



Assessment of Computational Thinking within the Scope of Literacy[#]

Barış Şaybak^{1,a,*}, Salih Birişçi^{1,b}

¹Faculty of Education, Bursa Uludağ University, Bursa, Türkiye

*Corresponding author

Research Article

History

Received: 08/10/2023

Accepted: 07/02/2024



This paper was checked for plagiarism using iThenticate during the preview process and before publication.

Copyright © 2017 by Cumhuriyet University, Faculty of Education. All rights reserved.

ABSTRACT

The fact that modern life is technology-oriented and complex leads to the emergence of complex problems. One of the most important 21st century skills to deal with and overcome these challenges is computational thinking. This study investigated the points on which the theory of computational literacy proposed by Andrea diSessa is based on and examined its relationship with computational thinking. When the literature is analysed, it is seen that the boundaries between the concepts of computational thinking and computational literacy are not clearly drawn. For this reason, within the scope of the study, firstly, the development of the concept of computational thinking was given and its sub-dimensions were elaborated within the approaches of various researchers. Then, the boundaries between computational thinking and computational literacy were emphasised and the similarities and differences between them were explained in terms of their characteristics. Finally, the main outlines of the theory of computational literacy were discussed and the studies in the literature of the last 5 years based on this theory were analysed. As a result, the study discussed whether computational thinking can be accepted as a literacy.

Keywords: Computational literacy, computational thinking, literacy, diSessa's computational literacy.

Bilgi İşlemsel Düşünmenin Okuryazarlık Kapsamında Değerlendirilmesi

*Sorumlu yazar

Süreç

Geliş: 08/10/2023

Kabul: 07/02/2024

Bu çalışma ön inceleme sürecinde ve yayımlanmadan önce iThenticate yazılımı ile taranmıştır.

Copyright



This work is licensed under Creative Commons Attribution 4.0 International License

ÖZ

Modern yaşamın teknoloji odaklı ve kompleks bir yapıda olması, taşıdığı problemlerin de karmaşık biçimlerde ortaya çıkmasına neden olmaktadır. Bu zorluklarla başa çıkmak ve bunların üstesinden gelebilmek için işe koşulacak en önemli 21. Yüzyıl becerilerinden biri de bilgi işlemsel düşünmedir. Bu çalışmada, Andrea diSessa tarafından ortaya atılan bilgi işlemsel okuryazarlık teorisinin dayandığı noktalar mercek altına alınarak bilgi işlemsel düşünmeyle olan ilişkisi incelenmiştir. Alanyazın incelendiğinde, bilgi işlemsel düşünme ve bilgi işlemsel okuryazarlık kavramları arasındaki sınırların net bir şekilde çizilmediği görülmektedir. Bu gereğyle, çalışma kapsamında öncelikle bilgi işlemsel düşünme kavramının gelişimine yer verilmiş ve alt boyutları çeşitli araştırmacıların yaklaşımları dahilinde detaylandırılmıştır. Ardından bilgi işlemsel düşünme ile bilgi işlemsel okuryazarlık arasındaki sınırlar üzerinde durularak aralarındaki benzerlik ve farklılıklar sahip olduğu özellikler dahilinde açıklanmıştır. Son olarak, bilgi işlemsel okuryazarlık teorisinin ana hatları ele alınmış ve bu teoriye dayanan son 5 yıla ait literatürde yapılan çalışmalar incelenmiştir. Sonuç olarak, bilgi işlemsel düşünmenin bir okuryazarlık olarak kabul edilip edilemeyeceği hakkında tartışılmıştır.

Anahtar Kelimeler: Bilgi işlemsel okuryazarlık, bilgi işlemsel düşünme, okuryazarlık, diSessa bilgi işlemsel okuryazarlık, kompütasyonel okuryazarlık.

Giriş

Günümüz eğitim ortamlarının yapılandırmacı yaklaşıma evrilmesiyle birlikte, öğrenmeler noktasında öğrencilerin birtakım becerilere sahip olma gerekliliği önemli bir durum olarak kendini göstermektedir. Özellikle içinde yer alınan 21.yüzyıl kapsamında düşünüldüğünde, problem çözme ve üst düzey düşünme şeklindeki becerilerin bu noktada kayda değer pay sahibi olduğunu söylemek yanlış olmayacaktır. Bilgi işlemsel düşünme (BİD), problemi tanımlamaya, anlamaya ve çözmeye yönelik stratejiler geliştirmek için bilgi işlemsel kavramları kullanma becerisidir (Martins vd., 2023). Ele alınan düşünme süreci, bilişsel süreçlerin kullanımını içermekte ve problemleri çözmek için bilgi ve becerilerin etkili bir şekilde kullanılmasını gerektirmektedir. Bu durum, evrensel bir yetkinlik olarak değerlendirildiğinde BİD'in 21. yüzyıl becerilerinden biri olarak kabul edilmesindeki önemini ortaya koymaktadır.

BİD becerilerini bir okuryazarlık olarak ele alan teorilerin varlığı bilinmektedir (diSessa, 2000; Guzdial, 2019; Kafai, Proctor ve Lui, 2019; Vakil, 2018). Okuryazarlıkların gelişimi göz önüne alındığında, orta çağlarda, yalnızca seçilmiş bir grubun okuma yazma bilmesi onları diğer insanlara göre güçlü kılmış ve dünya değiştikçe diğer insanlar için de okuma-yazma bir ihtiyaç haline gelmeye başlamıştır. Dijital okuryazar olmanın sıradan bir hale geldiği günümüzde, dijital okuryazarlık becerileri bile çağımız toplumunun bazı ihtiyaçlarını karşılamakta yetersiz kalabilmektedir. Yeni okuryazarlıkların kapısının aralanmaya başlandığı modern hayatın geldiği noktada, bilgi işlemsel okuryazarlığın (diSessa, 2000, 2018), birçok disipline ait olayları modellemek ya da simüle etmek amacıyla bir araç olarak kullanımı önem kazanmaktadır. Bununla birlikte söz konusu düşünme biçimi okullar kapsamında yer verilmekle birlikte Fransa, Finlandiya ve Kanada gibi eğitim yönünden gelişmiş bazı ülkelerde bilgi işlemsel düşünme bir okuryazarlık olarak matematik eğitim programlarına entegre edilmeye başlamıştır.

Andrea diSessa (2000), BİD becerisini, potansiyel olarak tüm STEM disiplinlerini özünde etkileyecek, öğrenme açısından yeni, derin ve son derece etkili bir okuryazarlık türü olarak ele alınması gerektiğini belirtmektedir (Blinkstein, 2018). diSessa'ya (2000) göre bilgi işlemsel okuryazarlık (BİO), yeni bir beceri veya problem çözme stratejileri sınıfında yer almamakla birlikte, yeni düşünme ve öğrenme yolları üreten bir dizi maddi, bilişsel ve sosyal unsur olarak gösterilmektedir. Buna göre BİO, yeni tür zihinsel işlemlere ve bilgi temsillerine olanak tanıyan, literatür yaratan, insanların kendilerini yeni yollarla ifade etmelerini mümkün kılan ve insanların bilişsel görevleri yerine getirme biçimlerini değiştiren bir okuryazarlık olarak ele alınmalıdır.

Alanyazın incelendiğinde, BİD ile BİO kavramları arasındaki sınırların net olarak çizilemediği görülmekte olup yapılan tanımsal ifadelerin bu durumun ortaya çıkmasında önemli bir etken olduğu düşünülmektedir. Örneğin, Grover (Akt., Blikstein, 2018) yeni bilgi işlemsel

düşünme tanımlarının çoğunlukla BİO ile ilişkilendirilen yaratıcılık ve iş birliğini içerecek şekilde geliştiğini iddia etmektedir. Guzdial (Akt., Blikstein, 2018) de bu görüşe benzer olarak, yeni BİD tanımlarının çok genişlediğinden duyduğu endişeyi dile getirmektedir. Bu görüşlerin aksine Resnick (Akt., Blikstein, 2018), alandaki BİD tanımlarının hala Wing'in 2006 tarihli makalesinde belirtilen orijinal tanımla çok bağlantılı olduğunu ileri sürmektedir. Yaşanan tanımsal karmaşıklıktan hareketle bu çalışmanın amacı, diSessa'nın (2000) ortaya attığı gelişmekte olan BİO teorisinin dayandığı noktaları açıklayarak literatürdeki çalışmalardan hareketle BİD ile olan ilişkisini ortaya koymak ve okuryazarlık kapsamında ele alınma durumunu tartışmaktır.

Bilgi İşlemsel Düşünme

1940'lı yıllarda John von Neumann, bilgisayarların sadece hesaplama yapmak için değil aynı zamanda bilimsel keşif amaçlı da kullanılabileceğini savunmuştur. Neumann'a göre bilgisayarlar, insanların yapamayacakları kadar karmaşık hesaplamaları kullanarak yeni keşifler yapabilirlerdi (Aspray, 1990). Bu keşifler, bilgisayar bilimleri sayesinde sistematik olarak genişlemeye başlamıştır. Denning'e (2009) göre, bilgisayar bilimlerinde oldukça uzun bir geçmişe sahip olan bilgi işlemsel düşünme, 1950 ve 1960'lı yıllar arasında algoritmik düşünme (AD) olarak ortaya çıkmıştır. Günümüzde AD, birçok soyutlama seviyesiyle düşünmeyi, matematik kullanmayı ve bir çözümün farklı boyutlardaki problemler arasında ne kadar iyi ölçeklendiğini incelemeyi içerecek şekilde genişletilmiştir. O yıllardaki tanımına dönecek olursak AD, bir sorunu çözmek için gerekli adımların sistematik bir şekilde tasarlanması ve bu adımların bir algoritma olarak ifade edilmesidir (Denning, 2009). AD'nin merkezinde bilgisayar bilimi ve üzerine yoğunlaşılacak problemi çözme hedefiyle bir algoritmanın oluşturulması, düzenlenmesi ve yorumlanması ile ilgili bir dizi yetkinlik içeren bilgisayar programlama üzerine odaklanılmaktadır (Futschek, 2006).

Fizik alanında Nobel ödülü bulunan Ken Wilson, fiziksel problemlerin çözümü için sayısal simülasyonlar oluşturarak fiziksel sistemlerin davranışını anlamaya çalışan Monte Carlo yöntemini ilk uygulayan kişilerden biri olmuştur. Wilson, 1975 yılında teorik fizikte simülasyonların gerçekleştirilmesi ve sayısal hesaplamaların yapılması için bilgisayarların kullanımını benimseyerek, malzemelerdeki faz geçişleri hakkında yeni bir anlayış oluşturmak için bilgi işlemsel modeller geliştirmiştir.

1980 yılında Seymour Papert ve ekibi, çocuklara bilgi işlemsel kavramları öğretmek adına "LOGO" adı verilen basit metin tabanlı bir programlama dilini ve bu dili kullanarak "Sibernetik Kaplumbağa" aracını geliştirmişlerdir. Sibernetik kaplumbağa, bir robotik kaplumbağa figürünün kâğıt üzerinde çizgi izleyerek hareket etmesini sağlar. Çocuklar, kaplumbağaya geometrik şekiller çizdirmek için LOGO dilini kullanarak

komutlar yazabilirler. Bu sayede çocuklar, programlama becerilerini geliştirmekle birlikte matematiksel ve mantıksal düşünme becerilerini kullanarak kaplumbağanın hareketlerini kontrol edebilmektedir (Papert, 1980). Siberetik kaplumbağa aracında görüldüğü gibi o yıllardaki çalışmalar, tamamıyla algoritma ve matematik öğretimi üzerine odaklanmıştır. Papert'e (1996) göre BİD, öğrenme sürecinde düşünmeye yardım eden bir temsil olarak öğrenenler tarafından bilgisayarın aktif bir şekilde kullanılmasıdır.

2006 yılında BİD kavramını popüler hale getiren Jeannette M. Wing'in, bu kavramı taşıdığı önem açısından okuma, yazma ve aritmetik gibi temel beceri setlerinin yanına konumlandırılmış olması dikkat çekicidir. AD'nin bir türü olarak tanımlanan BİD, yalnızca bilgisayar bilimi ile ilgilenen insanların değil aynı zamanda gündelik yaşamda yapılan okuma, yazma ve hesap yapma gibi her bireyin taşıması gereken temel bir beceri olarak görülmektedir (Wing, 2008). Wing'e (2006) göre, bilgi işlemsel biyoloji biyologların, bilgi işlemsel oyun teorisi ekonomistlerin, nano bilgi işleme kimyacıların, kuantum bilgi işleme ise fizikçilerin düşünme biçimlerini değiştirmiştir. Bu açıdan değerlendirildiğinde BİD'in odak noktası, mekanik bilgisayarlar gibi düşünmenin yanı sıra bilgi işlemsel kavramları kullanarak farklı bilim dallarına ait problemleri çözmek olarak ele alınmalıdır (Wing, 2006). Wing tarafınca gerçekleştirilen bu tanımsal ifadeler meslektaşları tarafından pek kabul görmemiş olup özellikle bilgisayar bilimi dışına çıkan ifadelere yer verilmesi bunda büyük etken olmuştur (Denning, 2017). Bu bağlamda Wing tarafınca yapılan tanımlarda birtakım güncellemelerin yaşanmış olması dikkat çekicidir. Bu bağlamda Wing BİD'i, kompleks problemlerin üstesinden gelebilme, problemlerin ve çözümlerin formüle edilmesini sağlama ve önerilen çözümlerin bir hesaplama ajanı tarafından uygulanabilir bir biçimde temsiline öz yaklaşımları içeren bir düşünme süreci olarak ifade etmiştir (Wing, 2011). En son olarak BİD ile bilgi işlemsel bir çözümü kabul etmek için bir problemi formüle etmedeki zihinsel aktiviteyi tanımladığını, çözümün ise insan ve makine kombinasyonları tarafından gerçekleştirilebileceği belirtilmektedir (Wing, 2023a). BİD hakkında yapılan tanımsal ifadelerin bu denli değişiklik göstermesinin ana nedeni birçok disiplini etkilemesinden dolayı oldukça geniş bir etki alanına sahip olmasıdır. Çeşitli bilim insanları ve uzmanlar BİD'i farklı şekillerde açıklayabilmektedir. Örneğin, bazıları zihinsel süreçlere odaklanırken, diğerleri ise algoritma veya hesaplama odaklı bir bakış açısına sahip olabilmektedir.

Birçok araştırmacı BİD'in tanımına katkıda bulunmak amacıyla araştırmalarını bu yönde yoğunlaştırırsa da alanyazında halen kesin tanım için görüş birliği bulunmamaktadır (Grover ve Pea, 2013; Román-González, 2015; Zhang ve Nouri, 2019). Nitekim Denning (2009), BİD'in 1950 ve 1960'lı yıllardan beri AD biçimi olarak ele alındığını, özellikle problemleri çözmek için sistematik bir basamak dizisi ve bu işlemleri otomatikleştirmek için bilgisayar kullanımı olarak tanımlandığını ileri sürmüştür. Benzer şekilde Brennan ve Resnick'e (2012) göre BİD,

öğrencilerin programlama yapma sürecinde kademeli bir şekilde sürekli kendini yinelemesi, hatalarla karşılaştığında onları ayıklaması ve tekrar iyileştirmeler yapması, çalışma arkadaşlarıyla iş birliği yapıp onlarla iletişim ve etkileşimde bulunmasıdır. Araştırmacılarca yapılan bu tanımlarda BİD'in daha çok bilgisayar ortamında gerçekleştirilen eylemler dizisi olarak ele alındığını söylemek yanlış olmayacaktır.

Araştırmacılarca gerçekleştirilmesinin yanı sıra birtakım uluslararası kuruluşlarca da BİD kapsamında tanımsal ifadelere yer verilmiş olması dikkat çekicidir. Bunlar arasında yer alan Uluslararası Eğitimde Teknoloji Topluluğu ve Bilgisayar Bilimi Öğretmenler Topluluğu, BİD'in problem çözme süreci kapsamındaki aşağıda maddeler halinde sıralanan işlevsellik özelliğine odaklanılmıştır (ISTE ve CSTA, 2011):

- Problemleri, çözmeye yardımcı olmak için bilgisayar ve diğer araçları kullanmamızı sağlayacak şekilde formüleştirmek,
- Verileri analiz ederek mantıksal olarak organize etmek,
- Verileri temsil etmek için modeller ve simülasyonlar (soyutlamalar) kullanmak,
- AD'yi kullanarak çözümleri otomatik hale getirmek,
- İzlenecek adımların ve kullanılacak kaynakların en üretken ve en etkili kombinasyonunu elde etmek amacıyla olası çözümleri belirlemek, analiz etmek ve işe koşmak,
- Problem çözme prosedürünü çeşitli problemlere aktararak genelleştirmek.

Alanyazında yapılan diğer tanımlara bakıldığında, BİD üzerine yapılan geçerli tanımların düşünme kavramını yeniden tanımlama ya da bu düşünme türüne yeni bir bakış açısı kazandırmaktan ziyade, problem çözüm yöntemlerine odaklanma üzerinde farklılaştıkları ifade edilebilir. Riley ve Hunt'a (2014) göre BİD, bilgisayar bilimi uzmanı gibi düşünebilme ve onların gibi sorgulama yapabilmektir. Yadav vd. (2014) BİD'i, yalnızca bilgisayar bilimleri için değil aynı zamanda tüm bilimler için elzem olan temel bir beceri olarak görmektedir. Selby ve Woollard (2013) BİD'i soyutlama, ayırıştırma, algoritmik tasarım, değerlendirme ve genelleme boyutlarından yararlanan düşünce süreçlerini içeren, problem çözmeye odaklanmış bir yaklaşım olarak tanımlamışlardır. Tüm bu tanımları göz önüne alarak BİD'i; bizi şekillendiren dünyadaki bilginin işlenmesinin alt boyutlarını tanımayı, doğal ve yapay sistemleri, bunlara ait süreçleri anlamayı ve bilgisayar biliminin sunduğu olanakları uygulama sürecinde kullanarak mantık yürütmeyi içeren üst düzey bir düşünme becerisi olarak tanımlamak mümkündür.

Bilgi İşlemsel Düşünmenin Alt Boyutları

Alanyazında BİD ile ilgili farklı tanımların bulunması hiç kuşkusuz BİD'in alt boyutlarının da farklılaşmasına yol açmaktadır. Bu bağlamda, farklı araştırmacı ve topluluk dahilinde BİD'e ait alt boyutlardaki farklılaşma durumu Çizelge 1'de gösterilmektedir.

Temelde soyutlama, bir problemi anlamayı kolaylaştırmak ve çözmek için önemsiz ayrıntıları filtrelemek ya da yok saymaktır. Soyutlamayı karmaşık bir

sistemin özünü damıtma olarak gören Shute vd. (2017), soyutlamanın üç alt kategoriye ayrılacağını ileri sürmektedir: Çizelge 1’de yer verilen BİD’e ait alt boyutlar ele alındığında ayrıştırma, soyutlama, algoritma geliştirme, hataları ayıklama, iteratif tasarım ve genelleme olmak üzere altı farklı bileşenin ortak kabul gördüğü tespit edilmiştir (Günbatar, 2019; Román-González, 2018; Shute vd., 2017). Ayrıştırma, kompleks bir problemi veya sistemi daha basit ve yönetilebilir parçalara ayırma işlemidir (Wing, 2006, 2023b). Çoğu zaman problemdeki örüntüyü yakalamanın bir yolu da problemi ayrıştırmaktan geçmektedir. Bu nedenle büyük ölçekli problemler, kompleks sistemler veya karmaşık görevlerle başa çıkmada ayrıştırma işlemi oldukça önemlidir (Selby ve Woollard, 2013). Ancak alanyazında yapılan BİD tanımlarının çoğu ayrıştırma bileşenini içerdiği görülse de çok azı ayrıştırma sürecini temel bir tanımın ötesine taşımaktadır (Rich vd., 2019). Bunun nedeninin, BİD ölçümünde ayrıştırma boyutunun ölçülmesinin zorluğundan kaynaklandığı düşünülmektedir. Etkili bir BİD ölçütüne sahip olmak için ayrıştırma sürecini ölçecek deneysel bir araca ihtiyaç duyulmaktadır (Rich vd., 2019; Rowe vd., 2018). Alan yazındaki bu boşluğun farkına varan Rich vd., ayrıştırma sürecini daha iyi anlamak için, ayrıştırma sürecine ilişkin çeşitli disiplinlere ait araştırma sonuçlarını analiz ederek BİD kapsamında ayrıştırma için bir çerçeve sunmuşlardır.

BİD’in bir diğer alt boyutu olan soyutlama, problemin çözümü için gerekli olmayan bilgiyi kapsam dışında bırakarak, gerekli olan bilgiye odaklanma ve tanımlama olarak ele alınmaktadır (Kalelioğlu vd., 2016). Wing’e (2008) göre BİD’in özünü oluşturan soyutlama, problemin çözümü için gerekli olmayan detayları görmezden gelmenin bir yoludur. Soyutlama kavramının yok sayma, bilgiyi gizleme, seviyelere veya katmanlara ayırma gibi kavramlarla ilişkili olduğu söylenebilir.

1. *Veri biriktirme ve analizi*: Birçok kaynaktan en uygun ve önemli bilgilerin toplanması ve çok katmanlı veri kümeleri arasındaki korelasyonların tespit edilmesi;
2. *Örüntü tanıma*: Veri DNA’sının altında yatan gizli kuralların tespit edilmesi;
3. *Model kurma*: Bir sistemin çalışma prensibini anlamak veya gelecekte nasıl olacağını tahmin etmek amacıyla modeller kurmak ya da simülasyonlar oluşturmak.

Algoritma tasarlama ya da algoritmik düşünme, BİD’in önemli bir ortak bileşeni olarak görülmektedir (Kalelioğlu vd., 2016; Selby & Woollard, 2013). Geniş kabul görmesi ve uygun tanımı nedeniyle algoritma tasarlama, BİD’in tanımına dahil edilebilmesinde önemli etken olarak görülmektedir (Selby ve Woollard, 2013). Algoritma kavramı, bilgisayar bilimleri haricinde diğer birçok disiplinde belirli görevleri gerçekleştirmek için bir dizi prosedürü takip etmek olarak düşünülebilir. Futschek (2006) AD’yi, bir şekilde algoritmaların tasarlanması ve anlamlandırılmasıyla ilişkili bir yetenekler havuzu olarak kabul etmektedir. Futschek (2006) bu yetenekleri; verilen problemleri analiz etme, bir problemin tüm parçalarını tam olarak tespit etme, probleme uygun temel müdahaleleri belirleme, bu müdahaleleri kullanarak verilen bir problem için doğru bir algoritma tasarlama, bir problemin muhtemel tüm özel ve normal durumlarını kavrama ve etkinliğini artırma olarak sıralamaktadır. Wing (2006), BİD’i tanımladığı makalesinde AD kavramı yerine sezgisel düşünme kavramını tercih etmiş olup, bir çözümü keşfetmek için sezgisel akıl yürütmeyi (planlama, çizelgeleme vb.) kullanmak olarak da tanımlamıştır. AD’ye paralel düşünme kavramının dahil edilmesiyle BİD tanımının genişletildiği görülmektedir (Wing, 2011). Shute vd.’ne (2017) göre algoritmalar, bir probleme çözüm geliştirmek için tasarlanan mantıksal ve sıralı talimatlardır. Shute vd. BİD’in tanımında bulunan algoritma bileşenini dört kategoride değerlendirmekte olup bunlar aşağıda sıralanmıştır:

Çizelge 1. Bilgi İşlemsel Düşünmenin Alt Boyutlarındaki Farklılaşma Durumu

| Alt Boyutlar | Shute vd. (2017) | Grover ve Pea (2013) | Selby ve Woollard (2013) | ISTE ve CSTA (2011) | Wing (2011) | Kalelioğlu vd. (2016) |
|----------------------|------------------|----------------------|--------------------------|---------------------|-------------|-----------------------|
| Ayrıştırma | a✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Soyutlama | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Algoritma Tasarlama | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Hata Ayıklama | ✓ | ✓ | | | | |
| İteratif Tasarım | ✓ | ✓ | | ✓ | | |
| Genelleştirme | ✓ | ✓ | ✓ | ✓ | | ✓ |
| Değerlendirme | | | ✓ | | | |
| Verileri Analiz Etme | | | | ✓ | | ✓ |
| Otomasyon | | | | ✓ | ✓ | ✓ |
| Simülasyon | | | | ✓ | | ✓ |
| Paralleleştirme | | | | ✓ | ✓ | ✓ |
| Örüntü Tanıma | | | | | ✓ | ✓ |

- (a) Algoritma tasarlama: Bir problemi çözmek için bir dizi sıralı basamak oluşturma;
- (b) Paralellik: İşe koşulacak adımların eşzamanlı olarak gerçekleştirilmesi;
- (c) Verimlilik: Bir problemi çözüme sürecinde gereksiz adımları kaldırarak en ergonomik şekilde adım tasarlama;
- (d) Otomasyon: Birbirine çok benzeyen problemleri çözmek için gerektiğinde prosedürün yürütülmesini otomatikleştirmek.

Bir algoritmanın düzgün çalışıp çalışmadığının kontrol edilmesi, problemin doğru çözümü için bir gereklilik olup bu durum BİD'e ait hata ayıklama şeklinde alt boyutunu oluşturmaktadır. Hata ayıklama, bir program ya da algoritmanın istenilen sonuçlara ulaşamadığında kullanılan, tekrarlamalı (iteratif) bilişsel bir süreçtir (Wong ve Jiang, 2018). Bu süreçte öğrencilerin, karmaşık problemlerle baş ederken hata ayıklama etkinliklerinde başarısızlıkla da karşılaşabileceklerinin farkında olmaları beklenmektedir (ISTE ve CSTA, 2011). Shute vd.'ne (2017) göre hata ayıklama, bir problemin çözümünün istenilen düzeye erişemediğinde, hataların tespit edilmesi, tanımlanması ve ardından çözülme sürecidir. Hata ayıklama boyutuna, bir problemin çözümünün değerlendirilmesinde ya da iyileştirilmesi basamağında odaklanılmaktadır (Kalelioğlu vd., 2016). Hata ayıklama boyutuna süreç açısından benzerlik gösteren ve BİD'in bir diğer alt boyutu olarak kabul edilen bileşen iterasyondur. İterasyon, en verimli ve en etkili adımların kombinasyonuna ulaşmak için muhtemel çözümlerin belirlenmesi, analiz edilmesi ve uygulanması olarak görülmektedir (Günbatar, 2019). Shute vd. (2017) iterasyon boyutunu, ideal sonuca ulaşılan kadar çözümü iyileştirmek için gerekli tasarım süreçlerini tekrarlama olarak tanımlamaktadır.

Her ne kadar alanyazında az kullanılsa da problem çözüm sürecinde en çok kullanılan BİD bileşenlerinden biri de genelleştirmedir. Genelleştirme, küçük parçaların nasıl yeniden kullanılabilirliğini veya benzer problemlere nasıl yeniden uygulanabileceğini fark etme adımdır (Selby & Woollard, 2013). Ayrıca, problemleri etkili ve verimli bir şekilde çözmek için BİD becerilerini çok çeşitli durumlara aktarılabilir olarak da ifade edilebilir (Shute vd., 2017). Daha kapsayıcı bir tanım yapılırsa genelleştirme, problem çözme sürecinin genelleştirilmesi ve çok çeşitlilik gösteren problem durumlarına aktarılmasıdır (Csizmadia vd., 2015; Günbatar, 2019; Israel vd., 2015; ISTE&CSTA, 2011; Román-González vd., 2017). Gereksiz detayları görmezden gelerek problemin karmaşıklığını azaltmada soyutlamadan faydalanılırken; genelleştirme, benzer işlevleri yerine getiren birden fazla varlığı tek bir yapıyla değiştirerek karmaşıklığı azaltmaktadır (Thalheim, 2000).

Bilgi İşlemsel Okuryazarlık (BİO)

Okuryazarlık; filoloji, psikoloji, sosyoloji, tarih, antropoloji, eğitim, iletişim ve teknoloji gibi birçok disiplin tarafından incelenmektedir. Okuryazarlık kavramı, ilk defa 1958 yılında UNESCO tarafından eğitim ile ilgili bir metinde

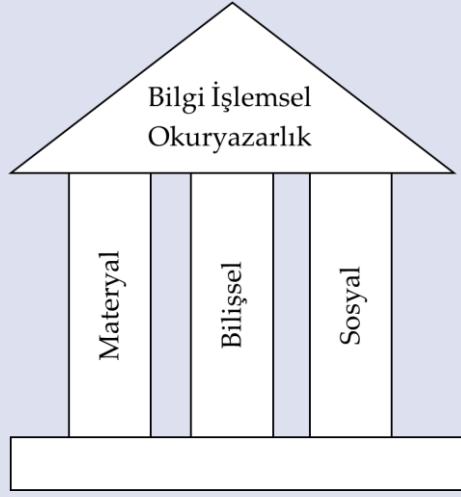
"okuryazarlık" şeklinde kullanılmaya başlanmıştır (Küslü, 2022). Metindeki okuryazarlık kavramı, "bireyin günlük hayatıyla ilgili kısa, basit bir ifadeyi anlayarak okuma ve yazma becerisi" olarak tanımlanmıştır (UNESCO, 2004, s.420). Ancak, toplumdaki sosyal, kültürel ve biçimsel değişiklikler farklı okuryazarlık türlerinin meydana gelmesinde etken olmuştur. Dijital çağın gerektirdiği yetkinliklere dayalı olarak ortaya çıkan teknoloji okuryazarlığı, bilgisayar okuryazarlığı, bilgi ve iletişim teknolojileri okuryazarlığı, ağ okuryazarlığı, bilimsel okuryazarlık, internet okuryazarlığı, dijital/sayısal okuryazarlık, enformasyon teknolojisi okuryazarlığı örnek olarak gösterilebilir.

diSessa (2018), okuryazarlığın iki temel prensibe sahip olduğunu ileri sürmekte olup bunlar şu şekildedir:

- (1) Okuryazarlık, bir kültür ya da medeniyetin, onlarca yıl ya da yüzyıllar boyunca yarışan birçok gücün sonunda geniş çaplı öğrenme, kullanım ve gelecekteki değerleri için belirli bir temsil biçiminde karar kıldığı büyük bir sosyal/entelektüel başarısıdır.
- (2) Bir okuryazarlığın özünde, ayırt edici ve eleştirel güçlü yanlarının yanı sıra sınırlılıklar ve kör noktalar da gösteren bir temsil sisteminin kitlesel olarak benimsenmesi ve dolayısıyla diğer temsil biçimleriyle olası bir tamamlayıcılık vardır.

Bu özelliklerden, bir okuryazarlığın gelişiminin oldukça karmaşık ve uzun yıllar süren sosyal ve kültürel değişimler gerektirdiği, ayrıca okuryazarlığın bir alanın düşünsel yapısını temelden değiştiren çok yönlü bir literatüre ihtiyaç duyduğu çıkarımları yapılabilir.

BİD'in bir okuryazarlık türü olarak ele alınması durumuna ilişkin çeşitli görüşler bulunmaktadır. Alanyazında BİO ile ilgili farklı teorik çerçeveler bulunsada (Guzdial, 2019; Kafai, Proctor ve Lui, 2019; Vakıl, 2018) BİD'in bir okuryazarlık olarak öne sürüldüğü en köklü düşünce, Andre diSessa'nın "Changing minds: Computers, learning and literacy" adlı kitabında yer almaktadır. diSessa (2000) kitabında aritmetik, okuma ve yazma kadar önemli olarak gördüğü bilgi işlemenin (hesaplamanın) hızlıca bir okuryazarlığa dönüştüğünden bahsetmekte ve bu okuryazarlık türünün BİD kavramından farklılaştığı yönleri detaylı bir şekilde açıklamaktadır. Ayrıca diSessa (2000), hesaplamayı kullanma şeklimizin yazı dili ve matematiği kullanmaya benzediğini, dolayısıyla bu alanda her bir beceri setinin, kullanımı için belirli kuralları (sözdizimi, cebir, dilbilgisi) ve özel düşünsel amaçları (programlama, iletişim, hesaplama) olan belirli bir temsil sistemine (kod, yazı veya sayılar) dayandığını ileri sürmektedir. diSessa (2018), fen ve matematik gibi disiplinlerin öğretimi bağlamında "programlamayı kullanma" kavramına odaklanmaktadır. diSessa (2000), BİO'nun sadece yeni bir iş becerisi veya bilgisayar bilimlerinden esinlenen problem çözme stratejisi şablonu olmadığını, aynı zamanda yeni bir bilme, düşünme, öğrenme ve bilgiyi temsil etme biçimi oluşturan bir dizi maddesel, bilişsel ve sosyal unsur olduğunu belirtmektedir. Resim 1'de gösterildiği gibi BİO; maddesel, bilişsel ve sosyal olmak üzere üç boyuta sahiptir.



Resim 1. Bilgi İşlemsel Okuryazarlığın Üç Boyutu (diSessa, 2000)

BİO'nun maddesel boyutunu, tüm programlamanın altında yatan temel temsil sistemi olan bilgisayar koduna aşinalık ve akıcılık oluşturur. Bu aşamada bir kişinin temel ya da matematiksel okuryazar olması için harflere, sayılara veya matematiksel sembollere aşına olması gerektiği gibi, BİO için de gerekli bir etmen olarak görülmelidir. Akıcılık, akış kuramına (Nakamura ve Csikszentmihalyi, 2002) dayalı olarak yeterlilik, yetkinlik ve okuryazarlıktan çok daha kapsamlı bir beceri setini içeren bir kavramdır. Maddesel akıcılığı kazanabilmek için değişken atama, fonksiyon tanımlama ve basit komut dosyalarını çalıştırma gibi işlemler de dahil olmak üzere en azından temel düzeyde programlamayı öğrenmek gerekmektedir. Ayrıca bir kodun yazım biçimine, hatalı kodları ayıklama yollarına ve kod kütüphaneleri gibi spesifik araçlara da aşına olmak gerekmektedir.

BİO'nun ikinci boyutu olan ve bilişsel ayak olarak adlandırılan bilişsel boyut, kişinin bilgi işlemenin temel araçlarını otantik problemlere ve durumlara uygulamanın zihinsel yollarını araması olarak vurgulanmaktadır. Bilgi işlemsel düşünmede tanımlanan ve problemleri algoritmik veya prosedürel yollarla çözmeyi amaçlayan bilişsel boyut kişinin bilişini geliştirmesine ve dünyayı farklı yollarla anlamasına olanak sağlamaktadır. BİO'nun bu yönü, bilgi işleme olmadan yapılamayacak görevleri yerine getirmek için programlamayı farklı sorunlara ve bağlamlara uygulamaya odaklanmaktadır (Odden, 2019). Yeterlilik düzeyinde bilişsel ayak, örneğin bir cismin çeşitli kuvvetler altındaki hareketini açıklamak amacıyla programlamayı kullanmayı içerirken, yetkinlik düzeyinde ise karmaşık problemleri çözebilmek adına problemi simüle etmek amacıyla programlamayı içermektedir.

Üçüncü boyut olarak adlandırılan sosyal boyut ise, gelişmiş okuryazarlıkların toplumdaki temelidir. Sosyal boyut, kişinin bir topluluk içinde bilgi işleme hakkında diğerleriyle fikir alışverişinde bulunmasına veya iletişim kurmasına odaklanmaktadır. Bir proje ekibi üyesinin, ekip arkadaşlarıyla fikir alışverişinde bulunarak yazdığı kodları daha okunaklı hale getirmesi BİO'nun sosyal ayağına örnek oluşturabilir. Yeterlilik düzeyinde sosyal ayak, örneğin

kişinin bir kodun anlamını diğer kişilere açıklamayı ya da rapor olarak sunmayı içerirken, yetkinlik düzeyinde ise, yazdığı kodları veya genel anlamda bilgi işleme çalışmalarını bir topluluğa sunmasını içermektedir.

Bilgi İşlemsel Okuryazarlığın Dayandığı Temel İlkeler

diSessa'ya (2018) göre BİO, dört temel ilkeye sahiptir. Bu ilkeler iyileştirme (remediation), yeniden formüle etme (reformulation), yeniden düzenleme (reorganization) ve yeniden canlandırma (revitalization) olarak sıralanabilir. BİO, kavramları yeniden formüle ederek (reformulation), matematiği yeni perspektiften gördüğümüz, bilişsel kaymaya yol açan bilişsel basitlikler sunmaktadır. Yeni bir bakış açısıyla görmek ve yeni araçlar kullanmak bir disiplinin yeniden formüle edilmesine neden olur, bu da bilişsel bir değişim yaratmakla birlikte bilişsel basitlikleri ortaya çıkarır. diSessa (2018), BİO'nun, kavramları ve ilişkileri dinamik olarak temsil etme becerisi aracılığıyla bir alana ilişkin fikirleri iyileştirdiğini (remediation) belirtmektedir. Herhangi bir alana ait kavramlar, problemler ve süreçler dinamik temsil yoluyla yeniden düzenlenebilir. BİO, bir çalışma alanının ya da bir disiplinin neyi yapabileceğini ve bunu kimlerin yapabileceğine dair entelektüel paradigmayı yeniden organize eder. diSessa'nın (2018) yeniden düzenleme (reorganization) olarak açıkladığı prensip, ilgili disiplinin kavramsal çerçevesinin yeniden düzenlenmesini sağlamaktadır. Bir çalışma alanı BİO'yu yaygın bir şekilde kullandığında, o disiplinin yeniden canlandırılmasına (revitalization), öğretmenlerin neyi nasıl öğrettiğine ve öğrencilerin nasıl öğrendiğine yol açar (diSessa, 2018).

Bilgi İşlemsel Düşünme ile Bilgi İşlemsel Okuryazarlık Arasındaki İlişki

Jacob ve Warschauer (2018), bilgi işlemsel düşünme ve BİO arasındaki ilişkiyi araştırmak için Resim 2'de gösterildiği gibi üç boyutlu bir çerçeve önermektedir:

- (1) Okuryazarlık Olarak Bilgi İşlemsel Düşünme
- (2) Okuryazarlık Aracılığıyla Bilgi İşlemsel Düşünme
- (3) Bilgi İşlemsel Düşünme Aracılığıyla Okuryazarlık



Resim 2. Bilgi İşlemsel Okuryazarlık ile Bilgi İşlemsel Düşünme Arasındaki İlişki

1-Okuryazarlık Olarak Bilgi İşlemsel Düşünme

Alanyazındaki bazı araştırmacıların, bilgi işlemsel düşünme ile bilgi işlemsel okuryazarlık kavramlarını birbirlerinin yerine kullandıkları görülmektedir (Grover ve Pea, 2013; Jacob ve Warschauer, 2018). Yapmış olduğu tanımda diSessa BİO'nun maddi, bilişsel ve sosyal yönlerinin bulunduğunu ileri sürmekte iken, benzer şekilde Wing de BİO'ya yol açan sosyal ortamları ve problem çözmede kullanılan bilişsel yaklaşımların varlığını kabul etmektedir (Jacob ve Warschauer, 2018). Ancak belirli bir uygulamayı okuryazarlık olarak tanımlama kararı büyük ölçüde sosyal olarak inşa edilmiş bağlamlara dayanmaktadır (diSessa, 2000). Dolayısıyla BİD'i bir okuryazarlık olarak tanımlama noktasında büyük özen gösterilmelidir. Ayrıca diSessa'ya (2000) göre bir okuryazarlık, maddi bir zekânın toplumda örüntülü ve yaygın bir şekilde kullanılmasını gerektirmektedir. Jacob ve Warschauer'e (2018) göre, BİD'i bir okuryazarlık olarak tanımlamak, BİO, yeni medya okuryazarlıkları ve bilgisayar okuryazarlığı literatürlerini bütünleştirme anlamı taşımaktadır.

2-Okuryazarlık Aracılığıyla Bilgi İşlemsel Düşünme

BİD'i bir okuryazarlık eğitimi yoluyla kazandırmak çeşitli faydalar sunabilir. Örneğin öğretmenler, kendilerine verilen müfredat programının değişmesini beklemeden bilgisayar bilimlerini İngilizce öğretim uygulamalarına entegre edebilirler (Jacob vd., 2018). Öğrenciler, problem çözme becerilerini çeşitli teknolojik araçlar kullanarak geliştirebilirler. Bir öğrencinin, problem durumlarını gerçek koda dönüştürülecek algoritmayı yazmadan önce sözde kod (pseudocode) kullanması, bilgi işlemsel yetkinliği geliştirmeye yönelik pedagojik bir teknik olarak görülebilir (Jacob ve Warschauer, 2018). Bu sayede öğrencilerde AD, dolayısıyla BİD'in de gelişeceği öngörülmektedir. Jacob ve Warschauer'e (2018) göre, BİD'i öğretim programına entegre etmenin diğer bir yolu da öğrencilerin anlatılar (storytelling) oluşturabilmeleri için onların Scratch ve Alice gibi uygulamaları kullanmalarını sağlamaktır. Blok tabanlı kodlama

mantığına dayanan Scratch ve Alice, öğrencilerin programlama becerilerine olan özgüvenlerini arttırmaktadır (Cooper vd., 2000; Kalelioğlu ve Gülbahar, 2014).

3-Bilgi İşlemsel Düşünme Aracılığıyla Okuryazarlık

Grover (2015), algoritmik kontrol akışının altını çizerek; sıra, döngü yapıları ve koşullu mantık gibi kavramları, bilgi işlemsel yetkinliklerin temeli olarak gördüğünü belirtmektedir. Benzer şekilde Papert'in (1980) de belirttiği üzere, öğrenciler bu algoritmik akışı kontrol ederek hatalı kodları ayıklama veya belirli bir kod dizisinin amacına ulaşip ulaşmadığını test ederek üst anlayış geliştirirler. Öğrenciler, geleneksel dillerini de bu süreçte benzer şekilde, bir anlamı ifade etmeye çalışırken hatalarını fark ederek ders aldıkları ve değerlendirdikleri sosyal bağlamlarda geliştirirler (Jacob ve Warschauer, 2018). BİD'i okuryazarlık alanına dahil etme konusunda, eşli programlama etkinlikleri, STEM grup aktiviteleri, iş birliğine dayalı programlama etkinlikleri örnek teşkil edebilir.

Bilgi İşlemsel Okuryazarlık Alanında Yapılan Çalışmalar

Bilgi işlemsel okuryazarlık teorisinin ortaya atılmasıyla birlikte alanyazında kısıtlı olsa da bazı araştırma sonuçlarına yer verildiği görülmektedir. Kim vd. (2019), BİO'nun her seviyedeki öğrenciye nasıl öğretileceği üzerine yapmış oldukları çalışmada, BİO teorisini analiz ederek bu teorisinin günümüz çağına uygun işlemsel tanımını yapmaya çalışmışlardır. Öte yandan BİO'nun BİD ile ilişkisini inceleyerek eğitimsel faktörler ile ilgili çeşitli çıkarımlarda bulunmuşlardır. Magana vd.'nin (2016) 130 mühendislik öğrencisi üzerinde yürüttükleri durum çalışmasında, mühendislik uygulamaları kapsamında yer verilen bilgi işlemsel eğitimin, öğrencilerin öz inançlarını nasıl etkileyeceğini ve bunların öğrencilerin akademik performanslarındaki farklılıkları açıklayıp açıklayamayacağını ortaya koymayı amaçlamışlardır. Araştırmacılar, likert tipi ölçek kullanarak öğrencilerin kontrol-değer teorisine göre mühendislik BİO'suna ilişkin mevcut kontrol ve değer değerlendirmelerini, öz

inançlarını ve bunun yanı sıra bu değerlendirmelerin öğrenci akademik performansı ile nasıl ilişkili olduğunu otantik uygulamalarla destekleyerek karakterize etmeye çalışmışlardır. Ayrıca araştırmacılar, bilgi işlemsel kurs eğitimlerini mühendislik derslerine entegre ederek öğrenciler üzerindeki etkilerini gözlemlemişlerdir. Araştırmanın sonucunda, öğrencilerin ön test ve son test ölçümlerinde kontrol ve değerlendirmeleri arasında genel olarak orta düzeyde korelasyon tespit etmişlerdir.

Bir başka çalışmada Odden vd. (2019), diSessa'nın (2000) bilgi işlemsel öğrenmenin materyal, bilişsel ve sosyal yönleri ayrılabilirliğini öne süren mevcut BİO teorisini, bir proje kapsamında lisans fizik dersi bağlamına uyarlamışlardır. Bu çalışmalarında araştırmacılar, "Bilgi işlemsel makale" olarak bilinen yeni geliştirdikleri bir öğretim aracını Oslo Üniversitesi Fizik Bölümü öğrencileri üzerinde uygulamışlardır. Çalışmada, lisans öğrencilerinden geçen dönem gördükleri fizik konuları hakkında programlama yaparak simülasyonlar oluşturmaları, bu simülasyonun kodlarını, çalıştıkları fizik konusunun özeti ve ilgili bir resmi gibi bilgilerini bu makalelerde belirtmeleri istenmiştir. BİO üzerine yapılan bu durum çalışmasında, diSessa'nın (2000) BİO teorisinin üç boyutu, öğrenci uygulamaları ve bilgi ve inançları açısından incelenmiştir. Araştırmanın sonucunda geliştirilen çerçeve, öğrencilerin bilgi işlemsel zorluklarını teorik olarak teşhis etmek, BİO'nun maddi ve bilişsel yönlerine odaklanan eğitim yaklaşımlarını ayırt etmek ve bilgi işlemsel denemeler gibi açık uçlu projelerin öğrencilerin öğrenmesine sağladığı faydaları vurgulamak için kullanılabilirliğini ortaya koymuştur.

Bilgi işlemsel düşünmeden okuryazarlığa geçiş anlamında yürütülen diğer bir çalışmada Dickens ve Farris (2019), bilgi işlemsel modelleme ve programlamayı ilkökul sınıfındaki daha geniş bir STEM çalışması topluluğunun bir parçası olarak düşünerek, BİO bağlamında genişletmeyi ve derinleştirmeyi amaçlamışlardır. Yürütülen bu durum çalışmasında, ilkökul 3.sınıf öğrencilerden diSessa'nın (2000) BİO teorisi üzerine inşa edilmiş STEM etkinliklerini, ViMAP adlı görsel programlama ajanını kullanarak gerçekleştirmeleri ve yaptıkları modellerle ilgili not almaları istenmiştir. BİO'nun farklı boyutlarının sınıfta gelişiminin izlendiği çalışmada, aynı kodu farklı perspektiflerden görme, başkaları için kod tasarlama ve kod hakkında konuşma ve yorumlamanın öğretimsel müdahalelerle nasıl geliştirilebileceği üzerinde çıkarımlarda bulunulmuştur.

Diğer araştırmacılar farklı olarak Horn vd. (2020), müzik okuryazarlığı ile BİO'yu birleştirdikleri çoklu okuryazarlık üzerine bir durum çalışması yürütmüşlerdir. Proje kapsamında, 500'den fazla ortaokul ve lisenin katkılarıyla web tabanlı olacak şekilde TunePad adlı ücretsiz bir kodlama-müzik platformu geliştirilmiştir. TunePad platformu aracılığıyla müzik notaları Python kodlarıyla manipüle edilerek çeşitli müzik eserlerinin oluşturulması sağlanmaktadır. Yürüttükleri çalışmayla Horn vd., özellikle diSessa'nın (2018) BİO teorisinde altını çizdiği programlama araçlarını kullanarak matematik ve fen bilimi öğretimi yapılabilir düşüncesinden hareketle,

programlama becerisi ile müzik okuryazarlığının da kazanılabileceğini vurgulamaktadır.

BİO ile ilgili bir başka çalışmada Droumeva ve Jenson (2020), ilköğretim 6. sınıf çağındaki 65 öğrenciyle bir durum çalışması yürütmüşlerdir. Nitel ve nicel olarak desenlenen çalışmada, öğrencilerden bilgisayar oyunları oluşturmaları istenmiştir. Öğrencilerin temel bilgi işlemsel düşünme yapılarını gözlemek ve değişken, işlem, fonksiyon ve koşullu ifadeler gibi temel kodlama becerilerinin kullanımını sağlamak amacıyla ücretsiz bir oyun tasarım programından (Game Maker Studio) yararlanılmıştır. Hem çalışmanın başlangıcında hem de 15 saatlik oyun geliştirme serüveninin bitiminde, öğrencilerden medya ve teknoloji kullanımı ile bilgisayarlara yönelik tutumlarını inceleyen bir medya kullanım anketi doldurmaları istenmiştir. Araştırmacılar, bildirilen medya/oyun kullanımı ve tutumlarında kızlar ve erkekler arasında istatistiksel olarak anlamlı farklılıklar olup olmadığını ve bunların BİO eğitimine hazırlığın yordayıcıları olup olmadığını incelemek için istatistiksel analizler kullanmışlardır. Çalışma sonucunda araştırmacılar, dijital yerli olarak adlandırılan günümüz gençlerinin yaygın gündelik medya kullanımını BİO için fiili bir iskele olarak kullanamayacağını ortaya koymuştur. Oyun deneyiminin, bilgisayar bilimi yeteneği için bir belirleyici olmasa da teknoloji ve medya kullanımında, oyun yapımı veya diğer BİO müfredatı tasarlanırken daha iyi anlaşılması ve hesaba katılması gereken cinsiyete dayalı farklılıkların bulunduğunu ileri sürmektedirler.

Gadanidis vd. (2022), Sümer Üçlülere (Pisagor teoremi) adlı matematik problemini diSessa'nın (2018) temel BİO ilkelerine (remediation, reformulation, reorganization, revitalization) atıfta bulunarak analiz etmişlerdir. Ayrıca öğretmenlerin etkinliğe tepkilerini ölçmek amacıyla yürüttükleri durum çalışmasına, 9 lisans öğrencisi katılmıştır. Oluşturulan 2 saatlik online toplantı kapsamında öğrencilere yöneltilen sorulara verilen yanıtlar kayıt altına alınmıştır. Toplanan yanıtlar içerik analizine tabii tutulmuştur. Çalışma, diSessa'nın (2018) BİO merceğini kullanarak bilgisayar programlamanın matematik eğitimi üzerindeki rolü ve etkisinin derinlemesine araştırılmasına olanak sağlamıştır. Yürütülen çalışma sonucunda, matematiğin yeniden formüle edilmesi, düzenlenmesi ve canlandırılmasının gerçekleşmediği sınıflarda bilgisayar programlamanın yaygın olarak kullanılabilirliği ileri sürülmüştür.

Tartışma ve Sonuç

Bundan yaklaşık 40 yıl önce Papert (1980), bilgi işlemsel düşünmenin öğrenme ve öğretmenin doğasını yeniden şekillendireceğini, öğrenenlere yeni fırsatlar sunacağını ve öğrenenlerin daha önce mümkün olmayan şeyleri başarmalarına olanak tanıyacağını ifade etmişti. Bu ifadeden 20 yıl sonra Andrea diSessa, BİO teorisini ortaya attığı görülmektedir. diSessa (2000), okuryazarlık kavramı ile dünyayı yarattığımız, anladığımız ve onunla iletişim kurduğumuz bir merceğe olduğunu savunmuştur. Günümüzde yavaş da olsa bu yeni okuryazarlık teorisi

hakkında farklı mercekler altında araştırmalar yapılmaya devam etmektedir.

BİD üzerine yapılan çoğu araştırmada, kod deneyimi genellikle öğrenciler tarafından bilgisayar ortamında oluşturulan kodlamalarla sınırlı kalmakta olup, öğretmenlere ve öğrencilere eğitim araştırmacıları tarafından tasarlanan yolların ötesinde kodun deneyimlenmesi noktasında çok az fırsat sunulmaktadır. Bu durumun sınırlılığını fark eden çoğu araştırmacı, bilimin amaçları doğrultusunda BİO becerilerinin geliştirilmesinin en iyi, öğrencilerin çeşitli amaçlar için birçok farklı faaliyete katıldığı uzun vadeli, çok disiplinli bir yaklaşımla gerçekleştirilebileceğini savunmaktadır. Kafai ve Burke'nin (2013:603) tanımladığı gibi, "kod yazmayı öğrenmek, ağırlıklı olarak bireysel ve araç odaklı bir süreç olmaktan çıkıp, dijital medyanın yaratılması ve paylaşılmasına sosyolojik ve kültürel olarak dayanan bir sürece dönüşmüştür".

Bilgisayar bilimleri, toplumsal yaşamın neredeyse her noktasına diğer bilimlere kıyasla daha geniş bir yelpazeden nüfuz etmektedir. Bu bağlamda, bilgi işlemsel düşünmeyi ete kemiğe bürüyen bilgi işlemsel okuryazarlığın eğitime entegrasyonu, bilgi işlemsel becerilerle donatılmış öğrencilere programlama becerileri kazandırarak, onların daha fazla sosyal kaynağa ulaşmalarını sağlayacaktır. Buna paralel olarak, bu süreç aynı zamanda onların kişisel kimlik değişimlerini destekleyerek iletişim, sanatsal ifade, eğlence ve iş birlikli problem çözme gibi çeşitli sosyal davranışlar kazandırmalarına da imkân sağlayacaktır. Çünkü okuryazar olmanın aynı zamanda çeşitli uygulama topluluklarına katılımın dönüşümünü de içerdiğini unutmamak gerekmektedir.

diSessa (2018), bilim ve matematikteki kavramların bilgi işlemsel temsiller kullanılarak daha basit hale getirilebileceğini savunmaktadır. Örneğin, hız ve ivmenin algoritmik olarak anlaşılması daha basittir ancak geleneksel cebirsel gösterimler kullanılarak öğrenilmesi gereksiz derecede karmaşıktır. BİO argümanı, programlama dillerinin öğretilmesi ihtiyacının ötesine uzanmaktadır. Bunlarla birlikte diSessa (2000), temel metin okuryazarlığının yüzyıllar önce pek çok disiplinin öğretimini değiştirdiği gibi, bilgi işlemsel araçlar kullanılarak öğretilmesi halinde çeşitli disiplinlerin temelden dönüştürülebileceği iddiasını ortaya koymaktadır. Bu noktadan hareketle, bu okuryazarlık türünün pilot çalışmalarla birlikte eğitim sistemlerine entegrasyonunun sağlanarak geliştirilmesi önerilmektedir.

Okuryazarlıklar, kültürel bağlamlar içinde nesnelere (materyallerle) etkileşime girerek çoklu bakış açılarını birbiriyle temas geçirir ve böylece gerçekliği algılama ve gerçeklikle etkileşime girme yollarımızı daraltmak yerine genişletir. Tarihin pek çok noktasında okuryazarlık ve onu destekleyen eğitim sistemleri sosyal, sınıfsal ve ekonomik güç yapılarını ve bu yapıları pekiştiren eşitsizlik sistemlerini sürdürmek için bir araç olarak kullanılmıştır (Vee, 2017). Neyin okuryazarlık olup neyin olmayacağına karar veren güç odaklarının, toplumların kültürel bilgi sistemlerini baskıladıkları örnekler de yok değildir.

Dolayısıyla, bir şeyin okuryazarlık olup olmaması aynı zamanda bazı toplulukların kuramsal çerçeveyi belirlemelerine ve ana hatları, yetkinlikleri ve yeterlilikleri açıklamalarına bağlıdır.

Okuryazarlık uygulamaları üzerine yapılan araştırmaların önemli bir kısmı sosyo kültürel perspektiflerden yararlanmaktadır (Barton ve Hamilton, 1998; Gee, 2000). Dolayısıyla, bilgi işlemsel düşünmenin okuryazarlık boyutunun bir parçası olmasının; BİO ile ilgili literatürün çeşitli çalışmalarla zenginleştirilmesi ve sosyokültürel bağlamda geniş bir çerçevede toplumsal kabul görmesine ihtiyaç duymaktadır. Çünkü okuryazarlık türlerinin ortaya çıkış biçimleri göz önünde bulundurulduğunda ele alındıkları kültürel ve tarihsel bağlamlara güçlü bir şekilde bağlı kaldıkları gerçeği kabul edilmelidir (Vee, 2017).

Extended Abstract

Introduction

Computational thinking can be seen as the ability to use computational concepts to develop strategies for identifying, comprehending and solving problems, regardless of domain (Martins et al., 2023). It is known that there are theories that treat computational thinking skills as literacy. (diSessa, 2000; Guzdial, 2019; Kafai, Proctor & Lui, 2019; Vakil, 2018). However, the most radical idea of computational thinking as a literacy is found in Andre diSessa's book "Changing minds: Computers, learning and literacy" by Andre diSessa (2000). In his book, diSessa (2000) mentions that information processing (computing), which he considers as important as arithmetic, reading and writing, is rapidly turning into a literacy and explains in detail the ways in which this type of literacy differs from the concept of computational thinking.

diSessa (2000) argues that the concept of computational literacy is not just a new job skill or a template for computer science-inspired problem-solving strategies, but a set of material, cognitive and social elements that constitute a new way of knowing, thinking, learning and representing knowledge. Furthermore, according to diSessa (2018), computational literacy has four fundamental principles. These principles can be listed as remediation, reformulation, reorganisation and revitalisation.

Jacob and Warschauer (2018) propose a three-dimensional framework for investigating the relationship between computational thinking and computational literacy: Computational thinking as literacy, computational thinking through literacy, and literacy through computational thinking. Jacob and Warschauer (2018) suggest that this framework can be used to teach computational thinking skills to students.

With the introduction of the theory of computational literacy, it is evident that some research findings have been included in the literature, albeit limited. Kim et al. (2019) conducted a study on how to teach computational literacy to students at all levels. Odden et al. (2019)

adapted diSessa's (2000) existing theory of computational literacy, which suggests that computational learning can be divided into material, cognitive and social aspects, to the context of an undergraduate physics course within a project. In another study, Dickens and Farris (2019) conducted a case study to extend and deepen computational literacy by considering computational modelling and programming as part of a broader community of STEM learning in primary school classrooms. In a different vein, Horn et al. (2020) conducted a case study on multiliteracies, combining musical literacy with computational literacy. Gadanidis et al. (2022) analysed the mathematical problem Sumerian Triads (Pythagorean Theorem) with reference to diSessa's (2018) basic principles of computational literacy (remediation, reformulation, reorganisation, revitalisation).

A review of the literature shows that the boundaries between the concepts of computational thinking and computational literacy are not clear-cut and that the definitional claims made are an important factor in creating this situation. Based on this definitional complexity, this study aims to explain the points on which the evolving theory of computational literacy proposed by diSessa (2000) is based, to show its relationship to computational thinking through studies in the literature, and to discuss its status within the framework of literacy.

Method

In this study, scientific texts were searched in the databases Scopus, WoS, ScienceDirect, Wiley Online Library, JSTOR, ERIC, EBSCO and Google Scholar using the keywords "computational thinking" and "computational literacy". The literature review included not only academic papers, but also books and websites that would serve the purpose of the research, in order to cover all aspects of the topic. Of the studies dealing with the theory of computational literacy, only those from the last 5 years were included in the scope of the review. Studies that were written in a language other than English or Turkish were not included.

Results and Discussion

diSessa (2000) argues that concepts in science and mathematics can be taught in a simpler way using computational representations. For example, learning about velocity and acceleration using algebraic representations is unnecessarily complex. Instead, it is simpler to teach these concepts algorithmically using a programming language but the argument for computational literacy goes beyond the need to teach programming languages. In addition, diSessa argues that just as basic textual literacy transformed the teaching of many disciplines centuries ago, it can fundamentally transform various disciplines if taught using computational tools. From this perspective, it is suggested that this type of literacy should be developed by integrating it into education systems through pilot studies such as STEM or STEAM courses.

A significant part of the research on literacy practices utilises socio-cultural perspectives (Barton & Hamilton, 1998; Gee, 2000). Therefore, the fact that computational thinking is a part of the literacy dimension needs to enrich the literature on computational literacy with various studies and to gain social acceptance in a wide range of sociocultural contexts.

Araştırmanın Etik Taahhüt Metni

Yapılan bu çalışmada bilimsel, etik ve alıntı kurallarına uyulduğu; toplanan veriler üzerinde herhangi bir tahrifatın yapılmadığı, karşılaşılabilecek tüm etik ihlallerde "Cumhuriyet Uluslararası Eğitim Dergisi ve Editörünün" hiçbir sorumluluğunun olmadığı, tüm sorumluluğun Sorumlu Yazara ait olduğu ve bu çalışmanın herhangi başka bir akademik yayın ortamına değerlendirme için gönderilmemiş olduğu sorumlu yazar tarafından taahhüt edilmiştir.

Çalışmanın derleme türü nitelik taşıması sebebiyle, 04.09.2023 tarihinde Bursa Uludağ Üniversitesi tarafından etik kurul izin gerektirmeyen çalışmalar arasında yer aldığı beyan edilmiştir.

Kaynaklar

- Aspray, W. (1990). John von Neumann and the origins of modern computing. Mit Press.
- Blikstein, P., & Moghadam, S. H. (2018). Pre-college computer science education: A survey of the field.
- Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. In Proceedings of the 2012 annual meeting of the American educational research association, Vancouver, Canada (Vol. 1, p. 25).
- Cooper, S., Dann, W., & Pausch, R. (2000). Alice: a 3-D tool for introductory programming concepts. Journal of computing sciences in colleges, 15(5), 107-116.
- Csizmadia, A., Curzon, P., Dorling, M., Humphreys, S., Ng, T., Selby, C., & Woollard, J. (2015). Computational thinking: A guide for teachers. Date of access: 09.04.2019. On the web: <http://community.computingschool.org.uk/resources/23-24>.
- Denning, P. J. (2009). The profession of IT Beyond computational thinking. Communications of the ACM, 52(6), 28-30.
- Denning, P. J. (2017). Remaining trouble spots with computational thinking. Communications of the ACM, 60(6), 33-39.
- Dickes, A. C., & Farris, A. V. (2019). Beyond isolated competencies: Computational literacy in an elementary science classroom. In Critical, transdisciplinary and embodied approaches in STEM education (pp. 131-149). Cham: Springer International Publishing.
- diSessa, A. (2000). Changing Minds: Computers, Learning and Literacy. Cambridge, MA: MIT Press.
- diSessa, A. (2018). Computational Literacy and "The Big Picture" Concerning Computers in Mathematics Education. Mathematical Thinking and Learning, 20(1), 3-31. doi:10.1080/10986065.2018.1403544
- Droumeva, M., & Jenson, J. (2020). Critical Pedagogy for the Media Generation: Youth Media Use and Computational Literacy through Game-Making. In Education for Democracy 2.0 (pp. 243-263). Brill.

- Futschek, G. (2006). Algorithmic thinking: the key for understanding computer science. In *Informatics Education–The Bridge between Using and Understanding Computers: International Conference in Informatics in Secondary Schools–Evolution and Perspectives, ISSEP 2006*, Vilnius, Lithuania, November 7-11, 2006. Proceedings (pp. 159-168). Springer Berlin Heidelberg.
- Gadanidis, G., Silva, R. S. R. da, Hughes, J., Namukasa, I., & Floyd, S. (2022). Computational Literacy & Mathematics Education. *Revista Internacional De Pesquisa Em Educação Matemática*, 12(4), 1-23. <https://doi.org/10.37001/ripem.v12i4.3144>
- Grover, S., & Pea, R. (2013). Computational thinking in K-12: A review of the state of the field. *Educational Researcher*, 42, 38-43. <https://doi.org/10.3102/0013189X12463051>
- Guzdial, M. (2019). Computing Education as a Foundation for 21st Century literacy. In *Proceedings of the 50th ACM Technical Symposium on Computer Science Education (SIGCSE '19)*(pp. 502-503.). New York, NY: ACM.
- Günbatar, M. S. (2019). Computational thinking within the context of professional life: Change in CT skill from the viewpoint of teachers. *Education and Information Technologies*, 24(5), 2629-2652.
- Horn, M.S., Banerjee, A., & West, M. (2020). Music and Coding as an Approach to a Broad-Based Computational Literacy. In: Giannakos, M. (eds) *Non-Formal and Informal Science Learning in the ICT Era. Lecture Notes in Educational Technology*. Springer, Singapore. https://doi.org/10.1007/978-981-15-6747-6_5
- International Society for Technology in Education (ISTE) (2023, Mayıs 10) ISTE Standards For Students(Permitted Educational Use). <http://www.iste.org/standards>
- Israel, M., Pearson, J. N., Tapia, T., Wherefel, Q. M., & Reese, G. (2015). Supporting all learners in school-wide computational thinking: A cross-case qualitative analysis. *Computers & Education*, 82, 263–279.
- ISTE & CSTA. (2011). Operational Definition of Computational Thinking for K–12 Education. https://cdn.iste.org/www-root/Computational_Thinking_Operational_Definition_ISTE.pdf?_ga=2.226223642.1465460610.1683620161-641442980.1681925191
- ISTE & CSTA. (2011). Operational Definition of Computational Thinking for K–12 Education. https://cdn.iste.org/www-root/Computational_Thinking_Operational_Definition_ISTE.pdf?_ga=2.226223642.1465460610.1683620161-641442980.1681925191
- Jacob, S., Nguyen, H., Tofel-Grehl, C., Richardson, D., & Warschauer, M. (2018). Teaching computational thinking to English learners. *NYS TESOL journal*, 5(2).
- Jacob, S. R., & Warschauer, M. (2018). Computational thinking and literacy. *Journal of Computer Science Integration*, 1(1).
- Kafai, Y. B., & Burke, Q. (2013). Computer programming goes back to school. *Phi Delta Kappan*, 95(1), 61-65.
- Kalelioglu, F., & Gülbahar, Y. (2014). The Effects of Teaching Programming via Scratch on Problem Solving Skills: A Discussion from Learners' Perspective. *Informatics in education*, 13(1), 33-50.
- Kalelioglu, F., Gülbahar, Y., & Kukul, V. (2016). A framework for computational thinking based on a systematic research review. *Baltic Journal of Modern Computing*, 4(3), 583-596
- Kim, S., Han, S., & Kim, H. (2009, August). How Can We Teach Computational Literacy to All Levels of Students?. In *2009 Fifth International Joint Conference on INC, IMS and IDC* (pp. 1395-1400). IEEE.
- Küslü, F.(2022). Öğretmenlerin eleştirel dijital okuryazarlık becerilerinin çeşitli değişkenler açısından incelenmesi [Doktora Tezi, Anadolu Üniversitesi]. YÖK Ulusal Tez Merkezi
- Magana, A. J., Falk, M. L., Vieira, C., & Reese Jr, M. J. (2016). A case study of undergraduate engineering students' computational literacy and self-beliefs about computing in the context of authentic practices. *Computers in Human Behavior*, 61, 427-442.
- Martins, E. C., da Silva, L. G. Z., & de Almeida Neris, V. P. (2023). Systematic mapping of computational thinking in preschool children. *International Journal of Child-Computer Interaction*, 100566.
- Nakamura, J., & Csikszentmihalyi, M. (2002). The concept of flow. *Handbook of positive psychology*, 89, 105.
- Papert, S. (1980). *Mindstorms: Children, Computers and Powerful Ideas*. New York: Basic Books, Inc.
- Papert, S. (1996). An exploration in the space of mathematics educations. *Int. J. Comput. Math. Learn.*, 1(1), 95-123.
- Peppler, K., Halverson, E. R., & Kafai, Y. B. (Eds.). (2016). *Makeology: Makers as Learners* (Volume 2) (Vol. 2). Routledge.
- Odden, T. O. B., Lockwood, E., & Caballero, M. D. (2019). Physics computational literacy: An exploratory case study using computational essays. *Physical Review Physics Education Research*, 15(2), 020152.
- Rich, P. J., Egan, G., & Ellsworth, J. (2019). A Framework for Decomposition in Computational Thinking. *Proceedings of the 2019 ACM Conference on Innovation and Technology in Computer Science Education - ITiCSE '19*. doi:10.1145/3304221.3319793
- Riley, D. D., & Hunt, K. A. (2014). *Computational thinking for the modern problem solver*. CRC press.
- Román-González, M. (2015). Computational thinking test: Design guidelines and content validation. *Proceedings of EDULEARN15 Conference*, 2436-2444
- Román-González, M., Pérez-González, J. C., & Jiménez-Fernández, C. (2017). Which cognitive abilities underlie computational thinking? Criterion validity of the computational thinking test. *Computers in Human Behavior*, 72, 678–691.
- Román-González, M., Pérez-González, J. C., Moreno-León, J., & Robles, G. (2018). Extending the nomological network of computational thinking with non-cognitive factors. *Computers in Human Behavior*, 80, 441–459.
- Rowe, E., Asbell-Clarke, J., Baker, R., Gasca, S., Bardar, E., & Scruggs, R. (2018). Labeling implicit computational thinking in pizza pass gameplay. In *Extended abstracts of the 2018 CHI conference on human factors in computing systems* (pp. 1-6).
- Selby, C., & Woollard, J. (2013). Computational thinking: the developing definition. https://eprints.soton.ac.uk/356481/1/Selby_Woollard_bg_soton_eprints.pdf
- Shute, V. J., Sun, C., & Asbell-Clarke, J. (2017). Demystifying computational thinking. *Educational Research Review*, 22, 142–158.
- Thalheim, B. (2000). *Fundamentals of entity-relationship modeling*. New York, NY: Springer
- Vakil, S. (2018). Equity in computer science education. *Harvard Educational Review*, 88(1), 26-53.
- Veel, A. (2017). *Coding literacy: How computer programming is changing writing*. MIT Press.
- 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 A: Mathematical, Physical and Engineering Sciences*, 366(1881), 3717–3725. doi:10.1098/rsta.2008.0118

- Wing, J. (2011). Research notebook: Computational thinking— What and why. *The link magazine*, 6, 20-23.
- Wing, J. M. (2023a, Mayıs 10). Research Notebook: Computational Thinking--What and Why? <https://www.cs.cmu.edu/link/research-notebook-computational-thinking-what-and-why>
- Wing, J.M. (2023b, Mayıs 16). Computational Thinking. http://www.cs.cmu.edu/afs/cs/usr/wing/www/Computational_Thinking.pdf
- Wong, G. K. W., & Jiang, S. (2018). Computational Thinking Education for Children: Algorithmic Thinking and Debugging. 2018 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE). doi:10.1109/tale.2018.8615232
- Yadav, A., Mayfield, C., Zhou, N., Hambrusch, S., ve Korb, J. T. (2014). Computational thinking in elementary and secondary teacher education. *ACM Transactions on Computing Education*, 14(1). <https://doi.org/10.1145/2576872>
- Zhang, L., & Nouri, J. (2019). A systematic review of learning computational thinking through Scratch in K-9. *Computers & Education*, 141, 103607.