

MATEMATİK PROBLEMLERİNİN ALGORİTMİK ÇÖZÜMÜ İÇİN GÖRSEL SÜRÜKLE- BIRAK PROGRAMLAMA DİLİ EKLENTİSİ GELİŞTİRİLMESİ

Prof. Dr. M. Melih İnal
Kocaeli Üniversitesi Enformatik Bölümü
minal@kocaeli.edu.tr

Okt. Uğur Yıldız
Kocaeli Üniversitesi Enformatik Bölümü
uguryildiz@kocaeli.edu.tr

Özet

Kodlama Eğitimi'nde yaygın olarak kullanılan görsel sürükle-bırak dilleri elektronik, robotik, graph algoritmaları, ajan-temelli simülasyon gibi farklı alanlarda uygulamalar gerçekleştirmek için de kullanılabilir. Bu bağlamda kodlama ile matematik alanlarının algoritma özelinde birbirleriyle ilişkili olduğu ve benzerlikler gösterdiği gözlemlenmektedir. Bir matematik probleminin çözümünde çeşitli teorem, çözüm teknikleri ve işlemlerin belirli bir mantıksal düzen içerisinde gerçekleştirilmesi gerekliliği matematik eğitiminde görsel sürükle bırak programlama dilleri araçlarının kullanımına imkan sağlamaktadır. Bu çalışmada Google tarafından geliştirilen görsel sürükle bırak programlama dili kütüphanesi olan Blockly yazılımı kullanılarak matematik teorem, teknik ve işlemleri için bileşenler (blocks) sunan bir uygulamanın geliştirilme süreci, deneyimi incelenmiştir. Bu tip disiplinler arası alanlarda gerçekleştirilecek çalışmalar için platform imkanları tartışılmaktadır.

Anahtar Sözcükler: Görsel Sürükle-Bırak Programlama Dili, E-öğrenme, JavaScript.

DEVELOPMENT OF DRAG-AND-DROP VISUAL PROGRAMMING ADD-IN FOR ALGORITHMIC SOLVING OF MATH PROBLEMS

Abstract

Visual drag-and-drop languages commonly used in Coding Education can also be used to perform applications in different areas such as electronics, robotics, graph algorithms, agent-based simulation. In this context, it is observed that the coding and mathematics fields which show similarities are related to each other in terms of algorithms. The necessity of implementing various theorems, solution techniques and operations in a logical order for the solution of a mathematical problem allows the use of visual drag and drop programming language tools in mathematics education. In this study, the development process and experience of an application that provides components (blocks) for mathematical theorems, techniques, and operations using Blockly software, a visual drag-drop programming language library developed by Google, is examined. Platform possibilities for such interdisciplinary studies are discussed.

Keywords: Visual Drag-and-Drop Programming Language, E-Learning, JavaScript.

GİRİŞ

Günümüzde problemleri matematiksel algoritmalara dayandırarak çözmek üzere geliştirilen yöntemler "Matematiksel Programlama" adı altında üniversitelerde okutulan bir ders halini almıştır (İTÜ, Matematiksel Programlama). Özellikle ilköğretimden itibaren kodlama eğitiminin verilmesi amaçlandığına göre bu alışkanlığın yer etmesi için görsel özelliklere sahip programlar ile uygulamalar geliştirmek son derece önemli görülmektedir. Bir matematik probleminin çözümünde çeşitli teorem, çözüm teknikleri ve işlemlerin belirli bir mantıksal düzen içerisinde gerçekleştirilmesi gerekliliği matematik eğitiminde görsel sürükle bırak programlama dilleri araçlarının kullanımına imkan sağlamaktadır.

Bu çalışmada Google tarafından geliştirilen görsel sürükle bırak programlama dili kütüphanesi olan Blockly yazılımı kullanılarak matematik teorem, teknik ve işlemleri için bileşenler (blocks) sunan bir

uygulama geliştirilme süreci ve deneyimi incelenmektedir. Çalışmanın ilerleyen bölümlerinde çalışmada kullanılan yöntemlerin açıklandığı görsel sürükle-bırak programlama dilleri, matematik eğitiminde kullanılan uygulama araçları, matematik problemlerinin algoritmik çözümü bölümlerinden oluşmaktadır. Ayrıca tartışma ve sonuç bölümünde ise disiplinler arası alanlarda gerçekleştirilecek çalışmalar için program geliştirme platform olanakları tartışılarak çalışmada elde edilen bulgular ve öneriler sunulmaktadır.

YÖNTEM

Bu bölümde Görsel Sürükle ve Bırak Programlama Dillerinin genel özellikleri açıklanmış olup, Matematik eğitiminde kullanılan uygulama araçları Blogly yazılımı için kısaca tanıtılmıştır. Ardından matematiksel bir problemin algoritmik uygulamasını Blogly’de geliştirmek amacıyla örnek bir yazılım geliştirilmiştir. Yazılım geliştirme sürecinde işlem önceliğini gibi durumları ifade etmek amacıyla esnek çözümler önerilmiştir.

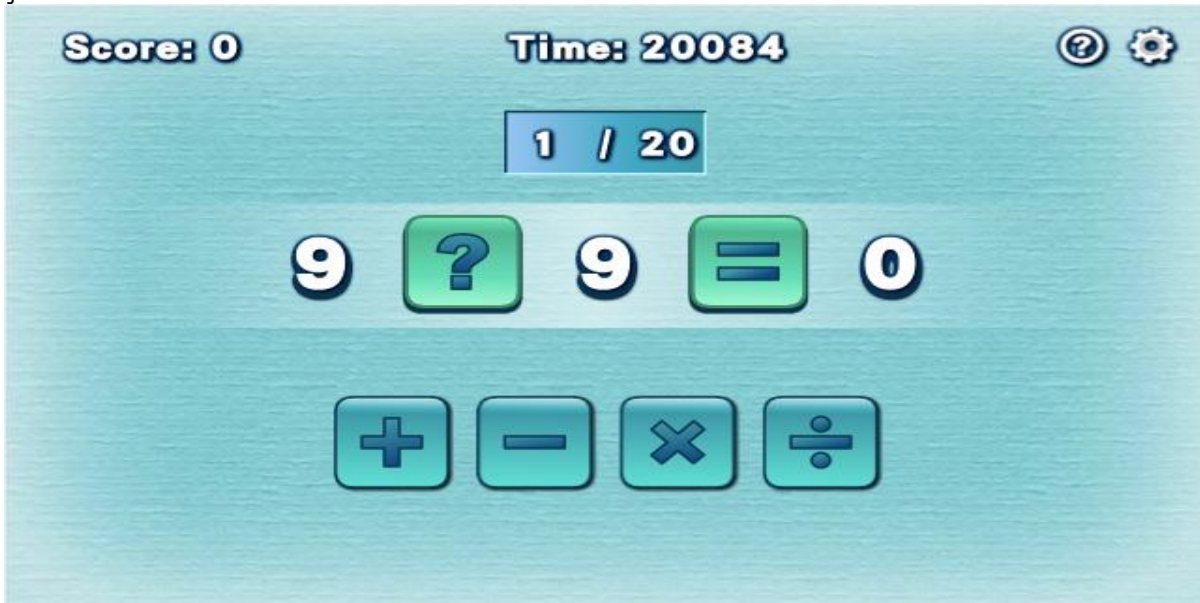
Görsel Sürükle ve Bırak Programlama Dilleri

Günümüzde birçok farklı platformda Görsel Sürükle ve Bırak (GSB) programlama dilleri geliştirilmektedir. İlk dönem örneklerle bakıldığında çoğunlukla masaüstü uygulama, daha sonraları ise web teknolojilerinin yaygınlaşması ile web tabanlı uygulamalar geliştirilmeye başlanmıştır. Google tarafından geliştirilen ve yaygın olarak kullanılan görsel sürükle bırak programlama dili kütüphanesi olan Blockly yazılımı; açık kaynak lisansı ile dağıtılması yaygınlaşmasının önünü açmış olup özellikle oyun ve robot türü uygulamaların geliştirilmesine olanak sunmuştur.

GSB programlarının kullanıcı arayüzlerinin ortak özelliği işlevlere karşılık gelen görsel simgelerden oluşan araç kutuları yer almaktadır. Genel olarak bu araç kutularının işlevleri Hareket, Görünüm, Ses, Veri, Olay, Kontrol, İşleç ve Algılama şeklinde sınıflandırılmış olarak yer almaktadır. Ayrıca birçok editör’de özelleştirilmiş blok (block) oluşturma olanağı da sunulmaktadır. Bu sayede çeşitli kod parçacıkları yazılarak robot benzeri harici donanımların özel işlevlerinin de benzetimi yapılmaktadır.

Matematik Eğitiminde Uygulama Araçları

Matematik eğitiminde özellikle görselleştirme için özgün geliştirilmiş birçok araç bulunmaktadır. Bu araçlar web veya masaüstü teknolojilerle ilgili eğitim içeriğine özel işlevler barındıracak şekilde tasarlanmaktadır. Bu araçlar sadece araç kutusu olarak kullanılabilirliği gibi, bir oyun senaryosu içinde de kullanılabilirler. Yaygın olarak bir oyun içerisinde ihtiyaç duyulan işlevleri barındıracak şekilde tasarlanmaktadır.



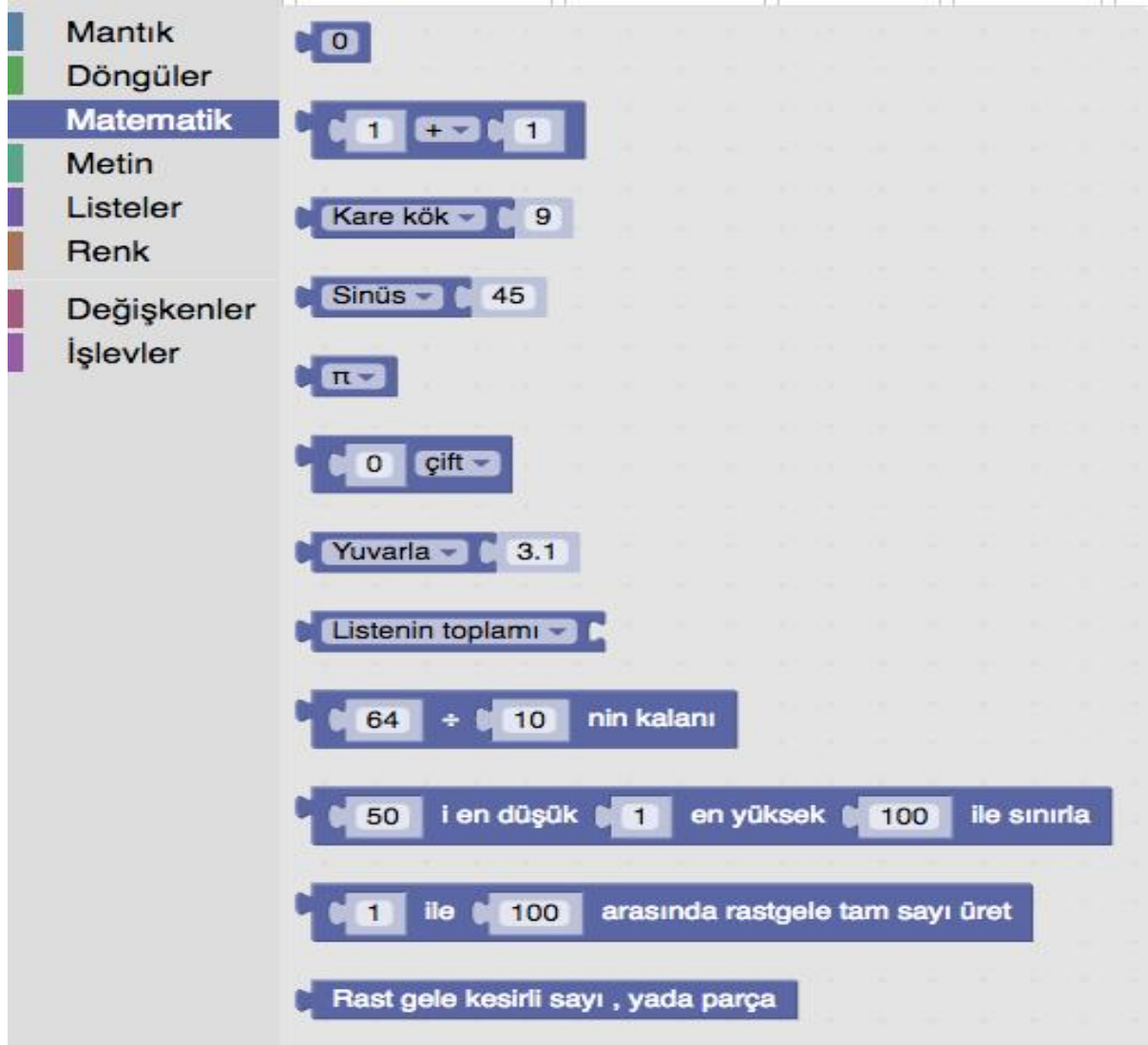
Şekil 1: Matematik Oyun Örneği

Şekil 1'de görülen matematik oyun örneği sadece hesaplanmanın yapılması için gerekli aritmetik işlevleri barındırmakta ve kullanıcının eşitliğin sağlanması için kullanması gereken işlevleri işaretlemesi şeklinde çalışmaktadır (MathGameTime, Matematik Oyun Örneği). Görsel sürükle-bırak programlama dillerinin bu tip oyunlara göre en büyük üstünlüğü problem çözümlerinin algoritmik olarak ifade edilmesinin yanısıra öğrencilerin kendilerinin de bu tip oyunlar tasarlayabilmesine olanak sağlamasıdır. Bunun yanısıra kodlama eğitiminde kullanılan görsel sürükle-bırak programlama dillerinin araç kutuları da çeşitli temel matematik işlevlerini gerçekleştirebilecek araçlar barındırmaktadır. Şekil 2'de blockly uygulamasının araç kutusundaki matematik işlevler görülmektedir (Blockly, Matematik İşlevleri).

Bu işlevler aşağıdaki başlıklarda sınıflandırılabilir;

- Temel Aritmetik
- Trigonometri
- Sabitler
- Diziler/Listeler

Bu işlevler birçok alanda kullanılan temel işlevler olmakla beraber, halen bazı tamamlayıcı işlevlere de ihtiyaç duyulduğu gözlenmektedir. Örneğin aritmetik işlevlerde işlem önceliği kuralının göz ardı edilmesi için paranteze alma işlevi bulunmamaktadır. Uygulamalarda bu basit işlevin bulunmaması nedeniyle daha çok adımda fazladan işlev kullanılarak gerçekleştirilmesi gerekmektedir. Bu ve buna benzer durumlar matematik problemlerinin algoritmik çözümlerinin ifadesi için zorluklar oluşturmaktadır.



Şekil 2: Blockly Matematiksel Araç ve Fonksiyonlar

Matematik Problemlerinin Algoritmik Çözümü

Bu çalışmada Reel Kök Bulma problemi için gerçekleştirilmesi gereken işlemlerin algoritmik olarak tasarlanması ve uygulanması gerçekleştirilmiştir. Problemin çözümünde sadece matematik işlevler değil mantık, metin ve değişken kategorisindeki atama, eğer ve yaz gibi işlevlere de ihtiyaç duyulmaktadır. Uygulamalarda kullanıcı ile etkileşime girilmeyebilir sadece sabit değerlerle de işlemler gerçekleştirilebilir. Fakat geliştirilen uygulamanın hem test edilmesi hem de ilgi uyandırması açısından kullanıcı etkileşimi barındırması tercih edilebilir.

Uygulama esnasında ihtiyaç duyulan bazı işlevlerin direk yer almaması nedeniyle bazı araçların farklı kullanımları ile çözümler üretilebileceği de deneyimlenmiştir. Örneğin bir değişkenin negatif değerini (tersini) veren bir işlev olmaması nedeniyle çarpma işlevi ilgili değişkenin reel değerini -1 ile çarpmak suretiyle negatif değerini bulmak için kullanılmıştır.

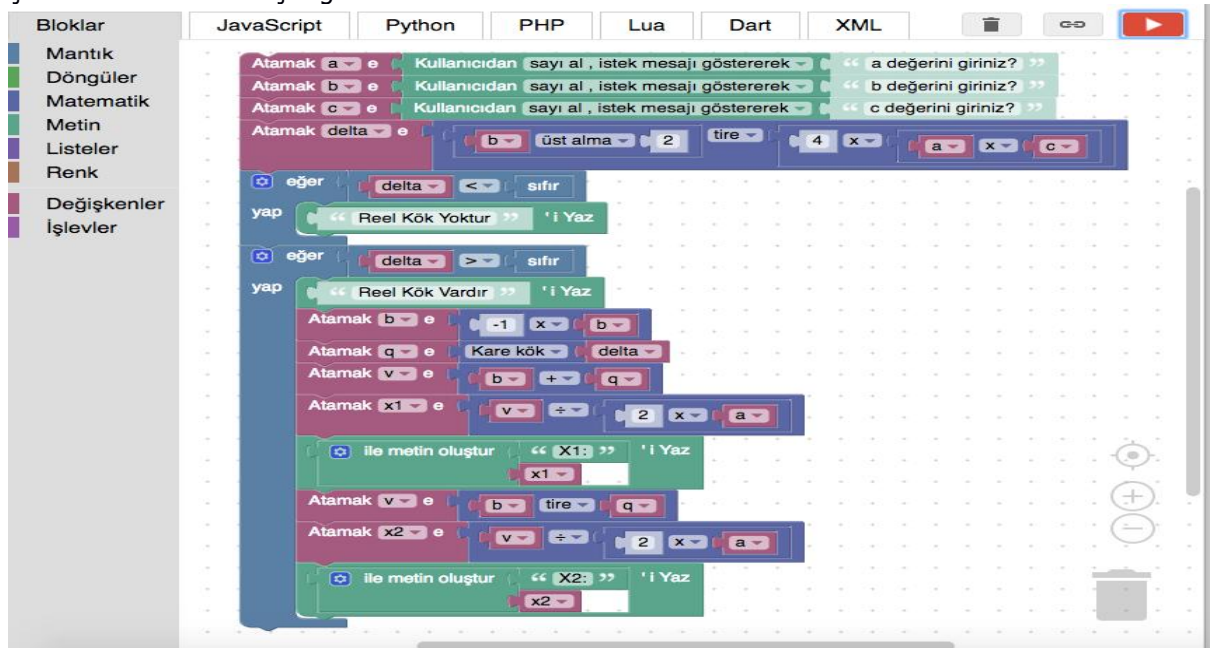
Reel Kök bulmak için gerekli işlem adımlarının sözde (pseudo) kod ile tasarımı aşağıda gösterilmektedir;

Algoritma

- 1- Başla
- 2- a,b,c gir
- 3- $\Delta = b^2 - 4ac$
- 4- $\Delta < 0$ ise "reel kök yoktur" yaz
- 5- $\Delta > 0$ ise "reel kök vardır" yaz
- 6- $x_1 = \frac{-b - \sqrt{\Delta}}{2a}$, $x_2 = \frac{-b + \sqrt{\Delta}}{2a}$
- 7- x_1, x_2 yaz
- 8- Dur

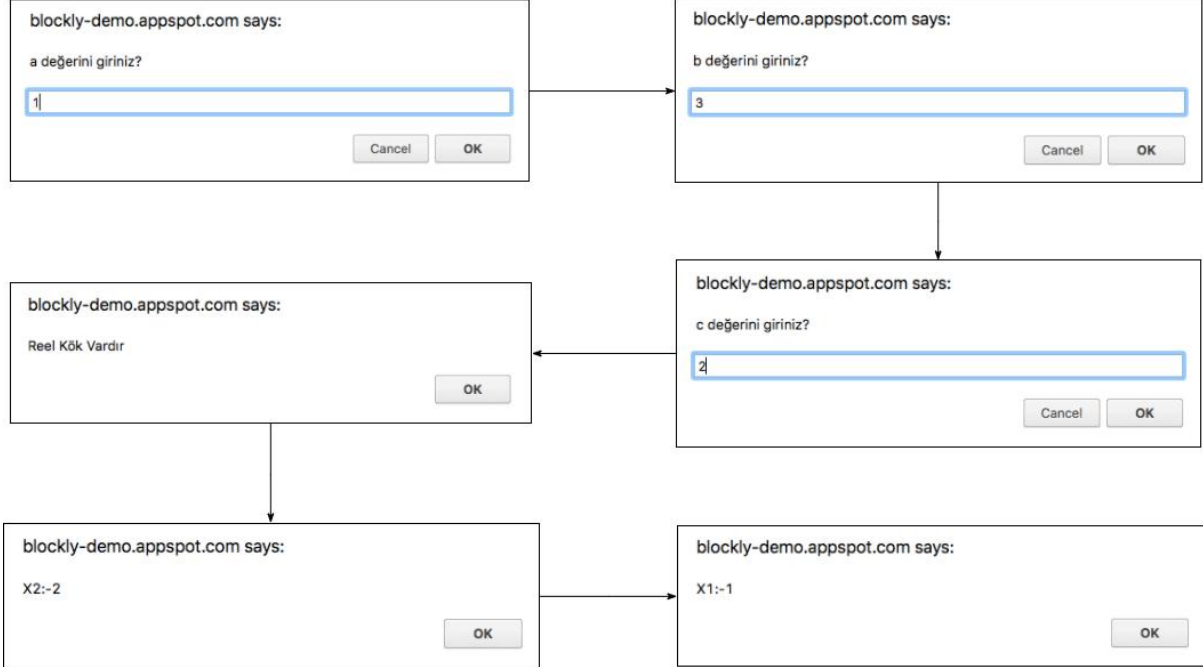
Şekil 3'te ise Reel Kök bulmak için gerekli işlem adımları kullanıcı ile etkileşime girecek şekilde görsel sürükle-bırak dili ile tasarımı görülmektedir. Uygulamanın gerçekleştirilmesi için sözde kodda öngörülen değişkenlerden daha fazla değişkene ihtiyaç duyulduğu, bunun işlem önceliği ihtiyacının karşılanmaması veya hesaplanan bir değer uygulama içerisinde farklı adımlarda tekrar kullanılması ihtiyacından kaynaklandığı gözlenmektedir.

Bu tip tekrarlı kullanılacak bölümler için özel işlev tasarlanabilme imkanı görsel sürükle-bırak programlama dillerinde sağlanmaktadır. İşlev tasarlayabilme matematik probleminin algoritmik çözümünde kod karmaşıklığını azaltacak önemli bir özelliktir.



Şekil 3: Geliştirilen Reel Kök Bulma Uygulaması

Şekil 4'te görülen şemada görüldüğü üzere reel kök bulma probleminin algoritmik çözümü işletilmiş ve algoritmanın doğruluğu test edilmiştir. Bu tip testlerde sınır değerler ve istisna yaratabilecek değerlerin kullanılmasının algorithmadaki hataları hızlıca görmemize olanak sağladığı gözlemlenmiştir.



Şekil 4: Algoritma Sonucunda İşletilen Program Çıktıları

TARTIŞMA VE SONUÇ

Yapılan değerlendirmeler sonucunda matematik problemlerinin algoritmik olarak çözümünde görsel sürükle-bırak programlama dillerinin kullanımının uygun olduğu fakat bu alanda daha verimli uygulamalar geliştirilebilmesi için yeni işlevlere de ihtiyaç duyulduğu gözlemlenmiştir. Negatif değer alma ve işlemleri paranteze alma gibi bu çalışmada ihtiyaç duyulan bu işlevler örnek olarak verilebilir.

Bunun yanında uygulama alanlarını arttırabilmek adına bu yeni işlevlere bilimsel hesaplama, finans formülleri veya istatistik ile ilgili özel işlevlerin de dahil edilmesi ile matematik temelli gelişmiş problemlere dair algoritmik tasarımların gerçekleşmesinin mümkün olabileceği öngörülmektedir.

Not: Bu çalışma 26-27 Ekim 2017 tarihlerinde Antalya'da düzenlenen 6'ncı Eğitim ve Öğretim Çalışmaları Dünya Kongresi'nde bildiri olarak da değerlendirilmiştir.

KAYNAKÇA

Blockly, Matematik İşlevleri, 06.11.2017 tarihinde <https://developers.google.com/blockly/> adresinden alınmıştır.

İTÜ, Matematiksel Programlama. 26.10.2017 tarihinde <http://web.itu.edu.tr/kabak/dersler/END508/> adresinden alınmıştır.

MathGameTime, Matematik Oyun Örneği, 23.10.2017 tarihinde <http://www.mathgametime.com/games/quick-math> adresinden alınmıştır.