

41. BÖLÜM

YENİ KAVRAMLAR, FARKLI KULLANIMLAR: BİLGİ-İŞLEMSEL DÜŞÜNMEYLE İLGİLİ BİR DEĞERLENDİRME

Arş. Gör. Ömer DEMİR
Prof. Dr. Süleyman Sadi SEFEROĐLU
Hacettepe Üniversitesi

Özet

Eleştirel, yaratıcı, yansıtıcı, analitik, mantıklı vb. düşünme becerileri eğitimin bireye sağlayacağı önemli kazanımlardandır. 1960'lerden beri farklı isimler altında gündemde olan bilgi-işlemsel düşünme becerisi de bireylere kazandırılması gereken düşünme becerileri arasında yer almaktadır. Bilgi-işlemsel düşünme becerisini diğer düşünme becerilerinden ayıran en önemli özellik bilgi-işlemsel düşünmenin insanların bilgisayarlarla işbirliği yapmalarına izin vermesidir. Bu nedenle özellikle 2000'li yıllardan sonra teknolojinin de gelişmesi ile bilgi-işlemsel düşünme daha çok mercek altına alınmaya başlanmıştır. Ancak bilgi-işlemsel düşünmenin eğitsel doğurgularının hala net bir şekilde anlaşılamadığı görülmektedir. Bu nedenle bu çalışma kapsamında bilgi-işlemsel düşünme ile ilgili bir alanyazın taraması yapılmıştır. Bu alanyazın taraması, ilk olarak bilgi-işlemsel düşünmenin tanımına, bilgi-işlemsel düşünmeye yönelik eleştirilere ve bilgi-işlemsel düşünme kavramının tarihteki gelişimine odaklanmaktadır. Bilgi-işlemsel düşünmenin eğitim ve oyun tabanlı öğrenme yöntemi ile ilişkisi, bilgi-işlemsel düşünmenin ölçülmesi ve zihinsel süreçler ile nasıl bir ilişki içerisinde olduğu bu alanyazın taramasının odaklandığı diğer konulardandır. Bu çalışmanın son bölümünde bilgi-işlemsel dü-

şünme ile ilgili çalışmaların hangi yöne evrilmesi, bilgi-işlemsel düşünmenin nasıl ölçülmesi ve öğretilmesi gerektiğine yönelik önerilerde de bulunulmuştur.

Anahtar Sözcükler: bilgi-işlemsel düşünme, bilgi-işlemsel düşünmede zihinsel süreçler, bilgi-işlemsel düşünmenin ölçülmesi, oyun tabanlı öğrenme, programlama

Hazırlık Soruları

1. Bilgi-işlemsel düşünme nedir?
2. Bilgi-işlemsel düşünme kavramı nasıl bir tarihi süreçten geçmiştir?
3. Bilgi-işlemsel düşünme ile oyun tabanlı öğrenme arasında nasıl bir ilişki vardır?
4. Bilgi-işlemsel düşünmenin ölçülmesinde hangi yaklaşımlar kullanılabilir?
5. Bilgi-işlemsel düşünme ile zihinsel süreçler arasında nasıl bir ilişki vardır?

Giriş

21. yüzyılda bireylerin sahip olması beklenen becerilerle ilgili olarak alanyazında bir dizi liste bulunmaktadır. Bu listelere 21. yüzyıl düşünme becerileri, sayısal vatandaşlık ve Uluslararası Eğitim Teknolojileri Topluluğu'nun (International Society for Technology in Education – ISTE) öğrenciler için standartlardan oluşan listeleri örnek olarak verilebilir. ISTE'nin 2016 yılında öğrenciler için geliştirdiği standartlar aşağıdaki şekildedir:

1. Güçlendirilmiş Öğrenen (Empowered Learner)
2. Dijital Vatandaş (Digital Citizen)
3. Bilgi İnşa Edici (Knowledge Constructor)
4. Yenilikçi Tasarımcı (Innovative Designer)
5. Bilgi-işlemsel Düşünür (Computational Thinker)
6. Yaratıcı İletişimci (Creative Communicator)
7. Küresel İşbirlikçi (Global Collaborator)

Bu listelerden de görülebileceği gibi “bilgi-işlemsel düşünme” (computational thinking) becerisi 21. yüzyılda herkesin sahip olması gereken bir beceri olarak karşımıza çıkmaktadır. Bunun en temel nedenlerinden birisi bilgi-işlemsel düşün-

menin, okuma, yazma ve temel matematik gibi evrensel bir yetenek olarak görülmesidir (Wing, 2006, 2011, 2016). “Bilgi-işlemsel düşünür” standardı kısaca öğrencilerin, çözümleri test etmek ve geliştirmek için teknolojik yöntemlerin gücünden yararlanacak bir şekilde problemleri çözmek ve anlamak amacıyla stratejiler geliştirip bunları kullanmasını ifade etmektedir. Bireylerin günlük hayatın her aşamasında kullanabilecekleri soyut düşünmeyle ilişkilendirilen (Lu & Fletcher, 2009) bilgi-işlemsel düşünme problem tespiti ve problem çözme sürecinin daha etkili hale getirilmesine yardımcı olmaktadır (Czerkowski & Lyman, 2015). Bilgi-işlemsel düşünme ayrıca, bilgisayarlar aracılığıyla bireylerin problem çözme becerilerinin geliştirilmesinde önemli olduğu bilinen yaratıcı düşünme, algoritmik düşünme, eleştirel düşünme ve işbirliği gibi becerileri vurgulamaktadır (ISTE, 2015). Bu nedenle son yıllarda yoğun bir şekilde eğitim sistemlerinde bilgi-işlemsel düşünme becerilerinin kazandırılmasına yönelik adımlar atılmaktadır. Bu kapsamda yüksek öğretimde bilgisayar bilimleri programlarının yanı sıra İngiltere gibi ülkelerin K-12 öğretim programlarına bilgisayar bilimlerinin temel kavramlarını öğreten dersler eklenmeye başlanmıştır (Wing, 2016). Türkiye de bu konuyla ilgili girişimlerde bulunan ülkeler arasında yer almaktadır.

Bu çalışmada Türkçe alanyazında “computational thinking” kavramıyla ilgili olarak gözlenen farklı kullanımlara yer verilecek ve kavramın FeTeMM (Fen bilimleri, Teknoloji, Mühendislik ve Matematik) ile ilişkisi kurulacaktır. Bu bağlamda yapılan alanyazın taraması sonucunda aşağıdaki sorulara yanıt aranacaktır:

Bilgi-işlemsel düşünme nedir? Bilgi-işlemsel düşünmenin bileşenleri nelerdir?

1. Bilgi-işlemsel düşünmeye yönelik eleştiriler nelerdir?
2. Bilgi-işlemsel düşünme kavramının gelişiminde hangi süreçler gözlenmektedir?
3. Bilgi-işlemsel düşünme ve eğitim arasında nasıl bir ilişki vardır?
4. Bilgi-işlemsel düşünme oyun tabanlı öğrenme yöntemi ile nasıl geliştirilmeye çalışılmaktadır?
5. Bilgi-işlemsel düşünme becerisinin ölçülmesinde hangi yöntemler kullanılmaktadır?
6. Bilgi-işlemsel düşünme ile zihinsel süreçler arasında nasıl bir ilişki vardır?
7. Bilgi-işlemsel düşünme kavramının geleceğiyle ilgili hangi kestirimler yapılabilir?

1. Türkçe Alanyazında “Computational Thinking” Kavramının Kullanımı

Türkçe alanyazın incelendiğinde “computational thinking” kavramının Türkçeleştirilmesi konusunda bir fikir birliğinin olmadığı görülmektedir (Bkz. Tablo 1). Bahsedilen kavram bazı araştırmacılar tarafından “bilgisayarca düşünme” (Korkmaz, Çakır & Özden, 2015; Özden, 2015) şeklinde çevrilerek kullanılmaktadır. Ancak bilgisayarca düşünme ifadesi “bilgisayar gibi düşünmek” olarak anlaşılma tehlikesi ile karşı karşıyadır. Nitekim bu konuda çok sayıda çalışması olan Wing (2008) “computational thinking” kavramından kastedilenin bilgisayar gibi düşünmek olmadığını ileri sürmektedir. Bu düşüncesini de bilgisayarların düşünmedikleri gerçeğine dayandırmaktadır. Bu durumda bilgisayarca düşünme ifadesi “bilgisayar gibi düşünmek” olarak anlaşılabilir.

Tablo 1. Türkçe Alanyazında “Computational Thinking” Kavramıyla İlgili Kullanımlar

Kullanılan kavramlar-karşılıklar	Alanyazın
Bilgisayarca düşünme	Çatlak, Tekdal & Baz, 2015; Korkmaz, Çakır & Özden, 2015; Özden, 2015
Bilişimsel düşünme	Özkeş, 2016; Sayın & Seferoğlu, 2016
Hesaplamalı düşünme	MEB, 2017
Bilgisayımsal düşünme	Doğan, Çınar, Bilgiç & Tüzün, 2015
Kompütasyonel düşünme	Aldağ & Tekdal, 2015; YTÜ BÖTE, 2016; Şahiner & Kert, 2016
Bilgi-işlemsel düşünme	Barut, Tuğtekin & Kuzu, 2016; Gülbahar, Kalelioğlu & Doğan, 2015; MEB, 2016

Aslında “computational thinking” kavramından kastedilen bilgisayar gibi düşünmek değil, bir bilgisayar bilimcisi gibi düşünmektir. Ancak “bilgisayar bilimcisi gibi düşünmek” ifadesi çok uzun olacağı için tercih edilmemiştir. Kavramın karşılığı olarak başka bir kullanım ise bilişimsel düşünmedir (Özkeş, 2016). Bu kullanım da bilgisayarca düşünme kullanımına paralel nedenlerle tercih edilmiştir. “Computational thinking” kavramıyla ilgili olarak alanyazında “bilgisayımsal düşünme” (Doğan, Çınar, Bilgiç & Tüzün, 2015) şeklinde bir kullanıma da rastlamak mümkündür. Ancak bilgisayar tarafından sayılan olgunun bilgi değil veri olduğu ve ayrıca zihnin bilgiyi matematiksel olarak saymadığı bunun yerine bilgiyi işleme kuramına göre işlediği (Miller, 1956) düşüncesinden hareketle “bilgisayımsal düşünme” kavramının uygun bulunmadığı söylenebilir. Bazı kaynaklarda “kompütasyonel düşünme” (Aldağ & Tekdal, 2015; Şahiner & Kert, 2016) ifadesi de kullanılmaktadır. Ancak bu şekilde Türkçeleştirmenin başvurulması

gereken son çare olduğu ve bu nedenle öncelikle kavramı karşılayan Türkçede var olan kavramlar üzerinden gidilmesi gerektiği düşünülmektedir. Bu kullanımların yanı sıra, “computational thinking” kavramının eğitim alanının dışında işlemsel düşünme, hesaplama dayalı düşünme ve bilişimsel muhakeme (Kuzu, 2016) olarak kullanıldığı da görülmektedir.

Milli Eğitim Bakanlığının 13.01.2017 tarihinde askıya çıkardığı “Bilişim Teknolojileri ve Yazılım” dersi taslak öğretim programında (MEB, 2017) “computational thinking” kavramının karşılığı olarak hesaplamalı düşünme ifadesini tercih edilmiştir. Ancak bu ifade sanki zihinden dört işlem yapmayı çağrıştırdığı için tercih edilmemiştir. Son olarak, bahsedilen kavramın “bilgi-işlemsel düşünme” (Barut, Tuğtekin & Kuzu, 2016; Gülbahar, Kalelioğlu & Doğan, 2015; MEB, 2016) şeklinde kullanımına da rastlamak mümkündür. Bahsedilen düşünme sürecinde zihne gelen bilginin işlendiği bilindiği için (Miller, 1956), bu çalışma kapsamında “computational thinking” kavramına karşılık olarak “bilgi-işlemsel düşünme” kavramının kullanılması benimsenmiştir.

2. FeTeMM ve Bilgi-işlemsel Düşünme

Uluslararası alanyazında bilgisayar programcılığı gelecek on yılın en gözde mesleklerinden birisi olarak gösterilmektedir. ABD İş Gücü İstatistikleri Bürosuna (Bureau of Labor Statistics) (2015) göre yazılım geliştiriciliği (software developer) alanındaki iş imkanları 2024 yılına kadar %17 artacaktır. Bu artış, iş kollarındaki ortalama artış oranı olan %7’den çok daha fazladır. Bu oranlar aynı zamanda yazılım geliştiriciliği ve bilgisayar programcılığı mesleklerinin gelecekteki konularının önemini de vurgulamaktadır. Programlama becerisi ile bilgi-işlemsel düşünme becerisi arasındaki pozitif ilişki dikkate alındığında (Oluk & Korkmaz, 2016) bu mesleklere olan ihtiyacı karşılamak için gelecekte bilgi-işlemsel düşünme becerisine sahip bireylere daha fazla ihtiyaç duyulacağı ileri sürülebilir.

Bilgi-işlemsel düşünme becerisi ele alındığında bu konuyla ilgili olarak kadınların durumuna da değinmekte yarar vardır. Çünkü kadınların bilgi-işlemsel düşünmeyi gerektiren bilgisayar ile ilgili bölümleri daha az tercih etmeleri ve tercih edenlerin ise bu bölümleri bırakmaları dikkat çekmektedir (Aldağ & Tekdal, 2015). Dahası, FeTeMM (STEM-Science Technology Engineering Mathematics) alanına kadınların ilgisinin giderek daha da düştüğü ileri sürülmektedir. 2013 yılında kadınların bilgisayar ve matematik ile ilgili bütün işlerin sadece %26’sında çalışmakta olduğunun ileri sürülmesi de bu durumu destekler niteliktedir. Ancak istatistikler bu oranın 1990 yılında %35 olduğunu göstermektedir. Benzer şekilde, 2013 yılında programlama lisans derecesi kazananların sadece %18’inin

kadın olduğu görülmektedir. Bu konuyla ilgili istatistiklere bakıldığında bu oranın yaklaşık 30 yıl önce çok daha yüksek olduğu görülmektedir. Örneğin 1985 yılında bu oranın yılda %37 olduğu belirtilmektedir. Genel olarak bakıldığında, tüm bilgisayar programcılarının sadece %24'ünü, yazılım geliştiricilerin ise %20'sini kadınlar oluşturmaktadır (Corbett & Hill, 2015). Bu rakamlar bu konuda cinsiyetler arasında bir dengesizlik olduğunu açıkça göstermektedir. Kadınların bilgisayar ile ilgili bölümleri seçmemesi konusunda çeşitli sosyal-kültürel etkenlerin rolü olabilir (Aldağ & Tekdal, 2015). Ancak bu durumun cinsiyetler arasında bilgi-işlemsel düşünme açısından bir farklılığın olduğu şeklinde yorumlanmasının uygun olmadığı düşünülmektedir. Nitekim öğrencilerin bilgi-işlemsel düşünme becerilerinin araştırıldığı bazı çalışmalarda cinsiyete göre genel olarak anlamlı bir farklılığın olmadığı ortaya konulmuştur (Korkmaz vd., 2015; Oluk & Korkmaz, 2016).

3. Bilgi-işlemsel Düşünme Nedir?

Bilgi-işlemsel düşünmenin tanımı hakkında fikir birliğine varılmadığı pek çok araştırmacı tarafından ifade edilmektedir (Gonzalez, 2015; Grover & Pea, 2013; Kalelioğlu, Gülbahar & Kukul, 2016). Ancak temel noktalarda bir fikir birliğinin olduğunu söylemek mümkündür.

Wing (2008) bilgi-işlemsel düşünmenin problem çözme becerisinin yeni bir ismi olmadığını ve bilgi-işlemsel düşünmenin bir çeşit analitik düşünme olduğunu vurgulamaktadır. ISTE (2015) ise bilgi-işlemsel düşünmeyi; yaratıcılık, algoritmik düşünme, eleştirel düşünme, problem çözme ve işbirliğinin bir birleşimi olarak tanımlamaktadır. Kazımoglu, Kiernan, Bacon ve MacKinnon (2012) ise problem çözme, algoritma inşa etme, hata yakalama, benzetim ve sosyalleşmenin bilgi-işlemsel düşünmenin beş temel becerisi olduğunu belirtmektedir. Bunun yanı sıra Ater-Kranov, Bryant, Orr, Wallace ve Zhang (2010) eleştirel düşünme ve problem çözmenin bilgi-işlemsel düşünmeyle ilgili alanyazında en çok kabul gören iki beceri olduğunu ifade etmektedir. Son olarak, Kalelioğlu vd. (2016) bilgi-işlemsel düşünmenin kapsamını belirlerken en çok kullanılan üç bileşenin sırasıyla; soyutlama, algoritmik düşünme ve problem çözme olduğu sonucuna ulaşmıştır. Alanyazındaki bu tartışma Tablo 2'de özetlenmiştir.

Tablo 2. Bilgi-işlemsel Düşünme Kavramına Yönelik Tanımlamalar

Kaynaklar	Tanımlamalar
Gonzalez, 2015 Grover & Pea, 2013	Üzerinde fikir birliğine varılan bir tanım yoktur.
Wing, 2006, 2008	Bilgi-işlemsel düşünme, problem çözme, eleştirel düşünme, soyutlama, analitik ve algoritmik düşünme gibi farklı süreçleri içermektedir.
Kazimoglu, Kieran, Bacon & MacKinnon, 2012	Bilgi-işlemsel düşünmenin beş temel becerisi “problem çözme, algoritma inşa etme, hata yakalama, benzetim ve sosyalleşme”dir.
ISTE, 2015	Bilgi-işlemsel düşünme yaratıcılık, algoritmik düşünme, eleştirel düşünme, problem çözme ve işbirliğinin bir bileşimidir.
Ater-Kranov, Bryant, Orr, Wallace & Zhang, 2010	Eleştirel düşünme ve problem çözme bilgi-işlemsel düşünme alanının da en çok kabul gören iki beceridir.
Kalelioğlu vd., 2016	Soyutlama, algoritmik düşünme ve problem çözme en çok kabul edilen 3 bileşendir.

Tablo 2’de görüldüğü gibi bilgi-işlemsel düşünme ile ilişkilendirilen pek çok kavram bulunmakta ve bu kavramlar çalışmalara göre büyük değişiklikler göstermektedir. Bu noktada bu kavramları sınıflayan çalışmalara ihtiyaç duyulmaya başlanmıştır. Bu soruna değinen Brennan ve Resnick (2012) bilgi-işlemsel düşünmeyi 3 boyutta ele almıştır (Bkz Tablo 3).

Tablo 3. Bilgi-işlemsel Düşünmenin Boyutları (Brennan & Resnick, 2012)

Boyutlar	Tanımlamalar	Örnekler
Bilgi-işlemsel kavramlar	Tasarımcıların program yazarken kullandıkları kavramlardır	Değişkenler, döngüler, olaylar
Bilgi-işlemsel uygulamalar	Tasarımcıların kavramlarla uğraşırken geliştirdikleri uygulamalardır.	Soyutlama, hata ayıklama, yeniden kullanma
Bilgi-işlemsel bakış açıları	Tasarımcıların kendisi ve dünya hakkında bir bakış açısı oluşturmasıdır.	İfade etme, bağlama, sorgulama

Wing (2011) bilgi-işlemsel düşünmede en önemli ve yüksek seviyedeki düşünce sürecinin soyutlama süreci olduğunu vurgulamıştır. Soyutlama, belirli örneklerden genelleme yapma, örüntüleri tanımlama ve nesnelere gereksiz ayrıntılarını gizleyerek temel özelliklerini yakalamak için kullanılmaktadır. Soyutlama bireye karmaşıklık ile mücadele etme gücü vermektedir. Örneğin, algoritma bir

sürecin soyutlamasıdır. Bilgi-işlemsel düşünme ile ilgili bir başka kavram da otomatikleşmedir. Programların çalıştırılması soyutlamanın otomatikleşmesi olarak değerlendirilebilir.

Bilgi-işlemsel düşünme matematiksel ve mühendisçe düşünme biçimlerini birleştirmektedir. Bilgi-işlemsel düşünme temel olarak matematikten yararlanmaktadır. Ancak yapılan işlemler bilgisayarlar tarafından çalıştırılacağı için işlem gücü bilgisayarların gücü tarafından kısıtlanmaktadır. Matematikte ise böyle bir sınırlama bulunmamaktadır. Bilgi-işlemsel düşünme mühendislikten de yararlanmaktadır, fakat her bilgisayar programı fiziksel çevreye doğrudan uygulanmak zorunda olmadığı için bu programlar ile fiziksel gerçeklik tarafından kısıtlanmayan sanal dünyalar yaratılabilmektedir. Mühendislikte ise her zaman tasarlanan fikrin/ürünün gerçek hayata uygulanma zorunluluğu bulunmaktadır. Bu bağlamda, günlük hayatımızı şekillendiren şeylerin sadece yazılım veya donanımlar olmadığı; kullandığımız bilgi-işlemsel kavramların da etkili olduğu ifade edilebilir (Wing, 2008). Bu noktada ABD Ulusal Araştırma Konseyi'nin (National Research Council) (2010) "dağıtılmış bilgi-işlemsel düşünme" (distributed computational thinking) ifadesine yer verdiğini belirtmek gerekir. Bu kavram bilgi-işlemsel düşünmeyi bilgisayar bilimlerinden ayırmakta ve bu düşünme biçimini herkesin günlük hayatında problem çözmek için kullanabileceğini ifade etmektedir.

Bilgi-işlemsel düşünme kavramı söz konusu olduğunda alanyazında genellikle Wing'in yaptığı tanımın ele alındığı görülmektedir. Bu tanıma göre, bilgi-işlemsel düşünme, programlamaya temel olan kavramlardan yararlanan insan davranışlarını anlamak, sistemler tasarlamak ve problemler çözmek için bir yaklaşım benimsemektir (Wing, 2006). Wing bu tanımı "Bilgi-işlemsel düşünme, etkili bir şekilde bilginin işlenmesi için problemlerin çözümlerini ve problemleri açık ve kesin şekilde ifade etmeyi içeren düşünce süreçleridir." şeklinde güncellemiştir (Wing, 2011). Bu tanım Aho (2012) tarafından "Bilgi-işlemsel düşünme, problemlerin çözümünde bilgi-işlemsel adımların ve algoritmaların kullanılmasını içeren düşünce süreçleridir." şeklinde kısaltılmıştır.

4. Bilgi-işlemsel Düşünmeye Yönelik Eleştiriler:

Son yıllarda alanyazında "bilgi-işlemsel düşünme" kavramına yönelik bazı eleştirilerin yapıldığı görülmektedir (Grover & Pea, 2013). Bu eleştiriler ve bunlarla ilgili tartışmalar aşağıdaki gibi özetlenebilir (Bkz. Tablo 4).

Tablo 4. Bilgi-işlemsel Düşünme Kavramına Yönelik Eleştiriler

Eleştiriler	Eleştirilere İlişkin Yapılan Açıklamalar
1. Bilgi-işlemsel düşünme diğer düşünme formlarından yeteri kadar ayırt edilememektedir (Jones, 2011; Lee vd., 2011).	Bu eleştiri kısmen haklıdır. Ancak bilgi-işlemsel düşünme bilgisayar bilimleri alanına özgü olan ve olmayan çeşitli düşünme formlarını içermekte ve bunları insan-bilgisayar işbirliği aracılığıyla günlük problemleri çözmek için kullanmaya imkân tanımaktadır.
2. Bilgi-işlemsel düşünme alana özgü müdür, yoksa disiplinler arası mıdır? (National Research Council, 2011)	Bazı ülkelerde ilköğretim öğretim programına eklenen programlama dersleri ile öğrencilerin bilgi-işlemsel düşünme becerileri geliştirilmeye çalışılmaktadır. Öte yandan, programlama aracılığıyla matematik ve fen gibi dersler de öğretilmeye çalışılmaktadır. Ancak, genel eğilim bilgi-işlemsel düşünmenin disiplinler arası olduğudur.
3. FeTeMM veya bilgisayar bilimleri alanlarında kariyer yapmak istemeyen öğrencilerin bilgi-işlemsel düşünme becerilerini geliştirmeye gerek var mıdır? (Grover & Pea, 2013)	“Matematik ile ilgili bir kariyer planı yapmayan bir öğrenciye matematik öğretmeye gerek var mıdır?” şeklindeki bir soruya verilebilecek cevap burada da kullanılabilir. Tıpkı matematikte olduğu gibi, bilgi-işlemsel düşünme herkesin sahip olması gereken bir düşünme becerisi olarak değerlendirilmektedir.
4. Bilgi-işlemsel düşünmenin net bir tanımı yoktur (Grover & Pea, 2013; Jones, 2011; Kalelioğlu vd., 2016).	Bu eleştiri özellikle bilgi-işlemsel düşünme kavramının ortaya çıktığı ilk yıllarda geçerlidir. Ancak bilgi-işlemsel düşünme ile neyin kastedildiği; ISTE, CSTA (Computer Science Teachers Association-Bilgisayar Bilimleri Öğretmenleri Birliği), NSF ve NRC gibi kurumların ve çeşitli araştırmacıların çabaları ve tanımları sayesinde netleşmeye başlamıştır.
5. Bilgi-işlemsel düşünme ile etik vb. sorunlar çözülemez (Jones, 2011).	Bu eleştiri haklı bir eleştiridir. Bilgi-işlemsel düşünme ile pek çok sorun çözülebilir. Ancak bilgi-işlemsel düşünme ile çözülemeyecek bazı problemler de vardır.

Bilgi-işlemsel düşünme kavramına yönelik eleştirilerden biri bilgi-işlemsel düşünmenin diğer düşünme biçimlerinden yeteri kadar ayırt edilebilir olmadığı yönündedir (Jones, 2011; Lee vd., 2011). Bu eleştiri ile paralel şekilde Czerkowski (2013) yükseköğretim ortamlarında çalışan öğretim tasarımcılarıyla görüşmeler yapmış ve öğretim tasarımcılarının bilgi-işlemsel düşünmeye herhangi bir diğer düşünme becerisi olarak baktığını tespit etmiştir. Bu nedenle bilgi-işlemsel düşünme becerisi, problem çözme ve eleştirel düşünme becerilerine benzer bir şekilde öğretilmek istenmektedir. Bu bağlamda bilgi-işlemsel düşünme becerisinin pek çok diğer düşünme biçimleri ile ilişkili olduğu söylenebilir. Ancak bilgi-işlemsel

düşünme, problem çözme becerisinin yeni bir ismi olarak düşünülmemelidir, çünkü bilgi-işlemsel düşünme problem çözmenin yanı sıra eleştirel düşünme, soyutlama ve algoritmik düşünme gibi farklı süreçleri de içermektedir (Wing, 2008). Bununla birlikte, Diaz-Herrera (2016) bilgi-işlemsel düşünmenin sadece eleştirel düşünme becerilerini değil, güçlü dijital araçlarla birleştirildiğinde insanın entelektüel kapasitesini de arttıracığını belirtmektedir. Kısaca, bilgi-işlemsel düşünmenin işbirliği ve yaratıcılık gibi 21 yy. becerilerini hem tamamladığı hem de yaydığı ifade edilebilir (Bienkowski, Snow, Rutstein & Grover, 2015). Ancak bunun için makineler ile insanların işbirliği yapması gerekmektedir.

Bilgi-işlemsel düşünme kavramına yönelik ikinci eleştiri ise bilgi-işlemsel düşünmenin eğitime, alana özgü bir konu olarak mı yoksa disiplinler arası bir konu olarak mı kaynaştırılacağı konusunda bir fikir birliğinin olmamasıdır (National Research Council, 2011). Kay ve Goldberg (1977) ile Papert (1980) çalışmalarında geliştirdikleri Smalltalk ve LOGO programlama dilleri ile matematik ve fen gibi alanların öğretilebileceğini göstermeye çalışmışlardır. Buna ek olarak, ISTE-CSTA (Computer Science Teachers Association-Bilgisayar Bilimleri Öğretmenleri Birliği) (2011) K-12 düzeyinde bilgi-işlemsel düşünme becerileri ile biyoloji, tarih, dil ve trafik gibi konuları ilişkilendirmeye yönelik bir program hazırlamıştır. Öte yandan, günümüzde pek çok ülkede K-12 seviyesinde verilen programlama dersleri aracılığıyla öğrencilere bilgi-işlemsel düşünme becerilerinin kazandırılmaya çalışıldığı görülmektedir. Bu iki yaklaşımla da öğrencilerin bilgi-işlemsel düşünme becerilerinin geliştirilebildiği görülmektedir. Nitekim Oluk ve Korkmaz (2016) ortaokul öğrencilerinin programlama becerileri ile bilgi-işlemsel düşünme becerileri arasında yüksek seviyede anlamlı bir ilişki olduğunu tespit etmiştir. Ancak alanyazındaki eğilim bilgi-işlemsel düşünmenin bir düşünme biçimi olması nedeniyle disiplinler arası bir konu olarak ele alınmasının gerektiği yönündedir.

Son zamanlarda gündemde olan FeTeMM alanlarının merkezinde bilgi-işlemsel düşünme kavramı yer almaktadır (Henderson, Cortina, Hazzan & Wing, 2007). Bilgisayar bilimleri veya FeTeMM ile ilgili bir kariyer planı bulunmayan öğrencilerin okullarda bilgi-işlemsel yeterliliklerini (computational competencies) geliştirmek için geçerli bir nedenin olup olmadığı konusunda bir tartışma söz konusudur (Grover & Pea, 2013). Bu noktada temel düzeyde matematik bilmenin ileri düzey matematik kariyeri gerektirmediği gibi bilgi-işlemsel düşünme becerilerine sahip olmanın da ileri düzey bir programlama kariyeri gerektirmediği ileri sürülebilir (Korkmaz, Çakır & Özden, 2015). Nitekim Northwestern Üniversitesi "CT-STEM" Computational Thinking-STEM) başlıklı projesi kapsamında lise öğrencileri için var olan FeTeMM derslerinin bir parçası olarak tasarlanan 60 bilgi-işlemsel düşünme etkinliği geliştirmektedir. Bu proje CT-STEM uygulamalarının

4 alanını tanımlamıştır. Bu alanlar; veri analizi, modelleme ve benzetim, bilgi-işlemsel problem çözüme ve sistem düşünmesidir.

Eğitimciler arasında bilgi-işlemsel düşünmenin çeşitli yorumlarının bulunduğu ve bu nedenle net bir tanımının olmadığı belirtilmektedir (Grover & Pea, 2013; Jones, 2011; Kalelioğlu vd., 2016). Ancak bu durum Wing gibi araştırmacıların, ISTE, CSTA ABD Ulusal Bilim Kurumu (National Science Foundation) ve ABD Ulusal Araştırma Konseyi (National Research Council) gibi kurumların katkıları sayesinde aşılmaya başlanmıştır.

Bilgi-işlemsel düşünme kavramına yönelik yapılan son eleştiri ise bilgi-işlemsel düşünmenin etik vb. konulardaki sorunları çözmede yararlı olamayacağı üzerinedir (Jones, 2011). Bu eleştiri aslında bilgi-işlemsel düşünme ile her problemin çözülebileceğinin düşünülmesine yöneliktir. Bilgi-işlemsel düşünmenin bilgisayar bilimleri ile ilgili problemlerin yanı sıra günlük hayata dair pek çok problemi çözmekte yararlı olabileceği iddia edilebilir. Ancak bilgi-işlemsel düşünme aracılığıyla çözülemeyecek veya çözümünde zorlanılacak etik vb. bazı sorunlar da mutlaka bulunmaktadır.

5. Bilgi-işlemsel Düşünme Kavramının Tarihçesi

Bilgi-işlemsel düşünme (computational thinking) ifadesi ilk defa Papert (1996) tarafından kullanılmıştır. Ancak, son 10 yılda daha sık duyulmaya başlanan bu kavramın ortaya ilk çıkışının yaklaşık 50 yıl öncesine dayanmakta olduğu ileri sürülebilir (Bkz. Tablo 5).

Tablo 5. Bilgi-işlemsel Düşünme Kavramının Gelişiminin Kısa Tarihçesi

Tarihler	Açıklamalar
Perlis, 1962	Bütün alanlardaki üniversite öğrencilerine programlamanın öğretildiği savunulmuştur.
Kay & Goldberg, 1977	Nesne tabanlı Smalltalk programlama dili ve Dynabook aracıyla problem çözümlerin kolaylaşabileceği ve matematik, fen ve sanat gibi alanların çocuklar da dâhil herkese öğretilebileceği iddia edilmiştir.
Papert, 1980, 1991	LOGO programlama dili aracılığıyla küçük bir robot olan kaplumbağanın (Turtle) hareket ettirilmesine imkân tanınarak öğrencilerin problem çözümlerine ve böylelikle matematiksel kavramları öğrenmelerine yardımcı olunmuştur.
diSessa, 2000	Bilgi-işlemsel okuryazarlık (computational literacy) kavramı ortaya atılmıştır. Burada matematik ve fen gibi alanların öğretimi için “bir ortam olarak programlama” kullanımı vurgulanmıştır.

Bogost, 2005	İşlemsel okuryazarlık (procedural literacy) kavramı detaylı biçimde açıklanmıştır. Bilgi-işlemsel düşünmeye çok benzer olan bu kavram programlama ile problem çözüme ve oyun ile ilişkilendirilmiştir.
Wing, 2006, 2008	Bilgi-işlemsel düşünme kavramının yeniden gündeme gelmesinde en çok katkısı olan isimlerin başında gelmektedir. Wing'in yayınlarında bilgi-işlemsel düşünme kavramının 21. yüzyılda bir bireyin mutlaka sahip olması gereken bir özellik olduğunun altı çizilmiştir.

Bilgi-işlemsel düşünme kavramının tarihine bakıldığında Perlis'in (1962) bütün alanlardaki üniversite öğrencilerine programlamanın öğretilmesi gerektiğini savunduğu bilgisiyle karşılaşılmaktadır. Perlis programlamayı, "Hesaplama Kuramını (Theory of Computation)" anlamaya yönelik bir adım olarak görmüştür. Hesaplama kuramına göre, programlama aracılığıyla öğrencilerin ekonomi ve hesap (calculus) gibi geniş yelpazedeki konuları anlamaları yeniden şekillendirilebilmektedir. Perlis'in yanı sıra, Kay ve Goldberg (1977) nesne tabanlı Smalltalk programlama dili ve Dynabook aracıyla problem çözümenin kolaylaşabileceğini ve matematik, fen ve sanat gibi alanların her yaş grubundaki bireye öğretilabileceğini savunmuştur. Bu gelişmeler nedeniyle işlemsel okuryazarlık (procedural literacy) kavramı dile getirilmeye başlanmıştır (Sheil, 1980). Bogost (2005) tarafından ayrıntılı bir şekilde anlatılan bu kavramın bilgi-işlemsel düşünme ile benzer olduğu görülmektedir. Ancak bu kavram çoğunlukla video oyunları ve diğer bilgi-işlemsel ortam ürünleri (computational media artifacts) oluşturma ile ilişkilendirilmektedir (Grover & Pea, 2013).

Bilgi-işlemsel düşünmenin tarihçesinde etkili olan isimlerden biri de Seymour Papert'tir. Papert, LOGO programlama dili aracılığıyla küçük bir robot olan kaplumbağanın (Turtle) hareket ettirilmesine imkân tanımıştır. Bu kaplumbağanın kuyruğu hareket ettikçe çizim yapmaktadır (Papert, 1980, 1991). Bu şekilde öğrencilerin işlemsel düşünerek (procedural thinking) karşılaştıkları problemleri çözmelerine ve böylelikle matematiksel kavramları öğrenmelerine yardımcı olmuştur.

diSessa (2000) bilgi-işlemsel okuryazarlık (computational literacy) kavramını ortaya atmıştır. Bu kavram her ne kadar bilgi-işlemsel düşünme gibi dijital çağ becerilerini vurgulasa da, diSessa bilgi-işlemsel düşünmenin programlama ortamı gibi materyal boyutunu bilişsel ve sosyal boyutlarından ayırmaktadır. Ayrıca, diSessa matematik ve fen gibi alanların öğretimi için "bir araç olarak programlama" kullanımını vurgulamaktadır.

Bilgi-işlemsel düşünme kavramının yeniden gündeme gelmesinde en çok katkısı olan isimlerden birisi Jeannette M. Wing'dir. Wing (2006, 2008) çalışmaları

rında bilgi-işlemsel düşünmenin bir bireyin sahip olması gereken bir özellik olduğunu vurgulamakta ve bu kavramı programlama ile ilişkilendirmektedir.

6. Bilgi-işlemsel Düşünme ve Eğitim Sistemi

Wing (2008) gelecek yıllarda bilgi-işlemsel düşünmenin eğitimin ayrılmaz bir parçası olacağını ve bu nedenle bilgi-işlemsel düşünme eğitiminin çocukluğun ilk yıllarında verilmesi gerektiğini ifade etmektedir. Nitekim Lawanto (2016) öğrencilere bilgi-işlemsel düşünmeyi öğretme ile ilgili çabaların son zamanlarda arttığını belirtmektedir. Ancak bilgi-işlemsel düşünme söz konusu olduğunda bazı kaygılar ve sorular gündeme gelmektedir. Bunlar üç başlıkta özetlenebilir.

1. Bilgi-işlemsel düşünme çocuklara etkili şekilde nasıl öğretilbilir? (Guzdial, 2008; Kalelioğlu vd., 2016; Wing, 2008)
2. Öğrenme sürecinde çocukların yetenekleri geliştikçe, öğretilcek kavramların sıralaması nasıl olmalıdır? (Wing, 2008)
3. Kavram öğretiminde kullanılacak araçlar öğrenme süreciyle nasıl bütünleştirilebilir? (Wing, 2008)

Yukarıdaki sorulara ek olarak bilgi-işlemsel düşünme becerisinin kazanılıp kazanılmadığının nasıl ölçüleceği ve bilgi-işlemsel düşünmenin bireylerin bilişsel süreçleri açısından ne ifade ettiği de eklenebilir.

Korkmaz vd. (2015) sınıf düzeyi ilerledikçe öğrencilerin bilgi-işlemsel düşünme becerilerinin düştüğünden hareketle bu sorulara yanıt vermenin önemli olduğunu vurgulamaktadırlar. Bu çalışmada, öğrencilerin aksine, bireylerin iş hayatlarında bilgi-işlemsel düşünme becerilerini zamanla geliştirdikleri sonucuna ulaşılmıştır. Bu bulgu bilgi-işlemsel düşünme becerilerinin iş hayatında kritik olduğunu ve bu nedenle bireylerin bu becerileri kendi kendilerine zamanla geliştirdikleri şeklinde yorumlanmıştır. Bilgi-işlemsel düşünme becerisinin eğitim-öğretim süreci boyunca neden geliştirilemediği hala bir soru işareti olarak durmaktadır. Czerkowski (2013) bilgi-işlemsel düşünme becerisinin geliştirilmesinin sadece bilgisayar bilimleri öğretim programının hedefi olmaması gerektiğini belirtmektedir. Aksine, bilgi-işlemsel düşünmeyi geliştirme hedefinin bütün akademik alanların öğretim programlarında bir kazanım olarak yer almasının uygun olacağı ifade edilmektedir.

Bilgi-işlemsel düşünmenin geliştirilmesi için yeni yaklaşımlar ortaya çıkmaktadır. Bu yaklaşımlardan dikkat çeken bir tanesi ABD'de College Board (üniversite kurulu) isimli bir kurum tarafından geliştirilmiştir. College Board (2016) lise öğ-

rencilerinin üniversite seviyesinde bilgisayar bilimleri dersleri almalarına imkân tanımak için “İleri Düzeyde *Bilgisayar Bilimleri İlkelerinin Ders ve Sınav Açıklamalarında Yer Alması (Advanced placement computer science principles course and exam description)*” isminde bir program geliştirmiştir. Bilgisayar bilimlerinin ilk yarıyılındaki derslere eşit olması hedeflenen bu dersler ile öğrencilerin yaratıcılık ve bilgi-işlemsel düşünme gibi becerilerinin geliştirilmesi planlanmaktadır. Programlama araçlarından bağımsız olarak geliştirilen bu ders problem çözmek için bilgi-işlemsel ürünlerin oluşturulmasını kapsamaktadır. Dersin içeriği yaratıcılık, soyutlama, veri ve bilgi, algoritma, programlama, internet ve küresel etki olmak üzere 7 başlığa ayrılmıştır. Dersin değerlendirmesi ise süreç içerisinde verilen performans görevleri ve dersin sonundaki başarı testinden oluşmaktadır.

Kalelioğlu vd. (2016) bilgi-işlemsel düşünmenin bir problem çözme süreci olarak ele alınabileceğini belirtmiştir. Bu bağlamda klasik bir problem çözme sürecinin aşamaları ile bilgi-işlemsel düşünmenin bileşenlerinin eşlendiği 5 aşamadan oluşan bir model ortaya atmıştır. Bu modelin bileşenleri sırasıyla; (1) problemi tanımlamak, (2) veriyi toplamak, temsil etmek ve analiz etmek, (3) çözümleri üretmek, seçmek ve planlamak, (4) çözümleri uygulamak, (5) çözümleri değerlendirmek ve geliştirmeye devam etmektir. Problem çözme süreci ile bilgi-işlemsel düşünmenin bileşenlerinin eşleştirilmesine örnek vermek gerekirse, problem çözme sürecinin ilk aşaması bilgi-işlemsel düşünmenin soyutlama ve ayrıştırma (decomposition) bileşenleriyle, sürecin son aşaması ise test etme, hata ayıklama ve genelleme bileşenleriyle eşleştirilmektedir. Bu modelin geliştirilen en iyi model olmayabileceği, her şeyi kapsamadığı ve bu nedenle geliştirilmeye açık olduğu belirtilmiştir. Ancak modelin bilgi-işlemsel düşünmeyi geliştirmeye yönelik hem dijital hem de dijital olmayan problem çözme etkinliklerin hazırlanmasında katkı sağlayabileceği öne sürülmüştür.

Çocukların bilgi-işlemsel düşünmelerinin gelişimini desteklemek için dünya çapında çeşitli etkinlikler düzenlenmektedir. Bu etkinliklerin başında Kodlama Saati (Hour of Code) etkinliği gelmektedir. Code.org tarafından düzenlenen kodlama saati etkinliği 180’den fazla ülkede 10 milyonlarca öğrenciye ulaşmıştır (Code.org, 2017). Bunun yanı sıra “bilge kunduz uluslararası enformatik ve bilgi-işlemsel düşünme etkinliği (Bebras International Challenge on Informatics and Computational Thinking)” de dünya genelinde bilgi-işlemsel düşünmeyi her yaşta öğrenciye kazandırmak için düzenlenmektedir. Bu etkinlik 2014 yılından beri Türkiye’de de düzenlenmektedir. Bu tip geniş kapsamlı etkinliklerle öğrencilere programlamayı sevdirmek ve giriş düzeyinde programlama öğretmek hedeflenmektedir.

Sentance ve Csizmadia'nın (2016) öğretmenlerin programlamayı öğretmede kullandıkları stratejileri incelediği çalışmasında 5 temel strateji belirlemiştir. Bunlar;

1. **Dijital olmayan tip etkinlikler (Unplugged type activities):** Bilgisayar ile ilgili kavramları öğretmek için sınıfta fiziksel ve kinestetik olarak elektronik aletler kullanılmadan gerçekleştirilen etkinliklerdir.
2. **Bağlama oturtma etkinlikleri (Contextualising activities):** Öğretmenlerin programlama içeriklerini öğretim programının başka yönleriyle ve/veya gerçek yaşam ile ilişkilendirmesini içerir.
3. **İşbirlikli öğrenme (Collaborative learning):** Öğretmenlerin sınıfta işbirlikli çalışma stratejileri kullanmasıdır.
4. **Bilgi-işlemsel düşünmeyi geliştirmek (Developing computational thinking):** Öğretmenlerin çeşitli öğrenme etkinlikleri aracılığıyla öğrencilerin soyutlama, algoritmik düşünme ve problem çözme gibi bilgi-işlemsel düşünme kavram ve süreç becerilerini geliştirmesidir.
5. **Programlama görevlerini kademeli desteklemek (Scaffolding programming tasks):** Öğretmenlerin, öğrencilerin program yazabilmelerini sağlamak için görevleri kademeli olarak desteklemelerini ifade eder.

7. Bilgi-işlemsel Düşünme ve Dijital Oyun Tabanlı Öğrenme

Oyun tabanlı öğrenme yaklaşımı teknolojinin gelişmesi ile birlikte özellikle 90'lardan itibaren eğitim alanındaki etkisini artırmaya başlamıştır. Bu yöntemdeki amaç zaten yoğun olarak oyun oynadığı bilinen öğrencilerin oyuna yönelik motivasyon/ilgilerini öğrenme ortamına aktarmak suretiyle öğrenmenin kalitesini arttırmaktır.

Oyun tabanlı öğrenmenin programlama öğretiminde kullanılabileceği Perlis'in (1962) çalışmasında kuramsal olarak görülmektedir. Öte yandan Papert'in (1980) yaklaşımının da bir çeşit oyun tabanlı öğrenme olduğu iddia edilebilir. Kuruvada, Asamoah, Dalal ve Kak (2010) dijital oyunlar ilgi çekici ve bağlayıcı olduklarından programlama öğretimi amacıyla kullanımının kadınları da içeren çok az temsil edilen grupların programlamaya olan ilgilerini arttırabileceğini belirtmiştir. Bu noktada özellikle K-12 seviyesinde oyunların bir öğretim aracı olarak kullanımı konusunda artan bir ilgi vardır (Isaac & Babu, 2016) çünkü oyunlar aracılığıyla bilgi-işlemsel düşünme becerilerinin geliştirilebileceği iddia edilmektedir (Bogost, 2005). Kalelioğlu vd. (2016) bilgi-işlemsel düşünme ile ilgili çalışmaların en çok temel aldığı iki kuramdan birinin oyun tabanlı öğrenme olduğunu belirtmiştir.

Alanyazında bilgi-işlemsel düşünme becerilerini geliştirmeyi ve giriş düzeyinde programlama öğretmeyi hedefleyen iki yaygın yaklaşım vardır. Bunlar; oyun geliştirme işlemi aracılığıyla öğrenme ve oyun oynarken programlama aracılığıyla öğrenme yaklaşımlarıdır. Oyun geliştirme, öğrencileri karmaşık problem çözmeye bağlayan yaratıcı bilgi-işlemsel ürünleri tasarlamamanın bir yolu olarak giderek daha çok kullanılmaktadır (Bienkowski vd., 2015). Bu yöntemde amaç oyun geliştirmeyi öğrenmek değil, oyun geliştirme süreci aracılığıyla soyut programlama kavramlarını öğrenmektir (Kazimoglu vd., 2012). Bu yaklaşımla öğrenenlerin çok az beceri gerektiren oyun geliştirme araçları ile dijital oyunların prototiplerini hızlı bir şekilde geliştirmelerine imkân tanınır. Bu yaklaşım; bilgisayar bilimleri kavramlarını öğrenmeyi desteklemek (Denner, Werner & Ortiz, 2012), öğrencileri problem çözme görevlerine bağlamak (Wu & Richards, 2011) ve öğrencilerin bilgi-işlemsel düşünme becerilerini geliştirmek amacıyla uygulanabilmektedir (Kuruvada vd., 2010). Bu çeşit ortamlara Gamestarmechanic, Kodu, Codingame, Stencyl ve Gamelooper vb. ortamlar örnek olarak verilebilir.

İkinci yöntemde ise insanların zaten oyuna yönelik var olan motivasyonları programlama eğitimine aktarılmaya çalışılmaktadır. Öğrencilerin bilgi-işlemsel düşünme becerilerini geliştirmenin etkili bir yolu onları bağlayıcı ve motive edici ortamlara maruz bırakmaktır. Bu tip ortamlarda öğrenciler bilgi-işlemsel düşünme ile ilgili fikirler, temsiller, kavramlar ve veriler hakkında düşünmeye yöneltiler ve teşvik edilebilirler (Wu & Richards, 2011). Örnek olarak, Kazimoglu, Kieranan, Bacon ve MacKinnon (2011) bireylerin hem bulmaca çözmeye hem de oyun oynama isteklerinden yararlanarak, bir bulmaca çözmeye oyunu aracılığıyla, bireylere giriş düzeyinde programlama öğretmişlerdir. Benzer bir yaklaşımı code.org web sitesinin de benimsediği görülmektedir. Scratch, Alice, CodeCombat, Lightbot, Hopscotch, Hackercan, Tynker ve Greenfoot gibi ortamlar bu amaçla kullanılacak ortamlara örnek olarak verilebilir. Bu ortamların bazılarında oyun da geliştirilebileceğini belirtmekte yarar vardır. Bu ortamların yanı sıra; Glitchspace, Codespell, Screeps ve Human resource machine gibi steam (buhar) dijital oyun dağıtım platformunda satılan ve yetişkinlere programlama öğretimi için kullanılacak pek çok dijital oyun bulunmaktadır.

Programlama öğretimi ve bilgi-işlemsel düşünme becerilerinin geliştirilmesinde başvurulabilecek yazılımlara ek olarak, pek çok robotik araç ve elektronik lego kullanılmaktadır. Bunlara; Kano, Littlebits, Osmo, Micro:bit, Hummingbird, Arduino, Raspberrypi, Mindstorm ve Dash & Dot gibi araçlar örnek olarak verilebilir. Bu araçlar çoğunlukla ya bütünleşmiş bir yazılım geliştirme aracı ya da yukarıdaki paragraflarda bahsedilen Scratch gibi bir araç ile kullanılır. Bilgi-işlemsel düşünmeyi artırma çabalarının büyük bir çoğunluğunun ilköğretim düzeyinde

olduğu ve bu yaştaki çocukların henüz somut işlemler dönemine geçemedikleri bilinmektedir. Bu noktada robotik gibi yöntemler öğrencilerin hem bilgi-ışlemsel düşünme kavramlarını somutlaştırmalarına hem de programlamaya motive olmalarına yardımcı olabilmektedir.

8. Bilgi-ışlemsel Düşünmenin Ölçülmesi

Alanyazında bilgi-ışlemsel düşünmenin nasıl ölçülmesi gerektiği hakkında bir uzlaşma bulunmamaktadır (Gonzalez, 2015; Kalelioğlu vd., 2016). Ayrıca, özellikle Türkçe alanyazında, bu amaçla geliştirilmiş yeterli kadar araç da yoktur. Öte yandan bilgi-ışlemsel düşünmenin ölçülmesi için çeşitli yaklaşımlar bulunmaktadır (Bkz. Tablo 6).

Tablo 6. Bilgi-ışlemsel Düşünmenin Ölçülmesine İlişkin Yaklaşımlar

Çalışmalar	Başvurulan Yaklaşımlar
Korkmaz, Çakır & Özden, 2015	Ölçek
College Board, 2016; Gonzales, 2015	Çoktan Seçmeli Test
College Board, 2016; Dr. Scratch, 2014	Performans Değerlendirme
Denner & Werner, 2011	Peri Değerlendirme
Brennan & Resnick, 2012	3 Anahtar Boyuta göre Değerlendirme
Grover, 2015	Değerlendirmeler Sistemi (Systems of Assessment)

Bilgi-ışlemsel düşünmenin ölçülmesinde kullanılan yaklaşımlardan biri ölçek kullanmaktır. Korkmaz, Çakır ve Özden (2017) genel olarak yetişkin öğrenciler olarak ifade edilebilecek olan bireylerin bilgi-ışlemsel düşünme becerilerini ölçmek için bir ölçek geliştirmiştir. Çalışma kapsamında çeşitli düzeylerde örgün veya uzaktan eğitim yoluyla öğrenim görmekte olan toplam 1306 öğrenciden veri toplanmıştır. Geliştirilen ölçek 29 madde ve 5 faktörden oluşmaktadır. Bu ölçek yaratıcılık, algoritmik düşünme, işbirliklilik, eleştirel düşünme ve problem çözme faktörlerinden oluşmaktadır. Ölçeğin Cronbach Alpha güvenirlik katsayısının .822 olarak hesaplandığı raporlanmıştır.

Korkmaz, Çakır ve Özden (2015), Korkmaz vd.'nin (2017) "Bilgi-ışlemsel Düşünme Ölçeği"ni geliştirmek için verileri örgün veya uzaktan eğitim yoluyla eğitim alan ön lisans, lisans ve pedagojik formasyon öğrencilerinden topladığını belirtmiştir. Bu nedenle ölçeğin ortaokul düzeyine uyarlanması gerektiği ifade edilmiştir. Bu amaçla ölçek 241 ortaokul öğrencisine uygulanmıştır. Uyarlanan ölçeğin 22 madde ve 5 faktörden oluştuğu sonucuna ulaşılmıştır. Bu ölçeğin faktörleri ile daha önce bahsedilen ölçeğin faktörleri aynıdır. 5'li likert tipindeki (1=Nadiren,

5=Her zaman) ölçeğin genel Cronbach Alpha güvenilirlik katsayısı .809 olarak hesaplanmıştır.

Bilgi-işlemsel düşünmenin ölçek ile ölçülmesine yönelik bazı eleştiriler de bulunmaktadır. Örneğin, bilgi-işlemsel düşünmeyi ölçmek için nitel ve daha derin yaklaşımlara ihtiyaç olduğu (Denner & Werner, 2011) belirtilmektedir. Bu noktada çoğu zaman öğrencilerin bilgi-işlemsel düşünmesini ölçmek için üst düzey düşünme becerisinin göstergesi olarak son ürün kullanılmaktadır (Denner vd., 2012). Bu duruma örnek olarak, Denner ve Werner (2011) tarafından kullanılan bilgi-işlemsel düşünmenin değerlendirilmesinin başka bir yöntemi olan peri değerlendirme (fairy assessment) yöntemi verilebilir. Bu yöntem Alice programlama ortamında ortaokul öğrencileri seviyesinde Werner, Denner, Campe ve Kawamoto (2012) tarafından da kullanılmıştır. Bu yaklaşım oyun geliştirme ortamlarına çeşitli görevler yerleştirilmesini içermektedir. Programlama ile ilgili bu görevler puanlanır ve bu puanlar bilgi-işlemsel düşünme ile ilgili kavramların birer ölçümü olarak yorumlanmaktadır.

Bilgi-işlemsel düşünmenin değerlendirilmesine yönelik bir başka yöntem de Dr. Scratch uygulamasıdır (Dr. Scratch, 2014). Dr. Scratch, Scratch ile kodlanan projelere dönüt veren web tabanlı ücretsiz bir araçtır. Bilgi-işlemsel düşünme yeteneğinin çeşitli boyutlarının geliştirilmesine yönelik olarak geliştirilen uygulama, projelere 0 ile 21 arasında puan vermektedir. Uygulama bunu temelde projenin kaynak kodunu inceleyerek yapmaktadır. Uygulama; kötü programlama alışkanlıkları, kod tekrarları ve hiç çalışmayan kodlar gibi durumları tespit etmektedir.

College Board (2016) bilgi-işlemsel düşünme ve “7 Büyük Fikir” (7 Big Ideas) konusunda değerlendirmeye iki açıdan yaklaşmaktadır. Bunlardan birincisi College Board tarafından sunulan dersin sonunda çoktan seçmeli bir başarı testinin uygulanmasıdır. Bu amaçla kurul tarafından 22 soruluk bir örnek test hazırlanmıştır. İkincisi ise ders boyunca bilgi-işlemsel ürünlerin oluşturulmasıdır. Bu amaçla kurul tarafından iki tane performans görevi oluşturulmuştur. Üniversite seviyesindeki bu değerlendirmede iki yaklaşım bir arada kullanılmaktadır. Bilgi-işlemsel düşünmeyi ölçmek için geliştirilmiş başka başarı testleri de mevcuttur. Gonzalez (2015) 12–13 yaşındaki öğrenciler için bir bilgi-işlemsel düşünme başarı testi oluşturmuştur. Çoktan seçmeli olarak tasarlanan ve 5 boyutlu olan test 4 şıktan oluşmaktadır. Bu ölçme aracı toplam 28 maddeden oluşmaktadır.

Brennan ve Resnick (2012) bilgi-işlemsel düşünmeyi 3 boyuta (bilgi-işlemsel kavramlar, bilgi-işlemsel uygulamalar ve bilgi-işlemsel bakış açıları) ayırdığı çalışmasında, bilgi-işlemsel düşünmenin 3 boyutunun proje analizi, ürün tabanlı görüşmeler (artifact-based interviews) ve tasarım senaryoları ile nasıl ölçülebileceği

üzerine odaklanmıştır. Çalışma sonucunda bilgi-işlemsel düşünmenin programlama aracılığıyla ölçülmesi noktasında; daha fazla öğrenmeyi desteklemek, ürünleri içermek (incorporating artifacts), süreçleri aydınlatmak, birden fazla noktada kontrol yapmak, birden fazla yolla bilmeye değer vermek ve birden fazla bakış açısı içermek olmak üzere 6 öneride bulunulmuştur.

Bilgi-işlemsel düşünmenin tek bir ölçme yöntemi ile ölçülemeyeceğini belirten Grover (2015) bilgi-işlemsel düşünmenin daha derin öğrenilmesi için birden fazla ölçme aracını içeren değerlendirmeler sistemi (Systems of Assessments) isminde bir yaklaşım önermiştir. Yaparak öğrenmeyi temel alan bu yaklaşımda programlama algısı gibi duyuşsal boyuttaki değişkenler, öğrenmenin transferi ve algoritmik düşünme becerisinin süreç içindeki gelişimi gibi bilişsel boyuttaki değişkenler; grup çalışması, derecelendirme anahtarı, açık uçlu ve çoktan seçmeli sorular ile değerlendirilmektedir.

9. Bilgi-işlemsel Düşünme ve Zihinsel Süreçler

Bilgi-işlemsel düşünme ve programlama söz konusu olduğunda zihinsel süreçlere ilişkin çalışmaların oldukça az sayıda olduğu görülmektedir. Programlamada zihinsel süreçleri ele alan çalışmaların uzman/acemi karşılaştırmalarına tarihsel olarak özel bir ilgi gösterdikleri söylenilebilir (Begel, 2014). Metin tabanlı (text-based) programlama ortamlarına odaklanan bu çalışmalar özellikle bilgi-işlemsel düşünmenin hata ayıklama bileşenine odaklanmaktadır. Bunun yanı sıra alanyazında, program kavrama ile ilgili çalışmalara da rastlanılmaktadır.

Program kavrama doğası itibariyle doğal dil okumadan oldukça farklıdır. Program kavramı sırasında daha uzun göz odaklanma süreleri ve geriye gidip tekrar okumalar söz konusudur (Busjahn, Schulte, & Busjahn, 2011). Bunun nedeni bir programı kavramanın doğrusal olmayan bir süreç olduğu gerçeğidir (Busjahn vd., 2015). Daha uzun göz odaklanma süreleri ise program kavramanın daha fazla yoğunlaşma gerektirdiği şeklinde yorumlanabilir.

Bir programı kavramak söz konusu olduğunda işaret (beacon) kavramına değinmekte yarar bulunmaktadır. İşaretler belli bir işlemin veya yapının göstergesidir. Bir değişkenin değerinin başka bir değişkenin tuttuğu değere eşit olup olmadığını kontrol etmek veya bir değişkenin tuttuğu sayıya belli bir sayı ekleyip başka bir değişkene atamak gibi programlama işlemleri işaretlere örnek olarak verilebilir. Bir programı kavramak ve daha sonra denencelerini test etmek için programcılar işaretleri aramaktadırlar (Wiedenbeck, 1986).

Lin, Wu, Hou, Lin, Yang ve Chang (2016) bir hata ayıklama görevinde yüksek performans gösteren öğrencilerle düşük performans gösteren öğrencileri karşılaştırmıştır. Düşük performans gösterenler:

- Programları satır satır okumaktadırlar.
- Programın üst seviye mantığını hızlı bir şekilde anlayamamaktadırlar.
- Programın mantığını takip etmeden şüphelenilen belirli ifadelerle odaklanmaktadırlar.
- Bilgi hatırlamak amacıyla önceki ifadelerle sıklıkla yeniden bakmaktadırlar.
- Elle programlamaya daha fazla zaman ayırmaktadırlar (Lin vd., 2016).
- Programın mantığını kavramadan genellikle deneme-yanılma yoluyla hata aramaktadırlar (Fitzgerald vd., 2008).

Hata ayıklama stratejileri söz konusu olduğunda uzmanlar:

- İşaretleri daha kolay tespit edebilmektedirler ve programı anlamak için işaretlere odaklanmaktadırlar (Wiedenbeck, 1986).
- Hatayı bulmadan önce programı tararken daha fazla zaman harcamaktadırlar (Sharif, Falcone & Maletic, 2012).
- Programı daha mantıklı bir şekilde analiz etmektedirler (Lin vd., 2016).
- İlk olarak kaynak kodu okumaktadırlar (Bednarik, 2012).

10. Sonuçlar ve Öneriler

Bu çalışma kapsamında bilgi-işlemsel düşünme kavramıyla ilgili bir alanyazın taraması yapılmıştır. Bu alanyazın taraması kapsamında “Computational Thinking” kavramının Türkçe karşılığı, tanımı, bu kavrama yönelik eleştiriler ve kavramın gelişim süreci incelenmiştir. Bunların yanı sıra, bilgi-işlemsel düşünme becerisinin geliştirilmesi, ölçülmesi ve zihinsel süreçlerdeki yerine yönelik çalışmalar da incelenmiştir.

“Computational Thinking” kavramının Türkçeye; bilgisayarca düşünme, bilişimsel düşünme, bilgisayarlı düşünme ve bilgi-işlemsel düşünme olarak çevrildiği görülmektedir. Bu Türkçe karşılıkların hepsinin kendi içerisinde geçerli olmalarına rağmen bilgi-işlemsel düşünme ifadesinin daha yaygın olduğu ve “computational thinking” kavramını daha iyi karşıladığı düşünülmektedir. Öte yandan, bir kavramla ilgili farklı kullanımlara İngilizce alanyazında da rastlanılmaktadır. İngilizce alanyazında “Computational Thinking”, “Computational Lite-

racy” ve “Procedural Thinking” kavramlarının benzer şeyleri ifade etmek amacıyla kullanıldıkları görülmektedir. Bu kavramlar arasında en çok rastlanan ve en güncel olanın “Computational Thinking” olduğu anlaşılmaktadır.

Bilgi-işlemsel düşünme kavramının motivasyon, tutum gibi duyuşsal değişkenler açısından çalışıldığı görülmektedir. Bunun yanı sıra bilgi-işlemsel düşünme ve programlamanın; problem çözme, algoritma, mantıksal düşünme, yaratıcı düşünme (Çatlak vd., 2015) gibi bilişsel değişkenler ile ilişkilendirildiğine de sıkça rastlanılmaktadır. Ancak alanyazında bilgi-işlemsel düşünme ve programlama esnasında “bireylerin zihninde ne olduğu” sorusuna doğrudan cevap verebilecek nitelikte çok az sayıda çalışmanın mevcut olduğu görülmektedir. Yani alanyazındaki çalışmalar genellikle zihinsel süreçlerin çıktıları üzerine odaklanarak zihinsel süreçlerin kendilerini ihmal etmektedirler. Lye ve Koh (2014) bu noktada programlama esnasında sesli düşünme protokolünün (think aloud protocol) uygulanabileceğini belirtmiştir. Ancak Busjahn vd. (2014) bu yöntemin eksikliklerinden bahsetmektedir. Bu nedenle sesli düşünme protokolüne ek olarak bir programlama görevi yerine getirilirken fare hareketlerinin bir ekran kaydedici aracılığıyla kaydedilmesi ve göz izleme verisi toplanmasının programlama sırasındaki zihinsel süreçleri açığa çıkarabileceği söylenebilir.

Alanyazında programlama sırasında göz izleme verilerini inceleyen çalışmalar dikkati çekmektedir. Ancak bu çalışmalar genellikle metin tabanlı programlar üzerine odaklanmaktadır. Bu çalışmalarda ayrıca yetişkinlerle çalışılmakta, özellikle hata ayıklama görevleri üzerine odaklanılmakta, uzman/acemi ve başarılı/başarısız öğrenci karşılaştırmaları yapılmaktadır. Genellikle nicel yöntemlerle gerçekleştirilen bu çalışmalar programlama esnasında zihinde neler olduğunu ayrıntılı bir şekilde ortaya koymakla birlikte bu konuda daha fazla çabaya ihtiyaç duyulduğu da görülmektedir. Bu noktada, son 10 yılda ortaya çıkan blok tabanlı programlama dillerini zihinsel süreçler açısından incelemekte fayda olduğu ifade edilebilir. Ayrıca, okullarda zorunlu/seçmeli programlama dersini almakta olan veya bu dersleri alan çocukların zihinsel süreçleri de incelenebilir. İlerideki çalışmalarda hata ayıklama görevlerinin yanı sıra programlama ve bir programın belli bir girdi ile çalıştırıldığında hangi çıktıyı ürettiği ve programın ürettiği belli bir çıktıya hangi girdinin neden olduğu vb. mantıklı düşünme görevleri de verilebilir. Çocukların bu programlama görevlerindeki performansları ekranın kaydedilmesi suretiyle fare hareketlerini izleme, göz izleme, görev tamamlama durumu ve görev tamamlama süreci gibi nicel yöntemlerin yanı sıra sesli düşünme protokolü, programlama görevlerine yönelik yansıma ve görüşme gibi nitel yöntemler de kullanılarak karma bir çalışma kapsamında çalışılabilir. Bunun yanı sıra, Ambrosio, Almeida, Macedo ve Franco (2014) çalışma belleği ile programlama becerisi ara-

sındaki ilişkiye dikkat çekmiştir. Bu bağlamda çalışma belleği, uzamsal bellek vb. bilişsel değişkenlerin programlama becerisi ve bilgi-işlemsel düşünme ile ilişkisi çalışılabilir.

Bilgi-işlemsel düşünme ile ilişkilendirilen diğer bütün değişkenlerin yanı sıra, duyuşsal değişkenlerin de önem taşıdığı ifade edilebilir. Nitekim Román-González, Pérez-González, Moreno-León ve Robles (2016) bilgi-işlemsel düşünme kavramını kişilik özellikleri ile ilişkilendirmiştir. Bu çalışmada bilgi-işlemsel düşünme ile örneğin tecrübeye açıklık (openness to experience) arasında pozitif bir ilişki bulunmuştur. Bilgi-işlemsel düşünme ve programlama becerisinin bu bağlamda incelenmesine yönelik daha fazla çalışmaya ihtiyaç duyulmaktadır.

Bilgi-işlemsel düşünme kavramı ve programlamanın genellikle bilgisayar bilimcileri tarafından çalışıldığı görülmektedir. Ancak Czerkowski ve Lyman'ın (2015) da belirttiği gibi, bilgi-işlemsel düşünme öğretim teknolojileri tarafından da çalışılması gereken bir konudur. Bu durum özellikle ortaokul öğretim programlarına programlama derslerinin konulmasıyla birlikte daha aşikâr bir hal almıştır. Burada programlama etkinlikleri aracılığıyla bilgi-işlemsel düşünmenin geliştirilmesi noktasında oyun tabanlı yaklaşımın ötesinde yeni pedagojik yaklaşımlara ihtiyaç duyulduğu ifade edilebilir.

Bilgi-işlemsel düşünme becerisiyle ilgili çalışmaların genellikle K-12 düzeyine yoğunlaştığı görülmektedir (Kalelioğlu vd., 2016). Oysa bilgi-işlemsel düşünmenin hayatın her yerinde olduğu ve bu nedenle de her meslek kolunun çalışanlarının sahip olması gereken bir düşünme becerisi olduğu ifade edilebilir. Bu nedenle, Czerkowski ve Lyman'ın da (2015) dediği gibi, alanyazında 1. ve 2. sınıf lisans öğrencileri açısından bilgi-işlemsel düşünme ile ilgili çalışmalara ihtiyaç duyulduğu söylenebilir.

Bilgi-işlemsel düşünmeyi ölçme konusunda ölçek, çoktan seçmeli test, proje sonucundaki ürünü değerlendirme (Dr. Scratch vb. araçların kullanımı ile), portfolyo değerlendirme ile görüşme ve tasarım senaryoları gibi yaklaşımların kullanıldığı görülmektedir. Bilgi-işlemsel düşünme doğası itibarıyla kapsamlı bir değişken olduğu için ölçülmesinde tek bir yöntemin kullanılmaması, bunun yerine Grover'ın (2015) da belirttiği gibi bir "değerlendirmeler sisteminin" kullanılması gerektiği önerilebilir. Bir başka nokta ise Türkçe alanyazındaki bilgi-işlemsel düşünmeyi ölçme araçlarının yetersizliğidir. Özellikle ilköğretim, ortaöğretim ve üniversite seviyelerinde bilgi-işlemsel düşünmenin farklı boyutlarına göre değerlendirme süreçleri geliştirilebilir.

Bilgi-işlemsel düşünmenin geliştirilmesi için dijital oyunların kullanımı oldukça yaygındır (Kalelioğlu vd., 2016). Bu noktada iki yaklaşım söz konusudur.

Bunlar; bir dijital oyun oluştururken ve bir dijital oyun oynarken programlama öğrenmektir. Bu iki yaklaşımın da avantajları olmakta birlikte bir dijital oyun oluşturma aracılığıyla bilgi-işlemsel düşünmeyi geliştirme yaklaşımının giderek daha da yaygınlaşmaya başladığı görülmektedir. Ayrıca bu yaklaşım bir ürün ortaya koyduğu için öğrencileri daha üretken bir hale getirmektedir. Bunun yanı sıra, bu şekilde öğrenciler hem kendilerinin hem de arkadaşlarının geliştirdikleri oyunları oynayabilirler.

Öğrencilerin bilgi-işlemsel düşünme becerilerini geliştirmek amacıyla programlama etkinliklerinin derslerle kaynaştırılması söz konusu olduğunda genellikle oldukça yüksek ücret gerektiren ya robotik araçlara ya da oyun ortamlarına ihtiyaç duyulmaktadır. Bu yüksek ücretler hem uygulama maliyetlerini arttırmakta hem de uygulamanın sürdürülebilirliğini tehlikeye atmaktadır. Üstelik bu araçların bir kısmının dilinin İngilizce olması, Türkiye'deki öğrencilerin İngilizce düzeylerinin genel olarak düşüklüğü nedeniyle sorun oluşturmaktadır. Wing'in (2008) de belirttiği gibi programlama öğretiminden beklenen, kavramların etkili bir şekilde öğretilmesi ve aracın kavramları anlamının önüne geçmesinin önlenmesidir. Bu soruna çözüm olarak Ben-Ari (1998) öğretmenlere bilgisayarlarla ilgili kavramları öğretmek için hemen bilgisayarlara koşmamalarını tavsiye etmektedir. Bu durum akıllara alanyazında uzun süredir var olan dijital olmayan etkinlik yaklaşımını getirmektedir. Dijital olmayan etkinlikler; sınıfta fiziksel olarak uygulanabilecek, herhangi bir elektrikli alet kullanımı gerektirmeyen ve programlamayı öğretmeyi hedefleyen etkinlikler olarak tanımlanabilir. Dijital olmayan etkinliklerle öğrencilerin kavramları pratik ve somut bir şekilde öğrenmeleri sağlanabilir. Dijital olmayan etkinliklere code.org sitesinde ve Lee, Mauriello, Ahn ve Bederson (2014) tarafından geliştirilen CTArcade yaklaşımında yer verilmiştir. Bu iki yaklaşımda genellikle önce dijital olmayan etkinliklere sonra ise elektronik aletlerin kullanımını gerektiren etkinliklere yer verilmektedir. Bilgi-işlemsel düşünmenin sadece bilgisayar kullanarak yapılan etkinliklerde değil hayatın her yerinde var olduğu düşünüldüğünde dijital olmayan etkinliklerin önemi daha iyi anlaşılabilir. Bu bağlamda ilerideki çalışmalarda dijital olmayan etkinliklerle dijital etkinliklerin nasıl harmanlanabileceği konusu çalışılabilir.

Yansıtma Soruları

1. Bilgi-işlemsel düşünme kavramı şimdiye kadar daha çok hangi açılardan çalışılmıştır? Bu kavramın gelecekte hangi açılardan çalışılması alanyazına daha çok katkı sağlayacaktır? Bu bağlamda bilgi-işlemsel düşünme kavramının önümüzdeki 10 yıl içerisinde nasıl bir eğilim gösterebileceğini tartışınız.
2. Bilgi-işlemsel düşünmenin geliştirilmesi için kullanılacak araçları belirtiniz. Bu araçların kavramların öğretiminin önüne geçmemesi için nelerin yapıldığı ve nelerin yapılmasının uygun olabileceğini tartışınız.
3. Bilgi-işlemsel düşünme kavramı 1960 yılından itibaren farklı isimler altında akademik çevrelerde tartışılmaktadır. Bu tartışma zaman zaman bazı akademisyenlerin çabaları ile bir süreliğine yoğunlaşmakta ve sonra tekrar minimum düzeye inmektedir. Bu durumun nedenlerini tartışınız.
4. “Computational thinking” kavramıyla ilgili olarak alanyazında farklı tanımlar yapılmaktadır. Bu kavramın Türkçeye çevirisinde de farklı karşılıklarının kullanıldığı görülmektedir. Bu kavramla ilgili olarak kendi tanımınızı yapınız ve kavram için nedenleriyle birlikte bir Türkçe karşılık öneriniz.
5. Son 10 yılda bilgi-işlemsel düşünme kavramı ile ilgili pek çok eleştiri ortaya atılmıştır. Bu eleştirilerin haklılık derecelerini tartışınız.

Kaynaklar

- Aho, A. V. (2012). Computation and computational thinking. *The Computer Journal*, 55(7), 832–835. doi:10.1093/comjnl/bxs074
- Aldağ, H., & Tekdal, M. (2015). Bilgisayar kullanımı ve programlama öğretiminde cinsiyet farklılıkları. *Proceeding of 1.Uluslararası Çukurova Kadın Çalışmaları Kongresi* (pp.236-243). Adana, Türkiye.
- Ambrosio, A. P., Almeida, L. S., Macedo, J., & Franco, A. (2014). Exploring core cognitive skills of computational thinking. *Proceedings of the 25th Annual Psychology of Programming Interest Group Workshop (PPIG)* (pp. 25-34). University of Sussex, UK
- Ater-Kranov, A., Bryant, R., Orr, G., Wallace, S., & Zhang, M. (2010). Developing a community definition and teaching modules for computational thinking: Accomplishments and challenges. *Proceedings of the 2010 ACM conference on Information technology education* (pp. 143-148). ACM.
- Barut, E., Tuğtekin, U., & Kuzu, A. (2016). Robot uygulamalar ile bilgi işlemsel düşünme becerilerine bakış. *3rd International Conference on New Trend in Education (ICNTE 2016)*. Seferihisar, İzmir, Türkiye.
- Bednarik, R. (2012). Expertise-dependent visual attention strategies develop over time during debugging with multiple code representations. *International Journal of Human-Computer Studies*, 70(2), 143-155.
- Begel, A. (2014). Applying cognitive theories to novice programmers. *Proceeding of Eye Movements in Programming Education II: Analyzing the Novice's Gaze* (pp. 5-9). Berlin, Germany.
- Ben-Ari, M. (1998). Constructivism in computer science education. *Proceedings of the twenty-ninth SIGCSE technical symposium on computer science education*. Atlanta, Georgia, United States: ACM.
- Bienkowski, M., Snow, E., Rutstein, D. W., & Grover, S. (2015). *Assessment design patterns for computational thinking practices in secondary computer science: A first look* (SRI Technical Report). Menlo Park, CA: SRI International. [Çevrim-içi: <http://pact.sri.com/resources.html>, Erişim tarihi: 10.11.2016.]
- Bogost, I. (2005). Procedural literacy: Problem solving with programming, systems, and play. *Telemidium*, 52(1-2), 32-36.
- Brennan, K., & Resnick, M. (2012). *New frameworks for studying and assessing the development of computational thinking*. Paper presented at the Annual American Educational Research Association meeting, Vancouver, BC, Canada.
- Bureau of Labor Statistics, U.S. Department of Labor (2015). *Occupational Outlook Handbook (2016-17 Edition)*, Software Developers. [Çevrim-içi: <https://www.bls.gov/ooh/>, Erişim tarihi: 10.11.2016.]
- Busjahn, T., Bednarik, R., Begel, A., Crosby, M., Paterson, J. H., Schulte, C., & Tamm, S. (2015). Eye movements in code reading: Relaxing the linear order. *Proceedings of the 2015 IEEE 23rd International Conference on Program Comprehension* (pp. 255-265). IEEE Press.

- Busjahn, T., Schulte, C., & Busjahn, A. (2011). Analysis of code reading to gain more insight in program comprehension. *Proceedings of the 11th Koli Calling International Conference on Computing Education Research* (pp. 1-9). ACM.
- Busjahn, T., Schulte, C., Sharif, B., Begel, A., Hansen, M., Bednarik, R., & Antropova, M. (2014). Eye tracking in computing education. *Proceedings of the tenth annual conference on International computing education research*(pp. 3-10). ACM.
- Code.org. (2017). Kodlama Saati (Hour of Code). [Çevrim-içi: <https://hourofcode.com/tr/gb>, Erişim tarihi: 03.02.2017.]
- College Board. (2016). *AP computer science principles course and exam description* (Including the curriculum framework). NY, New York, USA. [Çevrim-içi: <https://secure-media.collegeboard.org/digitalServices/pdf/ap/ap-computer-science-principles-course-and-exam-description.pdf>, Erişim tarihi: 04.01.2017.]
- Corbett, C., & Hill, C. (2015). *Solving the equation: The variables for women's success in Engineering and Computing* (Library of Congress Control Number: 2015933064). Washington, DC, USA: AAUW. [Çevrim-içi: http://www.aauw.org/aauw_check/pdf_download/show_pdf.php?file=solving-the-equation, Erişim tarihi: 10.11.2016.]
- Czerkawski, B. (2013). Instructional design for computational thinking. In R. McBride & M. Searson (Eds.), *Proceedings of Society for Information Technology & Teacher Education International Conference 2013* (pp. 10-17). Chesapeake, VA: AACE.
- Czerkawski, B. C., & Lyman III, E. W. (2015). Exploring issues about computational thinking in higher education. *TechTrends*, 59(2), 57-65. doi:10.1007/s11528-015-0840-3.
- Çatlak, Ş., Tekdal, M., & Baz F.Ç. (2015). Scratch yazılımı ile programlama öğretiminin durumu: Bir doküman inceleme çalışması. *Journal of Instructional Technologies & Teacher Education*, 4(3), 13-25.
- Denner, J., & Werner, L. (2011). *Measuring computational thinking in middle school using game programming*. Paper presented at the Annual Meeting of the American Educational Research Association (AERA), New Orleans, USA.
- Denner, J., Werner, L., & Ortiz, E. (2012). Computer games created by middle school girls: Can they be used to measure understanding of computer science concepts? *Computers & Education*, 58(1), 240-249. doi:10.1016/j.compedu.2011.08.006.
- Diaz-Herrera, J. L. (2016). Should computer science be a liberal art? *Proceeding of Securing America's future: The vitality of independent colleges*. Miami Beach, Florida, USA: *The Council of Independent Colleges (CIC)*.
- diSessa, A. A. (2000). *Changing minds: Computers, learning, and literacy*. Cambridge: MIT Press.
- Doğan, D., Çınar, M., Bilgiç, H. G., & Tüzün, H. (2015). Sarmal eğitsel oyun tasarımı modeline göre dijital oyun geliştirme süreci: <E-adventure> örneği. *Proceedings of International Play and Toy Congress* (pp. 442-452). Erzurum, Ankara, Türkiye.
- Dr. Scratch. (2014). *Dr. Scratch: Analyze your Scratch project here*. [Çevrim-içi: <http://drscratch.org/>, Erişim tarihi: 23.01.2017.]

- Fitzgerald, S., Lewandowski, G., McCauley, R., Murphy, L., Simon, B., Thomas, L., & Zander, C. (2008). Debugging: Finding, fixing and flailing, a multi-institutional study of novice debuggers. *Computer Science Education*, 18(2), 93-116.
- Gonzalez, M. R. (2015). Computational thinking test: Design guidelines and content validation. *Proceedings of EDULEARN15 Conference (pp. 2436-2444)*. Barcelona, Spain.
- Grover, S. (2015). "Systems of Assessments" for deeper learning of computational thinking in K-12. Paper presented at the Annual Meeting of the American Educational Research Association (AERA 2015). Chicago, USA.
- Grover, S., & Pea, R. (2013). Computational thinking in K-12: A review of the state of the field. *Educational Researcher*, 42, 38-43. doi: 10.3102/0013189X12463051
- Guzdial, M. (2008). Paving the way for computational thinking. *Communications of the ACM*, 51(8), 25-27.
- Gülbahar, Y., Kalelioğlu, F., & Doğan, D. (2015). *Bilge kunduz uluslararası enformatik ve bilgi işlemsel düşünme etkinliği: 2015 yılı uygulama raporu*. [Çevrim-içi: <http://www.bilgekunduz.org/wp-content/uploads/2016/01/bilgekunduz-rapor-2015.pdf>, Erişim tarihi: 14.12.2016.]
- Henderson, P. B., Cortina, T. J., Hazzan, O., & Wing, J. M. (2007). Computational thinking. *Proceedings of the 38th ACM SIGCSE, Technical Symposium on Computer Science Education (SIGCSE '07)* (pp. 195-196). New York, NY: ACM Press.
- Isaac, J., & Babu, S. V. (2016). Supporting computational thinking through gamification. *2016 IEEE Symposium on 3D User Interfaces (3DUI)* (pp. 245-246). IEEE.
- ISTE (2016). *ISTE standards for students*. [Çevrim-içi: <https://www.iste.org/standards/standards/for-students-2016>, Erişim tarihi: 10.11.2016.]
- ISTE (2015). *Computational thinking leadership toolkit* (First Edition). [Çevrim-içi: <https://www.iste.org/explore/articleDetail?articleid=152&category=Solutions&article=Computational-thinking-for-all>, Erişim tarihi: 16.11.2016.]
- ISTE-CSTA. (2011). *Computational thinking: Teacher resources* (2. edition) (Grant No: CNS-1030054). [Çevrim-içi: <https://www.iste.org/explore/articleDetail?articleid=152&category=Solutions&article=Computational-thinking-for-all>, Erişim tarihi: 04.01.2017.]
- Jones, E. (2011). *The trouble with computational thinking*. [Çevrim-içi: <http://csta.acm.org/Curriculum/sub/CurrFiles/JonesCTOnePager.pdf>, Erişim tarihi: 19.11.2016.]
- Kalelioğlu, F., Gülbahar, Y., & Kukul, V. (2016). A framework for computational thinking based on a systematic research review. *Baltic J. Modern Computing*, 4(3), 583-596.
- Kay, A., & Goldberg, A. (1977). Personal dynamic media. *IEEE Computer*, 10, 31-41.
- Kazimoglu, C., Kiernan, M., Bacon, L., & MacKinnon, L. (2012). Learning programming at the computational thinking level via digital game-play. *Procedia Computer Science*, 9, 522-531. doi: 10.1016/j.procs.2012.04.056.
- Kazimoglu, C., Kiernan, M., Bacon, L., & MacKinnon, L. (2011). Understanding computational thinking before programming: Developing guidelines for the design of games to learn introductory programming through game-play. *International Journal of Game-Based Learning*, 1(3), 30-52. doi:10.4018/ijgbl.2011070103.

- Korkmaz, Ö., Çakır, R., & Özden, M. Y. (2017). A validity and reliability study of the Computational Thinking Scales (CTS). *Computers in Human Behavior*.doi: 10.1016/j.chb.2017.01.005. [Çevrim-içi: <http://www.sciencedirect.com/science/article/pii/S0747563217300055>, Erişim tarihi: 20.01.2017.]
- Korkmaz, Ö., Çakır, R., & Özden, M. Y. (2015). Bilgisayarca düşünme beceri düzeyleri ölçeğinin (BDBD) ortaokul düzeyine uyarlanması. *Gazi Journal of Educational Sciences*, 1(2), 143-162.
- Korkmaz, Ö., Çakır, R., Özden, M. Y., Oluk, A., & Sarioğlu, S. (2015). Bireylerin bilgisayarca düşünme becerilerinin farklı değişkenler açısından incelenmesi. *OMÜ Journal of Faculty of Education*, 34(2), 68-87. doi: 10.7822/omuefd.34.2.5.
- Kuruvada, P., Asamoah, D.A., Dalal, N., & Kak, S. (2010). The use of rapid digital game creation to learn computational thinking, 1-9. [Çevrim-içi: <https://arxiv.org/ftp/arxiv/papers/1011/1011.4093.pdf>, Erişimtarihi: 04.01.2017.]
- Kuzu, A. (2016). *Computational thinking: Tanımı, özellikleri, boyutları, kullanımı*. 4. Uluslararası Öğretim Teknolojileri ve Öğretmen Eğitimi Sempozyumu – ITTES 2016 [4th International Instructional Technologies &Teacher Education Symposium – ITTES 2016], “e-Eğitimde Temel Güncel Kavramlara Bir Bakış” Paneli. 6-8 Ekim 2016, Fırat Üniversitesi, Elazığ.
- Lawanto, K. N. (2016). *Exploring trends in middle school students' computational thinking in the online scratch community: A pilot study* (Yayın numarası: 5072) Yayınlanmamış Yüksek Lisans Tezi, Utah Devlet Üniversitesi, Utah, A.B.D.
- Lee, I., Martin, F., Denner, J., Coulter, B., Allan, W., Erickson, J., & Werner, L. (2011). Computational thinking for youth in practice. *ACM Inroads*, 2, 32–37.
- Lee, T. Y., Mauriello, M. L., Ahn, J., & Bederson, B. B. (2014). CTArcade: Computational thinking with games in school age children. *International Journal of Child-Computer Interaction*, 2(1), 26-33. Doi: <http://dx.doi.org/10.1016/j.ijcci.2014.06.003>.
- Lin, Y. T., Wu, C. C., Hou, T. Y., Lin, Y. C., Yang, F. Y., & Chang, C. H. (2016). Tracking students' cognitive processes during program debugging—An eye-movement approach. *IEEE Transactions on Education*, 59(3), 175-186.
- Lu, J. J., & Fletcher, G. H. L. (2009). Thinking about computational thinking. *ACM Special Interest Group on Computer Science Education Conference (SIGCSE 2009)*. Chattanooga, TN, USA: ACM Press. [Çevrim-içi: <http://portal.acm.org/citation.cfm?id=1508959&dl=ACM&coll=portal>, Erişim Tarihi: 29.12.2016.]
- Lye, S. Y., & Koh, J. H. L. (2014). Review on teaching and learning of computational thinking through programming: What is next for K-12? *Computers in Human Behavior*, 41, 51-61.
- Miller, G. A. (1956). The magical number seven, plus or minus two: some limits on our capacity for processing information. *Psychological review*, 63(2), 81.
- National Research Council (2010). *Committee for the Workshops on Computational Thinking: Report of a workshop on the scope and nature of computational thinking*. Washington, DC: National Academies Press.

- National Research Council (2011). *Committee for the Workshops on Computational Thinking: Report of a workshop of pedagogical aspects of computational thinking*. Washington, DC: National Academies Press.
- Oluk, A., & Korkmaz, Ö. (2016). Comparing students' scratch skills with their computational thinking skills in terms of different variables. *I. J. Modern Education and Computer Science*, 11, 1-7. doi: 10.5815/ijmecs.2016.11.01.
- Özden, M. Y. (2015). *Computational thinking = Bilgisayarca düşünme becerileri?* [Çevrim-içi: <http://myozden.blogspot.com.tr/2015/06/computational-thinking-bilgisayarca.html>, Erişim Tarihi: 14.12.2016.]
- Özkeş, B. (2016). *Bilişimsel düşünme temelli ders etkinliklerinin öğrencilerin eleştirel düşünme becerileri ve problem çözme becerilerine yönelik algıları üzerine etkisinin incelenmesi*. Yayınlanmamış yüksek lisans tezi. Mevlana üniversitesi, Konya.
- Papert, S. (1996). An exploration in the space of mathematics educations. *International Journal of Computers for Mathematical Learning*, 1(1), 95-123.
- Papert, S. (1991). Situating constructionism. In I. Harel & S. Papert (Eds.), *Constructionism* (pp. 1-11). Norwood, NJ: Ablex.
- Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. New York, NY: Basic Books.
- Perlis, A. (1962). The computer in the university. In M. Greenberger (Ed.), *Computers and the world of the future* (pp. 180-219). Cambridge, MA: MIT Press.
- Román-González, M., Pérez-González, J. C., Moreno-León, J., & Robles, G. (2016). Does computational thinking correlate with personality? The non-cognitive side of computational thinking. *Proceedings of the Fourth International Conference on Technological Ecosystems for Enhancing Multiculturality* (pp. 51-58). ACM.
- Sayın, Z., & Seferoğlu, S. S. (2016). *Yeni bir 21. yüzyıl becerisi olarak kodlama eğitimi ve kodlamanın eğitim politikalarına etkisi*. Akademik Bilişim, Adnan Menderes Üniversitesi, Aydın.
- Sentance, S. E., & Csizmadia, A. (2016). Computing in the curriculum: Challenges and strategies from a teacher's perspective. *Education and Information Technologies*, 1-27. doi: 10.1007/s10639-016-9482-0
- Sharif, B., Falcone, M., & Maletic, J. I. (2012). An eye-tracking study on the role of scan time in finding source code defects. *Proceedings of the Symposium on Eye Tracking Research and Applications* (pp. 381-384). ACM.
- Sheil, B. (1980). Teaching procedural literacy. *Proceedings of the ACM 1980 annual conference* (pp.125-126). New York: ACM Press.
- Şahiner, A., & Kert, S. B. (2016). Komputasyonel düşünme kavramı ile ilgili 2006-2015 yılları arasındaki çalışmaların incelenmesi. *Avrupa Bilim ve Teknoloji Dergisi*, 5(9), 38-43.
- T.C. Milli Eğitim Bakanlığı Ortaöğretim Genel Müdürlüğü (MEB) (2016). *Bilgisayar bilimi dersi öğretim programı kur 1 - kur 2*. Milli Eğitim Bakanlığı, Ankara.

- T.C. Milli Eğitim Bakanlığı Temel Eğitim Genel Müdürlüğü (MEB) (2017). *Bilişim teknolojileri ve yazılım dersi öğretim programı (Taslak öğretim programı)*. Milli Eğitim Bakanlığı, Ankara. [Çevrim-içi: <http://mufredat.meb.gov.tr/ProgramDetay.aspx?PID=101>, Erişim tarihi: 06.02.2017.]
- Werner, L., Denner, J., Campe, S., & Kawamoto, D. C. (2012). The fairy performance assessment: Measuring computational thinking in middle school. *Proceedings of the 43rd ACM Technical Symposium on Computer Science Education (SIGCSE'12)* (pp. 215-220). New York, NY: ACM.
- Wiedenbeck, S. (1986). Beacons in computer program comprehension. *International Journal of Man-Machine Studies*, 25(6), 697-709.
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-35.
- Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society*, 366, 3717-3725. doi:10.1098/rsta.2008.0118.
- Wing, J. M. (2011). *Research notebook: Computational thinking—What and why?* The Link Magazine, Spring. Carnegie Mellon University, Pittsburgh. [Çevrim-içi: <https://www.cs.cmu.edu/link/research-notebook-computational-thinking-what-and-why>, Erişim tarihi: 09.11.2016.]
- Wing, J. M. (2016). *Computational thinking. 10 Years later*. Microsoft Research Blog. [Çevrim-içi: <https://www.microsoft.com/en-us/research/blog/computational-thinking-10-years-later/>, Erişim tarihi: 12.11.2016.]
- Wu, M. L., & Richards, K. (2011). Facilitating computational thinking through game design. *International Conference on Technologies for E-Learning and Digital Entertainment* (pp. 220-227). Berlin-Heidelberg:Springer.
- YTÜ BÖTE. (2016). *Ortaokul bilgisayar öğretmenleri için programlama dili eğitiminde kullanılan yeni teknolojiler ve metodolojiler*. [Çevrim-içi: <http://www.bilimsenligi.com/ortaokul-bilgisayar-ogretmenleri-icin-programlama-dili-egitiminde-kullanilan-yeni-teknolojiler-ve-metodolojiler.html>, Erişim tarihi: 14.12.2016.]