

# WebALGO: Algoritma Öğreniminde İnternet Tabanlı Bir Eğitimsel Materyal Geliştirilmesi

Utku KÖSE

Uşak Üniversitesi, Bilgisayar Bilimleri Uygulama ve Araştırma Merkezi, Uşak, utku.kose@usak.edu.tr

## Özet

Genel olarak bu çalışmanın amacı, bilgisayar programlama öğreniminde kritik öneme sahip olan algoritma kavramının anlaşılması için kullanılabilen, İnternet tabanlı bir eğitimsel materyali tanıtmak ve değerlendirmektir. Öğrencilere çeşitli zaman aralıklarıyla İnternet üzerinden algoritma problemleri sunan bu sistemde, tasarlanan bilgisayar algoritmalarının çalışmasını test etmek amacıyla, adım adım çalıştırma yaklaşımına dayalı, Çalıştırma Tablosu adı verilen, basit ancak etkin bir tablo yapısı kullanılmaktadır. Daha önce öğrencilerin ilgisini çekmesi nedeniyle Web ortamına taşınmış olan Çalıştırma Tablosu yaklaşımı üzerine kurulu bu eğitimsel materyal, verilen bir algoritmanın çözülmesi istenen problemi gerçekten çözüp çözmediğinin test edilmesini sağlamaktadır. Algoritmaların sağlamlasına yönelik bu test süreci sayesinde bireyler, algoritmik düşünce mantığı ve algoritma tasarımına yönelik bilgi ve becerileri pekiştirebilmektedir.

**Anahtar Sözcükler:** : Algoritmalar, bilgisayar programlama, internet tabanlı eğitim, bilgisayar eğitimi, çalıştırma tablosu

## Abstract

Generally objective of this work is to introduce and evaluate an İnternet based educational material that can be used for understanding the algorithm concept, which has a critical importance on learning computer programming. In this system, which provides algorithm problems to students in several time periods over İnternet, a simple but effective table structure, which is called as Running Table and based on step-by-step running approach, is used in order to test working of designed computer algorithms. With this educational material on Running Table approach, which has taken students' attract, it is possible to test an algorithm if it can solve a specific problem really or not. Thanks to this test process, individuals can consolidate their knowledge and abilities in the sense of algorithmic thinking logic and algorithm design.

## 1. Giriş

Bilişim Çağı'nın gereği olarak, bilgisayar odaklı birçok farklı bilgi ve becerinin etkin bir şekilde elde edilmesi oldukça önemlidir. Özellikle bilgisayar odaklı çalışmalarda istendik amaçlara ulaşmak için bazı becerilerin bilinmesi daha fazla önem kazanmaktadır. Bilgisayar programlama, özellikle Bilgisayar Bilimleri'nde ön planda olan, bilgisayar donanım bileşenlerinin ve genel anlamda bilgisayar tabanlı sistemlerin isteklerimiz doğrultusunda çalışabilmesi ve problemlere çözümler üretebilmesi konusunda yönlendirilebilmesi adına sahip olunması gereken bir beceridir. Kuşkusuz ki, günümüz hızlı gelişme ve iyileşmeleri neticesinde farklı teknolojileri de etkileyen ve çok-disiplinli bir yapıya odaklanan Bilgisayar Bilimleri; bilgisayar programlamaya yönelik temel düzeyde bilgi ve becerilerin farklı araştırma / bilim / faaliyet alanlarında da gerekli bir hale gelmesine sebep olmuştur [1-3]. Bilgisayar programlamaya yönelik artan bu ilgi, bilgisayar programlamaya yönelik söz konusu bilgi ve becerilerin kazandırılması yönünde eğitimsel süreçlerin de birçok araştırmaya konu olmasını sağlamıştır.

Süregelen çalışmalar ve eğitimsel eğilimler incelendiğinde, bilgisayar programlama öğretiminde genel yaklaşımın, algoritma adı verilen kavramın öğrenenlere anlatılması şeklinde olduğunu görebiliriz.

Bu yaklaşım bağlamında problem çözümlerinin algoritmik düşünce mantığıyla ortaya konulması ve 'problem çözümlerinin sonlu ve organize adımlar bir biçimde' ifade edildiği algoritmaların oluşturulması süreçleri izlenmekte, hatta algoritmaların şekilsel gösterimi olan akış şemaları ile bilgisayar programlama mantığının temelleri atılmaktadır. Bilgisayar programlamanın temellerine yönelik dersler ve kurslar dikkate alındığında; izlenen eğitimsel yolların ifade edilen çerçevede gerçekleştirildiği kolaylıkla görülebilmektedir.

Bu noktada dikkat çeken sorun, algoritmaların nasıl çalıştığı konusunda yaşanan tereddütler neticesinde algoritmik düşünce ve bilgisayar programlama mantığının sağlam temellere oturtulamamasıdır. Söz konusu problem, çeşitli yazılımsal çözümlerle giderilmeye çalışılmakta; eğitimsel temellere dayalı farklı yaklaşım, yöntem ve tekniklerle de etkin çözümler elde edilmeye çalışılmaktadır [4-9]. Hatta son gelişmeler, bilgisayar ve bilgisayar destekli sistemler olmaksızın algoritmik düşünce mantığını aşılama çalışmaları somut araçların da kullanımını gözler önüne sermektedir [10-12]. Burada önemli olan nokta, algoritmaların çalışma şeklini anlamaya yönelik, pekiştirme niteliğindeki yolların işe koşulabilmesidir.

Yapılan açıklamalar kapsamında bu çalışmanın amacı, bilgisayar programlama öğreniminde kritik öneme sahip olan algoritma kavramının anlaşılması için kullanılabilen, İnternet tabanlı bir eğitimsel materyali tanıtmak ve değerlendirmektir. Öğrencilere çeşitli zaman aralıklarıyla İnternet üzerinden algoritma problemleri sunan bu sistemde, tasarlanan bilgisayar algoritmalarının çalışmasını test etmek amacıyla, adım adım çalışma yaklaşımına dayalı, Çalıştırma Tablosu adı verilen, basit ancak etkin bir tablo yapısı kullanılmaktadır. Daha önce öğrencilerin ilgisini çekmesi nedeniyle Web ortamına taşınmış olan Çalıştırma Tablosu yaklaşımı üzerine kurulu bu eğitimsel materyal, verilen bir algoritmanın çözülmesi istenen problemi gerçekten çözüp çözmediğinin test edilmesini sağlamaktadır. Algoritmaların sağlanmasına yönelik bu test süreci sayesinde bireyler, algoritmik düşünce mantığı ve algoritma tasarımına yönelik bilgi ve becerileri pekiştirebilmektedir.

İlgili araştırma konusu bağlamında, çalışmanın ilerleyen bölümleri şu şekilde organize edilmiştir: Bir sonraki bölüm altında, geliştirilen İnternet tabanlı materyalin temellerinde yer alan Çalıştırma Tablosu yaklaşımına dair temel bilgilerden bahsedilmektedir. Ardından; Üçüncü Bölüm altında, ilgili yaklaşımın kullanıldığı, WebALGO adlı yazılım ortamı ve genel çalışma şekli tanıtılarak, üniversite öğrencileri arasında gerçekleştirilen değerlendirme sürecine odaklanılmaktadır. Çalışma, ulaşılan sonuçların ve gelecek çalışma yönelimlerinin tartışıldığı Dördüncü Bölüm ile sona ermektedir.

## 2. Çalıştırma Tablosu Yaklaşımı

Çalıştırma Tablosu, yazılan bir algoritmanın, ifade edilen problemi gerçekten çözüp çözmediğini anlamak adına, örnek değerler bütünü ile kontrol işlemi yapılmasını sağlayan bir tablo düzeneği / yapısıdır. Çalıştırma Tablosu'nun kullanımını oldukça basit olmakla birlikte sadece belli başlı bazı kurallara dikkat etmek gerekmektedir. Bir bakıma Çalıştırma Tablosu'nun kullanım şeklini de tarif eden bu kuralları kısaca şu şekilde açıklayabiliriz:

**Kural 1:** Algoritmadaki her bir değişken için tabloda bir sütun kullanılır ve sütunun başına ilgili değişkenin adı yazılır. Bunun dışında, algoritmada ADİM sütunu ve (eğer algoritmada ekrana yazdırma yapılmışsa) EKİRAN sütunu sabittir. ADİM sütununun en solda, EKİRAN sütununun da ADİM sütununun hemen sağında yer alması yerinde olacaktır.

**Kural 2:** Algoritmada ekranı temizleme gibi bir komut kullanılmadığı sürece; ekrana yazdırılanlar, EKİRAN sütununda (her yeni ekrana yazdırma adımında) alt alta eklenerek yazılır.

**Kural 3:** Mantıken, bir değişkenin aldığı son değer, başka adımlarda değişkenin içeriğine müdahale

olmadığı sürece sabit kalır ve dolayısıyla yeni adımlarda da ilgili değişkenin hücrelerine yazılmaya devam edilir. Eğer bir adımda ilgili değişkenin değeri değişirse, yeni değer, daire içine alınarak ilgili hücreye yazılır.

**Kural 4:** Eğer şartlarında satır iki parça halinde kullanılır; üstteki parçaya şart yazılır ve şartın DOĞRU mu YANLIŞ mı olduğu yazılır; alt parçaya ise sütunların değerleri yazılmaya devam edilir.

**Kural 5:** Bir sütun içerisine her adımda değer yazılır [eski veya yeni değer(ler)]. Sadece algoritma başladıktan sonra ilgili bir sütun içerisinde henüz bir değer değişimi yoksa tire (–) karakteri kullanılabilir.

**Kural 6:** Klavyeden okuma yapılan adımlarda, EKİRAN sütunundaki kısma okunan değer yazılır sonrasına (enter) ibaresi konularak, ilgili değer klavyeden yazılır, Enter tuşuna basılarak okunması sağlandığı ifade edilebilir.

**Kural 7:** Eğer problem çözümüne yönelik genel algoritma birden fazla algoritma bloğuna dayalı bir biçimde oluşturulmuşsa, o andaki algoritma adımının esasında hangi algoritma bloğunda olduğunu göstermek için ek olarak BLOK sütunu da eklenebilmektedir. Böylece algoritmik çözüm adımlarının işlevler boyutunda irdelenmesi sağlanmaktadır.

Çalıştırma Tablosu'na yönelik şöyle bir örnek ortaya koyabiliriz:

**Örnek 1:** Klavyeden girilen başlangıç ve bitiş değerlerine bağlı olarak; ilgili aralıktaki tamsayıların çarpımını bulup, ekrana yazdıran algoritma:

A1- Başla  
A2- carpim = 1  
A3- “Başlangıç değerini giriniz:” yaz  
A4- baslangic oku  
A5- “Bitiş değerini giriniz:” yaz  
A6- bitis oku  
A7- carpim = carpim \* baslangic  
A8- Eğer (baslangic != bitis) ise baslangic = baslangic + 1 ve A7'ye git  
A9- “Sonuç:” carpim yaz  
A10- Dur

Bu algoritmanın Çalıştırma Tablosu ortamında test edilmesine yönelik örnek bir çözüm şekli Tablo 1 altında gösterilmiştir.

Tablo 1. Örnek 1'deki algoritma için bir Çalıştırma Tablosu çözümü

ADIM	EKRAN	baslangic	bitis	carpim
A1	BAŞLA			
A2	-	-	-	1
A3	Başlangıç değerini giriniz:	-	-	1
A4	Başlangıç değerini giriniz: 1 (enter)	1	-	1
A5	Başlangıç değerini giriniz: 1 Bitiş değerini giriniz:	1	-	1
A6	Başlangıç değerini giriniz: 1 Bitiş değerini giriniz: 5 (enter)	1	5	1
A7	Başlangıç değerini giriniz: 1 Bitiş değerini giriniz: 5	1	5	1
A8	Eğer (baslangic != bitis) DOĞRU			
	Başlangıç değerini giriniz: 1 Bitiş değerini giriniz: 5	2	5	1
A7	Başlangıç değerini giriniz: 1 Bitiş değerini giriniz: 5	2	5	2
A8	Eğer (baslangic != bitis) DOĞRU			
	Başlangıç değerini giriniz: 1 Bitiş değerini giriniz: 5	3	5	1
A7	Başlangıç değerini giriniz: 1 Bitiş değerini giriniz: 5	3	5	6
A8	Eğer (baslangic != bitis) DOĞRU			
	Başlangıç değerini giriniz: 1 Bitiş değerini giriniz: 5	4	5	6
A7	Başlangıç değerini giriniz: 1 Bitiş değerini giriniz: 5	4	5	24
A8	Eğer (baslangic != bitis) DOĞRU			
	Başlangıç değerini giriniz: 1 Bitiş değerini giriniz: 5	5	5	24
A7	Başlangıç değerini giriniz: 1 Bitiş değerini giriniz: 5	5	5	120
A8	Eğer (baslangic != bitis) YANLIŞ			
	Başlangıç değerini giriniz: 1 Bitiş değerini giriniz: 5	5	5	120
A9	Başlangıç değerini giriniz: 1 Bitiş değerini giriniz: 5 Sonuç: 120	5	5	120
A10	DUR			

Örnek 1'deki algoritma için gerçekleştirilen Çalıştırma Tablosu çözümünden de anlaşılacağı üzere; klavyeden girilen iki değer arasındaki (değerler dâhil) sayıların çarpımı, ilgili algoritma tarafından doğru bulunabilmektedir. Burada örnek çözüm 1 ve 5 değerleri için yapılmış olsa da; farklı değerler için çıkacak sonuçlar da incelenerek algoritmanın adım adım çalıştırılması; yani testi gerçekleştirilmiş olmaktadır. Bu noktada öğrenenler, algoritmadaki çözüme yönelik yazım hataları dışında; mantık hatası olup olmadığını da denetlemek suretiyle algoritmik düşünce mantığını sağlama yoluyla pekiştirme yoluna gidebilmektedir. Söz konusu bu yaklaşımın anlaşılması oldukça basit olmakla birlikte, uygulamadaki pratikliği de dikkat çekmektedir. Bir başka örnek olarak ise:

**Örnek 2:** Klavyeden girilen bir sayının faktöriyelini bulup (x!), ekrana yazdıran algoritma:

- A1- Başla
- A2- faktoriyel = 1
- A3- "Bir sayı giriniz:" yaz
- A4- x oku
- A5- faktoriyel = faktoriyel \* x
- A6- Eğer (x > 1) ise x = x - 1 ve A5'ye git
- A7- "Sonuç:" faktoriyel yaz
- A8- Dur

### 3. Çalıştırma Tablosu ve İnternet Tabanlı Bir Eğitimsel Materyal: WebALGO

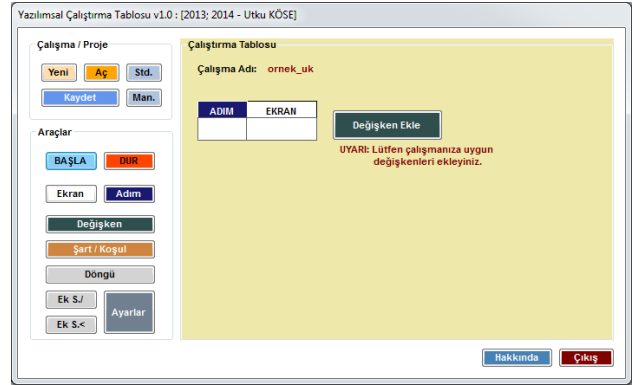
Çalıştırma Tablosu'nun eğitimsel süreçlerde daha hızlı ve etkin kullanılabilmesi adına, söz konusu yapıyı kolay bir şekilde tasarlamaya imkân sağlayan bir yazılım sistemi daha önceleri ortaya konmuştur. C# programlama dili yardımıyla bir tür Windows Form programı olarak tasarlanan ve geliştirilen bu yazılım ile birlikte, öğrenenlerin sağlamlasını yapacakları algoritmalara ait Çalıştırma Tablosu yapılarını bilgisayar ortamında oluşturup değerlendirme yapabilmeleri sağlanmıştır. Yazılımsal Çalıştırma Tablosu, tipik sağlama tablosu içerisindeki sütunlarda ve ilgili alanlarda yer alan düzenlemelerin yardımcı araçlarla kolaylıkla yapılabildiği, kullanıcının her adım gösterimini ve adımlar içerisinde gerçekleşen kontrolleri veya değişken değer değişimi gibi durumları pratik bir şekilde tanımlayıp gösterebildiği, etkileşimli bir ortam sunmaktadır. Yazılımsal Çalıştırma Tablosu'na ait genel arayüz - ekran görüntüsü Şekil 1 altında gösterilmiştir.

Sonraki süreçlerde bu yazılımın öğrenciler tarafından sevilmesi ve ilgiyle kullanılması, araştırmacının aklında İnternet / Web ortamının avantajlarından faydalanarak, daha etkin bir sürecin elde edilebileceği fikrini ortaya çıkarmıştır. Sonuç olarak da, geliştirilmiş olan sisteme Web desteği getirilmiş ve öğreticinin kendi arayüzünden sunduğu örnek Çalıştırma Tablosu problemlerinin otomatik olarak öğrencilerin havuzuna düştüğü bir eğitimsel ortam tasarlanmıştır. Geliştirilmiş olan bu yeni İnternet tabanlı eğitimsel materyale, Web Odaklı Algoritma Öğrenimi düşüncesinden türetilen, WebALGO adı verilmiştir.

### 4. Değerlendirme

WebALGO'nun algoritma ve dolayısıyla bilgisayar programlama öğretimi aşamasındaki etkinliğini değerlendirmek adına uygulamaya dayalı bir test yaklaşımı ortaya konulmuştur. Buna göre, ilgili yazılım ortamının, Uşak Üniversitesi – Bilgisayar Programcılığı Ön Lisans programına kayıtlı 92 öğrenci tarafından, Programlama Temelleri ve Programlamaya Giriş adlı dersler kapsamında (bir yarı-yıl süresince) kullanılması sağlanmış ve 14 haftalık sürece tekabül eden ders sonunda, 10 adet ifade içeren bir anket uygulanarak çeşitli bulgular elde edilmiştir.

WebALGO'nun, ilgili dersler boyunca kullanımına ve etkinliğine ilişkin düşüncelere odaklanan anket çalışması ile birlikte, Likert Ölçeği'ne göre (1: 'Tamamen Katılmıyorum'; 2: 'Katılmıyorum'; 3: 'Karasızım'; 4: 'Katılıyorum'; 5: 'Tamamen Katılıyorum') elde edilmiş olan bulgular Tablo 2'de sunulmuştur.



Şekil 1. Yazılımsal Çalıştırma Tablosu'na ait genel arayüz - ekran görüntüsü

Tablo 2. WebALGO'nun kullanım / test sürecine yönelik uygulanan anket ve elde edilen bulgular

İfade	Dönütler*				
	1	2	3	4	5
"WebALGO ile algoritma kavramını daha iyi anladım."	1	4	18	28	41
"WebALGO ile bilgisayar programlama öğretileridir."	2	5	15	33	37
"Çalıştırma Tablosu yaklaşımı öğrenme etkinliğimi olumsuz etkiledi."	34	38	13	7	0
"Programlama derslerinde buna benzer süreçlerin yer almasını isterim."	1	5	13	30	43
"Çalıştırma Tablosu ile adım adım çalıştırmayı anladım."	0	1	11	26	54
"Çalıştırma Tablosu ile algoritmalarındaki mantık hatalarımı daha kolay bulabilirim."	1	3	29	31	28
"Web üzerinden problem sunulmasını sevdim."	1	1	8	26	56
"Çalıştırma Tablosu bilgisayar programlamayı daha iyi anlamamı sağladı."	2	6	18	24	42
"WebALGO ile değişken, şart kontrolü, döngüler gibi; programlama yaklaşımlarını anladım."	0	5	12	18	48
"WebALGO'nun geliştirilmesini isterim."	0	1	22	27	42
Toplam Katılımcı: 92					
*Likert Ölçeği: 1: 'Tamamen Katılmıyorum'; 2: 'Katılmıyorum'; 3: 'Karasızım'; 4: 'Katılıyorum'; 5: 'Tamamen Katılıyorum'					

Elde edilen bulgular, değerlendirme sürecine dâhil olan öğrencilerin genel anlamda WebALGO'nun eğitimsel süreçlerde kullanımından memnun olduğunu göstermektedir. Bu bağlamda geliştirilen İnternet / Web tabanlı yazılımın ve içerisinde işe koşulan Çalıştırma Tablosu yaklaşımının etkin bir rol oynadığından bahsetmemiz mümkündür.

### 5. Sonuç ve Gelecek Çalışmalar

Bu çalışma kapsamında, bilgisayar programlama öğretiminde etkin bir araç olarak kullanılabilceği düşünülen; WebALGO yazılımı kısaca tanıtılmıştır. Temel anlamda algoritmaların adım adım çalıştırılmak suretiyle sağlamlasının yapılması mantığından yola çıkan Çalıştırma Tablosu yaklaşımını içeren bu yazılım, söz konusu öğrenim sürecinin organize bir şekilde İnternet üzerinden sunulmasına etkin katkı sağlamaktadır. Bunun yanında WebALGO, bilgisayar programlama kapsamında önemli olan değişkenler, şart kontrolleri, döngüler gibi kavramların özellik ve işlevlerinin ön plana çıkarılmasına odaklanmış bir algoritmik düşünce mantığı geliştirme serüveninin de adımlarını atmaktadır.

WebALGO'nun değerlendirme süreci ile birlikte üniversite öğrencilerinden olumlu dönütler alması, eğitimsel anlamda etkin bir yol izlendiği konusunda olumlu fikirler oluşmasını sağlamıştır. Bu noktada yazılım içerisinde sunulan Çalıştırma Tablosu yaklaşımına yönelik olumlu eğilimler de dikkat çekmiştir.

Özellikle değerlendirme süreciyle elde edilen bulgular, WebALGO'nun ilerleyen zamanlarda geliştirilmesi yönünde çeşitli fikirlerin de oluşmasını sağlamıştır. Buna göre, yazılım içerisindeki Çalıştırma Tablosu'nun daha karmaşık programlama yaklaşımlarını (Örneğin Nesne Yönelimli / Tabanlı Programlama) kapsayacak bir biçimde genişletilmesi ve böylece WebALGO'nun de kapsamının genişletilmesi düşünülmektedir. Ayrıca WebALGO'nun Android, iOS, ve Windows Phone gibi işletim sistemlerini kapsayacak şekilde, hem Web browser bazlı, hem de mobil bazlı olarak düzenlenmesi de düşünülmektedir.

### 6. Kaynaklar

- [1] Santos-Carrillo, R. "The importance of coding", Teq – It's all about learning, [Online] <http://www.teq.com/blog/2013/12/part-1-importance-coding/> (2013).  
 [2] Crow, D. "Why every child should learn to code", The Guardian, [Online]

<http://www.theguardian.com/technology/2014/feb/07/year-of-code-dan-crow-songkick/> (2014).

- [3] Tynker, "Four reasons why kids should learn programming", Tynker, [Online] <http://www.tynker.com/blog/articles/stem-education/four-reasons-why-kids-should-learn-programming/> (2013).  
 [4] Corbett, A. T. and Anderson J. R. "Student modeling in an intelligent programming tutor", E. Lemut, B. du Boulay, G. Dettori (Ed.), Cognitive Models and Intelligent Environments for Learning Programming, New York, Springer-Verlag: 135-144, (1993).  
 [5] Jiménez-Díaz, G., Gómez-Albarrán, M., Gómez-Martín, M. A. and González-Calero, P. A. "ViRPlay: Playing roles to understand dynamic behavior", Workshop on Pedagogies and Tools for the Teaching and Learning of Object Oriented Concepts, (2005).  
 [6] Baghaei, N. "A collaborative constraint-based adaptive system for learning object-oriented analysis and design using UML", V. P. Wade, H. Ashman, B. Smyth (Ed.), Adaptive Hypermedia and Adaptive Web-Based Systems, Berlin, Springer: 398-403, (2006).  
 [7] Othman, A. A., Pislaru, C. and Impes, A. M. "Improving the quality of technology-enhanced learning for computer programming courses", International Journal of Information and Education Technology, 4 (1): 83-88, (2014).  
 [8] Powell, L. M. and Wimmer, H. "Evaluating the effectiveness of self-created student screencasts as a tool to increase student learning outcomes in a hands-on computer programming course", Proceedings of the Information Systems Educators Conference, 2167: 1435, (2014).  
 [9] Malliarakis, C., Satratzemi, M. and Xinogalos, S. "Educational games for teaching computer programming", C. Karagiannidis, P. Politis, I. Karasavvidis (Ed.), Research on e-Learning and ICT in Education, New York, Springer-Verlag: 87-98, (2014).  
 [10] Henderson, P. "Computer science unplugged", Journal of Computing Sciences in Colleges, 23(3): 168-168, (2008).  
 [11] Bell, T., Alexander, J., Freeman, I. And Grimley, M. "Computer science unplugged: School students doing real computing without computers", The New Zealand Journal of Applied Computing and Information Technology, 13(1): 20-29, (2009).  
 [12] Visnjic, F. "From 0 to C' – Teaching programming using a tangible approach", Creative Applications Network, [Online] <http://www.creativeapplications.net/theory/from-0-to-c-teaching-programming-using-a-tangible-approach/> (2012).