

## Programlama Eğitiminde Algoritma Algısı: Bir Metafor Analizi\*

*Seyfullah GÖKOĞLU*

### Öz

Programlama öğretiminde algoritma tasarımı, çözülecek problemin tanımlanmasında özel bir yere sahiptir. Bu bağlamda programcı adaylarının algoritmayı nasıl algıladıkları önem kazanmaktadır. Bu nitel araştırmanın amacı Bilgisayar Programcılığı öğrencilerinin algoritma kavramına ilişkin algılarının metafor analizi yoluyla incelenmesi ve ortaya çıkan metaforların ortak özellikleri bakımından kavramsal kategoriler altında toplanmasıdır. Araştırmanın çalışma grubu, 2015-2016 eğitim-öğretim yılı bahar döneminde Kastamonu Üniversitesi'ne bağlı 5 farklı meslek yüksekokulu bünyesindeki Bilgisayar Programcılığı bölümlerinde öğrenim gören 372 önlisans öğrencilerinden oluşmaktadır. Araştırma verileri, araştırmacılar tarafından hazırlanan ve "Algoritma ..... gibidir, çünkü ....." cümlesini içeren formlar ile toplanmıştır. Veriler içerik analizi yöntemi ile çözümlenmiş ve yorumlanmıştır. Elde edilen veriler doğrultusunda Bilgisayar Programcılığı bölümü öğrencilerinin algoritmaya yönelik 63 geçerli metafor geliştirdikleri görülmüştür. Algoritma kavramına yönelik en çok geliştirilen metaforlar merdiven (%5.6,  $f=6$ ), ebeveyn (%4.7,  $f=5$ ), labirent (%4.7,  $f=5$ ) ve öğretmen (%4.7,  $f=5$ ) metaforlarıdır. Bilgisayar Programcılığı öğrencilerinin üretmiş oldukları metaforlarda algoritmanın en çok rehber, yol gösterici (%37.4,  $f=40$ ) ve sistematik bir yapı (%34.6,  $f=37$ ) yönü olduğunu vurgulanmıştır.

**Anahtar Kelimeler:** Algoritma, programlama öğretimi, metafor, algoritma algısı.

### Abstract

Algorithm design in programming instruction has a special significance in defining the problem to be solved. In this context, it gains importance that how programmer candidates perceive the algorithm. The aim of this qualitative study was to analyze the perceptions of the students at the Computer Programming Department regarding algorithm by metaphor analysis and to subsume under conceptual categories to the emerging metaphors in terms of common characteristics. The research sample consists of 372 associate degree students, studying in 5 different Computer Programming Department depending on the Kastamonu University in the 2015-2016 academic year. Data was collected with the forms prepared by the researchers and containing the phrase "Algorithms are like ....., because ....." . The data have been resolved and interpreted using the content analysis method. According to the data analysis, it was observed that the Computer Programming department students were developed 63 valid metaphors for the algorithm. Stairs (5.6%,  $f=6$ ), parents (4.7%,  $f=5$ ), labyrinth (4.7%,  $f=5$ ), and teacher (4.7%,  $f=5$ ) are the more produced metaphors according to the algorithm. In the metaphors developed by the Computer Programming students is more highlighted that algorithms have aspects as a director, guidance (37.4%,  $f=40$ ) and a systematic structure (34.6%,  $f=37$ ).

**Keywords:** Algorithm, programming instruction, metaphor, algorithm perception.

## GİRİŞ

Programlama genel olarak kullanıcıların belirli bir problemin çözümüne yönelik gerekli komutların programlama dilleri kullanılarak kod satırları şeklinde yazılması, derlenmesi ve çalıştırılması olarak ifade edilmektedir (Arabacıoğlu, Bülbül ve Filiz, 2007; Kesici ve Kocabaş, 2001). İnsan hayatına ilişkin bir olayın veya durumun etkili bir şekilde analiz edilerek ilgili duruma yönelik çözüm tasarımının bilgisayar ortamına aktarımı şeklinde gerçekleştirilen programlama (Eryılmaz, 2003), bu yönüyle gerçek hayatın bir modellemesi olarak değerlendirilmektedir (Çölkesen, 2002).

\* Bu çalışmanın bir bölümü 16-18 Mayıs 2016 tarihleri arasında gerçekleştirilen 10. Uluslararası Bilgisayar ve Öğretim Teknolojileri Sempozyumu'nda sözlü bildiri olarak sunulmuştur.

1 Öğr. Gör., Kastamonu Üniversitesi, Cide Rifat Ilgaz Meslek Yüksekokulu, [sgokoglu@kastamonu.edu.tr](mailto:sgokoglu@kastamonu.edu.tr)

Teknolojik gelişmeler ile birlikte öğrenen becerileri de değişmekte ve daha çok teknoloji odaklı becerilerin önemi üzerinde durulmaktadır. Bu çerçevede 21. yy öğrenen becerileri genel olarak 1) yaratıcılık ve yenilikçilik, 2) iletişim ve işbirliği, 3) araştırma ve bilgi akışı, 4) eleştirel düşünme, sorun çözme ve karar verme, 5) dijital vatandaşlık ve 6) teknoloji faaliyetleri ve kavramları şeklinde gruplandırılmaktadır (ISTE, 2007). Bu beceriler içerisinde yer alan ve günümüzdeki en önemli teknolojik okuryazarlık becerisi olarak değerlendirilen bir diğer beceri de programlama becerisidir (Lau ve Yuen, 2011). Bilgisayar tabanlı sistemlerin belirli bir amaca yönelik çalışması ve problemleri çözme konusunda yönlendirilmesi için sahip olunması gereken temel bir yeterlilik olan programlama becerisi, bilgisayar programı yazabilmenin yanı sıra günümüz bireylerinin sahip olması gereken üst düzey düşünme becerilerini de kullanmayı gerektiren bir yeterlilik olarak ifade edilmektedir (Yükseltürk ve Altıok, 2015). Bu özellikleri ile programlama becerisi, farklı araştırma, bilim ve faaliyet alanlarında ve öğrenme-öğretme süreçlerinde temel düzeyde sahip olunması gereken bir yeterlilik haline gelmiştir (Köse ve Tüfekçi, 2015).

Programlamaya yönelik artan ilgi ve gereksinim, ilgili bilgi ve becerilerin kazandırılması yönünde öğretimsel süreçlerin de yeniden şekillendirilmesini sağlamıştır. Bu doğrultuda programlama becerileri ve programlama öğretimine yönelik yapılan çalışmalarda artış olduğu gözlemlenmektedir (Akçay ve Çoklar, 2016; Yükseltürk ve Altıok, 2015). Programlama eğitime yönelik yapılan çalışmaları inceleyen Akçay ve Çoklar (2016), günümüzde önemli bir teknolojik okuryazarlığı becerisi olarak değerlendirilen programlama becerisinin program üretmenin yanı sıra eleştirel düşünme, algoritmik düşünme, analitik düşünme, problem çözme, çok yönlü düşünebilme ve sorgulama gibi becerileri de geliştirdiğini belirtmektedir. Bunun yanı sıra bireylere yaratıcı ve yansıtıcı düşünme becerisi kazandıran programlamanın, bilgisayar bilimlerine olan motivasyonu ve tutumu olumlu yönde etkilediği ifade edilmektedir. Programlama eğitiminin sağladığı tüm bu olumlu katkılar göz önüne alınarak okullarda programlama öğretimi yaklaşımı hızlı bir şekilde yayılmaya başlamıştır (Kafai ve Burke, 2013; Sterritt, Hanna ve Campbell, 2015).

Programlama eğitiminin ilk ve en önemli basamağında programlama mantığı öğretimi yer almaktadır (Arabacıoğlu, 2006; Arabacıoğlu, Bülbül ve Filiz, 2007). Çözümü aranan probleme yönelik gerçekleştirilen işlemler sırasında yazılan kodlar programlama dillerine göre değişkenlik gösterebilirken, programlama mantığı değişmemektedir. Bu doğrultuda herhangi bir programlama dili öğretimine geçmeden önce programlama mantığının yerleşmiş olması önem arz etmektedir. Programlama mantığının temelini ise algoritmalar oluşturmaktadır. Algoritma, sonlu bir işi tanımlamada kullanılan, belirgin bir şekilde açık seçik tanımlanabilen, yürütülebilir ve ardışık adımlardan oluşan yapılardır (Özkan, 2003). Bilgisayar yazılımlarının tasarlanmasında yararlanılan ve yazılımın geliştirilmesi için gerekli adımların kendi dilimize uygun bir şekilde ifade edildiği yapılardır (Arabacıoğlu, 2006). Algoritmalar kullanılarak programlar daha kolay bir şekilde yazılabilmektedir. Bu nedenle programlama öğretimi sürecinde öncelikle algoritma yapısının öğrenimi üzerinde durulmaktadır. Öte yandan algoritmalar sadece programlamanın planlanması ve yazılım tasarımı süreçlerinde kullanılan yapılar olmamakla birlikte insanların günlük yaşamlarında yapıyor oldukları bütün sonlu işlemleri kapsamakta ve bu işlemler algoritmik düşünülerek yapılmaktadır (Akçay ve Çoklar, 2007). Bu çerçevede algoritma kavramının etkin öğretimi üzerine kurgulanan çalışmalar son yıllarda oldukça rağbet görmektedir (Köse ve Tüfekçi, 2015).

Programlama eğitimi ve algoritma kavramına yönelik yapılan araştırmalar incelendiğinde öğrencilerin programlama derslerinde yaşadığı başarısızlıkların nedenleri ve programlama sürecinde yaşadıkları zorluklar incelenmekle (Özmen ve Altun, 2014) birlikte genellikle programlama ve algoritma mantığının öğretiminde kullanılmak üzere çeşitli yaklaşımlar geliştirilerek (Arabacıoğlu, Bülbül ve Filiz, 2007; Durak, 2009; Ersoy, Madran ve Gülbahar, 2011; Köse ve Tüfekçi, 2015) programlama öğretiminin etkililiği artırılmaya

çalışılmıştır. Özmen ve Altun (2014) üniversite öğrencilerinin programlama derslerindeki başarısızlıklarının nedenlerini ortaya koymak ve programlama sürecinde yaşadıkları zorlukları incelemek amacıyla fenomenoloji yaklaşımını kullanarak bir programlama dersinin laboratuvar etkinlikleri gözlemlenmiş ve bu derse devam eden 12 öğrenci ile mülakat yapmıştır. Araştırma sonucunda öğrencilerin programlama sürecinde yaşadıkları zorluklar programlama bilgisi, programlama becerisi, programın mantığını kavrama ve hata ayıklama olarak belirlenmiştir. Öğrenciler, programlamadaki başarısızlıklarının en büyük nedenlerini bilgi eksikliği, uygulama yetersizliği ve algoritma oluşturmama olarak vurgulamışlardır.

İnsan hayatındaki işlevselliği ve programlama öğretimindeki önemi göz önüne alınarak uzun yıllardır bilgisayar bilimleri öğrencilerine yönelik verilen eğitimlerde algoritma kavramının öğretimi önemli bir yer tutmaktadır. Bu doğrultuda gerek programlama öğretiminin etkililiği gerekse geleceğin programcıları olarak yetişecek öğrencilerin geliştirecekleri yazılımlara yönelik etkin tasarım ve analiz yapabilmeleri bağlamında algoritma kavramının programlama eğitimi alan öğrenciler tarafından nasıl algılandığının ortaya konulması yararlı olacaktır.

Doğası gereği soyut ve dinamik bir yapı gösteren algoritmaya (Hansen, Narayanan ve Hegarty, 2002) yönelik algıların bu tür soyut, karmaşık veya kuramsal bir olguyu anlamada ve açıklamada işe koşulabilecek güçlü bir zihinsel araç olarak değerlendirilen metaforlar (Saban, Koçbeker ve Saban, 2006) ile ortaya konulabileceği düşünülmektedir. Metaforlar bir nesnenin niteliğini, bir eylemin özelliğini daha iyi anlatabilmek için bir başka nesne veya eylemden yararlanarak, onu anımsatma yoluyla gerçekleştirilmektedir (Aksan, 1998). Palmquist (1996), metaforun iki kavramdan bilinmeyen daha iyi bilinen diğer kavrama benzeterek ne olduğunu daha iyi anlamamızı sağlayacağını ve olaylar hakkındaki algımızı geliştirerek ilave bakış açıları sağlayabileceğini belirtmektedir. Benzer şekilde metaforların somut kavramları kullanarak soyut kavramları ve olguları anlamayı geliştireceği vurgulanmaktadır (Heywood, 2002). Metaforların aralarında algısal benzerlik olan bir objeden diğerine transfer edilen anlamlar olarak tanımlanması (Wulf ve Dudis, 2005), güçlü bir bilişsel haritalama ve insanların dünyalarını algılamalarına yardımcı bir modelleme mekanizması olarak belirtilmesi (Arslan ve Bayrakçı, 2006) ve kişilerin algılarını açığa çıkartmada ve anlamada paha biçilmez araçlar olarak ifade edilmesi (Hansen, 2004), bilgisayar programcılığı öğrencilerinin algoritmaya yönelik algı ve tutumlarının ortaya konulabilmesi açısından metafor kullanımının önemli ipuçları sağlayacağı düşünülmektedir.

### **Araştırmanın Amacı**

Bu araştırmanın genel amacı, Bilgisayar Programcılığı bölümlerinde öğrenim gören öğrencilerin algoritma kavramına yönelik algılarının metaforlar aracılığıyla incelenmesi ve ortaya çıkan metaforların sınıf düzeyi ve cinsiyet açısından karşılaştırılmasıdır. Bu genel amaç doğrultusunda aşağıdaki sorulara yanıt aranmıştır:

- 1) Bilgisayar Programcılığı bölümü öğrencilerinin algoritma kavramına ilişkin geliştirdikleri metaforlar nelerdir?
- 2) Bilgisayar Programcılığı bölümü öğrencilerinin algoritma kavramına ilişkin geliştirmiş oldukları metaforlar, ortak özellikleri bakımından hangi kavramsal kategoriler altında toplanabilir?
- 3) Bilgisayar Programcılığı bölümü öğrencilerinin algoritma kavramına ilişkin geliştirmiş oldukları metaforlar cinsiyet ve sınıf düzeylerine göre farklılaşmakta mıdır?

### **YÖNTEM**

#### **Araştırma Deseni**

Bu çalışma, Bilgisayar Programcılığı bölümlerinde öğrenim gören öğrencilerin algoritma kavramına ilişkin algılarını ortaya koymaya yönelik olarak betimsel desende

gerçekleştirilmiştir. Betimleme arařtırmalarında, mevcut durumlar daha önceki olay ve kořullarla iliřkileri dikkate alınarak aıklamaya alıřılır (Kaptan, 1998). Arařtırmaya konu olan olay, olgu veya nesnenin kendi kořulları iinde ve olduėu gibi tanımlanmasına nem verilir (Karasar, 2005).

### Arařtırma Grubu

Arařtırmanın alıřma grubunu 2015-2016 eėitim-ėretim yılında Kastamonu niversitesi'ne baėlı 5 farklı Meslek Yksekokulu bnyesindeki Bilgisayar Programcılıėı blmlerinde ėrenim gren ve algoritma kavramının ėretildiėi Programlama Temelleri dersini almıř 372 nlisans ėrencisi oluřturmaktadır. Katılımcılardan elde edilen formlar zerinde yapılan incelemelerde eksik ya da hatalı doldurulan 21 form arařtırma kapsamı dıřında tutularak analizler 351 ėrenci zerinde gerekleřtirilmiřtir. Katılımcılara iliřkin bilgiler Tablo 1'de sunulmuřtur.

**Tablo 1. Katılımcıların Demografik Bilgileri**

		Ara Rafet Vergili MYO		Cide Rifat Ilgaz MYO		İnebolu MYO		Kastamonu MYO		Tařkpr MYO		Toplam	
		N	%	N	%	N	%	N	%	N	%	N	%
Cinsiyet	Kız	24	52.2	13	21.7	14	27.5	53	37.1	16	31.4	120	34.2
	Erkek	22	47.8	47	78.3	37	72.5	90	62.9	35	68.6	231	65.8
	<b>Toplam</b>	<b>46</b>	<b>100</b>	<b>60</b>	<b>100</b>	<b>51</b>	<b>100</b>	<b>143</b>	<b>100</b>	<b>51</b>	<b>100</b>	<b>351</b>	<b>100</b>
Sınıf Dzeyi	1. sınıf	18	39.1	29	48.3	25	49.0	76	53.1	31	60.8	179	51.0
	2. sınıf	28	60.9	31	51.7	26	51.0	67	46.9	20	39.2	172	49.0
	<b>Toplam</b>	<b>46</b>	<b>100</b>	<b>60</b>	<b>100</b>	<b>51</b>	<b>100</b>	<b>143</b>	<b>100</b>	<b>51</b>	<b>100</b>	<b>351</b>	<b>100</b>

Arařtırmaya katılan toplam 351 ėrencinin 120'sinin kız (%34.2) ve 231'inin erkek (%65.8) olmak zere bu ėrencilerin 179'unun 1. sınıf (%51.0) ve 172'sinin 2. sınıf (%49.0) ėrencisi oluřturmaktadır.

### Verilerin Toplanması

Metafor arařtırmalarında "gibi" ifadesi konu ile kaynak arasındaki benzerliėi aėrıřtırmak; "unk" ifadesi ise katılımcıların kendi benzetimlerine gereke veya mantıksal dayanak retmeleri iin kullanılmaktadır (Saban, 2004; Ocak ve Gndz, 2006). Bu doėrultuda arařtırma verileri, arařtırmacılar tarafından hazırlanan ve "Algoritma ..... gibidir, unk ....." ifadesini ieren formlar ile toplanmıřtır.

ėrenciler metaforlarını yazmadan nce metafor kavramına iliřkin aıklamalar yapılarak algoritma kavramını canlı, cansız, gerek veya hayali bir varlıėa benzetmeleri ve bu benzetimi gerekesiyle birlikte formdaki ilgili yerlere yazmaları istenmiřtir.

### Verilerin Analizi

Arařtırma kapsamında elde edilen veriler ierik analizi yntemi kullanılarak analiz edilmiřtir. İerik analizi kullanımındaki temel ama toplanan verileri aıklayabilecek kavramlara ve iliřkilere ulařmaktır (Yıldırım ve řimřek, 2011). Bu erevede birbirine benzeyen veriler belirli kavramlar ve kategoriler erevesinde bir araya getirilerek dzenlenmiř ve yorumlanmıřtır. Katılımcılar tarafından retilen metaforların analiz ve yorumlanma srecinde ařaėıdaki ařamalar takip edilmiřtir (Saban, 2004):

- ėrencilerin yazılarında bir metaforun belirgin bir řekilde ifade edilip edilmemesinin incelenmesi,

- Üretilen her bir metaforun algoritma kavramını anlamaya hizmet etme durumları incelenerek bu amaca hizmet etmeyen metaforların araştırma kapsamı dışında bırakılması,
- Üretilen metaforlara yönelik “çünkü .....” ifadesinden sonra yazılan mantıksal dayanak ve gerekçelerin analiz edilmesi,
- Aynı metafor içerisinde birden çok benzetme yapılan, veya birden çok fikir ileri sürülen ifadelerin incelenerek araştırma kapsamı dışına çıkarılması,
- Ortak özelliklere sahip olan metaforların belirli bir kategori altında toplanması ve yorumlanması.

Gerçekleştirilen analizler sonucunda araştırmaya katılan 351 öğrenci tarafından algoritma kavramına ilişkin üretilen 107 geçerli metafor araştırma kapsamına dahil edilmiştir. Bu metaforlar incelendiğinde içerisinde birbirinden farklı yapıda olan 63 metafor üretildiği belirlenmiştir. Üretilen metaforlara yönelik yüzde (%) ve frekans (*f*) değerleri hesaplanarak ortak özellikleri bakımından belirli kavramsal kategorilere ayrılmışlardır. Bilgisayar Programcılığı bölümü öğrencilerinin algoritmaya ilişkin algılarının cinsiyet ve sınıf düzeyine göre farklılık gösterip göstermediği Pearson ki-kare yöntemi kullanılarak analiz edilmiştir.

Araştırmanın güvenilirliğine yönelik olarak belirli kavramsal kategoriler altında toplanan metaforların ilgili kategoriye temsil etme düzeyini belirlemek amacıyla 63 metaforun ve 5 kavramsal kategorilerin sıralandığı bir liste ile uzman görüşüne başvurulmuştur. Uzmardan listede yer alan metaforları kavramsal kategoriler ile eşleştirmesi istenmiştir. Miles ve Huberman (1994) tarafından önerilen güvenilirlik hesaplaması (Güvenirlilik=Görüş birliği/[Görüş birliği + Görüş ayrılığı] X 100) formülü kullanılarak araştırmacılar arasındaki uyum test edilmiştir. Nitel araştırmalarda araştırma güvenilirliğinin sağlanması bakımından araştırmacı ile uzman görüşleri arasındaki uyumun en az .80 düzeyinde olması gerektiği belirtilmektedir (Creswell, 2013). Yapılan hesaplama sonucunda araştırma güvenilirliği .92 olarak bulunmuştur.

## BULGULAR

Bu bölümde gerçekleştirilen analizler sonucunda Bilgisayar Programcılığı bölümü öğrencilerinin algoritma kavramına yönelik geliştirmiş oldukları metaforlara ilişkin bulgulara yer verilmiştir. Bulgular, araştırma problemleri doğrultusunda algoritma kavramına yönelik geliştirilen metaforlar, bu metaforların ortak özellikleri bakımından altında toplandığı kavramsal kategoriler ve geliştirilen metaforların cinsiyet ve sınıf düzeyine göre farklılaşma düzeyleri şeklinde sunulmuştur.

Bilgisayar Programcılığı bölümünde öğrenim gören 351 öğrencinin algoritma kavramına ilişkin geliştirmiş oldukları 107 metafordan 63’ü birbirinden farklı yapıdadır. Geliştirilen metaforlara ilişkin frekans ve yüzdelik dağılımlar Tablo 2’de gösterilmiştir.

**Tablo 2. Algoritma Kavramına Yönelik Geliştirilen Metaforlar**

Metafor	<i>f</i>	%	Metafor	<i>f</i>	%	Metafor	<i>f</i>	%
Merdiven	6	5.6	Çark	2	1.9	Omurga	2	1.9
Ebeveyn	5	4.7	İskelet sistemi	2	1.9	Oyun	2	1.9
Labirent	5	4.7	Kalp	2	1.9	Rubik küp	2	1.9
Öğretmen	5	4.7	Navigasyon cihazı	2	1.9	Soy ağacı	2	1.9
Harita	4	3.7	Matematik	3	2.8	Tabela	2	1.9
Ağaç	3	2.8	Puzzle	3	2.8	Yemek yapmak	2	1.9
Çocuk	3	2.8	Yol	3	2.8	Yol gösterici	2	1.9
Hayat	3	2.8	Anne	2	1.9	Diğer	40	37.0
						<b>Toplam</b>	<b>107</b>	<b>100</b>

Tablo 2 incelendiğinde öğrencilerin algoritma kavramına yönelik olarak en fazla merdiven (%5.6,  $f=6$ ), ebeveyn (%4.7,  $f=5$ ), labirent (%4.7,  $f=5$ ) ve öğretmen (%4.7,  $f=5$ ) metaforlarını, bunun yanı sıra 40 metafordan ise sadece birer tane geliştirdikleri görülmektedir.

Geliştirilen metaforlar öğrencilerin “çünkü .....” ifadesine yönelik verdikleri yanıtlar çerçevesinde incelendiğinde ortak özellikleri bakımından 5 kavramsal kategori altında toplandığı belirlenmiştir. Geliştirilen metaforlara ilişkin kavramsal kategoriler Tablo 3’te gösterilmiştir.

**Tablo 3. Öğrenciler Tarafından Üretilen Metaforlara Yönelik Oluşturulan Kavramsal Kategoriler**

Kavramsal Kategoriler	Metaforlar (f)	f	%
Rehber, yol gösterici	anne(2), beyin(1), ebeveyn(5), eleştiri(1), ev projesi(1), harita(4), lider(1), merdiven(6), navigasyon cihazı(2), öğretici(1), öğretmen(5), öngörü(1), soy ağacı(2), yol(3), yol gösterici(2), tabela(2), uzman(1)	40	37,4
Sistemik bir yapı	ağaç(3), ağaç kökü(1), ağaç yaprakları(1), altın oran(1), asansör(1), basamak(1), bisiklet(1), bitki(1), biyografi(1), canlı(1), DNA(1), hayat(3), insan yaşamı(1), kalp(2), kırık bir vazodan(1), kozalak(1), kronoloji(1), matematik(3), oyun(2), örümcek ağı(1), puzzle(3), rubik küp(2), sınav(1), yemek yapmak(2), zincirleme bir reaksiyon(1)	37	34,6
Komut uygulayıcı	bebek(1), çark(2), çırak(1), çocuk(3), hesap makinesi(1), kalfa(1), köle(1), polis(1), toprak(1)	12	11,2
Karmaşık, can sıkıcı	buzdolabı(1), cam(1), canavar(1), kaplumbağa(1), labirent(5), takıntılı bir insan(1)	10	9,3
Temel yapıtaş	doğum(1), hücre(1), inşaat temeli(1), iskelet sistemi(2), omurga(2), su(1)	8	7,5
<b>Toplam</b>		<b>107</b>	<b>100</b>

Tablo 3 incelendiğinde, öğrencilerin algoritma kavramına yönelik geliştirmiş oldukları metaforların 1) rehber, yol gösterici, 2) sistemik bir yapı, 3) komut uygulayıcı, 4) karmaşık, can sıkıcı ve 5) temel yapıtaş kategorileri altında toplandığı görülmektedir. Geliştirilen metaforlar ortak özellikleri bakımından en fazla rehber, yol gösterici (%37.4,  $f=40$ ) ve sistemik bir yapı (%34.6,  $f=37$ ) kavramsal kategorileri altında toplanmıştır. Aşağıda ilgili kavramsal kategoriler ve bu kategorilere yönelik geliştirilen metaforlar açıklanmıştır.

### ***Rehber, yol gösterici***

Rehber, yol gösterici kavramsal kategorisi, algoritma ile ilgili en fazla metaforun geliştirildiği kategoridir. Bu kategoriye yönelik toplam 40 geçerli metafor türetilmiştir. Geliştirilen 17 metafor ise birbirinden farklı yapıdadır. Merdiven ( $f=6$ ), öğretmen ( $f=5$ ), ebeveyn ( $f=5$ ), harita ( $f=4$ ) ve yol ( $f=3$ ) bu kategori çerçevesinde en çok türetilen metaforlar olarak ön plana çıkmaktadır.

Programlama bölümü öğrencilerine göre algoritma daha çok belirli bir amaca ulaşmak ve karşılaşılan problemlerin çözümüne adım adım gidilmesi gereken bir merdiven olarak algılanmaktadır. Programlama sürecinde öğretici, yol gösterici olma ve belirli bir aşamadan başlayarak sonuca kadar rehberlik etmesi özelliği ile bir öğretmen olarak değerlendirilmektedir. Benzer şekilde bağlayıcı olma, sorulan sorulara yanıt verme ve çözümü adım adım göstermesi yönüyle de ebeveyn olarak görülmektedir. Bu kategoriye yönelik geliştirilen metaforlar genel olarak incelendiğinde öğrencilerin algoritmayı programlama eğitimi sürecinde kendilerine bir rehber, yol gösterici olarak algıladıklarını göstermektedir. Rehber, yol gösterici kavramsal kategorisi çerçevesinde geliştirilen metaforlara ilişkin örnekler aşağıda verilmiştir.

*Algoritma merdiven gibidir, çünkü bir amaca varmak, sonuca ulaşmak için basamak basamak ilerlemek gerekir (Ö173).*

*Algoritma öğretmen gibidir, çünkü insana doğru ve yanlış yolları ve bu yollarda karşılaşacağı sorunları öğretir (Ö104).*

*Algoritma ebeveyn gibidir, çünkü problemin farkına varır, analiz eder ve adım adım gösterir (Ö6).*  
*Algoritma harita gibidir, çünkü gideceğiniz yeri sokak sokak, adım adım takip ettirip varmak istediğimiz noktaya götürür (Ö26).*

*Algoritma yol gibidir, çünkü bir yerden bir yere gitmemize yardımcı olur (Ö61).*

### **Sistematik bir yapı**

Sistematik bir yapı kavramsal kategorisi içerisinde en fazla geliştirilen metaforlar ağaç ( $f = 3$ ), hayat ( $f = 3$ ), matematik ( $f = 3$ ) ve puzzle ( $f = 3$ ) metaforlarıdır. Toplam 37 geçerli metaforun yer aldığı kategoriye yönelik birbirinden farklı yapıda 25 metafor geliştirilmiştir.

Programlama bölümü öğrencileri algoritmayı; birbiriyle bağlantılı ve birbirini tamamlayan parçaların bir araya geldiği bir puzzle, zaman içerisinde büyüyüp gelişen, kök, dal ve yapraklarıyla belirli bir bütünlük arz eden bir ağaç, başlangıç ve sonu olan ve belirli bir düzen içerisinde ilerleyen hayat ve işleme dayalı ve mantıksal süreçler içermesi ile matematik şeklindeki benzetimler ile açıklamışlardır. Bu metaforlar incelendiğinde algoritmanın başlangıcı ve sonu belirli olan bir düzen içerisinde işleyen sistematik bir yapı olarak algılandığı görülmektedir. Bu kategoriye yönelik öğrenciler tarafından geliştirilen metaforlara ilişkin ifadelerden kesitler aşağıda sunulmuştur.

*Algoritma puzzle gibidir, çünkü bir birbirleriyle bağlantılı parçaların mantıklı bir şekilde bir araya gelmesiyle oluşur (Ö187).*

*Algoritma hayat gibidir, çünkü insan hayatının da bir başlangıç, gelişme ve bitiş aşaması vardır. İnsan kendi algoritmasını ne kadar iyi planlarsa hayatı da o kadar planlı ve düzenli geçer (Ö95).*

*Algoritma ağaç gibidir, çünkü kök kısmı temel mantığı, dalları koşullar ve döngüleri ve yaprakları verilmek istenen mesajı oluşturur (Ö111).*

*Algoritma matematik gibidir, çünkü mantıksal işlemler kullanılır (Ö158).*

### **Komut uygulayıcı**

Bu kategoride programlama bölümü öğrencilerinin algoritmanın işlevsel yapısına odaklandıkları ve algoritmayı verilen komutları uygulayan, işlemleri gerçekleştiren ve programlama sürecinde yardımcı bir eleman olarak algıladıkları gözlemlenmiştir. Komut uygulayıcı kavramsal kategorisine yönelik olarak en fazla çocuk ( $f=3$ ) ve çark ( $f=2$ ) metaforları geliştirilmiştir. Toplam 12 geçerli metaforun geliştirildiği kategoride birbirinden farklı yapıda 9 metafor yer almaktadır.

Algoritma, bir döngüsü içerisinde belirli şartlar çerçevesinde işleyen bir çark, dışarıdan müdahale olmadan kendi başına işlem yapamayan bir çocuk, ne yapması gerektiği sürekli olarak bir başkası tarafından öğretilen bir çırak ve kendisine verilen komutları uygulamak için çalışan bir köle olarak algılanmaktadır. Bu metaforlar genel olarak değerlendirildiğinde algoritmanın kullanıcılar tarafından yönlendirilen ve verilen komutları yerine getiren bir uygulayıcı olması yönünün vurgulandığı görülmektedir. Komut uygulayıcı kavramsal kategorisine ilişkin öğrencilerin ifadelerinden örnekler aşağıda verilmiştir.

*Algoritma çocuk gibidir, çünkü onu biz yönetiriz, ne yapacağını biz söyler ve gösteririz, kendi başına bir şey yapamaz (Ö266).*

*Algoritma çark gibidir, çünkü bir döngü içerisinde şartlar uygunsa çalışır, uygun değilse çalışmaz (Ö178).*

*Algoritma çırak gibidir, çünkü sürekli komut vermen gerekir (Ö153).*

*Algoritma köle gibidir, çünkü kral tarafından verilen komutları yerine getirinceye kadar çalışır (Ö112).*

### **Karmaşık, can sıkıcı**

Bu kategoriye yönelik geliştirilen metaforlar daha çok programlama bölümü öğrencilerinin algoritma hakkındaki hislerini yansıtmaktadır. Karmaşık, can sıkıcı kavramsal

kategorisi çerçevesinde en fazla labirent ( $f=5$ ) metaforu geliştirilmiştir. Toplam 10 geçerli metaforun üretildiği kategoride birbirinden farklı yapıda 6 metafor bulunmaktadır.

Bu kategori çerçevesinde algoritmanın, çıkışa ulaşabilmek için sürekli olarak içerisindeki yolların sorgulanması gerektiği bir labirent, titiz ve hata kabul etmeyen takıntılı bir insan ve ne kadar özen gösterilirse gösterilsin ufak bir hatada kırılabilen bir cam gibi benzetimler ile karmaşık bir yapıya sahip can sıkıcı bir süreç olarak algılandığı görülmektedir. Ayrıca algoritmanın oluşturulması sürecinde çok dikkatli olunmasının ve başından sonuna kadar tüm sürecin kontrol altında tutulmasının gerekliliği vurgulanmaktadır. Bu kategoriye yönelik öğrenci ifadelerinden kesitler aşağıda verilmiştir.

*Algoritma labirent gibidir, çünkü istediğimiz sonuca ulaşmak için gerekli olan yolu takip etmemiz gerekir, yanlış yola giderek doğru sonuç çıkmaz (Ö316).*

*Algoritma takıntılı bir insan gibidir, çünkü hata kabul etmez ve her şeyi noktası virgüline kadar ister (Ö51).*

*Algoritma buzdolabı gibidir, çünkü doğru yönetilmezse insanı hayattan soğutur (Ö155).*

*Algoritma kaplumbağa gibidir, çünkü yavaş yavaş adım adım ilerler (Ö40).*

*Algoritma cam gibidir, çünkü ne kadar özen gösterirsen göster bir hata yapılıncaya kırılır ve bir işe yaramaz (Ö87).*

### **Temel yapıtaşı**

Bu kavramsal kategoriye yönelik toplam 8 geçerli metafor geliştirilmiş olup bunlardan 6 tanesi birbirinden farklı yapıdadır. İskelet sistemi ( $f=2$ ) ve omurga ( $f=2$ ) öğrenciler tarafından en fazla geliştirilen metaforlar olarak ön plana çıkmaktadır.

Bu kategori çerçevesinde algoritmanın programlama sürecindeki yeri ve önemi üzerinde durulmaktadır. Algoritmanın programlama sürecinin temelini oluşturduğu, programların algoritmalar üzerine inşa edildiği ve programlama sürecinin unsurlarını üzerine taşıyan bir sistem olduğu vurgulanmaktadır. Öğrencilerin algoritma ile ilgili temel yapıtaşı kavramsal kategorisine yönelik geliştirdikleri metaforlara ilişkin örnekler aşağıda sunulmuştur.

*Algoritma iskelet sistemi gibidir, çünkü kemiklerden biri eksik olursa iskelet ayakta duramaz benzer şekilde algoritmanın da bir parçası eksik olursa program çalışmaz (Ö191).*

*Algoritma omurga gibidir, çünkü her şeyi bir arada tutar ve olmazsa olmazdır (Ö264).*

*Algoritma inşaat temeli gibidir, çünkü ne kadar sağlam olursa o kadar iyi bir program yazılabilir (Ö346).*

*Algoritma hücre gibidir, çünkü tüm canlılar gibi programları oluşturan ilk yapı da algoritmadır (Ö97).*

*Algoritma doğum gibidir, çünkü insanın hayatının doğumla başlaması gibi programlamanın da ilk evresi algoritmadır (Ö75).*

Öğrencilerin algoritma kavramına ilişkin geliştirmiş oldukları metaforlara yönelik ortaya çıkan 5 kavramsal kategorinin cinsiyete göre farklılık gösterip göstermediği ki-kare testi ile incelenmiştir. Test sonuçları Tablo 4'te gösterilmiştir.

**Tablo 4. Kavramsal Kategorilerin Cinsiyete Göre Karşılaştırılması**

Kavramsal Kategori	Kız		Erkek		Toplam	
	<i>f</i>	%	<i>f</i>	%	<i>f</i>	%
Rehber, yol gösterici	12	30.0	28	70.0	40	37.4
Sistemik bir yapı	17	45.9	20	54.1	37	34.6
Komut uygulayıcı	5	41.7	7	58.3	12	11.2
Karmaşık, can sıkıcı	6	60.0	4	40.0	10	9.3
Temel yapıtaşı	4	50.0	4	50.0	8	7.5

$\chi^2 = 4.13, p > .05$



Tablo 4'teki veriler incelendiğinde erkek öğrencilerin kız öğrencilere göre algoritma kavramına yönelik olarak daha çok rehber, yol gösterici olma, sistematik bir yapıda bulunma ve komut uygulayıcı nitelikte olma özelliklerine vurgu yaptığı görülmektedir. Kız öğrenciler ise algoritma kavramını erkek öğrencilere göre daha karmaşık ve can sıkıcı olarak algılamaktadırlar. Öte yandan her iki cinsiyet grubu da algoritmanın programlama için temel bir yapıtaşı olduğunu görüşünde hemfikirdiler. Yapılan ki-kare testi sonuçlarına göre bilgisayar programcılığı öğrencilerinin algoritma kavramına yönelik sahip oldukları algıların cinsiyet açısından anlamlı bir farklılık göstermediği belirlenmiştir ( $\chi^2 = 4.13, p > .05$ ).

Algoritma kavramına ilişkin olarak ortaya çıkan kavramsal kategorilerin sınıf düzeyine farklılık gösterip göstermediğine yönelik yapılan analiz sonuçları Tablo 5'te gösterilmiştir.

**Tablo 5. Kavramsal Kategorilerin Sınıf Düzeyine Göre Karşılaştırılması**

Kavramsal Kategori	1. sınıf		2. sınıf		Toplam	
	f	%	f	%	f	%
Rehber, yol gösterici	24	60.0	16	40.0	40	37.4
Sistematik bir yapı	12	32.4	25	67.6	37	34.6
Komut uygulayıcı	7	58.3	5	41.7	12	11.2
Karmaşık, can sıkıcı	4	40.0	6	60.0	10	9.3
Temel yapıtaşı	4	50.0	4	50.0	8	7.5

$\chi^2 = 6.68, p > .05$

Tablo 5'e göre bilgisayar programcılığı öğrencilerinin algoritma kavramına yönelik sahip oldukları algıların sınıf düzeyi açısından anlamlı bir farklılık göstermediği belirlenmiştir ( $\chi^2 = 6.68, p > .05$ ). 1. sınıfta öğrenim gören öğrencilerde algoritmaya yönelik rehber, yol gösterici ve komut uygulayıcı özelliğe sahip olma algısı ağırlıkta iken 2. sınıf düzeyindeki öğrenciler algoritmanın sistematik bir yapıda ve karmaşık, can sıkıcı olduğunu düşünmektedirler. Her iki sınıf düzeyindeki öğrenciler ise algoritmanın programlama için temel yapıtaşı olduğu görüşünde hemfikirdiler.

## TARTIŞMA VE SONUÇ

Bu çalışma ile programlama öğretiminin giderek önem kazandığı ve bu becerinin 21. yy öğrenen becerileri arasında değerlendirildiği günümüzde (Lau ve Yuen, 2011), Bilgisayar Programcılığı bölümünde öğrenim gören öğrencilerin algoritma kavramına yönelik algıları ortaya konulmuştur.

351 öğrenci üzerinde metafor yöntemi kullanılarak yürütülen araştırmada algoritma kavramına yönelik olarak 107 geçerli metafor geliştirilmiştir. Algoritma kavramına yönelik olarak en fazla merdiven ( $f=6$ ), ebeveyn ( $f=5$ ), labirent ( $f=5$ ) ve öğretmen ( $f=5$ ) metaforları üretilmiştir. Öğrenciler tarafından üretilen metaforlar ortak özellikleri bakımından; rehber, yol gösterici (%37.4,  $f=40$ ), sistematik bir yapı (%34.6,  $f=37$ ), komut uygulayıcı (%11.2,  $f=12$ ), karmaşık, can sıkıcı (%9.3,  $f=10$ ) ve temel yapıtaşı (%7.5,  $f=8$ ) kategorileri altında toplanmıştır.

Arabacıoğlu, Bülbül ve Filiz (2007), programlama mantığı öğretiminin programlama öğretimi sürecindeki ilk ve en önemli basamak olduğunu ve her bilim dalında olduğu gibi temelin sağlam atılmasının konunun anlaşılmasında oldukça önem taşıdığını ifade etmektedir. Bu araştırmada algoritmanın programlama sürecinde bir rehber, yol gösterici olarak görülmesi ve temel yapıtaşı olarak algılanması, öğrencilerin algoritma kavramının programlama sürecindeki önemini farkında olduklarına işaret etmektedir. Öğrenciler tarafından geliştirilen inşaat temeli, iskelet sistemi ve omurga metaforları algoritmanın programa süreci açısından bir temel teşkil ettiğinin göstergesi olarak yorumlanabilir.

Analizler sonucunda öğrencilerden bir kısmının ( $f=10$ ) öğrencinin algoritmaya yönelik olumsuz görüşe sahip olduğu görülmüştür. Doğası gereği soyut ve dinamik bir yapı gösteren algoritmanın bilgisayar bilimleri öğrencileri tarafından analizi ve tasarımı zor bir iş olarak görülmesi (Akçay ve Çoklar, 2016; Hansen ve diğ., 2002) ortaya çıkan bu olumsuz algıyı doğrular niteliktedir. Benzer şekilde Özmen ve Altun (2014) öğrencilerin programlama sürecinde özellikle programın mantığını kavrama ve algoritma oluşturmada sorun yaşadıklarını vurgulamaktadır. Öte yandan programlama dili eğitimi alan öğrencilerin programlamaya karşı olan olumsuz önyargılarının, programlama dilinin öğretimini zorlaştırdığı belirtilmektedir (Arabacıoğlu, Bülbül ve Filiz, 2007). Bu doğrultuda verilen programlama eğitiminin etkililiğinin artırılabilmesi için öğrencilerin olumsuz algılarının giderilmesine yönelik çalışmalar yapılması önem arz etmektedir.

Akçay ve Çoklar (2016), programlamanın bünyesinde birden fazla bilişsel süreçleri barındırdığından dolayı öğretilmesi ve öğrenilmesi zor bir süreç olduğunu ve bu sebeple programlama eğitimindeki başarının düşük seviyede kaldığını belirtmektedir. Öğrenciler ise genelde öğretmenlerinin anlattıklarını kavradıklarını ancak uygulamada sorun yaşadıklarını ifade etmektedirler. Bu çalışmada algoritma, programlama süreci açısından sistematik bir yapı olarak algılanmaktadır (%34.6,  $f=37$ ). Bu çerçevede kendi içerisinde belirli bir sisteme dayalı olan algoritmanın öğretimi sürecinde geleneksel yaklaşımlardan uzaklaşarak kısıtlı kod bilgisine sahip kişilerin bile programlama yapabilecekleri ve kod bloklarının sürüklenip bırakılması yöntemi kullanılarak kullanılabilmesi yeni programlama yaklaşımlarının tercih edilmesi faydalı olacaktır. Bu doğrultuda algoritma öğretilmesine yönelik geliştirilen eğitsel yazılımların programlamanın temeli olan algoritmik düşünce mantığını etkili bir şekilde öğretilmesi noktasında başarılı olduğu (Köse ve Tüfekçi, 2015) ve ilgili derslere yönelik akademik başarıların artmasında etkili olduğu (Durak, 2009) belirlenmiştir.

Sonuç olarak; günümüzde programlama eğitiminin sağladığı yararlar göz önüne alındığında algoritmanın öneminin hiçte azımsanmayacak düzeyde olduğu görülmektedir. Bu çerçevede programlama öğretiminin yoğun bir şekilde verildiği bölümler arasında yer alan Bilgisayar Programcılığı bölümü öğrencileri algoritma kavramını programlama sürecinde bir rehber ve yol gösterici olarak görmekte ve onları hedeflerine götüren bir merdiven olarak algılamaktadırlar. Bu durum öğrencilerin algoritma kavramının programlamaya yönelik öneminin farkında olduklarına işaret etmektedir.

Bu araştırma Bilgisayar Programcılığı bölümü öğrencilerine yönelik olarak gerçekleştirilmiştir. Temel ve ileri düzeyde programlama eğitiminin verildiği diğer bölümlerde öğrenim gören öğrencilerin algoritmaya ilişkin algılarının metaforlar aracılığıyla karşılaştırılmasına yönelik çalışmalar yapılarak daha geniş bir bakış açısı sunulabilir. Öğrencilerin algoritma kavramına yönelik bazı olumsuz görüşlere sahip olduğu görülmüştür. Bu olumsuz algıların nedenlerinin belirlenmesi ve giderilmesine yönelik araştırmalar gerçekleştirilerek geleceğin programcıları olarak yetişen öğrencilerin algoritmayı daha etkili kullanmalarına katkıda bulunulabilir.

## KAYNAKÇA

- Akçay, A. ve Çoklar, A. N. (2016). Bilişsel becerilerin gelişimine yönelik bir öneri: Programlama eğitimi. A. İşman, H. F. Odabaşı ve B. Akkoyunlu (Eds.), *Eğitim Teknolojileri Okumaları 2016* (s. 121-139). Ankara: The Turkish Online Journal of Educational Technology (TOJET).
- Aksan, D. (1998). *Dilbilim seçkisi: Günümüz dilbilimiyle ilgili yazılardan çeviriler*. Ankara: Türk Dil Kurumu.
- Arabacıoğlu, T. (2006). *İnternet destekli programlama mantığı öğretimi* (Yayımlanmamış Yüksek Lisans Tezi). Gazi Üniversitesi, Ankara, Türkiye.
- Arabacıoğlu, T., Bülbül, H. İ. ve Filiz, A. (2007, Şubat). *Bilgisayar programlama öğretiminde yeni bir yaklaşım*. IX. Akademik Bilişim Konferansı, Kütahya, Türkiye.
- Arslan, M. M. ve Bayrakçı, M. (2006). Metaforik düşünme ve öğrenme yaklaşımının eğitim-öğretim açısından incelenmesi. *Milli Eğitim*, 35(171), 100-108.
- Creswell, J. W. (2013). *Nitel araştırma yöntemleri: Beş yaklaşıma göre nitel araştırma ve araştırma deseni*. Ankara: Siyasal Kitabevi
- Çölkesen, R. (2002). *Veri yapıları ve algoritmalar*. İstanbul: Papatya Yayıncılık.
- Durak, G. (2009). *Algoritma konusunda geliştirilen "programlama mantığı öğretici-p.m.ö" yazılımının öğrenci başarısına etkisi* (Yayımlanmamış Yüksek Lisans Tezi). Balıkesir Üniversitesi, Balıkesir, Türkiye.
- Ersoy, H., Madran, R. O. ve Gülbahar, Y. (2011, Şubat). *Programlama dilleri öğretiminde bir model önerisi: Robot programlama*. XIII. Akademik Bilişim Konferansı, Malatya, Türkiye.
- Hansen, D. T. (2004). A poetics of teaching. *Educational Theory*, 54(2), 119-142.
- Hansen, S., Narayanan N. H., & Hegarty M. (2002). Designing educationally effective algorithm visualizations. *Journal of Visual Languages & Computing*, 13(3), 291-317.
- Heywood, D. (2002). The place of analogies in science education. *Cambridge Journal of Education*, 32(2), 233-247.
- International Society for Technology in Education (ISTE). (2007). *ISTE standards: Students*. [https://www.iste.org/docs/pdfs/20-14\\_ISTE\\_Standards-S\\_PDF.pdf](https://www.iste.org/docs/pdfs/20-14_ISTE_Standards-S_PDF.pdf) adresinden 25.08.2016 tarihinde edinilmiştir.
- Kafai, Y. B. ve Burke, Q. (2013). Computer programming goes back to school. *The Phi Delta Kappan*, 95(1), 61-65.
- Kaptan, S. (1998). *Bilimsel araştırma ve istatistik teknikleri*. Ankara: Bilim.
- Karasar, N. (2005). *Bilimsel araştırma yöntemi* (15. Baskı). Ankara: Nobel.
- Kesici, T. ve Kocabaş, Z. (2001). *Liseler için bilgisayar 2*. Ankara: MEB Yayınları.
- Köse, U. ve Tüfekçi, A. (2015). Algoritma ve akış şeması kavramlarının öğretiminde akıllı bir yazılım sistemi kullanımı. *Pegem Eğitim ve Öğretim Dergisi*, 5(5), 569-586.
- Lau, W. W. F. ve Yuen, A. H. K. (2011). Modelling programming performance: Beyond the influence of learner characteristics. *Computers & Education*, 57(1), 1202-1213.
- Miles, M. B. & Huberman, A. M. (1994). *Qualitative data analysis: An expanded sourcebook* (2nd Edition). California: SAGE Publications.
- Ocak, G. ve Gündüz, M. (2006). Eğitim fakültesini yeni kazanan öğretmen adaylarının öğretmenlik mesleğine giriş dersini almadan önce ve aldıktan sonra öğretmenlik mesleği hakkındaki metaforlarının karşılaştırılması. *Afyon Kocatepe Üniversitesi Sosyal Bilimler Dergisi*, 8(2), 293-309.
- Özkan, Y. (2003). *Programlama dilleri: C ile programlama*. İstanbul: Alfa Yayınları.
- Özmen, B. & Altun, A. (2014). Undergraduate students' experiences in programming: Difficulties and obstacles. *Turkish Online Journal of Qualitative Inquiry*, 5(3), 9-27.
- Palmquist, R. (1996, October). *The search for an internet metaphor: A comparison of literatures*. Paper presented at the American Society of Information Science Conference, Baltimore, USA.

- Saban, A. (2004). Giriş düzeyindeki öğretmen adaylarının “öğretmen” kavramına ilişkin ileri sürdükleri metaforlar. *Türk Eğitim Bilimleri Dergisi*, 2(2), 131-155.
- Saban, A., Koçbeker, B. N., & Saban, A. (2006). An investigation of the concept of teacher among prospective teachers through metaphor analysis. *Educational Sciences: Theory & Practice*, 6(2), 509-522.
- Sterritt, R., Hanna, P., & Campbell, J. (2015, March). *Reintroducing programming to the school environment*. Paper presented at the 9th International Technology, Education and Development Conference, Madrid, Spain.
- Wulf, A. & Dudis, P. (2005). Body partioning in ASL metaphorical blends. *Sign Language Studies*, 5(3), 317-332.
- Yıldırım, A. ve Şimşek, H. (2011). *Sosyal bilimlerde nitel araştırma yöntemleri* (8. Baskı). Ankara: Seçkin Yayıncılık.
- Yükseltürk E. ve Altıok, S. (2015). Bilişim teknolojileri öğretmen adaylarının bilgisayar programlama öğretimine yönelik görüşleri. *Amasya Üniversitesi Eğitim Fakültesi Dergisi*, 4(1), 50-65.

## **Algorithm Perception in the Programming Education: A Metaphor Analysis**

*Seyfullah GÖKOĞLU*

### **Summary**

#### **INTRODUCTION**

Programming skill is an essential competence for guiding computer-based systems' solving problems and working for a particular purpose. Nowadays, developments occurring in the technology and increasing requirements for access to the information make the programming skill have to be a basic competence in the field of research, science, and teaching-learning process. In the programming process, codes may vary according to the programming language, but programming logic does not change. For this reason, it is important that to be settled the programming logic before teaching any programming language. The fundamental of the programming logic is constituted with algorithms. In the programming education process, it is primarily focused on learning the structure of the algorithm due to programs can be performed more easily by using algorithms. Considering the algorithm's functionality in human life as well as importance in computer programming, trainings are given to the computer science students for many years in the teaching-learning process. Therefore, it is crucial that how the algorithm perceived by students studying for the algorithm in terms of teaching of programming courses as well as for the students to be trained as programmers of the future, can efficient design and analysis to the software developed by them.

#### **PURPOSE**

The overall objective of this research is to examine the students', studying in the Department of Computer Programming, perceptions of the algorithm via metaphor and to compare the emerging results in terms of grade level and gender.

#### **METHODOLOGY**

This research is a qualitative study conducted in descriptive pattern. The research sample consists of 372 associate degree students, studying in 5 different Computer Programming Department depending on the Kastamonu University in the 2015-2016 academic year, and have taken the Programming Fundamentals Course that the concept of the algorithm is taught. Data was collected with the forms prepared by the researchers and containing the phrase "Algorithms are like ....., because ....." . Before writing the metaphors for the algorithm, the students were given some explanations about the metaphor and were requested them liken the algorithm to an entity, living, dead, real, or imaginary, and write these analogies on the forms together with the reasons. In the analysis process, the steps recommended by Saban (2004) were followed. As a result of the analysis, 107 available metaphors related to the algorithm are produced by the students. There are 63 unique metaphors. Percentage (%) and frequency (*f*) values were estimated for the metaphors and they are divided into specific conceptual categories in terms of metaphors' common traits. Students' perceptions related to the algorithm were examined by Pearson's chi-square analysis according to the gender and the class level. For the reliability of the research, expert opinion has been consulted to determine the representation level of the emerging categories to the related metaphors. Expert

feedbacks were analyzed using the reliability formula proposed by Miles & Huberman (1994) and the reliability of the research was found .92.

## FINDINGS

Results indicate that stairs (5.6%,  $f=6$ ), parents (4.7%,  $f=5$ ), labyrinth (4.7%,  $f=5$ ), and teacher (4.7%,  $f=5$ ) are the more produced metaphors according to the algorithm by the students. Metaphors are grouped under the categories in terms of common characteristics: 1) directory, guidance, 2) a systematic structure, 3) command operator, 4) complex, tedious, and 5) basic consistent. Metaphors were more collected under the conceptual category which are director, guidance (37.4%,  $f=40$ ) and a systematic structure (34.6%,  $f=37$ ) in terms of common characteristics. The chi-square results shows that there isn't a significant difference between the students' perceptions about metaphor according to gender and the grade level ( $\chi^2(\text{gender})=4.13, p>.05$ ;  $\chi^2(\text{grade})=6.68, p>.05$ ).

Consequently, students who are studying at the Departments of Computer Programming, which are performing intensive programming instruction, regard the algorithm as directory and guidance and perceive it as a stair that leads them to their goal. This result indicates that the students are aware of the importance of the algorithm for the programming.