olumlu bağlılık vardır. Herkes birbirinin öğrenmesinden sorumludur.	5. Grup üyeleri arasında olumlu bağlılık görülmez.
6. Sosyal beceriler önemlidir ve süreçte iletişim becerileri öğretilir.	6. Sosyal beceriler önemli görülmez.
7. Öğretmen yönlendirici rolündedir, gruplarda çıkan problemleri çözer ve dönüt verir.	7. Öğretmen gruplara nadiren katılır, gözlemler ve bireysel çalışmalar değerlendirilir.
8. Öğretmen grubun iyi çalışması için işlemleri yapılandırır.	8. Öğretmen grubun işlemlerini yapılandırmaz.
9. Grubun ortak amacı için grup üyelerinin bireysel sorumluluğu vardır.	9. Grup üyelerinde grup adına bireysel sorumluluk gözlenmez.

2.1.5 İşbirlikli Öğrenme Teknikleri

İşbirlikli öğrenmenin öğretim sürecindeki etkisini artırma amacıyla ders etkinliklerinin ile sınıf ortamının düzenlenmesinden doğan işbirlikli öğrenmenin farklı teknikleri vardır (Doymuş ve Koç, 2012). İşbirlikli öğrenme teknikleri aşağıda sıralanmıştır (Bayrakçeken vd., 2013).

Öğrenci Takımları Başarı Bölümleri Tekniğinde (ÖTBB), öğrenme grupları heterojen olarak en çok altı kişilik şekilde ayrılırlar (Bayrakçeken vd., 2013). Bu teknikte

öğretilen konu ve içerikler öğretmen tarafından öğrencilere aktarılmaktadır (Slavin, 1987). Öğretmen dörder kişilik farklı özelliklere sahip heterojen öğrenme grupları oluşturur (Bayrakçeken vd., 2013). Öğretmen, öğrencilerin yapılandırılmış gruplarında çalışmalar gerçekleştirmesine olanak tanımaktadır. Konu anlatımı ve öğrenci çalışmaları tamamlandığında öğrencilere bireysel değerlendirme sınavları yapılmaktadır. Öğrencilerin aldıkları puanların grupları için ortalaması alınarak grup puanı belirlenmektedir. Grup puanlarına göre grup ödülü verilmektedir (Yanpar Yelken ve Akay, 2019).

Takım- Oyun- Turnuva Tekniği (TOT), ÖTBB tekniği ile çok benzer olan bir işbirlikli öğrenme tekniğidir (Yanpar Yelken ve Akay, 2019). Fakat TOT tekniği öğrencilerin birbiri ile yarışmasını ön plana çıkarmaktadır (Bayrakçeken vd., 2013). TOT tekniğinde öğretmen, konuyu geleneksel anlatım şeklinde sunmaktadır. İşbirlikli öğrenme grupları heterojen olarak oluşturulduktan sonra öğretmen tüm öğrencilere çalışma yaprakları dağıtarak yardımlaşmalarına olanak sağlar. Öğrenciler bu sayede birbirlerinin anlamadıkları veya eksik kalan yerleri giderirler (Efe vd., 2007). Sürecin sonunda öğrenciler çalışılan konu ile ilgili akademik oyun oynarlar (Yanpar Yelken ve Akay, 2019). Akademik oyun sürecinde farklı gruplardaki aynı başarı seviyesindeki öğrenciler yarışmaktadır (Efe vd., 2007). Yarışmada kazanan öğrencilerin puanları kendi gruplarının puanına yazılmaktadır (Bayrakçeken vd., 2013).

Takım Destekli Bireyselleştirme Tekniği (TDB), öğrenme grupları dört ile altı kişilik heterojen yapıdan oluşmaktadır (Bayrakçeken vd., 2013). TDB tekniğinde öğrenciler düzey belirleme veya geçmiş öğrenme düzeylerine yönelik bireyselleştirilmiş materyallerle çalışmaktadır. Bireysel çalışan öğrenciler heterojen yapıdaki farklı düzeydeki öğrencilerle bir araya gelerek çalışmaktadır. Öğrenme gruplarındaki öğrenciler birbirlerinin çalışmalarını kontrol etmektedir. Öğrencilere her ünite bazında sınav yapılır, takım puanları her ünite sonunda takım üyelerinin aldıkları puanlardan elde edilmektedir (Yanpar Yelken ve Akay, 2019).

Birlikte Soralım Birlikte Öğrenelim Tekniğinde (BSBÖ), öğrenme grupları büyüklüğü üç veya dört kişiden oluşmaktadır. Öğrenciler bir okuma parçasını içinden sessizce okur, öğretmen yönlendirmek ve temaları söylemek amacıyla önemli yerlerde

uyarılarda bulunur. Öğrenciler okudukları parçalardan ve temalardan yanıtı kolayca bulunabilecek sorular hazırlar ve bir kartın üzerine yazarlar. Bireysel sorular bir araya getirilerek grup soruları hazırlarlar. Oluşturulan grup sorularından rastgele seçilerek başka gruplara gönderilir. Gruplar kendilerine gelen soruyu grup sözcüleri sayesinde sınıfa sunarlar. Öğrenciler sunum yaparken öğretmen ve diğer öğrenciler sunumu değerlendirir. En son öğretmen konuyu özetler, tartışma başlatır ve en sonunda bireysel sınava alır. Sınavdan ve sunumdan toplanan puanlarla grup puanları belirlenir ve grup ödülü verilir (Bayrakçeken vd., 2013).

İşbirliğine Dayalı Birleştirilmiş Okuma ve Kompozisyon Tekniğinde (BİOK), sınıfta okuma grupları oluşturulur. Bu teknikte her okuma grubundan ikişer kişilik gruplar oluşturulmaktadır. Öğrenciler iki kişilik gruplarda karşılıklı olarak birbirlerinin okuma ve yazma becerilerini geliştirmesine yardımcı olmaktadır. Grup etkinlikleri sorma, özetleme, kompozisyon yazma, okuma ve yazma gibi etkinlikleri kapsamaktadır. Etkinlikler esnasında grup üyelerinin gösterdiği performansa göre üyeler ödüllendirilir (Bayrakçeken vd., 2013).

İş Birliği- İş Birliği Tekniği, öğrencileri merak ettikleri bir konuyu öğrendikleri ve grubun diğer üyeleri ile paylaştıkları bir tekniktir (Efe vd., 2007). Öğrenciler önce kendi merak ettikleri konuları araştırır daha sonra diğer arkadaşlarıyla paylaşım ve iş birliği yaparlar. Öğrenci merkezli sınıf tartışmaları için gruplar oluşturulur her grup 5 dakika sunum yaparlar (Bayrakçeken vd., 2013).

Birleştirme (Jigsaw) Tekniği, dört ile altı kişilik iş birliğini esas alan öğrenci grubunun öğrenci sayına göre grubun alt görevlere ayrıldığı bir tekniktir. Alt görevlerle ilgili olarak her gruptan aynı görevi alan öğrenciler çalışmak amacıyla yeni öğrenme grubu oluşturmaktadır. Aynı görevi alanların oluşturduğu bu grup uzmanlar grubu olarak ifade edilmektedir. Uzman gruptaki öğrenciler, yeni grubun üyelerinin eksiklerini gidermek ve birbirleriyle deneyimlerini paylaşmak amacıyla çalışma yapmaktadır. Bu grup eksikleri tamamlayıp gerekli paylaşımları ve tartışmaları yaptıktan sonra grup üyeleri eski gruplarına geri dönmektedirler. Eski grubundaki herkesin kendi uzmanlık alanındaki konuları diğer üyelere anlatmasıyla birbirlerinin öğrenmelerine yardımcı olurlar. Sürecin sonunda tüm öğrencilere bireysel değerlendirme yapılır.

Değerlendirme sonucuna göre hem bireysel hem grup puanları belirlenir (Yanpar Yelken ve Akay, 2019).

Grup Araştırması Tekniği J. Dewey'nin eğitim felsefesine dayanmaktadır. Grup araştırması öğrencilerin çeşitli deneyimleri kazanacak şekilde tasarlanmaktadır. Bu teknikte öğrenciler araştırmacı rolünü üstlenmektedir. Öğretmen, öğrencilere öğrencilerin çözebileceği problem teşkil eden bir konu verir, öğrenciler problemi çözme süreçlerinde kendi bilgilerini yapılandırırlar. Öğrenciler çözüme yönelik elde ettikleri bilgileri diğer grup üyeleriyle bir araya gelerek paylaşırlar. Öğrenciler araştırma sürecinde karar alma ve tartışma gibi konularda aktif olmaktadır (Efe vd., 2007).

Birlikte Öğrenme Tekniği, iki ile altı kişilik gruplardan oluşmaktadır. Birlikte öğrenme tekniği akademik becerilerin geliştirilmesini iş birliği becerilerine göre daha çok önemsemektedir. Öğretmen, öğrencilere bir materyali vererek bunu paylaşımlarını sağlamak veya bir bilginin birbirlerine öğretilmesini sağlayarak grubun bağlılığını arttırmayı amaçlamaktadır. Öğretmen öğrencilerin yapacakları akademik görevi açıklar öğrenciler verilen görevi yerine getirirler. Gruplar üyeleri birlikte çalışırken öğretmen gerekli yerlerde müdahale eder, açıklama yaparak üyeleri yönlendirir. Öğretmen grupların deneyimlerini paylaşımlarını sağlar. Süreç sonunda öğrenciler birlikte öğrendiği bilgileri özetlemesi ve açılması beklenmektedir (Bayrakçeken vd., 2013).

2.2 Programlama

2.2.1 Programlamanın Tanımı

Program, bilgisayara iş yaptırmak için verilen kod toplulukları olarak ifade edilmektedir (Çobanoğlu, 2009). Bilgisayar programlama, insan ve bilgisayar arasında etkileşim sağlayan yazılımlar veya problem çözme sürecidir (Erkoç, 2022). Van-Roy ve Haridi (2004) tarafından programlama, bireylerin bilgisayara istenilen işi yaptırabilmesi için uygulama geliştirme adımlarını ve bunun için yazılan komut dizilerini kapsayan bir ifade olarak belirtilmektedir. Programlama, bilgisayarı istenilen şekilde yönlendirmek için bilgisayara komut verme sürecidir (Turan vd., 2016;

Sönmez, 2019). Programlama, bir problemin çözümü için bilgisayara komut vermektir (Mıhcı Türker ve Pala, 2018; Anılan ve Gezer, 2020).

Yapılan tanımlar incelendiğinde programlamanın; bir problemi çözmek veya istenilen görevi yaptırmak amacıyla bilgisayara komut verme işlemi olarak ifade edilebilir.

2.2.2 Programlama Eğitimi

Programlama eğitimi, bireylerin üst düzey düşünme becerilerini geliştirmesini sağlayan bir yoldur (Li vd., 2023). Sistematik ve yaratıcı düşünmeyi gerektirir aynı zamanda problemlere özgün çözümler üretme sürecidir (Anılan ve Gezer, 2020). Programlama eğitimi, çocuğun günlük hayatta karşına çıkan sorunları kolayca çözmesini sağlayabilir (Aytekin vd., 2018). Problem çözme etkinlikleri sırasında öğrenciler düşüncelerini analiz eder, organize eder, değerlendirir ve ifade ederler (Buitrago Flórez vd., 2017). Karabak ve Güneş (2013)'e göre programlama mantığının erken yaşlarda öğrenilmesi diğer alanlardaki becerilerini de geliştirebilecektir. Programlama, keşfederek öğrenme sürecinin bir biçimidir. Programlama görevleri sırasında öğrenciler, problemlere çözüm üretirken eski ve yeni bilgiler arasında bağ kurarak öğrenirler (Li vd., 2023). Programlamayı öğrenmenin, öğrencilerin motivasyonlarını, problem çözmesini, analitik düşünmesini, işbirlikli öğrenmesini ve dijital okuryazarlıklarını olumlu yönde etkilediği bilinmektedir (Kaya ve Alpan, 2020). Ayrıca, programlamayı öğrenmenin çocukların yazma becerileriyle ilişkisi olduğu gibi çocukların etkileşimli hikayeler, oyunlar, animasyonlar ve simülasyonlar geliştirmesine olanak verir (Ching vd., 2018). Ergin ve Ercan (2022)'a göre, programlama eğitimi bireylerin olaylar arasındaki ilişkileri görmesini ve üst düzey zihinsel düşünme becerilerini hızlandırmaktadır. Programlama, Matematik ve Fen alanlarının temel kavramlarının öğretilmesinde ve günlük hayata uygulanabilirliğinde önemli bir rol oynamaktadır. Bu yönüyle programlamanın eğitim alanındaki disiplinlerarası rolü kaçınılmazdır (Demirer ve Sak, 2016). Karataş (2021) yaptığı literatür taraması sonucunda programlama eğitiminin diğer faydalarını şöyle sıralamıştır:

- Programlama araçlarının kullanımı ile dijital okuryazarlık geliştirilebilir.

- Hayal gücü ve yaratıcılığı destekler.
- Hem sonuç hem de süreç odaklı düşünmeyi sağlar.
- Motivasyon artmasında katkısı olabilir.
- Öğrencilerin problem çözme becerilerinin gelişmesine olanak tanır.
- Öğrencilerin uzamsal ve analitik düşünme becerilerini geliştirmesine olanak tanır.
- Karmaşık problemlere karşı çözüm üretme pratiği kazandırır.
- Kültürel, iş birliğine yönelik çalışma, yaparak öğrenme becerisi ve öğrenme becerileri geliştirebilir.

Programlama öğrenmenin özelliklerinden biriside iş birliğini içermesi ve sosyal becerilerin gelişmesine olanak vermesidir. Sosyal beceriler, insanların iyi ve etkili ilişkiler kurmasını sağlayan yeteneklerdir. Programlama etkinlikleri esnasında ilerleme sağlayan öğrencilerin geride kalan öğrencilere yardım ederek deneyimlerini paylaşmaları, programlamanın iş birliği yeteneğini geliştirmesine olanak sağladığının bir göstergesidir. Programlamayı öğrenme süreci içinde öğrencilerin diyalog kurmaları ve etkileşime geçmeleri iş birliği açısından önemlidir (Popat ve Starkey, 2019).

2.2.3 Programlama ve Bilgi İşlemsel Düşünme

Bilgi işlemsel düşünme (BİD) genellikle bilgisayar programlama ile karıştırılan kavramlardır fakat farklı etkinliklerdir. Programlama, verilen problemleri analiz etme, problemlere yönelik çözüm üretme ve bu çözümü uygulama etkinliklerini ifade eden bir kavramdır. BİD ise problemleri formüle etmek ve çözmek için dijital teknolojilerin çalışma mantığını kavramayı ifade eder (Kampala's vd., 2023). Wing (2006) tarafından “bilgi işlemsel düşünme becerisi”; bilgisayar bilimlerindeki kavramlardan yola çıkarak sistem tasarlama, problem çözme ve insan davranışlarını anlama olarak tanımlamıştır. BİD erken yaşlardan itibaren okuma, yazma ve matematiğin yanı sıra

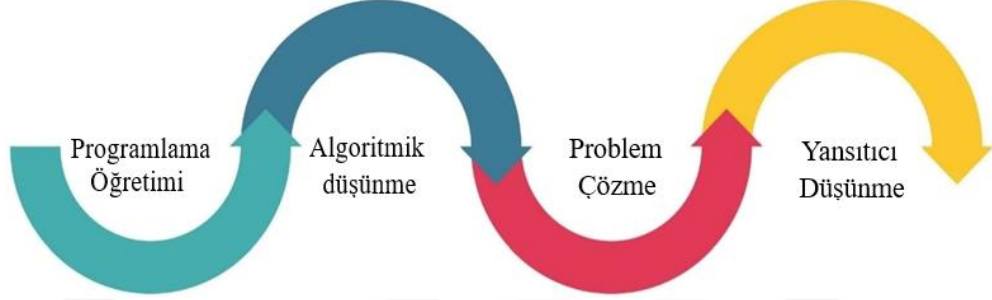
öğrenilmesi gereken temel bir analitik beceridir (Wing, 2006). Ching vd. (2018)'ne göre BİD, öğrencilerin akademik başarısını artırması ve akademiye gerçek hayatla harmanlaması için kritik öneme sahiptir. Programlama kavramı çoğu kez BİD'in unsurlarını da içeren geniş bir tanım içerir ve programlama, BİD'in kullanımını gerektirmektedir (Hsu vd., 2019). K-12 seviyesindeki öğrencilerin BİD becerilerini geliştirmek amacıyla kodlama ve programlama öğrenmek önemli hale gelmiştir (Göncü vd., 2018). BİD genel olarak problem çözme, mantıksal düşünme ve 21. yüzyıl becerileri altında toplanmaktadır (Hsu vd., 2019). BİD, bilgisayar kullanarak problem çözme kapasitesinin artırılması, yaratıcı düşünme, mantıklı düşünme ve eleştirel düşünme becerilerinin geliştirilmesini sağlamaktadır (Korkmaz vd., 2015). Buitrago Flórez vd. (2017), programlama öğretiminde program tasarlama, problemleri çözümleme, yakalanan hataları ayıklama ve benzeri programlama etkinliklerinin BİD gelişimini kolaylaştıran faktörlerden biri olduğunu öne sürmektedir.

Korkmaz vd. (2015) göre BİD, algoritmik düşünme, eleştirel düşünme, işbirlikli düşünme, yaratıcı düşünme ve problem çözme becerilerini de kapsamaktadır.

2.2.4 Programlama, Algoritma ve Algoritmik Düşünme

Çobanoğlu (2009) algoritmayı, bir probleme çözüm üretmek amacıyla izlenecek yol olarak ifade edilmektedir. Diğer yandan Erümit (2020) algoritmayı, yaratıcı ve mantıklı düşünme yaparak bir sürecin adımlarını doğru sıralama olarak ifade etmektedir. Algoritma, çözüme ulaşmak için gerekli adımların belirlenmesinde önemli bir oynamaktadır ve programlamayla birlikte öğretilebilen bir beceridir. Problem çözme ve diğer zihinsel beceriler algoritmik düşünmeye dayanmaktadır (Erümit, 2020). Programlamayı öğrenmek, yeni başlayanlar için birbiriyle etkileşimi olan faktörleri aynı anda dikkate almasını gerektiren karmaşık bir öğrenme sürecidir (Vieira vd., 2017; Espinal vd., 2022). Bu doğrultuda programlama dillerini öğrenmeden önce programlanın mantığını öğrenmek için algoritmanın aracılığı önemli görülmektedir. Çocukların önce algoritmayı öğrenmesi programlamayı da kolayca öğrenmesini sağlamaktadır (Bennedsen ve Carpersen, 2008). Programlama yapan bireyler problemi analiz eder, algoritmik çözüm üretir ve ardından bu algoritmayı program koduna dönüştürür (Saali vd., 2011).

Algoritmik düşünme, problem çözme becerisi ile bütünleşiktir çünkü algoritmik düşünme adımları problem çözme temele almaktadır (Erümit, 2020). Programlama ve bilişsel beceriler arasındaki ilişkiler Erümit (2020) tarafından Şekil 2.1’de gösterilmiştir.



Şekil 2.1 Programlama eğitimi ve bilişsel beceriler arasındaki ilişki (Erümit, 2020)

Algoritmik düşünme sadece programlamayı anlamak ve öğrenmek değildir. Aynı zamanda algoritmayı kullanarak belli bir görevi yerine getirmek için diziler oluşturup takip edebileceği düşünüldüğünde önemli bir yaşam becerisi olarak değerlendirilmektedir. Algoritmaların işleyişini anlamak ve hayatı buna göre düzenlemek, karşılaşılan sorunları basit parçalara ayırarak çözmek, en uygun yolları izleyerek istenilen sonuçlara ulaşmak, sistematik hareket edebilmek hayatta ve karşımıza çıkan sorunları çözebilmemiz için gereklidir (Jančec ve Vujičić, 2021).

2.2.5 Programlama ve Üst Düzey Düşünme Becerileri

Bilgisayar bilimlerinin temel yeterlilikleri arasında problem çözme yer almaktadır. Problemlerin çözümlerini planlamak ve tasarlamak gibi problem çözme stratejileri programlama başarısını etkileyen unsurlardan biri olarak kabul edilmektedir. Programlama yoluyla problem çözmek, bireylerin hem programlama yeteneklerini hem de problem çözme stratejilerini geliştirmesine olanak vermektedir (Chao, 2016). Programlamaya yeni başlayanların problem çözme becerisinden yoksun olması, programlama yoluyla problem çözme istediklerinde programlamanın temel kavramlarını da kullanamadıkları görülmüştür (Veerasamy vd., 2019).

Eleştirel düşünme becerisi programlama öğretiminde kullanılması gereken bir üst beceridir. Öğrenciler, problemlere çözümler üretirken sorgulama yapmak ve çözüm

bulmak için eleştirel düşünme süreçlerini kullanırlar (Bebek, 2022). Eleştirel düşünme, derin ve yaratıcı düşünceler yaparak problemlere yaratıcı çözümler bulmak için öncü bir düşünme türüdür (Doleck vd., 2017). BİD ve programlama becerisi eleştirel düşünmeyi kapsamaktadır. Çünkü programlama, BİD ve eleştirel düşünmenin ortak özellikleri problem çözmeye odaklanmalarıdır (Bebek, 2022).

Yaratıcılık, bireylerin bilgi ve becerilerini kullanarak yeni fikirler, özgün çözümler ve ürünler ortaya koyması anlamına gelmektedir (Haymana ve Özalp, 2020). Yaratıcılık, yeni yollar keşfederek problemleri çözmelerini sağlaması yönüyle programlama için kritik beceri olarak bilinmektedir (Wang vd., 2021). Yaratıcı düşünme problem çözmeyle de yakından ilişkilidir (Şendağ vd., 2023). Yaratıcı düşünme becerisi, algoritma ve programlamayı geliştirmeyi sağlamanın yanı sıra Scratch gibi BTP ortamları da yaratıcılığı geliştirmeye yardımcı olmaktadır (Şendağ vd., 2023). Bu doğrultuda programlama öğrenmenin yaratıcılığı geliştirdiği söylenebilir (Wang vd., 2021), programlama ve yaratıcı düşünme arasında karşılıklı bir ilişki söz konusudur.

2.2.6 Dünya’da Programlama Eğitimi

Bireylerin kariyer seçiminde erken yaşlarda programlama öğrenmiş olması kilit rol oynayacaktır (Ceylan ve Gündoğdu, 2018). Geleceğe yatırım yapmak için her yaştan birey programlama öğrenmeli ve programlama mantığını bilmelidir (Aytekin vd., 2018). Çünkü 21. yüzyılda bireylerden beklenen programlama becerilerine sahip olmasıdır. (Sayın ve Seferoğlu, 2016; Aytekin vd., 2018). Bu yüzden birçok kurum ve kuruluş programlama eğitimi olanakları sunmaya başlamıştır (Sayın ve Seferoğlu, 2016). Bu doğrultuda BİD ve programlama gibi becerilerin eğitim sistemine dahil edilmesine yönelik çalışmalar birçok ülkenin eğitim hedefi olmuştur (Ciftçi ve Bildiren, 2020). Avrupa ülkelerinden, Litvanya, İngiltere, Avusturya, Çek Cumhuriyeti, Danimarka, Bulgaristan, Estonya, İrlanda, Fransa, İspanya, Slovakya, Malta, Polonya, Macaristan ve Portekiz 2015 yılı itibarıyla programlama eğitimi müfredatlarına dahil etmişlerdir (Bocconi vd., 2016; Saygıner ve Tüzün, 2017; Sayın, 2020). Avrupa’daki ülkelerin programlamaya bakış açıları farklıdır örneğin bazı ülkeler programlama becerisini geliştirmeyi hedeflerken bazıları programlama sektöründeki istihdamı desteklemeyi hedeflemektedir (Sayın ve Seferoğlu, 2016).

Programlama ülkelerde farklı isimlerle eğitim programlarında yer almıştır. Örneğin Belçika’da “bilişimsel düşünce ve programlama”, Bulgaristan’da “algoritmik problem çözme ve programlama”, Estonya’da “programlama”, İspanya’da “programlama, algoritma ve robotik” ve İngiltere’de “computing” kavramları kullanılmaktadır (Bocconi vd., 2016).

Programlamanın önemini vurgulamak amacıyla 2013 yılı itibariyle Avrupa Dijital Gündemi Genç Danışmanlar tarafından AB ile ilişkileri olan ülkeler çapında “kod haftası (Code week)” etkinlikleri başlatılmıştır. Kod haftası, programlama yoluyla öğretmen ve öğrencilerin iş birliğini, yaratıcılıklarını ve problem çözme becerilerini geliştirme hareketidir. Kod haftası sayesinde her seviyedeki okullar, tüm branşlardaki öğretmenler ve öğrencileri programlama çalışmaları yapmaktadır (URL-2, 2023).

ABD’de K-12 düzeyinde öğrencilerin bilgisayar bilimleri ve programlamayı öğrenmeleri için “code.org” ile iş birliği sonucunda “herkes kodlama öğrenebilir” temasıyla “Kodlama Saati (Hour of Code)” gibi programlama etkinlikleriyle ön plana çıkmıştır (Sayın ve Seferoğlu, 2016; URL-3, 2023). Code.org sitesi, öğrencilerin bilgisayar bilimlerine kolayca ulaşmalarını sağlayarak programlama ve algoritma becerilerini geliştirmek için olanak sağlamaktadır. Amazon, Facebook, Microsoft, vb. teknolojik kurumlar tarafından desteklenmektedir (Çalışkan, 2020). Code.org sitesi üzerinden farklı ülkelerin dillerinde programlama eğitimleri sunularak ABD dışındaki öğrencilerin programlama eğitimleri almasına olanak verilmektedir (URL-4, 2023).

Diğer ülkeler, bilgisayar programlama ve programlama becerilerinin öğrenilmesiyle ilgili müfredat çalışmalarında için farklı politikalar benimsemiştir. Örneğin Estonya yedi yaşından itibaren öğrencilere programlama öğretmektedir (Tuomi vd., 2017). Japonya 2016 yılında programlama ve bilgi işlemsel düşünme becerilerini içeren, K-12 seviyesinde ilkokullar için 2020, ortaokullar için 2021 ve liseler için 2022’de yeni bir müfredat geliştirmiştir (Hsu vd., 2019). Hindistan’da ilkokul ve ortaokul düzeyinde programlama eğitimi zorunludur. Çin’de programlamanın küçük yaşlardan itibaren okul öncesi dönemde oyun kartlarıyla öğretilmeye başlandığı bilinmektedir (Balanskat ve Englehart, 2014; Saygıner ve Tüzün, 2017). İsrail ve Kanada programlama ilgi derslere lise seviyesinde yer vermektedir (Gülmez, 2019). Avustralya Eğitim

Bakanlığı 5 yaşından itibaren okullarda 2 dönem sürecince temel programlama eğitimi verilmesi planlanmıştır. Güney Kore 2017 yılı itibariyle programlama derslerini tüm kademelerde zorunlu hale getirmiştir (Karataş, 2021).

2.2.7 Türkiye’de Programlama Eğitimi

MEB, 2012 yılında yayınladığı 69 sayılı karar ile “Bilişim Teknolojileri” dersinin adını “Bilişim Teknolojileri ve Yazılım” olarak değiştirmiştir. İsim değişikliğiyle birlikte ders içeriklerinde kodlama vurgusunu ön plana çıkarmıştır. BTY dersi seçmeliyken zorunlu hale getirilmiştir bunun sonucunda ilk defa 5. Sınıftan itibaren öğrencilere kodlama eğitimi vermeye başlamıştır. Değişen BTY dersinin öğretim programlarında “problem çözme”, “programlama” ve “özgün ürün geliştirme” konu başlıkları altında kodlama ile ilgili içeriklere yer verilmiştir (Demirer ve Sak, 2016; Arslan ve Akçelik, 2019). MEB, 2023 yılında ortaokul haftalık ders çizelgesini değiştirmiş 7. ve 8. sınıflara seçmeli olarak “Yapay Zekâ Uygulamaları”, 5. ve 6. sınıflara “Robotik Kodlama” dersleri eklenmiştir (TTKB, 2023). MEB tarafından 2018 yılında hazırlanan BTY dersi öğretim programında öğrencilerin programlama becerilerinin geliştirilmesinin amaçlandığı açık bir şekilde açıklanmıştır. Bu doğrultuda dersin özel amaçları arasında aşağıdaki programlama becerilerine yer verilmiştir (MEB, 2018b):

- Öğrencilerin bilgi işlemsel düşünme, problem çözme beceri kazanmalarını ve bu becerileri geliştirmeleri,
- Öğrencilerin, algoritma tasarımına yönelik uygulama alışkanlığı geliştirmesi, algoritmaları sözel ve görsel olarak ifade etme becerileri,
- Öğrencilerin uygun programlama yaklaşımını kullanarak problemlere çözüm üretmeleri,
- Öğrencilerin, programlama hakkında teknik bilgi sahibi olmaları,
- Öğrencilerin programlama dillerini bilmeleri ve kullanmaları.

Türkiye’de MEB, sivil toplum kuruluşları, teknoloji ve yazılım şirketleri ve diğer kuruluşlar ile birlikte programlama etkinlikleri düzenlemektedir (Eryılmaz ve Deniz, 2019). 2014 yılında Türkiye Bilişimciler Derneğinin ve bazı üniversitelerin katkılarıyla düzenlenen “Bilgisayar Programlama Çocuk Oyuncağı” adlı etkinlikle Scratch, Microsoft Small Basic vb. farklı programlama araçları kullanılarak ilkokul, ortaokul ve lise öğrencilerinin ilk programlarını geliştirmelerini sağlamıştır (Saygıner ve Tüzün, 2017; Karataş, 2021). Çocuklarda erken yaşlardan başlayarak kodlama becerisinin geliştirilmesi, kodlama farkındalığının artması ve iş birliğinin teşvik edilmesi amacıyla Türkiye’de okullar kodlama haftası etkinliklerine katılmaktadır. Kodlama haftası etkinlikleri, 2018 yılı itibariyle Milli Eğitim Bakanlığının üyesi olduğu Avrupa Okul Ağı (European Schoolnet) kuruluşu tarafından yürütülen bir etkinliktir (URL-5, 2023). Türkiye’de kod haftasına katılım her yıl giderek artmıştır (Karataş, 2021). Kodlama haftasının 2022 yılına ait raporlarında Türkiye 19.357 kodlama etkinliği ile 70’i aşkın ülke arasında birinci olarak en çok kodlama etkinliği yapılan ülke olmuştur (URL-6, 2023). Cumhurbaşkanlığı tarafından yayınlanan “2023 Eğitim Vizyonu” belgesinde ilkokul, ortaokul ve lise seviyelerinde, okulda ve okul dışında öğrencilere, öğretmenlere, eğitim yöneticilerine verilecek eğitimlerle kodlama ve tasarım gibi bilişim becerilerinin destekleneceği belirtilmektedir (MEB, 2018a).

2.2.8 Programlama Öğretim Yaklaşımları

Programlama yeterliliği öğrenilen bir beceri olduğundan programlamanın bilgi ve becerilerinin etkili şekilde öğretilmesi amacıyla programlama öğretiminde kullanılacak çeşitli öğretim yaklaşımları ortaya çıkmıştır, ancak bu yaklaşımlar odak noktaları ve başarılı olma durumlarına göre farklılık göstermektedir. Etkili bir programlama öğretiminin programlama bağlamına ve programlama aracına bağlı olduğu bilinmektedir (Scherer vd., 2020). Bilgi ve iletişim teknolojileri ile programlama öğretimi yeni bir öğrenme alanı olduğundan dolayı yöntemleri kesin olarak formüle edilmiş değildir (Mohorovičić ve Strčić, 2011). Bu doğrultuda, öğretmenler, öğretim sürecini öğrenme ortamına, öğrencilerinin farklı yaş gruplarına, özel durum ve koşullarına göre özelleştirebilirler (Bjursten vd., 2023).

Erkoç (2022), programlama öğretim yöntemlerini doğrudan öğrenme, iş birliği öğrenme, durumlu öğrenme ve öz-düzenlemeli öğrenme olarak ifade etmektedir. Mohorovičić ve Strčić (2011) programlama öğretim yöntemlerini, probleme dayalı öğrenme, bulmaca tabanlı öğrenme, eşli programlama, önceden kaydedilmiş derslerle öğrenme, oyun tabanlı programlama ve karma yöntemler olarak belirtmektedir. Garcia (2021) programlama öğretiminde oyun tabanlı öğrenmeyi etkili bulmaktadır, ona göre önemli diğer öğretim stratejileri oyunlaştırma, multimedya araçları ve blok tabanlı ortamlardır. Bazı araştırmacılar tarafından bilgisayar programlama öğretim yöntemleri yönelimlerine göre sınıflandırılmaktadır. Örneğin ifade odaklı, araç kullanma odaklı, yazılım teknolojisi odaklı, görev odaklı, eylem odaklı olmak gibi amaca yönelik sınıflama yapmak söz konusudur (Mohorovičić ve Strčić, 2011). Kert (2018) ise programlama öğretim yaklaşımları odaklandıkları noktalara göre tarafından iki başlık altında toplamıştır ve Tablo 2.2’de gösterilmiştir.

Tablo 2.2 Programlama öğretim yaklaşımları

Öğrenen Odaklı Yaklaşımlar	Araç Odaklı Yaklaşımlar
• Eşli programlama • Oyun temelli programlama • Sorgulama temelli programlama • Proje temelli programlama • Disiplinlerarası Programlama	• Blok tabanlı programlama • Metin tabanlı programlama • Bilgisayarsız programlama • Fiziksel programlama • Hibrit programlama

Programlama öğrenmeye yeni başlayan öğrenciler programlama yeteneği, matematiksel yeteneği, motivasyonu ve tutumu gibi bireysel farklılıkları sebebiyle programlama öğrenmeye karşı zorlanabilirler. Bu doğrultuda programlama derslerinde öğrenme hedeflerine ulaşmak ve öğrencilerin performanslarını ortaya çıkarmak amacıyla araştırmacılar en uygun öğrenme stratejilerini ortaya çıkarmaya çalışmaktadırlar (Garcia, 2021). Garcia (2021)’ye göre uzmanların önerdiği programlama öğretim stratejilerine bakıldığında en çok araştırma bireysel programlama öğretim stratejileri üzerineyken grupla programlama öğretim stratejileri daha az araştırılmıştır. Grupla programlama öğretimi, işbirlikli öğrenme yaklaşımı ile programlama öğretimi üzerine odaklanmaktadır (Garcia, 2021).

Programlama öğretim stratejilerini belirledikten sonraki aşama programa dilini belirlemektir. Programa dillerinin seçiminde öğrenci dostu özelliklere sahip olması veya dil yapısının pedagojik özellikleri önemli rol oynamaktadır (Xinogalos, 2016). Price ve Barnes (2015)'a göre, programlama dilleri metin tabanlı ve görsel tabanlı olmak üzere ikiye ayrılır. Metin tabanlı kodlama dillerinin sözdizimsel yapısının programlamaya yeni başlayanları bilişsel olarak yorması ve kullanıcılara karmaşık gelmesi (Sırakaya, 2018; Espinal vd., 2022) ve yapısal olarak soyut olması (Gomes ve Mendes, 2007) sebebiyle uzmanlar blok tabanlı programlama ortamlarının kullanılmasını önermektedir (Espinal vd., 2022).

2.2.8.1 Eşli programlama

Eşli programlama, programlama öğrenme sürecinde iki öğrencinin yan yana oturması, öğrencilerden birisinin bilgisayarı kontrol etmesi ve kod yazması diğer öğrencinin programlama aktiviteleri için algoritma geliştirmesi, hataları yakalaması ve alternatif çözümleri bulması olarak tanımlanmaktadır (Denner vd., 2021; Erkoç, 2022). İki farklı etkileşim söz konusudur, birincisi bilgisayar ve öğrenciler arasında kurulan etkileşim ikinci olarak öğrenciler arasında kurulan etkileşimdir (Denner vd., 2021). Eşli programlama, iki öğrencinin birlikte çalıştığı işbirlikli bir öğrenme şeklidir (Denner vd., 2014). Eşli programlama pedagojisi işbirlikli öğrenme yaklaşımlarının önemli özelliklerine sahiptir (Han vd., 2009). Programlama öğretiminde, iş birliğine dayalı eşli programlama, öğrencilerin başarısı için önemli bir yöntemdir. Eşli programlama ile programlama öğrenme sürecinde öğrenciler iş birliği yapar ve problemleri eşlerinin yardımı ile çözerler. İki öğrenci yalnızca bir bilgisayarı ortak kullanırlar. Zhong vd (2016)'ne göre eşli programlama öğrencilerin bireysel programlama becerilerini geliştirmekte ve üretkenliği teşvik etmektedir. Eşli programlamada, eşli öğrencilerin kendilerine ait farklı sorumlulukları vardır. Bu sorumluluklar eşlerin birlikte çalışmasıyla planlı yürütülür (Albayrak ve Polat, 2021). Akranlar, öğrenme sürecinde teşvik ve motivasyon kaynağı olabilirler bu yönüyle akranla öğrenme etkilidir (Hartl vd., 2021). Etkili bir iş birliği sürecinde öğrenciler görevleri tartışarak karar alırlar, birbirlerinin düşüncelerinden etkilenirler (Denner vd., 2014). Bireysel öğrenciler eleştirel düşünme faaliyetlerine gerekli önemi vermezken, eşli programlama sürecinde eşler, programlama görevlerini tartışarak bu süreci verimli geçirirler (Rodríguez vd.,

2017). Hanks vd. (2011), programlama kurslarında eşli programlamanın öğrenci başarısına olumlu etkisi olduğuna dair bulgular elde etmiştir.

2.2.8.2 Bilgisayarsız programlama

Bilgisayarsız programlama yaklaşımında bilgisayar bilimi etkinlikleri yaptırılırken bilgisayar kullanılmaz. Bilgisayarsız programlama ile algoritma, veri sıkıştırma ve şifreleme, insan ve bilgisayar etkileşimi gibi konular öğretilmektedir (Ergin ve Ercan, 2022). Karmaşık programlama kavramlarını anlamlandırma ve programlama problem çözme etkinliklerinde bilgisayardaki programlama ortamlarının kullanımı öğrenciler için verimli olmayabilir bu durumlarda bilgisayarsız programlama etkinlikleri kullanılır (Göncü vd., 2020). Bu yaklaşımda öğretmen dijital materyal kullanmamaktadır. Algoritma ve döngüler gibi programlama kavramlarını öğretmek için kalem ve kağıt kullanılır, bunun dışında kartlar, mantık oyunları veya basit vücut hareketleri de kullanıldığı görülmektedir.

2.2.8.3 Fiziksel (Robotik) programlama

Robotik programlama yaklaşımı, öğrencilerin çevresiyle etkileşime geçebilen bir robota kodlar yazması ve yazılan kodların sonucunu somut olarak robotun fiziksel hareketleriyle gözlemlemesine dayanan bir programlama yaklaşımıdır (Göncü vd., 2020). Öğrenciler robot için kod yazdıktan sonra donanımda nasıl çalışabileceğini direkt gözleme fırsatı yakalamaktadır. Programlama eğitimlerinde robotik kodlama, elektronik devreler ile akıllı cihazların oluşumunu sağlaması yönüyle öğrenciler için eğlenceli öğrenme süreci sağlamaktadır (Haymana ve Özalp, 2020). Robotik kodlama eğitimi, akıllı cihazların ve nesnelerin internetinin çalışma mantığının anlaşılabilmesi için bir öğretim ortamı sunmaktadır (Güleryüz, 2022).

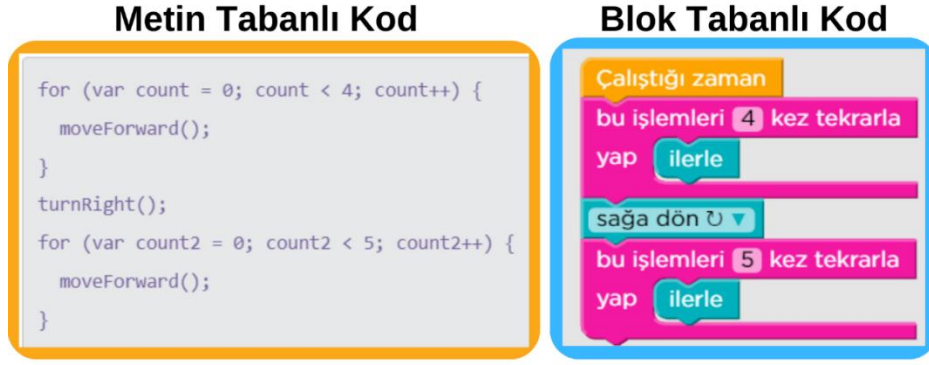
2.2.8.4 Metin tabanlı programlama

Metin Tabanlı Programlama (MTP) dilleri, programlama yaklaşımları arasında en eski olanıdır (Göncü vd., 2020). C++, Fortran, Cobol, VBasic, Java vb. klasik programlama dilleri ile komut satırları yazarak program oluşturmaktır (Ergin ve Ercan, 2022). Öğrencilerin programlama yapabilmesi için programlama dilinin söz dizimini ve söz

dizimleriyle alakalı temel kavramları bilmelidir (Göncü vd., 2020). Sözdizimi programlama dilinin temel dilbilgisi kurallarıdır, bu dilbilgisi Türkçe ve İngilizce dillerinin kurallarına benzemektedir (Özel, 2019). MTP ortamları ileri seviyede programlama yapmak için olanak sağlarlar (Göncü vd., 2020).

2.2.8.5 Blok tabanlı programlama

Blok Tabanlı Programlama (BTP), sözdizimi kurallarına göre kod yazmaya gerek kalmadan, ekrandaki görsel nesnelerin sürükleyip bırak yöntemiyle kodlama işlemi gerçekleştirilebilmeyi sağlayan programlama ortamlarıdır (Yıldız Durak, 2020). Geleneksel metin tabanlı programlama dillerinin yeni öğrenenlere karmaşık gelmesi sebebiyle arayüzünün kolay olması ve kodlama görevlerini kolaylaştırması gibi faktörlerden dolayı görsel özelliklere sahip olan scratch, code.org, blokly gibi programlama ortamları kullanılmaktadır (Aytekin vd., 2018). BTP ortamları, öğrencilerin programlama komutlarını yapboz parçalarıyla oluşturmaya olanak sağlamaktadır (Lin ve Weintrop, 2021). BTP ortamlarında kullanıcılar yapboz parçaları şeklindeki kodları sürükleyip bırak yöntemiyle bir araya getirir ve projeler üretir (Totan, 2021). Blok tabanlı ortamlarda, farklı özellikleri temsil eden kod parçaları farklı renklerle gösterilmektedir (Moors vd., 2018). Kodların şekli, rengi ve görsel ipuçları, blokların nasıl bir araya getirilebileceği konusunda öğrencilere rehberlik sağlar (Lin ve Weintrop, 2021). Blok tabanlı ortamda, çocuklar için kod blokları kolayca erişilebilir ve kategoriler halinde düzenleyebilir olduğu için öğrenciler ezberlemeye ihtiyaç duymamaktadır (Moors vd., 2018). Sürükleyip bırak yaklaşımının öğrenmede bireyleri yaratıcı düşünmesini sağlamaktadır ve metin tabanlı programlama dilleri yapısı itibarıyla yaratıcı düşünmeyi sağlamada yetersiz kalabilir (Noone ve Mooney, 2018). Görsel tabanlı ortamların çocukların düşünme gelişimini destekleyebileceğini ve çocuklara programlama öğretimi için daha uygun olduğunu göstermiştir (Kalelioğlu, 2015; Sun vd., 2023). BTP, öğrencilerin MTP deki sözdizimleriyle uğraşmadan program oluşturmaya odaklanmalarını sağlar (Hill, 2015). Aynı görevi yerine getiren metin tabanlı ve blok tabanlı kodların karşılaştırılması aşağıdaki Şekil 2.2’de gösterilmiştir.



Şekil 2.2 MTP ve BTP kodlarının karşılaştırılması (URL-3, 2023)

2.3 Öz-Yeterlik

2.3.1 Öz-Yeterliğin Tanımı ve Öz-Yeterlik İnancı

Öz-yeterlik sosyal bilişsel yaklaşıma dayalı olarak kişinin davranışları, kişisel düşünceler ve inançlar ve çevresel koşulların etkileşimine bağlı olarak ortaya çıkmıştır (Bandura, 1997). Bandura (1997)'ya göre öz-yeterlik, bireyin kendi yeteneklerine olan inancı olarak ifade edilmektedir. Öz-yeterlilik inancı bireylerin görev seçimini, çabasını, sebatını, dayanıklılığını ve başarısını etkileyen bir faktördür (Pajares ve Schunk, 2001). Öz-yeterliği yüksek bireyler eylemlerini gerçekleştirebilirken öz-yeterliği düşük olan bireyler eylemleri gerçekleştirmeleri ihtimali düşük kalmaktadır (Arseven, 2016). Öz-yeterliği düşük bireyler zorluklardan kaçma eğilimindedirler, hedefe bağlanmakta zorlanırlar (Wilson ve Narayan, 2016). Öz-yeterlik için bireylerin davranışın sonucuna beklentisi ve inançları önem taşımaktadır (Arseven, 2016). Öz-yeterlik bireylerin kendinden beklenen performansı ortaya çıkarması için gerekli motivasyonu ifade edebilir (Wilson ve Narayan, 2016). Öz-yeterlik düzeyi yüksek olması öğrencilerin başarısını etkilemektedir (Yılmaz, 2019). Çünkü öğrencilerin akademik performansının, başarının ve özgüveninin artması ile ilişkilidir. Bu bağlamda yüksek öz-yeterliğe sahip öğrencilerin daha fazla sorumluluk almasını beklenmektedir (Komarraju ve Nadler, 2013).

Öz-yeterlik çok yönlü bir faktördür bu sebeple belirli görev ve bağlamlara göre ölçülmesi gereklidir. Bireylere verilen görevler öz-yeterliği etkiler ve hangi faktörlerin etkilediği bilinmelidir (Wilson ve Narayan, 2016). Öz-yeterlik dış faktörlerden etkilenebilen bir yapıdadır bu yüzden öğrencilerin öz-yeterliklerinin geliştirilmesi için

öğrenme ortamlarının uygun şekilde düzenlemesi yerinde olacaktır. Öğrenme ortamlarında öğretmenlerin kullandığı stratejiler ile öz-yeterlik arasında olumlu yönde ilişki olduğu bilinmektedir. Öğretmenlerin öğrencilerine seçim hakkı sunması öğrencilerin öz-yeterliliğini yükselten faktörlerdendir çünkü bireylerin öz-yeterlik algılarını yükselten en önemli etkenlerden birisi öğrencinin geçirdiği yaşantılardır (Sakız, 2013). Akademik başarı için üç farklı öz-yeterlik türü önemli görülmektedir bunlar öğrenci, öğretmen ve okul öz-yeterliğidir. Öğrencilerin öz-yeterlikleri, üst bilişsel stratejileri kullanarak bilişsel ve öz-düzenleme becerilerini etkileme özelliğine sahiptir (Arseven, 2016).

2.3.2 Programlama Öz-Yeterliği

Bilgisayar ve programlama becerilerinin geliştirilmesinde öz-yeterlik kavramı önemli bir etken olarak görülmektedir (Çiftçi vd., 2018). Programlama eğitiminde öğrencilerin öğrenme deneyim ve yaşantıları programlamaya karşı olan inancı üzerinde etkilidir. Düşük programlama öz-yeterliğine sahip bireyler programlama etkinliklerinde daha fazla hatalarla karşılaşmaktadır ve güçlük çekmektedir (Ekici ve Çınar, 2020). Programlama dersinin üst düzey beceriler gerektirmesi ve öğrencilerin cinsiyet, tutum, motivasyon gibi çeşitli durumları öğrencilerin öz-yeterliklerini etkilemektedir. Bu doğrultuda öğrencilerin programlama derslerini zor olarak algılamaları, öz-yeterlik inançlarının düşük olmasından dolayıdır çünkü öğrencilerin programlamaya yönelik olumlu ve olumsuz algıları öğrencilerin performansını etkilemektedir (Kasalak ve Altun, 2017). Öğrencilerin programlamada başarılı olması için programlamaya ilişkin öz-yeterlik inançlarının yüksek olması gerekmektedir (Yılmaz, 2019). Programlama öğrenme sürecinin başlangıcında öğrencilerin programlamaya karşı ön bilgi ve ön deneyimleri programa öz-yeterliği için önemli bir etkindir (Mazman ve Altun, 2013).

Programlama öğrenme sürecinde öğrencilerin yeterli motivasyona sahip olmaması başarısız olma ve dersi bırakma risklerini içermektedir. Geleneksel programlama öğretim teknikleri programlama öğretiminde motivasyonu artırmada etkisiz kalabilmektedir. Bu bağlamda programlama öğretiminde motivasyonu artırmak ve ilgi

çekici hale getirmek amacıyla işbirlikli ve öğrenci merkezli yaklaşımların kullanılması yerinde olacaktır (Erkoç, 2022).

2.4 Öz-Yeterlik ve Öz-Düzenleme İlişkisi

Pamuk ve Elmas (2015) göre öz-yeterlik düzeyi yüksek olan bireylerin kendine olan inancı ve güveni öz-düzenleme becerilerinin gelişimine katkı sağlamaktadır. Öz-yeterlikleri yüksek bireylerin görev ve sorumluluk alma bilinci de gelişmiştir. Yüksek öz-yeterliğe sahip öğrenciler duygularını daha iyi yöneten, sorumluluk bilinci olan ve çalışkan özelliklere sahiptir. Bu nedenle yüksek öz-yeterliğe sahip öğrenciler düşük öz-yeterliğe sahip öğrencilere göre öz-düzenleme becerilerini daha etkin kullanma eğilimindedirler (Dilekli vd., 2023). Gündoğdu ve Çelik (2020), öz-düzenleme ve öz-yeterlik becerisi arasında güçlü bir bağlantı olduğunu ifade etmektedir. Öğrencilerin öz-yeterlikleri düzeyleri, bilişsel süreçlerini ve öz-düzenleme becerilerini yordama gücüne sahiptir. Öz-yeterlik ve öz-düzenleme arasındaki bu bağlantı sayesinde öğrencilerin öz-düzenleme stratejilerini kullanması akademik performansının da artmasını sağlamaktadır (Gündoğdu ve Çelik, 2020). Dilekli vd. (2023), öğretmen adayı öğrencilerin öz-düzenleme becerileri ve öz-yeterlik algıları arasında pozitif yönlü bir ilişki bulmuştur.

2.5 Öz-Düzenleme

2.5.1 Öz-Düzenlemenin Tanımı

Öz-düzenleme Bandura (1986) tarafından sosyal bilişsel kuramla ortaya çıkarılmıştır. Öz-düzenleme kapasitesi, bireylerin kendi davranışlarına ve yeteneklerine odaklanması anlamına gelmektedir. Öz-düzenleme becerisi, öğrencilerin kendi öğrenme hedeflerini belirlemesi, hedeflere ulaşmak için strateji geliştirmesi ve bu stratejileri denetlemesi olarak ifade edilebilir (Zimmerman, 1989). Pintrich (1995) öz-düzenlemeyi, bireylerin hedefe yönelmek için davranışlarını, bilişini ve güdülenmesini kontrol etme süreci olarak tanımlamaktadır. Bir başka tanıma göre ise öz-düzenleme, hedefe odaklanma ve dikkatini sürdürme kapasitesidir. Öz-düzenleme becerisi kazanmış bireyler sorumluluklarını iyi bilir, kendi hayatlarını düzenler ve kontrol ederler. Öz-düzenlemeye sahip öğrenciler öğrenme sorumluluğunu alır, kendi

öğrenme süreçlerinden en fazla verimle yararlanırlar (Gürer vd., 2019). Diğer yandan öz-düzenleme, öğrencilerin bilişlerini, motivasyonlarını ve davranışlarını yönetmesidir (Öztürk, 2022).

Yukarıdaki tanımlar incelendiğinde öz-düzenlemenin; bireylerin hedeflerini kendi belirmesi, bu hedefleri gerçekleştirmek için kendi davranışlarını, sorumluluklarını ve motivasyonlarını yönetme kapasitesi olarak tanımlanabilir.

2.5.2 Öz-Düzenleme Becerisi

Öz-düzenleme becerisinin çocukluk ve ergenlik döneminde kazanıldığı ve süreç içinde yaş aldıkça artarak geliştiği bilinmektedir. Öz-düzenleme becerisinin gelişmesinde aile, çevre, öğretmen ve okul etkilidir. (Aydın ve Atalay, 2015). Öz-düzenleme becerisi gelişen öğrencilerin öğrenme sürecine daha aktif katıldığı ve verilen görevleri yerine getirmede daha fazla çaba gösterdiği bilinmektedir (Eker, 2015). Öz-düzenleme becerisi kazanmış bireyler öğrenme sürecinde aile ve öğretmen gibi eğitim paydaşlarına güvenmek yerine kendisine güvenmeyi tercih etmektedir. Bu bağlamda bireyler güçlü ve güçsüz yönlerinin farkındadır diğer yandan kendi başarı ve başarısızlığının sorumluluklarını kendileri alırlar (Aydın ve Atalay, 2015). Ayrıca, öz-düzenleme becerilerini aktif olarak kullanabilen öğrencilerin çeşitli öğrenme stratejileri de kullandıkları ve öğrenme etkinliklerine katılmaya daha istekli oldukları bilinmektedir. Bu doğrultuda öğrencilerin başarılı olmasının altında yatan nedenlerden birinin öz-düzenleme becerisini aktif kullanmak olduğu görülmektedir (Feng ve Chen, 2014). Gürer vd. (2019)'ne göre öğrencilerin öz-düzenlemeli öğrenme stratejileri, motivasyonel inançları ve ders başarısı arasında da olumlu ve anlamlı yönde bir ilişki vardır. Bu doğrultuda öğrencilerin öz-düzenleme becerilerini destekleyecek etkinliklere ve öğrenme ortamlarına yer verilmelidir. Eğitim ortamlarını düzenleyen öğretmenler öğrencilerin öz-düzenleme becerisinin gelişmesinde önemli bir role sahiptir. Öğretmenin öğretim sürecinde öz-düzenleme stratejileri kullanması, öğretim yöntemleri, sınıf ortamını öz-düzenleme becerilerini geliştirici olarak düzenlemesi ve öğretmenin kişisel özellikleri öğrencilerin öz-düzenleme becerilerinin gelişmesine olanak sağlamaktadır (Aydın ve Atalay, 2015). Aydın ve Atalay (2015)'in literatür

taraması sonucunda öz-düzenlemeyi geliştiren unsurların arasında işbirlikli öğrenme ve teknoloji kullanımının olduğu görülmektedir.

2.5.3 Öz-Düzenleme Becerisi ve Programlama

Öz-düzenleme, bireyin kendi karmaşık öğrenme süreçlerini, bilişlerini, motivasyonları ve davranışlarını değerlendirmesidir (Çiltaş, 2011). Programlama dersleri öğrencilerin motivasyonlarını artıracak çeşitli öğrenme faaliyetlerini içerecek şekilde tasarlanmalıdır. Motivasyon ve öz-düzenleme arasında olumlu bir ilişki olduğundan dolayı öğrencinin akademik performansını da pozitif yönde etkilemektedir (Öztürk, 2022). Öz-düzenleme becerisine sahip tüm bireyler kendi öğrenme süreçlerinin her aşamasını tasarlar, değerlendirir ve düzenler (Özmenteş, 2008). Bilgisayar programlama sürecinin yapısı itibarıyla öz-düzenleme ile benzerlikleri vardır. Öz-düzenleme süreci planlama, izleme, kontrol ve değerlendirme adımlarından oluşmaktadır. Buna benzer bir süreç programlama yapan bireyler tarafından da kullanılmaktadır (Çoşğun ve Çoşğun, 2018). Öğrenciler programlama yaparken kendi hatalarını bulabilir, kendi kendini izleyebilir, değerlendirebilir ve düzeltebilirler. Diğer yandan, öz-düzenleme yaklaşımı bireylerin hedeflerini çevresel faktörlere göre yapılandığı, planladığı ve hedeflere ulaşma doğrultusunda motivasyonlarını ayarladığı aktif bir süreçtir. Bu doğrultuda programlama dili öğrenme süreçlerinin öz-düzenleme ile ilişkili olduğu söylenebilir. Programlama dillerini öğrenmenin bireysel düşünme ve uygulama hazırlama çabası bireyin öz-düzenleme becerilerini geliştirdiği düşünülmektedir (Kert ve Uğraş, 2009). Kert ve Uğraş (2009)'a göre öz-düzenlemenin bireylerin kendini gözlemlemesi, kendini yargılaması ve eyleme geçmesi adımlarının programlamanın alt boyutlarıyla bağlantılı olduğu da bilinmektedir. Çünkü programlama bireyin zihinsel süreçlerini yapılandırmaktadır (Kert ve Uğraş, 2009). Haşlaman ve Aşkar (2007), programlama dersi ile ilgili öz-düzenleme stratejileri ile başarıları arasındaki ilişkileri araştırmış sonuç olarak öz-düzenleme stratejilerinin başarının %71'ini açıkladığını ortaya koymuştur.

2.6 Öz-Düzenleme ve Akademik Başarı İlişkisi

Öz-düzenleme öğrencilerin kendi öğrenme süreçlerini yönettiği, bilişsel ve üstbilişsel stratejileri aktif kullandığı bir beceridir. Öz-düzenleme stratejilerini etkin kullanan öğrencilerin akademik başarıları da yüksektir. Öz-düzenlemeye sahip öğrenciler öğrenme süreci içinde sürekli aktiftirler, kendi öğrenme hedeflerini, motivasyonlarını ve stratejilerini belirlerler (Cabı, 2015). Tülübaş (2022)'a göre öz-düzenleme becerisi ve akademik başarı arasında pozitif yönlü bir ilişki vardır. Diğer yandan öz-düzenleme, öğrencilerin yaşam boyu öğrenme becerilerini geliştirmekte ve kariyerindeki başarısını artırmaktadır (Tülübaş, 2022). Temel vd. (2023), ilkökul dördüncü sınıf öğrencilerinin akademik başarı algıları ve öz-düzenleme becerileri arasında pozitif yönlü ve düşük düzeyde bir ilişki olduğunu savunmaktadır. Bu doğrultuda, öz-düzenleme becerisi yüksek öğrenciler motivasyonlarını, bilişlerini ve hedeflerini yöneterek daha etkin bir öğrenme gerçekleştirmektedir ve yüksek akademik başarı yakalamaktadır (Temel vd., 2023).

2.7 Akademik Başarı

Akademik başarı, okulda uygulanan öğretim programlarındaki kazanımların öğrenciler tarafından kazanılıp kazanılmadığını ortaya çıkarmak amacıyla öğrencilere uygulanan ölçme ve değerlendirme etkinlikleri ve öğrencilerin aldığı notlar olarak ifade edilmektedir (Erdoğan, 2006; Bahçetepe ve Giorgetti, 2015).

Bahçetepe ve Giorgetti (2015)'ye göre, akademik başarı etkileyen çeşitli etkenler vardır. Bunların başında anne ve babanın sosyoekonomik düzeyi ve tutumları gelmektedir. Erdoğan (2006)'ya göre ise akademik başarıyı etkileyen etkenlerin başında öğrenci-öğretmen ilişkileri yer almaktadır. Öğretmenin sınıf ortamı içinde öğrencilere karşı olan “demokratik” tutumu öğrencilerin akademik başarılarını, güdülerini, motivasyonu artırırken, “otoriter” tutumu ise azaltmaktadır (Erdoğan, 2006). Diğer yandan okulun fiziki özellikleri ve okulun kültürel yapısı öğrencilerin davranışlarını etkilemektedir (Bahçetepe ve Giorgetti, 2015).

2.8 İşbirlikli Öğrenme, Akademik Başarı, Öz-Yeterlik ve Öz-Düzenleme İlişkisi

İşbirlikli öğrenme, öğrencilerin öz-yeterlik, akademik başarı, tutum, sosyal beceriler gibi çeşitli becerilerini olumlu etkileyen bir öğrenme modelidir. Yılmaz (2019) çalışmasında birlikte öğrenme tekniği ile yapılan scratch öğretiminde öğrencilerin scratch öz-yeterliklerini ve akademik başarılarını olumlu etkilediği görülmüştür. Öz-yeterlik, öğrencilerin akademik performanslarını etkileyen önemli bir duyuşsal özelliktir (Köseoğlu, 2010). Çünkü öz-yeterlik öğrencinin kendine olan inancını ifade etmektedir (Bandura, 1997). Gülten ve Soytürk (2013) 6.sınıf öğrencilerinin geometrik öz-yeterliği ile akademik başarı not ortalamaları arasındaki ilişki olduğunu bulmuşlardır. Güvenç (2010), işbirlikli öğrenme ve günlük kullanımının öğretmen adaylarının öz-yeterlik ve öz-düzenlemeli öğrenme becerilerinin organizasyon, meta-bilişsel, eleştirel düşünme, kontrol stratejisi, performans ve ayrıntılandırma gibi stratejiler üstünde olumlu etkisini bulmuştur. Taşkapı (2015), öğretmen adaylarının öz-düzenleme becerileri ile işbirlikli öğrenme stilleri arasında pozitif yönde anlamlı bir ilişki olduğunu savunmaktadır. Demircan (2014), akademik başarı düzeyi yüksek öğrencilerin, öz-düzenleme becerileri, öz-yeterlik algıları, bilişsel stratejileri kullanım değerlerinin de yüksek olduğunu söylemektedir. Dadlı (2015), 8. Sınıf öğrencilerinin öz-düzenleme becerileri ve öz-yeterlikleri ile akademik başarıları arasındaki ilişkileri incelediği çalışmasında öz-düzenleme ve akademik başarı arasında düşük düzeyde, pozitif ve anlamlı bir ilişki; öz-yeterlikleri ile akademik başarıları arasında ise orta düzeyde, pozitif ve anlamlı bir ilişki olduğunu bulmuştur.

2.9 İlgili Bilimsel Çalışmalar

Bu başlık altında programlama, işbirlikli öğrenme, programlama öz-yeterliği ve öz-düzenleme becerisine yönelik yapılan bilimsel araştırma sonuçları yer almaktadır.

2.9.1 Yurt İçinde Yapılan Çalışmalar

Öztürk (2022), öz-düzenlemeli programlama eğitiminin lisans öğrencilerinin akademik başarısına ve motivasyonuna etkilerini incelemiştir. Çalışma grubu 15 deney ve 16 kontrol grubunda olmak üzere 31 kişiden oluşmaktadır. Veriler akademik

performans testleri, programlama motivasyon ölçeği ve görüşme formu aracılığıyla toplanmıştır. Buna ek olarak öğrencilerle görüşme gerçekleştirilmiştir. Araştırma bulgularında öz-düzenlemeli programlamanın geleneksel yöntemlere göre öğrencilerin akademik performansına ve motivasyonuna katkı sağlamaktadır. Öğrencilerle yapılan görüşmeler sonucunda ise öz-düzenlemeli öğrenme stratejilerinin öğrencilerin akademik performans ve motivasyonlarına olumlu katkısı olduğu belirtilmiştir (Öztürk, 2022).

Demir ve Seferoğlu (2021)'in işbirlikli öğrenmenin programlamaya yansımaları olan eşli programlamayı, tekli programlama ile kodlama başarısı, akış deneyimi ve kodlama kalitesi bakımından karşılaştırmıştır. Bu çalışmada kullanılan yöntem ön test-son test yarı deneysel desendir. Çalışma grubu 42 gönüllü son sınıf üniversite öğrencisinden (28 erkek, 14 kadın) oluşmaktadır. Kontrol grubuna tekli programlama ve deney grubuna ise eşli programlama yaklaşımı ile programlama öğretimi yapılmıştır. 6 haftalık uygulama sürecinin dört haftasında akış deneyimi bakımından deney grubu, kontrol grubundan daha yüksektir. Deneyin üç haftasında ise kodlama kalitesi bakımından deney grubu, kontrol grubuna göre daha yüksektir. Çalışmada deney ve kontrol grubunun kodlama başarı artmasına rağmen deney grubu lehine bir farklılık olmamıştır. Sonuç olarak işbirlikli öğrenmenin programlamaya yansımaları olan eşli programlama akış deneyimi ve kodlama kalitesi bakımından başarılı bulunmuştur (Demir ve Seferoğlu, 2021).

Aslan ve İşbulan (2021)'nin yaptığı çalışmada bireysel ve takım olarak yapılan etkinliklerin BTP öz-yeterliği ve robotik programlama tutumlarına etkisini araştırmıştır. Bu çalışmanın örneklemini 2018-2019 eğitim öğretim yılında İstanbul'un Sarıyer ilçesindeki bir özel okuldaki 32 yedinci sınıf öğrencidir. Çalışmada yarı deneysel desen kullanılmıştır. Bu çalışmanın bulgularına göre bireysel çalışma ve ekiple çalışma BTP öz-yeterliklerini ve robotik programlama tutumlarını etkilememiştir. Fakat hem bireysel çalışmaların hem ekiple çalışmaların blok tabanlı programlama ve robotik programlama tutumlarında anlamlı bir fark olduğu görülmüştür (Arslan ve İşbulan, 2021).

Tiryaki (2020) tez çalışmasında ortaöğretim öğrencileri ile robotik kodlama eğitiminin öğrencilerin programlama öz-yeterlik düzeylerini ve yaratıcı düşünme becerilerini etkileme durumunu araştırmıştır. Bu çalışmanın çalışma grubunu, 2019-2020 eğitim öğretim yılında Hatay ilinin Antakya ilçesinde bulunan bir ortaöğretim kurumunda robotik kodlama eğitimi gören 71 öğrenci oluşturmaktadır. Deney grubu öğrencileri araştırma süreci boyunca robotik kodlama eğitimleri verilirken kontrol grubunda yer alan öğrencilere öğretim programında yer alan eğitimler verilmiştir. Araştırmada elde edilen bulgulara göre öğrencilere uygulanan robotik kodlama eğitimlerinin öğrencilerinin programlama öz-yeterlik inançlarına arttırdığı ve yaratıcı düşünme becerilerini geliştirdiği ortaya çıkmıştır (Tiryaki, 2020).

Okal vd. (2020)'nin yaptığı kodlama eğitiminin 5., 6. ve 7. sınıf öğrencilerinin programlama öz-yeterliklerine ve teknolojiye karşı tutumlarına etkisi araştırılmıştır. Araştırmanın çalışma grubunu 2018-2019 eğitim-öğretim yılında bir ortaokulun 5, 6 ve 7. Sınıf 64 öğrenci oluşturmaktadır. Araştırmanın verileri programlama öz-yeterliği ölçeği, teknoloji tutum ölçeği ve yarı yapılandırılmış görüşme formu ile elde edilmiştir. Sonuçlara bakıldığında, kodlama eğitiminde geliştirilen modelin öğrencilerin programlama öz-yeterliklerine ve teknolojiye karşı tutumlarında olumlu etkisi olduğu gözlemlenmiştir. Öğrencilerin kodlama eğitimine ilişkin olumlu görüş belirttikleri tespit edilmiştir (Okal vd., 2020).

Çal ve Can (2020)'in ortaokul beşinci sınıf öğrencileri üzerinde grupla ve bireysel programlama öğretimi yaptığı çalışmada beşinci sınıf seviyesinde toplam 35 öğrenci, 13 kişi bireysel ve 22 kişi eşli programlama gruplarına ayrılmış şekilde blok tabanlı kodlama platformu olan scratch ile sekiz haftalık çalışmalar yapılmıştır. Eşli programlama öğretiminin öğrencilerin öz güveni başarısına etkisinin araştırıldığı çalışmada grupla programlama yapılan öğrencilerin öz güven ve başarısı bireysel programlama öğretimi yapılanlara göre daha yüksek bulunmuştur. Bilgisayar sayısının yetersiz ve rekabetin yüksek olduğu okullarda araştırmacılar bu yöntemi önermektedir (Çal ve Can, 2020).

Tosuntaş vd. (2020)'nin kodlama eğitiminde işbirlikli öğrenme yaklaşımı (eşli kodlama) üzerine araştırma yapmıştır. Çalışma grubunu 220 ortaokul öğrencileri

oluşturmaktadır. Araştırmanın nitel verileri altı sorudan oluşan yarı yapılandırılmış bir görüşme formu ve odak grup görüşmeleri yoluyla toplanmıştır. Öğrencilerin kodlamaya yönelik tutumları ile işbirlikli öğrenmeye yönelik tutumları arasında pozitif yönlü korelasyon olduğu tespit edilmiştir. Öğrencilerin eşli kodlamaya yönelik işbirlikli öğrenmenin olumlu etkileri görülmüştür. Sonuç olarak kodlamanın öğretiminde işbirlikli öğrenme örneği olan eşli kodlama ile öğrenmesinin avantajlı görülmektedir (Tosuntaş vd., 2020).

Özenoğlu ve Birişçi (2019)'nin grupla programlama öğretiminin yaratıcı düşünme becerileri üzerindeki etkisinin ortaya çıkarılmasının amaçlandığı araştırmada özel bir okulda Robotik ve Kodlama dersi kapsamında Lego Mindstorms Ev3 seti ile öğrenim görmekte olan 17 beşinci sınıf öğrencisi üzerinde altı hafta çalışılmıştır. Tek gruplu öntest-sontest deneysel çalışma yapılmış, elde edilen test puanları ve görüşmeler sonucunda grupla yapılan programlama eğitiminin yaratıcılığı olumlu yönde etkilediği sonucuna ulaşılmıştır (Özenoğlu ve Birişçi, 2019).

Yılmaz (2019), Scratch uygulamasının öğretiminde işbirlikli öğrenme tekniğinin kullanılmasının öğrencilerin akademik başarılarına ve öz-yeterlik algılarına etkisini araştırmıştır. Tez çalışmasının temel amacı, tasarlanan öğretim programının uygulandığı deney grubu ile geleneksel öğretim programının uygulandığı kontrol grubu öğrencileri arasındaki akademik başarı ve öz-yeterlik düzeyleri arasındaki farklılıkları ortaya koymaktır. Araştırmanın çalışma grubunu 2018-2019 eğitim öğretim yılında İstanbul ili Esenyurt ilçesinde bulunan bir ortaokulda öğrenim gören altıncı sınıf öğrencileri oluşturmaktadır. Çalışma, deney grubunda 28 öğrenci ve kontrol grubunda 28 öğrenci olmak üzere toplam 56 öğrenci ile yürütülmüştür. Araştırmanın sonuçlarına göre, scratch programına yönelik öz-yeterlik algısı ile akademik başarı notu arasında aynı yönde pozitif bir ilişki olduğu tespit edilmiştir. İşbirlikli öğrenme tekniği ile yapılan eğitimin öğrencilerin scratch programına yönelik öz-yeterlik algısını ve akademik başarısını olumlu yönde etkilediği ortaya çıkmıştır. Ancak öz-yeterlik algısı ile akademik başarı notu arasında anlamlı bir farklılık olmadığı tespit edilmiştir (Yılmaz, 2019).

Mıhçı Türker ve Pala (2019) öğrencilerin BİD becerileri ve blok tabanlı programlama öz-yeterliklerini araştırmışlardır. Araştırmanın çalışma grubunu Türkiye’deki bir ilde yer alan 5, 6, 7 ve 8. sınıflarında bulunan 464 öğrenci oluşturmaktadır. Araştırma verileri “Bilgi İşlemsel Düşünme Becerileri Ölçeği” ve “Blok Tabanlı Programlamaya İlişkin Öz-Yeterlik Algısı Ölçeği” ile elde edilmiştir. Araştırmanın sonuçlarına göre cinsiyetin, programlama eğitiminin, ders dışı programlama çalışmalarının, bilgi işlemsel düşünceleri ve blok tabanlı programlama öz-yeterliklerinde anlamlı farklılıklar görülmüştür (Mıhçı Türker ve Pala, 2019).

Çoşğun ve Çoşğun (2018), yaptığı araştırmada programlama öğretiminin ortaokul öğrencilerinin öz-düzenleme stratejileri ve motivasyonel inançları üzerindeki etkisini araştırmıştır. Bu araştırmanın çalışma grubunu MEB’e bağlı ortaokullarda öğrenim gören 130 öğrenci oluşturmaktadır. Çalışmada desen olarak tek gruplu öntest ve sontest kullanılmıştır. Öğrencilere 16 hafta boyunca BTY dersi programlama ünitesi konularının öğretimi sürecinin öncesinde ve sonunda öğrencilere uygulanan veri toplama aracından elde edilen bulgulara göre programlama öğretiminin ortaokul öğrencilerinin, bilişsel strateji kullanımı ve öz-yeterlik, öz-düzenleyici öğrenme becerileri, içsel değer ve sınav kaygısı üzerinde farklılık oluşturmadığı sonucuna ulaşılmıştır (Çoşğun ve Çoşğun, 2018).

Yüksel (2017) Scratch programı öğretiminde işbirlikli öğrenme tekniği olan ayrılıp-birleştirme tekniğinin kullanılmasının öğrencilerin derse karşı tutumları, akademik başarıları ve kalıcılıkları üzerindeki etkisini araştırmıştır. Çalışmada öntest-sontest kontrol gruplu yarı deneysel desen kullanılmıştır. Araştırma 2015-2016 eğitim-öğretim yılında 60 ortaokul 6. sınıf öğrencisinin öğretim sürecinde yapılmıştır, 30 kontrol ve 30 deney grubu öğrencisi bulunmaktadır. Kontrol grubunda eğitim programında var olan teknikler kullanılmış deney grubunda ise işbirlikli öğrenme tekniklerinden biri olan “ayrılıp birleşme” kullanılarak öğrencilerin tutumlarına ve akademik bilgilerinin kalıcılığına etkisi ortaya çıkarılmıştır. BTY dersi Scratch ile programlama ünitesi konularının yedi hafta süreyle ayrılıp birleşme tekniği ile yapılmıştır. Araştırma sonuçlarına göre ayrılıp birleşme tekniğinin öğrencilerin ders başarısını, derse yönelik tutumu ve öğrenilen bilgilerin kalıcılığını olumlu yönde etkilediği tespit edilmiştir (Yüksel, 2017).

Türkiye’de yapılan araştırmalar incelendiğinde, öz-düzenleme ve programlamanın birlikte değerlendirildiği çalışmaların az olduğu göze çarpmaktadır. Çalışmaların sonuçları ele alındığında ortaokul düzeyinde programlamanın öz-düzenlemeye etkisi bulunmadığı görülmektedir. Türkiye’de programlama üzerine yapılan araştırmaların programlama-öz-yeterliği becerisine yoğunlaştığı görülmektedir. Programlama öz-yeterliği üzerine yapılan araştırmaların ise olumlu etkisi görülmektedir. İşbirlikli öğrenmenin programlama öğretimine yansımaları olan eşli programlamanın ve yine bir işbirlikli öğrenmenin tekniği olan ayrılıp-birleşme tekniği ile araştırmaların çoğunlukta olduğu görülmektedir.

2.9.2 Yurt Dışında Yapılan Çalışmalar

Krizsan ve Lambic (2024), çalışmasında ilkokulların alt sınıflarında farklı beceri ve yeteneklere sahip öğrenciler arasında eşli programlamanın verimlilik, motivasyon ve stres üzerindeki etkisini incelemiştir. Çalışmanın örneklemini üç farklı ilkokuldaki 168 öğrenci oluşturmaktadır. Öğrenciler 5-11 yaş grubunda yer almaktadır. Çocuklar rastgele olacak şekilde iki farklı grup olan deney ve kontrol gruplarına ayrıldılar. Deney grubuna görsel programlama araçlarıyla eşli programlama, kontrol grubuna geleneksel programlama öğretimi ile bireysel programlama öğretimi yapılmıştır. Deney grubu 82, kontrol grubu 86 öğrenciden oluşmaktadır. Öğrencilerin stres ve motivasyon düzeyleri ölçmek amacıyla likert tipi ölçek ile ek olarak programlama etkinliklerini değerlendirmek için gözlemlene kullanıldı. Sonuç olarak, eşli programlama programlama verimliliğini arttırmış, stres ve motivasyonu ise etkilemediği ortaya çıkmıştır (Krizsan ve Lambic, 2024).

Hamer vd. (2024) ortaöğretimdeki öğrencilerin kodlama tutumlarını incelemiştir. Bu çalışmanın temel amacı İngiltere’deki okullarda gençlerden oluşan büyük bir örneklem üzerinde yapılan anket yoluyla kodlama tutumlarını etkileyen faktörleri ortaya çıkarmaktır. Çalışmanın örneklemini 11-16 yaşındaki, 4983 ortaokul öğrencisi oluşturmaktadır. Veriler anket ile topladılar ve keşfedici faktör analizi yürüttüler ve çok değişkenli lojistik regresyon modelleri oluşturdular. Bulgularda kodlama tutumlarının düzeyinin yüksekliği, bilişim derslerindeki öğrenci deneyimi, öğretmen

ve ebeveyn desteđi, bilgisayar bilimcilerine ilişkin algılar, evde biliřim ve cinsiyet dahil olmak üzere çeřitli faktörlerle ilişkili olduđunu ortaya koydu (Hamer vd., 2024).

Chen vd. (2023) öz-düzenleme temelli BİD becerileri öğreniminde zaman yönetimi ve yardım aramanın etkisini incelenmiřlerdir. Arařtırmanın alıřma grubunu Tayvan'da yer alan bir okulun 3. ve 4. sınıf düzeyindeki 168 öđrencidir. Ü sınıfa öz-düzenlemeli, diđer üç sınıfa öđretmen rehberliđi yoluyla BİD eđitim verilmiřtir. Arařtırmacı ilkokulda resmi olarak alıřan öđretmendir. İki gruptaki öđrencilerde Code.org'dan aynı dersi aldı ve görevleri bađımsız olarak tamamladı. Öz-düzenlemeli grupta öđrenciler kendi hızında ilerlemiřtir. Öđretmen rehberliđi olan grupta öđrenciler ders iinde öđretmen kontrolünde alıřmalarını tamamlamıřtır. Geçerli 153 veri, eřleřtirilmiř t testleri ve ANCOVA kullanılarak analiz edilmiřtir. alıřmanın Arařtırma, bulgularında öz-düzenlemeli grup, akranlarla ve öđretmen rehberliđindeki öğrenme grubuna göre daha iyi öğrenme performansı ortaya ıkarmıřtır (Chen vd., 2023).

Menon (2022) tarafından gerekleřtirilen alıřma, öđrencilerin kendi bildirdikleri öz-düzenleyici öğrenme özelliklerinin, çevrimii ierikli lisans düzeyinde programlamaya giriş dersinin öğrenme ieriđinin görev deđerleriyle nasıl ilişkili olabileceđini belirlemeye alıřmıřtır. alıřma grubunu java programlama öğrenen lisans düzeyinde derse katılan öđrenciler oluřturmaktadır. Öđrencilere yönelik ierikler öğrenme yönetim sisteminden kısa videolar, kod gösterimleri olarak sunulmuřtur. Öđrencilerin tartıřtıđı forumlar ve öz-deđerlendirme yapmalarına olanak sađlanmıřtır. Arařtırma bulgularında, öđrencilerin görev deđerleri ile algıladıkları öz-düzenleyici öğrenme özellikleri arasında orta düzeyde pozitif bir korelasyon olduđu görölmüřtür (Menon, 2022).

Wei vd. (2021) kısmi eřli programlamanın ilkokul öđrencilerinin biliřimsel düşünme becerileri ve öz-yeterlilikleri üzerine etkisi arařtırmıřtır. alıřma grubunu aynı öđretmenden ders alan 4. sınıf öđrencileri oluřturmaktadır. Deney grubu 84, kontrol grubu 87 öđrenciden oluřmaktadır. Deney grubundaki öđrenciler programlama ödevlerini birlikte yapmaları iin eřleřtirilmiřtir. Kontrol grubundaki öđrenciler ödevlerini bireysel olarak gerekleřtirmiřtir. Arařtırmanın bulguları, eřli programlama

yapılan deney grubu öğrencilerinin BİD becerileri ve öz-yeterlikleri kontrol grubu öğrencilerine göre anlamlı olarak gelişim göstermiştir. Her iki grubunda BİD becerileri ve öz-yeterlikleri arasında düşük korelasyon bulunmuştur. Öğretmen ve öğrencilerle yapılan görüşmelerden çıkan sonuçlar, eşli programlamanın BİD becerilerini ve öz-yeterlikleri geliştirmek için etkili bir yöntem olduğunu göstermektedir (Wei vd., 2021).

Loksa ve Ko (2016) programlama problemi çözme sürecinde ve başarısında öz-düzenlemenin rolünü araştırmışlardır. Çalışmanın amacı programlamada henüz acemi olan 37 öğrencinin programlama hatalarını çözme sürecinde öz-düzenlemeyi nasıl kullandıklarını ortaya çıkarmaktır. Eğitim sürecinde iki farklı öğrenme grubu oluşturulmuştur. Gruplar cinsiyet, yaş ve öz-yeterlilik açısından dengeli oluşturulmuştur. Araştırmanın sonuçlarına göre, öğrencilerin programlama belirli problemlerini çözmek için öz-düzenlemeyi kullandığını fakat öz-düzenleme ile birlikte öğrencilerin programlama bilgilerinin etkili olduğu görülmüştür. Öz-düzenlemenin farklı türleri üzerinde çalışılmış ancak sadece birkaç öz-düzenleme türünün programlama ile ilişkili olduğu sonucuna varılmıştır (Loksa ve Ko, 2016).

Denner vd. (2014), eşli programlama öğretimi stratejisinin K-12 düzeyinde etkililiğini ve eşlerin birbirine etkisini incelemişlerdir. Çalışma grubu, bireysel çalışan 138 öğrenci kişi ve akranla çalışan 182 olmak üzere toplam 320 ortaokul öğrencisinden oluşmaktadır. Eşli programlamanın bireysel programlamaya karşı üstün yönleri ortaya çıkmıştır. Programlama deneyimi düşük olan öğrencilerin hesaplamalı düşünme becerilerini geliştirme ve programlamanın temel bilgilerini kazandırma açısından avantajlı olduğu görülmüştür. Eşli programlamada eşinden daha çok programlama deneyimine sahip olan bireylerin bilgisayara karşı daha olumlu tutumlar geliştirdiği görülmüştür. Bu durum ilk bilgisayar deneyimlerinin iş birliğini etkilediğini ortaya çıkarmıştır (Denner vd., 2014).

Alhazbi ve Hassan (2013) bilgisayar programlama dersinde öz-düzenleme öğrenme yaklaşımının öğrencilerin motivasyonuna ve performansına etkisini araştırmıştır. Araştırmada öz-düzenlemeli öğrenme süreçlerinin kullanıldığı deney grubunun performansı kontrol grubuna göre daha yüksek bulunmuştur. Programlama eğitiminde

öz-düzenleme süreçlerini kullanan öğrencilerin daha yüksek performans gösterdiği görülmektedir. Programlama öğrencilerin bağımsız çalışmalarını gerektiren bir süreçtir bu süreç içinde öz-düzenleme becerisi öğrencilerin performansını yükseltebilmektedir (Alhazbi ve Hassan, 2013).

Yurt dışı literatürde, öz-düzenleme becerisinin etkisinin daha çok lisans seviyesindeki programlama derslerinde araştırıldığı ve bu becerinin programlama öğretimine olumlu katkılar sağladığı görülmektedir. Benzer şekilde, işbirlikli öğrenme ile ilgili yurt dışı çalışmaların çoğunlukla eşli programlama üzerine yoğunlaştığı ve bu yaklaşımın da programlama öğretiminde olumlu etkiler yarattığı ifade edilmektedir.



3. YÖNTEM

Bu bölüm, araştırmanın modeli, çalışma grubuna ilişkin bilgiler, veri toplama araçları, verilerin toplanması ve verilerin analizi, uygulama süreci, araştırmacının rolü, geçerlik ve güvenirlik çalışmaları ile alakalı bilgileri kapsamaktadır.

3.1 Araştırmanın Modeli

Araştırmada öntest-sontest kontrol gruplu yarı deneysel desen kullanılmıştır. Bu desende yansız atama yapılmadan deney ve kontrol grupları seçilmektedir (Büyüköztürk vd., 2008). Deney sürecinden önce her iki gruba ön-test uygulanır, deney süreci boyunca deney grubuna müdahalede bulunulur, kontrol grubuna ise müdahalede bulunulmadan araştırma süreci sonunda her iki gruba da son test uygulanır (Özmen, 2023). Deneysel desen, araştırmadaki değişkenler arasındaki neden-sonuç ilişkisini ortaya çıkarmak için kullanılan bir modeldir. Bu modelde deney ve kontrol grubuna işlem öncesi ve işlem sonrası birtakım ölçümler yapılır (Büyüköztürk vd., 2008). Deneysel çalışmalarda gerçek deneysel desenin kullanılması geçerliği artırmakta, deney ve kontrol gruplarının yansız (rastgele) seçilmesi birçok olumsuz durumu da ortadan kaldırmaktadır. Ancak eğitim araştırmalarında gerçek deneysel desen kullanılması zordur, bu durumun başlıca sebebi sınıflarının okul yönetimi tarafından belirlenmesi, üzerinde çalışılacak olan öğrencilerin deney ve kontrol gruplarına yansız atanmasını engellemesi durumudur (Özmen, 2023).

Bu çalışmada, 6. sınıf düzeyinde iki şube belirlenmiş ve bir şubesine işbirlikli öğretim modeline uygun tekniklerle kurgulanmış programlama öğretimi uygulanırken, diğer şubesine öğretim programındaki teknikler uygulanmıştır. Süreç sonunda deney grubuna ve kontrol grubuna uygulanan tekniklerin öğrencilerin programlama öz-yeterliği, öz-düzenleme becerisi ve akademik başarıları üzerindeki etkisi incelenmiştir. Araştırma deseninde yapılacak işlemler Tablo 3.1’de gösterilmiştir.

Tablo 3.1 Araştırma deseni

Grup	Ön-test	Uygulama / işlem	Son-test
Deney (N=20)	PÖÖ, AÖÖ, ABT	İşbirlikli öğrenme teknikleri ile programlama öğretimi	PÖÖ, AÖÖ, ABT
Kontrol (N=23)	PÖÖ, AÖÖ, ABT	BTY Öğretim programlarındaki tekniklerle programlama öğretimi	PÖÖ, AÖÖ, ABT

3.2 Araştırmanın Çalışma Grubu

Araştırmanın çalışma grubunu Kastamonu İli Taşköprü İlçesinde bir devlet ortaokulunda öğrenim gören 6. sınıf öğrencileri oluşturmaktadır. Çalışma grubu belirlenirken seçkisiz olmayan uygun örneklem yöntemi kullanılmıştır. Uygun örnekleme, araştırmacı ihtiyacı olan örneklemini ulaşabildiği yanıtlayıcılardan başlayarak oluşturmasını sağlayan bir yöntemdir (Büyüköztürk vd., 2008). Uygun örnekleme yöntemi, araştırmacının kendi öğrencileri ile araştırma yapmak istediğinde kullanabileceği bir örneklem türüdür (Canbazoglu Bilici, 2023). Örneklem gruplarını seçkisiz oluşturmak okul yapısında mümkün olmadığı için şubeler seçkisiz olmayan örnekleme ile belirlenmiştir. Deney ve kontrol grubuna haftalık 2 saat, 8 hafta programlama eğitimi verilmiştir. Deney ve kontrol grubu katılımcılarının gruplara göre dağılımı Tablo 3.2’de gösterilmiştir.

Tablo 3.2 Katılımcıların gruplara göre dağılımı

Sınıf	Kız	Erkek	Yöntem	Kod
6/A Sınıfı	10	10	İşbirlikli öğrenme teknikleri ile programlama öğretimi	Deney Grubu
6/B Sınıfı	12	11	BTY Öğretim programındaki öğretim teknikleri ile programlama öğretimi	Kontrol Grubu
Toplam	22	21		

3.3 Araştırma Verilerinin Toplanması

Araştırmaya katılan öğrencilerin programlama öz-yeterlikleri ile ilgili verileri toplamak amacıyla Kukul vd. (2017) tarafından geliştirilen “Programlama Özyeterliliği Ölçeği”, öz-düzenleme becerileriyle ilgili verilerini toplamak amacıyla Arslan ve Gelişli (2015) tarafından geliştirilen “Algılanan Öz-düzenleme Ölçeği” ve öğrencilerin programlama öğretimi süreci sonunda akademik başarı durumlarını belirlemek için araştırmacı öğretmen tarafından geliştirilen “Programlama Başarı Testi”nden yararlanılmıştır.

3.3.1 Algılanan Öz-Düzenleme Ölçeği

Araştırmada öğrencilerin öz-düzenleme düzeylerinin belirlenmesi için Arslan ve Gelişli (2015) tarafından geliştirilen “*Algılanan Öz-Düzenleme Ölçeği*” kullanılmıştır. AÖÖ, ortaokul öğrencilerinin öz-düzenleme algı düzeylerini ölçmek amacıyla geliştirilmiştir. Ölçek 16 maddeden oluşmakta ve tüm maddeler; “1-Hiçbir Zaman, 2-Nadiren, 3-Arasıra, 4-Sık sık ve 5-Her zaman” aralığında yanıtlanmaktadır. Açıklayıcı faktör analizi sonucunda boyutların toplam varyansın %54,3’ünü açıkladığı ifade edilmiştir. Ölçeğin iki faktörlü yapısının doğrulanması amacıyla yapılan doğrulayıcı faktör analizinde ise Ki-kare ($\chi^2 = 147,60$; $sd = 95$; $p = 0,00$) değerinin anlamlı olduğu görülmüştür. Uyum indeksi değerleri ise RMSEA= 0,042; NFI= 0,98; CFI= 0,99; IFI= 0,99; RFI= 0,97; CFI= 0,99; GFI= 0,94; AGFI= 0,92; SRMR= 0,035 olarak bulunmuştur. Cronbach alfa iç tutarlılık güvenirlik katsayıları ölçeğin bütünü için 0,90; açık olma alt ölçeği için 0,84; arayış alt ölçeği için 0,82 olarak bulunmuştur (Arslan ve Gelişli, 2015).

Öğrencilerin algılanan öz-düzenleme becerilerinin düzeylerini göstermesi amacıyla kesme noktalarına göre algılanan öz-düzenleme ölçeğinden alınan puanlar kategorilere ayrılmıştır (Mahmod vd., 2023).

- Çok Düşük: 16 - 28 puan
- Düşük: 29 - 41 puan

- Orta: 42 - 54 puan
- Yüksek: 55 - 67 puan
- Çok Yüksek: 68 - 80 puan

3.3.2 Programlama Öz-yeterliği Ölçeği

Araştırmada öğrencilerin programlama öz-yeterliklerinin belirlenmesi için Kukul vd. (2017) tarafından geliştirilen *Programlama Öz-yeterliliği ölçeği* kullanılmıştır. PÖÖ, ortaokul öğrencilerinin programlama öz-yeterlik düzeylerini ölçmek amacıyla geliştirilmiştir. Ölçek 31 maddeden oluşmakta ve tüm maddeler; “1-Kesinlikle Katılmıyorum, 2-Katılmıyorum, 3-Kararsızım, 4-Katılıyorum ve 5-Kesinlikle Katılıyorum” aralığında yanıtlanmaktadır. Ölçekte ters puanlanan madde bulunmamaktadır.

Kukul vd. (2017), eş değer yarılama yöntemi kullanarak güvenilirliği belirlemiştir. Testin güvenilirliğini belirlemek amacıyla başka bir yöntem olan Cronbach alpha güvenirlik analizi yapılmış ve sonucunda ölçeğin güvenirlik katsayısı 0,95 olarak bulunmuştur. Testin tamamına yönelik güvenirlik belirlemek amacıyla Spearman-Brown yöntemi ile güvenirlik analizi yapılmış ve elde edilen güvenirlik katsayısı $r = 0,966$ bulunmuştur (Kukul vd., 2017). Ölçek için açıklayıcı faktör analizi yapılmış ve sonucunda 33 maddeden 2 tanesi çıkarılmıştır. PÖÖ’nün nihai olarak 31 madde yer almıştır. Tek faktör altında değerlendirilmiş varyansı %41,15 bulunmuştur. Açıklayıcı Faktör Analizi (AFA) sonucu elde edilen faktör yapısına ilişkin modelin uygunluğu Doğrulayıcı Faktör Analizi (DFA) ile test edilmiştir (Kukul vd., 2017).

Analizler sonucunda PÖÖ’nün tek faktörlü yapıyı gösterdiği ifade edilmiştir. Öğretmenlerin ölçeği Scratch, Alice, logo vb. BTP platformları kullanarak öğrencilerin programlama becerilerini geliştirme sürecinde programlama öz-yeterlik düzeylerini ölçmek amacıyla kullanabileceği belirtilmiştir (Kukul vd., 2017).

Öğrencilerin programlama öz-yeterlik düzeylerini belirlemek amacıyla kesme noktalarına göre programlama öz-yeterlik ölçeğinden alınan puanlar kategorilere ayrılmıştır (Mahmod vd., 2023).

- Çok Düşük: 31 - 55 puan
- Düşük: 56 - 80 puan
- Orta: 81 - 104 puan
- Yüksek: 105 - 129 puan
- Çok Yüksek: 130 - 155 puan

3.3.3 Akademik Başarı Testi

Programlamaya yönelik Akademik Başarı Testi, MEB tarafından hazırlanan BTY dersi 6. sınıf öğretmen rehberinde yer alan problem çözme ve programlama ünitesindeki etkinlikler taranarak araştırmacı öğretmen tarafından geliştirilmiştir. Testin hazırlanması sürecinde alanda geçerlik ve güvenilirliğini sağlamak amacıyla uzman iki akademisyen, üç Bilişim Teknolojileri öğretmenin görüşleri alınmıştır. Diğer yandan testin açık ve anlaşılır olup olmadığını anlamak amacıyla bir Türkçe öğretmeninden görüş alınmıştır. Akademik başarı testi, deney sürecinde öğrencilerin programlamaya ilişkin becerileri ne düzeyde kazandıklarını ortaya çıkarmak amacıyla geliştirilmiştir. Başarı testinde ölçülmek istenen bilgi ve becerilerin ilgili kazanımların tamamının temsil edildiği 26 soruluk çoktan seçmeli test hazırlanmıştır (EK F). Testin, ölçülmek istenilen hedef ve davranışları tamamıyla yansıtıp yansıtmadığını belirlemek amacıyla konu ve hedeflerin karşılaştırıldığı belirtke tablosu hazırlanır (Büyüköztürk vd., 2008). Başarı testi hazırlanması sürecinde ölçülmesi hedeflenen kazanımlar ve madde sayılarının yer aldığı belirtke tablosu Tablo 3.3'te gösterilmiştir.

Tablo 3.3 Belirtke tablosu

Kazanım	Madde Sayıları	Maddeler
BT.6.5.1.3. Bir problemi alt problemlere böler.	2	1, 2
BT.6.5.1.4. Temel fonksiyonları problem çözme sürecinde kullanır.	2	3, 23
BT.6.5.1.6. Bir algoritmanın çözümünü test eder.	4	4, 5, 26
BT.6.5.1.7. Farklı algoritmaları inceleyerek en hızlı ve doğru çözümü seçer.	1	15, 19
BT.6.5.1.8. Hatalı bir algoritmayı doğru çalışacak biçimde düzenler.	2	10, 16
BT.6.5.2.1. Blok tabanlı programlama aracının arayüzünü ve özelliklerini tanır.	5	6, 7, 8, 9, 25
BT.6.5.2.5. Doğrusal mantık yapısını içeren programlar oluşturur.	2	13, 14
BT.6.5.2.11. Döngü yapısını içeren programlar oluşturur.	4	21, 22, 24,17
BT.6.5.2.7. Karar yapısını içeren programlar oluşturur.	2	12, 20
BT.6.5.2.9. Çoklu karar yapıları içeren programlar oluşturur.		
BT.6.5.2.12. Döngü yapısını içeren programları test ederek hatalarını ayıklar.	1	18
BT.6.5.2.8. Karar yapısını içeren programları test ederek hatalarını ayıklar.	1	11
Toplam	26	

Testin güvenilirlik ve geçerliğini arttırmak amacıyla testin pilot uygulaması yapılmıştır. Pilot çalışma, araştırmanın yapıldığı okulda bir eğitim dönemi önce BTY dersine katılmış olan 7. sınıfta öğrenim gören 34 öğrenciye başarı testi uygulanması olarak gerçekleştirilmiştir. Pilot çalışma sonrasında test maddelerinin madde güçlükleri ve madde ayırt edicilikleri incelenmiştir. Akademik başarı testi madde analizleri Tablo.3.4'te gösterilmiştir.

Tablo 3.4 Madde analizleri

Madde No	Madde Güçlüğü	Madde Ayırt Ediciliği	Madde No	Madde Güçlüğü	Madde Ayırt Ediciliği
1	0,71	0,33	14	0,32	0,44
2	0,85	0,56	15	0,59	0,78
3	0,47	0,56	16	0,79	0,22
4	0,62	0,33	17	0,32	0,44
5	0,50	-0,30	18	0,68	0,67
6	0,47	0	19	0,44	0,33
7	0,12	0,22	20	0,50	0,89
8	0,44	0,44	21	0,56	0,44
9	0,26	0,22	22	0,76	0,78
10	0,74	0,56	23	0,29	0,33
11	0,50	0,67	24	0,35	0,22
12	0,65	0,56	25	0,41	0,67
13	0,56	0,33	26	0,26	0,44

Madde güçlüğü, başarı testlerinde maddelerin doğru cevaplanma oranını göstermektedir (Büyüköztürk vd., 2008). Karaca (2022)'ya göre madde güçlüğü 0,00 ile 1,00 arasında değer alır. Maddelerin güçlük değeri 1,00 değerine yaklaştıkça madde kolaylaşırken 0,00 değerine yaklaştıkça madde zorlaşmaktadır (Karaca, 2020). Bu doğrultuda Büyüköztürk vd. (2008)'ne göre başarı testlerinde ortalama güçlüğü 0,50 olması beklenmektedir. Madde güçlüğü teste son hal verilirken madde seçiminde bir ölçüt olarak kullanılmaktadır (Büyüköztürk vd., 2008). Başarı testinin maddeleri üzerinde yapılan çalışmalar sonucunda testin ortama güçlüğü 0,50 olarak bulunmuştur. Testin uygulandığı grubun önceki eğitim döneminde derste programlama konuları öğrenmiş olduğundan testin güçlüğü yeterli seviyede çıkmıştır.

Ayırt edicilik indeksi -1,00 ile 1,00 arasında değer alır ve madde 1,00 değerine yaklaştıkça ayırt ediciliği artar (Karaca, 2022). Ayırt edicilik indeksi 0,40 ile 1,00 değerleri arasında yüksek düzeyde ayırt edici, 0,30 ile 0,40 değerleri arasında orta düzeyde ayırt edici, 0,20 ile 0,30 değerleri arasında düşük düzeyde ayırt edici, 0,00 ile 0,20 değeri arasında çok düşük düzeyde ayırt edicidir (Büyüköztürk vd., 2008). Büyüköztürk vd. (2008)'ne göre ayırt edicilik indeksi 0,30 ile 0,40 arasındaki maddeler düzeltmeden testte tutulabilir, 0,20 ile 0,30 arasındaki maddeler ise

düzeltilerek ve geliştirilerek teste alınabilir. Başarı testinde bulunan 5, 6 numaralı maddelerin ayırt edicilikleri yeterli olmadığından testten atılmasına karar verilmiştir. Başarı testinde bulunan 7, 9, 16, 24 numaralı sorular düşük düzeyde ayırt edici olarak hesaplanmış ve uzman görüşü alınarak testin kapsam geçerliliğini düşürmemek için bu sorular düzeltilip tekrardan teste dahil edilmiştir. Kapsam geçerliliği, bir testin ölçülmek istenen davranışların bütününe kapsayacak şekilde olmasıdır (Büyüköztürk vd., 2008). Başarı testinde gerekli düzenlemeler yapıldıktan sonra test nihai olarak 24 sorudan oluşmaktadır.

KR-20, KR-21, Cronbach Alpha gibi farklı güvenirlik belirleme yöntemleri vardır. Başarı testinin Güvenirliğini belirlemek için Cronbach Alpha değeri hesaplanmıştır (Büyüköztürk, 2008; Yurdabakan, 2022). Başarı testinin güvenirliğini belirlemek için yapılan analizler sonucunda Cronbach Alpha değeri 0,72 olarak bulunmuştur. Testteki ayırt ediciliği düşük 5 ve 6 numaralı sorular atılınca yeniden hesaplanan Cronbach Alpha değeri 0,77 bulunmuştur. Bu doğrultuda başarı testinin güvenilir olduğu sonucu ortaya çıkmaktadır (Karaca, 2022).

Öğrencilerin akademik başarı düzeylerini belirlemek ve göstermek amacıyla kesme noktalarına göre akademik başarı testinden alınan puanlar kategorilere ayrılmıştır (Mahmod vd., 2023).

- Çok Düşük Başarı: 0 - 4 doğru cevap
- Düşük Başarı: 5 - 9 doğru cevap
- Orta Başarı: 10 - 14 doğru cevap
- Yüksek Başarı: 15 - 19 doğru cevap
- Çok Yüksek Başarı: 20 - 24 doğru cevap

3.4 Araştırmanın Etik İzinleri

Veriler Kastamonu Üniversitesi Sosyal ve Beşeri Bilimler Araştırma Etik Kurulundan ve Kastamonu İl Milli Eğitim Müdürlüğünden izin alındıktan sonra toplanmıştır (EK B). Katılımcılara çalışmanın amacı, kişisel bilgilerin gizliliği, verilerin ne amaçla kullanılacağı, ölçeklerin açıklaması ve ölçeklerin nasıl doldurulması gerektiği konuları hakkında yüz yüze bilgi verilmiştir. Araştırma kapsamında katılımcı öğrencilerin velilerinden gerekli izinler alınmıştır.

3.5 Uygulama Süreci

Bu başlık altında araştırmacının 8 haftalık uygulama sürecine ve deney sürecinde kullanılan ders planı, günlük planı ve materyallere yer verilmiştir.

Deneyi uygulama sürecinde çalışma gruplarına ön-test uygulanarak grupların programlama önbilgisi bakımından denklikleri kontrol edilmiş ve denk oldukları tespit edilmiştir. Deney ve kontrol gruplarında öğretim etkinliklerini gerçekleştirmek amacıyla Bilişim Teknolojileri sınıfı kullanılmıştır. BTY dersinin teorik konularında 4-6 kişilik gruplar, bilgisayarda uygulamalı konularda ise ikili gruplar oluşturulmuştur. Öğrenci grupları her hafta öğrencilerin önceki hafta performansları baz alınarak yeniden oluşturulmuştur. Kontrol grubunda BTY öğretim programında yer alan yöntem ve teknikler ile programlama etkinlikleri gerçekleştirilmiştir. Deney grubunda programlama etkinlikleri işbirlikli öğrenme teknikleri ile kurgulanarak gerçekleştirilmiştir. Deney 8 hafta sürmüştür, çok fazla devamsızlık yapan öğrenciler deney sürecine dahil edilmemiştir.

Öğrencilere öğretilen konular, BTY öğretim programına göre hazırlanan ünitelendirilmiş yıllık plandan “problem çözme ve programlama” ünitesinin ardışık 8 haftalık konular kapsamında yürütülmüştür. Deney sürecinde kanıt olması açısından fotoğrafları alınmış kişisel verileri koruma amacıyla öğrencilerin yüzleri kapatılarak ekler bölümüne eklenmiştir (EK G).

3.5.1 Dersin Genel Planı

Hazırlanan ders planı, BTY dersi problem çözme ve programlama ünitesi kapsamında öğrencilere kazandırılması amaçlanan ortak kazanımlar ile deney ve kontrol gruplarında ayrı ayrı uygulanacak teknik, yöntem ve materyalleri kapsamaktadır. Ders planı Tablo 3.3’de gösterilmiştir.

Tablo 3.5 Ders planı

Hafta	Konu ve Kazanım	Deney Grubu	Kontrol Grubu
1. Hafta	Problem Çözme - Problem nedir? - Basit Problemler - Karmaşık Problemler Kazanımlar: BT.6.5.1.3. Bir problemi alt problemlere böler. BT.6.5.1.4. Temel fonksiyonları problem çözme sürecinde kullanır.	Materyaller: - Problem türleri çalışma yaprağı - Balık kılçığı etkinlik yaprağı Öğretim Strateji, Yöntem ve Teknikleri: - Beyin fırtınası - Buluş yolu stratejisi - Ayrılıp-birleşme tekniği	Materyaller: - Problem türleri çalışma yaprağı - Balık kılçığı etkinlik yaprağı Öğretim Strateji, Yöntem ve Teknikleri: - Sunuş yolu stratejisi, - Düz anlatım, - Gösterip-yaptırma, - Örnek olay
2. Hafta	Algoritma - Algoritma ve faydaları - Algoritma örnekleri - Algoritma oluşturma - Hatalı algoritma bulma - Algoritma- problem çözme bağlantısı Kazanımlar: -BT.6.5.1.5. Problemin çözümü için bir algoritma geliştirir. -BT.6.5.1.6. Bir algoritmanın çözümünü test eder. -BT.6.5.1.7 Farklı algoritmaları inceleyerek en hızlı ve doğru çözümü seçer. -BT.6.5.1.8 Hatalı bir algoritmayı doğru çalışacak şekilde düzenler.	Materyaller: - Bilgisayarsız kodlama etkinlikleri, - Algoritma çalışma yaprağı, - Compuware.it platformu Öğretim Strateji, Yöntem ve Teknikleri: - Beyin fırtınası - Buluş yolu stratejisi, - Öğrenci takımı başarı bölümü	Materyaller: - Keşfetprojesi.org sunumları ve materyalleri - Bilgisayarsız kodlama etkinlikleri, - Compuware.it platformu, - Algoritma çalışma yaprağı Öğretim Strateji, Yöntem ve Teknikleri: - Sunuş yolu stratejisi, - Düz anlatım, - Gösterip-yaptırma, - Örnek olay

Tablo 3.5'in devamı

Hafta	Konu ve Kazanım	Deney Grubu	Kontrol Grubu
4. Hafta	Doğrusal programlama - Hareket blokları - Görünüm blokları - Olaylar blokları Kazanımlar: -BT.6.5.2.5 Doğrusal mantık yapısını içeren programlar oluşturur.	Materyaller: - Scratch Platformu, - Code.org platformu Öğretim Strateji, Yöntem ve Teknikleri: - Gösterip-yaptırma - Buluş yolu stratejisi - Öğrenci takımları başarı bölümleri - Eşli programlama	Materyaller: -Scratch Platformu - Code.org platformu Öğretim Strateji, Yöntem ve Teknikleri: - Sunuş yolu stratejisi, - Düz anlatım, - Gösterip-yaptırma
5. Hafta	Karar ve Çoklu Karar Yapıları -Kontrol blokları /Eğer ve eğer-ise blokları Kazanımlar: -BT.6.5.2.7 Karar yapısını içeren programlar oluşturur. -BT.6.5.2.9 Çoklu karar yapısını içeren programlar oluşturur.	Materyaller: - Scratch Platformu, - Code.org platformu Öğretim Strateji, Yöntem ve Teknikleri: - Gösterip-yaptırma - Buluş yolu stratejisi - Takım-Oyun-Turnuva tekniği - Eşli programlama	Materyaller: -Scratch Platformu - Code.org platformu Öğretim Strateji, Yöntem ve Teknikleri: - Sunuş yolu stratejisi, - Düz anlatım, - Gösterip-yaptırma,
6. Hafta	Döngü Yapıları -Tekrarlama blokları/ ..defa tekrarla, sürekli tekrarla -İşlem blokları Kazanımlar: -BT.6.5.2.11 Döngü yapısını içeren programlar oluşturur.	Materyaller: - Scratch Platformu, - Code.org platformu Öğretim Strateji, Yöntem ve Teknikleri: - Buluş yolu stratejisi - Eşli programlama	Materyaller: - Scratch Platformu, - Code.org platformu Öğretim Strateji, Yöntem ve Teknikleri: - Sunuş yolu stratejisi, - Anlatım, - Gösterip-yaptırma
7. Hafta	Hata Ayıklama ve Test Etme -Değişken oluşturma, -Algılama blokları, -Karar yapıları Kazanımlar: -BT.6.5.2.8 Karar yapısını içeren programları test ederek hatalarını ayıklar. -BT.6.5.2.10 Çoklu karar yapısını içeren programları test ederek hatalarını ayıklar.	Materyaller: - Scratch Platformu, - Code.org platformu Öğretim Strateji, Yöntem ve Teknikleri: - Buluş yolu stratejisi - Takım-oyun-turnuva - Eşli programlama	Materyaller: - Scratch Platformu, - Code.org platformu Öğretim Strateji, Yöntem ve Teknikleri: - Sunuş yolu stratejisi, - Düz anlatım, - Gösterip-yaptırma

Tablo 3.5'in devamı

Hafta	Konu ve Kazanım	Deney Grubu	Kontrol Grubu
8. Hafta	Hata Ayıklama ve Test Etme -İşlemler, -Değişkenler -Döngü yapıları Kazanımlar: -BT.6.5.2.12 Döngü yapısını içeren programları test ederek hatalarını ayıklar.	Materyaller: - Scratch Platformu - Code.org platformu Öğretim Strateji, Yöntem ve Teknikleri: - Beyin fırtınası - Buluş yolu stratejisi - Öğrenci takımı başarı bölümleri - Eşli programlama	Materyaller: - Scratch Platformu - Code.org platformu Öğretim Strateji, Yöntem ve Teknikleri: - Sunuş yolu stratejisi, - Düz anlatım, - Gösterip-yaptırma

3.5.2 Günlük Plan

BTY öğretim programındaki kazanımlar ile ilişkili olarak işbirlikli öğrenme teknikleriyle kurgulanan programlama öğretimi dersleri için hazırlanan ve deney grubuna uygulanan birinci haftaya ait günlük ders planı aşağıdaki Tablo 3.6'da gösterilmiştir. Diğer planlar ise ekler bölümünde sunulmuştur (EK C).

Tablo 3.6 Birinci haftaya ait örnek günlük plan

Ders: Bilişim Teknolojileri ve Yazılım		
Sınıf: 6	Hafta: 1	Süre: 40 dk+40 dk
Ünite: Problem Çözme ve Programlama		
Konular: BT.6.2.3 Böl, Parçala, Çöz		
• Problem nedir? • Problem Çeşitleri • Basit Problemler • Karmaşık Problemler		
Kazanımlar:		
• BT.6.5.1.3. Bir problemi alt problemlere böler. • BT.6.5.1.4. Temel fonksiyonları problem çözme sürecinde kullanır.		
Öğretme, Öğrenme Yöntem ve Teknikleri:		
• Buluş yolu Stratejisi • Beyin fırtınası • Örnek olay • İşbirlikli Öğrenme Tekniği: Ayrılıp-Birleşme • Balık kılıcı etkinliği yaprağı • MEB Öğretmen Kılavuzu		

Tablo 3.6'nın devamı

Öğretme-Öğrenme Etkinlikleri

1-Dikkat Çekme: Öğretmen öğrencilere gece uyuduktan 1 saat sonra elektriklerin kesilmesi bizim için problem midir? Ödev yaparken elektriklerin kesilmesi bizim için problem midir? soruları sorulur.

2-Güdüleme: Öğretmen öğrencilere günlük hayatta en sık karşılaşılan problemlerden örnek olaylar sunar.

3-Hedeften haberdar etme: Hayatımızda birçok problem olduğunu ve derste öğrenilecek problem çözme basamakları kullanılarak karşımıza çıkan her problemi çözebileceğimiz öğretmen tarafından öğrencilere ifade edilir.

4-Ön koşul öğrenmelerinin hatırlatılması: soru-cevap etkinliğiyle öğrencilerin bir önceki hafta konusu olan veri konusuna ait bilgilerle bağ kurulur.

Öğrenme ve Öğretme Etkinlikleri:

Öğretmen ayrılıp-birleşme tekniğine göre dört öğrenciden oluşan grup oluşturur. Gruplar öğrencilerin akademik başarı durumlarına göre heterojen olarak oluşturulur. Öğretmenin öğrencilere öğrenmesi için verdiği dört konu:

1. Problem Nedir?
2. Problem Çözme Basamakları
3. Basit Problemler
4. Karmaşık Problemler

Her grup üyesine birden dörde kadar numara verilir. Her gruptan bir numaraya sahip öğrenciler problem nedir? konusunu, iki numaralı öğrenciler problem çeşitleri konusunu, üç numaralı öğrenciler basit problemler konusunu, dört numaralı öğrenciler karmaşık problemler konusunu bir araya gelerek çalışırlar. Konuları öğrenmeleri için öğretmen çalışma kâğıdı ve öğrencileri öğrenmeye yönlendiren sorular verecektir. Süreç sonunda her öğrenci dört kişilik kendi grubuna geri dönecek ve grubunun diğer üyelerine kendi konusunu anlatacaktır. İlk grup öğrenmesi için 10 dakika, ikinci grup öğrenmesi için 20 dakika verilecektir. Öğrenme süreci içinde her grup aşağıdaki etkinlikleri yerine getirecektir:

I. Etkinlik: Öğrencilere günlük hayatta karşılaşılabilen bir problemlerden biri olan “sabahları okula geç kalma” durumunu öğrencilerin balık kılçığı tekniği ile bu problemi analiz etmeleri sağlanacaktır. Balık kılçığı tekniğinde balığın kılçıklarında boş bırakılan yerlere problemin alt nedenleri yazılır. Balığın başına ise problemin kendisi yazılır. Ortaya çıkan balık kılçığında sabahları okula geç kalma probleminin ana nedenleri ve alt nedenleri saptanmış olur. Etkinlik için 10 dakika süre verilecektir.

II. Etkinlik: Problem çözme basamaklarına -Problemi anlama, plan yapma, planı uygulama ve çözümü değerlendirme- göre balık kılçığında analiz edilen probleme çözüm üreteceklerdir. Problemi anlamada, öğrenciler problemin farkına varırlar ve balık kılçığında analiz ederler. Plan yapmada, öğrenciler probleme çözüm için fikir üretirler. Planı uygulamada, fikir olarak bulunan çözüm uygulamaya konulur. Çözümü değerlendirme de ise çözüm uygulandığında başarılı olup olmama durumu değerlendirilir. Etkinlik için 5 dakika süre verilecektir.

III. Etkinlik: Öğrenci gruplarına karmaşık ve basit problemler ile ilgili birbirinden farklı örnekler verilir. Öğrencilerin bu örnekleri değerlendirerek sınıfa sunar. Etkinlik için 5 dakika süre verilecektir.

Tablo 3.6'nın devamı

Ölçme ve Değerlendirme:

1. Bireysel öğrenme etkinliklerine yönelik ölçme-değerlendirme: Her grup üyesine 5 soruluk kısa sınav yapılacak ve sınav için 10 dakika süre verilecektir. Sorular aşağıdaki gibidir:

1. Aşağıdaki seçeneklerden hangisi karmaşık problem olabilir?
 - A) Orijinal bir proje tasarlamak
 - B) Markete gitmek
 - C) Kek yapmak
 - D) Bilgisayarı açmak
2. Aşağıdakilerden hangi bir problem çözme adımı değildir?
 - A) Problemi Anlama
 - B) Problemi Çözme
 - C) Planı Uygulama
 - D) Çözümü Değerlendirme
3. Aşağıdakilerden hangi Problem Çözme stratejisidir?
 - A) Ve diyagramı
 - B) Kavram haritası
 - C) Kavram Karikatürü
 - D) Balık kılçığı
4. Aşağıdakilerden hangisi bizim için bir problem olabilir?
 - A) Ders çalışırken elektriklerin kesilmesi
 - B) Gece yatınca elektriklerin kesilmesi
 - C) Bilgisayarı kapatınca elektriklerin kesilmesi
 - D) Telefonun şarjı varken elektrik kesilmesi
5. Aşağıdakilerden hangisi basit bir problemdir?
 - A) Derse geç kalmak
 - B) Geç kalmamak için alarm kurmak
 - C) Test çözmek
 - D) Arabayı tamir etmek

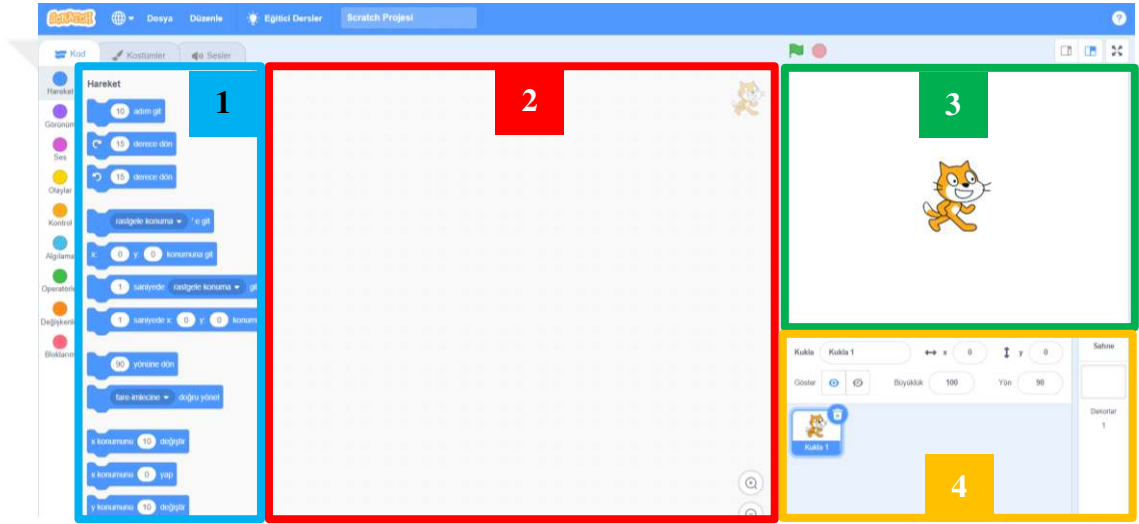
Her grup üyesinin bireysel olarak topladığı puanlar toplanarak grup puanları belirlenecektir.

3.5.3 Eğitim Materyalleri

Bu araştırmanın deney sürecinde BTY dersindeki etkinliklerde kullanılan materyaller keşfet projesi BTP ve bilgisayarsız kodlama materyalleridir. Keşfet projesi, MEB-İstanbul İl Milli Eğitim müdürlüğü ve Google iş birliği ile 2014 yılında öğretmen, veli ve öğrencilerin kullanıma sunulan BTY materyallerini barındıran bir eğitim web sitesidir. Keşfet projesinin uzantısı olarak “Kodlamayı Keşfediyorum projesi,” geliştirilmiştir. Bu proje kapsamında BİD ve BTP etkinliklerine yer verilmiştir.

Çocukların Algoritmik düşünme, ayrıştırma, hata yakalama, çözüm değerlendirme vb. BİD becerilerini geliştirmeyi amaçlayan proje kapsamında BTP ve bilgisayarsız kodlama ile alakalı görsel, sunum, video, çalışma kâğıdı materyallerine yer verilmiştir (Tonbuluğlu, 2022).

BTY dersi kapsamında “problem çözme ve programlama” konularının öğrenciler tarafından öğrenildikten sonra uygulanması amacıyla BTY Öğretmen kılavuzunda yer alan ve BTP platformu olan Scratch yazılımı kullanılmıştır. Scratch platformunun arayüzüne ait bilgiler ve görseller Şekil 3.1’de gösterilmiştir.



Şekil 3.1 Scratch platformu arayüzü (URL-7, 2023)

- 1 numaralı alan, kod blokları alanıdır (URL-7, 2023).
- 2 numaralı alan, projenin hayata geçtiği alandır (URL-7, 2023).
- 3 numaralı alan, kod bloklarını sürükleyerek programlama yapılan alandır (URL-7, 2023).
- 4 numaralı alan, sahne ve kukla olarak ifade edilen nesnelerin bulunduğu alandır (URL-7, 2023).

Tablo 3.7 Scratch kod blokları (URL-7, 2023)

Kategori	Tanım
Hareket blokları	Sahnedeki kukla ve nesnelerin hareketlendirilmesini sağlayan kodlardır (URL-7, 2023).
Olaylar blokları	Yazılan kodların çalışması için kod başlatıcı veya sinyal verici olarak kullanılan başlangıç kodlarıdır (URL-7, 2023).
Görünüm blokları	Sahne ve kuklaların görünümünü yani renk, boyut değiştirme veya konuşma balonları oluşturmak amacıyla kullanılan kodlardır (URL-7, 2023).
Kontrol blokları	Kodları yazarken tekrar eden kodları döngüye almamızı sağlayan kodlar ile şartlara bağlı olarak yönlendirme yapılacak kodlar yer alır (URL-7, 2023).
Ses blokları	Kuklaya nota ve sesler eklemek için gerekli kodların bulunduğu kategoridir (URL-7, 2023).
Algılama blokları	Sahne ve kuklaların bulunduğu konumu tespit etme gibi durumlarını algılamayı sağlayan kodlardır (URL-7, 2023).
Operatörler (İşlem) blokları	Program yazılırken gerekli olan matematiksel ve mantıksal kodların yer aldığı kodlardır (URL-7, 2023).
Değişkenler (Veri) blokları	Verileri depolamak ve kullanmak amacıyla değişken ve liste oluşturma kodları yer alır (URL-7, 2023).
Bloklarım (Özel Taşlar)	Scratch içinde halihazırda bulunmayan ve programcının kendine özgü kod dizileri oluşturmasını sağlayan kodlardır (URL-7, 2023).

İşbirlikli öğrenme ile programlama öğretimi etkinliklerinde kullanılan bir başka BTP platformu code.org web sitesidir. Code.org, okullardaki öğrencilerin bilgisayar bilimlerini öğrenmesini sağlamak amacıyla kar amacı olmayan ve herkese açık bir projedir (Tonbuloğlu, 2022). Code.org, içerisinde programlama eğitimi sağlayan oyunlar barındırmaktadır. Öğrenciler bu oyunları oynayarak programlamanın yapısını kavrayabilirler (Selçuk, 2022). Bilgisayar bilimi içeriklerinin dünyanın her yerindeki ilkokul ve ortaokul öğrencilerine sunmakta ve kodlama saati adı altında etkinlik düzenleyerek tüm öğrencilerin kodlama etkinlikleri yapmasını sağlamaktadır (Tonbuloğlu, 2022).

3.6 Verilerin Analizi

Eğitimin ilk haftasından önce ön-test verileri toplanmış ve öğrencilere süreç hakkında bilgi verilmiştir. Araştırmacı tarafından sekiz haftalık bir programlama eğitimi sonrasında son-test verileri sınıf ortamında toplanmıştır. Katılımcı öğrencilerin özel bilgileri ölçeklerde yer almamıştır. Katılımcıların devamsızlık durumları ve ön-testi doldurmamış olma sebepleri ile ancak 39 öğrenciden veri toplanabilmektedir. Toplanan veriler SPSS programı ile analiz edilmiştir.

Deney ve kontrol gruplarında verilerin normal dağılım gösterip göstermediğini belirleyebilmek amacıyla Kolmogorov-Smirnov ve Shapiro-Wilk testleri yapılmıştır. Büyüköztürk (2012) normal dağılımı belirlemek amacıyla kullanılan testlerden Shapiro-Wilk testini örneklem büyüklüğünün 50'nin altında olduğu durumlarda kullanılmasını önermektedir. Shapiro-Wilk testinde p(Sig) değerinin 0,050 üstünde olması durumunda verilerin normal dağılım gösterdiği bilinmektedir (Büyüköztürk, 2012). Diğer yandan da katılımcılara uygulanan testlerin çarpıklık katsayısı ve basıklık katsayısı değerlerine bakılarak verilerin normal dağılıma sahip olup olmadığı tespit edilebilir. Çarpıklık katsayısı ve basıklık katsayısı değerlerinin -1,96 ve +1,96 değerleri arasında olması normal dağılım olduğunu göstermektedir (Best ve Kahn, 2017).

Tablo 3.8 Akademik başarı testi normallik verileri

Test	Grup	Kolmogorov-Smirnov			Shapiro-Wilk		
		İstatistik	df	Sig.	İstatistik	df	Sig.
Akademik Başarı Kontrol		0,103	21	0,200	0,972	21	0,779
Ön- Testi	Deney	0,125	18	0,200	0,961	18	0,616
Akademik Başarı Kontrol		0,127	21	0,200	0,940	21	0,222
Son-Testi	Deney	0,207	18	0,040	0,849	18	0,008

Tablo 3.8’de gösterilen Shapiro-Wilk testinin verileri incelendiğinde akademik başarı ön-testi kontrol grubu $p= 0,779$ ve deney grubu $p= 0,616$ değerleri elde edilmiştir. Ortaya çıkan değerlere göre Akademik Başarı ön-testi normal dağılım göstermektedir. Akademik başarı son-testi kontrol grubu da $p= 0,222$ normal dağılım göstermektedir ancak deney grubunun normal dağılım göstermediği görülmektedir ($p= 0,008$).

Tablo 3.9 Akademik başarı testi çarpıklık ve basıklık değerleri

Test	Grup	Ortalama	ÇK	BK
Akademik Başarı Ön-Testi	Kontrol	12,71	-0,88	-0,64
	Deney	12,83	-1,10	-1,1
Akademik Başarı Son-Testi	Kontrol	14,71	-0,001	-1,23
	Deney	17,83	-0,60	-1,2

Tablo 3.9’de yer alan çalışma gruplarının çarpıklık katsayısı ve basıklık katsayısı değerleri incelendiğinde akademik başarı ön-testi deney grubu (ÇK= -1,10 ve BK=-1,1) ve kontrol grubu (ÇK=-0,88 ve BK= -0,64) ve akademik başarı son-testi deney grubu (ÇK= -0,60 ve BK=-1,2) ve kontrol grubu (ÇK=0,001 ve BK= -1,23) değerlerinin -1,96 ve +1,96 değerleri arasında olması normal dağılım olduğunu göstermektedir.

Tablo 3.10 Programlama öz-yeterliği testi normallik verileri

Test	Grup	Kolmogorov-Smirnov			Shapiro-Wilk		
		İstatistik	df	Sig.	İstatistik	df	Sig.
Programlama Öz-yeterliği Ön Testi	Kontrol	0,140	21	0,200	0,962	21	0,558
	Deney	0,077	18	0,200	0,966	18	0,721
Programlama Öz-yeterliği Son Testi	Kontrol	0,098	21	0,200	0,965	21	0,625
	Deney	0,267	18	0,001	0,858	18	0,011

Tablo 3.10’da gösterilen Shapiro-Wilk testinin verileri incelendiğinde programlama öz-yeterliği ön-testi kontrol grubu $p = 0,558$ ve deney grubu $p = 0,721$ değerleri ile programlama öz-yeterliği son-testi kontrol grubu $p = 0,625$ değerleri incelendiğinde normal dağılım gösterdiği ancak deney grubu $p = 0,011$ değeri incelendiğinde normal dağılım göstermediği görülmektedir.

Tablo 3.11 Programlama öz-yeterliği testi çarpıklık ve basıklık değerleri

Test	Grup	Ortalama	ÇK	BK
Programlama Öz-yeterliği Ön-Testi	Kontrol	68,00	0,05	-0,92
	Deney	73,36	0,33	-0,31
Programlama Öz-yeterliği Son-Testi	Kontrol	99,34	-0,26	-0,81
	Deney	135,04	-1,28	1,05

Tablo 3.11’de yer alan çalışma gruplarının çarpıklık katsayısı ve basıklık katsayısı değerleri incelendiğinde programlama öz-yeterliği ön-testi deney grubu (ÇK= 0,33 ve BK=-0,31) ve kontrol grubu (ÇK= 0,05 ve BK= -0,92) ve programlama öz-yeterliği son-testi deney grubu (ÇK= -0,26 ve BK=-0,81) ve kontrol grubu (ÇK= -1,28 ve BK= 1,05) değerleri incelendiğinde verilerin normal dağılım gösterdiği görülmektedir.

Tablo 3.12 Öz-düzenleme testi normallik verileri

Test	Grup	Kolmogorov-Smirnov			Shapiro-Wilk		
		İstatistik	df	Sig.	İstatistik	df	Sig.
Öz-düzenleme	Kontrol	0,103	21	0,200	0,955	21	0,422
Ön-Testi	Deney	0,117	18	0,200	0,981	18	0,964
Öz-düzenleme	Kontrol	0,153	21	0,200	0,939	21	0,207
Son-Testi	Deney	0,185	18	0,107	0,918	18	0,121

Tablo 3.12’de gösterilen Shapiro-Wilk testinin verileri incelendiğinde öz-düzenleme ön-testi kontrol grubu $p= 0,422$ ve deney grubu $p= 0,964$ değerleri ile öz-düzenleme son-testi kontrol grubu $p= 0,207$ ve deney grubu $p= 0,121$ değerleri elde edilmiştir. Elde edilen değerlere göre deney ve kontrol gruplarının normal dağılım gösterdiği görülmektedir.

Tablo 3.13 Öz-düzenleme testi çarpıklık ve basıklık değerleri

Test	Grup	Ortalama	ÇK	BK
Öz-düzenleme Ön-Testi	Kontrol	43,07	-0,27	-0,98
	Deney	61,17	-0,36	0,25
Öz-düzenleme Son-Testi	Kontrol	50,34	-0,90	0,15
	Deney	63,22	-0,37	-0,94

Tablo 3.13’de yer alan çalışma gruplarının çarpıklık katsayısı ve basıklık katsayısı değerleri incelendiğinde programlama öz-yeterliği ön-testi deney grubu (ÇK= -0,36 ve BK= 0,25) ve kontrol grubu (ÇK= -0,27 ve BK= -0,98) ve programlama öz-yeterliği son-testi deney grubu (ÇK= -0,37 ve BK=-0,94) ve kontrol grubu (ÇK= -0,90 ve BK= 0,15) değerleri incelendiğinde verilerin normal dağılım gösterdiği görülmektedir.

Çalışma gruplarının bazı test puanlarının normal dağılım bazı testlerin normal olmayan dağılım göstermesi üzerine verilerin analizini yapma amacıyla hem non-parametrik hem de parametrik testler kullanılmıştır. Çalışma gruplarının ön-test ve son-test puanları arasındaki farkı bulabilmek için non-parametrik test olarak Wilcoxon testi, parametrik test olarak bağımlı ilişkili t-testi ile deney ve kontrol grupları arasındaki son-test puanlarının farkını tespit etmek amacıyla non-parametrik test olarak Mann-Whitney U testi ve parametrik test olarak bağımsız ilişkili t-test kullanılmıştır.

3.7 Araştırmacının Rolü

Araştırmacı lisans eğitimini Bilgisayar ve Öğretim Teknolojileri Eğitimi alanında yapmış ve MEB bünyesinde dört yıldır bilişim teknolojileri öğretmeni olarak görev yapmaktadır. Araştırmacı deney sürecinin uygulanmasını görev yaptığı okulda kendisi yürütmüştür. Araştırmacı-öğretmenin bu çalışmadaki görevleri aşağıda açıklanmıştır:

- Araştırmanın yapılacağı okulun bağlı olduğu İl Milli Eğitim Müdürlüğünden izinleri almak.
- Katılımcıları gönüllü olarak çalışmaya dahil etmek ve katılımcılar için gerekli izinleri almak.
- Deney sürecini araştırmacı-öğretmenin derslerinde ve haftada iki saat yürütmek.
- Deney ve kontrol gruplarına uygulamak amacıyla ders planları oluşturmak.
- Deney sürecinde uygulanacak eğitim etkinliklerini tasarlamak ve uygulamak.
- Derslerin uygulamasına yönelik Bilişim Teknolojileri sınıfını düzenlemek.
- Araştırmada katılımcıların başarı düzeylerini ortaya çıkarmak amacıyla “akademik başarı testi” hazırlamak ve pilot çalışmasını yapmak.
- Akademik Başarı Testinin geçerlik ve güvenirlik çalışmalarını yapmak.

- Katılımcılara “Programlama Öz-yeterliği”, “Algılanan Öz-düzenleme” ölçeklerini ve “Akademik Başarı Testi”ni uygulamak.
- Araştırmanın geçerliğini ve güvenirliğini sağlamak.
- Verileri toplamak ve analiz etmek.
- Araştırmanın raporlamasını yapmak.

3.8 Geçerlik ve Güvenirlik Çalışmaları

Geçerlik, ölçme aracının ölçmeye çalıştığı özelliklere dışarıdaki değişkenleri karışmaması ve amaca uygun ölçümler yapılmasıdır. Geçerlik artırabilmek amacıyla çeşitli geçerlik belirleme yolları belirlenmiştir (Yurdabakan, 2022). Bu araştırma kapsamında geçerlik belirleme amacıyla çalışmalar açıklanmıştır. Kapsam geçerliği, ölçme aracının ölçmesi hedeflenen davranışlar bütünü temsil etmesidir (Yurdabakan, 2022). Bu çalışma kapsamında araştırmacı tarafından öğrencilerin programlama başarılarını ölçmek amacıyla “akademik başarı testi” geliştirilmiştir. Akademik başarı testi geliştirme sürecinde öğrencilerin eğitim sürecinde kazanması beklenen hedef ve davranışlar, BTY dersi öğretim programı taranarak belirlenmiştir. Öğrencilerin programlama hedef ve davranışlarında hangilerini, ne kadar kazandığını ölçmek amacıyla kazanım ve soru dağılımını gösteren belirtke tablosu hazırlanmıştır. Belirtke tablosuna ilişkin detaylar Tablo 4.3’de gösterilmiştir. Testin geçerliğini belirlemek amacıyla, testin hazırlanması sürecinde alanda uzman iki akademisyen, bir Türkçe öğretmeni, üç Bilişim Teknolojileri öğretmeni görüşleri alınmıştır. Geçerlik artırmak amacıyla pilot çalışmalar yapılmıştır. Araştırmanın yapıldığı okuldaki bir eğitim dönemi önce BTY dersine katılmış olan 7.sınıftaki 34 öğrenciye hazırlanan akademik başarı testi uygulanmıştır. Pilot çalışma sonrasında soruların madde güçlükleri ve madde ayırt edicilikleri analiz edilerek Tablo.3.4’te gösterilmiştir.

Çalışmanın geçerliğini artırmak amacıyla yapılan diğer çalışmalar:

- Deneyi uygulama sürecinde derse düzenli olarak devamsızlık yapan ve araştırmaya katılmak istemeyen öğrenciler araştırmaya dahil edilmemiştir.

- Veri toplama sürecinde ön-teste katılıp son-teste katılmayan ve ön-teste katılmayan öğrencilerin verileri araştırmadan çıkarılmıştır.
- Katılımcı sayısı, sınıf düzeyleri, araştırma modeli *3.1 Araştırma Modeli* başlığı altında açıklanmıştır.
- Deney sürecine kılavuzluk yapması amacıyla ders planı, günlük planlar hazırlanmıştır ve 3.5 Uygulama Süreci başlığı altında detaylı olarak açıklanmıştır.

Güvenirlik, ölçme aracının ölçmek istenilen özellikleri hata karışmadan ve ölçme tekrarlandığında tutarlı olarak ölçmesidir (Büyüköztürk vd., 2008). Sıklıkla kullanılan güvenilirlik belirleme yöntemlerinin başında KR20, KR21 ve Cronbach Alfa gibi tutarlık hesaplanmasını sağlayan yöntemler yer almaktadır (Büyüköztürk, 2008; Yurdabakan, 2022). Araştırmacı tarafından hazırlanan “akademik başarı testi”nin güvenilirliğini belirleyebilmek amacıyla Cronbach Alfa katsayısı hesaplanmış, sonuç 0,72 bulunmuştur. Güvenirlik kat sayısı 1,00 değerine yaklaştıkça güvenilirlik artar. Testten çıkarılması gereken maddeler atıldıktan sonra hesaplanan nihai güvenilirlik katsayısı 0,77 olarak hesaplanmıştır. Bu doğrultuda test yeterince güvenilir olarak kabul edilebilir (Karaca, 2022). Programlama öz-yeterliği ölçeğinin güvenilirliği ve geçerliği Kukul vd. (2017) tarafından hesaplanmış cronbach alpha güvenilirlik katsayısı 0,95 olarak bulunmuştur. Algılanan öz-düzenleme testinin güvenilirlik ve geçerlilik hesaplamaları Arslan ve Gelişli (2015) tarafından yapılmış ve cronbach alfa iç tutarlılık güvenilirlik katsayıları ölçeğin bütünü için 0,90; açık olma alt ölçeği için 0,84; arayış alt ölçeği için 0,82 olarak bulunmuştur.

Çalışmanın güvenilirliğini artırmak amacıyla yapılan diğer çalışmalar:

- Veri toplama araçlarının madde sayısı, cevaplama süresi, puan değeri ve nasıl cevaplanacağı ile ilgili yönerge hazırlanmıştır.
- Ölçeklerin anlaşılabilirliği için uzman görüşü alınmıştır.
- Akademik başarı testinin cevap anahtarı hazırlanmıştır.

- Ölçekler her öğrenciye aynı koşullar altında uygulanmıştır.
- Ön-test ve son-testler her öğrenciye aynı zaman aralıklarında uygulanmıştır.

Programlama öz-yeterliği ve algılanan öz-düzenleme ölçeklerinin geçerliği ve güvenirliği üzerine çalışmalar ölçekleri geliştiren araştırmacılar tarafından yapılmıştır. Ölçeklerin geçerlik ve güvenirliklerine ilişkin detaylar 3.3 Araştırma Verilerinin Toplanması başlığı altında açıklanmıştır.



4. BULGULAR

Bu başlık altında, verilerin ön-test puan ortalamaları, son-test puan ortalamaları ve araştırma sorularına ait bulgular yer almaktadır.

4.1 Çalışma Gruplarının Ön-Test Puanlarına Ait Betimsel İstatistik Bulguları

Çalışma gruplarına işlem öncesinde uygulanan ön-testlerin öğrenci puan ortalamalarına ait istatistikler Tablo 4.1’de gösterilmiştir:

Tablo 4.1 Çalışma grupları ön-test puan ortalamaları

Testler	Gruplar	N	$\bar{X}$	Maks. Puan	Mini. Puan	SS	SH
Akademik Başarı Ön-Testi	Kontrol	21	12,71	24	0	5,02	1,18
	Deney	18	12,83	24	0	5,02	1,18
Öz-düzenleme Ön-Testi	Kontrol	21	43,07	80	16	9,92	2,16
	Deney	18	61,17	80	16	9,32	2,19
Programlama Öz-yeterliği Ön-Testi	Kontrol	21	68,00	155	31	14,52	3,17
	Deney	18	73,36	155	31	16,09	3,79

Programlama akademik başarı testi ön-test puan ortalamaları deney grubunda $\bar{x}= 12,83$ ve kontrol grubunda $\bar{x}= 12,71$ olarak bulunmuştur (Tablo 4.1). Bu durumda her iki grubun da akademik başarısının seviyesinin ortalama düzeyde ve programlama başarısının işbirlikli öğretim uygulanmadan önce birbirine yakın oldukları görülmektedir. Öz-düzenleme ölçeği ön-test puan ortalamaları incelendiğinde deney grubunda $\bar{x}= 61,17$ ve kontrol grubunda $\bar{x}= 43,07$ olarak bulunmuştur. Bu sonuca göre deney grubunun öz-düzenleme becerisi düzeyi yüksek, kontrol grubun düzeyi ise orta düzeyde bulunmuştur. Çalışma gruplarının öz-düzenleme beceri düzeylerinin işbirlikli öğretim uygulanmadan önce deney grubu lehine olduğunu göstermektedir. Programlama öz-yeterliği ölçeği puan ortalamaları incelendiğinde ise deney grubunda $\bar{x}= 73,36$ ve kontrol grubunda $\bar{x}= 68,00$ olarak bulunmuştur. İki grubun da programlama öz-yeterliklerinin düşük düzeyde olduğu görülmektedir. İşbirlikli öğretim uygulanmadan önce programlama öz-yeterliği ortalamaların birbirine çok

yakın olmadığı ve programlama öz-yeterliğinin deney grubu lehine olduğu görülmektedir.

4.2 Çalışma Gruplarının Son-Test Puanlarına Ait Betimsel İstatistik Bulguları

Çalışma gruplarına uygulanan son-testlerin öğrenci puan ortalamalarına ait istatistikler Tablo 4.2’de gösterilmiştir.

Tablo 4.2 Çalışma gruplarının son-test puan ortalamaları

Testler	Gruplar	N	$\bar{X}$	Maks Puan	Mini Puan	SS	SH
Akademik Başarı Son-Testi	Kontrol	21	14,71	24	0	5,23	1,14
	Deney	18	17,83	24	0	2,99	0,70
Öz-düzenleme Son-Testi	Kontrol	21	50,34	80	16	8,92	1,94
	Deney	18	63,22	80	16	9,44	2,22
Programlama Öz-yeterliği Son-Testi	Kontrol	21	99,34	155	31	17,99	3,92
	Deney	18	135,04	155	31	15,54	3,66

Programlama akademik başarı testi son-test puan ortalamaları deney grubunda $\bar{x}$ = 17,83 ve kontrol grubunda $\bar{x}$ = 14,71 olarak bulunmuştur (Tablo 4.2). Bu bulgu her iki gruba işbirlikli öğretim uygulandıktan sonra programlama başarısının deney grubu lehine olduğunu göstermektedir. Deney grubunun akademik başarı düzeyinin yüksek, kontrol grubunun akademik başarısının orta düzeyde olduğu görülmektedir. Öz-düzenleme ölçeği son-test puan ortalamaları incelendiğinde deney grubunda $\bar{x}$ = 63,22 ve kontrol grubunda $\bar{x}$ = 50,34 olarak bulunmuştur. Deney grubunun öz-düzenleme becerisi düzeyi yüksek, kontrol grubunun ise orta düzeyde olduğu görülmektedir. Bu sonuç ise çalışma gruplarının öz-düzenleme beceri düzeylerinin iş birlikli öğretim uygulandıktan sonra deney grubu lehine olduğunu göstermektedir. Programlama öz-yeterliği ölçeğinin son-test puan ortalamaları incelendiğinde deney grubunda $\bar{x}$ = 135,04 ve kontrol grubunda $\bar{x}$ = 99,34 olarak bulunmuştur. Deney grubunun programlama öz-

yeterlik düzeyinin çok yüksek, kontrol grubunun ise orta düzeyde olduğu görülmüştür. İş birlikli öğretim uygulandıktan sonra programlama öz-yeterliği ortalamalarının deney grubu lehine olduğu görülmektedir.

4.3 Programlama Öz-Yeterliği Bulgularının Çalışma Gruplarına Göre Karşılaştırılması

Çalışmanın birinci araştırma sorusu olan *deney grubu öğrencileri ile kontrol grubu öğrencilerinin programlama öz-yeterliği ön-test puanları arasında anlamlı bir farklılık var mıdır?* doğrultusunda bağımsız örneklem t-testi yapılmıştır. Araştırma sorusuna ait bulgular Tablo 4.3'te gösterilmiştir.

Tablo 4.3 Çalışma gruplarının programlama öz-yeterliği ön-testine ait bağımsız örneklem t-testi sonuçları

Grup	N	$\bar{X}$	SS	t	df	P
Deney	18	73,36	16,06	1,07	37	0,29
Kontrol	21	68	14,53			

Tablo 4.3 incelediğinde, deney sürecinden önce programlama öz-yeterliği düzeyi ortalamalarına göre kontrol grubunun ortalaması ($\bar{X}=68$) ve deney grubunun ortalaması ($\bar{X}=73,36$) arasında anlamlı farklılık bulunmamaktadır, $t(37)=1,07$; $P > 0,05$. Bu sonuç çalışma gruplarının arasında öz-yeterlik değişkeni açısından istatistiksel olarak anlamlı bir fark olmadığını göstermektedir. Programlama öz-yeterliği ön-testi sonuçlarına ilişkin etki büyüklüğü Cohen's $d = 0,02$ olarak hesaplanmıştır. Bu sonuç çalışma grupları arasında küçük bir etki büyüklüğünü ve güçlü farklılık olmadığını göstermektedir. Bu bulgu deney sürecinden önce öz-yeterlik algıları bakımından çalışma gruplarının denk olduğunu göstermektedir.

Çalışmanın ikinci araştırma sorusu olan *deney grubu öğrencileri ile kontrol grubu öğrencilerinin programlama öz-yeterliği son-test puanları arasında anlamlı bir farklılık var mıdır?* doğrultusunda mann-whitney u testi yapılmıştır. İkinci araştırma sorusuna ait bulgular Tablo 4.4'te gösterilmiştir.

Tablo 4.4 Çalışma gruplarının programlama öz-yeterliği son-testine ait mann-whitney u sonuçları

Grup	N	$\bar{X}$	SO	ST	U	Z	P
Deney	18	135,04	28,92	520,50	28,50	-4,52	0,001
Kontrol	21	99,34	12,36	259,50			

Tablo 4.4 incelendiğinde, deney sürecinden sonra programlama öz-yeterliği düzeyleri bakımından deney grubunun ortalamasının ($\bar{X}= 135,04$); kontrol grubunun ortalamasına ($\bar{X}= 99,34$) göre yüksek olduğu görülmektedir. Ayrıca, deney ve kontrol grubu öğrencilerinin programlama öz-yeterliği son-test puanları istatistiksel olarak deney grubu lehine anlamlı farklılık göstermektedir, $U= 28,50$; $P=0,001$. Programlama öz-yeterliği son-testi sonuçlarına ilişkin etki büyüklüğü $-0,69$ olarak bulunmuştur. Bu sonuç çalışma grupları arasında büyük bir etki büyüklüğü ve çalışma grupları arasında belirgin ve güçlü bir farklılık olduğunu göstermektedir.

4.4 Öz-Düzenleme Becerisi Bulgularının Çalışma Gruplarına Göre Karşılaştırılması

Çalışmanın üçüncü araştırma sorusu olan, *deney grubu öğrencileri ile kontrol grubu öğrencilerinin öz-düzenleme becerileri ön-test puanları arasında anlamlı bir farklılık var mıdır?* doğrultusunda bağımsız örneklem t-testi yapılmıştır. Üçüncü araştırma sorusuna ait bulgular Tablo 4.5'te gösterilmiştir.

Tablo 4.5 Algılanan öz-düzenleme ön-testinin çalışma gruplarına ait bağımsız örneklem t-testi sonuçları

Grup	N	$\bar{X}$	Ss	t	df	P
Deney	18	61,17	9,42	5,74	37	0,001
Kontrol	21	43,07	9,89			

Tablo 4.5 incelediğinde, deney sürecinden önce çalışma gruplarının öz-düzenleme algısı ortalamalarına göre kontrol grubunun ortalaması ($\bar{X}= 43,07$) ve deney grubunun ortalaması ($\bar{X}= 61,17$) arasında anlamlı farklılık bulunmaktadır, $t(37)= 5,74$; $P<0,05$. Öz-düzenleme ön-testi sonuçlarına ilişkin etki büyüklüğü Cohen's $d = 1,87$ olarak hesaplanmıştır. Bu sonuç çalışma grupları arasında büyük bir etki büyüklüğünü ve güçlü farklılık olduğunu göstermektedir.

Çalışmanın dördüncü araştırma sorusu olan, *deney grubu öğrencileri ile kontrol grubu öğrencilerinin öz-düzenleme becerileri son-test puanları arasında anlamlı bir farklılık var mıdır?* doğrultusunda bağımsız örneklem t-testi yapılmıştır. Dördüncü araştırma sorusuna ait bulgular Tablo 4.6’da gösterilmiştir.

Tablo 4.6 Algılanan öz-düzenleme ölçeği son-testinin çalışma gruplarına ait bağımsız örneklem t-testi sonuçları

Grup	N	SS	$\bar{X}$	t	df	P
Deney	18	9,51	63,32	4,27	37	0,001
Kontrol	21	8,93	50,34			

Tablo 4.6 incelediğinde, deney sürecinden sonra öz-düzenleme algıları ortalamalarına göre kontrol grubunun ortalaması ($\bar{X}$ = 50,34) ve deney grubunun ortalaması ($\bar{X}$ = 63,32) arasında anlamlı farklılık bulunmaktadır, $t(37)= 4,27$; $P<0,05$. Öz-düzenleme son-test sonuçlarına ilişkin etki büyüklüğü Cohen's $d = 1,41$ olarak hesaplanmıştır. Bu sonuç çalışma grupları arasında büyük bir etki büyüklüğünü ve güçlü farklılık olduğunu göstermektedir.

4.5 Akademik Başarı Testi Bulgularının Çalışma Gruplarına Göre Karşılaştırılması

Çalışmanın beşinci araştırma sorusu olan, *deney grubu öğrencileri ile kontrol grubu öğrencilerinin akademik başarı ön-test puanları arasında anlamlı bir farklılık var mıdır?* doğrultusunda bağımsız örneklem t-testi yapılmıştır. Araştırma sorusuna ait bulgular Tablo 4.7’de gösterilmiştir.

Tablo 4.7 Çalışma gruplarına göre akademik başarı ön-testine ait bağımsız örneklem t-testi sonuçları

Grup	N	SS	$\bar{X}$	t	df	P
Deney	18	4,82	12,83	0,39	37	0,69
Kontrol	21	4,19	12,71			

Tablo 4.7 incelediğinde programlama başarı ön-testi ortalama düzeyleri bakımından kontrol grubu ortalaması ($\bar{X}$ = 12,71) ve deney grubunun ortalaması ($\bar{X}$ = 12,83) arasında farklılık olmadığı gözlemlenmektedir. Ayrıca deney ve kontrol grubu

öğrencilerinin programlamaya yönelik akademik başarı ön-test bulgularına göre deney ve kontrol grupları arasında istatistiksel olarak anlamlı fark ortaya çıkmamıştır, $t(37)=0,39$; $P>0,05$. Bu durum deney sürecinden önce çalışma gruplarının programlama başarısına yönelik hazır bulunuşluk bakımından denk olduğunu göstermektedir. Akademik başarı ön-test sonuçlarına ilişkin etki büyüklüğü Cohen's $d = 0,027$ olarak hesaplanmıştır. Bu sonuç çalışma grupları arasında çok küçük bir etki büyüklüğü olduğunu göstermekte ve grupların denk olduğunu göstermektedir.

Çalışmanın altıncı araştırma sorusu olan, *Deney grubu öğrencileri ile kontrol grubu öğrencilerinin akademik başarı son-test puanları arasında anlamlı bir farklılık var mıdır?* doğrultusunda mann-whitney u testi yapılmıştır. Altıncı araştırma sorusuna ait bulgular Tablo 4.8'de gösterilmiştir.

Tablo 4.8 Çalışma gruplarına göre akademik başarı son-testine ait mann-whitney u sonuçları

Grup	N	$\bar{X}$	SO	ST	U	Z	P
Deney	18	17,83	23,78	428	121	-1,92	0,054
Kontrol	21	14,71	16,76	352			

Tablo 4.8 incelediğinde deney sürecinden sonra deney grubunun programlama başarısı puan ortalamasının ($\bar{X}= 17,83$); kontrol grubunun programlama başarısı puan ortalamasına ($\bar{X}= 14,71$) göre daha yüksek olduğu görülmektedir. Ancak, deney ve kontrol grubu öğrencilerinin programlamaya yönelik akademik başarı son-test bulguları, deney grubu lehine istatistiksel olarak anlamlı farklılık göstermemektedir, $U=121$; $P= 0,054$. Akademik başarı son-test sonuçlarına ilişkin etki büyüklüğü Cohen's $d= -0,31$ olarak hesaplanmıştır. Bu sonuç çalışma grupları arasında orta düzeyde bir etki büyüklüğünü ve çok güçlü olmayan farklılaşma olduğunu göstermektedir.

4.6 Deney Grubu Ön-Test ve Son-Test Bulgularının Karşılaştırılması

Çalışmanın yedinci araştırma sorusu olan, *deney grubu öğrencilerinin akademik başarı testi ön-test ve son-test puanları arasında anlamlı fark var mıdır?* doğrultusunda wilcoxon işaretli sıralar testi yapılmıştır. Araştırma sorusuna ait bulgular Tablo 4.9'da gösterilmiştir.

Tablo 4.9 Deney grubu akademik başarı wilcoxon işaretli sıralar testine ait betimsel istatistikler

Test	N	$\bar{X}$	SS	Medyan
Akademik Başarı Ön-test	18	12,83	5,02	13,50
Akademik Başarı Son-test	18	17,83	2,99	19

Tablo 4.9 incelediğinde deney grubu akademik başarı son-test ortalamasının ($\bar{X}$ = 17,83), akademik başarı ön-test ortalamasına ($\bar{X}$ = 12,83) göre yüksek olduğu ortaya çıkmıştır.

Tablo 4.10 Deney grubu akademik başarı wilcoxon işaretli sıralar testine ait sonuçları

Testler	Sıralar	N	SO	ST	Z	P
Akademik Başarı	Negatif sıra	1	1,50	1,50		
Ön-testi ve	Pozitif sıra	15	8,97	134,50		
Akademik Başarı	Eşit	2			-3,45	<0,001
Son-testi	Toplam	18				

Tablo 4.10 incelediğinde deney grubu akademik başarı son-testinin akademik başarı ön-testine göre istatistiksel olarak anlamlı farklılık gösterdiği görülmektedir, $Z = -3,45$; $P = 0,001$. Deney grubu akademik başarı ön-test ve son-test sonuçlarına ilişkin etki büyüklüğü Cohen's $d = -0,81$ olarak hesaplanmıştır. Bu sonuca göre deney grubunun ön-test ve son-test sonuçları arasında büyük ve güçlü bir etki olduğu görülmektedir.

Çalışmanın sekizinci araştırma sorusu olan, *Deney grubu öğrencilerinin programlama öz-yeterliği ön-test ve son-test puanları arasında anlamlı fark var mıdır?* doğrultusunda wilcoxon işaretli sıralar testi yapılmıştır. Araştırma sorusuna ait bulgular Tablo 4.11'da gösterilmiştir.

Tablo 4.11 Deney grubu programlama öz-yeterliği wilcoxon işaretli sıralar testine ait betimsel istatistikler

Test	N	$\bar{X}$	SS	Medyan
Programlama Öz-yeterliği Ön-testi	18	73,36	16,09	73,50
Programlama Öz-yeterliği Son-testi	18	135,04	15,54	140

Tablo 4.11 incelendiğinde deney grubu öğrencilerinin programlama öz-yeterliği son-test ortalamasının ($\bar{X}= 135,04$) programlama öz-yeterliği ön-test ortalamasından ($\bar{X}= 73,36$) yüksek olduğu görülmektedir.

Tablo 4.12 Deney grubu programlama öz-yeterliği wilcoxon işaretli sıralar testine ait sonuçları

Testler	Sıralar	N	SO	ST	Z	P
Programlama Öz-yeterliği Ön-testi	Negatif sıra	0	0,00	0,00	-3,72	<0,001
Programlama Öz-yeterliği Son-testi	Pozitif sıra	18	9,50	171		
Programlama Öz-yeterliği	Eşit	0				
	Toplam	18				

Tablo 4.12 incelendiğinde deney grubunun programlama öz-yeterliği ön-testi (Medyan= 73,50) ve son-testi (Medyan= 140) arasında son-test lehine istatistiksel olarak anlamlı farklılık olduğu görülmektedir, $Z=-3,72$, $P<,050$. Deney grubu programlama öz-yeterliği ön-test ve son-test sonuçları etki büyüklüğü Cohen's $d=-0,88$ olarak hesaplanmıştır. Bu sonuç deney grubunun programlama öz-yeterliği ön-testi ve son-testi arasında büyük etki düzeyi ve güçlü bir farklılık olduğunu göstermektedir.

Çalışmanın dokuzuncu araştırma sorusu olan, *deney grubu öğrencilerinin öz-düzenleme becerileri ön-test ve son-test puanları arasında anlamlı fark var mıdır?* doğrultusunda ilişkili örneklem t-testi yapılmıştır. Dokuzuncu araştırma sorusuna ait bulgular Tablo 4.13 gösterilmiştir.

Tablo 4.13 Deney grubu öz-düzenleme ilişkili örneklem t-testine ait sonuçlar

Test	N	$\bar{X}$	SS	t	df	P
Öz-düzenleme Ön-test	18	61,17	9,32	-1,07	17	0,300
Öz-düzenleme Son-test	18	63,22	9,44			

Tablo 4.13 incelendiğinde deney grubu öğrencilerinin öz-düzenleme ön-test ($\bar{X}= 61,17$) ve öz-düzenleme son-test ($\bar{X}= 63,22$) puanları arasında anlamlı farklılık görülmemektedir, $t(17)= -1,07$; $P> 0,050$. Deney grubu öz-düzenleme ön-test ve son-test sonuçlarına ilişkin etki büyüklüğü Cohen's $d=0,22$ olarak hesaplanmıştır. Bu

sonuç ön-test ve son-test arasındaki etki büyüklüğünün düşük etkili ve farklılaşmanın az olduğunu göstermektedir.

4.7 Kontrol Grubu Ön-Test ve Son-Test Bulgularının Karşılaştırılması

Çalışmanın onuncu araştırma sorusu olan, *kontrol grubu öğrencilerinin akademik başarı testi ön-test ve son-test puanları arasında anlamlı fark var mıdır?* doğrultusunda ilişkili örneklem t-testi yapılmıştır. Araştırma sorusuna ait bulgular Tablo 4.14’te gösterilmiştir.

Tablo 4.14 Kontrol grubu akademik başarı ilişkili örneklem t-testine ait sonuçlar

Test	N	$\bar{X}$	SS	t	df	P
Akademik Başarı Ön-test	21	12,71	4,19	-2,4	20	0,025
Akademik Başarı Son-test	21	14,71	5,23			

Tablo 4.14 incelendiğinde kontrol grubu öğrencilerinin akademik başarı ön-test ($\bar{X}$ = 12,71) ve akademik başarı son-test ($\bar{X}$ = 14,71) puanları arasında son-test lehine istatistiksel olarak anlamlı farklılık olduğu görülmektedir, $t(20)$ = -2,41; $P<0,050$. Kontrol grubu akademik başarı ön-test ve son-test sonuçlarına ilişkin etki büyüklüğü Cohen’s d = 0,42 olarak hesaplanmıştır. Bu sonuç akademik başarı ön-testi ve son-testi arasında orta düzeyde bir etki büyüklüğünü ve farklılaşma olduğunu göstermektedir.

Çalışmanın on birinci araştırma sorusu olan, *kontrol grubu öğrencilerinin programlama öz-yeterliği testi ön-test ve son-test puanları arasında anlamlı fark var mıdır?* doğrultusunda ilişkili örneklem t-testi yapılmıştır. Araştırma sorusuna ait bulgular Tablo 4.15’te gösterilmiştir.

Tablo 4.15 Kontrol grubu Programlama öz-yeterliği ilişkili örneklem t-testine ait sonuçlar

Test	N	$\bar{X}$	SS	t	df	P
Programlama Öz-yeterliği Ön-test	21	68	14,52	-11,62	20	0,001
Programlama Öz-yeterliği Son-test	21	99,34	17,99			

Tablo 4.15 incelendiğinde kontrol grubu öğrencilerinin programlama öz-yeterliği ön-test ($\bar{X}$ = 77,47) ve programlama öz-yeterliği son-test ($\bar{X}$ = 97,53) puanları arasında son-

test lehine anlamlı farklılık olduğu görülmektedir, $t(20) = -11,62$; $P < 0,050$. Kontrol grubu programlama öz-yeterliği ön-test ve son-test sonuçlarına ilişkin etki büyüklüğü Cohen's $d = 1,91$ olarak hesaplanmıştır. Bu sonuç kontrol grubunun programlama öz-yeterliği ön-testi ve son-testi arasında büyük bir etki büyüklüğü olduğunu göstermektedir.

Çalışmanın on ikinci araştırma sorusu olan, *kontrol grubu öğrencilerinin öz-düzenleme becerileri ön-test ve son-test puanları arasında anlamlı fark var mıdır?* doğrultusunda ilişkili örneklem t-testi yapılmıştır. Araştırma sorusuna ait bulgular Tablo 4.16'da gösterilmiştir.

Tablo 4.16 Kontrol grubu algılanan öz-düzenleme ilişkili örneklem t-testine ait sonuçlar

Test	N	$\bar{X}$	SS	t	df	P
Öz-düzenleme Ön-test	21	43,07	9,92	-4,01	20	0,001
Öz-düzenleme Son-test	21	50,34	8,92			

Tablo 4.16 incelendiğinde kontrol grubu öğrencilerinin öz-düzenleme ön-test ($\bar{X} = 43,07$) ve öz-düzenleme son-test ($\bar{X} = 50,34$) puanları arasında son-test lehine anlamlı farklılık olduğu görülmektedir, $t(20) = -4,01$; $P < 0,050$. Kontrol grubu öz-düzenleme ön-test ve son-test sonuçlarına ilişkin etki büyüklüğü Cohen's $d = 0,77$ olarak hesaplanmıştır. Bu sonuç kontrol grubunun öz-düzenleme ön-testi ve son-testi arasında etki büyüklüğünün büyük etki olduğunu ve testler arasında belirgin farklılaşma olduğu görülmektedir.

5. SONUÇ VE TARTIŞMA

Bu çalışmada işbirlikli öğrenme teknikleriyle programlama öğretimi etkinliklerinin öğrencilerin programlama öz-yeterlikleri, öz-düzenleme becerileri ve akademik başarıları üzerinde etkisi olup olmadığı incelenmiştir. Bu bölümde programlama akademik başarısına yönelik bulgular, programlama öz-yeterliliğine yönelik bulgular ve öz-düzenleme becerisine yönelik bulgular tartışılmıştır.

5.1 Programlama Akademik Başarısına Yönelik Tartışma

İşbirlikli öğrenme yaklaşımıyla programlama öğretiminin ortaokul öğrencilerinin programlama akademik başarısına etkisinin olup olmadığının ortaya çıkarılmasını amaçlayan araştırma sorusu deney grubuna uygulanan ön-test ve son-test puanları incelendiğinde deney grubunun son-test ortalamasında yükseliş olduğunu göstermiştir. Bu artışın ön-test ve son-test arasında son-test lehine anlamlı farklılık ve büyük etki büyüklüğü olduğu bulunmuştur. Benzer bir çalışmada Garcia (2021) işbirlikli öğretim tekniklerinden biri olan jigsaw tekniğinin programlama öğretimine etkisini incelemiş ve jigsaw tekniği kullanılan deney grubunun programlama performansı daha yüksek bulunmuştur. Yılmaz (2019) ise BTY dersinde programlama öğretiminde birlikte öğrenme tekniğinin kullanılmasının programlama öğretiminin akademik başarısını olumlu etkilediği sonucunu elde etmiştir. Gelici ve Bilgin (2011)'e göre de işbirlikli öğrenme öğrenci-öğrenci etkileşimini arttırması sebebiyle ders içerisinde konuları anlamayan veya derste sorun çıkaran öğrencilere müdahale etme fırsatı sağlayarak başarı arttırabilir. Bayrakçı vd. (2013) öğrenme gruplarında bireysel çalışmalardan ziyade grup çalışmalarındaki işbirlikli çabanın başarıyı yüksek düzeye çıkardığına vurgu yapmaktadır. Buna göre, işbirlikli öğrenme yaklaşımı ile programlama öğretimi etkinliklerinde deney sürecinden önce düşük başarı seviyeli olarak değerlendirilen öğrencilerin, öğretim süreçlerinde yüksek başarı seviyeli öğrenciler ile etkileşime girmesi, onları model alması ve konuları tartışarak ele almaları düşük başarı seviyeli öğrencilerin akademik başarı seviyelerini yükselttiği söylenebilir.

Kontrol grubunun ön-test ve son-test puan ortalamaları kıyaslandığında son-test puanlarında artış, testler arasında son-test lehine arasında anlamlı fark ve orta düzeyde

etki görülmüştür. Bunun nedeni ders etkinliklerinde kullanılmış olan blok tabanlı scratch uygulaması olabilir. Blok tabanlı scratch uygulamasının arayüzü sade ve çocuklar tarafından eğlenceli olarak görülmektedir (Kert ve Uğraş, 2009). Scratch arayüzünün tasarımı ile öğrencilerin oyunlar, hikayeler ve animasyonlar yapmasına olanak sağlayarak eğlenceli programlamayı eğlenceli hale getirir. Bu durum scratch uygulamasının ilkokuldan üniversiteye kadar her yaştan bireye algoritma ve programlama öğretmesini sağlamaktadır (Erol ve Kurt, 2017). Diğer yandan blok tabanlı programlama öğrenci başarısını artırmaktadır (Gürbüztürk ve Tanataş, 2024). Deniz (2021)'e göre farklı görsel programlamanın öğrencilerin başarılarını etkilediği görülmektedir. Eraytaç (2019) ortaokul beşinci sınıf öğrencilerinin BTY dersinde robotik kodlama eğitiminin blok tabanlı kodlama yöntemi ile öğretilmesinin öğrencilerin akademik başarısına etkisini yüksek bulmuştur. Bu çalışmadaki programlama öğretimi etkinliklerinde hem deney grubunda hem de kontrol grubunda blok tabanlı programlama uygulamaları olan scratch ve code.org kullanılmıştır. Blok tabanlı uygulamalarda yer alan kod bloklarının renkli olması, sürekli-bırak özelliğiyle yapboz parçaları şeklinde bir araya getirilmeleri öğrencilerin ilgisini çekmekte ve programlamayı öğretimini kolaylaştırmaktadır. Bu durumun kontrol grubunun son-testi lehine anlamlı farklılık görülmesinin sebebi olduğu düşünülmektedir.

Deney sürecinden önce grupların akademik başarılarını belirlemek üzere bir akademik başarı testi uygulanmıştır. Test sonuçları, çalışma gruplarının akademik başarı ön-testi ortalama düzeylerinin birbirine yakın olduğunu ve aralarında anlamlı farklılık olmadığını göstermiştir. Deney sonrasında ise akademik başarı açısından deney grubunun puan ortalamasının kontrol grubu ortalamasından yüksek olmasına karşın deney ve kontrol grubu son-testleri arasında istatistiksel olarak anlamlı fark çıkmamıştır. Demir ve Seferoğlu (2021) işbirlikli öğrenmenin programlamaya yansması olan eşli programlamayı, tekli programlama ile kodlama başarısı, akış deneyimi ve kodlama kalitesi bakımından karşılaştırmıştır ve sonuç olarak deney ve kontrol gruplarının kodlama başarılarının yükselmesine karşın deney grubu lehine farklılık ortaya çıkmamıştır. Bayrakçeken vd. (2013)'ne göre grup çalışmalarına öğrencilerin motive olması ve grup oluşturma süreçlerini benimsemesi zaman alabilir. Dikmen ve Tuncer (2020)'in öğrencilerin baskın öğrenme stilinin başarıyı etkilediğini savunmaktadır. İşbirlikli öğrenme süreçlerinde öğrenciler bazen grup olma olgusunu

tam kavrayamamakta ve bazı öğrenciler grup çalışmalarına karşı tepkisel yaklaşabilmektedir. Diğer yandan grup çalışmalarının diğer öğrenme yaklaşımlara üstün yönleri olmasına karşın bazı öğrencilerin bireysel öğrenme stillerinin işbirlikli öğrenme gruplarındaki akademik başarı performanslarını etkilediği düşünülmektedir. Bundan dolayı deney grubu ve kontrol grubu akademik başarı son-testleri arasında anlamlı fark çıkmadığı düşünülebilir.

Yukarıda yer alan programlama akademik başarısına yönelik tartışmalar doğrultusunda BTY dersi öğretim programlarında yer alan öğrenme ve öğretme içeriklerinin öğrencilerin akademik başarılarını artırıcı iş birliğine dayalı etkinliklerle düzenlenmesi gerektiği söylenebilir. Öğretim programları boyutunda ise ilk olarak, ortaokul düzeyindeki BTY dersinin öğretim programlarında işbirlikli öğrenme tekniklerinin yer alması söylenebilir. Ayrıca ikinci olarak, özel ve devlet okullarında görev yapan Bilişim Teknolojileri öğretmenlerine uygulanan mesleki hizmet içi eğitim programlarında ve Bilişim Teknolojileri öğretmeni yetiştiren lisans düzeyindeki öğretim programlarında, programlama başarısını artırmak amacıyla İşbirlikli Öğrenme Modeline yer verilmesi gerektiği söylenebilir.

5.2 Programlama Öz-Yeterliliğine Yönelik Tartışmalar

Çalışmada ikinci olarak işbirlikli öğrenme yaklaşımıyla programlama öğretiminin ortaokul öğrencilerinin programlama öz-yeterliliğine etkisinin olup olmadığı araştırılmıştır. Deney grubuna uygulanan ön-test ve son-test puanları incelendiğinde deney grubunda son-test lehine istatistiksel olarak anlamlı fark ve büyük etki büyüklüğü görülmüştür. Kontrol grubunun programlama öz-yeterliliğinde de ön-test ve son-test arasında son-test lehine istatistiksel olarak anlamlı fark ve büyük etki büyüklüğü görülmüştür. Uzun ve Baltalı (2020)'ya göre programlama öğretiminde görselliğin kullanılması öğrenci motivasyonunu ve başarısını başlıca etkileyen faktördür ve ilköğretim öğrencilerinin programlamaya karşını ilgisini de artırmaktadır. Programlama öğretimi sırasında hem deney hem de kontrol grubunda görsel programlama araçlarına ve materyallere yer verilmiş olması her iki grupta da anlamlı farkın çıkmasına neden olmuş olabileceği söylenebilir.

Deney grubunun programlama öz-yeterliği ön-test ve son-test puanları karşılaştırıldığında, son test lehine artış olması ve anlamlı farklılık görülmesi işbirlikli öğrenme yaklaşımının programlama öz-yeterliğini arttırmada öğretim programındaki öğretim yaklaşım ve tekniklerine göre daha etkili olduğu göstermektedir. Sosyal destek olarak bilinen aile ve arkadaş çevresinin öğrenci tarafından model alınması yoluyla öz-yeterlik algısı yükselmektedir (Polat, 2023). Deney grubunda, işbirlikli öğrenme gruplarındaki öğrencilerin aktif olarak öğrenme sürecine katılması ve birbirlerini model alarak birbirlerinden olumlu yönden etkilenmesi, deney grubunun programlama öz-yeterliği düzeylerinin son-test lehine anlamlı fark çıkmasına sebep olduğu söylenebilir.

Deney ve kontrol gruplarının programlama öz-yeterliği son-test bulguları karşılaştırıldığında test puanları deney grubu lehine istatistiksel olarak anlamlı fark ve testler arasında büyük etki göstermektedir. Hangün (2019)'e göre robotik kodlama eğitimi programlama öz-yeterliğini olumlu yönde etkilemektedir. Tiryaki (2020), robotik kodlama eğitiminin ortaöğretim öğrencilerinin programlama öz-yeterliğine etkisini incelemiş ve kodlamanın programlama öz-yeterliğini olumlu etkilediği sonucuna ulaşmıştır. Phillip ve Brooks (2017)'a göre öğrencilerin, kodlama (programlama) etkinliklerine katılması öz-yeterlik düzeylerini yükseltmektedir. Sırakaya (2018)'ya göre ise blok tabanlı uygulama olan scratch, öğrenmeyi kolaylaştırmakta ve öğrenci motivasyonunu artırmaktadır. Tsai (2019) de görsel programlama dilleriyle programlama öğretiminde öz-yeterliğin rolünü araştırmış ve sonuç olarak orta ve düşük öz-yeterliğe sahip öğrencilerde görsel programlamanın etkisini yüksek bulmuştur. Huang vd. (2025)'ne göre görsel tabanlı programlamanın öğrencilerin öz-yeterliğine etkisi olumludur. Diğer yandan, işbirlikli öğrenmedeki grup çalışmaları öğrencilerin motivasyonlarını artırmaktadır (Yıldız vd., 2020; Arslan ve Engin, 2021). İşbirlikli öğrenmenin en önemli avantajlarından birisi bireyleri sosyal, duygusal ve sosyal yönden geliştirmesidir. İşbirlikli öğrenme yaklaşımında grupların heterojen dağılması öğrencilerin etkileşimini artırır ve düşük başarı seviyeli öğrenciler yüksek başarı seviyeli öğrencileri model alırlar (Bayrakçeken vd., 2013). Deney grubu ile gerçekleştirilen işbirlikli öğrenme yaklaşımı ile kurgulanmış programlama derslerinde kolay ve renkli arayüze sahip olan blok tabanlı uygulamalarından olan scratch ve code.org kullanılması öğrencilerin, öğrenme süreci

içindeki motivasyonlarını olumlu etkilediği düşünülebilir. Diğer yandan programlama öğretimi sırasında işbirlikli öğrenme gruplarında bireylerin öğretim sürecine aktif olarak katılmaları, gruptaki tüm öğrencilerin etkileşim içinde olmaları, birbirilerini desteklemeleri ve düşük programlama öz-yeterlik seviyesine sahip öğrencilerin yüksek programlama öz-yeterlik seviyesine sahip öğrencileri model alması, öğrencilerin programlama öz-yeterlik düzeylerini etkileyen unsurlar olarak sıralanabilir. Bu nedenlerden dolayı deney grubu ve kontrol grubu programlama öz-yeterliği son-testleri arasında anlamlı fark çıkmış olduğu düşünülmektedir.

Yukarıda yer alan programlama öz-yeterliğine yönelik tartışmalar doğrultusunda BTY dersi öğretim programlarında yer alan öğrenme ve öğretme içeriklerinin öğrencilerin programlama öz-yeterlik inançlarını artırıcı etkinliklerle düzenlemesi gerektiğini ortaya koymaktadır. Öğretim programları boyutunda ise ilk olarak, ortaokul düzeyindeki BTY dersinin öğretim programlarında işbirlikli öğrenme tekniklerinin yer alması yerinde olacaktır. Ayrıca ikinci olarak, özel ve devlet okullarında görev yapan Bilişim Teknolojileri öğretmenlerine uygulanan mesleki hizmet içi eğitim programlarında ve Bilişim Teknolojileri öğretmeni yetiştiren lisans düzeyindeki öğretim programlarında yer alan programlama öğretim yaklaşımları arasında İşbirlikli Öğrenme Modeline yer verilmesi gerekliliği ortaya çıkmıştır.

5.3 Öz-Düzenleme Becerisine Yönelik Tartışmalar

İşbirlikli öğrenme yaklaşımıyla programlama öğretiminin ortaokul öğrencilerinin öz-düzenleme becerisine etkisinin olup olmadığının ortaya çıkarılmasına ilişkin araştırma sorusu kontrol grubu öz-düzenleme ön-test ve son-test arasında son-test lehine istatistiksel olarak anlamlı fark olduğunu ve testler arasında büyük bir etki büyüklüğü olduğu göstermiştir. Sarı ve Akınoğlu (2009)'na göre öz-düzenleme becerilerinin gelişmesi öğrencilerin bireysel çalışmalara odaklanması, kendi hedef ve öğrenmelerini düzenlemesi ile sağlanmaktadır. Kontrol grubunda programlama derslerinin bireysel etkinlikler yoluyla gerçekleştirilmesi, öğrenme süreci içerisinde öğrencilerin öz-düzenleme becerilerinden planlama, izleme, kontrol ve değerlendirme adımlarını daha fazla kullanmasını ve içselleştirmesini sağlamış olabilir.

Deney grubu öz-düzenleme ön-test ve son-test arasında da son-test lehine puan ortalamalarında artış olmasına karşın istatistiksel olarak anlamlı bir fark görülmemiştir ve testler arasında büyük bir etki büyüklüğü olduğu görülmektedir. Çoşğun ve Çoşğun (2018), programlama öğretiminin öz-düzenlemeli öğrenme becerilerini etkilemediğini ortaya koymuştur. Atalay ve Aydın (2015)'a göre öz-değerlendirme, öğrencilerin kendi çalışma süreçlerinin kalitesini değerlendirmesinin önemli bir yoludur. İşbirlikli programlama öğretimi etkinlikleri sırasında öğrencilerin kendilerini değerlendirebilmesi için ders sonlarında ne öğrendiklerini kısaca özetlemesi, kendisini ve arkadaşlarını değerlendirerek puanlaması istenmiştir. Bu doğrultuda işbirlikli öğrenme süreci içerisinde kullanılan öz-değerlendirme etkinliklerinin öğrencilerin öz-düzenleme becerilerini etkileyen unsurdur. Öğrenciler öz-değerlendirme etkinliklerini benimseyemedikleri ve objektif değerlendirme yapmaktan uzak bir değerlendirme anlayışı ile grup arkadaşlarını değerlendirmeleri deney grubu öz-düzenleme son-testi lehine anlamlı fark çıkmamasını sağlayan bir faktör olduğu söylenebilir.

Her iki grup karşılaştırıldığında, deney grubu ve kontrol grubu öz-düzenleme ölçeği son-test puanları arasında deney grubu lehine istatistiksel olarak anlamlı farklılık bulunmuştur. Bu durum, işbirlikli öğrenme yaklaşımı ile programlama öğretiminin öğrencilerin öz-düzenleme becerisini geliştirmede etkili olduğu sonucunu ortaya çıkarmaktadır. Aktan (2012)'ın öğrencilerin akademik başarıları ile öz-düzenleyici öğrenme becerileri, motivasyonları ve öğretmenlerin öğretim stili arasındaki ilişkiyi incelediği çalışmasında öğretmenlerin öğrencilerin öz-düzenleme stratejilerinin geliştirilmesinde önemli rol oynadığı sonucuna ulaşmıştır. Bayrakçeken vd. (2013)'ne göre işbirlikli öğrenme yaklaşımının doğasında var olan grup çalışmaları öğrencilerin etkileşimini artırarak sosyalleşmesini sağlamaktadır. Atalay ve Aydın (2015)'a göre etkili öz-düzenleme becerisine sahip öğrenciler yardım istemekten çekinmezler ve bu yardım talebi öğrencilerin sosyalliği ile de ilgilidir. Dönmez ve Gündoğdu (2018) işbirlikli öğrenme tekniği olan ayrılıp birleşmenin öz-düzenleme becerisine etkisini bulmuştur. Öz-düzenleme sürecinin planlama, izleme, kontrol ve değerlendirme adımları aynı şekilde bilgisayar programlama sürecinde de yer alan adımlardır. Ayrıca programlamanın öz-düzenleme yeterliklerini geliştirdiği bilinmektedir (Çoşğun ve Çoşğun, 2018). Loksa ve Ko (2016) programlama problemlerini çözmede programlama bilgisiyle birlikte öz-düzenleme becerisinin etkili olduğu sonucuna

ulařmıřtır. Öz-düzenlemeyi etkileyen başlıca unsur öğretmennin kullandığı öğrenme yaklaşımıdır. Deney grubunda uygulanmış olan işbirlikli yaklaşım öğrencilerin deney süreci içerisinde işbirlikli öğrenme gruplarında etkileşime girmelerini sağlamıştır. Ayrıca, deneysel süreçte yardımlaşarak yazdıkları kodları planlama, izleme, kontrol etme ve hata ayıklama gibi programlama adımlarını öğrenme sürecinde aktif olarak kullanmışlardır. Bu nedenlerden dolayı deney grubu öğrencilerinin öz-düzenleme becerilerinin son-testi lehine anlamlı bir şekilde yükselmiş olduğu söylenebilir.

Yukarıda yer alan programlama öz-düzenleme becerilerine yönelik tartışmalar doğrultusunda BTY dersi öğretim programlarında yer alan öğrenme ve öğretme içeriklerinin öğrencilerin öz-düzenleme becerilerini artırıcı etkinliklerle düzenlenmesi gerektiği söylenebilir. Öğretim programları boyutunda ise ilk olarak, ortaokul düzeyindeki BTY dersinin öğretim programlarında işbirlikli öğrenme tekniklerinin yer alması gerektiği söylenebilir. Ayrıca ikinci olarak, özel ve devlet okullarında görev yapan Biliřim Teknolojileri öğretmenlerine uygulanan mesleki hizmet içi eğitim programlarında ve Biliřim Teknolojileri öğretmeni yetiřtiren lisans programlarında işbirlikli öğrenme ile kurgulanan programlama öğretiminin öz-düzenleme becerisini geliřtirmek amacıyla programlarda alması gerektiği söylenebilir.

6. ÖNERİLER

Bu araştırmada, ilk olarak işbirlikli öğrenme yaklaşımının programlama öz-yeterliğini ve öz-düzenleme becerisini olumlu yönde arttırmada etkili olduğu saptandığından, programlama içeren öğretim programlarında işbirlikli öğrenme yaklaşımına yer verilmesi önerilmektedir. Diğer öneriler, uygulamaya yönelik öneriler ve yeni araştırmalara yönelik öneriler başlıkları altında ele alınmıştır.

6.1 Uygulamaya Yönelik Öneriler

- Grupla programlama öğretiminde bilgisayar üzerinde uygulamalı konuların öğretiminde ikili gruplar oluşturulması ve buna uygun işbirlikli öğretim teknikleri seçilmesi önerilmektedir.
- BTY dersi problem çözme ve programlama ünitesinde yer alan konuların bazılarının bilgisayarda uygulamaya bazı konuların ise çalışma yapraklarıyla işlenmesine olanak tanınması sebebiyle konunun yapısına uygun teknik seçimi yapılmalıdır.
- İşbirlikli öğrenme modelinin uygulanması sırasında teorik bilgilerin yapılandırmasında dört ile altı kişilik gruplar, konuların bilgisayarda pratik uygulamasında iki kişilik gruplar ile etkinliklerin yürütülmesi önerilmektedir.
- Bilgisayar üzerindeki uygulamalı programlama konularının öğretiminde ikili gruplar kurgulanırken, öğrenci etkileşiminin artırılması için her hafta yeni grupların oluşturulması önerilmektedir.
- Öğrenci gruplarının oluşturulmasında, öğrencilerin bir önceki haftanın performans değerlendirilmesi göz önüne alınarak başarı anlamında heterojen grupların oluşturulması önerilmektedir.
- İşbirlikli öğrenme teknikleri uygulama sürecinde grupların öğrencilerin cinsiyet, başarı, motivasyon, önbilgi açılarından heterojen oluşturulması önerilmektedir.

6.2 Yeni Arařtırmalara Yönelik Öneriler

- İşbirlikli öğrenme ile programlama öğretim süreçlerinde grup çalışmalarının yanında öğrencilerin bireysel olarak çalışmasına ve öğrenme süreçlerini düzenlemesine olanak verilmelidir. Örneğin bireysel çalışma yapacakları programlama web sitelerinden ev ödevleri verilebilir. Programlama sürecinde yaptığı işlemlere yönelik blog yazması sağlanarak öz-düzenleme becerilerinin gelişmesi sağlayabilir.
- Öğretmen öğrencilerin sosyal ve öz-düzenleme becerilerini geliştirmek için öğretim sürecinde işbirlikli ve öz-düzenlemeli bir karma model kullanılabilir. Örneğin, sınıf içi uygulamasında öğrenci programlamayı yönlendirmeli programlama platformlarında bireysel olarak öğrendikten sonra grup çalışmaları ile öğrenmelerini derinleştirmesine olanak verilmelidir.
- Programlama öğretiminde BTP ile öz-düzenlemeli öğrenme modeli kullanılırsa öz-düzenleme becerisi ve programlama bağlantısı daha net olarak ortaya çıkarılabilir. Bu doğrultuda bilgisayarda programlama öğretim süreçlerine öz-düzenlemeli öğrenme entegre edilmelidir.
- Bu çalışmadaki sınıf mevcutları 20-25 kişidir bu durum işbirlikli öğrenme modelindeki grup çalışmalarının uygulanmasını etkileyebilecek bir unsurdur. Öğretmenin grupları oluşturmasını ve öğretim sürecine rehberlik etmesini kolaylaştırabilmektedir. Gelecekteki çalışmalarda daha kalabalık sınıf mevcutlarında işbirlikli öğrenme ile programlama öğretimi üzerine arařtırmalar yapılabilir.
- İşbirlikli öğrenmenin, programlama ve öz-düzenleme arasındaki aracılık rolü arařtırılabilir.
- Öz-düzenleme becerilerinin programlama akademik performans üzerindeki etkisi arařtırılabilir.

- İşbirlikli öğrenmenin, programlama öğretimindeki etkisi gerçek deneysel desen kullanılarak yeniden araştırılabilir.



KAYNAKLAR

- Aktan, S. (2012). *Öğrencilerin akademik başarısı, öz düzenleme becerisi, motivasyonu ve öğretmenlerinin öğretim stilleri arasındaki ilişki*. [Doktora Tezi]. Balıkesir Üniversitesi.
- Albayrak, E., & Polat, E. (2022). Pair programming experiences of prospective information technologies teachers. *Bartın Üniversitesi Eğitim Fakültesi Dergisi*, 11(2), 351-363. <https://doi.org/10.14686/buefad.991448>
- Alhazbi, S., & Hassan, M. (2013). Fostering self-regulated learning in introductory computer programming course. In *18th Annual Western Canada Conference on Computing Education* (pp. 1-4).
- Anılan, H. ve Gezer, B. (2020). Kodlama etkinliklerine ve analitik düşünme becerisine yönelik sınıf öğretmenlerinin görüşlerinin incelenmesi. *Anadolu Üniversitesi Eğitim Fakültesi Dergisi (AUJEF)*, 4(4), 307-324. <https://doi.org/10.34056/aujef.801254>
- Arseven, A. (2016). Öz yeterlilik: Bir kavram analizi. *Electronic Turkish Studies*, 11(19), 63-80. <http://dx.doi.org/10.7827/TurkishStudies.10001>
- Arslan, A. (2011). Ayrılıp birleşme IV tekniğinin erişiyeye, öz yeterlik inancına ve öz düzenleme becerisine etkisi. *Uluslararası Yönetim İktisat ve İşletme Dergisi*, 7(13), 369-385.
- Arslan, A., ve Engin, A. O. (2021). İşbirlikli Öğretim Yönteminin Türkçe Dersinde Akademik Başarı, Motivasyon ve Öğrenme Kalıcılığı Üzerindeki Etkisinin Belirlenmesi. *Eğitim ve Toplum Araştırmaları Dergisi*, 8(1), 205-227. <https://doi.org/10.51725/etad.940163>
- Arslan, E., ve İşbulan, O. (2021). The effect of individual and group learning on block-based programming self-efficacy and robotic programming attitudes of secondary school students. *Malaysian Online Journal of Educational Technology*, 9(1), 108-121. <https://doi.org/10.17220/mojet.2021.9.1.249>
- Arslan, K. ve Akçelik, M. (2019). Programlama eğitiminde Scratch'in kullanılması: Öğretmen adaylarının tutum ve algıları. *Ulusal Eğitim Akademisi Dergisi (UEAD)*, 3(1), 41-61. <https://doi.org/10.32960/uead.455502>
- Arslan, S., ve Gelişli, Y. (2015). Algılanan öz-düzenleme ölçeği: Bir ölçek geliştirme çalışması. *Sakarya University Journal of Education*, 5(3), 67-74. <https://doi.org/10.19126/suje.07146>

- Aydın, S., ve Atalay, T. D. (2015). *Öz-düzenlemeli öğrenme*. Pegem Akademi Yayıncılık.
- Aytekin, A., Çakır, F. S., Yücel, Y. B., ve Kulaöz, İ. (2018). Geleceğe yön veren kodlama bilimi ve kodlama öğrenmede kullanılabilecek bazı yöntemler. *Avrasya Sosyal ve Ekonomi Araştırmaları Dergisi*, 5(5), 24-41.
- Bahçetepe, Ü., ve Giorgetti, F. M. (2015). Akademik başarı ile okul iklimi arasındaki ilişki. *Istanbul Journal of Innovation in Education*, 1(3), 83-101.
- Balanskat, A., ve Engelhardt, K. (2014). *Computing our future: Computer programming and coding-priorities, school curricula and initiatives across Europe*. European Schoolnet.
- Bandura, A. (1986). The explanatory and predictive scope of self-efficacy theory. *Journal of social and clinical psychology*, 4(3), 359-373.
- Bandura, A. 1997. Self-efficacy: The exercise of control. New York: W.H. Freeman and Company.
- Bayburt, B., ve Eğin, F. (2021). Teknoloji ve sanayideki gelişmelerin yansıması olarak eğitim 4.0. *Bilgi Ekonomisi ve Yönetimi Dergisi*, 16(2), 137-154. <https://doi.org/10.54860/beyder.1010372>
- Bayrakçeken, S., Doymuş, K., ve Doğan, A. (2013). *İşbirlikli öğrenme modeli ve uygulanması*. Pegem Akademi Yayıncılık
- Bebek, M. (2022). *Bilişim teknolojileri öğretmenlerinin programlamaya ilişkin öz yeterlilikleri ile eleştirel düşünme eğilimleri arasındaki ilişkinin incelenmesi*. [Yüksek Lisans Tezi]. Mersin Üniversitesi.
- Bennedsen, J. ve Carpersen, M. E. (2008). Exposing the programming process. Bennedsen, J., Carpersen, M. E. ve Kolling, M. (Eds.). *Reflections on the Teaching of Programming: Methods and Implementations* (pp. 6-16). New York: Springer Berlin Heidelberg Journal. https://doi.org/10.1007/978-3-540-77934-6_2
- Best, J. W., & Kahn, J. V. (2017). *Eğitimde araştırma yöntemleri*. (O. Köksal, Çev.). Eğitim Yayınevi.
- Bilgin, İ., ve Gelici, Ö. (2011). İşbirlikli öğrenme tekniklerinin tanıtımı ve öğrenci görüşlerinin incelenmesi. *Adıyaman University Journal of Educational Sciences*, 1(1), 40-70.
- Bjursten, E. L., Nilsson, T., & Gumaelius, L. (2023). Computer programming in primary schools: Swedish Technology Teachers' pedagogical strategies.

International journal of technology and design education, 33(4), 1345-1368.
<https://doi.org/10.1007/s10798-022-09786-7>

Bocconi, S., Chiocciariello, A., Dettori, G., Ferrari, A., Engelhardt, K., Kampylis, P., & Punie, Y. (2016). *Developing computational thinking in compulsory education: Joint Research Center (European Commission)*. [Çevrim-içi: <https://edudoc.ch/>], Erişim Tarihi: 20.10.2024.

Buitrago Flórez, F., Casallas, R., Hernández, M., Reyes, A., Restrepo, S., & Danies, G. (2017). Changing a generation's way of thinking: Teaching computational thinking through programming. *Review of Educational Research*, 87(4), 834-860. <https://doi.org/10.3102/0034654317710096>

Büyüköztürk, Ş. (2012). *Sosyal bilimler için veri analizi el kitabı*. Pegem Akademi Yayıncılık

Büyüköztürk, Ş., Kılıç-Çakmak, E., Akgün, Ö., Karadeniz, Ş., ve Demirel, F. (2008). *Bilimsel araştırma yöntemleri*. Pegem Akademi Yayıncılık

Cabı, E. (2015). Öğretmen adaylarının öz-düzenleme stratejileri ve akademik başarısı: boylamsal bir araştırma. *Gazi Üniversitesi Gazi Eğitim Fakültesi Dergisi*, 35(3), 489-506.

Canbazoglu Bilici, S. (2023) Deneysel araştırma yöntemi. Özmen, H. & Karamustafaoğlu, O. (Eds.). *Eğitimde araştırma yöntemleri*. (pp. 56-78). Pegem Akademi Yayıncılık

Ceylan, V. K., ve Gündoğdu, K. (2018). Bir olgubilim çalışması: kodlama eğitiminde neler yaşanıyor?. *Eğitim Teknolojisi Kuram ve Uygulama*, 8(2), 1-34. <https://doi.org/10.17943/etku.340103>

Chao, P. Y. (2016). Exploring students' computational practice, design and performance of problem-solving through a visual programming environment. *Computers & Education*, 95, 202-215. <https://doi.org/10.1016/j.compedu.2016.01.010>

Chen, C. Y., Su, S. W., Lin, Y. Z., & Sun, C. T. (2023). The effect of time management and help-seeking in self-regulation-based computational thinking learning in taiwanese primary school students. *Sustainability*, 15(16), 12494. <https://doi.org/10.3390/su151612494>

Ching, Y. H., Hsu, Y. C., & Baldwin, S. (2018). Developing computational thinking with educational technologies for young learners. *TechTrends*, 62, 563-573. <https://doi.org/10.1007/s11528-018-0292-7>

- Çal, H. ve Can, G. (2020). The influence of pair programming on secondary school students' confidence and achievement in computer programming. *Trakya Eğitim Dergisi*, 10 (1), 221-237. <https://doi.org/10.24315/tred.575098>
- Çalışkan, E. (2020). Code.org etkinliklerinin ortaokul öğrencilerinin problem çözme becerilerine ve programlama öz-yeterliklerine etkisinin incelenmesi. *Journal of Instructional Technologies and Teacher Education*, 9(2), 114-124.
- Çelik, N., ve Gündoğdu, K. (2020). Matematik öğretmen adaylarının öz düzenleme ve öz yeterlik algılarının incelenmesi. *Uluslararası Bilim ve Eğitim Dergisi*, 3(2), 101-118.
- Çiftçi, S., Çengel, M. ve Paf, M. (2018). Bilişim öğretmeni adaylarının programlama ilişkin öz-yeterliklerinin yordayıcısı olarak bilişimsel düşünme ve problem çözmeye ilişkin yansıtıcı düşünme becerileri. *Ahi Evran Üniversitesi Kırşehir Eğitim Fakültesi Dergisi*, 19 (1), 321-334.
- Çiltaş, A. (2011). Eğitimde öz-düzenleme öğretiminin önemi üzerine bir çalışma. *Mehmet Akif Ersoy Üniversitesi Sosyal Bilimler Enstitüsü Dergisi*, 3(5), 1-11. <https://doi.org/10.20875/sb.72394>
- Çobanoğlu, B. (2009). *Algoritma geliştirme ve veri yapıları*. Pusula Yayıncılık.
- Çoşğun, Ü. Ç., ve Çoşğun, V. (2018). Programlama öğretiminin ortaokul öğrencilerinin öz-düzenleme stratejileri ve motivasyonel inançları üzerindeki etkisi. *Gaziantep Üniversitesi Eğitim Bilimleri Dergisi*, 2(2), 59-71.
- Dadlı, G. (2015). *Ortaokul 8. sınıf öğrencilerinin fen ve teknoloji dersine yönelik öz düzenleme becerileri ve öz yeterlikleri ile akademik başarıları arasındaki ilişkinin incelenmesi*. [Yüksek Lisans Tezi]. Kahramanmaraş Sütçü İmam Üniversitesi.
- Demir, Ö., ve Seferoglu, S. S. (2021). A comparison of solo and pair programming in terms of flow experience, coding quality, and coding achievement. *Journal of Educational Computing Research*, 58(8), 1448-1466. <https://doi.org/10.1177/0735633120949788>
- Demircan, Y. (2014). *5. sınıf öğrencilerinin sınıf içi etkinlik ve akademik başarı düzeylerine göre öz-düzenleme stratejileri ve motivasyonel inançlarının incelenmesi*. [Yüksek Lisans Tezi], Mersin Üniversitesi.
- Demirer, V., ve Sak, N. (2016). Programming education and new approaches around the world and in Turkey/Dünyada ve Türkiye'de programlama eğitimi ve yeni yaklaşımlar. *Eğitimde Kuram ve Uygulama*, 12(3), 521-546.

- Demirören, S. (2016). Eşli programlama etkinliğinin ortaokul öğrencilerinin programlama başarılarına etkisi. *Educational Sciences and Practice*, 15 (29), 1-20.
- Deniz, T. (2021). *Farklı görsel programlarla tasarlanan kodlama eğitiminin ortaokul öğrencilerinin akademik başarı ile kodlamaya karşı tutum ve öz-yeterliklerine etkisi*. [Yüksek Lisans Tezi], Necmettin Erbakan Üniversitesi
- Denner, J., Green, E., & Campe, S. (2021). Learning to program in middle school: How pair programming helps and hinders intrepid exploration. *Journal of the Learning Sciences*, 30(4-5), 611-645. <https://doi.org/10.1080/10508406.2021.1939028>
- Denner, J., Werner, L., Campe, S., & Ortiz, E. (2014). Pair programming: Under what conditions is it advantageous for middle school students?. *Journal of Research on Technology in Education*, 46(3), 277-296. <https://doi.org/10.1080/15391523.2014.888272>
- Dikmen, M., ve Tuncer, M. (2020). Öğrenme stiline dayalı eğitimin akademik başarıya etkisi. *Ege Eğitim Dergisi*, 21(1), 71-88. <https://doi.org/10.12984/egeefd.695452>
- Dilekli, Y., Orakcı, Ş., ve Ayçiçek, B. (2023). Öğretmen adaylarının öz yeterlik algıları ve öz düzenleyici öğrenme becerileri arasındaki ilişkinin incelenmesi. *Türk Eğitim Bilimleri Dergisi*, 21(3), 1360-1383. <https://doi.org/10.37217/tebd.1313393>
- Doleck, T., Bazalais, P., Lemay, D. J., Saxena, A., & Basnet, R. B. (2017). Algorithmic thinking, cooperativity, creativity, critical thinking, and problem solving: exploring the relationship between computational thinking skills and academic performance. *Journal of computers in education*, 4, 355-369. <https://doi.org/10.1007/s40692-017-0090-9>
- Doymuş, K., ve Koç, Y. (2012). Fen ve Teknoloji öğretmenlerinin işbirlikli öğrenme modelinin sınıftaki uygulaması. *Dicle Üniversitesi Ziya Gökalp Eğitim Fakültesi Dergisi*, (19), 174-183.
- Dönmez, B., ve Gündoğdu, K. (2018). 7. sınıf türkçe dersinde ayrılıp birleşme tekniği (jigsaw) kullanımının akademik başarı, öz-düzenleme becerisi, tutum, erişimi ve bilginin kalıcılığına etkisi. *İlköğretim Online* 959-983. <https://doi.org/10.17051/ilkonline.2018.419348>
- Efe, R., Hevedanlı, M., Ketani, Ş., Çakmak, Ö., ve Efe, H. A. (2008). *İşbirlikli öğrenme teori ve uygulama*. Eflatun Yayınevi

- Eker, C. (2015). Öz-düzenleme becerilerinin öğretimi sürecinde kwl (bil-iste-öğren) stratejisinin etkisi. *Akademik Bakış Uluslararası Hakemli Sosyal Bilimler Dergisi*, (51), 168-182.
- Ekici, M., ve Çınar, M. (2020). Bilgisayar programlama öz-yeterlik ölçeğinin türkçe formunun geçerlik ve güvenirlik çalışması. *Anadolu Journal of Educational Sciences International*, 10(2), 1017-1040. <https://doi.org/10.18039/ajesi.725161>
- Eraytaç, Ö. F. (2019). *Robotik kodlama eğitiminde blok tabanlı kodlama yönteminin ortaokul öğrencilerinin akademik başarısına etkisi*. [Yüksek Lisans Tezi]. Çukurova Üniversitesi.
- Erdoğan, M. Y. (2006). Yaratıcılık ile öğretmen davranışları ve akademik başarı arasındaki ilişkiler. *Elektronik sosyal bilimler dergisi*, 5(17), 95-106.
- Ergin, A. Z., ve Ercan, Z. G. (2022). Okul öncesi öğretmen adaylarının kodlamaya ilişkin görüşleri. *Eğitim Yönetimi ve Politikaları Dergisi*, 3(1), 70-82.
- Erkoç, F. (2022). Programlama öğretimi stratejileri. Kert S.B. (Eds.). *Programlama öğretimi yaklaşımları* (pp. 42-59). Nobel Akademik Yayıncılık
- Erol, O., ve Kurt, A. A. (2017). The effects of teaching programming with scratch on pre-service information technology teachers' motivation and achievement. *Computers in Human Behavior*, 77, 11-18. <https://doi.org/10.1016/j.chb.2017.08.017>
- Erümit, A. K. (2020). Effects of different teaching approaches on programming skills. *Education and Information Technologies*, 25(2), 1013-1037. <https://doi.org/10.1007/s10639-019-10010-8>
- Eryılmaz, S., ve Deniz, G. (2019). Türkiye’de programlama eğitimi ile ilgili yapılan çalışmaların incelenmesi: Bir betimsel analiz çalışması. *Eğitimde Kuram ve Uygulama*, 15(4), 319-338. <https://doi.org/10.17244/eku.645387>
- Eryılmaz, S., ve Uluyol, Ç. (2015). 21. yy becerileri ışığında FATİH projesi değerlendirmesi. *Gazi Eğitim Fakültesi Dergisi*, 35(2), 209-229.
- Eskici, G. Y., Mercan, S., ve Hakverdi, F. (2020). Robotik kodlama eğitiminden yansımalar: zihinsel imajlar. *Amasya Üniversitesi Eğitim Fakültesi Dergisi*, 9(1), 30-64.
- Espinal, A., Vieira, C., & Guerrero-Bequis, V. (2022). Student ability and difficulties with transfer from a block-based programming language into other programming languages: a case study in Colombia. *Computer Science Education*, 1-33. <https://doi.org/10.1080/08993408.2022.2079867>

- Feng, C. Y., & Chen, M. P. (2014). The effects of goal specificity and scaffolding on programming performance and self-regulation in game design. *British Journal of Educational Technology*, 45(2), 285-302. <https://doi.org/10.1111/bjet.12022>
- Feurzeig, W., Papert, S. A., & Lawler, B. (2011). Programming languages as a conceptual framework for teaching mathematics. *Interactive Learning Environments*, 19(5), 487-501. <https://doi.org/10.1080/10494820903520040>
- Garcia, M. B. (2021). Cooperative learning in computer programming: A quasi-experimental evaluation of Jigsaw teaching strategy with novice programmers. *Education and Information Technologies*, 26(4), 4839-4856. <https://doi.org/10.1007/s10639-021-10502-6>
- Gomes, A., & Mendes, A. J. (2007, September). Learning to program-difficulties and solutions. *Proceedings of the International Conference on Engineering Education (ICEE), Portugal*, 283-287.
- Gökoğlu, S. (2017). Programlama eğitiminde algoritma algısı: Bir metafor analizi. *Cumhuriyet Uluslararası Eğitim Dergisi*, 6(1), 1-14.
- Göksün, D. O. (2016). Öğretmen adaylarının 21. yy. öğrenen becerileri ve 21. yy. öğreten becerileri arasındaki ilişki. [Doktora Tezi]. Anadolu Üniversitesi.
- Göncü, A., Çetin, İ., ve Top, E. (2018). Öğretmen adaylarının kodlama eğitimine yönelik görüşleri: bir durum çalışması. *Mehmet Akif Ersoy Üniversitesi Eğitim Fakültesi Dergisi*, (48), 85-110.
- Güleryüz, H. (2022). The effect of robotic coding (mblock-arduino) activities on students' self-efficacy and attitudes. *Acta Scientific Computer Sciences*4(8), 2-9.
- Gülmez, I. (2009). Programlama öğretiminde görselleştirme araçlarının kullanımının öğrenci başarı ve motivasyonuna etkisi. [Doktora Tezi]. Marmara Üniversitesi.
- Gülten, D., ve Soytürk, İ. (2013). İlköğretim 6. sınıf öğrencilerinin geometri öz-yeterliklerinin akademik başarı not ortalamaları ile ilişkisi. *Mehmet Akif Ersoy Üniversitesi Eğitim Fakültesi Dergisi*, 1(25), 55-70.
- Gürbüztürk, O., ve Tanataş, D. Y. (2024). Blok tabanlı kodlama araçlarının akademik başarı, tutum ve bilgi işlemsel düşünme becerisine etkisi: Meta-analiz çalışması. *İnönü Üniversitesi Eğitim Bilimleri Enstitüsü Dergisi*, 11(21), 58-79. <https://doi.org/10.29129/inujse.1425193>
- Gürer, M. D., Gül, D., ve Konyaoğlu, C. (2019). Bilişim teknolojileri ve yazılım dersi öğretmen adaylarının mesleki öz-yeterlik algıları ile öz-düzenleme becerileri

arasındaki ilişki. *Abant İzzet Baysal Üniversitesi Eğitim Fakültesi Dergisi*, 19(2), 517-529.

Güvenç, H. (2010). İşbirlikli öğrenme ve ders günlüklerinin öğretmen adayı öğrencilerin öz düzenlemeli öğrenmeleri üzerindeki etkileri. *Kuram ve Uygulamada Eğitim Bilimleri*, 10(3), 1459-1487.

Hamer, J. M., Kemp, P. E., Wong, B., & Copsey-Blake, M. (2024). Cracking the code: exploring student attitudes towards coding in secondary education. *Cambridge Journal of Education*, 54(4), 495-516. <https://doi.org/10.1080/0305764X.2024.2387335>

Han, K. W., Lee, E., & Lee, Y. (2009). The impact of a peer-learning agent based on pair programming in a programming course. *IEEE Transactions on Education*, 53(2), 318-327. <https://doi.org/10.1109/TE.2009.2019121>

Hangün, M. E. (2019). *Robot programlama eğitiminin öğrencilerin matematik başarısına, matematik kaygısına, programlama özyeterliliğine ve STEM tutumuna etkisi*. [Yüksek Lisans Tezi]. Fırat Üniversitesi.

Hanks, B., McDowell, C., Draper, D., & Krnjajic, M. (2004, June). Program quality with pair programming in CS1. In *Proceedings of the 9th annual SIGCSE conference on Innovation and technology in computer science education* (pp. 176-180). <https://doi.org/10.1145/1007996.1008043>

Hartl, A. C., DeLay, D., Laursen, B., Denner, J., Werner, L., Campe, S., & Ortiz, E. (2015). Dyadic instruction for middle school students: Liking promotes learning. *Learning and Individual Differences*, 44, 33-39. <https://doi.org/10.1016/j.lindif.2015.11.002>

Haşlaman, T. ve Aşkar, P. (2007). Programlama dersi ile ilgili öz-düzenleyici öğrenme stratejileri ve başarı arasındaki ilişkinin incelenmesi. *Hacettepe Üniversitesi Eğitim Fakültesi Dergisi*, 32, 110-122.

Haymana, İ., ve Özalp, D. (2020). Robotik ve kodlama eğitiminin ilkökul 4. sınıf öğrencilerinin yaratıcı düşünme becerilerine etkisi. *İstanbul Aydın Üniversitesi Eğitim Fakültesi Dergisi*, 6(2), 247-274.,

Hill, C. (2015). *Programming environments for children: Creating a language that grows with you*. [Yüksek Lisans Tezi], University of California. ProQuest Dissertations & Theses Global. <https://www.proquest.com/dissertations-theses/programming-environments-children-creating/docview/1731931047/se-2>

Holt, L. (1993). *Cooperative learning in action*. National Middle School Association

- Hsu, Y. C., Irie, N. R., & Ching, Y. H. (2019). Computational thinking educational policy initiatives (CTEPI) across the globe. *TechTrends*, 63, 260-270. <https://doi.org/10.1007/s11528-019-00384-4>
- Huang, C. L., Fu, L., Hung, S. C., & Yang, S. C. (2025). Effect of visual programming instruction on students' flow experience, programming self-efficacy, and sustained willingness to learn. *Journal of Computer Assisted Learning*, 41(1), e13110. <https://doi.org/10.1111/jcal.13110>
- Jančec, L., & Vujičić, L. (2021). Project “algorithmic thinking skills through play-based learning for future's code literates”. In *2021 44th International Convention on Information, Communication and Electronic Technology (MIPRO)* (pp. 641-644). IEEE. <https://doi.org/10.23919/MIPRO52101.2021.9597078>
- Johnson, D.W., & Johnson, R.T. (1999). Making cooperative learning work. *Theory into Practice*, 38 (2), 67-73. <https://doi.org/10.1080/00405849909543834>
- Kalelioğlu, F. (2015). A new way of teaching programming skills to k-12 students: code. *Org. Computers in Human Behavior*, 52, 200-210. <https://doi.org/10.1016/j.chb.2015.05.047>
- Kampala's, P., Darien, V., Bocconi, S., Ciccarelli, A., Engelhardt, K., Stepfriend, G., Stepfriend, V., Jesuit, E., Madagali, C., Horvath, M., & Earp, J. (2023). Integrating computational thinking into primary and lower secondary education: A systematic review. *Educational Technology & Society*, 26(2), 99-117. <https://doi.org/10.2791/792158>
- Karabak, D., ve Güneş, A. (2013). Ortaokul birinci sınıf öğrencileri için yazılım geliştirme alanında müfredat önerisi. *Eğitim ve Öğretim Araştırmaları Dergisi*, 21(2-3), 163-169.
- Karaca, E. (2022). Test ve madde analizi. Erkan, S., Gömleksiz, M. (Eds.). *Eğitimde ölçme ve değerlendirme* içinde (s. 240-304). Nobel Akademik Yayıncılık
- Karataş, H. (2021). 21. Yy. becerilerinden robotik ve kodlama eğitiminin Türkiye ve Dünyadaki yeri. *21. Yüzyılda Eğitim ve Toplum*, 10(30), 693-729.
- Kasalak, İ., ve Altun, A. (2018). Blok temelli programlamaya ilişkin öz-yeterlik algısı ölçeği geliştirme çalışması: Scratch örneği. *Eğitim Teknolojisi Kuram ve Uygulama*, 8(1), 209-225. <https://doi.org/10.17943/etku.335916>
- Kaya, A. B., ve Alpan, G. (2020). Oyunlaştırılmış kodlama eğitimi: öğrenme yönetim sistemi model önerisi. *Eğitimde Teknoloji Uygulamaları Dergisi*, 1(1), 1-25. <https://doi.org/10.29329/jtae.2020.283.1>

- Kayhan, E., Altun, S., ve Gürol, M. (2019). Sekizinci sınıf Türkçe öğretim programı (2018)'nın 21. yüzyıl becerileri açısından değerlendirilmesi. *Adnan Menderes Üniversitesi Eğitim Fakültesi Eğitim Bilimleri Dergisi*, 10(2), 20-35.
- Kert, S. B. ve Uğraş, T. (2009, Mayıs). Programlama eğitiminde sadelik ve eğlence: Scratch örneği. *The First International Congress of Educational Research*, Çanakkale, Türkiye. [Çevrim-ici: academia.edu], Erişim Tarihi: 18.07.2024.
- Kert, S.B. (2018). Programlama öğretimi için pedagojik yaklaşımlar. Gülbahar, Y., Karal, H. (Eds.). *Kuramdan uygulamaya programlama öğretimi*. içinde (s. 93-130). Pegem Akademi Yayıncılık
- Komarraju, M., & Nadler, D. (2013). Self-efficacy and academic achievement: Why do implicit beliefs, goals, and effort regulation matter?. *Learning and individual differences*, 25, 67-72. <https://doi.org/10.1016/j.lindif.2013.01.005>
- Korkmaz, Ö., Balcı, H., Çakır, R., ve Uğur Erdoğan, F. (2021). Görsel programlama ortamlarında yapılan oyun geliştirme etkinliklerinin etkililiği. *Mehmet Akif Ersoy Üniversitesi Eğitim Fakültesi Dergisi*, (57), 52-73.
- Korkmaz, Ö., Çakır, R. ve Özden, M. Y. (2015). Bilgisayarca düşünme beceri düzeyleri ölçeğinin (bdbl) ortaokul düzeyine uyarlanması. *Gazi Eğitim Bilimleri Dergisi*, 1, 143-162.
- Korkmaz, Ö., Çakır, R., Özden, M. Y., Oluk, A., ve Sarıoğlu, S. (2016). Bireylerin bilgisayarca düşünme becerilerinin farklı değişkenler açısından incelenmesi. *Ondokuz Mayıs University Journal of Education Faculty*, 34(2), 68-87. <https://doi.org/10.7822/omuefd.215276>
- Köseoğlu, P. (2010). Biyoloji eğitiminde birleştirme tekniği temelli öğretimin akademik başarı, özyeterlik ve tutuma etkisi. *Hacettepe Üniversitesi Eğitim Fakültesi Dergisi*, 39(39), 244-254.
- Krizsan, T., & Lambic, D. (2024). Examining the impact of pair programming on efficiency, motivation, and stress among students of different skills and abilities in lower grades in elementary schools. *Education and Information Technologies*, 1-24. <https://doi.org/10.1007/s10639-024-12859-w>
- Kukul, V., ve Gökçearslan, Ş. (2014). *Scratch ile programlama eğitimi alan öğrencilerin problem çözme becerilerinin incelenmesi*. 8th International Computer & Instructional Technologies Symposium, Edirne, Turkey
- Kukul, V., Gökçearslan, Ş., ve Günbatır, M. S. (2017). Computer programming self-efficacy scale (CPSES) for secondary school students: Development, validation and reliability. *Eğitim Teknolojisi Kuram ve Uygulama*, 7(1), 158-179. <https://doi.org/10.17943/etku.288493>

- Laal, M., & Ghodsi, S. M. (2012). Benefits of collaborative learning. *Procedia-social and behavioral sciences*, 31, 486-490. <https://doi.org/10.1016/j.sbspro.2011.12.091>
- Li, M. P., & Lam, B. H. (2013). Cooperative learning. *The Hong Kong Institute of Education*, 1, 33.
- Li, W., Huang, J. Y., Liu, C. Y., Tseng, J. C., & Wang, S. P. (2023). A study on the relationship between student learning engagements and higher-order thinking skills in programming learning. *Thinking Skills and Creativity*, 49, 101369. <https://doi.org/10.1016/j.tsc.2023.101369>
- Lin, Y., & Weintrop, D. (2021). The landscape of block-based programming: Characteristics of block-based environments and how they support the transition to text-based programming. *Journal of Computer Languages*, 67, 101075. <https://doi.org/10.1016/j.col.2021.101075>
- Loksa, D., & Ko, A. J. (2016). The role of self-regulation in programming problem solving process and success. In *Proceedings of the 2016 ACM conference on international computing education research* (pp. 83-91). <https://doi.org/10.1145/2960310.296033>
- Mahmod, N. H. N. M., Hamzah, R. N. N. R. A., Hamid, N. F. A., & Jaafar, A. (2023). Perception of Malaysian Dental Practitioner on the provisional COVID-19 Guidelines in the Dental Practice: A Cross-Sectional Study. *Malaysian Journal of Health Sciences/Jurnal Sains Kesihatan Malaysia*, 21(1), 85-94.
- Marr, M. B. (1997). Cooperative learning: A brief review. *Reading & Writing Quarterly: Overcoming Learning Difficulties*, 13(1), 7-20. <https://doi.org/10.1080/1057356970130102>
- Mazman, S. G. ve Altun, A. (2013). Programlama – I dersinin BÖTE bölümü öğrencilerinin programlamaya ilişkin öz yeterlilik algıları üzerine etkisi. *Journal of Instructional Technologies and Teacher Education*, 2 (3), 24-29.
- MEB (2018a). 2023 Eğitim vizyon belgesi. [Çevrim-ı: <https://ikizce.meb.gov.tr/www/meb-2023-egitim-vizyonu-belgesi/icerik/542>], Erişim Tarihi: 09.10.2023.
- MEB. (2018b). *Bilişim teknolojileri ve yazılım dersi öğretim programı (ortaokul 5 ve 6. sınıflar)*. [Çevrim-ı: <http://mufredat.meb.gov.tr>], Erişim Tarihi: 07.10.2023.
- Menon, P. (2022). The role of task value and online learning strategies in an introductory computer programming course. *Information Systems Education Journal*, 20(4), 19-31.

- Mıhçı Türker, P. ve Pala, F. K. (2018). Ortaokul öğrencilerinin, öğretmenlerin ve öğrenci velilerinin kodlamaya yönelik görüşleri. *İlköğretim Online*, 17(4), 2013– 2029. <https://doi.org/10.17051/ilkonline.2019.506939>
- Mohorovicic, S. & Strcic, V. (2011). An overview of computer programming teaching methods. Varazdin: Faculty of Organization and Informatics Varazdin. [Çevrim-ıçı: <https://www.proquest.com/conference-papers-proceedings/overview-computer-programming-teaching-methods/docview/1322999502/se-2>], Erişim Tarihi: 11.10.2023.
- Noone, M., & Mooney, A. (2018). Visual and textual programming languages: A systematic review of the literature. *Journal of Computers in Education*, 5(2), 149–174. <https://doi.org/10.1007/s40692-018-0101-5>
- Okal, G., Yıldırım, B., ve Timur, S. (2020). The effect of coding education on 5th, 6th and 7th grade students' programming self-efficacy and attitudes about technology. *Educational Policy Analysis and Strategic Research*, 15(2), 143-165. <https://doi.org/10.29329/epasr.2020.251.8>
- Özdişci, S., ve Katrancı, Y. (2020). Ortaokul öğrencilerinin problem çözme ve problem oluşturma becerilerinin incelenmesi. *Milli Eğitim Dergisi*, 49(226), 149-184.
- Özel, O. (2019). *Programlama yöntemlerinin ortaokul öğrencilerinin bilgi işlemsel düşünme becerisine yönelik öz yeterlik algısına ve programlama başarısına etkisi*. [Yüksek Lisans Tezi]. Marmara Üniversitesi.
- Özenoğlu, Y. E., & Birişçi, S. (2019). *Grupla programlama öğretiminin yaratıcı düşünme becerileri üzerindeki etkisi* [Konferans sunumu]. Kongre. [Çevrim-ıçı: https://www.researchgate.net/profile/Yunus-Oezenoglu/publication/365184557_Grupla_Programlama_Ogretiminin_Yaratıcı_Dusunme_Becerileri_Uzerindeki_Etkisi/links/6369155437878b3e878dc030/Grupla-Programlama-Ogretiminin-Yaratıcı-Duesuenme-Becerileri-Uezerindeki-Etkisi.pdf], Erişim Tarihi: 15.11.2013.
- Özmen, H. (2023) Deneysel araştırma yöntemi. H. Özmen, O. Karamustafaoğlu, (Eds.). *Eğitimde araştırma yöntemleri*. (pp. 198-226). Pegem Akademi Yayıncılık
- Özmenteş, S. (2008). Çalgı eğitiminde öz-düzenlemeli öğrenme taktikleri. *İnönü Üniversitesi Eğitim Fakültesi Dergisi*, 9(16), 157–175.
- Öztürk, M. (2022). The effect of self-regulated programming learning on undergraduate students' academic performance and motivation. *Interactive Technology and Smart Education*, 19(3), 319-337. <https://doi.org/10.1108/ITSE-04-2021-0074>

- Pajares, F., & Schunk, D. (2001). The development of academic self-efficacy. *Development of achievement motivation. United States*, 7, 1-27.
- Pamuk, S., ve Elmas, R. (2015). Üst-bilişsel öz-düzenlemenin, öz-yeterlik ve hedef yönelimi ile açıklanması: Afyon ili örneği. *Amasya Üniversitesi Eğitim Fakültesi Dergisi*, 4(2), 175-189.
- Phillips, R. S., & Brooks, B. P. (2017). The hour of code: Impact on attitudes towards and self-efficacy with computer science. *Code: Seattle, WA, USA*.
- Pintrich, P. R. (1995). Understanding self-regulated learning. *New directions for teaching and learning*, 1995(63), 3-12. <https://doi.org/10.1002/tl.37219956304>
- Polat, K. (2023). Ortaokul öğrencilerinin bilişim teknolojileri öz-yeterlik algılarının belirlenmesi. *Bayterek Uluslararası Akademik Araştırmalar Dergisi*, 6(1), 142-156. <https://doi.org/10.48174/buaad.1286998>
- Popat, S., & Starkey, L. (2019). Learning to code or coding to learn? A systematic review. *Computers & Education*, 128, 365-376. <https://doi.org/10.1016/j.compedu.2018.10.005>
- Price, T. W., & Barnes, T. (2015, August). Comparing textual and block interfaces in a novice programming environment. In Proceedings of the eleventh annual international conference on international computing education research (pp. 91-99). <https://doi.org/10.1145/2787622.2787712>
- Rodríguez, F. J., Price, K. M., & Boyer, K. E. (2017, March). Exploring the pair programming process: Characteristics of effective collaboration. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education* (pp. 507-512). <https://doi.org/10.1145/3017680.3017748> [çevrim-içi: <https://dl.acm.org/doi/abs/10.1145/3017680.3017748>], Erişim tarihi: 10.11.2023
- Saeli, M., Perrenet, J., Jochems, W. M., & Zwaneveld, B. (2011). Teaching programming in Secondary school: A pedagogical content knowledge perspective. *Informatics in education*, 10(1), 73-88.
- Sakız, G. (2013). Başarıda anahtar kelime: Öz-yeterlik. *Uludağ Üniversitesi Eğitim Fakültesi Dergisi*, 26 (1), 185-210.
- Sarı, A., ve Akınoğlu, O. (2009). Öz-düzenlemeli öğrenme: Modeller ve uygulamalar. *Marmara Üniversitesi Atatürk Eğitim Fakültesi Eğitim Bilimleri Dergisi*, 29(29), 139-154.

- Saygıner, Ş., ve Tüzün, H. (2017). Erken yaşta programlama eğitimi: yurt dışı ve yurt içi perspektiflerinden bir bakış. *Uluslararası Bilgisayar ve Öğretim Teknolojileri Sempozyumu (ICITS)*, 69-77. Malatya, Türkiye.
- Sayın, Z. (2020). Öğretmenlerin kodlama eğitiminde eğilimlerinin belirlenmesi. *Journal of Instructional Technologies and Teacher Education*, 9(1), 52-64.
- Sayın, Z., ve Seferoğlu, S. S. (2016). Yeni bir 21. yüzyıl becerisi olarak kodlama eğitimi ve kodlamanın eğitim politikalarına etkisi. *Akademik Bilişim Konferansı*, 3(5), 1-13.
- Scherer, R., Siddiq, F., & Viveros, B. S. (2020). A meta-analysis of teaching and learning computer programming: Effective instructional approaches and conditions. *Computers in Human Behavior*, 109, 106349. <https://doi.org/10.1016/j.chb.2020.106349>
- Scratch. (2023). Scratch hakkında. [Çevrim-içi: <https://scratch.mit.edu/about>], Erişim tarihi:18.12.2023.
- Selçuk, M. (2022). Araç odaklı programlama öğretim yaklaşımları. Kert S.B. (Eds.). *Programlama öğretimi yaklaşımları içinde* (s. 61-86). Nobel Akademik Yayıncılık
- Sırakaya, M. (2018). Kodlama eğitimine yönelik öğrenci görüşleri. *Ondokuz Mayıs University Journal of Education Faculty*, 37(2), 79-90. <https://doi.org/10.7822/omuefd.394649>
- Slavin, R. E. (1987). Cooperative learning and the cooperative school. *Educational Leadership*, 45(3), 7-13.
- Slavin, R. E., & Cooper, R. (1999). Improving intergroup relations: Lessons learned from cooperative learning programs. *Journal of Social issues*, 55(4), 647-663. <https://doi.org/10.1111/0022-4537.00140>
- Sönmez, S. (2019). *Maker öğretmenlerin yürütmüş oldukları eğitim faaliyetlerine ilişkin görüşleri* [Yüksek Lisans Tezi]. Hatay Mustafa Kemal Üniversitesi.
- Sun, L., Guo, Z., & Zhou, D. (2023). Measuring development of young students' coding ability through a graphical teaching intervention: further explanation of the effect of coding experience and coding interest. *Interactive Learning Environments*, 1-24. <https://doi.org/10.1080/10494820.2023.2190356>
- Şendağ, S., Yakin, I., ve Gedik, N. (2023). Fostering creative thinking skills through computer programming: Explicit or integrated teaching?. *Education and Information Technologies*, 28(9), 10819-10838. <https://doi.org/10.1007/s10639-023-11629-4>

- Taşkapı, C. (2015). *Öğretmen adaylarının öğrenme stillerinin çeşitli değişkenler açısından incelenmesi ve öz düzenleme becerileri ile olan ilişkisi*. [Yüksek Lisans Tezi]. Doğu Akdeniz Üniversitesi.
- Temel, S., Kansu, C. Ç., ve Temel, N. (2023). İlkokul dördüncü sınıf öğrencilerinin başarı algıları ile öz düzenleme becerileri arasındaki ilişki. *Milli Eğitim Dergisi*, 52(237), 167-198.
- Tiryaki, A. (2020). *Robotik kodlama eğitiminin ortaöğretim öğrencilerinin programlama öz yeterlik düzeylerine ve yaratıcı düşünme becerilerine etkisi*. [Yüksek Lisans Tezi]. Mustafa Kemal Üniversitesi.
- Tonbuloğlu, B. (2022). Bilgisayarsız programlama öğretimi yaklaşımları. Kert S.B. (Eds.). *Programlama öğretimi yaklaşımları* (pp. 87-118). Nobel Akademik Yayıncılık
- Tosuntaş, Ş. B., Emirtekin, E., ve Kircaburun, K. (2020). Kodlama eğitiminde işbirlikli öğrenme yaklaşımı: eşli kodlama. *OPUS International Journal of Society Researches*, 16(27), 490-515. <https://doi.org/10.26466/opus.680327>
- Totan, H. N. (2021). *Blok tabanlı kodlama eğitiminin ortaokul öğrencilerinin bilgi işlemsel düşünme becerileri ve kodlama öğrenimine yönelik tutumlarına etkisi: blocky örneği*. [Yüksek Lisans Tezi]. Necmettin Erbakan Üniversitesi.
- Tsai, C. Y. (2019). Improving students' understanding of basic programming concepts through visual programming language: The role of self-efficacy. *Computers in Human Behavior*, 95, 224-232. <https://doi.org/10.1016/j.chb.2018.11.038>
- TTKB. (2023). *İlköğretim kurumları (ilkokul ve ortaokul) haftalık ders çizelgesi*. [Çevrim-ichi: https://ttkb.meb.gov.tr/meb_iys_dosyalar/2023_09/14104730_2023-43_Temel_Egitim_Ylkokul_Ortaokul.pdf], Erişim tarihi: 07.10.2023.
- Tuomi, P., Multisilta, J., Saarikoski, P., & Suominen, J. (2018). Coding skills as a success factor for a society. *Education and Information Technologies*, 23, 419-434. <https://doi.org/10.1007/s10639-017-9611-4>
- Turan, İ., Akça, M., ve Küçük Kurt, M. (2016). *Temel kodlama eğitimi*. Pusula Yayıncılık
- Tülübaş, T. (2022). Çevrim içi öğrenmede öz-düzenleme becerisinin akademik başarıya etkisi. *Anadolu Journal of Educational Sciences International*, 12(2), 389-416. <https://doi.org/10.18039/ajesi.1021613>
- Türker, P. M., & Pala, F. K. (2019). A study on students' computational thinking skills and self-efficacy of block-based programming. *i-Manager's Journal on School Educational Technology*, 15(3), 18-31.

- URL-1 (2019). *Europe code week news*. <https://blog.codeweek.eu/post/182448459500/eu-code-week-2018-breaks-all-time-record-with-27>. Eriřim tarihi: 12.30.2023.
- URL-2 (2023). *Hakkında*. <https://codeweek.eu/about>, Eriřim tarihi: 30.09.2023.
- URL-3 (2023). *Ders 6: Labirent: Döngüler*. <https://studio.code.org/s/course2/lessons/6/levels/4>. Eriřim Tarihi: 06.10.2023.
- URL-4 (2023). *Hakkımızda*. <https://code.org/about>. Eriřim Tarihi: 30.09.2023.
- URL-5 (2023). *Kod haftası nedir?*. https://codeweekturkiye.eba.gov.tr/?page_id=1061, Eriřim Tarihi: 10.10.2023.
- URL-6 (2023). *Kod haftası 2022 raporu*. https://codeweekturkiye.eba.gov.tr/?page_id=705. Eriřim Tarihi: 10.10.2023.
- URL-7 (2023). www.kesfetprojesi.org/files/uploads/6-sinif-bilisim-teknolojileri-ve-yazilim-dersi-ogretmen-rehberi.pdf. Eriřim Tarihi: 25.12.2023.
- Uzun, A., ve Baltalı, S. (2020). Programlama öğretiminde kullanılabilecek yazılımlara ilişkin öğretmen görüşleri. *Journal of Uludag University Faculty of Education*, 33(1), 129-156. <https://doi.org/10.19171/uefad.561833>
- Van-Roy, P., & Haridi, S. (2004). *Concepts, techniques, and models of computer programming*. MIT pres.
- Veerasamy, A. K., D'Souza, D., Lindén, R., & Laakso, M. J. (2019). Relationship between perceived problem-solving skills and academic performance of novice learners in introductory programming courses. *Journal of Computer Assisted Learning*, 35(2), 246-255. <https://doi.org/10.1111/jcal.12326>
- Vieira, C., Magana, A. J., Falk, M. L., & Garcia, R. E. (2017). Writing in-code comments to self-explain in computational science and engineering education. *ACM Transactions on Computing Education (TOCE)*, 17(4), 1–21. <https://doi.org/10.1145/3058751>
- Wang, L., Geng, F., Hao, X., Shi, D., Wang, T., & Li, Y. (2021). Measuring coding ability in young children: relations to computational thinking, creative thinking, and working memory. *Current Psychology*, 1-12. <https://doi.org/10.1007/s12144-021-02085-9>
- Wei, X., Lin, L., Meng, N., Tan, W., & Kong, S. C. (2021). The effectiveness of partial pair programming on elementary school students' computational thinking skills and self-efficacy. *Computers & education*, 160, 104023. <https://doi.org/10.1016/j.compedu.2020.104023>

- Weintrop, D., & Wilensky, U. (2018). How block-based, text-based, and hybrid block/text modalities shape novice programming practices. *International Journal of Child-Computer Interaction*, 17, 83-92. <https://doi.org/10.1016/j.ijcci.2018.04.005>
- Wilson, K., & Narayan, A. (2016). Relationships among individual task self-efficacy, self-regulated learning strategy use and academic performance in a computer-supported collaborative learning environment. *Educational Psychology*, 36(2), 236-253. <https://doi.org/10.1080/01443410.2014.926312>
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-35. <https://doi.org/10.1145/1118178.1118215>
- World Economic Forum. (2020). *The Future of Jobs Report 2020*. Geneva: World Economic Forum.
- Xinogalos, S. (2016). Designing and deploying programming courses: Strategies, tools, difficulties and pedagogy. *Education and Information Technologies*, 21, 559-588. <https://doi.org/10.1007/s10639-014-9341-9>
- Yalçın, S. (2018). 21. Yüzyıl becerileri ve bu becerilerin ölçülmesinde kullanılan araçlar ve yaklaşımlar. *Ankara University Journal of Faculty of Educational Sciences (JFES)*, 51(1), 183-201. <https://doi.org/10.30964/auebfd.405860>
- Yanpar Yelken, T., ve Akay, C. (2019). *Öğretim ilke ve yöntemleri*. Anı Yayıncılık.
- Yıldız Durak, H. (2020). The effects of using different tools in programming teaching of secondary school students on engagement, computational thinking and reflective thinking skills for problem solving. *Technology, Knowledge and Learning*, 25, 179-195. <https://doi.org/10.1007/s10758-018-9391-y>
- Yıldız, A. G. M., Çifçi, E., ve Karal, H. (2017). Bilişimsel düşünme ve programlama. H.F. Odabaşı, B. Akkoyunlu, A. İşman, (Ed.). *Eğitim teknolojileri okumaları 2017 içinde* (s. 75-86.). The Turkish Online Journal of Educational Technology.
- Yıldız, E., Ağgöl, Ö., Çalıklar, Ş., ve Şimşek, Ü. (2020). Eğitsel oyun ve işbirlikli öğrenmenin 6. sınıf öğrencilerinin akademik başarılarına, sosyal becerilerine ve öğrenme motivasyonlarına etkisi. *Anemon Muş Alparslan Üniversitesi Sosyal Bilimler Dergisi*, 8(6), 1703-1716. <https://doi.org/10.18506/anemon.671575>
- Yıldız, T. (2019). *Programlama öğretiminde akran öğretimi yönteminin öğrencilerin derse karşı tutum, akademik başarı ve programlama öz-yeterliklerine etkisi*. [Doktora Tezi]. Necmettin Erbakan Üniversitesi.

- Yıldız, V. (1999). İşbirlikli öğrenme ile geleneksel öğrenme grupları arasındaki farklar. *Hacettepe Üniversitesi Eğitim Fakültesi Dergisi*, 16(17), 155-163.
- Yılmaz, Ş. (2019). *Scratch programı öğretiminde birlikte öğrenme tekniği kullanımının öğrencilerin akademik başarısına ve öz yeterlik algısına etkisi*. [Yüksek Lisans Tezi]. Afyon Kocatepe Üniversitesi.
- Yurdabakan, İ. (2022). Eğitimde kullanılan ölçme araçlarının nitelikleri. S. Erkan, M. Gömleksiz, (Eds.). *Eğitimde ölçme ve değerlendirme içinde* (s. 38-66). Nobel Akademik Yayıncılık
- Yüksel, S. (2017). *Scratch programı öğretiminde ayrılıp birleşme tekniği kullanımının öğrencilerin derse yönelik tutumuna akademik başarısına ve kalıcılığa etkisi*. [Yüksek Lisans Tezi]. Adnan Menderes Üniversitesi.
- Zhong, B., Wang, Q., & Chen, J. (2016). The impact of social factors on pair programming in a primary school. *Computers in Human Behavior*, 64, 423-431. <https://doi.org/10.1016/j.chb.2016.07.017>
- Zimmerman, B. J. (1989). A social cognitive view of self-regulated academic learning. *Journal of educational psychology*, 81(3), 329. <https://doi.org/10.1037/0022-0663.81.3.329>
- Zurnacı, B., ve Turan, Z. (2022). Türkiye'de okul öncesinde kodlama eğitimine ilişkin yapılan çalışmaların incelenmesi. *Kocaeli Üniversitesi Eğitim Dergisi*, 5(1), 258-286. <https://doi.org/10.33400/kuje.1062803>



EKLER

EK A Ölçek Kullanım İzinleri

**Mehmet Dervis** <[redacted]>
Alıcı: [redacted]

26 Ağu 2023 Cmt 13:33 ☆ 😊 ↶ ⋮

Merhaba Sayın Hocam,
Ben Mehmet DERViŞ Kastamonu Üniversitesi Eğitim Programları ve Öğretim Ana Bilim Dalında Yüksek Lisans öğrencisiyim.
Hocam "Algılanan Öz-Düzenleme Ölçeği'nin Geliştirilmesi: Geçerlik ve Güvenirlilik Çalışması" adlı çalışmamızı inceledim. "Algılanan Öz-Düzenleme Ölçeği" eğer sizin için de uygunsa tezimde kullanmak istiyorum.
Yardımlarınız ve desteğiniz için şimdiden çok teşekkür ederim.
Saygılarımla

**Yücel Gelişli** <[redacted]>
Alıcı: ben

27 Ağu 2023 Paz 23:27 ☆ 😊 ↶ ⋮

Merhaba
Ölçeği tez çalışmada kullanabilirsin, başarılar

Kimden: "Mehmet Dervis" <[redacted]>
Kime: [redacted]
Gönderilenler: 26 Ağustos Cumartesi 2023 13:33:42

**Mehmet Dervis** <[redacted]>
Alıcı: [redacted]

1 Ağu 2023 Sal 12:20 ☆ 😊 ↶ ⋮

Merhaba Sayın Hocam,
Ben Mehmet DERViŞ Kastamonu Üniversitesi Eğitim Programları ve Öğretim Ana Bilim Dalında Yüksek Lisans öğrencisiyim.
Hocam, sizin "Ortaokul Öğrencileri İçin Programlama Özyeterlik Ölçeği: Geliştirme, Geçerlik Ve Güvenirlilik" adlı çalışmamızı inceledim. "Ortaokul Öğrencileri İçin Programlama Özyeterlik Ölçeğini" eğer sizin için de uygunsa tezimde kullanmak istiyorum.
Yardımlarınız ve desteğiniz için şimdiden çok teşekkür ederim hocam.
Saygılarımla

**SAHİN GOKCEARSLAN**
Merhaba Mehmet Ölçme aracını kullanabilirsiniz. Tezinizde başarılar dilerim. Şahin Kimden: "Mehmet Dervis" <[redacted]> Kime: [redacted] t

1 Ağu 2023 Sal 12:39 ☆

**Mehmet Dervis**
Sayın hocam, ölçeği paylaşabilir misiniz? 1 Ağu 2023 Sal 12:39 tarihinde SAHİN GOKCEARSLAN <[redacted]> şunu yazdı:

1 Ağu 2023 Sal 12:57 ☆

**SAHİN GOKCEARSLAN**
Ektedir Kimden: "Mehmet Dervis" <[redacted]> Kime: "SAHİN GOKCEARSLAN" <[redacted]> Gönderilenler: 1 Ağustos Salı 2023 12:57:51 Konu: R

1 Ağu 2023 Sal 13:02 ☆

**Mehmet Dervis** <[redacted]>
Alıcı: SAHİN

1 Ağu 2023 Sal 13:03 ☆ 😊 ↶ ⋮

Teşekkür ederim, iyi çalışmalar dilerim.

1 Ağu 2023 Sal 13:02 tarihinde SAHİN GOKCEARSLAN <[redacted]> şunu yazdı:

EK B Resmi İzinler



T.C.
KASTAMONU ÜNİVERSİTESİ REKTÖRLÜĞÜ
Sosyal ve Beşeri Bilimler Bilimsel Araştırmalar ve
Yayın Etiği Kurulu

TOPLANTI SAYISI
1

KARAR SAYISI
15

TOPLANTI TARİHİ
10.01.2024

DAĞITIM YERLERİNE

Eğitim Fakültesi'nde Öğretim Üyesi olarak görev yapan Doç. Dr. Mustafa Öztürk AKCAOĞLU'nun yardımcı araştırmacı, Mehmet Derviş'in sorumlu araştırmacı olarak yapmayı planladıkları "**İşbirlikli Öğrenme Yaklaşımı ile Programlama Öğretiminin Ortaokul Öğrencilerinin Programlama Özyeterlikleri, Özdüzenleme Becerileri ve Akademik Başarıları Üzerindeki Etkisi**" isimli çalışması; Kurulumuzca uygun bulunmuştur.

Bu bilgiler ışığında; onam formlarının gönüllülere imzalatılarak gerekli bilgilendirmelerin yapılması, veri toplama araçlarının (anket, ölçek, mülakat,...) uygulama izninin gerekli kurumlardan alınması, çalışmada ses/görüntü kaydı alınacaksa katılımcılardan gerekli izinlerin alınması, veri toplamaya etik kurul kararından sonra başlanması ve etik davranış ilkelerine uyulması şartıyla söz konusu araştırmanın yapılması Etik Kurulumuzca uygun görülmüş ve onaylanmasına toplantıya katılan üyelerin oybirliği ile karar verilmiştir.

Prof. Dr. Muharrem ÇETİN
Kurul Başkanı

Prof. Dr. Yavuz UNAT
Başkan Yardımcısı

Prof. Dr. Kutay OKTAY
Kurul Üyesi

Prof. Dr. Eyüp AKMAN
Kurul Üyesi

Prof. Dr. Burhan BALTACI
Kurul Üyesi

Prof. Dr. Serkan DİLEK
Kurul Üyesi

Prof. Dr. Tolga ULUSOY
Kurul Üyesi

DAĞITIM LİSTESİ

Sosyal ve Beşeri Bilimler Bilimsel Araştırmalar ve Yayın Etiği Kuruluna
Sayın Doç. Dr. Mustafa Öztürk AKCAOĞLU

Belge Doğrulama Kodu: M1119DM1

Bu belge, güvenli elektronik imza ile imzalanmıştır.

Belge Takip Adresi: www.kastamonu.edu.tr/belgedogrulama

Adres: Kastamonu Üniversitesi Rektörlüğü, Kuzeykent Kampüsü, Merkez/Kastamonu

Telefon No: (0 366) 2801012

e-Posta:

Kep Adresi: kastamonuuniversitesi@hs01.kcp.tr

Faks No: (0 366) 2801139

İnternet Adresi: www.kastamonu.edu.tr

Bilgi için :

Serdar Durur

Raportör

Telefon No:

(0 366) 2801000 - 1144



EK B'nin devamı



T.C.
KASTAMONU VALİLİĞİ
İl Millî Eğitim Müdürlüğü



Sayı : E-84616369-44-98579575
Konu : Tez Çalışma İzni (Mehmet DERVİŞ)

11/03/2024

VALİLİK MAKAMINA

İlgi : a) Millî Eğitim Bakanlığı Yenilik ve Eğitim Teknolojileri Genel Müdürlüğünün 21/01/2020 tarihli ve 1563890 sayılı emirleri.
b) Kastamonu Üniversitesinin 01/03/2024 tarih ve 2400029870 sayılı yazısı.

Kastamonu Üniversitesinin ilgi (b) tarih ve sayılı yazısına istinaden Kastamonu Üniversitesi Sosyal Bilimler Enstitüsü Eğitim Bilimleri Anabilim Dalı Tezli Yüksek Lisans programı öğrencisi Mehmet DERVİŞ'in hazırlanmış olduğu "İşbirlikli Öğrenme Yaklaşımı ile Programlama Öğretiminin Öğrencilerin Programlama Özyeterliliğine, Özdüzenleme Becerilerine ve Akademik Başarılarına Etkisi" konulu tez çalışmasını ilimiz Taşköprü İlçesi Alatarla Ortaokulunda 2023-2024 eğitim öğretim yılında gönüllülük esasına göre kurumun eğitim öğretim faaliyetlerini aksatmadan uygulanması ve sonuçlarının değerlendirilmesi Müdürlüğümüzce uygun görülmektedir.

Makamunuzca da uygun görüldüğü takdirde Olurlarınıza arz ederim.

Hasan GÜMÜŞ
İl Millî Eğitim Müdürü

OLUR
Aydın ERGÜN
Vali a.
Vali Yardımcısı

Ek :
1-İlgi yazı ve ekleri
2-Değerlendirme Formu
3-Anket Formu

Bu belge güvenli elektronik imza ile imzalanmıştır.

Adres : Alınış Mah.Yalıncı Cad.12A 37100 Merkez KASTAMONU

Belge Doğrulama Adresi : <https://www.turkiye.gov.tr/meb-eb>

Telefon No : 0 (366) 290 37 00

Bilgi için: Lutfiye ÇALIŞKAN

E-Posta:

İnternet Adresi: www.kastamonu.gov.tr

Ünvan: Memur

Kayıt Adresi : meb@be01.ksp.tr

Faks: 3662903709

Bu belge güvenli elektronik imza ile imzalanmıştır. <https://www.sagun.gov.tr/adresler/2b71-7b6f-3828-8eb7-59dc> kodu ile doğrulanabilir.

EK C Günlük Planlar

Ders: Bilişim Teknolojileri ve Yazılım		
Sınıf: 6	Hafta: 2	Süre: 40 dk+40 dk
Ünite: Problem Çözme ve Programlama		
Konular: BT.6.2.4 Problem Çözmek Benim İşim! • Algoritma ve faydaları • Algoritma örnekleri • Algoritma oluşturma • Hatalı algoritma bulma - Algoritma- problem çözme bağlantısı		
Kazanımlar: • BT.6.5.1.6. Bir algoritmanın çözümünü test eder. • BT.6.5.1.7. Farklı algoritmaları inceleyerek en hızlı ve doğru çözümü seçer. • BT.6.5.1.8. Hatalı bir algoritmayı doğru çalışacak biçimde düzenler.		
Öğretme, Öğrenme Yöntem ve Teknikleri: • Buluş yolu Stratejisi • Beyin fırtınası • Örnek olay • İşbirlikli Öğrenme Tekniği: Öğrenci Takımı Başarı Bölümleri		
Kullanılan Öğretim Teknolojileri, Araç-Gereçler ve Kaynakça: • MEB Öğretmen kılavuzu ve sunumları • Bilgisayarsız kodlama etkinlikleri, • Algoritma çalışma yaprağı		
ÖĞRENME VE ÖĞRETME ETKİNLİKLERİ: 1.Dikkat Çekme: Öğretmen derste bir öğrenciyi tahtaya kaldırır ve öğrencinin okulun bulunduğu köydeki caminin yerini adım adım tarif etmesi sağlar. 2.Güdüleme: Öğretmen öğrencilere günlük hayatımızdaki her eylemin adımlardan oluştuğunu, bilgisayarlarında her işlemi adım adım yaptığından bahsedilir. 3.Hedeften haberdar etme: Öğretmen, eylemlerimizin adımlardan oluştuğunu buna algoritma olarak adlandırıldığını bu derste algoritmanın ne olduğunu öğreneceklerini söyler. 4.Ön koşul öğrenmelerinin hatırlatılması: Geçen hafta öğrenilen problemi çözme basamaklarının bir algoritma olduğunu adım adım ilerlediği belirtilir. Soru-cevaplarla geçmiş öğrenmeler ortaya çıkarılır. 5.Ders İçi Etkinlikler: Öğretmen öğrenci gruplarını oluşturduktan sonra öğrencilere gruplara ayrılmadan düz anlatım yöntemiyle algoritmanın ne olduğunu özelliklerini ve faydalarını kavram haritaları kullanarak öğrencilere anlatır. Konu anlatımı içinde öğrencilere soru-cevap etkinlikleriyle konuyu özümsemesini sağlar. Öğretmen buluş yolu stratejisi kullanarak öğrencilerin günlük hayatta yaptığı birçok etkinliğin algoritmik adımlardan oluştuğunu öğrenmelerine kılavuzluk eder.		

EK C'nin devamı

5.1. İşbirlikli Öğrenme Etkinliği: Öğrenci Takımı Başarı Bölümleri

Öğretmen kendini örneklerini sunduktan sonra öğrencilerin 5 kişilik öğrenci gruplarına ayırır ve her gruba çalışma kağıdı dağıtır. Çalışma kağıtlarında çeşitli algoritma oluşturma soruları yer almaktadır öğrencilerin algoritma oluşturmalarını sağlayacak grup çalışmaları vardır. Çalışma kağıdında yer alacak olan konular:

- Algoritma oluşturma,
- Hatalı algoritma bulma,
- Algoritma- problem çözme bağlantısı,

Çalışma kağıdında yer alan sorular grup öğrencileri tarafından tartışılarak birlikte yapılır. Öğretmen gerektiği yerlerde öğrencilere kılavuzluk yapar. Buluş yöntemini kullanarak öğrencilerin bilgilerini derinleştirmesine olanak sağlar.

Grup çalışması puan verilerek değerlendirilir. Grubun birinci puanı belirlenir. Daha sonra öğretmen tüm öğrencilere bireysel bir test dağıtır ev öğrenciler cevapladıktan sonra bireysel testlerden alınan puanlar aynı grupta olanlar için birlikte toplanır daha önceki grup puanına eklenir. Böylece grupların başarı puanları sıralanmış olur. Hem grup içi hemde gruplar arasındaki başarı durumları ortaya çıkarılır.

Ölçme ve Değerlendirme :

1.Bireysel öğrenme etkinliklerine yönelik ölçme-değerlendirme:

Öğrencilere 3 soruluk mini quiz yapılır. Quizden aldıkları puanların toplanır bireysel sonuç ortaya çıkar.

Soru 1-Günlük hayatta en sık yaptığınız davranışın algoritmasını çıkarın.

Soru 2-Algoritmanın faydalarına iki örnek veriniz.

Soru 3-Verilen algoritmanın hatalı adımını bulun.

(1) Başla

(2) Birinci sayıyı gir.

(3) Sayıları topla

(4) ikinci sayıyı gir.

(5) Sonucu ekrana yaz.

(6) Bitir.

Ders: Bilişim Teknolojileri ve Yazılım

Sınıf: 6

Hafta: 3

Süre: 40 dk+40 dk

Ünite: Problem Çözme ve Programlama

Konular: BT.6.2.10 Blok Tabanlı Kodlama Programını Keşfediyorum

- Blok tabanlı Programlama yazılımı indirme ve kurma
- Blok tabanlı Programlama yazılımı arayüzü ve özellikleri tanıtımı

Kazanımlar:

- BT.6.5.2.1. Blok tabanlı programlama aracının arayüzünü ve özelliklerini tanıtır.
- BT.6.5.2.3. Blok tabanlı programlama aracında sunulan bir programın hatalarını ayıklar.

EK C'nin devamı

Öğretme, Öğrenme Yöntem ve Teknikleri: • Buluş yolu Stratejisi • Beyin fırtınası • Örnek olay • İşbirlikli Öğrenme Tekniği: Öğrenci takımları başarı bölümleri, Eşli programlama
Kullanılan Öğretim Teknolojileri, Araç-Gereçler ve Kaynakça: • Scratch uygulaması • Code.org uygulaması • BTP arayüz çalışma yaprağı
ÖĞRENME VE ÖĞRETME ETKİNLİKLERİ: 1.Dikkat çekme: Öğretmen derse yapboz getirir, öğrencilere yapboz parçalarının nasıl birleştirdiklerini sorar. Öğrencilerden birini tahtaya kaldırarak basit yapbozu yaptırır. 2.Güdüleme: Öğretmen öğrencilere kodlamanın yapboz parçaları gibi iç içe geçecek şekilde birleştirildiğini söyler, günlük hayatımızdaki tüm yazılımların buna benzer yapıdan oluşturulduğunu söyler. 3.Hedeften haberdar etme: Öğretmen, derste kodlama sayesinde oyun, animasyon gibi yazılımlar oluşturmak için kodlama platformunu tanıyacaklarını açıklar. 4.Ön koşul öğrenmelerinin hatırlatılması: Geçen hafta öğrenilen algoritma konusunu ortaya çıkarmak için öğretmen öğrencilere soru-cevap yapar. Algoritmaların kodlara dönüştürüldüğü söyleyerek iki konu arasında bağlantı kurmaktadır. 5.Ders içi Öğrenme Etkinlikleri: İlk olarak, öğretmen derste gösteri tekniğini kullanarak konuyu anlatmaktadır. Kodlama platformunu indirmeyi ve bilgisayara kurmayı, arayüzünü anlatır. Kukla değiştirme ve ekleme, dekor ekleme, kodlama bloklarının özelliklerinden ve çizim bölümünü soru-cevap tekniğini kullanarak öğrencileri öğrenme etkinliğinin içine çekerek göstermektedir. Öğretmen ikili öğrenme grupları kurar. Öğrencilere öğretmenin anlattığı konuyu özümsemesi için serbest çalışma zamanı verir. Öğrenciler bu süre içinde eşli-programlama grubu kurarlar yardımlaşır, dijital kaynakları kullanarak öğretmenin anlattığı konuyu uygular ve derinleştirirler. 5.1 İşbirlikli Öğrenme Etkinliği: Öğrenci takımları başarı bölümleri tekniği, Eşli programlama. Öğretmen bilgisayar başına oturan eşli-ikili gruplar (Eşli programlama) oluşturur. Bu gruplar öğrenme süreci içerisinde görev paylaşımı yapmaktadır. Öğretmen, öğrencilere 2 adet görev vermiştir. İkili öğrenci gruplarından birinci üye kendine verilen görevi bilgisayar kullanıcısı olarak yerine getirirken arkadaşı hatalı ve eksik yerlerde rehberlik yapar. Daha sonra üyeler karşılıklı görev değişimi yaparlar ikinci üye kendine verilen görevi bu kez bilgisayarı kullanarak yerine getirir. Birinci üye (Bilgisayar kullanıcısı): Orman dekoru ekleme, ayı kuklası ekleme ve ayı kuklasını sahne içinde yürütme kodu yazmak. İkinci üye (Rehberlik görevli): Çöl dekoru ekleme, papağan kuklası ekleme ve papağan kuklasını uçurma ve kanat çırpma efekti vermek. Öğretmen bu süreç sonunda ikili gruplar için etkinlik kağıdı verir grup puanı belirler.

EK C'nin devamı

Daha sonra BTP arayüzüne dair bilgileri ölçen test dağıtır öğrencilerin bireysel puanları belirlenir. Grup ve bireysel puanlar toplanarak grubun yeni puanı ortaya çıkarılır.

Ölçme ve Değerlendirme :

1. Blok tabanlı kodlama nedir?
2. Doğru olan cümlelerin başına (D) yanlış olan cümlelerin başına (Y) yazınız.
() Sahneye eklenen dekorlar eklendikten sonra değiştirilemez.
() Çizim şeklindeki kuklaların renkleri “kostümler” sekmesinden değiştirilmektedir.
() Hareket, Görünüm, ses gibi özelliklere sahip alana kod blokları denilmektedir.
() Bilgisayardan kodlama platformuna kukla eklenemez.
3. Algoritma konusundaki sabit ve değişkenler ifadelerin kod karşılığı olan kod bloğunun adı nedir?

DERS: Bilişim Teknolojileri ve Yazılım

SINIF: 6

HAFTA: 4

SÜRE: 40 dk+40 dk

ÜNİTE: Problem Çözme ve Programlama

KONULAR: BT.6.2.12 - Nesneler Konuşuyor

- Doğrusal programlama
- Hareket blokları
- Görünüm blokları
- Olaylar blokları

KAZANIMLAR:

BT.6.5.2.5 Doğrusal mantık yapısını içeren programlar oluşturur.

ÖĞRETME, ÖĞRENME YÖNTEM VE TEKNİKLERİ:

- Buluş yolu stratejisi
- Beyin fırtınası
- İşbirlikli öğrenme tekniği: Öğrenci takımları başarı bölümleri, Eşli programlama

KULLANILAN ÖĞRETİM TEKNOLOJİLERİ, ARAÇ-GEREÇLER VE KAYNAKÇA:

- Scratch uygulaması
- Çalışma yaprağı

EK C'nin devamı

ÖĞRENME VE ÖĞRETME ETKİNLİKLERİ:

1.Dikkat çekme: Öğretmen sınıfa iki yapboz getirir, yapbozlardan birini masanın üstüne dağıtır, diğerini ise yapılmış şekliyle masanın üstüne koyar. Öğretmen tablo olarak asmak istense hangi yapbozu asarsınız diye sorar? Gelen cevaplar eşliğinde BTP kodlarını düzenleyeceklerini açıklar.

2.Güdüleme: Öğretmen öğrencilere bugün öğrenecekleri doğrusal programlama ile kodlar yazarak günlük hayatlarında çözmek istedikleri problemlere yönelik yazılım üreteceklerini açıklar.

3.Hedeften haberdar etme: Öğretmen öğrencilere önceki derste gördükleri BTP platformundaki kod yapılarını art arda ve iç içe dizerek doğrusal yazılım üreteceklerini açıklar.

4.Ön koşul öğrenmelerinin hatırlatılması: Geçen hafta öğrenilen BTP platformunun arayüzü ile alakalı bilgilerini ortaya çıkarmak için öğretmen öğrencilere soru-cevap yapar.

5.İşbirlikli Öğrenme Etkinlikleri: Öğrenci takımları başarı bölümleri, Eşli Programlama

Öğretmen öğrencilere kod bloklarını nasıl bir araya getireceklerini, bir amaç doğrultusunda kurgulayacaklarını ve nasıl çalıştıracaklarını gösterip-yaptırma ile anlatır. Öğretmen sonraki aşamada öğrencileri iki kişilik gruplara ayırır (Eşli programlama). Öğrenciler Code.org platformunda Code.org /Kurs 2/Ders 6'da yer alan etkinlikleri sıra sıra yaparlar. Bu etkinlikler sırasında en erken bitiren gruba 10 puan, ikinci sırada bitiren 9 puan, üçüncü sırada bitiren gruba 8 puan verilecek şekilde gruplar sıralanırlar. Grupların bu etkinlikler esnasında programlamanın mantığını sunduktan sonra öğrencilere farklı BTP platformunda bloklar yardımıyla kendi kodlarını yazmalarını ister. Öğrencilere iki farklı senaryo verir. Birinci senaryo birinci grup üyesinin bilgisayar kullanıcısı olduğu ikinci grup üyesinin rehber ve yönlendirici olduğu; ikinci senaryo ise ikinci grup üyesinin bilgisayar kullanıcısı, birinci üyenin ise rehber-yönlendirici olduğu senaryodur. Öğretmen süreç sonunda bireysel olarak yapılan ürünü rubrik ile değerlendirir. Grup puanı ve bireysel puanlar toplanır grubun yeni puanı belirlenir.

ÖLÇME VE DEĞERLENDİRME:

Bu aşamada ölçme ve değerlendirme Grupla yapılır. Grup üyelerinin puanlarının yarısı bireylerin puanını oluşturur.

Soru: BTP platformundan uçan papağan karakterini ekleyip kodları biraraya getirerek uçan papağan animasyonunu programlamayınız.

Ders: Bilişim Teknolojileri ve Yazılım		
Sınıf: 6	Hafta: 5	Süre: 40 dk+40 dk
Ünite: Problem Çözme ve Programlama		
Konular: BT.6.2.12 - Nesneler Konuşuyor		
- Karar ve çoklu karar yapıları - Kontrol blokları / eğer ise, eğer kodları		

EK C'nin devamı

Kazanımlar: • BT.6.5.2.7. Karar yapısını içeren programlar oluşturur. • BT.6.5.2.9. Çoklu karar yapıları içeren programlar oluşturur.
Öğretme, Öğrenme Yöntem ve Teknikleri: • Buluş yolu Stratejisi • Beyin fırtınası • Örnek olay • İşbirlikli Öğrenme Tekniği: Takım-Oyun-Turnuva, eşli kodlama
Kullanılan Öğretim Teknolojileri, Araç-Gereçler ve Kaynakça: • Scratch uygulaması • Code.org uygulaması • Karar yapıları çalışma yaprağı
ÖĞRENME VE ÖĞRETME ETKİNLİKLERİ: 1.Dikkat çekme: Öğretmen öğrencilere günlük hayatlarında iki seçenek arasında hiç karar verip vermediklerini sorar. Gelen cevaplardan yola çıkarak öğretmen öğrencilere: Bugün hava yağmurlu mu? Yağmurlu ise yanına şemsiye al. Eğer yağmurlu değilse şemsiye alma der. 2.Güdüleme: Öğretmen öğrencilere bilgisayar yazılımları oluşturulurken çoğu kez iki cevaplı karar yapıları olduğunu açıklar. 3.Hedeften haberdar etme: Öğretmen, öğrencilerine bugüñ kodlama aracındaki kod bloklarından karar yapılarını öğrencelerini söyler. 4.Ön koşul öğrenmelerinin hatırlatılması: Geçen hafta öğrenilen doğrusal programlama ile alakalı bilgilerini ortaya çıkarmak için öğretmen öğrencilere soru-cevap yapar. Bu özelliklerden yeni bir kod olan karar yapılarını gösterir ve iki konu arasında bağlantı kurar. 5.Ders İçi Etkinlikler: Öğretmen karar verme yapılarına ait kodları gösterip-yaptırma yöntemi ile anlatır. Öğrenme süreci içinde sor-cevap yöntemi, kullanır. Ayrıca karar yapılarını iki kola ayrılan yola benzeterek öğrencilere metafor yaptırır. İşbirlikli öğrenme etkinlikleri: Takım – Oyun – Turnuva Tekniği Öğretmen öğrencileri ikili gruplara ayırır gruptaki üyelere bir ve iki numaraları verilir ve her gruba iki adet kod yazacakları senaryo verir. Öğrencilerden bu senaryoları grup arkadaşlarıyla çalışırlar. Daha sonra aynı numaraya sahip öğrenciler aynı kodlama senaryosunu bireysel olarak yapmaya çalışırlar. En erken bitiren grubuna 10, ikinci bitiren 9, üçüncü bitiren 8 şeklinde öğrencilere puanlar verilir. Öğrencilerin yazdıkları kodlar rubrik ile değerlendirilir. Aynı gruptaki öğrencilerin puanları toplanarak grup puanı belirlenir.

EK C'nin devamı

ÖLÇME VE DEĞERLENDİRME

Soru-1: Aşağıda verilen algoritmanın kodları yanında verilmiştir. Eksik bırakılan kodları yazınız.

ALGORİTMA

- 1-Başla
- 2-Yaşın ne? diye sor.
- 3- Yaş 18'den büyük mü?
- 4-Eğer cevap evet ise 6. adıma git.
- 6-Ekranda 5 saniye “ehliyet alabilirsin” de.
- 8-Bitir.

KOD



Ders: Bilişim Teknolojileri ve Yazılım		
Sınıf: 6	Hafta: 6	Süre: 40 dk+40 dk
Ünite: Problem Çözme ve Programlama		
Konular: BT.6.2.13 - Düşünen Bilgisayar • Tekrarlama(Döngü) blokları • Haber blokları • İşlem blokları		
Kazanımlar: • BT.6.5.2.11. Döngü yapısını içeren programlar oluşturur.		
Öğretme, Öğrenme Yöntem ve Teknikleri: • Buluş yolu Stratejisi • Beyin fırtınası • Örnek olay • İşbirlikli Öğrenme Tekniği: Eşli programlama		
Kullanılan Öğretim Teknolojileri, Araç-Gereçler ve Kaynakça: • Scratch uygulaması • Code.org uygulaması • Döngüler çalışma yaprağı		
Öğrenme-Öğretme Etkinlikleri: 1.Dikkat çekme: Öğretmen öğrencilere günlük hayatlarında tekrarlayarak yaptıkları davranışlarını sorar. Sınıfa getirdiği rüzgar gülünü üzerine 5 sayısını yazar, rüzgar gülü her döndüğünde 5 sayısını dönme sayısı kadar arttırıldığını açıklar. Öğretmen öğrencilerine rüzgar gülü 10 kere döndüğünde sonucu sorar. 2.Güdüleme: Öğretmen öğrencilere bilgisayar yazılımları oluşturulurken bazen kodların tekrarladığını bu amaçla döngü kodlarının ne olduğunu açıklar.		

EK C'nin devamı

3.Hedeften haberdar etme: Öğretmen, öğrencilerine bugün kodlama aracındaki kod bloklarından döngü(tekrarlama) bloklarını öğreneceklerini açıklar.

4.Ön koşul öğrenmelerinin hatırlatılması: Geçen hafta öğrenilen kodlama aracını arayüzünün özellikleri ile alakalı bilgilerini ortaya çıkarmak için öğretmen öğrencilere soru-cevap yapar. Bu özelliklerden yeni bir kod olan karar yapılarını gösterir ve iki konu arasında bağlantı kurar.

5.Ders İçi Öğrenme Süreçleri: Öğretmen öğrencilere gösterip-yatırma yöntemi ile döngü yapılarını anlatır. Öğrenciler, Code.org / Kurs 2 / Ders 6: Döngüler etkinliklerini birlikte yaparlar.

İşbirlikli öğrenme etkinlikleri: Eşli programlama

Öğretmen öğrencilere bilgisayarsız kodlama ve matematiksel işlemleri barındıran bir çalışma kağıdı dağıtır. Gruptaki öğrencilere numaralar verilir, aynı numaraya sahip öğrenciler biraraya gelerek aynı konuyu çalışır bu sırada öğretmen öğrencilere sorular yönelterek çalışma kağıdındaki döngü yapılarını kavramasını sağlar. Daha sonra kendi gruplarına dönen öğrenciler birbirlerine döngü yapısına dair anladıklarını anlatırlar. Sonraki aşamada öğretmen karşılıklı sorgulama amacıyla iki kişilik yeni takımlar oluşturur, takımın bir üyesi bilgisayar kullanıcısı rolünde diğer üyesi ise arkadaşına rehberlik yapma rolünü üstlenmiştir.

Etkinlik I: Birinci takım üyesi (bilgisayar kullanıcısı rolü) görevi, tekrarlama bloğunu kullanarak 1'den başlayıp 20'ye kadar sayan bir sayaç yapması istenir. Bu sıra da takımın diğer üyesi (rehberlik yapan) rolü ise arkadaşına sorular sorarak yaptığı iş ve işlemler hakkında bilgi edinmektir.

Süreç sonunda öğrenciler rolleri değiştirirler öğretmen ikinci takım üyesine bir soru sorar ve onu bilgisayar kullanıcı yapar. Birinci takım üyesi ise arkadaşını yönlendiren sorular soran rol üstlenir.

Etkinlik II: İkinci takım üyesi görevi (bilgisayar kullanıcısı rolü), 100 adım ileriye giden karakterin kodlarını ... defa tekrarla bloğu kullanarak yapması istenir. Bu sıra da takımın birinci üyesi rehberlik rolü üstlenir arkadaşına sorular sorarak yaptığı iş ve işlemler hakkında bilgi edinmektir.

Öğrenciler bu etkinlikleri yaptıktan sonra öğrenciler birbirlerinin çalışmalarını denetler ve eksik kısımlar için yardımlaşırlar.

Ölçme ve Değerlendirme :

Soru-1: Aşağıda verilen algoritmanın kodları eksik bırakılmıştır. Yengecin 4 numaralı kareye ulaşması gerekmektedir. Yengecin 4 numaralı kareye gidiş yolunu öğrendiğiniz döngü ve hareket kodlarını kullanarak yapınız.

1	2	3	4
8	7	6	5
	10	11	12
13	14	15	16

EK C'nin devamı

Ders: Bilişim Teknolojileri ve Yazılım		
Sınıf: 6	Hafta: 7	Süre: 40 dk+40 dk
Ünite: Problem Çözme ve Programlama		
Konular: • Oluşturulan programları test etme • Hata yakalama • Algılama blokları • Değişken oluşturma		
Kazanımlar: • Karar yapısını içeren programları test ederek hatalarını ayıklar. • Çoklu karar yapısını içeren programları test ederek hatalarını ayıklar.		
Öğretme, Öğrenme Yöntem ve Teknikleri: • Buluş yolu Stratejisi • Beyin fırtınası • Örnek olay • İşbirlikli Öğrenme Tekniği: Takım-Oyun -Turnuva		
Kullanılan Öğretim Teknolojileri, Araç-Gereçler ve Kaynakça: • Scratch uygulaması • Code.org uygulaması • Test etme ve hata yakalama çalışma yaprağı		
ÖĞRENME VE ÖĞRETME ETKİNLİKLERİ: 1.Dikkat çekme: Öğretmen öğrencilere günlük hayatlarında yeni aldıkları ürünlerin bozuk çıkma ihtimaline karşı test edip etmediklerini sorar, elinde sınıfa getirdiği el fenerini ve pilleri gösterir. El fenerini yeni aldığını nasıl test edeceğini sorar. 2.Güdüleme: Öğretmen öğrencilere bilgisayar yazılımları oluşturulurken bazen kodların hata verebileceğini bu yüzden test etme ve hata ayıklamanın önemini açıklar. 3.Hedeften haberdar etme: Öğretmen, öğrencilerine önceki haftalarda öğrendikleri kodlama yapılarını kullanarak üretilen programların ve algoritmaları test etmeyi ve hatalarını yakalamalarını öğreceklerini açıklar. 4.Ön koşul öğrenmelerinin hatırlatılması: Önceki hafta öğrenilen kod yapılarını öğrencilere sorular sorarak bilgilerini hatırlaması sağlanır. Yeni konuya ait içerikleri göstererek iki konu arasında bağlantı olduğunu, bu derste önceki öğrenilen bilgileri kullanarak test etme ve hata yakalama işlemleri yapılacağını öğrencilere açıklar. 5.Ders İçi Öğrenme Süreçleri: Öğretmen derste geliştirilen uygulamayı test etmeyi ve hataları ayıklamayı gösterip yaptırma, soru-cevap ve örnek olay yöntemleri kullanarak öğrencilere aktarır. Sonraki süreçte 4'er kişilik beş farklı grup oluşturur. Her gruba hata bulma ve test etme etkinlikleri içeren bir çalışma kağıdı dağıtır.Öğrenciler çalışma kağıtlarında yer alan hataları yakalar ve programları kağıt üzerinde test eder. Sonra bilgisayarda kağıt üzerindeki programları yazarak test ederler. Sonraki süreçte grup üyelerine bireysel test yapılır. Bu test puanlarına göre her üye sıralanır.		

EK C'nin devamı

Ölçme ve Değerlendirme :

Soru-1: Öğrencilerin BTY dersinde aldıkları sınav notunun ortalamasını hesaplayan eğer not 50 den büyükse geçer diyen, eğer not 50'den küçükse kalır diyen programın kodlarını yazıp kendi notlarınızı girerek test edin. (Bu soruyu grup üyeleri birlikte yapar.)

Ev ödevi: Öğrencilerin konuyu pekiştirmesi için hata yakalama ve test etme içeren çalışma kağıdını ev ödevi olarak verilir.

Ders: Bilişim Teknolojileri ve Yazılım		
Sınıf: 6	Hafta: 8	Süre: 40 dk+40 dk
Ünite: Problem Çözme ve Programlama		
Konular: • Tekrarlama blokları • Hata yakalama ve test etme • Değişkenler • İşlemler blokları		
Kazanımlar: • Döngü yapısını içeren programları test ederek hatalarını ayıklar		
Öğretme, Öğrenme Yöntem ve Teknikleri: • Buluş yolu Stratejisi • Beyin fırtınası • Örnek olay • İşbirlikli Öğrenme Tekniği: Öğrenci takımı başarı bölümü		
Kullanılan Öğretim Teknolojileri, Araç-Gereçler ve Kaynakça: • Scratch uygulaması • Code.org uygulaması • Test etme ve hata yakalama çalışma yaprağı		
ÖĞRENME VE ÖĞRETME ETKİNLİKLERİ: 1.Dikkat çekme: Öğretmen tahtaya hatalı bir algoritma yansıtır, öğrencilere algoritmanın doğru çalışıp çalışmadığını sorar. 2.Güdüleme: Öğretmen öğrencilerine döngü kodlarını hatırlatır bu kodların gerçekte birçok teknolojik sistemin çalışmasını sağlayan kod yapısını olduğunu döngü kodunun kullanıldığı sistemler hata verdiğinde nasıl çözeceklerini açıklar. 3.Hedeften haberdar etme: Öğretmen, öğrencilerine döngü yapısında hata yakalama ve döngü yapısı kullanılan algoritmaları test etme konusunu öğreneceklerini açıklar. 4.Ön koşul öğrenmelerinin hatırlatılması:Öğretmen, öğrencilerine döngü kod yapısında hata yakalama ve test etme konusunun, önceki derste işlenen karar yapılarında hata ayıklama ve test etme konusunun devamı olduğunu açıklar. Öğrencilere önceki derste dağıttığı çalışma kağıdının dijital halini etkileşimli tahtaya yansıtır konu ile alakalı sorular sorarak bilgilerini açığa çıkarır.		

EK C'nin devamı

5.Ders İçi Öğrenme Süreçleri: Öğretmen işbirlikli öğrenme tekniği olan öğrenci takımları başarı bölümleri tekniğine uygun olarak öğrencileri beşli gruplara ayırır ve her gruba birden dörde kadar numara verilir. Öğretmen Bilişim sınıfının ortasında duran masanın üstüne birden dörde kadar numaralanmış içinde döngü kodları içeren algoritmalar yer almaktadır. Öğrenciler her gruptan grubu temsilen bir öğrenci zarfı alırlar kendi gruplarıyla bir araya gelerek bu zarfın içindeki hatalı verilen kodları düzenler ve test ederler. Gruptaki tüm öğrenciler her zarftaki görevleri yerine getirmiş olurlar. Süreç içerisinde öğrenciler takıldıkları yerleri öğretmene sorabilir, öğretmen gerekli yerlerde öğrencileri yönlendirir.

Ölçme ve Değerlendirme :

Soru-1: Döngü kodlarını kullanarak ekrana 10 kere adınızı yazan programın kodlarını yazın. İsimlerinizi girerek test edin. (Grup üyeleri birlikte yapar.)

EK D Programlama Öz-Yeterliği Ölçeği

Programlama Özyeterlik 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.						
		Kesinlikle Katılmıyorum	Katılmıyorum	Kararsızım	Katılıyorum	Kesinlikle Katılıyorum
1	Karşılaştığım bir problemin, programlama problemi olup olmadığını anlayabilirim.					
2	Karmaşık programlama problemlerini, küçük alt problemlere ayırarak çözebilirim.					
3	Bir programlama probleminin çözülüp çözilemeyeceği hakkında yorum yapabilirim.					
4	Programlama problemini çözmek için gerekli bilgiyi araştırırım.					
5	Programlama problemini çözmek için en uygun bilgiyi seçerim.					
6	Problemin çözümüne farklı çözüm yolları ile ulaşabilirim.					
7	Farklı çözüm yollarından hangisinin bir problemin çözümü için en doğrusu olduğunu belirleyebilirim.					
8	Programlama problemlerini çözmek için bana verilen çözüm yollarından farklı çözüm yolları önerebilirim.					
9	Programlama probleminin çözümünü adım adım belirlerim.					
10	Farklı çözüm adımlarından programlama probleminin çözümüne en uygun olanı seçerim.					
11	Çözüm adımlarını kâğıt üzerinde şekiller çizerek gösterebilirim.					
12	Programlama probleminin çözümü için bulduğum adımları arkadaşlarımla paylaşıyorum.					
13	Çözüm adımları yanlış verilen bir programlama problemini düzeltebilirim.					
14	Programlama problemini çözmek için geliştirilen farklı adımları tartışabilirim.					
15	Programlama problemini çözmek için gereken hazırlıkları (değişken ve işlemleri belirleme) yapabilirim.					
16	Programlama değişkenlerinin nasıl kullanıldığını bilirim.					
17	Programlama problemini çözmek için tasarladığım işlemlerin sırasını gerektiğinde değiştirebilirim.					
18	Tekrar eden komutlar yerine döngü kullanabilirim.					

EK D'nin devamı

14	Programlama problemini çözmek için geliştirilen farklı adımları tartışabilirim.					
15	Programlama problemini çözmek için gereken hazırlıkları (değişken ve işlemleri belirleme) yapabilirim.					
16	Programlama değişkenlerinin nasıl kullanıldığını bilirim.					
17	Programlama problemini çözmek için tasarladığım işlemlerin sırasını gerektiğinde değiştirebilirim.					
18	Tekrar eden komutlar yerine döngü kullanabilirim.					

19	Programlamada +, -, *, /, >, <, = operatörlerinin ne anlama geldiğini bilirim.					
20	Programlamanın aşamalarını bilirim.					
21	Program kodlarının nerede yazılacağını bilirim.					
22	Oluşturduğum programı çalıştırabilirim.					
23	Programın hatasız çalışmasını sağlayabilirim.					
24	Programın doğru sonuç üretmesini sağlayabilirim.					
25	Program üzerinde değişiklikler yapabilirim.					
26	Programdaki kodlama ile ilgili hataları düzeltebilirim.					
27	Oluşturduğum programı kaydedebilirim.					
28	Oluşturduğum programı internette başkalarıyla paylaşabilirim.					
29	Bir yazılım projesi geliştirme sürecini açıklayabilirim.					
30	Birden fazla yazılım projesi arasından verilen ölçütlere uygun olanını seçerim.					
31	Yazılım projesi fikrimi adım adım açıklayabilirim.					
*	Kukul, V., Gökçearsan, Ş., & Günbatır, M. S. (2017). Computer programming self-efficacy scale (CPSES) for secondary school students: Development, validation and reliability. <i>Eğitim Teknolojisi Kuram ve Uygulama</i> , 7(1),158-179.					
	Ölçek tek boyuttan oluşmaktadır. Ölçekte ters madde bulunmamaktadır.					

EK E Algılanan Öz-Düzenleme Ölçeği

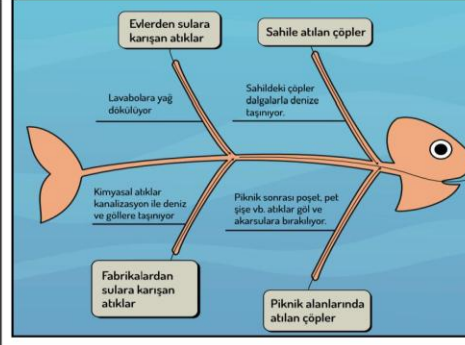
	ALGILANAN ÖZ-DÜZENLEME ÖLÇEĞİ	Hiç bir zaman	Nadiren	Arasıra	Sık sık	Her zaman
1	Eğer istersem en zor konuları bile rahatlıkla öğrenebilirim	1	2	3	4	5
2	Belirlediğim hedefler doğrultusunda çalışmalarımı yapabilirim	1	2	3	4	5
3	Yeni bir konuyu rahatlıkla öğrenebilirim	1	2	3	4	5
4	Bir konuyu anlamadığım zaman arkadaşlarımdan yardım isterim	1	2	3	4	5
5	Bir konuyu öğrenirken yenilikleri kolaylıkla fark edebilirim	1	2	3	4	5
6	Bir şeyler istemediğim şekilde giderse bu durum beni rahatsız eder	1	2	3	4	5
7	Hatalarımdan öğrenebilirim	1	2	3	4	5
8	Bir konuyu öğrenirken o dersteki notlarıma bakarak başarımlı sorgularım	1	2	3	4	5
9	Bir konuyu öğrenirken farklı yollar bulmaya çalışırım	1	2	3	4	5
10	Başarısız olduğumda çalışma yöntemimi değiştiririm	1	2	3	4	5
11	Hedeflerime doğru ilerleme sürecimi takip edebilirim	1	2	3	4	5
12	Bir konuyu öğrenirken karşılaştığım problemlerin çözümünü için farklı yollar geliştiririm	1	2	3	4	5
13	Bir konuyu öğrenirken yapmış olduğum plana uyarım	1	2	3	4	5
14	Bir konuyu öğrenirken başka yöntemler kullanmaya çalışırım	1	2	3	4	5
15	Çoğu zaman bir konuyu öğrenirken neler yaptığıma dikkat ederim	1	2	3	4	5
16	Yanlış öğrendiğimi fark ettiğim bir şeyi değiştirmek için birçok farklı yolu deneyebilirim	1	2	3	4	5

EK F Akademik Başarı Testi

BİLİŞİM TEKNOLOJİLERİ VE YAZILIM 6. SINIF PROBLEM ÇÖZME ve PROGRAMLAMA BAŞARI TESTİ

Yönerge: Bu test bilimsel bir araştırma kapsamında programlama ve problem çözme başarınızı ölçmek amacıyla geliştirilmiştir. Toplam 24 soru bulunmaktadır. Çözmeniz için yaklaşık 40 dakikanız bulunmaktadır. Her soru aynı puan değerindedir ve her sorunun bir cevabı vardır. Doğru cevabı yuvarlak içine alıp karalayınız. Başarılar dilerim.

AD: SOYAD: NO: SINIF:



ANA
PROBLEM

1- Yandaki balık kılçığında alt nedenleri analiz edilmiş olan ana problem aşağıdaki seçeneklerden hangisinde yer almaktadır?

- A) Kimyasal kirlenme
- B) Suların kirlenmesi
- C) Piknik alanın kirlenmesi
- D) Hava kirlenmesi

2-“Karmaşık problemler, duruma özgü ve şartlara göre değişebilen çözüm adımlarından oluşan ve alt problemlere ayrılabilen problem türleridir.” Buna göre aşağıdakilerden hangisinde karmaşık problem yer almaktadır?

- A) Kek yapmak
- B) Çay demlemek
- C) Bozulan bilgisayarı tamir etmek
- D) Arabayı yıkamak

Ela: Banu ders çalışmam gerekiyor fakat masamın bacağı kırıldı. Sence ne yapmalıyım?

Banu: Bence marangoza götürüp yeni bir masa bacağı taktrabilirsin Ela.

Ela: Teşekkür ederim Banu bu iyi fikir!

3-Yukarıda iki arkadaşın konuşmasına göre Banu’nun çözüm fikri problem çözmenin hangi basamağına örnektir?

- A) Problemi anlama
- B) Plan Yapma
- C) Planı Uygulama
- D) Çözümü değerlendirme

4-Seçeneklerde verilenlerden hangisi algoritmanın faydalarından biri değildir?

- A) Problemleri daha hızlı ve sistematik olarak çözeriz.
- B) Hatalı işlem yapma olasılığımızı azaltırız.
- C) Çözüme ulaşmak için farklı yolları deneyebiliriz.
- D) Problem çözme sürecini takip etmemize gerek kalmaz.

5- Derste kullanılan kodlama yazılımıyla ilgili aşağıdakilerden bilgilerden hangisi doğrudur?

- A) Sahne’ye eklenen arka plan resimlerine “dekor” denilmektedir.
- B) Sahne üzerindeki “kukla” değiştirilemez.
- C) X konum, 15 derece dön gibi kod blokları “algılama blokları” altında yer alır.
- D) + - * / gibi işlemsel kodlar “hareket blokları” altında yer alır.

EK F'nin devamı

6-Yukarıda yer alan kodlar, kodlama yazılımında hangi kod bloğu başlığı altında yer alır?

A) Değişkenler B) Algılama C) Hareketler D) Kontrol

7- Kodlama yazılımında skor, puan, can, yıldız, toplanan meyve gibi farklı değer alabilen verilerin oluşturulduğu kod hangi kod blokları altında yer alır?

A) Değişkenler B) Algılama C) Hareketler D) İşlemler

(1)- İkinci sayıyı giriniz.
(2)- Ekranı sonucu yaz.
(3)- Bitir
(4)- Birinci sayıyı giriniz.
(5)- Başla
(6)- Sonuç= Birinci sayı + İkinci Sayı

8-Yanda yer alan iki sayıyı toplamını hesaplayan algoritma karmaşık dizilmiştir. Yukarıdan aşağıya, doğru şekilde sıralanma yapılıyor. Buna göre doğru sıralama aşağıdakilerden hangisinde yer alır?

A) 5-4-1-6-2-3
B) 4-1-5-6-2-3
C) 5-6-3-2-1-4
D) 5-1-4-6-2-3

9-Kodlama yazılımında, notun kaç olduğunu soran ve 50'den büyükse ekrana "Geçtin" yazan, 50'den büyük değilse "Kaldın" yazan programın kodu aşağıdakilerden hangisidir?

A

B

C

D

E

10-Yanda yer alan kodlar çalıştığında gerçekleşen işlemin sonucu hangi seçenekte yer almaktadır?

A) 15
B) 5
C) 2
D) 50

EK F'nin devamı

11- Kodlama yazılımda, havanın yağmurlu olup olmadığını soran, cevap evet ise "şemsiye al" cevap hayır ise "şemsiye alma" yazan programın kodu aşağıdakilerden hangisidir?

A	B
C	D

	12- Kodlama yazılımında önce puan değişkeni sonra yanda yer alan kodlar oluşturulmuştur. Buna göre kodlar çalıştığında puan değişkeninin değeri kaç olur? A) 0 B) 20 C) 1 D) 10
--	---

B	A: Kısa kenar B: Uzun Kenar Çevre= 2 * (A+B)			
13- Yukarıda dikdörtgenin çevresinin uzunluğunu hesaplayan formül yer almaktadır. Buna göre dikdörtgenin çevre uzunluğunu hesaplayan algoritma hangi seçenekte yer almaktadır? <table border="1"> <tr> <td data-bbox="370 1556 619 1722"> A 1- Başla 2- Kenar uzunluğunu giriniz. 3- Kenar uzunluğunu 4 ile çarp çevreyi bul 4- Toplam çevre uzunluğunu ekrana yaz. 5- Bitir. </td> <td data-bbox="619 1556 849 1722"> B 1- Başla 2- Kısa kenar uzunluğunu giriniz. 3- Uzun kenar uzunluğunu giriniz. 4- Kısa kenar ve uzun kenarı topla. 5- Kenar toplamını 2 ile çarp. 6- Çevre uzunluğunu ekrana yaz. 7- Bitir. </td> <td data-bbox="849 1556 1098 1722"> C 1- Başla 2- Birinci kenar uzunluğunu gir 3- İkinci kenar uzunluğunu gir 4- Kenarları topla 5- Toplanan kenarları 2'ye böl 6- Sonucu ekrana yaz 7- Bitir. </td> <td data-bbox="1098 1556 1332 1722"> D 1- Başla 2- Birinci sayıyı gir. 3- İkinci sayıyı gir. 4- Sayıları çarp. 5- Sonucu ekrana yaz. 6- Bitir. </td> </tr> </table>	A 1- Başla 2- Kenar uzunluğunu giriniz. 3- Kenar uzunluğunu 4 ile çarp çevreyi bul 4- Toplam çevre uzunluğunu ekrana yaz. 5- Bitir.	B 1- Başla 2- Kısa kenar uzunluğunu giriniz. 3- Uzun kenar uzunluğunu giriniz. 4- Kısa kenar ve uzun kenarı topla. 5- Kenar toplamını 2 ile çarp. 6- Çevre uzunluğunu ekrana yaz. 7- Bitir.	C 1- Başla 2- Birinci kenar uzunluğunu gir 3- İkinci kenar uzunluğunu gir 4- Kenarları topla 5- Toplanan kenarları 2'ye böl 6- Sonucu ekrana yaz 7- Bitir.	D 1- Başla 2- Birinci sayıyı gir. 3- İkinci sayıyı gir. 4- Sayıları çarp. 5- Sonucu ekrana yaz. 6- Bitir.
A 1- Başla 2- Kenar uzunluğunu giriniz. 3- Kenar uzunluğunu 4 ile çarp çevreyi bul 4- Toplam çevre uzunluğunu ekrana yaz. 5- Bitir.	B 1- Başla 2- Kısa kenar uzunluğunu giriniz. 3- Uzun kenar uzunluğunu giriniz. 4- Kısa kenar ve uzun kenarı topla. 5- Kenar toplamını 2 ile çarp. 6- Çevre uzunluğunu ekrana yaz. 7- Bitir.	C 1- Başla 2- Birinci kenar uzunluğunu gir 3- İkinci kenar uzunluğunu gir 4- Kenarları topla 5- Toplanan kenarları 2'ye böl 6- Sonucu ekrana yaz 7- Bitir.	D 1- Başla 2- Birinci sayıyı gir. 3- İkinci sayıyı gir. 4- Sayıları çarp. 5- Sonucu ekrana yaz. 6- Bitir.	

EK F'nin devamı

	X	X		X			
X	X		X	X		X	
X	X	X		X	X	X	X
X							
X				○	X	X	X
X						X	X
X	X	X	○	X	X	X	X
X	X				X	X	X
○	○	○			X	X	
☘					X	X	X

Zıpla
4 adım ilerle
2 adım ilerle

Zıpla
2 adım ilerle
Sağa dön
Sola dön

Aşağıda bulmacalarda piyonun varış noktasına ulaştırmaya çalışacağız. Bunun için sırası karışık olarak verilmiş algoritmaları doğru bir şekilde sıralamanı ve bu sıralamayı bulmacaların altındaki boş sütuna yazman gerekiyor. Piyon tek seferde yalnızca bir çukurun üzerinden atlayabilir. X'ler aşlamayan engelleri gösterir.

☘ Piyon
○ Çukur
X Engel
☘ Varış Noktası

14- Yukarıdaki görselde kaplumbağanın başlangıçtan varışa kadar tüm adımları karmaşık verilmiştir. Buna göre kaplumbağanın adımlarının doğru düzenlendiği algoritma hangi seçenekte yer almaktadır?

A	B	C	D
4 adım ilerle Zıpla Sağa dön Zıpla 2 adım ilerle Sola dön 2 adım ilerle	4 adım ilerle Sağa dön 2 adım ilerle Zıpla 2 adım ilerle Zıpla Sola dön	4 Adım ilerle Sola dön 2 adım ilerle Zıpla Zıpla Sağa dön 2 adım ilerle	4 Adım ilerle Sola dön 2 adım ilerle Sağa dön 2 adım ilerle Zıpla Zıpla

15-Yandaki program kodlarında; eğer yaşı 18'den büyükse ehliyet alabileceğini, eğer 18 yaşından küçükse ehliyet alamayacağını söyleyen programın kodu yazılırken hangi kod yapısı kullanılmıştır?

The diagram shows four different Scratch code snippets for checking if a person is eligible to vote based on their age.

- A:** A script starting with a 'when green flag clicked' event, followed by a 'say Hello! for 2 sec.' speech bubble block, and then a 'say Hello! for 2 sec.' speech bubble block. The code is incomplete.
- B:** A script starting with a 'when green flag clicked' event, followed by a 'say Hello! for 2 sec.' speech bubble block, and then a 'say Hello! for 2 sec.' speech bubble block. The code is incomplete.
- C:** A script starting with a 'when green flag clicked' event, followed by an 'if age > 18 then' conditional block. Inside the 'if' block is a 'say Hello! for 2 sec.' speech bubble block. Outside the 'if' block is a 'say Hello! for 2 sec.' speech bubble block.
- D:** A script starting with a 'when green flag clicked' event, followed by an 'if age > 18 then' conditional block. Inside the 'if' block is a 'say Hello! for 2 sec.' speech bubble block. Outside the 'if' block is a 'say Hello! for 2 sec.' speech bubble block.

EK F'nin devamı

	16- Birsen evden çıkıp 500 adım ilerleyince evinin yakınındaki markete varmaktadır. Birsen markete giden yolu kodlara dönüştürmek istemektedir. Fakat kodlar arasında sadece 1 adım git () kodu bulunmaktadır. Birsen'in 500 adım gitmesi için 1 adım git () kodunu 500 kere kullanmak için Birsen'e gerekli kod aşağıdakilerden hangisidir? A) Karar kodları C) Değişken Kodları B) Döngü Kodları D) Algılama Kodları
--	---

17- Kodlama yazılımına eklenen oyun karakterinin 100 birim ilerlemesi için kod yazılıyor. Buna göre karakterin ilerlemesini sağlayan kod aşağıdakilerden hangisidir?			
A	B	C	D

	18- Yanda yer alan kuşun () hareket edip diğer karakteri () yemesi gerekmektedir. Buna göre kuşun amacına ulaşması için yazılan doğru kod hangisidir?		
A	B	C	D

EK F'nin devamı

Arının çiçeğe varıp çiçekteki bal nektarını alması gerektiren kodlar hatalı yazılmıştır.

19-Hata yapılan kod kaç numaralı adımdadır?

A) 1 B) 2 C) 3 D) 4

A noktasından B noktasına doğru yola çıkan birisi yuvarlak engellere takılmadan hareket ettiğinde en kısa yoldan B'ye varmaktadır. (Yuvarlak engelden geçemez)

20-Aşağıdakilerden hangisi en kestirme yoldur?

A) 3 Doğu- 1 Güney-2 Doğu- 3 Güney
B) 2 Doğu- 2 Güney- 3 Doğu- 1 Güney
C) 1 Doğu- 3 Güney- 4 Doğu
D) 5 Doğu- 3 Güney

Türkiye’de herkesin ilgisini çeken bir göl bulunmaktadır. Bu göle aileler piknik yapma ve yüzmeye amacıyla gitmektedir. Fakat herkesi üzen bir olay yaşanmıştır. Göl suyu renk değiştirmiş, kirlenmiş ve yüzemeye giren insanlar hasta olmuşlardır. Berk ve arkadaşları bu duruma çözüm geliştirmek amacıyla balık kılçığında problemin çözümüne yönelik analiz yapmışlardır. Su kirliliğini ölçen bir yazılım geliştirmişlerdir. Suyun kirliliğini ölçmeyi başarmışlardır.

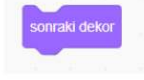
21- Yukarıda yer alan metne göre Berk ve arkadaşlarının geliştirdikleri yazılımla su kirliliğini ölçmesi problem çözme adımlarından hangisini kullandıklarını gösterir?

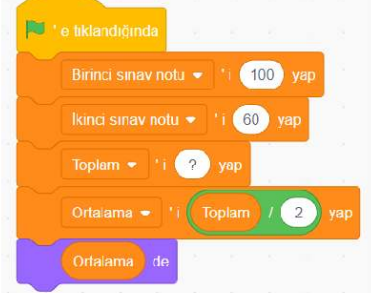
- A) Problemi anlama B) Plan Yapma C) Planı Uygulama D) Çözümü değerlendirme

22-Aşağıdaki algoritmada tekrar eden adımlar bir döngü içinde gösterilmek isteniyor. Buna göre tekrarlanan adımlar seçeneklerdeki döngülerden hangisinin içine yazılır?

1. İleri 2. Sağa dön 3. İleri 4. Sağa dön 5. İleri 6. Sağa dön 7. İleri 8. Sağa dön	A	B
	bu işlemleri 4	bu işlemleri 5
	C	D
	bu işlemleri 10	bu işlemleri 6

EK F'nin devamı

23- Programlama aracında altına yazılan kod bloklarını çalışmasını sağlayan başlangıç kodu aşağıdakilerden hangisidir?			
A	B	C	D
			

	24- Yanda verilen kodlar, sınav notları girilince iki sınav notunun ortalamasını bulan programdır. Program yazılırken bir adım eksik bırakılıyor. Buna göre, eksik bırakılan adıma yazılması gereken kod aşağıdakilerden hangisidir? A) Ortalama B) Toplam C) Sınav notu D) Birinci sınav notu + ikinci sınav notu
---	---

EK G Uygulamaya Ait Görseller

