

MİMAR SİNAN GÜZEL SANATLAR ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**ORTAOKUL ÖĞRENCİLERİNİN ÇOKLU ZEKÂ ALANLARI VE BLOK TABANLI
PROGRAMLAMAYA YÖNELİK ÖZ-YETERLİLİK ALGILARININ İNCELENMESİ**

YÜKSEK LİSANS TEZİ

Mediha Ayşen YÜKSEL

Enformatik Anabilim Dalı

Bilgisayar Ortamında Sanat ve Tasarım Programı

Tez Danışmanı: Dr. Öğr. Üyesi Nazım Ziya PERDAHÇI

MART 2022

MİMAR SİNAN GÜZEL SANATLAR ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**ORTAOKUL ÖĞRENCİLERİNİN ÇOKLU ZEKÂ ALANLARI VE BLOK TABANLI
PROGRAMLAMAYA YÖNELİK ÖZ-YETERLİLİK ALGILARININ İNCELENMESİ**

YÜKSEK LİSANS TEZİ

Mediha Aşen YÜKSEL

Enformatik Anabilim Dalı

Bilgisayar Ortamında Sanat ve Tasarım Programı

Tez Danışmanı: Dr. Öğr. Üyesi. Nazım Ziya PERDAHÇI

MART 2022

..... tarafından hazırlanan adlı bu
tezin tezi olarak uygun olduğunu onaylarım.

.....
Tez Yöneticisi

Bu çalışma, jürimiz tarafından Anabilim Dalında
..... tezi olarak kabul edilmiştir.

Başkan : _____

Üye : _____

Üye : _____

Üye : _____

Üye : _____

Bu tez, Mimar Sinan Güzel Sanatlar Üniversitesi Enstitüsü tez yazım
kurallarına uygundur.



Mimar Sinan Güzel Sanatlar Üniversitesi Fen Bilimleri Enstitüsü tez yazım kılavuzuna uygun olarak hazırladığım bu tez çalışmasında;

- tez içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel etik kurallarına uygun olarak sunduğumu,
- başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
- atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
- kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- ücret karşılığı başka kişilere yazdırmadığımı (dikte etme dışında), uygulamalarımı yaptırmadığımı,
- ve bu tezin herhangi bir bölümünü bu üniversite veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

beyan ederim.

Mediha Ayşen Yüksel



Aileme,

İÇİNDEKİLER

	Sayfa
ÖNSÖZ	viii
ÖZET	ix
ABSTRACT	x
ÇİZELGE LİSTESİ	xi
ŞEKİL LİSTESİ	xii
KISALTMALAR	xiii
1. GİRİŞ	1
1.1 Problem Tanımı ve Araştırmanın Amacı.....	2
1.2 Araştırma Soruları ve Hipotez.....	3
1.3 Araştırmanın Önemi.....	5
1.4 Varsayımlar.....	5
1.5 Sınırlılıklar.....	6
2. SAYISAL TEKNOLOJİLER VE PROGRAMLAMA	6
2.1 Bilgisayar bilimi, algoritma ve bilgi işlemsel düşünme	7
2.2 Programlama.....	9
2.3 21. yüzyılda programlama becerisi.....	12
2.4 Küçük yaş grubuna verilen programlama eğitimi.....	14
2.5 Scratch.....	16
3. ÖĞRENME VE ÇOKLU ZEKÂ TEORİSİ	19
3.1 Matematiksel/Mantıksal zekâ.....	21
3.2 Sözel/Dilsel zekâ.....	21
3.3 Görsel/Uzamsal zekâ.....	22
3.4 Duyumsal/Müziksel zekâ.....	22
3.5 İçsel/Öze dönük zekâ.....	22
3.6 Sosyal/Kişiler arası zekâ.....	23
3.7 Bedensel/Kinestetik zekâ.....	23
3.8 Doğasal zekâ.....	24
4. KURAMSAL ÇERÇEVE	24
4.1 Scratch ile İlgili Araştırmalar.....	24
4.2 Programlama Ve Düşünme Becerileri.....	28
4.3 Programlama ile Bilgi İşlemsel Düşünme.....	30

4.4 Programlama, Bilgisayara Yönelik Tutum Ve Zekâ Alanları.....	31
4.5 Zekâ Alanları.....	33
4.6 Programlama Eğitimindeki Sorunlar.....	34
4.7 Öz-Yeterlilik Algısı Ve Performans İlişkisi.....	35
4.8 Özgünlük Boyutu.....	36
5. ARAŞTIRMA YÖNTEMİ VE ÇALIŞMA PLANI.....	36
5.1 Araştırmanın Modeli.....	37
5.2 Çalışma Grubu.....	37
5.3 Çalışma Planı, Sınırlılık ve Varsayımlar.....	38
5.4 Veri Toplama Araçları.....	39
5.5 Verilerin Toplanması ve Çözümlemesi.....	40
6. BULGULAR.....	41
6.1 Blok Tabanlı Programlamayı Öğrenme Ortamları.....	41
6.2 Blok Temelli Programlamaya İlişkin Öz-Yeterlilik Algısı Ölçeği.....	41
6.3 Çoklu Zekâ Alanlarını Değerlendirme Ölçeği.....	42
6.4 Korelasyon Analizi.....	44
7. TARTIŞMA, SONUÇ VE ÖNERİLER.....	45
7.1 Tartışma ve Sonuç.....	45
7.2 Öneriler.....	49
8. KAYNAKLAR.....	51
9. EKLER.....	62
EK-A Blok temelli programlamaya ilişkin öz-yeterlilik algısı ölçeği.....	62
EK-B1 Matematiksel/Mantıksal zekâ alanı değerlendirme ölçeği.....	63
EK-B2 Sözel/Dilsel zekâ alanı değerlendirme ölçeği.....	64
EK-B3 Görsel/Uzamsal zekâ alanı değerlendirme ölçeği.....	65
EK-B4 Duyumsal/Müziksel zekâ alanı değerlendirme ölçeği.....	66
EK-B5 İçsel/Öze dönük zekâ alanı değerlendirme ölçeği.....	67
EK-B6 Sosyal/Kişiler arası zekâ alanı değerlendirme ölçeği.....	68
EK-B7 Bedensel/Kinestetik zekâ alanı değerlendirme ölçeği.....	69
EK-B8 Doğasal zekâ alanı değerlendirme ölçeği.....	70
EK-C Uygulama izin yazıları.....	71
10. ÖZGEÇMİŞ.....	73

ÖNSÖZ

Bu tezi yazmamda bana yardımcı olan danışmanım Dr. Öğr. Üyesi Nazım Ziya Perdahçı hocama, istatistiksel yöntemleri anlamamda bana destek olan Dr. Özlem Albayrakođlu hanıma, ihtiyacım olan verilere ulaşmamı kolaylaştıran ablalarım Selcen Yüksel Arvas, Gökçen Yüksel Karaca'ya ve paylaşımlarıyla araştırmanın yayılmasını sağlayan iyiliksever sosyal medya kullanıcılarına, yazım hatalarımı düzelten Türkçe aşığı sevgili babam Nebi Yüksel'e, beni büyüten sevgili annem Salise Yüksel'e, MSGSÜ Enformatik Bölümü hocalarına ve bölümü tanımama vesile olan teyzem Deniz Batukan'a ve son olarak, çalışmaya katılan tüm güzel çocuklara ve ilgili velilerine teşekkür ederim.

ORTAOKUL ÖĞRENCİLERİNİN ÇOKLU ZEKÂ ALANLARI VE BLOK TABANLI PROGRAMLAMAYA YÖNELİK ÖZ-YETERLİLİK ALGILARININ İNCELENMESİ

ÖZET

Çocuklara programlamayı öğretmek amacıyla pek çok bilgisayar programı ve robotlar eğitim pazarına sunulmuştur. Bu teknolojiler arasında Massachusetts Teknoloji Enstitüsü tarafından geliştirilen Scratch ile çocuklar blok tabanlı programlama dilini kullanarak bilgisayar programlama mantığını öğrenmektedirler.

Çoklu zekâ teorisi kişinin sahip olduğu bilişsel becerileri kategorilere ayırarak açıklamaya çalışmaktadır. Bu teoriye göre insanlar belli bilişsel becerilere yatkın olabilirler. Bu bilişsel beceriler 8 zekâ alanına ayrılmaktadır: matematiksel/mantıksal, sözel/dilsel, görsel/uzamsal, duyuşsal/müziksel, içsel/öze dönük, sosyal/kişiler arası, bedensel/kinestetik ve doğasal zekâ. Çoklu zekâ teorisine göre mantıksal zekâ alanı baskın olan kişilerin meslek seçimlerinde mantık ve matematik konularına ağırlıklı meslekler seçilmesi önerilir. Ör: muhasebeci, bilgisayar mühendisi, bilgisayar programcısı, yazılımcı vs. Bu teoriye göre çocukların zekâ alanlarındaki baskın alanın/alanların tespit edilmesi ile onların meslek seçimine erkenden yönlendirilmesi mümkün olabilir.

Bu tezde, “Çocukların çoklu zekâ alanları ile blok tabanlı programlama dili olan Scratch’e yatkınlıkları arasında anlamlı bir ilişki var mıdır?” araştırma sorusuna cevap aranmıştır. Öğrencilerin çoklu zekâ alanlarından bazıları ile blok tabanlı programlamaya yönelik öz-yeterlilik algıları arasında anlamlı bir ilişki olduğu hipotez olarak öne sürülmüştür. Bunun için, blok tabanlı Scratch platformunda programlama etkinlikleri yapan ortaokul düzeyindeki öğrencilerin blok tabanlı programlamaya ilişkin öz-yeterlilik algıları ve bu öğrencilerin çoklu zekâ alanları arasında anlamlı bir ilişki olup olmadığının tespit edilmesi amaçlanmıştır. Araştırma grubunu 45 ortaokul öğrencisi oluşturmaktadır. Çalışmada nicel araştırma yöntemleri kullanılmıştır. Araştırmada iki ölçek kullanılmıştır. Ölçeklerden biri “Çoklu Zekâ Kuramı Değerlendirme Ölçeği”, diğeri ise “Blok Temelli Programlamaya İlişkin Öz-yeterlilik Algısı Ölçeği”dir. Araştırma sonucunda, ortaokul öğrencilerinin blok tabanlı programlamaya yönelik öz-yeterlilik algıları ile çoklu zekâ alanlarının bazıları ile pozitif yönde, anlamlı ilişkiler tespit edilmiştir.

Anahtar kelimeler: Çoklu zekâ, Scratch, Matematiksel düşünme, Kodlama, Bilgi işlemsel düşünme, Programlama

INVESTIGATION OF SECONDARY SCHOOL STUDENTS' MULTIPLE INTELLIGENCES AND PERCEIVED SELF-EFFICIENCY RELATED TO BLOCK-BASED PROGRAMMING

ABSTRACT

Many computer programs and robots have been presented to the education market in order to teach programming to children. Among these technologies, children learn computer programming logic by using block-based programming language with Scratch developed by Massachusetts Institute of Technology.

The Gardner's theory of multiple intelligences tries to explain the cognitive skills of a person. According to this theory, people may be predisposed to certain cognitive abilities. These cognitive abilities are divided into 8 intelligence types: mathematical/logical, verbal/linguistic, visual/spatial, sensory/musical, intrapersonal, social/interpersonal, bodily/kinesthetic, and natural intelligence. According to the theory of multiple intelligences, it is recommended that people with a dominant logical intelligence type choose professions that are focused on logic and mathematics. According to this theory, it may be possible to guide pupils to the choice of profession early by identifying the dominant type(s) in their intelligence types.

Research question of the thesis was: "Is there a significant relationship between pupil's multiple intelligence types and their predisposition to Scratch, which is a block-based programming language?". It has been hypothesized that there is a significant relationship between some of the students' multiple intelligence types and their self-efficacy perceptions towards block-based programming. For this purpose, it is aimed to determine the self-efficacy perceptions of the students at the secondary school level who do programming activities on the block-based Scratch platform and the multiple intelligence types of these students, and to determine whether there is a significant relationship between the intelligence types. The research group consists of 45 secondary school students. Quantitative research methods were used in the study. Two scales were used in the study. One of the scales is the "Multiple Intelligence Theory Evaluation Scale" and the other is the "Self-efficacy Perception Scale for Block-Based Programming". As a result of the research, positive and significant correlations were found between the self-efficacy perceptions of middle school students towards block-based programming and some of the types intelligence.

Keywords: Multiple intelligences, Scratch, Mathematical thinking, Coding, Computational thinking, Programming

ÇİZELGE LİSTESİ

Sayfa

Çizelge 2.1 : Çoklu zekâ alanları.....	20
Çizelge 6.1 : Scratch'i öğrenme ortamları.....	41
Çizelge 6.2 : BPÖA ortalaması.....	42
Çizelge 6.3 : BPÖA normallik testi.....	42
Çizelge 6.4 : BPÖA basıklık ve çarpıklık değerleri.....	42
Çizelge 6.5 : Çoklu zekâ alanları ortalaması.....	43
Çizelge 6.6 : Çoklu zekâ alanları normallik testi.....	43
Çizelge 6.7 : Çoklu zekâ alanları basıklık ve çarpıklık değerleri.....	44
Çizelge 6.8 : BPÖA ve Çoklu Zekâ Alanları Korelasyon Analizi.....	44
Çizelge A : Blok temelli programlamaya ilişkin öz-yeterlik algısı ölçeği.....	59
Çizelge B.1 : Mantıksal/Matematiksel Zekâ Alanı Değerlendirme Ölçeği.....	62
Çizelge B.2 : Sözel/Dilsel Zekâ Alanı Değerlendirme Ölçeği.....	63
Çizelge B.3 : Görsel/Uzamsal Zekâ Alanı Değerlendirme Ölçeği.....	64
Çizelge B.4 : Duyumsal/Müziksel Zekâ Alanı Değerlendirme Ölçeği.....	65
Çizelge B.5 : İçsel/Öze Dönük Zekâ Alanı Değerlendirme Ölçeği.....	66
Çizelge B.6 : Sosyal/Kişiler Arası Zekâ Alanı Değerlendirme Ölçeği.....	67
Çizelge B.7 : Bedensel/Kinestetik Zekâ Alanı Değerlendirme Ölçeği.....	68
Çizelge B.8 : Doğasal Zekâ Alanı Değerlendirme Ölçeği.....	69

ŞEKİL LİSTESİ

	<u>Sayfa</u>
Şekil 2.1 : Matematiksel ve bilgi işlemsel düşünme kümeleri.....	9
Şekil 2.2 : Blok tabanlı kodlama örneği.....	11
Şekil 2.3 : Metin tabanlı kodlama örneği.....	11
Şekil 2.4 : Scratch'te kod ekleme.....	17
Şekil 2.5 : Scratch'te koşul kodu ekleme.....	17
Şekil 2.6 : Frekansa göre öğretilen bilgisayar dilleri.....	19

KISALTMALAR

	<u>Sayfa</u>
ISTE: International Society for Technology in Education (Uluslararası eğitimde teknoloji topluluğu).....	13
IQ : Intelligence quotient (zekâ katsayısı).....	19
BPÖA : Blok temelli programlamaya ilişkin öz-yeterlik algısı.....	41

1. GİRİŞ

Teknolojinin ilerlemesiyle, iş dallarında bilgisayar mühendisliği ve programlama alanına olan ilgi ve talep büyümüştür. Ünlü bilgisayar bilimci, politikacı ve silikon vadisi girişimcilerinin yönlendirmeleri üzerine pek çok ebeveyn çocuklarının programlamayı erken yaşta öğrenmeleri için talepte bulunmuştur. Bu talep doğrultusunda çocuklara programlamayı öğretmek amacıyla pek çok bilgisayar programı ve robotlar eğitim pazarına sunulmuştur. Bu teknolojiler arasında Massachusetts Teknoloji Enstitüsü tarafından geliştirilen Scratch ile ilkokul, ortaokul ve lise düzeyindeki öğrenciler blok tabanlı programlama dilini kullanarak bilgisayar programlama mantığını öğrenmektedirler. Scratch ile blok tabanlı programlama dersleri sayesinde öğrenciler erken yaşta bu alanı keşfetmiş ve aşama kaydetmiş olacaklardır. Ancak öğrencilerin bir kısmının programlama derslerinde zorlandıkları gözlemlenmiştir. Öğrencilerin programlama performanslarındaki bu farklılıklar, çoklu zekâ teorisine göre açıklanabilir mi?

Çoklu zekâ teorisi, her bireyin farklı bilişsel becerilere sahip olduğunu açıklamaktadır. Bu sebeple, öğretim programlarının herkes için geçerli ve standart olması yerine, eğitim müfredatı öğrencilerin çeşitliliği ve esnekliğine uyum sağlayabilmelidir. Birbirinden farklı çeşitli bilişsel becerilere sahip öğrencilerin çalışma ve ilgi alanlarına yatkınlıkları, alacakları eğitimler öncesinde tespit edilerek, öğrencilerin bu alanlarla ilişkili derslere yönlendirilmesi çoklu zekâ alanlarının tespiti ile mümkün olabilir. Örnek olarak, matematiksel/mantıksal zekâsı yüksek olan bireyler, bilgisayar programcılığı, mühendislik ve muhasebe gibi hesaplamalı ve analitik düşünceye yönelik alanlarda meslek edinebilirler. Bu teori, blok tabanlı programlama becerisinin çoklu zekâ teorisi ile ilişkili olup olmadığı araştırılarak test edilecektir.

Bu araştırma, ortaokul öğrencilerinin Scratch ile blok tabanlı programlama yönelik öz-yeterlilik algıları ve çoklu zekâ alanları ile basit korelasyon analizi ile, çoklu zekâ alanlarının blok tabanlı programlamayla ilişkili olup olmadığını, ilişkili ise hangi alanların ilişkili olduğunu incelemeye yöneliktir. Böylelikle, gelecekte bilgisayar teknolojisinde yenilik üretecek başarılı bireylerin erken yaşta fark edilmesi ve bu alana yönlendirilmesi mümkün olabilir.

1.1 Problem Tanımı ve Araştırmanın Amacı

Bilgisayar bilimi ve kullanılan teknolojiler; günlük yaşamımızın, ekonominin ve bilimsel gelişmenin kalbinde yer almaktadır (Wilson ve diğ., 2010). 21. yüzyılın işgücü ihtiyacı olan bilgisayar teknolojisini üretecek başarılı bireylerin ihtiyaç duyacakları becerileri kazandırmak amacıyla ve bilgisayar biliminin temelleri hakkında daha derin bir anlayışa sahip olmaları için günümüzde çocuklara yönelik programlama eğitimleri yaygınlaşmaktadır (Ceylan, 2020; Totan, 2021; Wilson ve diğ., 2010). Bu eğitimler genel olarak ilkokul, ortaokul ve lise düzeyindeki öğrencilere Scratch gibi blok tabanlı programlama dillerinin öğretilmesine yöneliktir.

Çocukların gelecekteki mesleklerine daha iyi yönlendirilmesi amacıyla başlatılan bu programlama eğitimlerinin her çocuk için gerekli olup olmadığı tartışılmaktadır. Çünkü, programlama dersleri her çocuğu bir programcıya dönüştürmeyecektir (Hinsliff, 2015). Her insanın tabiatı ve bilişsel becerileri farklıdır. Öğretim programlarının herkes için geçerli ve standart olması insan tabiatına aykırıdır. Bu yüzden öğretim müfredatı, öğrencilerin çeşitliliği ve esnekliğine uyum sağlamalıdır (Millî Eğitim Bakanlığı, 2018). Birbirinden farklı çeşitli bilişsel becerilere sahip öğrencilerin bu becerileri, alacakları eğitimler öncesinde tespit edilerek, öğrencilerin ilgili alan ve derslere yönlendirilmesi mümkün olabilir. Bunun için, bilgisayar teknolojilerini geliştirmede başarılı olacak bireylerin tespiti, hangi çocuğun programlama eğitimi alması gerektiği konusu eğitim bilimciler için bir araştırma ve ebeveynler için de merak konusudur.

Programlama konusunda becerikli olanlar, programlama dersleri ile erken yaşta bu alanı keşfetmiş ve aşama kaydetmiş olacaktırlar. Programlamada daha az becerikli olanlar ise en azından içinde yaşadıkları sayısal dünyayı daha rahat anlayacaklardır (Hinsliff, 2015). Ancak öğrencilerin bir kısmının programlama derslerinde zorlandıkları gözlemlenmiştir (Papatğa, 2016; Vatansever, 2018; Bala ve Alacapınar, 2021; Çubukluöz, 2019). Programlamayı öğrenmek, her öğrenci için benzersiz bir deneyimdir ve bir öğrencinin neden diğerinden daha iyi ve daha hızlı programlamayı öğrendiği tam olarak anlaşılmamıştır (Ramalingam ve diğ., 2004).

Bu tezin amacı, blok tabanlı programlama becerisinin çoklu zekâ teorisi ile ilişkili olup olmadığını araştırarak, blok tabanlı programlama konusunda başarılı bireyleri tespit etmeye çalışmaktır. Çünkü, çoklu zekâ teorisi, matematiksel/mantıksal zekâsı yüksek olan bireylerin bilgisayar programcısı gibi hesaplamalı ve analitik düşünceye yönelik alanlarda meslek edinebileceklerini belirtmektedir (Babacan ve Dilci, 2012). Buna göre, baskın zekâ alanları ile blok tabanlı programlama becerisi arasında anlamlı bir korelasyon aranabilir. Bunun için ise Scratch ile programlama yapan ortaokul öğrencilerinin, blok tabanlı programlamaya yönelik öz-yeterlilik algıları ile çoklu zekâ alanlarıyla ilişkisi araştırılmıştır. Bu ilişki, basit korelasyon analizi yöntemiyle araştırılmış olup araştırmanın çalışma grubunu, 2021-2022 öğretim yılında Türkiye’de öğrenim gören ve Scratch ile programlama yapabilen 45 ortaokul öğrencisi oluşturmuştur. Bu öğrencilerin velilerine çevrim içi platformlar üzerinden ulaşarak öğrenciler çalışmaya davet edilmiştir.

1.2 Araştırma Soruları ve Hipotez

Çocuklara yönelik blok tabanlı programlama eğitiminde, bazı çocukların Scratch etkinliklerinde daha fazla verim aldığı, konuyu daha hızlı kavradıkları ve çok rahat bir şekilde çözüme ulaştıkları gözlenmiştir. Bazı çocukların ise Scratch’te kod mantığını oluştururken veya hata ayıklarken zorlandıkları ortaya çıkmıştır (Papatğa, 2016; Vatansever, 2018; Bala ve Alacapınar, 2021; 2019; Çubukluöz, 2019). Bu gözlemler, çocukların blok tabanlı programlamada öz-yeterlilik algılarının çoklu zekâ alanlarıyla ilişkili olup olmadığı sorusunu ortaya çıkarmıştır. Çocuklarda Scratch etkinliklerinde oluşan bu farklılıklar, farklı zekâ alanlarının baskınlığı ile ilgili midir?

Çoklu zekâ teorisine göre bireylerde baskın olan zekâ alanlarının bilinmesi, bireylerin meslek seçimlerinde doğru tercih yapmasına faydalı olabilir. Örnek olarak bu teoriye göre, mantıksal/matematikselsel zekâ alanı baskın bireylerin bilgisayar programcısı vb. gibi meslek seçimleri onlar için doğru olabilir (Babacan ve Dilci, 2012). Buna göre, ortaokul öğrencilerinin zekâ alanları ve blok tabanlı programlamaya yönelik öz-yeterlilik algıları arasında anlamlı bir ilişki olabilir mi? Baskın zekâ tiplerinin önceden belirlenmesi, programlama eğitimi almış veya alacak çocuklar için ön kılavuzluk eder

mi? Bu çocukların erken yaşta meslek seçimine yönlendirilmesi mümkün müdür? Bu sorular araştırmanın motivasyonunu oluşturmaktadır.

Tüm bu soruları temellendirdiğimizde, çocuklardaki blok tabanlı programlama becerilerinin çoklu zekâ alanlarıyla ilişkilendirip ilişkilendirilemeyeceği, ilişkili zekâ alanlarının hangi oranlarda ilişkili olduğu araştırmanın zeminini oluşturmaktadır.

Araştırma sorusu: Çoklu zekâ alanları ve blok tabanlı programlama becerisi arasındaki ilişki nedir?

Teorik örtüşmenin ampirik bir ilişkide gösterilmesi beklenmektedir. Bu nedenle, aşağıdaki hipotez önerilmektedir.

Hipotez: Çoklu zekâ alanlarından bazıları ile blok tabanlı programlamaya yönelik öz-yeterlilik algısı arasında anlamlı bir ilişki vardır.

Alt araştırma sorusu 1: Matematiği seven ve matematikte başarılı olan çocuklar, bu zekâ alanı baskın ise, mantıkla ilişkili konular olan programlama ve algoritmayı da başarılı olarak anlayabilir ve rahatlıkla blok tabanlı programlama yapabilir mi?

Alt hipotez 1: Mantıksal/matematikselsel zekâ alanı ile blok tabanlı programlama becerisi arasında pozitif yönde anlamlı bir ilişki vardır.

Alt araştırma sorusu 2: Scratch'te programlama yaparken, uzun süre bilgisayar karşısında fazla bedensel hareket olmadan oturulduğu için, bazı çocukların kinestetik/bedensel zekâları baskın ise onların Scratch'ten sıkılmalarını beklemeli miyiz?

Alt hipotez 2: Bedensel/kinestetik zekâ alanı ile blok tabanlı programlamaya yönelik öz-yeterlilik algısı arasında negatif yönde ilişki vardır.

Alt araştırma sorusu 3: Scratch blok tabanlı görsel bir programlama dilidir. Buna göre Scratch'te başarılı öğrencilerin çoklu zekâ kuramına göre görsel/uzamsal alanları baskın mıdır?

Alt hipotez 3: Görsel/uzamsal zekâ alanı ile blok tabanlı programlamaya yönelik öz-yeterlilik algısı arasında pozitif yönde ilişki vardır.

1.3 Araştırmanın Önemi

Çocuklara yönelik programlama dersleri, dünya çapında, 21. yüzyıl becerileri olarak öngörülen programlama becerilerini erken yaşta edinmeleri için hükümetler, özel şirketler, silikon vadisi girişimleri ve bilimsel kuruluşların atılımlarıyla gerçekleştirilmektedir (Tsarava ve diğ., 2017). Tüm dünyada, programlama ve bilgi teknolojileri becerileri genç yaştan itibaren öğrencilere tanıtılmaya başlanmıştır. Bunun için Türkiye Millî Eğitim Bakanlığı, bilişim teknolojileri ve yazılım adıyla ortaokul öğrencilerine zorunlu ders olarak Scratch ile programlama derslerini müfredata eklemiştir (Şoltan, 2018). Ancak öğrencilerin hepsinin bu dersten istenilen verimi alamadıkları gözlemlenmiştir. Bazı öğrenciler, Scratch'te programlama yapmayı sevseler bile, programlama yaparken zorlanmaktadırlar (Papatğa, 2016; Vatansever, 2018; Bala ve Alacapınar, 2021; Çubukluöz, 2019). Bazı öğrencilerin Scratch'i öğrendikten sonra ileri dönemlerde Scratch'i kullanmaya devam edip bazılarının ise devam etmedikleri gözlemlenmiştir. Bu durumda, bu dersin her çocuk için zorunlu tutulmaması ya da çocukların potansiyellerine göre seçmeli dersler olarak alma imkanının sağlanması için bu araştırmanın ışık olabileceği düşünülmektedir. Ayrıca, okullarda ve özel kurslarda programlama eğitimlerine katılacak öğrencilere ön kılavuzluk etmek, aileyi öğrencinin geleceği hakkında doğru yönlendirmek için önemli olduğu düşünülmektedir.

1.4 Varsayımlar

- Çalışmaya katılan öğrencilerin öz-değerlendirmelerini dürüstlikle yaptıkları ve kullanılan veri toplama aracındaki soruları kendileri için doğru olarak ve tarafsız bir şekilde cevapladıkları kabul edilmiştir.
- Araştırmanın verilerinin toplandığı bütün aşamalarda güvenilir ölçümler gerçekleştirilmiştir.
- Araştırmaya katılan öğrencilerin Scratch'te programlama yapmayı bildikleri varsayılmıştır.

- Örneklem büyüklüğünün korelasyon analizi için yeterli olduğu ve kullanılan ölçme yöntemlerinin araştırmaya uygun olduğu uzman görüşü alınarak kabul edilmiştir.
- Araştırmaya katılan öğrencilerin öz-yeterlilik algılarının, öğrencilerin gerçek performanslarıyla aynı olduğu kabul edilmiştir.
- Araştırmada kullanılan envanterlerin ortaokul öğrencileri üzerinde uygulanabilir olduğu uzman görüşü alınarak kabul edilmiştir.

1.5 Sınırlılıklar

- Araştırmanın çalışma grubu, 2021-2022 öğretim yılında Türkiye’de öğrenim gören ve Scratch ile programlama yapabilen 45 ortaokul öğrencisi ile sınırlıdır.
- Araştırma için toplanan veriler 2021-2022 eğitim-öğretim yılı güz dönemi ile sınırlıdır.
- Araştırma grubu, sadece internet üzerinden ulaşılan katılımcılarla sınırlıdır.
- Araştırma, veri toplama araçları ile toplanan verilerle sınırlıdır.
- Araştırmanın değişkenleri, “Blok tabanlı programlamaya yönelik öz-yeterlilik algısı” ve “Çoklu zekâ alanları değerlendirme ölçeği” envanterlerindeki değişkenler ile sınırlıdır.

2. SAYISAL TEKNOLOJİLER VE PROGRAMLAMA

Teknolojideki ilerlemelerle birlikte, bilgisayara olan ilgi çığır açacak şekilde artmıştır (Kong ve diğ., 2019). Modern teknolojinin neredeyse her alanında kullanılan bilgisayarlar, iş gücünü ve ekonomiyi de derinden etkilemektedir. ABD Çalışma İstatistikleri Bürosu, bilgisayar sektörünün önümüzdeki 10 yıl içinde 1,5 milyon iş ilanına sahip olacağını tahmin ederek, bunu en hızlı büyüyen ekonomik alanlardan biri haline geleceğini bildirmiştir (Wilson ve diğ., 2010). Bu işlere giden birçok yol olsa da bu alandaki en hızlı büyüyen meslekler, yazılım ve bilgisayar mühendisliğidir. Ayrıca gelecekte, bilgisayar bilimi tarafından verilecek bilgi ve becerilerden büyük ölçüde yararlanılacağı belirtilmiştir (Wilson ve diğ., 2010). Ayrıca, hayat yazılımlar tarafından

yönetilmektedir. Akıllı cihazlar, evler, arabalar, hastaneler, okullar, iletişim araçları, bankalar ve kamusal ihtiyaçlar yazılımlara bel bağlamıştır (Şoltan, 2018).

2.1 Bilgisayar Bilimi, Algoritma ve Bilgi İşlemsel Düşünme

Bilgisayar bilimi, ilkeleri, donanım ve yazılım tasarımları, uygulamaları ve toplum üzerindeki etkileri dahil olmak üzere bilgisayarlar ve algoritmik süreçlerin incelenmesidir (Tucker ve diğ., 2006). Başka bir tanımda ise, bilgiyi tanımlayan ve dönüştüren algoritmik süreçlerin sistematik çalışması: Teorisi, analizi, tasarımı, verimlilik hesabı, gerçekleştirilmesi ve uygulanmasıdır. Bilgiyi işleme mantığının altında yatan temel soru şudur: “Neler verimli bir şekilde otomatikleştirilebilir?” (Denning, ve diğ., 1988). Newell ve diğ. (1967) ise bilgisayar bilimini, bilgisayarların ve onları çevreleyen başlıca fenomenlerin incelenmesi olarak açıklamıştır. İlk bilgisayar bilimleri lisans programı 1953'te Cambridge Üniversitesi'nde başlamış, ABD'deki ilk program ise 1962'de Purdue Üniversitesi'nde açılmıştır. Ancak, bilgisayar bilimlerine yönelik eğitimlerin, üniversite öncesi eğitim seviyesine geçişi oldukça yavaş olmuştur (Kong ve diğ., 2019).

Algoritma, bir problem veya hedefte; çözüme, sonuca ulaşmak için yapılacak işlerin adım adım hesaplanması ve sıralanmasıdır. Başka bir deyişle, bir algoritma, adım adım, belirli bir sırayla yürütülen, bir görevi çözmek için bir modeli temsil eden bir dizi talimattır (Varela ve diğ., 2019).

Algoritma baştan sonra doğru, karşılaşılabilecek problemlerin teker teker ve sırasıyla ele alınmasıyla oluşturulur. Algoritma günlük hayatta da insanın düşünce sisteminde de bulunur (Sanz, 2015). Buna en basit örnek olarak yemek yapmak gösterilir. Bir yemekteki malzemeler girdi, talimatlar sonucu ortaya çıkan ürün ise çıktı olarak açıklanabilir. Talimat algoritmadır, kullanılacak aletler ise donanımdır. Bir yemeğin yapılışında hedef bellidir, sonuca ulaşmak için adım adım talimatlar izlenmek zorunludur. Aşağıdaki dört madde bir algoritmanın sahip olması gereken özelliklerdir (Shyamasundar, 1996):

1. Her bir talimat, eldeki donanımsal yeterliliğe uygun olmalıdır. Algoritmik adımları tanımlamak için kullanılan tüm temel işlemler, sınırlı bir süre içinde, sezgi olmadan, mekanik gerçekleştirilebilmelidir.

2. Her adım açık bir şekilde tanımlanmalıdır.

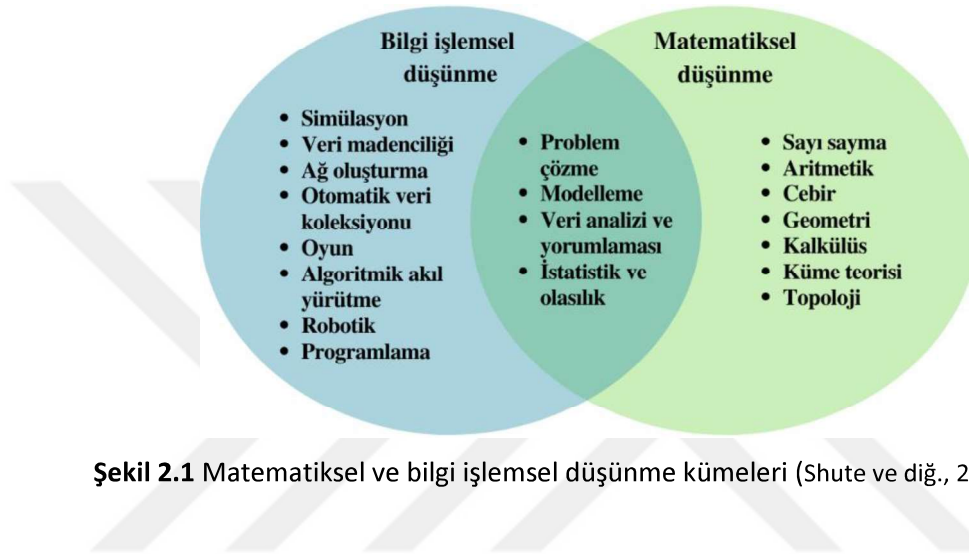
3. Talimattaki dil, çeşitli girdilerde çalışabilen farklı türde algoritmaları tanımlamak için yeterince genel olmalıdır.

4. Talimatlar bir sonuç elde etmek üzere hazırlanır. Girdiyi ve kesin çıktı sağlayan adımları tanımlayabilmelidir.

Algoritma bilgisayar biliminin temelini oluşturur. (Shyamasundar, 1996). Algoritmik düşünme ise algoritmaları anlama, yürütme, değerlendirme ve oluşturma becerisidir (Varela ve diğ., 2019). Genellikle, algoritmik düşünmeye mantıksal/matematikselsel düşünme de denir. Gardner'ın çoklu zekâ üzerine yaptığı çalışmalarda, mantıksal/matematikselsel düşünme; örüntüleri tanıma, tündengelimli akıl yürütme ve mantıkla düşünme becerileri ile ilişkilendirilir (Varela ve diğ., 2019). Algoritma; bilgi işlemsel düşünme, mantıksal/matematikselsel düşünme ile yakından ilişkilidir, ancak tam olarak aynı değildir. Her iki durumda da soyutlama kapasitesine büyük önem verilmesine rağmen, matematikselsel düşünme durumunda bu soyutlama kullanılan yapılarla (denklemler) ilişkilidir (Varela ve diğ., 2019).

Bilgi işlemsel (computational) -hesaplamalı, kompütasyonel ya da bilgisayarca düşünme becerisi, bilgisayarın adım adım uygulayacağı algoritmayı belirlemeyi sağlar. Bilgi işlemsel düşünme, bilgisayar bilimlerindeki kavramları kullanarak problem çözme becerisi olarak tanımlanabilir. Bununla birlikte bilgi işlemsel düşünme, karmaşık problemleri anlama ve çözme konusundaki genel beceri olarak görülmektedir (Boom ve diğ., 2018; Üzümcü ve Bay, 2019). Araştırmacılar, alan yazına baktığında, algoritmik düşünme becerisinin, bilgi işlemsel düşünme becerisini kapsadığını belirtmiştir (Adsay ve diğ., 2020). Bilgi işlemsel düşünme, bilgisayar bilimi için temel kavramlardan yararlanarak problem çözmeyi, sistemler tasarlamayı ve insan davranışını anlamayı içerir. Bilgi işlemsel düşünme, bilgisayar bilimi alanı kadar geniş ve zengin zihinsel becerileri içerir (Wing, 2006). Bilgi işlemsel düşünme şekli, bir problemi çözme yoluna gidebileceğimiz analitik düşünceyle, büyük ve karmaşık bir

sistemi anlamamızı, tasarlamamızı ya da değerlendirmemizi sağlayan bilimsel düşünce ile örtüşmektedir. Bilgi işlemsel düşünme becerisi ile problemleri çözmeye yönelik algoritmalar oluşturulur. Bir problem çözümü için algoritma mantığına dayalı olarak, problem önce ayrıntılı olarak analiz edilir, daha sonra birden fazla alt problemlere parçalanır, daha sonra bu alt problemler teker teker ele alınarak çözümlenir. Böylece problem çözüme ulaşmış olur (Şoltan, 2018; Sanz, 2015).



Şekil 2.1 Matematiksel ve bilgi işlemsel düşünme kümeleri (Shute ve diğ., 2017, s.146)

Matematik ve bilgisayar bilimi disiplinleri de yakından ilişkilidir. Mevcut kolej ve üniversitelerin bilgisayar bilimleri bölümlerinin çoğu, matematik bölümlerinden ayrılan öğretim üyeleri grupları tarafından oluşturulmuştur. Şimdi bile, matematik ve bilgisayar bilimlerini tek bir bölümde birleştirmek nadir değildir. Bilgi işlemsel düşünme her zaman matematik ve matematik eğitiminin bir parçası olmuştur (Moursund, 2006). Buna ek olarak, Tiflis (2018)'e göre programlama dersinin temelini oluşturan matematik dersi programlama öğrenimi için elzemdir. Programlama eğitimi konusunda başarısını kanıtlamış ülkeler göz önüne alındığında, bu ülkelerde matematik eğitim başarısı da yüksek düzeyde olduğundan, matematiğin programlama eğitimindeki önemi anlaşılmaktadır.

2.2 Programlama

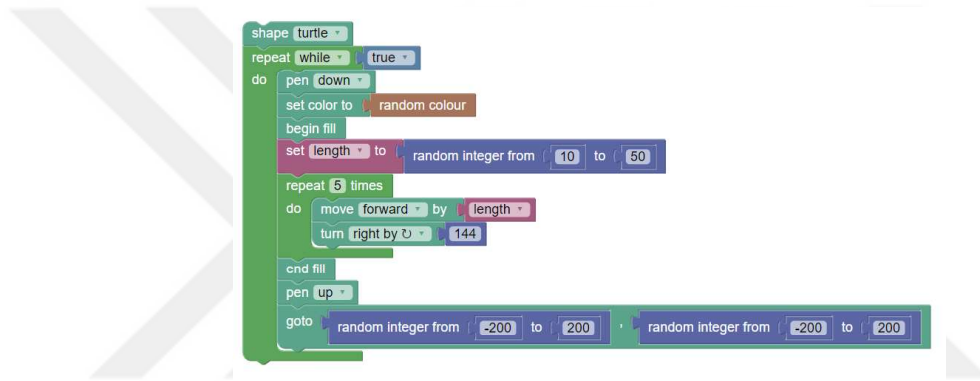
Programlama veya diğer bir adıyla kodlama becerileri, yazılım programlarını işlevsel bilgisayar uygulamaları olarak yazma ve tasarlamının pratik becerisidir (Tsarava ve

diğ., 2017). Bir diğeri tanıma göre programlama, gerçek hayattaki durumların modellenmesidir. Bu modellerin temel alınarak bilgisayar ortamına aktarılması işlemidir (Totan, 2018). Ayrıca, bilgi teknolojileri, sistematik ve hesaplama odaklı prosedürlere dayalı daha genel bilişsel problem çözme becerilerini ifade eder (Tsarava ve diğ., 2017). Programlama, kesin olarak hangi komutların (kodların) girileceğinin önceden hesaplanmasını, bu komutların sonuçlarının öngörülebilmesi için de analitik düşünme becerisini gerektirir. Problem çözme becerileri, planlama, adım adım mantığıyla yöntemlerin organize edilmesi, girdi ve çıktıların ve problemi çözmek için gereken aşamaların analizi ve tüm tasarımlar, programlama öncesinde belirlenmeli ve uygulanmalıdır (Çiftçi ve Bildiren, 2019). Programlama sadece kod yazmaktan ibaret değildir, programlama dilinden bağımsız olarak, algoritma oluşturmak, mantıksal düşünme ve problem çözme stratejilerini de içerir (Tekerek ve Altan, 2014).

Programlama, bilgisayarı komuta etmek için yazılan bir dizi yönerge (Sağay, 2019). İnsan ve bilgisayar arası etkileşim kodlar aracılığı ile sağlanır (Şoltan, 2018). Programlama yaparken ara yüz olarak iki tür yöntem; metin tabanlı ve blok tabanlı programlama dilleri, araçları ve düzenleyicileri kullanılmaktadır (Totan, 2021). Metin olarak yazılan kodlar yeni bir konuşma dili öğrenmek gibi, sözdizimi yazım kurallarını öğrenmeyi kapsamaktadır. Harf ve işaretlemeler kullanılan metin tabanlı programlama, öğrenilmesi zor bir beceri olarak bilinir. Metin tabanlı programlamaya yönelik olumsuz bakış açısı, öğrencilerde programlamaya yönelik öz-yeterliliklerinin düşük olmasına sebep olmakta, verilen görevi tamamlamakta güçlük çekmektedirler. (Mohammad ve diğ., 2018; Çiftçi ve Bildiren, 2019). Buna ek olarak, problem çözme, plan geliştirme ve programlama dillerini etkili kullanma konusunda öğrenciler sıkıntı yaşamaktadırlar (Çiftçi ve Bildiren, 2019). Bu yüzden programlama öğrenme yöntemini daha ilginç ve kolay hale getirmek önemlidir (Muhammad ve diğ., 2018). Ayrıca, programlamanın sadece profesyoneller tarafından değil herkes tarafından öğrenilmesi için çalışmalar yapılmaktadır (Tekerek ve Altan, 2014). Bu sebeple, program yazımını kolaylaştırmak için blok tabanlı görsel programlama dilleri ortaya çıkmıştır. Blok tabanlı programlamanın iki temel amacı vardır: Programlama sözdizimini basitleştirmek ve programlamaya ilgi duyan öğrenci sayısını artırmak için

hedef öğrencilerin dikkatini programlamaya çekmektir (Yıldız ve diğ., 2020). Blok tabanlı programlamada kodlar hazır yazılmış olarak bulunmaktadır. Tek yapılması gereken istenilen programa göre blokların alt alta sıralanmasıdır. Bloklar, lego oyuncakları gibi birbirine geçmeli tarzdadır, gerekli boşluklara başka bloklar sürüklenip bırak tarzında eklenebilmektedir. Blok tabanlı programlama, metin tabanlı programlamaya göre daha kolay anlaşılabilir ve uygulanmaktadır (Totan, 2021). Ancak, blok tabanlı programlama dilleri, profesyonel ve mesleki olarak, program ve yazılım geliştirmede kullanılmamaktadır (Robinson, 2016).

Örnek olarak rastgele yıldızlar oluşturmak için bloklar ve metin ile birebir olarak aynı yazılmış kodlar incelenebilir.



Şekil 2.2 Blok tabanlı kodlama örneği

```
import turtle
import random

turtle.shape("turtle")
while True:
    turtle.pendown()
    turtle.color('#%06x' % random.randint(0, 2**24 - 1))
    turtle.begin_fill()
    length = random.randint(10, 50)
    for count in range(5):
        turtle.forward(length)
        turtle.right(144)
    turtle.end_fill()
    turtle.penup()
    turtle.goto(random.randint(-200, 200), random.randint(-200, 200))
```

Şekil 2.3 Metin tabanlı kodlama örneği

Bunları bir araya getirdiğimizde, bir bilgisayarı komuta etmek için bilgi işlemsel düşünme kullanarak sorunları ayrıntılı analiz etmeli, nasıl parçalanacağını planlamalı

ve bilgisayar dostu çözümler aramalıdır. Bu inceleme genellikle diyagramlarla ifade edilen süreçleri ve verileri içerir (Shyamasundar, 1996). Neyin alakalı olduğunu, alt sistemlerin nasıl bağlantılı olduğunu ve hangi bilgi akışlarının dahil edildiğini bilmek için önemli bir soyutlama düzeyi göstermek gerektiğinden, bu tür adımlar zihni gerçekten zor ve farklı şekillerde çalıştırır. Öte yandan, problem çözme stratejileri tasarlamak gerekmektedir. Sonuç, dizilere, tekrarlara ve koşullara dayalı bir dizi talimattan oluşan bir algoritma olacaktır. Tüm bu sürecin son adımı başlı başına kodlamadır. Bu çalışma sırasında kodu test etmek ve hata ayıklamak için protokoller takip edilmelidir (Sanz, 2015; Vatansever, 2018).

Programlama eğitiminin bilgi işlemsel düşünme becerilerini kavramanın ve geliştirmenin bir yolu olduğu söylenebilir (Korkmaz, 2016). Ayrıca bilgi işlemsel düşünme becerisini kapsayan problem çözme becerilerinin kazandırılması sürecinde, programlamanın ve programlama kavramını kapsayan algoritmik düşünmenin kazandırılması sürecinde, bilgi işlemsel düşünme becerisinin olumlu bir etkisi olduğu ifade edilebilir (Adsay ve diğ., 2020).

Programlama öğretimi ile bireylerin matematiksel düşünme becerilerini geliştirebileceği, problemlere -programlama ile kazanılan- pratik ve yaratıcı fikirlerle yaklaşarak alternatif düşünme yollarını bulabilecekleri, çözümlere eleştirel bakabilecekleri belirtilmiştir. Bu sayede, programlama becerileri bireylere algoritma oluşturma, problem çözme, mantıksal, analitik ve yaratıcı düşünme becerilerini kazandırabileceği, bu becerilerin geliştirilmesinde katkı sağlayabileceği ifade edilmektedir. (Ersoy ve diğ., 2011; Çetin, 2012; Bala ve Alacapınar, 2021; Mohammed ve diğ., 2018)

2.3 21. Yüzyılda Programlama Becerisi

Bilim ve teknoloji alanında yaşanan hızlı değişimler, 21. yüzyıl becerileri kavramının oluşmasına zemin hazırlamıştır. Gelecek neslin dünyasının, günümüzden daha çevrim içi ve sayısal olacağı öngörülmektedir. Hızla gelişen bu dünyada oluşabilecek problemlerin nasıl çözüleceği, mantığın nasıl kullanılacağı ve yaratıcılığın nasıl sağlanacağını bilen, bu dünyaya ayak uydurabilecek yetişkinlere ihtiyaç duyulacaktır.

Bugünün çocuklarını, geleceğin dünyasına hazırlamak için, onlara geleceğin ihtiyaçlarına yönelik bilgi ve becerileri kazandırmak önem arz etmektedir (Şoltan, 2018).

21. yüzyıl becerileri: Eleştirel, analitik, bilgi işlemsel, yaratıcı ve yansıtıcı düşünebilen; yetkili öğrenen, problem çözebilen, yüksek verimlilik sağlayabilen, nitelikli, etkili iletişim kurabilen ve işbirlikçi çalışabilen, sayısal okuryazarlık becerisine sahip, bilgi teknolojilerini kullanabilen, yenilikçi ve yaratıcı tasarımcı, küresel yetkinliğe ve finansal okuryazarlık becerisine sahip, esnek ve uyarlanabilir olarak çeşitlendirilmektedir (Çoklar ve Akçay; 2018; Korucu ve Ünüvar, 2020; Sayın ve Seferoğlu, 2016; Şoltan 2018, ISTE, 2016; Konyaoğlu; 2019, Papatğa, 2016; Tsarava, 2017; Akpınar ve Altun; 2014). Bilgi işlemsel düşünme becerisinin 21. yüzyıl becerilerinden olduğu pek çok araştırmacı tarafından dile getirilmiştir (Otu, 2020).

Bu temel becerilerin birkaçının altyapısını oluşturan; mantıksal akıl yürütme ve bilgi işlemse düşünmenin bir parçası olarak görülen programlama becerisi de bunlar arasında sayılmaktadır (Sayın ve Seferoğlu, 2016). Pek çok araştırmacıya göre, programlama, bilgi işlemsel düşünme becerilerini geliştirmektedir (Çakır, 2020). Programlama becerisi ayrıca; problem çözme, akıl yürütme, ilişki kurma, analitik düşünme, bilgi okuryazarlığı, sayısal okuryazarlık, iletişim kurma, kendini yönetme ve iş birliği kurma, sosyal sorumluluk alma, eleştirel ve yaratıcı düşünme becerilerinin geliştirilmesinde yardımcı olmaktadır (Wing, 2006; Sayın ve Seferoğlu, 2016; Şoltan 2018, ISTE, 2016; Konyaoğlu; 2019, Papatğa, 2016; Tsarava, 2017; Çoklar ve Akçay, 2018).

Öğrenciler programlama becerilerini öğrendiklerinde, matematikteki kavramların çoğunu daha kolay anlayabilecek; bilgi teknolojilerini sürekli kullanacakları için öğrencilerin problem çözme ve analitik düşünme; işbirlikçi çalışma ve öğrenme becerileri ve dijital okuryazarlıkları gelişecektir. Ayrıca öğrencilerin okulla olan ilişkileri, derslere olan motivasyonlarının artabileceği de görülmektedir (Akpınar ve Altun, 2014).

“Kodlama, fiziki ve coğrafi sınırlar gözetmeksizin insanların birlikte veya uzaktan çeşitli ortamlarda işbirlikçi çalışmalar yaparak yaratıcılıklarını geliştirmelerini ve evrensel bir dilde iletişim kurmalarını sağlamaktadır” (Şoltan, 2018, s. 8).

Bireyler programlama yaparken, önce problemi kavrayabilmeleri, çözüme yönelik adım adım analiz yapmaları, işleyiş sırası belirlenmeli, sonuca ulaşmak için bu işleyiş sırası takip edilerek bir algoritma oluşturmaları, daha sonra uygun bir programlama dilinde kodlanması ve değerlendirilmesi; tüm bunları yaparken bilgi işlemsel düşünme becerini geliştirmeleri gerekmektedir (Şoltan; 2018; Ceylan, 2020). Başka bir ifadeyle, algoritma oluşturmak, programlama için bir temel oluşturmaktadır.

Ayrıca, programlama becerisine sahip öğrencilerin matematik, fizik, go, satranç gibi alanlarda daha başarılı olmaları, onlarla daha ileri düzeyde bilgi sahibi olmaları ve hatta bu becerileri geliştiren programlama bilgisine sahip olmaları ile açıklanabilir (Çoklar ve Akçay, 2018).

Tüm bunlara göre, 21. yüzyıl becerilerinin pek çoğuyla ilişkili olduğu düşünülen programlama eğitimi her yaşta bireyin almasının doğru bir yaklaşım olacağı söylenebilir (Adsay ve diğ., 2020).

2.4 Küçük Yaş Grubuna Verilen Programlama Eğitimi

Tüm dünyada programlama eğitimleri önemli bir rol oynamaya devam etmektedir. Günümüzün sayısal toplumunda giderek daha önemli olarak algılanan, problem çözme ve mantıksal düşünme becerileri gibi becerilerle tüm öğrencilerin donatmak ve bilgi işlem alanlarında çalışacak vasıflı işgücü eksikliği, programlamayı her seviyedeki eğitim müfredatına entegre etmenin sebepleridir (Balanskat ve Engelhardt, 2015). Bu yüzden de dünya çapında hükümetler, araştırmacılar, teknoloji ve bilim üretiminde söz sahibi kurum ve kuruluşlar, bilgi işlemsel düşünmenin temel bir unsuru olan programlamanın çocuklara öğretilmesi taraftarıdır (Çakır, 2020; Şoltan, 2018, Adsay ve diğ., 2020). Çünkü, 21. yüzyıl bireylerinde bulunması gereken; işbirlikçi öğrenme, problem çözme, eleştirel düşünme, yaratıcılık, algoritmik ve bilgi işlemsel düşünme gibi temel beceriler programlama ve bilgisayar bilimleriyle kazandırılabilir (Çatlak ve diğ., 2015).

Sayın ve Seferoğlu (2016) makalelerinde, gelecekte programlama eğitiminin çağın ihtiyaçlarını karşılamak için zorunlu olacağını dile getirmişlerdir. Bilgisayarla donatılan çevremizde, çocukların bilgisayarları nasıl kullanacaklarını bilmenin ötesinde, onların

kullandıkları bilgisayar programları ve oynadıkları bilgisayar oyunlarının nasıl yapıldığı, altında yatan programlama mantığına dair fikir edinmeleri gerekmektedir. Bilgi işlemsel düşünmeyi programlama ile öğrenmek, bu beceriyi geliştirmede önemli bir katkı sağlamaktadır. Bu, insan ile bilgisayar arasındaki ilişkiyi anlayarak, karışılacakları güncel problemleri çözmelerini kolaylaştırır. Ayrıca, 21. yüzyıl becerilerinden olan yenilikçi ve yaratıcı tasarım, etkili iletişim, sayısal okur yazarlık gibi becerilere de sahip olacaklardır (Şoltan, 2018; Otu, 2020). Buna ek olarak 21. yüzyıl insanı, problem çözme, mantıksal ve eleştirel düşünebilme becerilerine sahip bireyler olarak yetiştirilmelidirler. Çocuklar programlama ile sıfırdan bir ürün oluşturabilir, tasavvurlarını bilgisayarın işleyebileceği koda dönüştürmek için bilgi işlemsel düşünme becerilerini kullanırlar. Eğer beklenen çıktı elde edilemezse, neyin yanlış gittiğini anlamak için hatalarını sorgulayarak, mantıksal akıl yürütme ve problem çözme becerilerini kullanarak sorunu çözümlenip doğru çıktıya ulaşabilirler (Akkus ve diğ., 2019).

Çocukların programlama deneyimlerini olabildiğince erken yaşta eğlenerek öğrenmeleri, onların günlük hayattaki yaşamlarına, ileride karşılaşacakları karmaşık programlama dillerini ve mantıksal akıl yürütmeyi öğrenmelerini kolaylaştıracağı için yardımcı olacaktır (Otu, 2020). Çocuklar programlama öğrendikten sonra bu alanda kendilerini geliştirmeseler de programlama mantığını küçük yaşlarda kavradıkları için diğer alanlarda da başarılı olabilirler. Çünkü, programlama ile tasarım ilkelerini ve sürecini de kavrayacaklardır. Bu da özgün fikirler bulmalarını, fikirlerini uygulamalarını, hatalarla karşılaştığında bunları tespit edip ayıklamalarını, arkadaşları ile iş birliği içinde çalışma becerilerini öğrenmelerini sağlar (Demirer ve Sak, 2016; Çakır, 2020; Sağay, 2019; Vatansever, 2018).

Programlama eğitimi için, uygulama yapılacak platformun mutlaka çocuğun seviyesine, gelişimine ve bilgi düzeyine uygun olması önemlidir. Doğru bir platformun seçilmesi öğrenime büyük katkı sağlayacak, çocukların motivasyonlarını arttıracaktır. Bu yüzden programlama eğitimi için hangi platformun seçileceği konusu çok önemli bir meseledir (Sağay, 2019). Programlamaya yeni başlayan bireylerin ve çocukların programlamayı öğrenmelerini ve uygulama yapmalarını kolaylaştırmak ve programlamayı daha eğlenceli hale getirmek için, blok tabanlı görsel programlama

araçları ve oyun tabanlı platformlar kullanılmaktadır (Totan, 2021). Bu platformlardan bazıları: Scratch, Mblock, Alisblock, code.org, thinkerCad circuits, blockly, alice, code game vb. programlardır (Yıldız ve diğ., 2020). Bunlara ek olarak, oyun tabanlı programlama eğitimlerinde platforma entegre edilen resim, müzik, video görsel efektler gibi multimedya araçları öğrencilerin derse yönelik tutumlarını olumlu etkilemektedir (Totan, 2021).

Türkiye’de ise ortaokul müfredatındaki Bilişim Teknolojileri ve Yazılım dersi içeriği ile öğrenciler programlamanın temel mantık kurallarını; sıralama, değişkenler, koşullar, döngüler, operatörler ve nesne tabanlı programlamayı öğrenmektedirler (Şoltan, 2018; Millî Eğitim Bakanlığı, 2018).

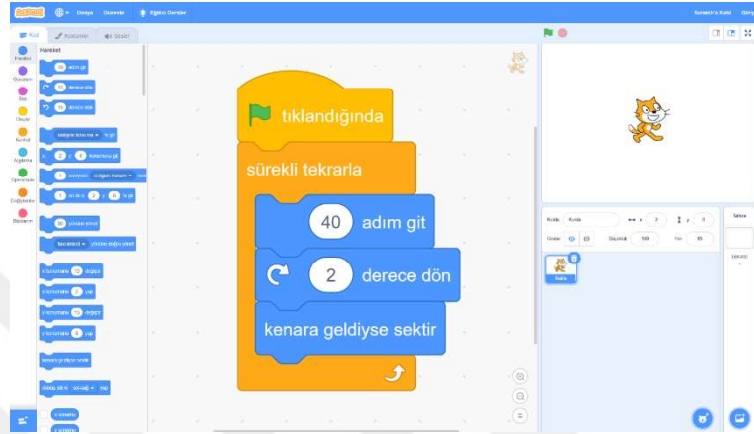
İngiltere 2014 yılında, ilkokul ve ortaokullarda çocuklara kodlama öğretimini zorunlu kılan dünyadaki ilk ülke olmuştur. Bilgisayar dersi müfredatı, bilgisayarların nasıl çalıştığı ve programlamanın temelleri dahil olmak üzere bilgisayar bilimini vurgulamaktadır. Öğrencileri gerçek dünya problemlerini ele almak için bilgisayar programları tasarlamaya teşvik etmektedir, 5 yaşından 16 yaşına kadar tüm okullarda zorunludur (Birleşik Krallık Hükümeti, 2014).

2.5 Scratch

Scratch, Massachusetts Teknoloji Enstitüsü tarafından geliştirilmiş 8-16 yaş çocuklara yönelik bir projedir (Çakır, 2020). Bu projenin amacı teknoloji kullanma becerilerini arttırmak için çocukların anlamlı ve değerli ürünleri teknolojiyi kullanarak ortaya çıkarmalarını sağlamaktır. Scratch herkesin bilgisayar programlamayı kolay bir şekilde anlamasını sağlamak için oluşturulmuştur ve Scratch’in temel tasarım özellikleri sayesinde bu mümkündür (Çakır, 2020). Ayrıca Scratch, temel programlama eğitimi için bireylerin programlama mantığını ve algoritmik düşünme becerilerini geliştirmeyi amaçlamaktadır (Yünkül ve diğ., 2017).

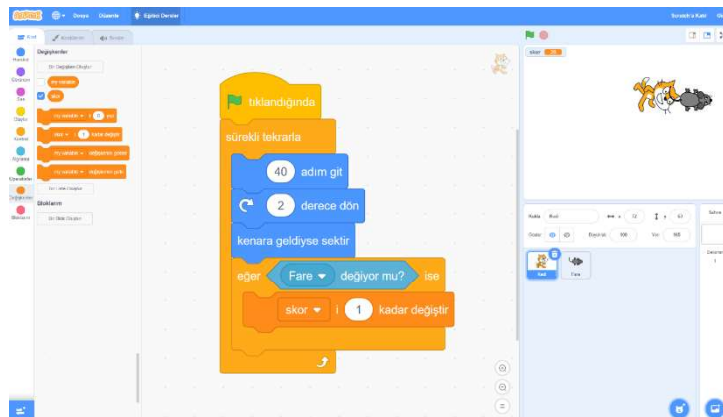
Bir blok tabanlı programlama aracı olarak Scratch, programlama ve bilgisayar bilimine yeni başlayanlar için görsel ipuçları vermekte, sürükle-bırak seçenekleri sunarak kolay kullanım sağlamakta ve sözdizimi hatalarının oluşmasını engellemektedir (Yıldız ve diğ., 2020).

Scratch'te kod blokları hazır olarak bulunmaktadır. Bir karakteri hareket ettirmek için gerekli olan kod bloğu kategoriler altından seçilerek editöre sürüklenir. Kod blokları alt alta dizilerek yukarıdan aşağıya çalışacak şekilde bir yönerge oluşturulur. Bu algoritmayı hazırlamak için oldukça kolay bir yöntemdir. Bu yönergeye göre ekrandaki kukla sırasıyla komutları uygular.



Şekil 2.4 Scratch'te kod ekleme

Bu komutlar, mantıksal kavramları da içermektedir. Kodu bir koşula bağlı olarak çalıştırmak mümkündür. Örnek olarak, kedi kuklasının fare kuklasına değdiği takdirde skor artacaktır. Skoru 1 kadar değiştirme kodu, kedi kuklasının fare kuklasına değme şartına bağlıdır.



Şekil 2.5 Scratch'te koşul kodu ekleme

Bu kodların sürekli tekrarlama döngüsü içinde bulunma sebebi ise, başlangıçtan itibaren değme şartını sağlayıp sağlamadığını kontrol etmek için gereklidir. Eğer döngü olmasaydı, kod sadece bir kere çalışacak ve işlem sonlandırılacaktı.

Scratch programlama komutlarını ortadan kaldırarak, sözdizimi problemleriyle uğraşmadan, sadece algoritma yapıları kullanılarak pratik ve kolay bir biçimde kod yazmayı sağlamaktadır. Dolayısıyla Scratch ile programlama öğretimi sürecinde ilk olarak algoritma ve algoritma yapısının öğrenilmektedir. Bu ise, önemli bir beceri olan algoritmik düşünmenin geliştirilmesini sağlar (Şoltan, 2018; Yünkül ve diğ., 2017).

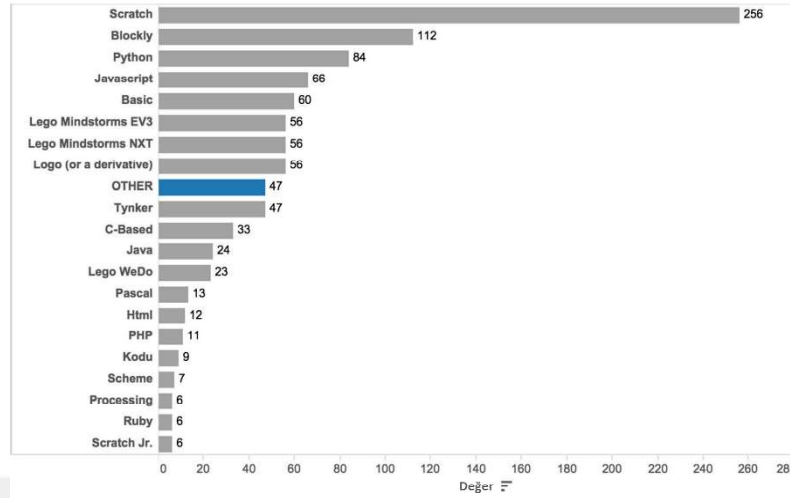
Scratch, ilkökul öğrencilerinin bilgisayar programlamanın temellerini öğrenmelerine yardımcı olmaktadır. Görsel programlama, öğrencilerin metinleri kullanarak geleneksel yöntemle kıyasla programlama dilini öğrenmede daha fazla motive olmasını sağlar sözdizimi hatalarıyla uğraşmak zorunda kalmazlar (Muhammad ve diğ., 2018)

Öğrenciler Scratch programının oldukça kolay kullanılır, anlaşılır ve eğlenceli olduğunu belirtmişlerdir (Korkmaz, 2016). Scratch'in çocukların matematiksel düşünme, problem çözme, mantık ve analitik düşünme becerisini geliştirdiği için, öğrencilerin erken yaşta programlama eğitimine Scratch ile başlaması gerektiği tavsiye edilmektedir (Korkmaz, 2016).

Öğrenciler programlama yaparken kod blokları oluşturmak için bazı matematiksel ve cebirsel konseptler kullanırlar. Bunlardan biri olan değişkenler ifadesini çocuklar, Scratch sayesinde eğlenceli bir şekilde öğrenirler. Hazırladıkları oyunlarda skor, süre gibi kavramları değişken bir değer olarak belirlerler. Böylece değişkenlerin neyi temsil ettiğini öğrenerek cebirsel bir ifade olan değişken kavramını kolaylıkla anlamaktadırlar. Bunlar gibi, soyut kavramları görsellik katarak ve uygulayarak öğrenmek, kavramları somut hale getirerek öğrencilere sevdirmeyi sağlar (Okuducu, 2020).

Çocukların kullandığı programlama dilleri sıralamasında, Scratch dünya çapında en çok kullanılan blok tabanlı görsel programlama dili olup diğer dillere göre daha çok düşündüren, daha çok akıl yürütme becerilerinin kullanıldığı ve daha çok

sosyalleştiren bir tasarıma sahiptir (Rich ve diğ., 2019; Resnick ve diğ., 2009; Çatlak ve diğ., 2015).



Şekil 2.6 Frekansa göre öğretilen bilgisayar dilleri (Rich ve diğ., 2019, s.318)

3. ÖĞRENME VE ÇOKLU ZEKÂ TEORİSİ

On yıllar boyunca bilim insanları ve zekâyı araştıranlar tarafından üç soru tartışılmaya devam etmiştir: Zekâ tekil midir, yoksa çeşitlenebilir mi? Zekâ ağırlıklı olarak kalıtsal mıdır? Zekâ testleri taraflı mıdır? Gardner'a göre insan zekâsı, IQ (zekâ katsayısı) gibi testlerle ölçülmektedir, ancak bu tür testler ise dilsel ve mantıksal beceriyi ölçme ile sınırlandırılmıştır. Bu tür testler bu sorunların çözümüne dayalı akademik başarıyı tahmin edebilirler, ancak profesyonel, sosyal veya kişisel başarıyı tahmin edemezler (Rosiña, ve diğ., 2020; Gardner, 1999; Gülşen, 2015; Gardner, 1983). Halbuki insanın bilişsel becerisi çok daha geniş bir yelpazeye sahiptir. Gardner'ın (1983) önerdiği çoklu zekâ teorisi, zekâ konusundaki asırlardır süren tek zekâ türü baskısını kırmaya yardımcı olmuştur. Beyin bilimciler ve genetikçiler, insan kapasitelerinin inanılmaz farklılaşmasını belgelemekte, bilgisayar programcıları her defasında farklı şekillerde akıllı sistemler yaratmakta ve eğitimciler, öğrencilerinin kendine özgü güçlü ve zayıf yönleri olduğunu yeni yeni kabul etmektedirler (Gardner, 1999).

Gardner (1983) zekâyı, bireyin günlük veya mesleki hayatında karşılaştığı sorun ve zorlukları çözmek amacıyla etkili bir ürün ortaya koymasını sağlayan bir dizi problem çözme becerileri, buna ek olarak, yeni bilgiler edinmeye zemin hazırlayacak problemler bulma veya çıkarma olarak tanımlamıştır. Günlük hayatta karşılaştığımız

problemler çok çeşitli olduğundan, bireyler farklı düşünüş tarzlarına sahip olmalıdır. Buna teoriye göre, her birey karşılaşacağı problemleri çözmeye sekiz zekâ alanını kullanmaktadır. Bu zekâ türleri, matematiksel/mantıksal, sözel/dilsel, görsel/uzamsal, duyuşsal/müziksel, içsel/öze dönük, sosyal/kişisel arası bedensel/kinestetik ve doğasal zekâ olarak ayrılmaktadır (Gülşen, 2015).

Her bireyde bu zekâ alanları birbirinden bağımsız olarak, biri ya da birkaçı, nadiren de olsa tümü güçlü veya zayıf olabilir (Cirit, 2018). Bireyler, karşılaştıkları problemleri daha aktif ve gelişmiş olan zekâ alanları ile çözerlerse başarılı olma ihtimalleri daha yüksek olabilir (Cirit, 2018; Kesici, 2016). Örnek olarak, matematiksel/mantıksal zekâsı daha güçlü olan bir birey, karşılaştığı problemleri bu zekâ alanının özellikleri ile ilişkilendirerek çözmeye eğilimindedirler. Bireylerin problem çözmeye becerileri gibi, bir kavrama, olaya karşı tutumları da farklılık göstermektedir (Cirit, 2018).

Gardner çoklu zekâ teorisinin eğitimde doğrudan uygulanabilirliğinin altını çizmektedir (1983). Çoklu zekâ teorisi eğitim müfredatında uygulandığında, farklı öğrencilerin öğrenmesine ve gelişmelerine yardımcı olabilir. Müfredatın öğrencilerin özelliklerine göre uyarlanması, çoklu zekâ teorisine uygun sınıf yönetim stratejilerinin belirlenmesi, öğrenciler için daha tatmin edici deneyimler ve dolayısıyla anlamlı öğrenmeyi destekleyerek derse yönelik olumlu duygular üretecektir (Rosiña ve diğ., 2020; Gülşen, 2015).

Çizelge 3.1 Çoklu zekâ alanları

Zekâ Türü	Üstün olduğu konular	Yapmayı sevdiği işler	En iyi öğrenme şekli
Sözel/Dilsel	Dilbilimsel-sözlü okuma, yazma, hikâye anlatma, tarihleri ezberlemek, kelimeleri düşünmek	Okumak, yazmak, hikayeler anlatmak, konuşmak, ezberlemek, yapbozlar yapmak	Sözcükleri okumak, dinlemek ve görmek, konuşmak, yazmak, tartışmak ve münazara etmek
Mantıksal/Matematiksel	Matematik, akıl yürütme, mantık, problem çözmeye, örüntüler	Problem çözmeye, sorgulama, sayılarla çalışma, deneyler	Örüntüleri ve ilişkileri kullanma, sınıflandırma, soyutla çalışma
Görsel/Uzamsal	Görselleştirme, tasarım, sahneleri tanıma, ince detayları fark etme	Tasarlamak, çizmek, inşa etmek, yaratmak, hayal kurmak, çizimlere bakmak	Çizimlerle çalışmak ve renkler, görselleştirme, zihin gözüyle görme, çizim
Bedensel/Kinestetik	Atletizm, dans, dramatik sanat, el sanatları, alet kullanımı	Hareket etmek, dokunmak ve konuşmak, beden dilini kullanmak	Dokunma, hareket etme, bedensel duyuşlar yoluyla bilgiyi işleme

Müziksel/Duyumsal	Şarkı söyleme, sesleri tanıma, melodileri, ritimleri hatırlamak	Şarkı söylemek, mırıldanmak, enstrüman çalmak, dinlemek	Ritim, melodi, şarkı söyleme, müzik ve melodi dinlemek
Kişiler arası/Sosyal	İnsanları anlama, liderlik etme, organize etme, iletişim kurma, anlaşmazlıkları çözme, satış	Arkadaş edinmek, insanlarla konuşmak, insanlarla bir araya gelmek	Paylaşmak, karşılaştırmak, ilişkilendirmek, görüşme, iş birliği
İçsel/Öze Dönük	Kendini tanıma, kendini tanıma güçlü ve zayıf yönler, hedefler belirleme	Yalnız çalışmak, düşünmek, ilgi alanlarını takip etmek	Tek başına çalışmak, projeleri kendi hızında yapmak, kendine yer açmak, yansıtmak
Doğasal	Doğayı anlama, ayırım yapma, flora ve faunayı tanımlama	Doğaya katılın, ayırım yapın	Doğal ortamda çalışmak, yaşamı keşfetmek canlılar, bitkiler ve doğanın konuları hakkında bilgi edinme

Kaynak: Rosiña, ve diğ., (2020).

3.1 Matematiksel/Mantıksal Zekâ

Bu zekâ alanı güçlü olan bireyler, sayıları çok iyi kullanır ve mantık yürütürler. Sınıflandırma, tümdengelimli ve tümevarım kullanarak akıl yürütme, soyut problem çözme, hesaplama ve deneme yoluyla konuyu, yapıyı ve karmaşık ilişkileri, benzer yönleri anlama, neden-sonuç ilişkileri kurabilme becerilerine sahiptirler (Babacan ve Dilci, 2012, Cirit, 2018). Bu zekâ, problemleri mantıksal olarak analiz etme, matematiksel işlemleri gerçekleştirme ve sorunları bilimsel olarak araştırma kapasitesinden oluşur. Sadece bilgisayar ve mühendislik alanlarında değil, aynı zamanda günlük konularda da problem çözme becerisi bu zekâ ile ilişkilidir (Swanson ve diğ., 2015). Bilim adamı, matematikçi, muhasebeci, bilgisayar mühendisi, bilgisayar programcısı, yazılımcı ve istatistikçiler bu zekâ alanının meslek grupları arasında yer almaktadır (Babacan ve Dilci, 2012).

3.2 Sözel/Dilsel Zekâ

Dilin anlamsal, sözdizimi ve ses bilimsel boyutları vasıtasıyla birine bir şey hatırlatma, anlatma, açıklama ya da bir konu üzerinde konuşma üzerine kullanabilme becerisi bu zekâ alanıyla ilişkilidir. Bu zekâ alanı baskın bireyler, sözlü ya da yazılı kelimeleri editör, gazeteci ya da bir yazar gibi etkili kullanma; bir masalcı ya da bir politikacı gibi

hitabet becerilerine; hatırlatma, açıklama yapma gibi dilsel fonksiyonlara sahiplerdir. (Cirit, 2018; Babacan ve Dilci, 2012). Bu zekâyı uygulamak, öğrencilerin daha etkili iletişim kurmaya hazırlanmalarına yardımcı olacaktır. İşverenler, sunum yapmak için net bir şekilde iletişim kurabilen, farklı kültürlerden insanlarla sohbet edebilen ve değişim girişimlerini tartışabilen adaylar aramaktadırlar (Swanson ve diğ., 2015). Bu zekâ alanı güçlü olanlar genellikle avukatlar, spikerler, şairler, editörler, yazarlar ve hikâye anlatıcılardır (Babacan ve Dilci, 2012).

3.3 Görsel/Uzamsal Zekâ

Görsel unsurları tam olarak algılayabilme ve bir şeyleri farklı biçimlere büründürebilme gibi becerilere sahip bireylerde görsel zekâ alanı güçlüdür. Bu bireyler; renk, çizgi, şekil, yapı, mekân ve bunlar arasında ilişki kurma konusunda duyarlıdır. Görsel/Uzamsal zekâ alanı güçlü bireyler, iyi bir alan ve yön duygusuna sahiptir. Bu bireyler, tek bir nesneye bakabilir ve nesneyi birçok açıdan kolaylıkla görebilir (Swanson ve diğ., 2015). Görsel dünyayı doğru algılayabilirler, görsel yapıları yeniden oluşturabilme becerisine sahiplerdir (Cirit, 2018). Mimarlık, harita mühendisliği, cerrahlık, mucitlik ve grafik tasarım gibi mesleklere uygundur (Babacan ve Dilci, 2012).

3.4 Duyumsal/Müziksel Zekâ

Bu zekâ alanı güçlü olan bireyler müziksel olarak düşünebilir; bir olayı müziksel olarak algılama, yorumlama ve müzikle iletişime geçme becerilerine sahiplerdir (Cirit, 2018). Bir müzik parçasındaki ritim, müzik tonları, melodi akort, ses tınısı ve ses rengine karşı hassastırlar. Bestekar, piyanist, müzik eleştirmeni, orkestra şefi ve şarkıcılar bu zekâ alanı güçlü bireylerdir (Babacan ve Dilci, 2012).

3.5 İçsel/Öze Dönük Zekâ

İçsel zekâ, bireyin sahip olduğu huyları, potansiyeli, korkuları, zevkleri, becerileri, hırsı gibi özelliklerini, kısaca kendini tanıma, iç dünyasında yaşadığı deneyimlerini

anlama ve takdir etme; bu bilgi ve anlayış ile çevresiyle uyumlu olabilme kapasitesini içerir (Babacan ve Dilci, 2012; Cirit, 2018). Bireyin kendisine dair etkili bir çalışma modeline sahip olması ve bu bilgileri hayatını düzenlemek için kullanabilmesi bu zekâ türü ile bağıntılıdır (Swanson ve diğ., 2015). Bu becerilerle edindiği bilgileri başkalarıyla da paylaşmaya meyillidir. İçsel zekâ alanını güçlü bireyler kendi öğrenmelerini planlama ve yönetme becerilerine sahiptirler. Ressam, terapist ve ruhani liderler bu alanı güçlü kimselerdir (Babacan ve Dilci, 2012).

3.6 Sosyal/Kişisel Arası Zekâ

Bu zekâ alanı güçlü kimseler, diğer insanların sahip oldukları özelliklerin farkında olarak, bireylerin ayrımlarını görebilme ve bu doğrultuda başkalarının yararına rehberlik yapabilme becerisine sahiptirler. Bireylerin duygularını, ruh hallerini, güdülerini, motivasyonlarını, heves ve niyetlerini, nasıl çalıştıklarını, onlarla nasıl ortaklaşa çalışılabileceğini ve problemleri nasıl çözebileceklerini kolaylıkla algılayabilirler ve ayırt edebilirler. Yüz ifadeleri, ses ve mimiklerini ustaca kullanma konusunda başarılıdırlar (Babacan ve Dilci, 2021; Swanson ve diğ., 2015; Cirit, 2018). İşverenler, işbirlikçi ekip projeleri ve karar verme için bu zekâdan gelen iletişim becerilerine ihtiyaç duyar. Bu zekâ, bir kişinin diğer insanlarla etkili bir şekilde çalışmasına izin verir. Eğitimci, satış görevlisi, dini ve siyasi liderler, danışmanlar, dernek kurucuları, idareciler sosyal zekâ alanı güçlü bireylerdir (Babacan & Dilci, 2012; Swanson ve diğ., 2015).

3.7 Bedensel/Kinestetik Zekâ

Bedensel hareketlerini kontrol eden ve hedeflerini başarılı bir biçimde gerçekleştiren bireyler bu zekâ alanına girmektedirler. Bu zekâ, denge, güç, esneklik, hız, el becerisi, koordinasyon gibi becerileri de içermektedir. Bedensel zekâ alanı baskın bireyler, fikirleri ve duyguları vücudu kullanarak ifade edebilme, problem çözebilme becerilerine sahiptirler (Cirit, 2018). Bu zekanın usta örnekleri arasında aktör, sanatkar, atlet, dansçı ve heykeltıraşlar gelir (Babacan ve Dilci, 2012). Ayrıca, bedensel

zekâsı baskın olanların hareketli ve yerinde duramayan bireyler olduđu da belirtilmiştir (Kuru, 2001).

3.8 Doğasal Zekâ

Bu alanda güçlü olan bireyler, doğaya ve çevreye ilişkin bir bilinç oluşturan duyarlı bireylerdir. Bu bireyler, çevreyi gözlemlene, doğadaki türleri belirleme, dürbün ve teleskop gibi araçlarla doğayı inceleme gibi etkinliklerle öğrenmeyi tercih ederler. Bulut kümeleri, dağlar, hayvan ve bitki topluluklarını tanımak ve onlarla ilgilenmek isterler. Doğadaki önemli farklılık ve değişiklikleri ayırt edebilir, bu beceriyi avcılık, tarım ya da biyolojik bilimler gibi ürün ve sonuç çıkartabilecek bir şekilde kullanabilirler (Cirit, 2018). Çevrede olan değişikliklere ve farklılıklara karşı duyarlıdırlar. Zoolog, biyolog, veteriner, avcı gibi meslek grupları bu zekâ alanında yer alır (Babacan ve Dilci, 2012; Swanson ve diğ., 2015).

4. KURAMSAL ÇERÇEVE

Araştırmanın bu bölümünde Scratch, programlama, düşünme becerileri, bilgi işlemsel düşünme, bilgisayara yönelik tutum ve zekâ alanları üzerine yapılan araştırmalara; programlama eğitimdeki sorunlara, öz-yeterlilik algısı ile performans ilişkisine ve özgünlük boyutuna yer verilmiştir.

4.1 Scratch ile İlgili Araştırmalar

Talan (2020), yaptığı araştırmasında Scratch üzerine yapılmış olan 76 farklı çalışmayı incelemiş, bu incelemeler sonucunda Scratch yazılımını kullanmanın motivasyon, öz-yeterlilik, tutum, üst düzey düşünme ve akademik başarıya olumlu etkileri olduđu sonucuna ulaşmıştır. Ayrıca Scratch ile programlama öğrenmenin, soyut ve karmaşık yapıdan çıkararak soyutlaştırmayı başarılı bir şekilde sağlamaktadır. Buna ek olarak, Scratch, 21. yüzyıl becerilerini erken yaşta kazandırmaktadır. Öğrencilerin ilgi ve motivasyonlarını yükseltmesi, kendilerine güven kazandırması gibi olumlu yanlarından da söz edilmektedir.

Papatğa (2016) tarafından yapılan arařtırmada, ilkokul öğrencileriyle Scratch ile hazırladıkları projeler sonucunda okuduğunu anlama sorunu yaşayan öğrencilerin okuduğunu anlama becerilerinin geliştiđi gözlemlenmiştir.

Brennan ve Resnick (2012) yaptıkları çalışmada Scratch projelerinin bilgi işlemsel düşünme becerilerini geliřtirdiđini belirtmişlerdir.

Serim (2019) tarafından 5. ve 6. Sınıf öğrencileriyle yapılan arařtırmada, Scratch ile oyunlařtırma yaklařımı ile yapılandırılan 8 haftalık programlama eğitimi sonunda öğrencilerin programlamaya yönelik öz-yeterlilik algılarının ve bilgi işlemsel düşünme becerilerinin olumlu etkilendiđi bulunmuştur. Ayrıca bilgi işlemsel düşünme becerileri ile programlamaya yönelik öz-yeterlilik algıları arasında yüksek düzeyde pozitif anlamlı bir ilişki olduđu ortaya çıkmıştır.

Ökten ve Atasoy'un (2020) arařtırmaları sonucu, Scratch ile programlama yapan öğrencilerin uzamsal akıl yürütme becerilerinin programlama öğrenmemiş olan öğrencilere göre anlamlı düzeyde farklı olduđu; deney grubunun derse karşı olumlu bir tutum sergilediđi ve bilişsel, duyuşsal ve psikomotor becerilerini üst düzeyde kullandıkları ortaya çıkmıştır. Bu bağlamda öğrencilerin görsel-uzaysal akıl yürütme becerilerini geliřtirmede programlamanın önemli bir faktör olabileceđi ifade edilmiştir.

Kaleliođlu ve Gülbahar (2014) 5. Sınıf öğrencileriyle yaptıkları arařtırmalarında, Scratch platformunda programlamanın, öğrencilerinin problem çözme becerilerinde anlamlı bir farklılıđa neden olmadıđını ortaya koymuştur. Sadece “problem çözme becerisine güvenme” faktörünün ortalamasında anlamlı olmayan bir artış olduđunu belirtmişlerdir. İlköğretim öğrencilerinin düşüncelerine bakıldıđında tüm öğrencilerin programlamayı sevdiklerini ve programlamalarını geliřtirmek istedikleri ve öğrencilerin çođunun Scratch platformunun kullanımını kolay bulduklarını açıklamışlardır

Sade (2020) tarafından gerçekteřtirilen çalışmadan elde edilen bulgulara göre Scratch ile blok tabanlı programlama eğitimi alan ortaokul öğrencilerinde, geleneksel bilgisayar eğitimi alanlara göre, matematik kaygısı daha düşük, problem çözme algısı

daha yüksek tespit edilip bilgi işlemsel düşünme becerisinde istatistiksel olarak anlamlı bir fark tespit edilememiştir.

Oluk ve diğ., (2018) yaptıkları araştırmada, Scratch kullanımının algoritma geliştirme ve bilgi işlemsel düşünme becerilerini geliştirmede etkisini 62 5. Sınıf öğrencisi ile incelemiştir. Deney grubu öğrencilerine Scratch programı kullanılarak, kontrol grubu öğrencilerine mevcut müfredata göre algoritma anlatılmıştır. Öğrencilere ön ve son test olarak bilgi işlemsel düşünme ölçeği ve algoritma geliştirme başarı testi uygulanmıştır. Scratch ile programlama yapan öğrencilerin algoritma geliştirme ve bilgi işlemsel düşünme becerilerinin kontrol grubuna göre anlamlı derecede daha fazla geliştiği gözlemlenmiştir.

Calder (2018), öğrencilerin Scratch ile programlama etkinliklerine yönelik yaptığı araştırma raporunda, oyunlar da dahil olmak üzere matematiksel ve sayısal öğrenme nesnelere oluşturmak için yazılım programını kullanan 10 yaşındaki öğrencilerin matematiksel fikirlerle etkileşimlerinin kolaylaştığını ortaya koymuştur.

Gökdağ (2019), Scratch ile matematiksel şekillerle süsleme etkinlikleri yapan 6. Sınıf öğrencileriyle yaptığı araştırmasında, öğrencilerin matematiksel süsleme oluşturma sürecinde keşif ve sezgilerden yararlanmışlardır. Bunun yaratıcı düşünme ile doğrudan ilişkisi olduğu açıklanmıştır. Ayrıca yazar, öğrencilerde matematiksel estetik kavramlarını ortaya çıkardığı için Scratch gibi programlama programlarının kullanılmasının önem kazandığını belirtmektedir.

Bala ve Alacapınar (2021), 6. Sınıf öğrencileriyle 8 saatlik yaptığı araştırmada, 22 öğrencinin ilk ve son testlerine göre, Scratch programının öğrencilerin problem çözme becerileri üzerinde istatistiksel olarak anlamlı bir etkisinin olmadığını, derse yönelik tutuma ilişkin son test puanlarında artış olduğunu tespit etmiştir.

Ceylan (2020) tarafından Scratch ile programlama etkinliklerinin bilgi işlemsel düşünme becerileri düzeylerine etkisi hakkında yaptığı çalışmada, 6. Sınıf öğrencilerine yönelik eğitimin olumlu sonuçlar ortaya çıkardığını ifade etmiştir. Ön test ve son test ile yaptığı deneyinin sonunda her iki gruba da bilgi işlemsel düşünme becerileri ölçeğinde Scratch eğitimi alan öğrencilerin daha yüksek puanlar elde ettiği görülmüştür. Buna göre, Scratch'in problem çözme, algoritmik düşünce ve yaratıcı

düşünme becerileri üzerinde anlamlı bir etkisi olduğu sonucuna varılabileceğini ifade etmiştir. Korelasyon testi sonucunda dönem sonu programlama dersi sınavı puanları ile bilgi işlemsel düşünme becerileri ile istatistiksel anlamda yüksek düzeyde ilişki saptanmıştır. Öğrencilerin bilgi işlemsel düşünme becerileri ve derse ait başarı puanları Scratch eğitimi sonrasında olumlu yönde değişmiştir.

Vatansever (2018), 5. ve 6. Sınıf öğrencilerine yönelik yaptığı çalışmasında, Scratch ile programlama eğitiminin öğrencilerin problem çözme becerileri üzerindeki etkisini araştırmıştır. Bu araştırmaya göre, Scratch etkinliklerinin problem çözme becerilerine orta düzeyde etkili olduğu saptanmıştır. Ayrıca, problem çözme becerisi gelişmiş öğrenciler yetiştirmek için Scratch ile programlama etkinliklerinin gerçekleştirilmesi önerilmektedir.

Çubukluöz (2019) tarafından 6. sınıf öğrencilerine 6 haftalık programlama eğitimi verilmiş, Scratch programlama aracını kullanarak matematiksel oyun hazırlamaları sağlanmıştır. Öğrencilerin matematik dersinde zorlanarak öğrendikleri konuların Scratch programlama aracı ile tasarlanan matematiksel oyunlarla giderildiği ortaya çıkmıştır. Öğrencilerin Scratch ile etkinlikleri sayesinde dört işlem önceliğini anladıkları, üslü ifadelerle ilgili işlemleri doğru yapabildiklerini, matematiksel problem çözme becerilerinin geliştiğini, değişken kavramını, asal çarpanları öğrendiklerini araştırma sonucunda ortaya koymuştur.

Korkmaz (2016), elektrik-elektronik bölümünde okuyan üniversite öğrencileri üzerinde yaptığı araştırmasında, Scratch ile ilgili oyun etkinliklerine dayalı eğitim programının, öğrencilerin mantıksal matematiksel düşünme becerilerine eğitsel robotiklerle yapılan tasarım etkinliklerine ve geleneksel öğretim etkinliklerine dayalı eğitim programlarına göre daha olumlu katkı sağladığını ve bu katkı özellikle matematiksel ilişki faktörü için anlamlı olduğunu ortaya çıkarmıştır.

Özkandemir (2019), ilkokul müzik derslerinde Scratch ve Makey Makey kiti kullanarak yapılan nota okuma yazma çalışmaları yapmıştır. Nitel araştırma yöntemleriyle yapılan inceleme sonucunda, öğrencilerin nota okuma yazma becerilerinde geliştikleri gözlenmiştir. Ayrıca, uygulama sonunda öğrencilerin çoğunluğunun kullanılan uygulamaları sevdiği, Scratch ve Makey Makey ile yapılan nota okuma yazma etkinliklerinden keyif aldıkları belirtilmiştir.

Saygıner (2017), tarafından yapılan arařtırmaya gre, fen bilgisi ğretmenlięi blmnde okuyan niversite ğrencilerine ynelik Scratch ğretimi sonucunda, ğrencilerin eriři, mantıksal dřnme ve motivasyon puanlarında istatistiksel olarak anlamlı bir artıř ve yksek dzeyde bir etki byklę hesaplanmıřtır.

Dięer bir arařtırmada Bahar (2021), ocukların İngilizce dili ve biliřsel geliřimleri zerinde Scratch'in etkisini incelemiřtir. Sonulara gre Scratch'te programlama aktivitelerinin, ocukların İngilizce dinleme ve bilgi iřlemsel dřnme becerileri zerinde nemli bir etkiye sahipken, akademik bařarı, dil kullanımı, anlatım becerileri zerinde nemli bir etkiye sahip olmadığı grlmřtr.

Kert ve dię. (2020), ortaokul ğrencileriyle yaptıkları alıřmanın sonucunda, eęitsel robotiklerle yapılan programlama eęitiminin, bilgi iřlemsel dřnme becerilerini geliřtirmede Scratch'ten daha etkili bir yntem olduęunu gstermiřlerdir.

Son olarak, Arslan ve Iřbulan (2021) ortaokul ğrencilerinin Scratch ve robotik derslerinde uygulanan bireysel ve ekip alıřması etkinliklerinin, programlamaya ynelik z-yeterlilik algılarında ve bu etkinliklere ynelik tutumlarında farklılık oluřturmadıęını ortaya koymuřtur.

4.2 Programlama ve Dřnme Becerileri

 farklı blok tabanlı grsel programlama platformunda programlama eęitimi alan ilkokul ğrencileri zerinde yapılan bir arařtırmaya gre, programlama eęitiminin ocukların problem zme becerilerini geliřtirdikleri sonucuna ulařılmıřtır. Ayrıca, ğrencilerin programlamaya ynelik tutumlarının eęitimden sonra olumlu olarak kaldıęı grlmřtr (Asad ve dię., 2016).

Psycharis ve Kallia (2017) tarafından lise ğrencileri zerinde yapılan arařtırmada, bilgisayar programlamanın lise ğrencilerinin matematikte akıl yrtme becerileri, problem zme ve z-yeterlilik zerinde bir etkisi olup olmadığı arařtırılmıřtır. Arařtırmanın rneklemini eęitim ynelimlerine gre deney ve kontrol grubu olmak zere iki gruba ayrılan 66 lise ğrencisi oluřturmuřtur. Arařtırma bulguları, programlama dersine katılan ğrencilerin muhakeme becerilerinde, katılmayan ğrencilere gre anlamlı bir farklılık olduęunu gstermektedir. Ayrıca deney

grubundaki öğrencilerin öz-yeterlilik göstergeleri kontrol grubundaki öğrencilere göre anlamlı farklılık göstermiştir. Ancak sonuçlar, bilgisayar programlamasının öğrencilerin problem çözme becerilerini büyük ölçüde geliştirdiği hipotezini desteklememiştir.

Çakır (2020) araştırmasında, ortaokullar arasında programlama eğitimi alan ve almayan okulları tespit ederek, bu okullarda okuyan çocuklar arasında yaratıcı problem çözme ve bilişüstü farkındalık düzeylerini incelemiştir. Programlama eğitiminin uygulandığı okullarda öğrenim görmekte olan öğrencilerin yaratıcı problem çözme becerileri ve bilişüstü farkındalık puanları, programlama eğitimi uygulanmayan okullardaki öğrencilere göre daha yüksek düzeydedir. Böylece programlama eğitiminin, öğrencilerin bilişüstü farkındalıklarını ve yaratıcı problem çözme becerilerini geliştirdiğini belirtmektedir.

Sebetci ve Aksu (2014) tarafından bilgisayar programcılığı bölümünde öğrenim gören öğrencilere yönelik mantıksal ve analitik düşünme becerilerinin programlama dilleri başarısı üzerindeki etkisi araştırılmıştır. “Bütüncül ve Analitik Düşünme Ölçeği” ile “Mantıksal Düşünme Becerileri” ölçeği 147 öğrenciye uygulanmıştır. Sonuç olarak, öğrencilerin analitik düşünme becerileri ile programlama dilleri başarısı arasında pozitif yönde, yüksek düzeyde ve istatistiksel açıdan anlamlı; mantıksal düşünme ile programla başarısı arasında ise pozitif yönde, orta düzeyde ve istatistiksel açıdan anlamlı bir ilişki bulunmuştur. Buna göre, öğrencilerin analitik ve mantıksal düşünme becerilerini gelişmesiyle, öğrencilerin programlamada daha başarılı olacakları ifade edilmiştir.

Sáez-López ve diğ. (2019), ilkokul öğrencilerinin mantık ve matematik öğelerini anlamalarını önemli ölçüde geliştirmelerini sağlamak için bir araç olarak görsel programlama dili aracılığıyla programlamayı içeren bir eğitim tasarımının önemini vurgulamaktadır. Yapılan çalışmada, programlama, matematik dersine entegre edilmiştir. Programlama, çalışılan içeriğe motivasyon, coşku, bağlılık, eğlence ve ilgi aşılayan aktif bir pedagojik uygulama ile matematiksel kavramların anlaşılmasında ve hesaplamalı kavramların kazanılmasında istatistiksel olarak önemli gelişmeler sağlamıştır. Yazarlara göre, programlama, mantıksal düşünme, matematik ve yaratıcılığın avantajlarını öne çıkarmaktadır.

Blakemore (2017), ilkokul öğrencilerine yönelik yaptığı araştırmasının sonucunda, programlama aktivitelerine katılmanın problem çözme etkinliğinde elde edilen toplam puanı arttırdığını ortaya koymuştur. Bu da bilgisayar programlamanın problem çözme becerilerini ve dolayısıyla bilgi işlemsel düşünmeyi teşvik etmek için önemli bir pedagojik yaklaşım olduğunu belirtmiştir.

Çiftçi ve Bildiren (2019), okul öncesi çocuklar için programlama kursuna katılan çocuklarda, programlamanın sözel olmayan bilişsel becerilerin gelişimi üzerinde bir etkisi olduğunu, ancak çocukların problem çözme becerilerini geliştirmede anlamlı bir etkisinin olmadığı gözlemlenmiştir.

4.3 Programlama ile Bilgi İşlemsel Düşünme

Yıldırım (2020), alan yazını incelemesinde, programlama öğretiminin ve oyunlarının bilgi işlemsel düşünme becerileri üzerine etkisini inceleyen çalışmaların çoğunda, öğrencilerde bu becerinin arttığını görmüştür. Çalışmalardan sadece birinde görsel programlama etkinliklerinin öğrencilerin bilgi işlemsel düşünme becerisinde anlamlı bir değişim oluşturmadığını belirtmiştir.

Yünkül ve diğ. (2017), çalışmalarının sonucunda bilgi işlemsel düşünme beceri düzeyleri ile programlama becerisi arasında yüksek düzeyde ilişki olduğunu ve bilgi işlemsel düşünme becerileri yüksek olan öğrencilerin programlama konusunda da başarılı olduklarını saptamıştır.

Durak ve diğ. (2019), yaptıkları araştırmada, ortaokul öğrencilerinin problem çözmeye yönelik bilgi işlemsel düşünme, programlama öz-yeterliliği ve yansıtıcı düşünmeye ilişkin beceri düzeylerini belirlemek ve programlama eğitimi sürecindeki deneyimlerini incelemiştir. 6. ve 7. sınıflardan 55 öğrenci ile 10 haftalık uygulama yapılmıştır. Sonuç olarak öğrencilerin bilgi işlemsel düşünme becerilerinin, programlama öz-yeterliliklerinin ve problem çözmeye yönelik yansıtıcı düşüncelerinin orta düzeyde olduğu tespit edilmiştir. Ayrıca, programlamadaki başarı; analitik düşünme ile yüksek, mantıksal düşünme becerisi ile orta, böylece ikisiyle de istatistiksel açıdan anlamlı bir korelasyona sahiptir. Bundan da analitik

düşünme stili, programlama öz-yeterliliği ve programlama tutumu için önemli bir unsur çıkarımı yapılmıştır.

Başka bir araştırmada ise, blok ve oyun tabanlı programlama ortamını kullanan ortaokul öğrencilerinin bilgi işlemsel düşünme ile yaratıcı düşünme becerisi arasındaki ilişki incelenmiştir. Bu araştırma, çözüme dayalı özgünlük ve görev zorluğu arasındaki ilişkiyi incelemiştir. Ancak, yaratıcı düşünme ile bilgi işlemsel düşünme becerisi arasında bir ilişki bulunamamıştır (Herskovitz ve diğ., 2019).

Adsay ve diğ., (2020) ortaokul 6,7 ve 8.sınıf öğrencilerinin blok tabanlı programlama eğitimine dönük öz-yeterlilik algı düzeyleri ve bilgi işlemsel düşünme beceri düzeyleri faktörleri arasında istatistiksel açıdan anlamlı, pozitif bir ilişki görüldüğünü belirtmişlerdir.

Totan (2021) blok tabanlı programlama eğitiminin ortaokul öğrencilerinin bilgi işlemsel düşünme becerisi öz-yeterliliklerinin ve programlama öğrenmeye yönelik tutumları üzerindeki etkisini incelemiştir. 4 hafta süren eğitim sonucunda, blok tabanlı programlama eğitiminin hem bilgi işlemsel düşünme becerileri hem de programlamaya eğitime yönelik tutumlarında yüksek seviyede etkisi olduğu sonucuna ulaşmıştır.

4.4 Programlama, Bilgisayara Yönelik Tutum ve Zekâ Alanları

Çoklu zekâ alanlarıyla programlama becerisini ilişkilendiren ilk kişi Çoklu Zekâ teorisinin kurucusu olan Gardner'dır (1983).

“Bir bilgisayarı programlamayı öğrenme, çoklu zekâ alanları ilgili olabilir. Mantıksal/matematiksel zekâ, programlama becerisi için merkezde görünüyor, çünkü programlama, bir sorunu çözmek veya sınırlı sayıda adımda bir hedefe ulaşmak için katı prosedürlerin uygulanmasına bağlıdır. Programın yazılması, adımların açık, kesin ve kesinlikle mantıklı bir sırada organize edilmesini gerektirir. Kılavuzlar ve bilgisayar dilleri sıradan dili kullandığı sürece, dilsel zekâ da önemlidir. Programın bir hikâye olarak metaforu (alt konularla tamamlanan), dilsel bir eğilimi olan bazı programcılara yardımcı olabilir. Bireylerin belirli alanlarla ilgili sahip oldukları sezgiler, onlara

programlamayı öğrenmede yardımcı olabilir. Bu nedenle, güçlü bir müzikal eğilimi olan bir kişi, programlamaya en iyi şekilde basit bir müzik parçasını programlamaya (veya beste yapan bir programın kullanımında ustalaşmaya) girişerek tanıtılabilir. Güçlü uzamsal algıya sahip bir kişi, bir tür bilgisayar grafiği aracılığıyla programlama öğrenmeye başlatılabilir ve aynı zamanda, bir akış şeması veya başka bir uzamsal diyagramın kullanılması yoluyla programlama görevinde yardımcı olabilir. İçsel zekâ da önemli bir rol oynayabilir. Programlama yaparken gerçekleştirilen kapsamlı adım ve hedefleri planlamak, içsel düşünme biçimlerine dayanır. Karmaşık bir görevi yerine getirmek veya yeni hesaplama becerileri öğrenmek için gereken iş birliği, bireyin bir ekiple çalışma becerisine bağlı olabilir. Bedensel zekâ, bilgisayarın kendisiyle çalışırken rol oynayabilir, terminalde beceriyi kolaylaştırabilir ve bir programın konusunun vücudun kullanımını içerdiği durumlarda (bir dans veya bir dizi futbol oyununun programlanması) kullanılabilir.” (s.390).

Şerefhanoglu (2007) çalışmasında, öğrencilerin, bilgisayara yönelik tutumları ile çoklu zekâ alanları arasındaki ilişkilere bakmış; sözel/dilsel ve bedensel/kinestetik zekâ alanı ile bilgisayara yönelik tutumlar arasında bir ilişkinin olmadığını, içsel/öze dönük ve doğasal zekâ alanları arasında zayıf bir ilişki olmasına rağmen bu ilişkinin anlamsız olduğu sonucuna ulaşmıştır.

Cirit (2018), araştırma sonucunda öğrencilerin çoklu zekâ alanları ile bilgisayara yönelik tutumları arasında düşük ve orta düzeyde anlamlı ilişki olduğu saptanmıştır. Bu bulgulara göre öğrencilerin çok gelişmiş ve gelişmiş olan zekâ alanları bilgisayara yönelik tutumlarında önemli bir farklılık oluşturmamaktadır.

Washington Üniversitesi'nden yapılan bir araştırma, dil öğrenme becerisinin, programlamayı öğrenmek için, temel matematik bilgisinden veya aritmetikten daha güçlü bir kolaylaştırıcı olduğunu ortaya koymaktadır. Bunun nedeni, kod yazmanın aynı zamanda ikinci bir dil öğrenmeyi, o dilin kelime ve gramerini öğrenme becerisini ve fikirleri ve niyetleri iletmek için birlikte nasıl çalıştıklarını da içermesidir. Problem çözme ve belleğin kullanımı gibi her iki alana bağlı diğer bilişsel işlevler de kilit rol oynamaktadır (Prat ve diğ., 2020).

Günbatır (2018), üniversite öğrencileri ve mezunlarının düşünme stilleri programlamaya yaklaşımlarını incelediği makalesinde, programlama başarısının, analitik düşünme ile yüksek ve anlamlı bir korelasyona ve mantıksal düşünme becerisi ile orta ve anlamlı bir korelasyona sahip olduğunu ortaya koymuştur. Bu çalışmada programlama tutumu ve düşünme stiline programlama öz-yeterliliğini yordadığı gösterilmiştir. Bu çalışmaya göre analitik düşünme stili; programlama öz-yeterliliği ve programlama tutumu için önemli bir unsurdur.

Daha önce yapılan çalışmalar, erken çocukluk ortamlarında, çocukların sıralama beceriğini, tasarım öğrenmesini, yürütme işlevini, matematik ve dil gelişimini ve mantıksal akıl yürütmeyi geliştirmek için teknoloji ve mühendisliği erken yaşta tanıtmının faydalarını göstermiştir (Strawhacker ve Bers, 2019).

4.5 Zekâ Alanları

Boom ve diğ. (2018) tarafından yürütölen bir çalışma, bilgi işlemsel düşünmenin zekâ ile nasıl ilişkili olduđu sorusunu ele almıştır. Bu soruya cevap bulmak için 71 öğretmen adayı üzerinde uygulanan araştırma sonucunda bilgi işlemsel düşünme ve genel problem çözme becerisinin yüksek oranda ilişkili olduđu saptanmıştır.

Rosina ve diğ. (2020) tarafından gerçekleştirilen başka bir araştırmaya göre, ilköğretim 6. ve ortaöğretim 2. ve 4. sınıflarında öğrenim gören 143 öğrencide, ağırlıklı olarak mantıksal-matematiksel ve/veya görsel-uzaysal zekaya sahip öğrencilerin fen derslerine daha pozitif yaklaştıkları ve derse uyum becerilerinin daha iyi olduđu bulunmuştur.

Messer ve diğ. (2018) tarafından, okul öncesi çocuklar üzerinde gerçekleştirilen bir araştırmada, çocuklar üç gruba ayrılarak eğitsel programlama etkinlikleri gerçekleştirilmiştir. İlk grup tablet ile blok tabanlı programlama uygulamasını kullanmış, ikinci grup kâğıt ve kalemle programlama etkinlikleri gerçekleştirmiş, üçüncü grup ise kalem ve kâğıtla toplama-çıkarma görevlerini içeren karşılaştırma koşulu etkinliđi yapmışlardır. 6 hafta süren eğitimler sonucunda, üç grupta da matematiksel becerilerin ve uzamsal farkındalıkların arttığı ortaya konmuştur.

Oluk ve akır (2019), meslek yksekokulu đrencilerinin bilgi iřlemisel dřnme becerileri ile problem zme becerisi ve mantıksal/matematıksel zekâ z-algıları arasında korelasyon analizi yapmıřlardır. Yapılan analizler sonucunda đrencilerin bilgi iřlemisel dřnme becerileri ile mantıksal matematıksel zekâ z algıları ve problem zme beceri dzeyleri arasında pozitif ynde anlamlı bir iliřki olduđu saptanmıřtır.

oklu zekâ teorisinin nrolojik kanıtları da aranmıřtır. Bařarıyı ve yařam geliřimini en st dzeye ıkarmak iin đretimi kiřiselleřtirme hedefiyle stn zekâlı bireylerin zihinlerini anlamaya ynelik ıkarımlar nrolojik incelemelerle yapılmıřtır. Nrolojik grntleme alıřmaları, beyin aktivasyonu ve yapısı rntleri, iřleyen bellek ve diđer yksek biliřsel beceriler aısından bireyler arasında sistematik řekillerde deđiřiklik gstermektedir. Buna gre, oklu zekâ alanlarının beyinde tutarlı sinirsel mimarilerle iliřkili olduđu, bu bireysel farklılıkların insan performansı zerinde etkilerinin olabileceđi iddia edilmiřtir. (Shearer, 2020; Shearer ve Karanian, 2017).

4.6 Programlama Eđitimindeki Sorunlar

Programlama her bireyin kolaylıkla anlayabileceđi bir konu deđildir. Programlama performansı ve becerisi, kiřiden kiřiye deđiřmekte; bazı đrenciler iin olduka gz korkutucu olabilmektedir. Programlamayı đrenmek, her đrenci iin benzersiz bir deneyimdir ve bir đrencinin neden diđerinden daha iyi ve daha hızlı programlamayı đrendiđi tam olarak anlařılmamıřtır (Ramalingam ve diđer., 2004). Bazı đrenciler, problem zme, plan geliřtirme ve programlama dillerini etkili kullanma konusunda sıkıntı yařamaktadırlar (ifti ve Bildiren, 2019). đrencilerin bir kısmının programlama derslerinde ve Scratch'te kod mantıđını oluřtururken veya hata ayıklarken zorlandıkları gzlemlenmiřtir (Papatđa, 2016; Vatansever, 2018; Bala ve Alacapınar, 2021; ubukluz, 2019). Ayrıca, Kaleliođlu ve diđer. (2020), yaptıkları alıřmada, ortaokul đretmenlerinin uyguladıkları programlama mfredatına ynelik eleřtirilerine yer vermiřlerdir. Bu eleřtiriler: Okullarda 5. sınıf algoritmaları ve problem zme konularının bilgi ve anlayıř dzeyinde kalması, etkinliklerin 5. sınıf đrencileri iin zor olması, etkinliklerin kalabalık gruplarda uygulanmasının

meşakkatli olması, sınıflardaki seviyeyi ayarlamanın ve dengelemenin çok güç olması, Scratch gibi platformların öğrencilere zor gelmesi, şeklinde sıralanmıştır.

Başka bir çalışmada, öğrencilerin basit kodlama gerektiren görevleri yapabildikleri fakat karmaşık kodlama gerektiren görevleri yapamadıkları; Scratch'in özgün ürün geliştirme ve ileri seviye programlama gibi çeşitli sınırlılıkları olduğu belirtilmiştir (Arslan ve Akçelik, 2019). Buna ek olarak, programlama dillerinin ağır kavramsal yapısının özellikle yeni başlayanlar için oldukça zorlayıcı olduğu vurgulanmaktadır.

Son olarak Vatansever (2018), öğrencilerin çoğunun Scratch programında problem çözme sürecinde zorlandıklarını; bir kısım öğrencinin kod bloklarını senaryoya göre yanlış yerleştirdiklerini, öğrencilerin birçok kez, problemi tam anlamadan problemin çözümüne yönelik eyleme geçtiklerini, koşullu ifadelerin seçiminde zorlandıklarını ve hatalar yaptıklarını bildirmiştir.

4.7 Öz-Yeterlilik Algısı ve Performans İlişkisi

Öz-yeterlilik, bireyin belirli bir alanda bir görevi yerine getirme becerisine ilişkin yargısı, kendisine inanması, bir performansta ya da görevde başarılı olacağına dair inancıdır. Araştırmalar, yüksek öz-yeterliliğin, bilgi edinme ve aktarma için önemli olduğunu göstermiştir. Öğrencilerde öz-yeterliliğin ders performansını da etkilediğini göstermektedir. Öz-yeterlilik inancı, eğitim hayatında akademik başarının önemli bir belirleyicisi olmasının yanı sıra, öğrencilerin okul dışındaki tüm yaşamlarında kullandıkları önemli bir beceridir (Ralingam ve diğ., 2004; Arseven, 2016). Öz-yeterliliği yüksek olan bireyler her zaman problemlerini çözme taahhüdüne sahiptirler ve kullandıkları stratejilerin başarısız olduğunu gördüklerinde kolay kolay vazgeçmezler, düşük öz-yeterliliğe sahip olanların ise kolaylıkla cesareti kırılır ve görevde başarısız olurlar (Muhammad ve diğ., 2018).

Hysong ve Quinones (1997) yaptıkları meta-analiz çalışmasında, 30 araştırmayı incelemiş, öz-yeterlilik ve performans arasında anlamlı, pozitif bir ilişki bulmuşlardır.

Stajkovic ve Luthans (1998), 114 çalışma ile yaptıkları meta-analiz araştırmasında öz-yeterlilik ile işle ilgili performans arasındaki ilişkiyi incelemişlerdir. Araştırmasının

sonucunda, öz-yeterlilik ve işle ilgili performans arasında önemli bir ağırlıklı ortalama ($r= 0.38$) korelasyon bulunmuştur.

Programlamaya yönelik performansı etkileyen faktörler arasında öz-yeterlilik algısı önemli bir konumdadır (Ralingam ve diğ., 2004).

Kasalak (2017) yüksek lisans tezinde, bireylerin bir konuya yönelik öz-yeterlilik algıları ile o konudaki başarılarının doğrudan ilişkili olduğunu belirtmiştir. Çocukların blok tabanlı programlamaya yönelik öz-yeterlilik algılarını, blok tabanlı programlamadaki performansları ile ilişkilendirmiştir.

4.8 Özgünlük Boyutu

Literatür tarama sonucunda ortaokul öğrencilerinin Scratch ile blok tabanlı programlamaya yönelik öz-yeterlilik algıları ile çoklu zekâ alanlarının tamamı arasında bir ilişkinin incelenmediği, çoklu zekâ alanları ve blok tabanlı programlama becerisi arasındaki ampirik ilişkiyi araştıran bir çalışma yapılmadığı görülmüştür. Gardner (1983) çoklu zekâ alanları ve programlama arasında ilişki olabileceğini kitabında öne sürmüş olup nicel verilerle desteklememiştir. Bir çalışmada üniversite öğrencilerinin bilgisayar programlama dersinde gösterdikleri başarı ve bu öğrencilerin çoklu zekâ alanları arasında bir korelasyon bulunmuş olup çoklu zekâ kuramının öğrencilerin programlama derslerindeki performanslarının tahmini için kullanılabileceği ortaya konmuştur (Ninrutsirikun ve diğ., 2016). Ayrıca Korkmaz (2016), elektrik-elektronik bölümünde okuyan üniversite öğrencileri üzerinde yaptığı araştırmasında, Scratch ile ilgili oyun etkinliklerine dayalı eğitim programının, öğrencilerin mantıksal/matematikselsel düşünme becerilerine olumlu katkı sağladığını ve bu katkının özellikle matematiksel ilişki faktörü için anlamlı olduğunu ortaya çıkarmıştır. Bunun dışında çoklu zekâ teorisi ve programlama etkinlikleri üzerinde herhangi bir çalışmaya rastlanmamıştır.

5. ARAŞTIRMA YÖNTEMİ VE ÇALIŞMA PLANI

Araştırmanın bu bölümünde araştırmanın modeli, çalışma grubu, çalışma adımları ve planı, veri toplama araçları, verilerin toplanması ve çözümlenmesine yer verilmiştir.

5.1 Araştırmanın Modeli

Araştırmada nicel araştırma yöntemi kullanılmıştır. Nicel araştırma yöntemlerinden bağlantısal model kullanılarak, öğrencilerin çoklu zekâ alanları ve Scratch ile blok tabanlı programlamaya yönelik öz-yeterlilik algıları arasında basit (ikili) korelasyon analizi yapılmıştır. Test yoluyla yapılan araştırmada anket tarzı, Likert ölçekler kullanılmıştır. Ölçeklerden elde edilecek sayısal veriler normal dağılım göstermediği için parametrik olmayan korelasyon tekniği olan Spearman'ın sıralama korelasyon katsayısı hesaplanmıştır (Yücel ve Güneri, 2017).

Bu çalışmada, iki ölçek kullanılmıştır: İbrahim Kasalak (2017) tarafından ortaokul öğrencilerinin programlama becerilerini ölçmeye yönelik geliştirilen "Blok temelli programlamaya ilişkin öz-yeterlilik algısı ölçeği" ile çocukların Scratch etkinliklerine yönelik öz-yeterlilik algıları soruşturulmuştur.

İkinci ölçek, Gülşen (2015) tarafından geliştirilen "Çoklu zekâ alanları değerlendirme ölçeği" olmuştur. Bu ölçek ile öğrencilerin çoklu zekâ alanları belirlenmiştir. Her iki ölçek eşit aralıklı, likert tipi ölçeklerdir.

Bu ölçeklere ek olarak, öğrencilerin Scratch'i ne tür ortamlarda öğrendikleri ve kaçınıcı sınıfta okudukları sorulmuştur. Ancak bu bilgiler değişken olarak araştırmaya katılmamıştır.

Bu verilerin ışığında blok tabanlı programlamaya yönelik öz-yeterlilik algıları ile çoklu zekâ alanları arasında ilişkinin olup olmadığı tespit edilmiştir. Araştırma bulgularının değerlendirilmesinde SPSS 16.0 istatistik programı kullanılmıştır.

5.2 Çalışma Grubu

Çocuklara yönelik Scratch ile programlama eğitimleri Türkiye çapında pek çok okul, özel ve kamu kuruluşları, atölye ve kurslar tarafından gerçekleştirilmektedir. Bazı çocuklar bireysel olarak, internetten, kitaplardan veya eğitimciler ile gerçekleştirdikleri birebir özel dersler ile Scratch etkinliklerini yapmaktadırlar.

Bu çalışmada örneklem: Kolay ulaşılabilir, seçkisiz olmayan örnekleme yönteminden uygun/kazara örneklemedir. Yöntem; zaman, para ve işgücü yönünden ekonomik olma sebebi ile tercih edilmiştir (Şentürk ve diğ., 2020). Araştırmanın çalışma grubunu 2021-2022 eğitim-öğretim yılında Türkiye’de Scratch ile programlama yapmayı bilen 45 ortaokul öğrencileri oluşturmaktadır. Bu öğrenciler, Türkiye’deki ilköğretim okullarında 5, 6, 7 ve 8. sınıfta okumakta; Scratch’i ise okulda, uzaktan eğitimle, atölye/kurs ortamlarında, ebeveynleri ile, kitap ve internet üzerinden bireysel olarak veya eğitmen ile birebir özel dersler aracılığıyla öğrenmişlerdir. Çalışma grubuna uygun olan sınıf seviyesindeki öğrencilerin velilerinin kullandığı sosyal ağlara çalışma ile ilgili envanter gönderilerek gönüllülük esasına göre, anonim olarak ve tamamen rastgele katılım sağlanmıştır.

5.3 Çalışma Planı, Sınırlılık ve Varsayımlar

Araştırma, Mimar Sinan Güzel Sanatlar Üniversitesi Senatosu’nun 24.05.2018 tarih ve 10 sayılı toplantısında alınan 11 numaralı kararına uygun olarak, 26.11.2021 tarihli Etik Kurul Onayı alınarak yapılmıştır.

Ortaokul öğrencilerinin blok tabanlı programlamaya yönelik öz-yeterlilik algıları ve çoklu zekâ alanları arasında korelasyon analizi yapılan araştırma, 2021-2022 öğretim yılında Türkiye’de öğrenim gören ve Scratch isimli blok tabanlı programlama dili ile programlama yapabilen 45 ortaokul öğrencileri ile sınırlıdır. Çalışma grubuna uygun olan sınıf seviyesindeki öğrencilerin velilerine internet üzerinden ulaşılarak öğrenciler çalışmaya davet edilmiştir.

Örneklem, kolay ulaşılabilir, seçkisiz olmayan örnekleme yönteminden uygun/kazara örneklemedir ve araştırmanın evrenini temsilen rahat ulaşabilecek şekilde seçilmiştir. Örneklem büyüklüğünün korelasyon analizi için yeterli olduğu ve kullanılan ölçme yöntemlerinin araştırmaya uygun olduğu uzman görüşü alınarak kabul edilmiştir.

Çalışmaya katılan öğrencilere, toplamda 92 sorudan oluşan “Blok temelli programlamaya ilişkin öz-yeterlilik algısı ölçeği” (Kasalak, 2017) ve “Çoklu zekâ alanları değerlendirme ölçeği” (Gülşen, 2015) envanterleri uygulanmıştır. Bu envanterlerin ortaokul öğrencilerine uygulanabilirliği, envanterleri geliştiren Kasalak (2017) ve

Gülşen (2015) tarafından onaylanmıştır. Çalışmaya katılan öğrencilerin öz-değerlendirmelerini dürüstlikle yaptıkları ve soruları kendileri için doğru olarak cevapladıkları varsayılmıştır.

Araştırmanın verilerinin toplandığı bütün aşamalarda güvenilir ölçümler gerçekleştirilmiştir. Nicel araştırma yöntemlerinden bağlantısal model kullanılarak, öğrencilerin çoklu zekâ alanları ve Scratch ile blok tabanlı programlamaya yönelik öz-yeterlilik algıları arasında basit (ikili) korelasyon analizi yapılmıştır. Ölçeklerden elde edilen sayısal veriler Shapiro-Wilk normallik testi sonucunda normal dağılım göstermediği için parametrik olmayan korelasyon tekniği olan Spearman'ın sıralama katsayısı hesaplanarak korelasyon analizi yapılmış ve sonuçlar tartışılmıştır.

5.4 Veri Toplama Araçları

Bu araştırma için, ortaokulda okumakta olan çocukların programlama becerileri, buna ilişkin öz-yeterlilik algıları kapsamında sorgulanmıştır. Öğrencilerin programlama becerileri üzerine kendilerini değerlendirmeleri için Kasalak (2017) tarafından geliştirilen “Blok temelli programlamaya ilişkin öz-yeterlilik algısı ölçeği” kullanılmıştır. Bu ölçek eşit aralıklı düzenindedir. Ölçekte 12 soru bulunmaktadır, her bir soruya verilen cevaplar “1- Hiç Güvenmiyorum”, “2- Biraz Güveniyorum”, “3-Yarı yarıya güveniyorum”, “4-Oldukça Güveniyorum”, “5- Tamamen Güveniyorum” ile derecelendirilmektedir. Ölçekteki maddelerin Cronbach's Alpha (α) güvenilirlik katsayısı 0.893 olarak bulunmuştur. Bu değere bakılarak ölçekteki maddelerin oldukça güvenilir olduğu belirtilmiştir.

İkinci ölçek, Gülşen (2015) tarafından geliştirilen “Çoklu zekâ alanları değerlendirme ölçeği” olmuştur. Bu ölçek yine çocukların kendilerini değerlendirmeleri üzerine Çoklu Zekâ alanlarının tespiti için kullanılmaktadır. Toplamda 8 zekâ alanı, her bir alan için ise 10 adet soru bulunmaktadır. 8 boyutlu bu ölçekte 10'ar soruluk; Matematiksel/Mantıksal, Sözel/Dilsel, Görsel/Uzamsal, Duyumsal/Müziksel, İçsel/Öze dönük, Sosyal/Kişiler arası, Bedensel/Kinestetik ve Doğasal zekâ değerlendirme ölçekleri bulunmaktadır.

Bu ölçek, eşit aralıklı ölçek düzenindedir. Anket sorularına katılım dereceleri “1-Benim için hiç uygun değil-1.00–1.80”, “2-Benim için çok az uygun-1.81–2.60”, “3-Benim için kısmen uygun-2.61–3.40”, “4-Benim için oldukça uygun-3.41- 4.20”, “5-Benim için tamamen uygun-4.21–5.00.” ağırlıklarıyla sınırlandırılmıştır. Ölçekteki maddelerin Cronbach’s Alpha (α) güvenilirlik katsayısı 0.965 olarak bulunmuştur. Bu değere bakılarak ölçekteki maddelerin oldukça güvenilir olduğu belirtilmiştir.

5.5 Verilerin Toplanması ve Çözülmesi

Veriler SPSS 16.0 istatistik programında çözümlenmiştir. Ölçeklerden alınan veriler 9 değişkene ayrılmıştır.

İlk olarak blok tabanlı programlamaya ilişkin öz-yeterlik algısı ölçeğinden alınan 12 cevabın her çocuk için ortalaması alınmıştır. Ortalaması alınan veriler SPSS programında ilk değişken olarak girilmiştir. Öncelik olarak verilerin normal dağılım gösterip göstermediği incelenmiştir. Örneklem büyüklüğü 50’den az olduğu için Shapiro-Wilk normallik testi yapılmıştır. Bu teste göre verilerin normal dağılım gösterebilmesi için Shapiro-Wilk katsayısı $p > 0.05$ olmalıdır. Daha sonra dağılımların basıklık ve çarpıklık değerlerine bakılmıştır. Basıklık (kurtosis) ve çarpıklık (skewness) katsayıları verilerin normal dağılıma özelliği göstermesi için -1.5 ile 1.5 arasında olmalıdır. En ideal dağılımda bu değerler 0’a yakınsamalıdır (Mishra ve diğ., 2019; Erbay ve Beydoğan, 2017).

Aynı şekilde 80 soruluk çoklu zekâ değerlendirme ölçeğinde her bir zekâ alanına verilen cevaplar 8 boyuta ayrılarak, her 10 maddenin ortalama puanları alınmış, öğrencilerdeki her bir zekâ alanının ortalama değeri böylece değişken olarak kaydedilmiştir. Daha sonra her bir değişkenin normal dağılım gösterip göstermediği incelenmiştir.

Verilerde toplamda 9 değişken bulunmaktadır. Bu değişkenler: Blok temelli programlamaya ilişkin öz-yeterlik algısı ve 8 ayrı zekâ alanıdır. Daha sonra bu değişkenlerin normallik testi sonucunda korelasyon katsayıları hesaplanarak, öğrencilerin blok tabanlı programlamaya yönelik öz-yeterlilik algıları ile çoklu zekâ alanları arasındaki ilişki incelenmiştir.

6. BULGULAR

Araştırmanın bu bölümünde blok temelli programlamayı öğrenme ortamları, blok temelli programlamaya ilişkin öz-yeterlik algısı, çoklu zekâ alanları ve korelasyon analizine yer verilmiştir.

6.1 Blok Tabanlı Programlamayı Öğrenme Ortamları

Öğrencilerin Scratch'i ne tür ortamlarda öğrendikleri sorulmuştur. Birden fazla ortam seçimi imkânı verilmiştir. Ancak bu bilgiler değişken olarak araştırmaya katılmamıştır.

Çizelge 6.1. Scratch'i öğrenme ortamları

Öğrenme ortamı	Öğrenci sayısı	Yüzelik oranı
Okulumdaki sınıf ortamında	17	%28
Birebir eğitimci ile	7	%10.2
Kurs/atölye çalışmaları ile	19	%27.9
Ebeveynim ile evde	2	%2.9
Çevrim içi video eğitimleri ile	4	%5.9
İnternette araştırarak	5	%7.3
Hiçbir yardım almadan kendi kendime	13	%19
Uzaktan eğitim ile	1	%1.4

6.2 Blok Temelli Programlamaya İlişkin Öz-Yeterlik Algısı Ölçeği

Blok temelli programlamaya ilişkin öz-yeterlik algısı (BPÖA) ölçeğinin uygulandığı örneklem büyüklüğü 45 kişidir. Ölçekten elde edilen değerlerin ortalaması alınarak normal dağılım gösterip göstermediği incelenmiştir.

Çizelge 6.2 BPÖA ortalaması

	Aritmetik ortalama	En düşük	En yüksek
BPÖA	4.3118	1.00	5.00

Daha sonra normallik testi Shapiro-Wilk uygulanmıştır.

Çizelge 6.3 BPÖA normallik testi

	Shapiro-Wilk katsayısı
BPÖA	$p = 0.000$

Bu teste göre verilerin normal dağılım göstermediği tespit edilmiştir ($p < 0.05$). Daha sonra basıklık ve çarpıklık değerleri incelenmiştir.

Çizelge 6.4 BPÖA basıklık ve çarpıklık değerleri

	Basıklık	Çarpıklık
BPÖA	-2.242	6.983

Basıklık ve çarpıklık değerlerinin -1.5 ile 1.5 arasında olmaması, verilerin normal dağılmadığını belirtmektedir (Mishra ve diğ., 2019; Erbay ve Beydoğan, 2017). Bu bilgiye göre BPÖA verilerinin normal dağılım yapmadığı tespit edilmiştir.

6.3 Çoklu Zekâ Alanlarını Değerlendirme Ölçeği

Çoklu zekâ alanlarını değerlendirme ölçeğinin uygulandığı örneklem büyüklüğü 45 kişidir. Ölçek 8 boyutludur. Her bir boyuttaki verilerin ortalaması, en düşük ve en yüksek puanları elde edilmiştir.

Çizelge 6.5 Çoklu zekâ alanları ortalaması

Çoklu Zekâ Alanları	Aritmetik ortalama	En düşük	En yüksek
Matematiksel/Mantıksal zekâ	4.0614	2.20	5.00
Sözel/Dilsel zekâ	3.8755	2.00	5.00
Görsel/Uzamsal zekâ	3.9116	2.50	5.00
Duyumsal/Müziksel zekâ	3.5903	1.30	5.00
Bedensel/Kinestetik zekâ	3.6736	1.70	4.90
İçsel/Öze dönük zekâ	3.7728	2.60	4.90
Sosyal/Kişiler arası zekâ	3.8927	2.30	5.00
Doğasal zekâ	3.3674	1.10	5.00

Ölçekten elde edilen değerlerin ortalaması alınarak normal dağılım gösterip göstermediği incelenmiştir. Daha sonra normallik testi Shapiro-Wilk uygulanmıştır.

Çizelge 6.6 Çoklu zekâ alanları normallik testi

Çoklu Zekâ Alanları	Shapiro-Wilk katsayısı
Matematiksel/Mantıksal zekâ	$p = 0.014$
Sözel/Dilsel zekâ	$p = 0.016$
Görsel/Uzamsal zekâ	$p = 0.410$
Duyumsal/Müziksel zekâ	$p = 0.013$
Bedensel/Kinestetik zekâ	$p = 0.018$
İçsel/Öze dönük zekâ	$p = 0.174$
Sosyal/Kişiler arası zekâ	$p = 0.007$
Doğacı zekâ	$p = 0.528$

Sonuçlara göre verilerin çoğunlukla normal dağılım göstermediği tespit edilmiştir ($p < 0.05$). Daha sonra basıklık ve çarpıklık değerleri incelenmiştir.

Çizelge 6.7 Çoklu zekâ alanları basıklık ve çarpıklık değerleri

Çoklu Zekâ Alanları	Basıklık	Çarpıklık
Matematiksel/Mantıksal zekâ	-0.845	0.179
Sözel/Dilsel zekâ	-0.715	-0.303
Görsel/Uzamsal zekâ	-0.263	-0.49
Duyumsal/Müziksel zekâ	-0.499	-0.709
Bedensel/Kinestetik zekâ	-0.75	0.227
İçsel/Öze dönük zekâ	-0.007	-0.945
Sosyal/Kişiler arası zekâ	-0.484	-1.016
Doğasal zekâ	-0.277	-0.353

Çoklu zekâ alanlarının basıklık ve çarpıklık değerleri çoğunlukla -1.5 ile +1.5 arasında olsa da, Shapiro-Wilk katsayısına göre 8 verinin 5'i normal dağılım göstermemektedir.

6.4 Korelasyon Analizi

BPÖA ölçeğinden alınan verilerde Shapiro-Wilk katsayısının 0.05'ten küçük olması ($p = 0.000$), basıklık (-2.242) ve çarpıklık (6.938) değerlerinin -1.5 ve +1.5 sınırları dışında olması sebebiyle, BPÖA ve Çoklu Zekâ alanları arasında Spearman'ın sıralama korelasyon katsayısı hesaplanarak ilişki aranmıştır.

Çizelge 6.8 BPÖA ve Çoklu Zekâ Alanları Korelasyon Analizi

Spearman'ın sıralama korelasyon katsayısı		Blok temelli programlamaya ilişkin öz-yeterlik algısı (BPÖA)
Matematiksel/Mantıksal zekâ	Korelasyon Katsayısı	0.568
	Anlamlılık düzeyi (2-yönlü)	0.000
Sözel/Dilsel zekâ	Korelasyon Katsayısı	0.391
	Anlamlılık düzeyi (2-yönlü)	0.008
Görsel/Uzamsal zekâ	Korelasyon Katsayısı	0.323
	Anlamlılık düzeyi (2-yönlü)	0.030

İçsel/Öze dönük zekâ	Korelasyon Katsayısı	0.514
	Anlamlılık düzeyi (2-yönlü)	0.000
Sosyal/Kişisel arası zekâ	Korelasyon Katsayısı	0.147
	Anlamlılık düzeyi (2-yönlü)	0.342
Duyumsal/Müziksel zekâ	Korelasyon Katsayısı	0.192
	Anlamlılık düzeyi (2-yönlü)	0.205
Bedensel/Kinestetik zekâ	Korelasyon Katsayısı	0.106
	Anlamlılık düzeyi (2-yönlü)	0.495
Doğasal zekâ	Korelasyon Katsayısı	0.085
	Anlamlılık düzeyi (2-yönlü)	0.583

Çizelge 6.8 BPÖA ve Çoklu Zekâ Alanları Korelasyon Analizi

Bu verilere göre:

BPÖA ile matematiksel/mantıksal zekâ arasında ($r=0.568$, $p=0.000$) pozitif ve anlamlı, Sözel/Dilsel zekâ arasında ($r=0.391$, $p=0.008$) pozitif ve anlamlı, Görsel/Uzamsal zekâ arasında ($r=0.323$, $p=0.030$) pozitif ve anlamlı, İçsel/Öze dönük zekâ arasında ($r=0.514$, $p=0.000$) pozitif ve anlamlı ilişkiler bulunmuştur.

7. TARTIŞMA, SONUÇ VE ÖNERİLER

Araştırmanın bu bölümünde çalışma sonucunda elde edilen bulgular doğrultusunda ulaşılan sonuçlar alan yazındaki benzer araştırmalar ile tartışılmış ve sonraki araştırmacılar için önerilerde bulunulmuştur.

7.1 Tartışma ve Sonuç

Bu araştırmada Scratch ile blok tabanlı görsel programlama etkinliklerine katılan ortaokul öğrencilerinin çoklu zekâ alanlarının ve blok tabanlı programlamaya yönelik öz-yeterlilik algılarının (BPÖA) tespiti yapılmış, bunlar arasında istatistiksel açıdan anlamlı bir ilişkinin olup olmadığı incelenmiştir.

Hipoteze göre çoklu zekâ alanlarının bazıları ile BPÖA arasında anlamlı bir ilişki vardır, bulgular ise hipotezi desteklemektedir. Gardner (1983)'e göre bir bilgisayar programlamayı öğrenme, çoklu zekâ alanları ile ilgili olabilir. Ninrutsirikun ve diğ., (2016) de araştırmalarında üniversite öğrencilerinin bilgisayar programlama dersinde gösterdikleri başarı ve bu öğrencilerin çoklu zekâ alanları arasında bir korelasyon bulmuş olup çoklu zekâ kuramının öğrencilerin programlama derslerindeki performanslarının tahmini için kullanılabileceği ortaya koymuştur.

Yapılan korelasyon analizine göre BPÖA ile matematiksel/mantıksal zekâ ($r=0.568$, $p=0.000$) arasında pozitif yönde ve anlamlı bir ilişki bulunmuştur. Bu sonuç Alt Hipotez 1'i doğrulamaktadır. Matematiksel/mantıksal zekâ diğer zekâ alanları içinde BPÖA ile en yüksek ilişki düzeyine sahiptir. Mantıksal/matematiksel düşünme ile algoritma oluşturmanın, bilgi işlemsel düşünme, programlama ve bilgisayar biliminin temelini oluşturması (Varela ve diğ., 2019) temel alındığında BPÖA ile matematiksel/mantıksal zekâ arasındaki pozitif yönde ve anlamlı ilişki beklenen bir sonuçtur. Korkmaz (2016) araştırmasında, Scratch ile ilgili oyun etkinliklerine dayalı eğitim programının, öğrencilerin mantıksal/matematiksel düşünme becerilerine olumlu katkı sağladığını ortaya çıkarmıştır. Saygıner (2017), Scratch öğretimi sonucunda, öğrencilerin mantıksal düşünme puanlarında istatistiksel olarak anlamlı bir artış ve yüksek düzeyde bir etki büyüklüğü hesaplandığını bildirmiştir. Bu durumda, matematiksel/mantıksal zekânın, Scratch ile blok tabanlı programlama etkinlikleri ile doğrudan ilişkili olduğu söylenebilir. Çocukların Scratch ile algoritmik düşünme becerilerini geliştirmeleri (Yünkül ve diğ., 2017; Oluk ve diğ., 2018) sebebiyle BPÖA mantıksal düşünme becerisi ile dolaylı olarak ilişkilendirilir. Çünkü Varela ve diğ. (2019), genellikle algoritmik düşünmenin mantıksal/matematiksel düşünme olarak ifade edildiğini çünkü çoklu zekâ üzerine yapılan çalışmalarda, mantıksal/matematiksel düşünmenin, örüntüleri tanıma, mantık ve tümdengelimli akıl yürütme ile bağlantılı olduğunu açıklamışlardır. Ayrıca Gardner (1983), mantıksal/matematiksel zekâyı, programlama becerisi için merkezde görmekte, çünkü programlamanın, bir sorunu çözmek veya sınırlı sayıda adımda bir hedefe ulaşmak için, adımların açık, kesin ve kesinlikle mantıklı bir sırada organize edilmesini gibi katı prosedürlerin uygulanmasına bağlı olduğunu açıklamıştır. Buna ek olarak,

Sebetci ve Aksu (2014) öğrencilerin analitik ve mantıksal düşünme becerilerini gelişmesiyle, öğrencilerin programlamada daha başarılı olacakları ifade etmiştir.

Ancak programlama eğitimlerinin problem çözme becerileri üzerinde anlamlı bir etkisi olmadığı, Bala ve Alacapınar (2021), Kalelioğlu ve Gülbahar (2014), Psycharis ve Kallia (2017) ve Çiftçi ve Bildiren (2019) tarafından ortaya konmuştur. Buna karşıt, programlama; algoritma oluşturmak, mantıksal düşünme ve problem çözme stratejilerini de içerir (Tekerek ve Altan, 2014). Mantıksal düşünme becerileri ile problem çözme becerileri de ilişkilendirilmektedir (Akkus ve diğ., 2019; Swanson ve diğ., 2015; Oluk ve Çakır, 2019). Hepsi göz önüne alındığında; önceki araştırmacılar tarafından ortaya konan programlama eğitiminin problem çözme becerisi üzerindeki etkisizliğı, araştırmacının sonucunu desteklememiştir.

BPÖA ile içsel/öze dönük zekâ arasında pozitif yönde ve anlamlı bir ilişki ($r=0.514$, $p=0.000$) bulunmuştur. İçsel/öze dönük zekâ diğeri zekâ alanları içinde BPÖA ile en yüksek ikincil ilişki düzeyine sahiptir. Gardner (1983) kitabında, içsel zekânın programlama öğreniminde önemli bir rol oynayabileceğinden bahsetmiştir. Yazara göre, programlama yaparken adım ve hedeflerin hesaplanması, içsel düşünme becerilerine dayanmaktadır. İçsel zekâ ve BPÖA arasındaki pozitif yönde ve anlamlı ilişkinin, içsel zekânın sosyallik gerektirmeyen bir zekâ türü olması ve bilgisayarla geçirilen vakitlerin genellikle bireysel olması ile alakalı olduğu söylenebilir. Ancak örnekleme katılımcıların %55.9'u sosyal ortamlarda, %44.1'i bireysel veya eğitici ile ikili ortamlarda Scratch'i öğrenmişlerdir. Ancak, İşbulan ve Arslan (2021) yaptıkları araştırmada, bireysel ve ekip çalışması olarak iki farklı şekilde yapılan Scratch eğitimleri sonucunda, ortaokul öğrencilerinin blok tabanlı programlamaya yönelik öz-yeterlilik algılarında ve bu etkinliklere yönelik tutumlarında farklılık oluşturmadığını ortaya koymuşlardır. İleride yapılacak araştırmalarda, bireysel ve ekip çalışmaları ile yapılan programlama eğitimlerinin içsel ve sosyal zekâ ile ilişkisinin incelenmesi önerilmiştir.

BPÖA ile sözel/dilsel zekâ ile pozitif yönde ve anlamlı bir ilişki ($r=0.391$, $p=0.008$) bulunmuştur. Bu zekâ türünü programlama becerisi ile ilişkilendiren çalışma, Prat ve diğ., (2020) tarafından metin tabanlı programlama diline yönelik yaptıkları araştırma olmuştur. Metin tabanlı programlamayı öğrenmede, dil öğrenme becerisinin,

programlamayı öğrenmek için, temel matematik bilgisinden veya aritmetikten daha güçlü bir kolaylaştırıcı olduğunu ortaya koymuştur. Bunun nedeni, kod yazmanın aynı zamanda ikinci bir dil öğrenmeyi, o dilin kelime ve gramerini öğrenme becerisini ve fikirleri ve niyetleri iletmek için birlikte nasıl çalıştıklarını da içermesidir. Alan yazında blok tabanlı programlama becerisi sözel zekâ ile ilişkilendirilmese de öğrencilerin hazır kod bloklarını okuyup anlamaları, kod bloklarından algoritmaya uygun bir “mantıksal hikâye” oluşturmaları, Gardner’ın (1983) teorisi ile örtüşmektedir. Yazara göre, bilgisayar dilleri sıradan dili kullandığı sürece, dilsel zekâ da önemlidir. Programı bir hikâye olarak değerlendirdiğimizde, dilsel eğilimi olanlar için programlamayı öğrenmek kolay olabilir. Ayrıca, Papatğa (2016) ilkokul öğrencileriyle Scratch ile hazırladıkları projeler sonucunda, okuduğunu anlama sorunu yaşayan öğrencilerin okuduğunu anlama becerilerinin geliştiğini gözlemlemiştir. Ek olarak, Bahar (2021), Scratch’te programlama aktivitelerinin, çocukların İngilizce dinleme becerileri üzerinde önemli bir etkiye sahip olduğunu bildirmiştir. Bu sonuçlar sözel/dilsel zekâ ile BPÖA ilişkisini destekler niteliktedir.

BPÖA ile görsel/uzamsal zekâ ($r=0.323$, $p=0.030$) ile pozitif yönde ve anlamlı bir ilişki olarak bulunmuştur. Bulgu Alt Hipotez 3’ü desteklemektedir. Ökten ve Atasoy (2020), Scratch ile programlama yapan öğrencilerin uzamsal akıl yürütme becerilerinin programlama öğrenmemiş olan öğrencilere göre anlamlı düzeyde farklı olduğu; öğrencilerin görsel-uzamsal akıl yürütme becerilerini geliştirmede programlamanın önemli bir faktör olabileceği ifade edilmiştir. Messer ve diğ. (2018) okul öncesi çocuklar üzerinde gerçekleştirilen bir araştırmada, blok tabanlı programlama uygulamasını kullanmış öğrencilerde uzamsal farkındalıkların arttığını ortaya koymuşlardır. Gardner’a (1983) göre, güçlü uzamsal algıya sahip bir kişi, bir tür bilgisayar grafiği aracılığıyla programlama öğrenmeye başlatılabilir ve aynı zamanda, bir akış şeması veya başka bir uzamsal diyagramın kullanılması yoluyla programlama görevinde yardımcı olabilir. Buna ek olarak, bir bilgisayarı komuta etmek için bilgi işlemsel düşünme kullanarak sorunları ayrıntılı analiz etmeli, nasıl parçalanacağını planlamalı ve bilgisayar dostu çözümler aramalıdır. Bu inceleme genellikle diyagramlarla ifade edilen süreçleri ve verileri içerir (Shyamasundar, 1996). Scratch’in görsel blok tabanlı programlama yapısı, diyagram ve akış şemalarına benzer şekilde

algoritma oluşturmaya sağlamaktadır. Buna göre görsel zekânın, Scratch ile yapılan programlama etkinliklerinde kolaylaştırıcı etkisi olabilir.

BPÖA ile doğasal, sosyal, müziksel, bedensel zekâ arasında anlamlı bir ilişki bulunamamıştır. Alt Hipotez 2, -BPÖA ile bedensel zekâ arasında negatif yönde anlamlı ilişki- doğrulanamamıştır. Gardner (1983) teorisinde, müziksel, sosyal ve bedensel zekânın programlama becerisiyle ilişkili olabileceğinden bahsetmiştir. Özkandemir (2019), ilkokul müzik derslerinde Scratch ve Makey Makey kiti kullanarak yapılan nota okuma yazma çalışmaları sonucunda öğrencilerin nota okuma yazma becerilerinde geliştikleri gözlenmiştir. Ancak müziksel zekâ ve BPÖA arasındaki bir ilişki bahsedilmemektedir. Son olarak, doğasal zekâ ve programlama becerileri arasında herhangi bir çalışmaya rastlanmamıştır.

Araştırmanın sonucunda olarak, blok temelli programlamada performans ve BPÖA arasındaki doğrusal ilişkiye (Kasalak, 2017) dayanarak, mantıksal/matematikselsel, içsel/öze dönük, sözel/dilsel, görsel/uzamsal zekâ türlerinden herhangi birisi, ikisi, üçü veya dördü baskın olan ortaokul öğrencilerinin blok temelli programlamayı öğrenmede ve uygulamada başarılı olabilecekleri söylenebilir.

Her insanın tabiatı ve bilişsel becerileri farklıdır. Öğretim programlarının herkes için geçerli ve standart olması insan tabiatına aykırıdır. Bu yüzden öğretim müfredatı, öğrencilerin çeşitliliği ve esnekliğine uyum sağlamalıdır (Millî Eğitim Bakanlığı, 2018). Birbirinden ayrı çeşitli bilişsel becerilere sahip öğrencilerin bu becerileri, alacakları eğitimler öncesinde tespit edilerek, öğrencilerin ilgili alan ve derslere yönlendirilmesi çoklu zekâ alanlarının tespiti mümkün olabilir (Gardner, 1983). Araştırmanın sonucuna göre ilişkili zekâ alanlarına sahip öğrencilerin ebeveynlerine blok tabanlı programlama eğitimlerini değerlendirmeleri önerilir.

7.2 Öneriler

Araştırmanın sonucunda göre sonraki araştırmacılara aşağıdaki öneriler sunulmuştur.

- Sosyalleşmeye imkân sağlayan bir sınıf ortamında veya bireysel/yalnız olarak yapılan programlama eğitimlerinin içsel ve sosyal zekâ ile ilişkisi daha sonraki araştırmacılar için bir araştırma konusu olabilir.

- Yaş grubu değiştirilerek, cinsiyet farklılıkları gözetilerek, örneklem büyüklüğü arttırılarak BPÖA ile çoklu zekâ alanları arasında ilişki tekrar incelenebilir.
- Halihazırda programlama, yazılım ve bilgisayar mühendisliği alanında çalışan yetişkinlerin çoklu zekâ alanları araştırılması önerilir.
- Bireylerin çocukluk ve yetişkinlikteki bilişsel becerilerinin ilişkisi, yetişkinlikte edinilen mesleğin bilişsel becerilerle ilişkisi incelenebilir.
- Programlama becerisi ile bilişsel beceriler arasındaki ilişkinin farklı teknikler kullanılarak ölçülmesi önerilir.
- Çoklu zekâ alanları teorisi ile programlama becerisi arasındaki ilişkinin farklı yöntem ve metotlarla araştırılması önerilir.
- Blok tabanlı programlamaya yönelik öz-yeterlilik algısı ile, bu alandaki akademik başarı veya performansın çoklu zekâ alanlarıyla ilişkisinin araştırılması önerilir.
- Problem çözme becerileri ile bilgi işlemsel düşünme, programlama becerisi ve zekâ alanları arasındaki ilişki incelenebilir.
- Bilgi işlemsel düşünme becerileri ile çoklu zekâ alanları arasındaki ilişki incelenebilir.
- Metin tabanlı programlama becerisi ve çoklu zekâ alanları üzerinde araştırmaların yapılması önerilir.
- Araştırmanın normal dağılım gösteren veriler üzerinde tekrarlanması önerilir.

8. KAYNAKLAR

- Adsay, C., Korkmaz, Ö., Çakır, R. & Uğur Erdoğan, F. (2020). *Ortaokul Öğrencilerinin Blok Temelli Kodlama Eğitimine Dönük Öz-yeterlik Algı Düzeyleri, STEM ve Bilgisayarca Düşünme Beceri Düzeyleri*. Eğitim Teknolojisi Kuram ve Uygulama, 10 (2), 469-489. DOI: 10.17943/etku.696224
- Akkus, I., Özhan, U. & Kan, Ad. (2019). *Ortaokul Öğrencileri için Kodlamaya Yönelik Tutum Ölçeği: Geçerlik ve Güvenirlik Çalışması*. İlköğretim Online. 18. 10.17051/ilkonline.2019.562064.
- Akpınar, Y., Altun, A. (2014). *Bilgi toplumu okullarında programlama eğitimi gereksinimi*. İlköğretim Online, 13(1). Erişim adresi: <http://ilkogretim-online.org.tr/index.php/io/article/view/2099/1935> Erişim tarihi: 28/12/2021.
- Albayrakoğlu, Ö. (Kişisel iletişim, 1 Ekim 2021).
- Arseven, A. (2016). *Öz Yeterlilik: Bir Kavram Analizi*. Journal of Turkish Studies. 11. 63-63. 10.7827/TurkishStudies.10001.
- Arslan, K., Akçelik, M. (2019). *Programlama eğitiminde Scratch'in kullanılması: Öğretmen adaylarının tutum ve algıları*. Ulusal Eğitim Akademisi Dergisi (UEAD), 3(1), 41-61.
- Arslan, E., 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, v9 n1 p108-121.
- Asad, K., Tibi, M. & Raiyn, J. (2016). *Primary School Pupils' Attitudes toward Learning Programming through Visual Interactive Environments*. World Journal of Education. 6. 10.5430/wje.v6n5p20.
- Babacan, T. Dilci, T. 2012. *Çoklu zekâ ölçeğinin türkçeye uyarlama çalışmaları*. E-Journal of New World Sciences Academy2012, Volume: 7, Number: 3, Article Number: 1C0557

- Bahar, N. (2021). *The effect of scratch on children's english language and cognitive development*. (Yayınlanmış yüksek lisans tezi). Ortadoğu Teknik Üniversitesi, Ankara.
- Bala, R. B. Alacapınar, F. G. (2021). *Scratch In Teaching Programming: Effect On Problem Solving Skill And Attitude*. International Journal of Quality in Education, 5 (2), 63-81. Erişim adresi: <https://dergipark.org.tr/tr/pub/ijqe/issue/62879/954582> Erişim tarihi: 28/12/2021.
- Balanskat, A., Engelhardt, K. (2015). *Computing our future: Computer programming and coding, Priorities, school curricula and initiatives across Europe*. European Schoolnet. Erişim adresi: <http://www.eun.org/documents/411753/817341/Computing+our+future+final+2015.pdf/d3780a64-1081-4488-8549-6033200e3c03>. Erişim tarihi: 28/12/2021.
- Birleşik Krallık Hükümeti. (2014). Erişim adresi: <https://www.gov.uk/government/publications/d5-london-summit-themes/d5-london-teaching-children-to-code>. Erişim tarihi: 20/12/2021.
- Blakemore, L. (2017). *Does teaching computer programming within Key Stage 1 of the primary curriculum enhance children's problem solving skills?* EdD thesis, University of Sheffield.
- Boom, K.D., Bower, M., Arguel, A. Siemon, J. & Scholkmann, A. (2018). *Relationship between computational thinking and a measure of intelligence as a general problem-solving ability*. 206-211. 10.1145/3197091.3197104.
- Brennan, K., Resnick, M. (2012). *New frameworks for studying and assessing the development of computational thinking*. Proceedings of the 2012 Annual meeting of the American educational research association, Vancouver, Canada 1–25.
- Calder, N. (2018). *Using Scratch: An Integrated Problem-solving Approach to Mathematical Thinking*. APMC 15 (4), 9-14.

- Ceylan, V. K. (2020). *Senaryo temelli Scratch öğretim programının öğrencilerin bilgi işlemsel düşünme becerilerine, problem çözme ve programlama ünitesi erişilerine etkisi*. (Yayınlanmamış doktora tezi). Aydın Adnan Menderes Üniversitesi, Sosyal Bilimler Enstitüsü, Aydın.
- Cirit, G. A. (2018). *Ortaokul Öğrencilerinin Çoklu Zekâ Alanları ile Bilgisayara Yönelik Tutumları Arasındaki İlişkinin İncelenmesi*. 9. 155-164.
- Çakır, B. (2020). *Ortaokullarda kodlama eğitiminin öğrencilerin yaratıcı problem çözme ve bilişüstü farkındalıklarına etkisi*. (Yayınlanmamış yüksek lisans tezi). Aydın Adnan Menderes Üniversitesi, Sosyal Bilimler Enstitüsü, Aydın.
- Çatlak, Ş., Tekdal, M. & Baz, F. (2015). *Scratch yazılımı ile programlama öğretiminin durumu: bir doküman inceleme çalışması*. Journal of Instructional Technologies and Teacher Education, 4 (3).
- Çetin, E. (2012). *Bilgisayar programlama eğitiminin çocukların problem çözme becerileri üzerine etkisi*. (Yayınlanmamış yüksek lisans tezi). Gazi Üniversitesi, Eğitim Bilimleri Enstitüsü, Ankara
- Çiftçi, S., Bildiren, A. (2019). *The effect of coding courses on the cognitive abilities and problem-solving skills of preschool children*, Computer Science Education, DOI: 10.1080/08993408.2019.169616
- Çoklar, A. N., Akçay, A. (2018). *Evaluating programming self-efficacy in the context of inquiry skills and problem-solving skills: A perspective from teacher education*. World Journal on Educational Technology: Current Issues. 10(3), 153-164.
- Çubukluöz, Ö. (2019). *6. sınıf öğrencilerinin matematik dersindeki öğrenme zorluklarının Scratch programıyla tasarlanan matematiksel oyunlarla giderilmesi: bir eylem araştırması*. (Yayınlanmamış yüksek lisans tezi). Bartın Üniversitesi, Eğitim Bilimleri Enstitüsü, Bartın.
- Demirer, V., Sak, N. (2016). *Programming education and new approaches around the world and in Turkey*. Journal of Theory and Practice in Education. 12. 521-546.
- Denning, P., Comer, D. E., Gries, D., Mulder, M. C., Tucker, A. B., Turner, A. J., & Young, P. R. (1988). *Computing as a discipline: preliminary report of the ACM task*

- force on the core of computer science*. Proceedings of the Nineteenth SIGCSE Technical Symposium on Computer Science Education - SIGCSE '88. doi:10.1145/52964.52975
- Durak, H. Y., Yilmaz, F. G. K. & Yilmaz, R. (2019). *Computational Thinking, Programming Self-Efficacy, Problem Solving and Experiences in the Programming Process Conducted with Robotic Activities*. Contemporary Educational Technology, 10 (2), 173-197. DOI: 10.30935/cet.554493
- Erbay, Ş., Beydoğan, H. Ö. (2017). *Eğitimcilerin eğitim araştırmalarına yönelik tutumları*. Ahi Evran Üniversitesi Kırşehir Eğitim Fakültesi Dergisi, 18 (3), 246-260.
- Ersoy, H., Madran, R. O. & Gülbahar, Y. (2011). *Programlama dilleri öğretimine bir model önerisi: robot programlama*. Akademik Bilişim'11 Konferansı, Malatya.
- Gardner, H. (1983). *Frames of Mind: The Theory of Multiple Intelligences*. NYC: Basic Books.
- Gardner, H. (1999). *Intelligence reframed: multiple intelligences for the 21st century*. NYC: Basic Books.
- Günbatar, M. S. (2018). *Examination of Undergraduate and Associate Degree Students' Computer Programming Attitude and Self-Efficacy According to Thinking Style, Gender and Experience*. Contemporary Educational Technology , 9 (4) , 354-373 . DOI: 10.30935/cet.471004
- Gülşen, C. 2015. *Çoklu zekâ alanları değerlendirme ölçeği geliştirilmesi çalışması*. International Journal of Human Sciences, 12(2), 1918-1930.
- Hershkovitz, A., Sitman, R., Israel-Fishelson, R., Eguíluz, A., Garaizar, P. & Guenaga, M. (2019). *Creativity in the acquisition of computational thinking*. Interactive Learning Environments. 27. 1-17. 10.1080/10494820.2019.1610451.
- Hinsliff, G. (2015, 3 Aralık). *Should kids learn to code?* The Guardian Gazetesi. Erişim adresi <https://www.theguardian.com/news/2015/dec/03/should-kids-learn-code> Erişim tarihi: 21/12/2021

- Hysong, S., Quinones, M. (1997). *The Relationship Between Self-Efficacy and Performance: A Meta-Analysis*. 10.13140/2.1.1453.8880.
- ISTE. (2016). *The ISTE National Educational Technology Standards (NETS•S) and Performance Indicators for Students*. Erişim adresi: <http://www.iste.org/standards/standards/for-students-2016>, Erişim tarihi: 28/12/21
- Kalelioglu, F., Gulbahar, Y. (2014). *The Effects of Teaching Programming via Scratch on Problem Solving Skills: A Discussion from Learners' Perspective*. *Informatics in Education*. 13. 33-50.
- Kalelioglu, F., Gulbahar, Y. & Kert B., S. (2020). *Teaching computing at secondary school level: Understanding teachers' experiences and challenges*. *İlköğretim Online (elektronik)*, 19(3), 1781- 1796.
- Kasalak, İ. (2017). *Robotik kodlama etkinliklerinin ortaokul öğrencilerinin kodlamaya ilişkin öz-yeterlik algılarına etkisi ve etkinliklere ilişkin öğrenci yaşantıları*. (Yayınlanmış yüksek lisans tezi). Hacettepe Üniversitesi, Eğitim Bilimleri Enstitüsü, Ankara.
- Kert, S., Erkoç, M. & Yeni, S. (2020). *The Effect of Robotics on Six Graders' Academic Achievement, Computational Thinking Skills and Conceptual Knowledge Levels*. *Thinking Skills and Creativity*. 38. 100714. 10.1016/j.tsc.2020.100714.
- Kesici, A. (2016). *Matematik bölümü öğretmen adaylarının çoklu zekâ alanlarının belirlenmesi ve sosyal becerileri ile öğretmenlik mesleğine ilişkin tutumlarının incelenmesi*. 10.13140/RG.2.2.24542.89921.
- Konyaoğlu, C. (2019). *Robotik Kodlama Eğitiminin Ortaokul Öğrencilerinin Problem Çözme Becerilerine Etkileri ve Öğrencilerin Robotik Kodlama Etkinliklerine İlişkin Görüşleri*. (Yayınlanmamış yüksek lisans tezi). Hacettepe Üniversitesi Eğitim Bilimleri Enstitüsü, Ankara.
- Korkmaz, Ö. (2016). *The Effect of Scratch- and Lego Mindstorms Ev3-Based Programming Activities on Academic Achievement, Problem-Solving Skills and Logical-Mathematical Thinking Skills of Students*. *Malaysian Online Journal of Educational Sciences* 2016 (Volume4 - Issue 3).

- Korucu, A. T., Ünüvar, M. (2020). *Bilgisayar ve Öğretim Teknolojileri Öğretmenliği Bölümünde Okuyan Öğretmen Adaylarının Aldıkları Eğitim ile Bilişim Liderliği Vasfı ve 21. Yüzyıl Becerileri Arasındaki İlişki*. Bilgi ve İletişim Teknolojileri Dergisi , 2 (1) , 44-53.
- Kuru, E. (2001). *Kinestetik Zekâ ve Beden Eğitimi*. G.Ü. Gazi Eğitim Fakültesi Dergisi Cilt 21, Sayı 2 (2001) 217-229.
- Messer, D., Thomas, L. & Holliman, A. (2018). *Evaluating the effectiveness of an educational programming intervention on children's mathematics skills, spatial awareness and working memory*. Educ Inf Technol 23, 2879–2888. <https://doi.org/10.1007/s10639-018-9747-x>
- Millî Eğitim Bakanlığı. (2018). *Bilişim teknolojileri ve yazılım dersi öğretim programı*. Ankara.
- Mishra, P., Pandey, C. M., Singh, U., Gupta, A., Sahu, C. & Keshri, A. (2019). *Descriptive statistics and normality tests for statistical data*. Annals of cardiac anaesthesia, 22(1), 67–72. https://doi.org/10.4103/aca.ACA_157_18
- Moursund, D. (2006). *Computational Thinking and Math Maturity: Improving Math Education in K-8 Schools*, University of Oregon, Eugene.
- Muhammad, M. H., Ahmad, A., & Zulhazlin, A. (2018). *Computer Self-Efficacy in Programming Language for Non-Technical Students using Scratch Programming*. Indian Journal of Pure and Applied Mathematics. 118. 381-3388.
- Newell, A., Perlis, A. & Simon, H. (1967). *What is computer science?* Science, 157, 1373-4.
- Ninrutsirikun, U., Watanapa, B., Arpnikanondt, C. & Phothikit, N. (2016). *Effect of the Multiple Intelligences in multiclass predictive model of computer programming course achievement*. IEEE Region 10 Conference (TENCON) Region 10 Conference (TENCON) :297-300.
- Okuducu, A. (2020). *Scratch Destekli Matematik Öğretiminin 6. Sınıf Öğrencilerinin Cebirsel İfadeler Konusundaki Akademik Başarılarına ve Tutumlarına Etkisi*.

(Yayınlanmamış yüksek lisans tezi). Ağrı İbrahim Çeçen Üniversitesi Fen Bilimleri Enstitüsü, Ağrı.

Oluk A., Çakır, R. (2019). *Öğrencilerin bilgisayarca düşünme becerilerinin mantıksal matematiksel zekâ ve matematik akademik başarıları açısından incelenmesi*. Kuramsal Eğitimbilim Dergisi, 12(2), 457-473.

Oluk, A., Korkmaz, Ö. & Oluk, H.A. (2018). *Scratch'ın 5. Sınıf Öğrencilerinin Algoritma Geliştirme ve Bilgi-İşlemsel Düşünme Becerilerine Etkisi*. Turkish Journal of Computer and Mathematics Education, 9(1), 54-71.

Otu, T. (2020). *Kodlama ortamlarının ortaokul öğrencilerinin başarı, tutum ve bilgi işlemsel düşünme becerisine etkisi*. (Yayınlanmamış yüksek lisans tezi). Bolu Abant İzzet Baysal Üniversitesi / Lisansüstü Eğitim Enstitüsü, Bolu.

Özdağ, G. (2019). *6. Sınıf öğrencilerinin Scratch ile süsleme etkinliklerinde matematiksel estetik rollerinin incelenmesi*. (Yayınlanmamış yüksek lisans tezi). Mersin Üniversitesi, Eğitim Bilimleri Enstitüsü, Mersin.

Özden, C., Atasoy, R. (2020). *The Effect of Coding on Students' Visual-spatial Reasoning Skills*. Croatian Journal of Education - Hrvatski časopis za odgoj i obrazovanje. 22. 10.15516/cje.v22i3.3522.

Özkandemir, O. (2019). *İlkokul müzik derslerinde robotik ve kodlama programlarının kullanılmasına yönelik örnek bir çalışma*. (Yayınlanmamış yüksek lisans tezi). Marmara Üniversitesi, Eğitim Bilimleri Enstitüsü, İstanbul.

Papatğa, E. (2016). *Okuduğunu Anlama Becerilerinin Scratch Programı Aracılığıyla Geliştirilmesi Developing Reading Comprehension Skills Through Scratch Program*. (Yayınlanmamış yüksek lisans tezi). Anadolu Üniversitesi, Eskişehir.

Prat, C.S., Madhyastha, T.M. & Mottarella, M.J. (2020). *Relating Natural Language Aptitude to Individual Differences in Learning Programming Languages*. Sci Rep 10, 3817. <https://doi.org/10.1038/s41598-020-60661-8>

Psycharis, S. Kallia, M. (2017). *The effects of computer programming on high school students' reasoning skills and mathematical self-efficacy and problem solving*. Instructional Science. doi:10.1007/s11251-017-9421-5

- Ramalingam, V., Labelle, D., & Wiedenbeck, S. (2004). *Self-efficacy and mental models in learning to program*. ACM Sigcse Bulletin. 36. 171-175. 10.1145/1007996.1008042.
- Resnick, M., Maloney, J., Hernández, A. M., Rusk, N., Eastmond, E., Brennan, K., Millner, A. D., Rosenbaum, E., Silver, J., Silverman, B., & Kafai, Y.B. (2009). *Scratch: Programming for all*. Communications of the ACM, 52(11), 60–67.
- Rich, P.J., Browning, S.F. & Perkins, M. (2019). *Coding in K-8: International Trends in Teaching Elementary/Primary Computing*. TechTrends 63, 311–329. <https://doi.org/10.1007/s11528-018-0295-4>
- Rosiña, E., Bermejo, M., Del Barco, M. A., Cañada Cañada, F. & Sánchez-Martín, J. (2020). *Multiple Intelligences Analysis and Emotional Implications in STEM Education for Students up to K-12*. 10.4018/978-1-7998-0249-5.ch013.
- Sade, A. (2020). *Kodlama öğretiminin 6. sınıf öğrencilerinin bilgisayarca düşünme becerilerine, matematik kaygı algılarına ve problem çözme algılarına etkisi*. (Yayınlanmamış yüksek lisans tezi). Mersin Üniversitesi, Eğitim Bilimleri Enstitüsü Teknolojileri Eğitimi Ana Bilim Dalı, Mersin.
- Sáez-López, JM., Sevillano-García, ML. & Vazquez-Cano, E. (2019). *The effect of programming on primary school students' mathematical and scientific understanding: educational use of mBot*. Education Tech Research Dev 67, 1405–1425 <https://doi.org/10.1007/s11423-019-09648-5>
- Sağay, M. A. (2019). *İlköğretim öğrencileri için oyunlaştırma yöntemiyle kodlama eğitimi uygulaması*. (Yayınlanmamış yüksek lisans tezi). İstanbul Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- Sanz, A. (2015). *Why Teaching and Learning How to Code in Schools*. EdTech Review. Erişim adresi: <http://edtechreview.in/trends-insights/insights/1934-why-teaching-and-learning-how-to-code-in-schools> Erişim tarihi: 28/12/21.
- Saygıner, Ş. (2017). *Blok tabanlı görsel ve metin tabanlı programlama öğretimlerinin erişimi, mantıksal düşünme ve motivasyona etkileri*. (Yayınlanmamış yüksek lisans tezi). Hacettepe Üniversitesi, Ankara.

- Sayın, Z., Seferođlu, S. S. (2016). *Yeni bir 21. y¼zyıl becerisi olarak kodlama eđitimi ve kodlamanın eđitim politikalarına etkisi*. Akademik Biliřim Konferansı, 3-5 řubat 2016, Adnan Menderes niversitesi, Aydın. Eriřim adresi: https://yunus.hacettepe.edu.tr/~Sadi/yayin/AB16_Sayin-Seferoglu_Kodlama.pdf Eriřim tarihi: 28/12/2021.
- Sebetci, ., Aksu, G. (2014). *đrencilerin mantıksal ve analitik d¼ř¼nme becerilerinin programlama dilleri bařarisına etkisi*. Adnan Menderes niversitesi. 13. 65-83.
- Serim, E., . (2019). *Oyunlařtırma yntemiyle tasarlanan kodlama eđitimi ile đrencilerin hesaplamalı d¼ř¼nme becerileri ve kodlamaya iliřkin z-yeterlik algılarının incelenmesi*. (Yayınlanmamıř y¼ksek lisans tezi). Balıkesir niversitesi, Fen Bilimleri Enstit¼s¼, Balıkesir.
- Shearer, B. C. (2020) *Multiple Intelligences in Gifted and Talented Education: Lessons Learned From Neuroscience After 35 Years*. Roper Review, 42:1, 49-63, DOI:10.1080/02783193.2019.1690079
- Shearer, C. B., Karanian, J. M. (2017). *The neuroscience of intelligence: Empirical support for the theory of multiple intelligences?* Trends in Neuroscience and Education, 6, 211–223. doi:10.1016/j.tine.2017.02.002
- Shute, V. J., Sun, C. & Asbell-Clarke, J. (2017). *Demystifying computational thinking*, Educational Research Review, Volume 22, Pages 142-158, ISSN 1747-938X, <https://doi.org/10.1016/j.edurev.2017.09.003>.
- Shyamasundar, R. (1996). *Algorithms*. Resonance. 2. 16-32. 10.1007/BF02838966.
- Stajkovic, A. D., Luthans, F. (1998). *Self-efficacy and work-related performance: A meta-analysis*. Psychological Bulletin, 124(2), 240–261. <https://doi.org/10.1037/0033-2909.124.2.240>
- Strawhacker, A., Bers, M. U. (2019). *What they learn when they learn coding: investigating cognitive domains and computer programming knowledge in young children*. Education Tech Research Dev 67, 541–575 (2019). <https://doi.org/10.1007/s11423-018-9622-x>

- Swanson, A., Platine, D. & Zobisch, P. (2015). *The Theory of Multiple Intelligences and Critical Thinking*. GLOKALde. 1.
- Şentürk B., C., Kurnaz, A., Kaynar, H. & Doğrukök, B. (2020). *İlkokul öğrencilerinin uzaktan eğitime ilişkin görüşleri*. *Turkish Studies*, 15(8), 3753-3774. <https://dx.doi.org/10.7827/TurkishStudies.44496>
- Şerefhanoglu, H. (2007). *İlköğretim ikinci kademe öğrencilerinin bilgisayara yönelik tutumları ile çoklu zekâ alanlarının karşılaştırılması*. (Yayınlanmamış yüksek lisans tezi). Balıkesir Üniversitesi Fen Bilimleri Enstitüsü: Balıkesir.
- Talan, T. (2020). *Investigation of the Studies on the Use of Scratch Software in Education*. 18. 95-111. 10.30786/jef.556701.
- Tekerek, M., Altan, T., (2014). *The effect of Scratch environment on student's achievement in teaching algorithm*. *World Journal on Educational Technology*, 6(2), 132-138.
- Tiflis, Ö. (2018). *Türkiye, İngiltere ve Rusya ortaokul müfredat ve ders kitaplarında kodlama eğitimi konusunun yeri üzerine karşılaştırmalı bir çalışma*. (Yayınlanmamış yüksek lisans tezi). Uludağ Üniversitesi, Eğitim Bilimleri Enstitüsü, Bursa.
- Totan, H. N. (2021). *Blok tabanlı kodlama eğitiminin ortaokul öğrencilerinin bilgi işlemsel düşünme becerileri ve kodlama öğrenimine yönelik tutumlarına etkisi: Blocky örneği*. (Yayınlanmış Yüksek Lisans Tezi). Necmettin Erbakan Üniversitesi, Eğitim Bilimleri Enstitüsü Bilgisayar ve Öğretim Teknolojileri Eğitimi Anabilim Dalı, Konya.
- Tsarava, K., Moeller, K., Pinkwart, N., Butz, M., Trautwein, U. & Ninaus, M. (2017). *Training Computational Thinking: Game-Based Unplugged and Plugged-in Activities in Primary School*.
- Tucker, A., McCowan, D., Deek, F., Stephenson, C., Jones, J., & Verno, A. (2004). *Implementation challenges for a K-12 computer science curriculum*. *SIGCSE Bull.* 36, 1, 334–335. DOI:<https://doi.org/10.1145/1028174.971418>

- Üzümcü, Ö. & Bay, E. (2018). *Eğitimde Yeni 21. Yüzyıl Becerisi: Bilgi İşlemsel Düşünme*. 1-16.
- Varela, C., Rebollar, C., García, O., Bravo, E. & Bilbao, J. (2019). *Skills in computational thinking of engineering students of the first school year*. Heliyon. 5. e02820. 10.1016/j.heliyon.2019.e02820.
- Vatansever, Ö. (2018). *Scratch ile programlama öğretiminin ortaokul 5. Ve 6. Sınıf öğrencilerinin problem çözme becerisi üzerindeki etkisinin incelenmesi*. (Yayınlanmış Yüksek lisans tezi) Uludağ Üniversitesi, Eğitim Bilimleri Enstitüsü, Bursa.
- Wilson, C., Sudol, L., Stephenson, C. & Stehlik, M. (2010). *Running on Empty: the Failure to Teach K--12 Computer Science in the Digital Age*. 10.1145/3414583.
- Wing, J., M. (2006). *Computational thinking*. Communications of the ACM, 49(3), 33–. doi:10.1145/1118178.1118215
- Yıldırım, S., T. (2020). *Robotik kodlama öğretiminde probleme dayalı öğrenme yaklaşımının başarı, pozitif duygu ve bilgi işlemsel düşünmeye etkisi*. (Yayınlanmamış yüksek lisans tezi). Çanakkale 18 Mart Üniversitesi, Eğitim Bilimleri Enstitüsü, Çanakkale.
- Yıldız, S. N., Ates, C., A. & Kışla, T. (2020). *Perceived Acceptance and Use of Scratch Software for Teaching Programming: A Scale Development Study*. International Journal of Computer Science Education in Schools, 4(1), 53 - 71. <https://doi.org/10.21585/ijcses.v4i1.59>
- Yücel T., B., Güneri T., N. (2007). *Sosyal Bilimler Alanındaki Araştırmalarda Bilimsel Araştırma Süreci, İstatistiksel Teknikler ve Yapılan Hatalar*. Gazi Üniversitesi Ticaret ve Turizm Eğitim Fakültesi Dergisi , (1) , 1-20.
- Yünkül, E., Durak, G., Çankaya, S. & Misirli, Z., A. (2017). *Scratch Yazılımının Öğrencilerin Bilgisayarca Düşünme Becerilerine Etkisi*. Necatibey Eğitim Fakültesi Elektronik Fen ve Matematik Eğitimi Dergisi. 502-517. 10.17522/balikesirnef.373424.

9. EKLER

EK-A BLOK TEMELLİ PROGRAMLAMAYA İLİŞKİN ÖZ-YETERLİK ALGISI ÖLÇEĞİ

Aşağıdaki Scratch programına ilişkin verilen görevleri yaparken kendinize olan güveninizi 1 ile 5 arasında derecelendirerek belirtiniz. Anlamadığınız soru olursa boş bırakınız. Scratch programını kullandığınız zamanları düşünerek, aşağıdaki ifadeleri okuyunuz.

		Hiç güvenmiyorum	Biraz güveniyorum	%50/50	Oldukça güveniyorum	Tamamen güveniyorum
		1	2	3	4	5
1	Scratch'te yazılmış bir program (yazılar) gördüğümde, çalıştırıldığında neler olacağını söyleyebilirim.					
2	Başkası tarafından hazırlanan bir programı (yazılarını) okuyup anlayabilirim.					
3	Bir kuklaya herhangi bir hareket vermek istediğimde, Scratch'te bunu nereden yapabileceğimi bilirim.					
4	Sahnedeki kuklayı istediğim hızda hareket ettirebilirim.					
5	Sahnedeki kuklayı sürekli hareket ettirebilirim.					
6	Scratch'te bir kuklanın görünümünü (kostüm, renk, boyut, konuşma gibi) bir koşula bağlı olarak (örneğin: eğer ise) değiştirebilirim.					
7	Scratch'te bir kuklanın hareketlerini (hızı, yönü, konumu gibi) bir koşula bağlı olarak (örneğin: eğer ise) değiştirebilirim.					
8	Bir oyunda kullanıcının elde ettiği puan değerinin tutulacağı bir değişken oluşturabilirim.					
9	Bir oyunda istenilenler başarılıldıkça "Puan" veya "Skor" değerinin arttığı veya azaldığı bir program hazırlayabilirim.					
10	İstenilenler açıkça tanımlandığında oldukça karmaşık ve uzun kodlardan (yazılardan) oluşan bir oyun hazırlayabilirim.					
11	Scratch'te hazırlanan bir programdaki hataları bulabilirim.					
12	Scratch'te hazırlanan bir programdaki hataları düzeltip çalışabilir hale getirebilirim.					

EK-B ÇOKLU ZEKÂ ALANLARI DEĞERLENDİRME ÖLÇEĞİ

EK-B1 MANTIKSAL/MATEMATİKSEL ZEKÂ ALANI DEĞERLENDİRME ÖLÇEĞİ

Aşağıdaki soruların kendinize olan uygunluğunu 1 ile 5 arasında derecelendirerek belirtiniz.
Anlamadığınız soru olursa boş bırakınız.

		Benim için hiç uygun değil	Benim için çok az uygun	Benim için kısmen uygun	Benim için oldukça uygun	Benim için tamamen uygun
		1	2	3	4	5
1	Bilgileri öğrenirken kendim için anlamlı sınıflamalar oluşturur, fikirler arasındaki benzerlik ya da farklılıkları ortaya koyarım.					
2	Matematik oyunlarından hoşlanırım ve aritmetik problemlerini kafadan hesaplarım.					
3	Bir sorunun çözümü için farklı alternatifler üretip sorunun çözümüne yönelik çok çabuk stratejiler geliştirebilirim.					
4	Satranç ve benzeri akıl oyunlarını severim.					
5	Mantık bulmacalarını, beyin jimnastiğini severim ve mantıksal problemleri kolaylıkla çözerim.					
6	Bilgisayarda oyunlardan çok hoşlanırım.					
7	Deneylerden, yeni denemeler yapmaktan hoşlanırım, çelişkileri kolaylıkla fark edebilirim.					
8	Arkadaşıma oranla daha soyut düşünebilirim ve durumu irdeleyici sorular sorarım.					
9	Bir aracın, bir makinenin nasıl çalıştığına dair sorular sorar ve çalışma sistemini kolaylıkla kavrarım.					
10	Sebep-sonuç ilişkilerini kurmaktan zevk alır ve olayların ana fikir ve yan fikirleri kolaylıkla bulabilirim.					

EK-B2 SÖZEL/DİLSEL ZEKÂ ALANI DEĞERLENDİRME ÖLÇEĞİ

Aşağıdaki soruların kendinize olan uygunluğunu 1 ile 5 arasında derecelendirerek belirtiniz.
Anlamadığınız soru olursa boş bırakınız.

		Benim için hiç uygun değil	Benim için çok az uygun	Benim için kısmen uygun	Benim için oldukça uygun	Benim için tamamen uygun
		1	2	3	4	5
1	Resimlerden çok yazılar dikkatimi çeker.					
2	İsimler, yerler, tarihler konusunda belleğim iyidir.					
3	Kitap okumayı severim.					
4	Düşüncelerimi kelimeleri doğru bir şekilde telaffuz ederek sözlü ya da yazılı olarak kolaylıkla ifade edebilirim.					
5	Bilmecelerden, kelime oyunlarından hoşlanır ve başarılı bir performans sergilerim.					
6	Duyduğum ya da okuduklarımı kolaylıkla hatırlar ve dinleyerek daha iyi öğrenirim.					
7	Yaşıma göre kelime hazinem iyidir. Problemleri ve çözümlerini kolaylıkla sözcüklere dökerim.					
8	Yazı yazmaktan hoşlanırım ve yazı yazarken sözcükleri yerli yerinde kullanırım.					
9	Öğrendiğim yeni kelimeleri kullanmayı sever ve bunları kullanmaya çalışırım.					
10	Sözel tartışmalarda karşımdaki kişileri konuşmamla etkilerim.					

EK-B3 GÖRSEL/UZAMSAL ZEKÂ ALANI DEĞERLENDİRME ÖLÇEĞİ

Aşağıdaki soruların kendinize olan uygunluğunu 1 ile 5 arasında derecelendirerek belirtiniz.
Anlamadığınız soru olursa boş bırakınız.

		Benim için hiç uygun değil	Benim için çok az uygun	Benim için kısmen uygun	Benim için oldukça uygun	Benim için tamamen uygun
		1	2	3	4	5
1	Renklere karşı çok duyarlıyım, renkler arasındaki kombinasyonu kolaylıkla kurabilirim.					
2	Plan, kroki ya da harita türü materyalleri daha kolay algıladığımdan, bilgileri şematize ederek daha kolay hatırlarım.					
3	Arkadaşlarıma oranla daha fazla hayal kurar ve hayallerimdeki ayrıntılara da önem veririm.					
4	Resim yapmayı ve boyamayı çok sever, resimleri yazılardan daha anlamlı bulurum ve anlatımlarımda resimler kullanmayı daha çok tercih ederim.					
5	Yap-boz, lego gibi oyunlardan hoşlanır, bu oyunlarındaki gerekli parçaları kolaylıkla bulurum.					
6	Daha önce gittiğim yerleri kolayca hatırlarım.					
7	Bulmaca çözmekten hoşlanırım.					
8	Rüyalarımı çok net ve ayrıntılarıyla hatırlar, gözlerimi kapadığımda nesnelere zihnimde kolaylıkla canlandırırım.					
9	Resimli kitapları daha çok severim					
10	Çalışırken karalamalar yapar, kitaplarıma, defterlerime, diğer materyallerine çizgiler çizer, notlar alırım.					

EK-B4 DUYUMSAL/MÜZİKSEL ZEKÂ ALANI DEĞERLENDİRME ÖLÇEĞİ

Aşağıdaki soruların kendinize olan uygunluğunu 1 ile 5 arasında derecelendirerek belirtiniz.
Anlamadığınız soru olursa boş bırakınız.

		Benim için hiç uygun değil	Benim için çok az uygun	Benim için kısmen uygun	Benim için oldukça uygun	Benim için tamamen uygun
		1	2	3	4	5
1	Şarkıların melodilerini rahatlıkla hatırlar ve küçük bir bölümünü duyduğum melodiyi kolaylıkla tamamlayabilirim.					
2	Güzel şarkı söyler, kendimce basit besteler oluştururum.					
3	Müzik aleti çalar ya da çalmayı çok isterim.					
4	Müzik dersini çok severim ve bir müzik eserindeki ritim bozukluklarını anlayabilirim.					
5	Ritmik konuşur ya da hareket ederim.					
6	Farkında olmadan mırıldanırım. Herhangi bir şeyi hatırlamak istediğimde de bir ritim oluştururum.					
7	Çalışırken veya bir konu üzerinde düşünürken elimle ya da ayağımla ritim tutarım.					
8	Çevredeki sesler çok dikkatimi çeker ve bu sesleri kolaylıkla taklit edebilirim.					
9	Çalışırken veya yorgun olduğumda müzik dinlemek çok hoşuma gider.					
10	Müzikle uğraşmayı ve öğrendiğim şarkıları paylaşmayı severim.					

EK-B5 İÇSEL/ÖZE DÖNÜK ZEKÂ ALANI DEĞERLENDİRME ÖLÇEĞİ

Aşağıdaki soruların kendinize olan uygunluğunu 1 ile 5 arasında derecelendirerek belirtiniz.
Anlamadığınız soru olursa boş bırakınız.

		Benim için hiç uygun değil	Benim için çok az uygun	Benim için kısmen uygun	Benim için oldukça uygun	Benim için tamamen uygun
		1	2	3	4	5
1	Çevremdekilerden etkilenmeden davranışlarımın neden ve sonuçlarını ortaya koyarak, bağımsızca düşünmeyi ve bağımsız davranmayı severim.					
2	Güçlü ve zayıf yanlarımı bildiğim için hayatıma ilişkin uzun ve kısa vadeli hedefler belirlerim.					
3	Tek başıma zaman geçirmekten keyif alır, bu nedenle de yalnız çalışmayı daha çok severim.					
4	Çoğunlukla yalnız olmayı sever, duygularımı yazılı olarak ifade etmek için günlük tutarım					
5	Yaptığım işleri arkadaşlarımla paylaşmayı severim.					
6	Yaptığım işlerin bilincinde olduğum için kendimi tarafsız bir gözle değerlendirerek duygu ve düşüncelerimi kolaylıkla dile getiririm.					
7	Pek kimseye akıl danışmam, fikirlerimi ve olayları kendim analiz ederim.					
8	Kendime saygım yüksek olduğundan zaman zaman neden sorusunu kendime sorabilirim.					
9	Yoğun olarak uğraştığım bir ilgi alanım ve hobim vardır.					
10	Yardım istemeden kendi başıma problemleri çözmeyi, ürünler ortaya koymayı severim.					

EK-B6 SOSYAL/KİŞİLER ARASI ZEKÂ ALANI DEĞERLENDİRME ÖLÇEĞİ

Aşağıdaki soruların kendinize olan uygunluğunu 1 ile 5 arasında derecelendirerek belirtiniz.
Anlamadığınız soru olursa boş bırakınız.

		Benim için hiç uygun değil	Benim için çok az uygun	Benim için kısmen uygun	Benim için oldukça uygun	Benim için tamamen uygun
		1	2	3	4	5
1	Arkadaşlarımla oyun oynamaktan, grupla yapılan farklı organizasyonlarda yer almaktan hoşlanırım.					
2	Çevremde bir lider olarak görülür ve içinde bulunduğum gruplarda kendiliğinde liderlik rolünü elde ederim.					
3	Çevremdeki insanların sorunlarında onlara yardım eder, problemi olan arkadaşlarıma öğütler veririm.					
4	Arkadaşlarım fikirlerime değer verir, çevremdekileri kolaylıkla yönlendirebilirim.					
5	Grupla yapılan çalışmalarda uyumu bir kişilik sergilediğim için organizasyonların vazgeçilmez elamanıyım.					
6	Yeni insanlarla tanışmak için girişimlerde bulunmaktan, onlara bir şeyler anlatmaktan çok hoşlanırım.					
7	İletişime girdiğim insanlarla yakın ilişkiler kurar ve onları sık sık ararım.					
8	Arkadaşlarımla sorunlarına yardımcı olmaktan hoşlanırım.					
9	Çevremdekiler benimle arkadaşlık kurmak ister.					
10	İnsanlara selam verir, hatırlarını sorarım.					

EK-B7 BEDENSEL/KİNESTETİK ZEKÂ ALANI DEĞERLENDİRME ÖLÇEĞİ

Aşağıdaki soruların kendinize olan uygunluğunu 1 ile 5 arasında derecelendirerek belirtiniz.
Anlamadığınız soru olursa boş bırakınız.

		Benim için hiç uygun değil	Benim için çok az uygun	Benim için kısmen uygun	Benim için oldukça uygun	Benim için tamamen uygun
		1	2	3	4	5
1	Spor yaptığımda zihinsel olarak rahatladığımı hissedeyim, bu nedenle koşmalı, atlamalı sporları çok severim.					
2	Oturduğum yerde duramam, kıvılcıdanırım.					
3	Düşüncelerimi mimik-davranışlarla rahat ifade ederim.					
4	Bir şeyi okumak, oturup düşünmek yerine, hareket ederek, yaparak öğrenmeyi severim.					
5	Merak ettiğim şeyleri elimle alarak incelemek isterim.					
6	Boş vakitlerimi dışarıda geçirmek isterim.					
7	Arkadaşlarımla fiziksel oyunlar oynamayı severim.					
8	El becerilerim gelişmiştir bir araç bozulduğunda tamir edebilirim.					
9	Sorunlarımı ya da bir sözcüğü anlatırken vücut hareketlerini ve beden dilimim rahatlıkla kullanabilirim.					
10	İnsanlara ve eşyalara dokunmaktan hoşlanırım. Dokunduğumu, etkileşimde bulunduğumu ve incelediğimi daha kolay öğrenirim.					

EK-B8 DOĞASAL ZEKÂ ALANI DEĞERLENDİRME ÖLÇEĞİ

Aşağıdaki soruların kendinize olan uygunluğunu 1 ile 5 arasında derecelendirerek belirtiniz.
Anlamadığınız soru olursa boş bırakınız.

		Benim için hiç uygun değil	Benim için çok az uygun	Benim için kısmen uygun	Benim için oldukça uygun	Benim için tamamen uygun
		1	2	3	4	5
1	Hayvanlara karşı çok meraklı olduğumdan onların davranışlarını yakından takip edip yaşamları hakkında bilgi toplarım.					
2	Doğada meydana gelen olayların neden ve sonuçlarını yakından takip eder, doğa konusunda çevremdekileri bilinçlendirmeye çalışırım.					
3	Evde hayvan besler ya da beslemeyi çok sever, hayvanları kolaylıkla tanır ve kategorize ederim.					
4	Bahçede toprakla, bitkilerle oynamayı çok sever, bitkileri kolaylıkla tanır ve kategorize edebilirim					
5	Farklı bitkileri incelemeyi ve yetiştirmeyi severim.					
6	Çevremde meydana gelen değişimlere karşı duyarlı davranır, çevre kirliliğine karşı mücadele ederim.					
7	Doğayla, bitkilerle ya da hayvanlarla ilgili belgesellere ilgi duyarım.					
8	Çevre gezileri ve incelemeler yapar, mevsimlerle ve iklim olaylarıyla çok ilgilenirim.					
9	Değişik meyve ve sebzelere karşı ilgilenirim.					
10	Doğa olaylarıyla ilgilenir ve doğal yaşamın korumak için elimden geleni yaparım.					

EK-C UYGULAMA İZİN YAZILARI

Evrak Tarih ve Sayısı: 01.12.2021-38804



T.C.
MİMAR SİNAN GÜZEL SANATLAR ÜNİVERSİTESİ REKTÖRLÜĞÜ
Fen Bilimleri Enstitüsü

Sayı : E-96241527-302.14.99-38804
Konu : Mediha Ayşen YÜKSEL (Etik Kurul Onayı)

Sayın Mediha Ayşen YÜKSEL

İlgi : a) 19.11.2021 tarihli dilekçe.
b) 26.11.2021 tarih, 2021-24-05 sayılı Yönetim Kurulu Kararı.

İlgi a) dilekçenize ilişkin Enstitü Yönetim Kurulunun 26.11.2021 tarih, 24 sayılı toplantısında alınan 5 numaralı kararı ekte sunulmuştur. Bilgilerinizi ve gereğini rica ederim.

Prof. Dr. Burçin Cem ARABACIOĞLU
Enstitü Müdürü

Ek:26.11.2021 tarih, 2021-24-05 sayılı Yönetim Kurulu Kararı

Bu belge, güvenli elektronik imza ile imzalanmıştır.

Belge Doğrulama Kodu :BSR32V8ZYZ Pin Kodu :15213

Belge Takip Adresi :

http://ebys.msgsu.edu.tr/enVision/Validate_Doc.aspx

Adres:Meclis-i Mebusan Caddesi No: 24 34427 Fındıklı / İstanbul

Bilgi için: Ersin KOCAAYLA Telefon:(0212) 252

16 00 - 248 Faks:(0212) 244 05 41

Unvanı: Enstitü Sekreteri e-Posta:fenbil@msgsu.edu.tr

Tel No:

2122440541



Kep Adresi:msgsu@hs03.kep.tr

26.11.2021 tarih, 2021-24-05 sayılı Yönetim Kurulu Kararı:

Enformatik Anabilim Dalı, Bilgisayar Ortamında Sanat ve Tasarım Yüksek Lisans Programı öğrencisi Mediha Ayşen YÜKSEL'in Etik Kurul Başvuru Formuna ilişkin dilekçesi incelendi.

Yapılan değerlendirmeler sonucunda; adı geçen öğrencinin lisansüstü tez çalışması kapsamında kullanılan yöntem ve anketlere ilişkin tez danışmanı tarafından yapılan etik inceleme Üniversite Senatosu'nun 24.05.2018 tarih ve 10 sayılı toplantısında alınan 11 numaralı kararı doğrultusunda uygun görüldü.