



Yeni Bir Programlama Ortamı : Scratch

(Hazırlayan : Muhammet Yorulmaz)

Aslında “scratch” kelimesi her ne kadar “çizmek” ve “kedi tırmığı” anlamlarına gelse de isim olarak buradan türetilmemiştir. Bu terim Disk Jokey (DJ) olarak isimlendirilen kişilerin taş plakları çalarken ellerini disk üzerinde hareket ettirerek oluşturdukları şarkı aralarındaki efektler içinde kullanılmaktadır. Geliştiriciler bu yaratıcı eylemden esinlenerek yeni programlama ortamına bu adı vermişlerdir.

Scratch, MIT Üniversitesi’nde Amerikan Ulusal Bilim Kurumu tarafından desteklenen bir proje olarak geliştirilmiştir. Proje 2003 yılında başlamış 2007 yılında bitirilmiştir. Ekonomik olarak gelişmemiş toplum kesimlerindeki faaliyet gösteren okul dışındaki merkezlerde vakit geçiren gençlerin teknolojiyi kullanmada daha maharetli olmalarını yani teknolojik akıcılıklarını arttırmayı amaçlamaktadır [1]. Teknolojik akıcılık kavramından anlatılmak istenen kişinin teknolojik araçları nasıl kullanılabileceğini bilmekle beraber bu araçları kullanarak anlamlı şeyler yapabilmesi üretebilmesidir.

Teknolojik akıcılığı dildeki akıcılığa benzetebiliriz. Nasıl ki öğrenilen bir yabancı dili akıcı olarak kullanabilmek için o dildeki sözcükleri ve deyimleri bilmek yetmez. Bu sözcük ve deyimleri kullanarak kompozisyonlar, mektuplar hikayeler yazmak konuşmalar yapmak gerekir. Benzer şekilde teknolojiyi de akıcı olarak kullanabilmek için bilinenleri birleştirerek yeni ürünler geliştirmek gerekir.

MIT Medya Laboratuvarı’ndan Mitchel Resnick, UCLA Üniversitesi’nden Doçent Dr. Yasmin Kafai ve yine MIT Medya Laboratuvarı’ndan John Maeda araştırma grubuna öncülük etmişlerdir [1].

Çoğunlukla bilgisayar programlamaya kullanımı zor olan programlama dillerinin öğretimi ile giriş yapılır. Ve öğretimin ilk aşamalarında yapılan etkinlikler gençlerin ilgileri ile örtüşmez. Sonuçta pek çok insanın bilgisayar programlamaya bakışı dar bir alanda kalır. Toplumun küçük bir kesimi için uygun bir etkinlik olmaya devam eder. Scratch ile beraber bilgisayar programlama herkes için daha kolay anlaşılabilir yapmak mümkündür [1].

Bu programlama dilinin genel amaçları aşağıdaki gibi sayılabilir.

- 1- Okul dışı merkezlerdeki teknoloji kullanımını temel bilgisayar kullanımından daha öteye taşıyarak genç insanların etkin olarak bilgi teknolojilerini daha derin akıcı kullanmalarını sağlamak
- 2- Gençlere yeni teknolojilerde yaratıcı ve tasarımcı olmak için gerekli olanakları arttırmak
- 3- Gayri resmi bilişim eğitimi ve matematik eğitimi için yeni bilgi iletişim teknolojilerinin yenilikçi tasarımların ortaya çıkmasını sağlamak
- 4- Araştırma tabanlı eğitim teknolojilerini daha geniş ve farklı izleyicilere ulaştırmak
- 5- Gençlerin coğrafi, kültürel ve dil engeli olmaksızın işbirliği yapabilmelerini sağlamak

Temel Özellikleri

Geliştirilen bu yeni programlama ortamı tasarlanırken aşağıdaki temel özelliklere göz önünde tutulmuştur [2].

Blok yapılarla programlama: Öğrencilerin program parçalarını grafiksel blok yapıları sürükleyip bırakarak yapabilmeleri sağlanmıştır. Farklı komutlar alt alta getirildiklerinde birbirine tam uyan geometrik şekillerle ifade edilir. Böylece bir programlama dilinin öğreniminde yeni başlayanların en büyük sorunu olan sözdizimi (syntax) hataları ortadan kaldırılmıştır. Daha da fazlası öğrenciler komut yazılımlarını ezberlemek ya da öğrenmek zorunda kalmayacaklardır [3].

Bu bloklar ile yapılan programdaki her bir karakter nesnesi için birbirine paralel çalışan alt program parçaları (prosedür, fonksiyon, alt rutin) yaratılabilir.

Zengin medyanın programlanabilir kullanılması: Geleneksel programlama öğretimindeki ilk etkinlikler sıkıcı sayı, dizgi ve basit grafik işlemleri üzerine kuruludur. Oysa bu yeni programlama ortamında resimler, animasyonlar, film parçaları ve sesler üzerinde işlemler yapılabilir. Bu tür materyaller gençlerin ilgilerini daha fazla çekeceği için öğrenme güdülerini artacaktır.

Daha fazla paylaşılabilirlik: Gençler yaptıkları projeleri web üzerinden arkadaşları ile paylaşabilir ve fikir alışverişlerinde bulunabilirler. Geliştirdikleri teknikleri işbirliği yaparak birbirlerine aktarabilirler.

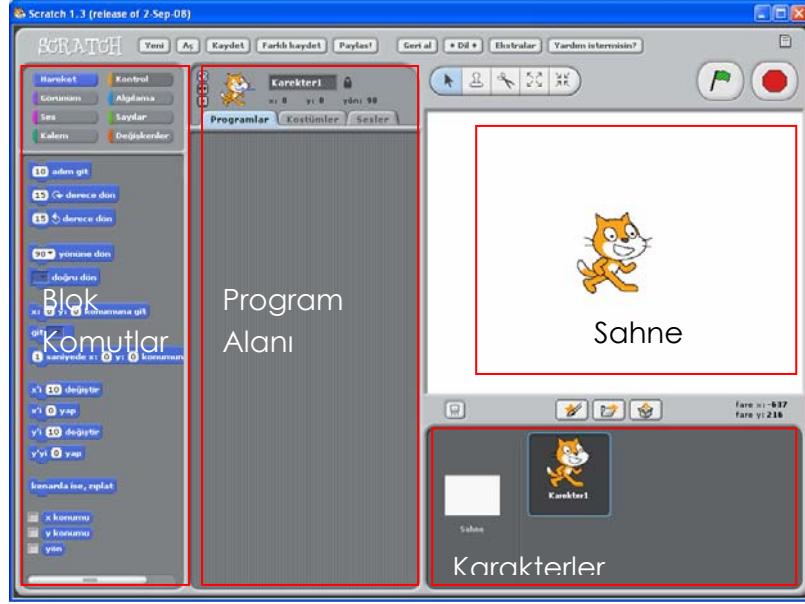
Gerçek dünya ile bütünleşme: Bilgisayarlara bağlanabilecek olan bir takım donanım elemanları ile gençler gerçek dünyadaki makineleri de kontrol edebilirler. Örneğin oyuncak bir robotun kontrolü gerçekleştirilebilir.

Çoklu dil desteği: Farklı dillerde konuşan genç insanların dil engeline takılmadan işbirliği yapabilmeleri için pek çok dili destekleyebilir. Yazılan programlar yukarıdaki “Dil” düğmesi ile anında desteklenen 42 dilden herhangi birine dönüştürülebilir.

Scratch programlama ortamı bir stüdyo gibi yapılandırmacı öğrenme teorisi baz alınarak tasarlanmıştır. Yapılandırmacı yaklaşım Seymour Papert tarafından ortaya atılmış, eğitim ve öğrenimi derinden etkilemiştir [3]. Bu teoriye göre öğrenme en iyi öğrenenlerin sürece katılmaları ile mümkün olabilir [4]. Genç insanlar bu ortamla proje geliştirerek öğrenmektedirler.

Programlama Ortamı

Scratch programı, scratch.mit.edu web adresinden ücretsiz olarak indirilebilir. Bilgisayara kurma işlemi bittikten sonra çalıştırıldığında Şekil-1 deki gibi bir ekranla karşılaşılır.



Şekil-1 Scratch Ekranı

Ekranın sol tarafında farklı geometrik şekillerle gösterilen blok diyagram halindeki komutlar bulunmaktadır. Sağ altta gerçekleştirilecek projede kullanılacak olan karakter nesnelere (Orijinal olarak bu karakterlere "sprite" adı verilmektedir). Karakterler program içinde yapılabileceği gibi daha önceden çizilmiş olanlar ya da kişisel arşivde bulunan resimlerden de kullanılabilir. Seçilen nesne için çalıştırılacak olan kodlar ekranın sol tarafındaki bloklar ile sahne arasında bulunan alana sürükleyip bırakılarak alt program parçaları oluşturulur.

Kodlamaya Başlama

Projeye başlarken, projede kullanılacak nesnelere ekranın sağ alt taraftaki bölüme getirilir. İstenirse var olan kütüphanedeki nesnelere ya da kullanıcının kendi çizdiği nesnelere oluşturulabilir. Burada anlatılacak örnek proje için iki adet karakter Şekil-2 deki gibi nesne alanına taşınmıştır.

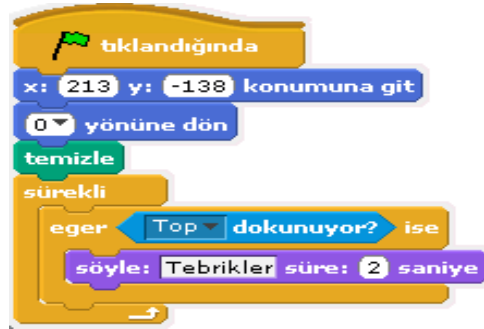


Şekil-2 Örnek Projedeki Karakterler ve Sahne

Bu örnekte yukarıdan bakılan bir labirentte “Kedi” karakterinin topa ulaşmasını sağlayan küçük bir oyun benzetimi yapılmaya çalışılacaktır. Sahne labirent izlenimi verecek şekilde dik çizgiler çizilerek oluşturulmuştur.

“Top” karakteri Sahnemizin sol üst köşesine yerleştirilmiştir. Oyuncu “Kedi” karakterini sol üst köşeye ok tuşlarını kullanarak ulaştıracaktır. “Kedi” karakterinin topa ulaşmak için izlediği rotayı belli etmek için karakter yol aldığı sürece izlediği yolu çizerek ilerleyecektir.

Bütün projelerde programı başlatmak için sağ üst köşedeki yeşil bayrak düğmesine tıklanır. Programları durdurmak için bu düğmenin yanındaki kırmızı düğmeye basılır. Proje yeşil bayrağa tıklanarak başlatılacağı için bazı ilklendirme işlemleri ve çalışma boyunca yapılacak kontroller bu eyleme bağlantılı bir biçimde Şekil-3 deki gibi oluşturulmuştur. Bu eylemler “Kedi” karakteri seçildikten sonra oluşturulmalıdır.



Şekil-3 Proje Başlama ve Sonrasında Yapılacaklar

Oluşturulan bu ilk program parçasında “Kedi” karakteri bir önceki oyunda ekranın farklı bir yerinde kaldığı düşünülerek başlangıçta her zaman ekranın sağ alt köşesine “x: 213 y: -138 konumuna git” emri ile konumlandırılmaktadır. Scratch sahnelerine farenin hareket ederken göreceli koordinat sisteminin değerleri sahne altında verilmektedir. Bu koordinat sistemine göre ekranın tam ortası x:0, y:0 konumu olarak belirlenmiştir. X değerleri -240 ile 240 arasında Y değerleri ise -180 ile 180 arasında değerler alabilir.

Bir sonraki adımda “0 yönüne dön” emri ile önceki oynamalardan ekranda yönü değişmiş olan “Kedi” karakterinin yönü yukarı olarak ilklendirilmektedir. Sonraki “Temizle” emri rota çizimlerini ve diğer çizimlerin temizlenmesini sağlar.

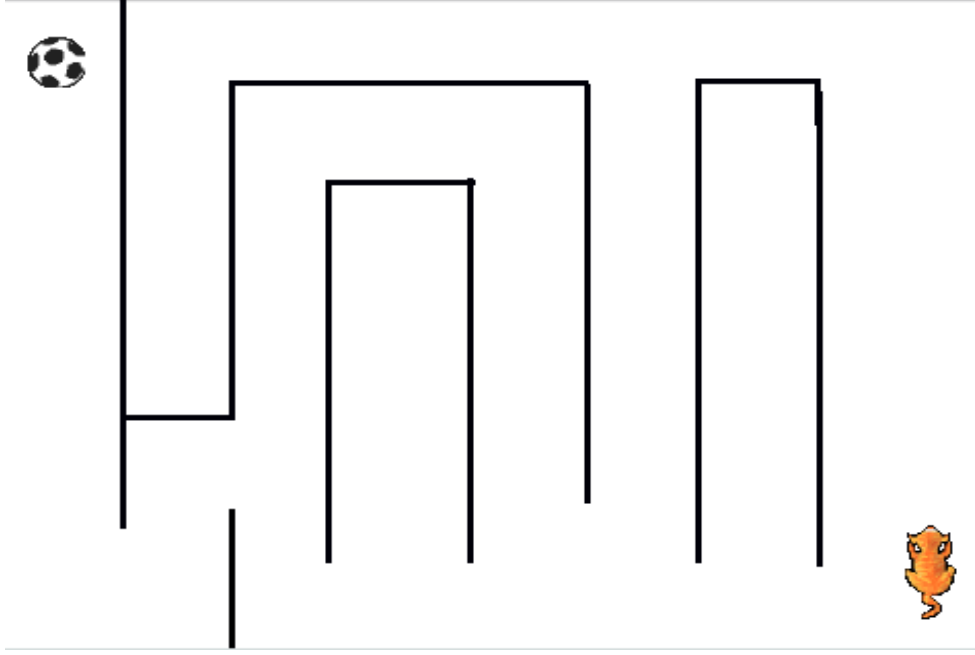
Bundan sonra “Sürekli” döngü emrinin içinde program boyunca devamlı olarak bu döngünün içindeki koşul kontrol edilecektir. Aynen emirde de belirtildiği gibi “Kedi” karakteri “Eğer Topa dokunuyor ise” 2 saniye süre ile “Tebrikler” konuşma balonu görüntülenecektir.

Oyuncu ok tuşlarına bastığında da “Kedi” karakterinin basılan yönde ilerlemesi Şekil-4 deki program parçacıkları ile kontrol edilebilir.

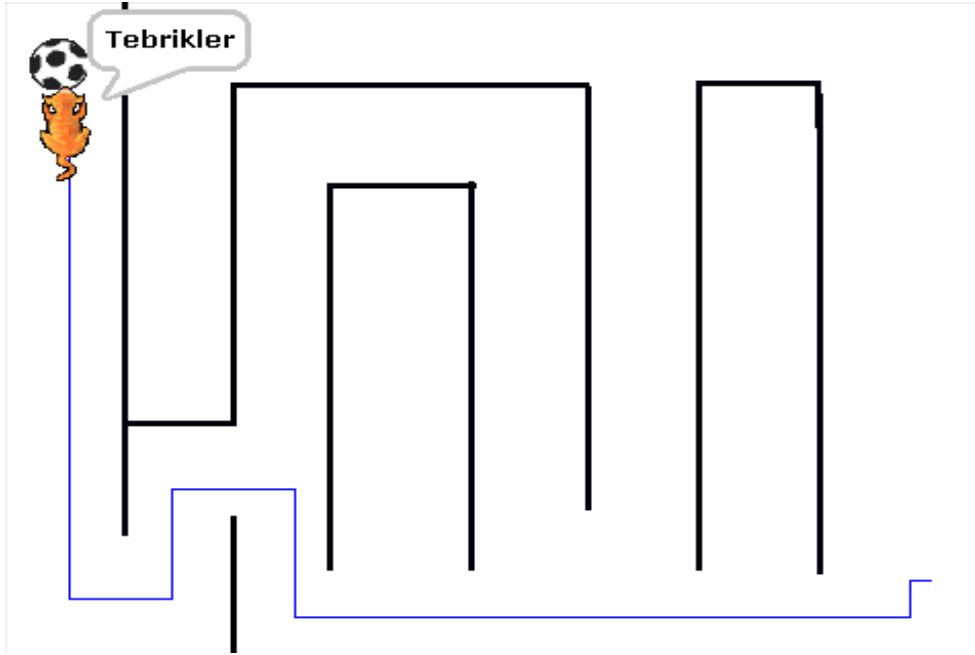


Şekil-4 Ok tuşlarına basıldığında yapılacaklar

Bu parçacıklara göre “Kedi” karakteri basılan ok tuşuna uygun yöne dönerek “kalemi bastır” komutu ile rota çizerek “10 adım git” emri ile 10 adım hareket edecektir. Projenin ilk başlama hali ve hedefe ulaşılma anındaki hali Şekil-6a ve Şekil-6b de verilmiştir.



Şekil-6 a Oyunun başlangıç durumu



Şekil-6 b Oyunun son durumu

Gerçekleştirilen proje tanıtım amaçlı olduğu için bu hali ile aslında tam olarak bir labirent oyunu değildir. Örneğin labirent duvarlarını geçememe durumu kontrol edilmemiştir.

Kaynaklar

- [1] Resnick, M., Kafai, Y., Maloney, J., Rusk, N., Burd, L., & Silverman, B. (2003). [A Networked, Media-Rich Programming Environment to Enhance Technological Fluency at After-School Centers in Economically-Disadvantaged Communities](#). Proposal to National Science Foundation.
- [2] Maloney, J., Burd, L., Kafai, Y., Rusk, N., Silverman, B., and Resnick, M. (2004). [Scratch: A Sneak Preview](#). Second International Conference on Creating, Connecting, and Collaborating through Computing. Kyoto, Japan, pp. 104-109.
- [3] Peppler, K., & Kafai, Y. (2005). [Creative coding: The role of art and programming in the K-12 educational context](#).
- [4] Resnick, M., and Silverman, B. (2005). [Some Reflections on Designing Construction Kits for Kids](#). Proceedings of Interaction Design and Children conference, Boulder, CO.