

Proje Ana Alanı : Yazılım

Proje Tematik Alanı : Robotik ve Kodlama

Proje Adı (Başlığı) : Ulusal Yazılım ve Kodlama Eğitim Sistemi (UYAKES)

Özet

Dünyada kodlama eğitimine verilmesi gereken önem gün geçtikçe artmaktadır. Ülkemizde ise maalesef henüz ilk, orta ve lise öğretiminde (özellikle devlet okullarında) yeterince eğitim verilememektedir. Yapılan araştırmalar sonucunda bu durum için çözümler üretildiği bulunsa da yeterince kapsamlı olmadıkları tespit edildi.

Bu proje ile öğrencilerin eğlenerek ve öğrenerek kendini yazılım ve kodlama alanında geliştireceği bir platform olan UYAKES'i (Ulusal Yazılım ve Kodlama Eğitim Sistemi) geliştirerek öğrencilerin kullanımına uygun bir sistem oluşturmak hedeflendi. Bu sayede her öğrenci mezun olduğunda en azından temel seviyede yazılım ve kodlama bilecektir.

Bunun için öncelikle eğitim sistemindeki İngilizce kelimeler gözden geçirilip ilkökul seviyesine yazılım öğrenimi için gereken kelimeler eklendi. Sonra da UYAKES için bir internet sitesi oluşturuldu. Blog, sınıf sistemi gibi birçok özellik eklendi.

Son olarak UYAKES'in eğitim sistemine eklenebilmesi için eğitim sistemini incelendi ve sisteme entegre edilmesi için gereken şartlar yerine getirildi. Yeni bir örnek müfredat oluşturuldu. Bu doğrultuda öğrencilerin motivasyonunu korumak için gayret gösteren öğrencilere verilecek bir belge hazırlandı. Örnek ders hazırlandı. Yazılım ve kodlama derslerinde yazılı, sözlü gibi not sistemlerinin uygulanmamasına karar verildi. UYAKES'in ilk önce uygun şartları içeren okullarda test edilip eksikler giderildikten sonra kademe kademe uygulanmaya başlanması kararlaştırıldı. Bu sayede çıkabilecek olası sorunların daha kontrollü olarak giderilebileceği fark edildi.

Geliştirilen sistemde 2, 3 ve 4. Sınıfların İngilizce derslerine yazılım ve kodlama alanında kullanacakları İngilizce kelimeler eklendi. Bu sayede Öğrencilerin 5. Sınıftan itibaren başlayacakları yazılım ve kodlama derslerinde kullanacakları kelimelere aşina olmaları hedeflendi.

Anahtar kelimeler: Yazılım, Eğitim, Kolaylık, Erişilebilirlik, Sistem

Project Main Area : Software

Project Thematic Area : Robotics and Coding

Project Name (Title) : National Software and Coding Education System (UYAKES)

Abstract

The importance to be given to coding education in the world is increasing day by day. In our country, unfortunately, there is not enough education in primary, secondary and high school education (especially in public schools). As a result of the researches, although it was found that solutions were produced for this situation, it was determined that they were not comprehensive enough.

With this project, it was aimed to create a system suitable for the use of students by developing UYAKES (National Software and Coding Education System), which is a platform where students can develop themselves in the field of software and coding by having fun and learning. In this way, every student will know at least basic software and coding when they graduate.

For this, first of all, the English words in the education system were reviewed and the words required for software learning at the primary school level were added. Then a website was created for UYAKES. Many features such as blog, class system have been added.

Finally, the education system was examined in order for UYAKES to be added to the education system and the necessary conditions were met for its integration into the system. A new sample curriculum was created. In this direction, a document was prepared to be given to students who make an effort to maintain the motivation of the students. Sample lesson has been prepared. It was decided not to apply grading systems such as written and oral in software and coding courses. It has been decided that UYAKES will be tested in schools with appropriate conditions and will be implemented gradually after the deficiencies are corrected. In this way, it was realized that possible problems could be solved in a more controlled manner.

In the developed system, English words that will be used in the field of software and coding were added to the English lessons of the 2nd, 3rd and 4th graders. In this way, it was aimed that the students become familiar with the words they will use in the software and coding lessons that they will start from the 5th grade.

Keywords: Software, Education, Convenience, Accessibility, System

Amaç

Bugünün öğrencileri hangi mesleği seçerlerse seçsinler, artık onlardan teknolojiye hakim olmaları bekleniyor. Kodlama en iyi erken yaşlarda öğretiliyor. Amacımız bu doğrultuda, projenin kullanıma açılması her okulda kullanılır hale getirilip ilköğretim ortaokul ve lise öğrencilerinin yazılım ve kodlama eğitimi alması. Bunun için en uygun erişilebilir ve anlaşılır platformu oluşturarak öğrencilerin fikir alışverişi içinde eğlenerek eğitim almasını sağlamak.

Bunun için Türkiye’de bilgisayar ve internet erişiminin okullara getirilmesi, öğretmenlerin bilişim teknolojilerini kullanmalarına yönelik eğitilmesi ve bilişim teknolojilerinin eğitim sistemi ile bütünleşme çalışmaları bir süredir devam etmektedir. Ancak yaptığımız inceleme ve anketlere göre bunlardan elde edilen verim çok düşüktür.

Bu sorunun çözülmesi için bizce yapılabilecek en iyi şey çocukları sıkmadan eğlendirerek ve öğreterek bir eğitim sistemi oluşturmaktır.

Giriş

Yazılım ve kodlama eğitiminin birçok faydası olduğu tespit edildi. Örnek olarak yaratıcılık, takım çalışması, işlemsel düşünme yeteneği gösterilebilir. Bu sebeple çocuklara yazılım ve kodlama eğitimi verilmesinin ülkemize ve çocukların kendilerine birçok faydası dokunacaktır. Bunun sağlanması için en uygun yöntem ise herkesin kolayca ulaşılacağı bir platform yapılması ve uygulanacak sistemin oluşturulması olduğuna karar verildi. Bu yazılım ve kodlama eğitimi sınavlı bir şekilde ama not verilmeden yapılması ne kadar belli bir kısım öğrencinin ve velinin bu eğitimi önemsiz görmesine sebep olacak olsada, büyük bir kısmın verimini ve aldığı zevki arttıracaktır. Bu sınavlar dönem aralarında günlük yaşamdan problemler çözme şeklinde olacaktır.

Öğretmen tarafından verilen bu soruna her öğrenci belli bir süre içinde kendi cevabını verecek ve cevabı hazırladığımız kriterlere göre değerlendirilecek. Belirlediğimiz bu kriterlere göre her proje ya da birden ona kadar sayısal bir değer alacak. Sayısal değerler verildikten sonra yine belirlediğimiz kurallara göre bir sıralama yapılacak ve bu sıralamaya göre belli bir sayıda öğrenciye madalya ya da sertifika gibi ödüller verilecek. Bu ödüller hem çocuklar arasındaki çekişmeyi arttıracak hem de başkalarına bu konuda bilgi sahibi olduklarını kanıtlamaları için ellerinde bir belge olacak.

Özellikle ortaokul ve ilköğretim öğrencilerine verilecek yazılım ve kodlama dersinin içeriği, o sene gördükleri İngilizce dersi müfredatına ek eklenecek İngilizce yazılım terimleri ile birlikte belirlenecek. Bu seviye sınıftan sınıfa geçmekle beraber, belirlediğimiz haftalık program ile yapılacaktır. Düzenli bir şekilde yapılırsa birçok ülkede olduğu gibi ülkemizde de her öğrenci belli bir seviyede kodlama ve yazılıma hakim olacak.

Yöntem

1.1) Mevcut Durum Araştırması

Araştırma yöntemince ilk olarak internette yerli ve yabancı sitelerde bulunan makaleler incelendi, bu eğitimi uygulayan ülkelerin öğrenci profilleri karşılaştırıldı ve projenin kullanılması sonucunda ne gibi faydaları olacağını belirlendi. Yaptığımız araştırmalara göre Bulgaristan, Hong Kong, Kıbrıs, Çek Cumhuriyeti, Danimarka, Estonya, Yunanistan, İrlanda, İtalya, Litvanya, Polonya, Portekiz ve İngiltere gibi birçok ülke ilkokuldan itibaren kodlama eğitimine son derece önem veriyor. Örnek vermek gerekirse, İngiltere 5-16 yaş için yazılım eğitimi resmen başlattı. Amaç ise “Başkasının yazdığı programı kullanmak yerine, kendi programını yazmanın öğretilmesi” olduğu söylendi.

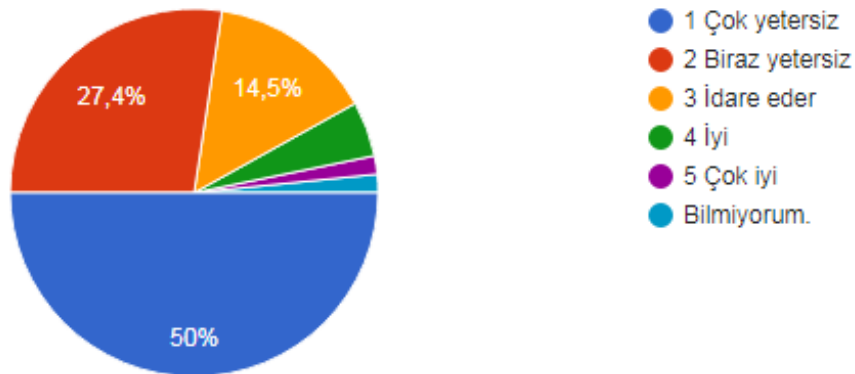
Ülkemizde Eğitim Bakanlığı tarafından yayınlanan ders çizelgesinde 2013-2014 eğitim-öğretim yılından beri ortaokul 5. ve 6. sınıflarda zorunlu 7. ve 8. sınıflarda seçmeli ders olarak yer almıştır.

Ancak yapmış olduğumuz ankette (Şekil 1) ve özellikle devlet okullarında gördüğümüz yazılım eğitimi çok yüzeyseldir ve etkili değildir. Ayrıca zorunlu ders olması ve not sistemine tabi tutulması, bu dersi herkesin bilmesi gereken interaktif bir ders yerine; not için çalışılan ve hemen unutulmuş bir ders haline getirmiştir.

Şekil 1 :

Günümüzde ülkemizde bilişim teknolojileri eğitimi 5 üzerinden puanlasaydınız kaç puan verirdiniz?

62 yanıt



Not : “4 İyi” seçeneğini işaretleyenlerin oranı %4.8, “5 Çok iyi” seçeneğini işaretleyenlerin oranı %1.6 dır.

1.2) Sistemlerin İncelenmesi

Sistemlere örnek vermek gerekirse 2015 yılında Uluslararası Mühendislik için

Öğretim, Değerlendirme ve Öğrenme Konferansı(TALE)’nda yapılan bir çalışmaya göre “Kodlama eğitimini okul müfredatına entegre etmenin başarısı, okullardaki öğretmenlerin ve müdürlerin algılarına ve kodlama eğitimine ne kadar inandıklarına ve yenilikçi öğretme ve öğrenmeye ne kadar hazırlandıklarına bağlıdır.” Bu çalışmada yapılan araştırmada Hong Kong’daki 42 ilk ve ortaokulda anket yapan büyük ölçekli bir nicel anketin sonuçlarını bildirmektedir. Anket sonuçlarından da (Şekil 2)belli olduğu gibi hazır kod ve oyuncak gibi daha basit anlaşılır kodlama teknikleri en küçük yaşlarda başlanırken, JavaScript ve Visual Studio gibi daha kompleks ve zor konuların öğretimi 5. ve 6. sınıfı geçmiyor.

Buradan ve başka çalışmalardan [1] da görebileceğimiz gibi erken yaşlarda kodlama eğitimi yapılabilir ve hatta yapılması daha iyi oluyor. Bu sebepten dolayı ülkemizde de gelecekte mühendislik eğitimi teşvik etmenin en önemli anahtarlarından biri olarak kabul edilen kodlama ve yazılım eğitimi en kısa sürede küçük yaşlarda erişilebilir ve öğretilbilir hale getirmemiz gerekmektedir.

Şekil 2: “Hong Kong’da yıl yıl ilkokulda öğretilen kodlama araçları.”

TABLE II. CODING TOOLS TAUGHTS IN SCHOOLS (PRIMARY)

	P1	P2	P3	P4	P5	P6
Scratch	-	-	1	1	9	6
App Inventors 2	-	-	-	-	2	2
Alice	-	-	-	-	-	1
HTML/CSS	-	-	-	-	1	3
JavaScript	-	-	-	-	-	1
Kudo	-	-	1	1	1	2
Game Maker	-	-	-	1	1	1
Touch Develop	-	-	1	1	1	2
Visual Studio	-	-	-	-	-	1
Hour of code	2	2	1	1	1	1
Lego Robot	-	1	2	2	2	2
2DIY	-	-	1	1	-	-
Smart App Creator	-	-	-	1	1	1
None/Not Specified	6	5	4	4	2	2
Total	8	8	11	13	21	25

2.1) Anket oluşturma ve toplanan bilgilerin analizi

Halkımızın mevcut bilişim teknolojileri eğitim sistemi ve ona verilmesi gereken önem hakkında günümüz düşüncelerini öğrenmek ve analiz edebilmek için bir anket düzenlendi. Anketin bilimsel bir dayanak olması da anket yapmamızın bir diğer nedenidir. “UYAKES Projesi Kapsamında Ülkemizde Yazılım ve Kodlama Eğitimi Hakkında Anket” adında bu anket hazırlandı. Pandemi süreci nedeniyle anket internet üzerinden gerçekleştirildi. 18 yaş üstü katılımcıların ankete katılım onayı anketin ilk sorusunda alındı. 18 yaşının altındaki kişiler için ise velilerinden izin alındığının

onaylanmasını zorunlu hale getirildi.

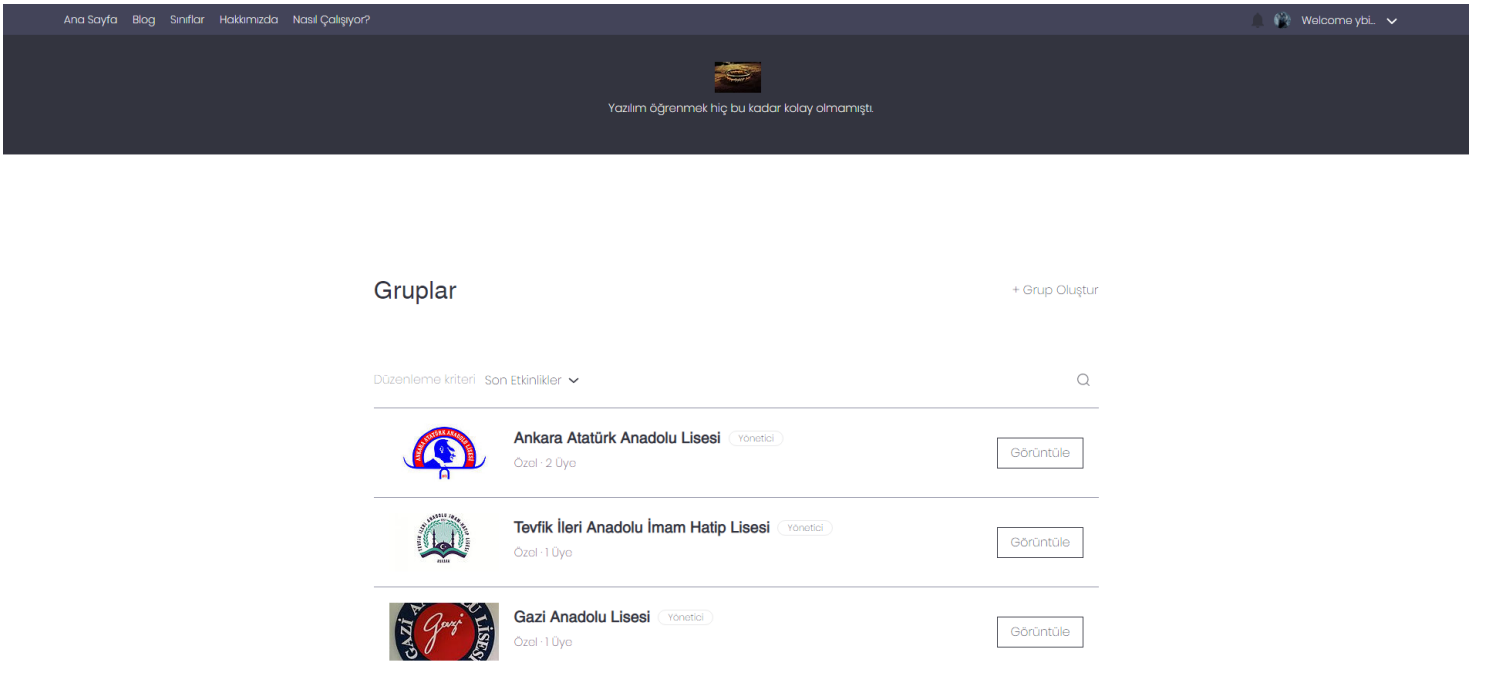
Anketimizin sonuçlarından çıkardıklarımız sonuç kısmına eklendi. Ayrıca sorular, cevaplar ve bilgilendirme metni de ek kısmında (**Ek 1**) gösterildi.

2.2) İnternet sitesinin yapımı

Yaptığımız bu sistemi daha fazla kişiye tanıtılabilmek için ve bir elimizde olması için bir internet sitesi oluşturuldu bu siteye birçok özellik eklendi bu özelliklerden bazıları saymamız gerekirse:

- Bir sınıf sistemi oluşturuldu (**Şekil 3**). Bu sınıf sistemine çeşitli okullar eklenebiliyor sınıfların içine üye olunabiliyor. Sınıf özel herkese açık veya kapalı şeklinde düzenlenebiliyor, bu şekilde gruba giriş çıkışlar düzenlenebiliyor.
- Öğretmenlerin soru paylaşması ve öğrencilerden cevap beklemesi için bir blog sistemi eklendi. Bu blog sistemi sayesinde hem konu anlatımı video ve fotoğraflar paylaşılabilir hem de yorum sistemi sayesinde sorular sorulabilir.
- Projemizin nasıl çalıştığını özet bir şekilde anlatabilmek için nasıl çalışıyor adında değil bölüm eklendi. Bu bölümde (**Şekil 4**) kısa bir şekilde sitemizden ve projemizden bahsederken puanlama sistemi ve projenin temel işleyişi gibi konularda herkese genel bir fikir verecek.

Şekil 3 : <https://ybitiren.wixsite.com/uyakes/groups> linkindeki “Sınıflar” adlı sayfa



Şekil 4 : <https://ybitiren.wixsite.com/uyakes/projem-2> linkindeki “Nasıl çalışıyor?” adlı sayfa



2.3) Eğitim ile ilgisi sistem unsurlarının hazırlanması (Gerçekleşme bilgisi)

Bu kısmı üç temel ayrı alanda değerlendirdik:

- Ders müfredatı
- Sınav sistemi
- Değerlendirme sistemi ve ödüllendirme

2.3.1) Ders müfredatı

Hem kendi gözlemlerimizden hem de yaptığımız ankete göre yazılım ve kodlama eğitimi müfredatı hem kolay ve anlaşılır aynı zamanda bilgi verici ve kapsamlı olması gerekmektedir. Öğrencilerin duydukları her şeyi ezberlemesi yerine konuya ilgi duymaları ve iyi bir şekilde öğrenmeleri gerekmektedir. Problemlerin içinden çıkmak için bilgisayar gibi düşünmeyi yani sayısal düşünmeyi (computational thinking) öğrenmeleri gerekmektedir. Bu şekilde karmaşık problemlerin içinden çıkma konusunda daha başarılı olacaklardır.

Wing [2], öğrencilerin bilgisayarla çözülebilecek şekilde problemleri çözebilmeleri için sayısal düşünmeyi öğrenmeleri gerektiğini önermiştir. Başka bir deyişle, sayısal düşünme zihniyetine sahip öğrenciler, problemleri daha verimli bir şekilde çözmek için algoritmik yaklaşımları (ör. Özyineleme ve yineleme [Recursion ve Iterative]) kullanabilirler . Gerçekten de, mevcut çalışmalar sayısal düşünme eğitimi yoluyla öğrencilerin karmaşık, belirsiz ve açık uçlu problemler ile başa çıkma konusunda güven oluşturduğunu göstermektedir.

Yukarda belirttiğimiz faktörleri de göze alarak ve günümüz müfredatına [3] ek olarak şu şekilde bir müfredat hazırladık:

1 ile 4. Sınıf arasında temel bilgisayar kazanımlarını öğrenmiş bir öğrencinin 5. sınıfta hafta hafta öğrenmesi gerekenler: (şekil 5) (Proje raporu 20 sayfayı geçeceğinden buraya örnek bir sayfası konuldu devamı eklerdedir.)

Şekil 5 : 5.sınıflar için yazılım ve kodlama dersi müfredatı sayfa 2/8 (devamı eklerdedir.)

HAFTA	KONU	İÇERİK VE ETKİNLİKLER	KAZANIM
5. HAFTA	String Metotları	1) Uygulamalı bir biçimde string metotları öğretilir.	Öğrenci; a) String metotlarını(Büyük harf, Başlık harfleri, Boşluk silme vb.) kullanmayı öğrenir. b) Verilen farklı kelimeleri, cümleleri istediği formatta işleyebilir.
6. HAFTA	- Problem çözme - Index Kavramı - Len() – Lenght	1) String ifadeleri işlemek üzerine interaktif olarak alıştırma yapılır. 2) Index ile istenen parçayı alma yöntemleri öğretilir. 3) Len() metodu ile ifadenin karakter, kelime ve item gibi parçalarının sayısını ölçer.	Öğrenci; a) Index kavramını öğrenirken len() ile de adet ölçebilir hale gelir. b) Kullanıcıdan aldığı isim vb. bilgilerde büyük-küçük harf durumunu ayarlamayı öğrenir. c) İfadeye sayı ya da harf bulunma durumunu öğrenebilir.
7. HAFTA	- Liste Kavramı - Liste Metotları	1) Liste kavramını ve özelliklerini öğrenir. 2) Uygulamalı bir biçimde liste metotları öğretilir.	Öğrenci; a) Liste oluşturmaya ve listeye çevirmeyi öğrenir. b) Verilen listeden istediği bilgileri alabilir. c) Listeleri işleyebilir.
8. HAFTA	Problem çözme	1) Listeleri işlemek üzerine interaktif olarak alıştırma yapılır.	Öğrenci; a) Liste metotlarını pekiştirir
9. HAFTA	- Max() - Min() - Genel Tekrar	1) Max(), Min() ile listelerin en büyük ve küçük karakterleri tespit edilir. 2) Geçmiş haftalardaki konular bir özet şeklinde yeniden işlenir.	Öğrenci; a) Genel tekrar ile konularını pekiştirir.

2.3.2) Sınav sistemi

Yazılım ve kodlama eğitiminde sınavların öğrencileri sıkıntıya düşürmeden eğlendirerek ama öğreterek yapılması gerekmektedir. Bize göre, bunu yapmanın en iyi yolu dönem başı iki kere olancal problemler ve onların çözüm yollarının bulunmasıdır. Yaptığımız bu sisteme göre önce öğretmen öğrencilere bir problem belirler öğrenciler bu problemi belli bir süre içinde bitirmeye çalışır, belli bir süre sonra öğrenciler yanıtlarını öğretmene gönderir. Öğretmen bunların hepsini değerlendirir ve yine belirlediğimiz puanlandırma sistemine göre puanlandırır sene sonunda en yüksek puanı alan belli bir sayıda öğrenci belirlenir. Bu öğrencilere belge madalya verilir. Bu şekilde öğrenciler arasındaki rekabet ortamı sağlanır.

Tabii yazılım ve kodlama hakkındaki problemler ve problemlerin çözümlerinin karne notuna işlememesinin dezavantajları da vardır. Bazı öğrenciler karne notunu daha çok önemseydiği için karne notunu etkilemeyen bir derse öncelik vermeyebilirler. Bu durumun engellenebilmesi için nitelikli öğretmenlerin öğrencilerin konuya olan ilgisini yüksek tutması gerekir.

Sınavlar okuldaki sınav haftalarından 1 veya 2 hafta önce yapılır bu şekilde diğer sınavların yazılım ve kodlama problem çözümünün önüne geçilmesi önlenir. Sınav konsepti kopya ihtimaline karşın okulda yapılır. Okulun imkanlarına göre numaralı kağıt veya doğrudan okul bilgisayarlarında yapılabilir.

2.3.3) Değerlendirme sistemi ve ödüllendirme

Yapılan çözümler değerlendirilirken puan vermede en önemli kriter programın çalışıp çalışmadığını kontrol etmektir. Bundan sonra en önemli ikincisi de boşluk kullanımı ve düzendir. Öğretmenlerin önceden hazırladıkları belli sayıda girdi değerleri ile beklenen sonuç belirlenmelidir. Sonra öğretmen bütün çözümleri deneyerek hepsinin çıktılarının doğru olup olmadığını kontrol etmelidir bir eşitlik olması durumunda az kod kullanan avantajlı durumda olmalıdır. Bir problem örneği olarak ve örnek çözümü (şekil 6) :

2.3.3.1) Örnek problem ve çözümü

Problem : Belli bir miktarda borç çeken bir kredi kartının sahibinin yıllık faiz ile 1 sene boyunca aylık en az ne kadar ödemelidir ki borcu sene sonunda bitsin? Bisection arama metodu ve python kullanarak cevabı bulunuz.

Bilinmesi gerekenler:

Aylık faiz = Yıllık faiz / 12

Aylık ödeme alt limiti = Borç / 12

Aylık ödeme üst limiti = (Borç x (1 + Aylık faiz)^12) / 12

Virgülden sonra 2 basamak kullanınız Örn: 342,23

(Borç ve yıllık faiz oranı öğretmen tarafından belirlenecektir)

Şekil 6 : Örnek problem için doğru bir çözüm

```
balance = 320000
annualInterestRate = 0.2

bal = balance
for a in range(12):
    bal += bal*annualInterestRate/12
maxPay = bal / 12
minPay = balance / 12
x = (minPay + maxPay) / 2
def remBal(x):
    bala = balance
    for a in range(12):
        bala -= x
        bala += bala*annualInterestRate/12
    return(round(bala))
while abs(remBal(x)) > 0.01 :
    print(str(x))
    if remBal(x) < 0:
        maxPay = x
        x = (minPay + maxPay) / 2
    else:
        minPay = x
        x = (minPay + maxPay) / 2
print("Lowest Payment: " + str(round(x,2)))
```

2.3.3.2) Değerlendirme Sistemi

Yaptığımız değerlendirme sistemi ise şu şekildedir:

- 10 girdi denenir. Toplamda 80 puan olmak üzere her doğru çıktı 8 puandır.
- Doğru çıktılarının bir kısmının yada hepsinin gelmemesi durumunda öğretmen yapılan hataları inceler. Hatalı kısımların miktarına göre 0 ila 70 arasında öğretmen tarafından bir puan verilir. Bu miktar hatalı satırların doğru yazılmış satırlara oranına göre verilir.
- Programlama için gerekli nitelikler (Fonksiyon kullanımları, boşluk kullanımı, stil vb.) öğretmen tarafından karar verilir, 20 puandır

2.3.3.3) Ödüllendirme Sistemi

Dönem sonunda ve sene sonunda sıralamalar yapılır. Sıralamalar yapılırken ilk olarak aldıkları puana göre ikinci olarak ise kullandıkları kod satırı sayısı göz önünde bulundurulur. Dönem sonunda her okulda ilk % 5 lik kesime (4 adet 30 kişilik sınıfı olan bir okulda 6 kişiye) madalya ve belge verilir. Ayrıca en çok madalya alan sınıfa ayrıyeten bir de plaka asılır. Bu şekilde öğrenciler ve sınıflar arasında rekabet sağlanır. Belge örneği (Ek2)

Proje İş-Zaman Çizelgesi

İşin Tanımı	Kasım	Aralık	Ocak	Şubat
Literatür Taraması	X	X	X	
Arazi Çalışması (eğitim sistemi , müfredat belirlenmesi)		X	X	
Verilerin Toplanması ve Analizi			X	X
Proje Raporu Yazımı				X

Sonuç ve Tartışma

1. Yaptığımız “UYAKES Projesi Kapsamında Ülkemizde Yazılım ve Kodlama Eğitimi Hakkında Anket” adındaki anketten (**Ek1**) elde ettiğimiz sonuçlar şu şekilde özetlenebilir:

- Öncelikle katılımcılarımız büyük oranla (%87.1) hedefimiz olan kitle olan 10 – 19 yaş aralığındadır. Bu bize doğrudan sadece güncel bilgi seviyesini ölçerken netlik kazandırmamış, ayrıca gençlerin bu konuda olan ilgi ve alakasını ya da en azından ne kadar önemli olduğunu düşündüklerini bilme imkanı sağlamıştır.
- Bizim ve dünya üstündeki birçok ülkenin de düşündüğü gibi katılımcılarımız kodlama ve yazılım eğitimi için en uygun dönemi **%52.4** oranla ortaokul ve yüzde **%30.2** oranla ilkokul olarak görmüşlerdir. Bu bizim araştırmalarda elde etmiş olduğumuz kodlama eğitimi için en uygun yaşın küçük yaşlarda olduğu bilgisinin, katılımcılarda da mevcut ve farkında olduğunu göstermektedir. Bundan dolayı yapmış olduğumuz proje, en küçük yaşlarda (2. Sınıftan itibaren) başlaması gerektiğini ve günümüz sisteminin (5. sınıfta başlıyor ve sayısal düşünme gibi konuları öğretmede yetersiz) eksik olduğunu göstermiştir. Yetkililerin bu konuda derhal birşeyler yapması gerekmektedir yoksa milletimiz bilgisayar çağındaki dünyada gelecekte öğrenci potansiyeli konusunda geri kalacaktır.
- Katılımcılarımız her öğrencinin yazılım ve kodlama bilmesi gerektiği konusuna **%14.3** tamamen ve **%30.2** kısmen katılmıştır. Bu konuya tarafsız bakan veya kararsız olan kişilerin oranı **%34.9** ken , herkes değil sadece ilgisi veya yeteneği olanların öğrenmesi gerektiğine tamamen katılan kişilerin oranı **%6.3** kısmen katılan kişilerin oranı **%14.3** tür. Buradan çıkarılabilecek en önemli sonuç kıyasla büyük çoğunluk herkesin bilmesini isterken ciddi bir oran da bu konuda kararsızdır. Sadece toplamda **%20.6** lık bir oran belli kişilere öğretilmesi gerektiğini savunmuştur. Bu konuda bilincin sağlanması şarttır. Balakrishnan'ın da

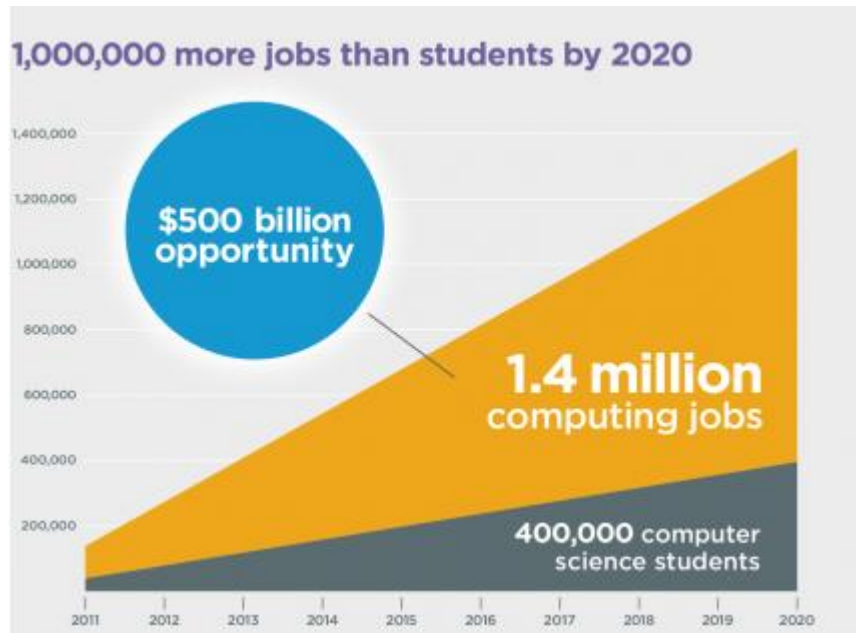
söylediği gibi [4] herkesin kodlama bilmesi toplumsal bir soruna daha çok zekice çözüm üretilmesini sağlar. Ve ayrıca bireylere çözüm bulma yetenekleri ve bilgisayar konusunda bağımsızlık sağlar. Gelişen ve teknolojinin her tarafta olduğu dünyamızda bu yüzden herkesin en azından fikir sahibi olması önemlidir. Projemizle bu sebepten dolayı sınav notları karneye işlememektedir ancak daha iyi yapanlar teşvik edilmektedir.

- Katılımcılarımızın %66.7 gibi büyük bir kısmı bilgisayar teledon gibi teknolojik aletlerde 3 saatten fazla vakit geçirmektedir. Bu vakitlerini ise genelde ne ile geçirdiklerinin oranları ise şu şekildedir. Sadece %1.6 oranında insan iletişim, %33.3 lük bir oran eğlence, %9.5 lik bir oran ise eğitim için kullanılmaktadır. %54 lük bir oranda insan hepsinden seçeneğini işaretlemiştir. Ayrıca başka bir kaynaktan [5] aldığımız verilere göre Türkiye'deki ortalama bilgisayar kullanma yaşı 8 dir. Türkiye'deki 6-15 yaş arası çocukların yüzde 60,5'i bilgisayar ; yüzde 50,8'i internet kullanmaktadır. Aynı zamanda bu yaş aralığındaki çocuklar interneti yüzde 79,5 oranında oyun oynamak ; yüzde 53,5 oranında sosyal medyada gezinmek için kullanmaktadır. Bu sanal ortamda bu kadar vakit geçmesine rağmen o sanal ortamın tasarlanması kullanılması için kullanılan sistemler hakkında bilgi oranı (sonraki maddede belirtilmiştir) çok vahim durumdadır. **Projemizi yapmamızın en temel sebebi budur.**
- Ezici yoğunluğun (%72.6) 16-19 yaşlarının oluşturduğu anketimizde nesne tabanlı programlama dilleri (C C++ C# Python Java..) hakkında bilgi seviyesi hakkında sorusunda çok iyi derecede bildiğini söyleyenlerin oranı %0 dır yani kimse bu derecede bilmemektedir. İyi derecede bilenlerin oranı %1.6 dır. Biraz bilen ve daha önceden küçük çaplı projeleri olan kişilerin oranı ise sadece %17.5 dir, geriye kalan %80.3 lük kesim ya çok az bilmektedir yada hiçbir fikri yoktur. Bu konuda ülkemizdeki liseler (özellikle devlet okulları) malesef çok geridedir. Yapılan başka araştırmalarda [6] Amerika'da 2014 yılında en az bir dili iyi derecede bilen öğrencilerin oranı %23 lere ulaşmaktadır. Ülkemizde bunun hakkında bilinç kısmen yerleşmiş olsa da yeterli gereksinimler sağlanmadığı için bilen kişilerin oranı çok düşüktür. Yetkili kişilerin bizimki gibi yazılım ve kodlamanın eğitimi konusundaki projeleri değerlendirip en kısa sürede nitelikli öğretmenler ve devlet tarafında okullara sağlana akıllı tahta daha tabletler aracılığıyla bu konuda yatırımlar yapması gerekmektedir.

2. Yazılım ve kodlamanın öğretilmesi için İngilizce eğitiminin de öğretilmesi önemlidir bizimki ve bizimki gibi projelerin sonucunda yeni Türkçe bir yazılım dili oluşana kadar öğrenmek için ingilizce kelimelerin öğretilmesi gerekecektir. Bu eğitim de diğer kodlama ve yazılım eğitimi gibi küçük yaşlardan itibaren verilmelidir. Soruvecevaplar.com sitesinde yapılan çalışmaya göre İngilizce eğitiminin kalıcı olması için bilinenden bilinmeyene doğru gidilmesi gerekmektedir. Özellikle Türkçe ve İngilizcede ortak olan sözcüklerden yararlanarak derse başlanması (Örneğin doctor, zebra, lemon vb.) ve çocukların sevdiği ve bildiği medya ve kültür öğelerine de öncelik vermek önemlidir. Ayrıca Üniteler/temalar arasında geçiş akıcı olmalıdır. Her ne kadar sırayla gidilecek olsa da bir sonraki ünite/tema önceki ünite/temada öğrenilenlerle ilişkilendirilmeli ve böylece birbirlerine entegre edilmelidir. Hazırladığımız yazılım ve kodlama eğitimiyle birlikte bu ingilizce dersleri de düzenlenmelidir.

3. Tabi böyle bir eğitimin verilmesi için en önemli unsurlardan biri de nitelikli öğretmen yetiştirilmesidir. Öğretmenlerin bu konu hakkında öğrencilerle iletişim halinde olması, takıldığı yerlerde onların sorularını cevaplaması ve dönem başı yapılacak belirlediğimiz sınavlara göre yine aynı şekilde belirlediğimiz değerlendirme sistemine göre değerlendirildikten sonra öğrencilere eksik konuları anlatmalıdır. Bu şartların yerine getirilmesi ülkemizde gelecekte bilgisayar konularında nitelikli öğrenciler elde etmek için çok önemlidir. Bilgisayar sektörü günümüzde bol iş olanağı yüksek maaşlar ödenen ve ülkenin gelişmesine doğrudan etkileyen bir sektördür. Öyle ki yapılan araştırmaya [7] göre 2020 yılında bilgisayar ile ilgili işler nitelikli öğrenci sayısını 1 milyon geçmektedir. Toplam işten para imkanı(şekil 7) ise 500 milyar dolardır.

Şekil 7 : Yapılan “Bilgisayar bilimleri eğitimi için artan talep” adındaki araştırmaya [7] göre bilgisayar hakkında işler ve nitelikli öğrenci analizi.



Öneriler

Uyakes isimli projemizde ülkemize yazılım ve kodlama eğitiminin eğitim sistemine nasıl entegre edilebileceğini araştırdık. Bunun yapılabilmesi için eğlenceli ama öğretici bir sistemin çok önemli olduğunu saptadık. Bundan dolayı gençlerimizde ilgi ve merak uyandırıp bu temel bilgilerin gençliğe benimsetilmesi ülkemizin geleceği için azami derecede önemlidir. Bunun gibi projeler oluştururken erişilebilirlik, çekicilik ve uygulanabilirlik gibi değerlere dikkat etmek gerekir çünkü ilgi uyandırmayacak bir ders öğrencilerde kalıcı olmaz. Bu tarz projelerde toplumsal yapı ve öğrenci profilini iyi belirlemek oldukça önemlidir.

Kaynaklar

- [1] Kentwartz, N. 3 Reasons to start coding young. Erişim: 17 Ocak 2021, <https://bitsbox.com/blog/three-reasons-to-start-coding-young/>
- [2] Wing, J. M. (2006), Computational Thinking , Erişim tarihi 1 Şubat 2021 : <http://www.cs.cmu.edu/~15110-s13/Wing06-ct.pdf>

- [3] MEB, (2018) BİLİŞİM TEKNOLOJİLERİ VE YAZILIM DERSİ ÖĞRETİM PROGRAMI (Ortaokul 5 ve 6. Sınıflar), Erişim: 12 Ocak 2021, <http://mufredat.meb.gov.tr/Dosyalar/2018124103559587-Bili%C5%9Fim%20Teknolojileri%20ve%20Yaz%C4%B1%C4%B1m%205-6.%20S%C4%B1n%C4%B1flar.pdf>
- [4] Balakrishnan, A. (2018), Fascinating Debates: Should everyone learn to code?, Erişim tarihi : 20 Ocak 2021, <https://akbgunner.medium.com/fascinating-debates-should-everyone-learn-to-code-4354fba30bbf>
- [5] Mengi, Z. (2017, 9 Haziran). Her 10 gençten 1'i bilgisayar bağımlısı. Hürriyet. Erişim adresi: <https://www.hurriyet.com.tr/egitim/her-10-gencten-1-i-bilgisayar-bagimlisi-25509833>
- [6] Devaney, L. (2014) , Students: We need coding skills, Erişim tarihi: 21 Ocak 2021, <https://www.eschoolnews.com/2014/08/25/students-coding-skills-934/>
- [7] Soper, T. (2014), Analysis: The exploding demand for computer science education, and why America needs to keep up, Erişim tarihi : 16 Ocak 2021, <https://www.geekwire.com/2014/analysis-examining-computer-science-education-explosion/>
- Uzgur, B. Ç. ve Aykaç, N. (2016). Bilişim Teknolojileri Ve Yazılım Dersi Öğretim Programının Öğretmen Görüşlerine Göre Değerlendirilmesi. Erişim: 15 Ocak 2021, <https://dergipark.org.tr/en/download/article-file/226453>
- MEB, (2018) Bilişim Teknolojileri Ve Yazılım Dersi Öğretim Programı (Ortaokul ve İmam Hatip Ortaokulu 7 ve 8. Sınıflar), Erişim: 12 Ocak 2021, [https://mufredat.meb.gov.tr/Dosyalar/2018813171426130-2-2018-81Bili%C5%9Fim%20Teknolojileri%20ve%20Yaz%C4%B1%C4%B1m%20Dersi%20\(7%20ve%208.%20S%C4%B1n%C4%B1flar\).pdf](https://mufredat.meb.gov.tr/Dosyalar/2018813171426130-2-2018-81Bili%C5%9Fim%20Teknolojileri%20ve%20Yaz%C4%B1%C4%B1m%20Dersi%20(7%20ve%208.%20S%C4%B1n%C4%B1flar).pdf)
- RoboGarden, Top 5 Countries in Coding and Programming, Erişim: 5 Ocak 2021, <https://robogarden.ca/blog/top-5-countries-in-coding-and-programming>
- Bonfiglio, C. (2018), 8 Reasons why every child should learn to code, Erişim: 27 Aralık 2020, <https://teachyourkidscode.com/why-coding-is-important-to-learn/>

- Admin,(2019), CSS AÇIKLAMA (YORUM) SATIRI NEDİR? NASIL KULLANILIR?. Erişim tarihi: <http://www.tasarimveyazilim.com/css-aciklama-yorum-satiri-nedir-nasil-kullanilir/>
- Küçük, T. T. (2020) , Yazılımcı Sözlüğü, Erişim tarihi : 20 Ocak 2020 <https://medium.com/5bayt/yaz%C4%B1l%C4%B1mc%C4%B1-s%C3%B6zl%C3%BC%C4%9F%C3%BC-8879f99ae04a>
- Nuhuz, Y. (2016) Programlama Terimleri, Erişim tarihi 25 Ocak 2020, <https://caylakyazilimci.com/post/programlama-terimleri>
- Ak, Y. E. (2020), Programlamada bilmen gereken terimler ve mimari yaklaşımlar, Erişim tarihi: 10 Ocak 2021: <https://lib.yemreak.com/programlama/temel/terimler>
- Wong, G. (2015), School perceptions of coding education in K-12: A large scale quantitative study to inform innovative practices, Erişim tarihi : 12 Ocak 2021, https://www.researchgate.net/publication/304295837_School_perceptions_of_coding_education_in_K-12_A_large_scale_quantitative_study_to_inform_innovative_practices
- Lee, I. (2011), Computational Thinking for Youth in Practice, Erişim tarihi : 20 Ocak 2021, https://www.researchgate.net/publication/234810765_Computational_thinking_for_youth_in_practice
- “Soru Ve Cevaplar” isimli yazar, (2020) , İngilizce Öğretim Programının Uygulanmasına Dair Önemli Hususlar , Erişim tarihi: 16 Aralık 2020, <https://soruvecevaplar.com/ingilizce-ogretim-programinin-uygulanmasi/>

Ekler

EK 1:” UYAKES Projesi Kapsamında Ülkemizde Yazılım ve Kodlama Eğitimi Hakkında Anket” sorularına verilen yanıtların oranları:

Açıklama: Bu gönüllü anket formu, bir proje adına yürütülen bir araştırma için hazırlanmıştır. Çalışma ülkemizdeki eğitim sisteminin yeterli olup olmadığını belirleme amacındadır. Bilimsel nitelik taşıyan bu araştırmanın, idari veya siyasi herhangi bir yönü yoktur. Lütfen anket formuna adınızı, soyadınızı veya kimliğinizi belirten hiçbir şey yazmayınız. Sorulara eksiksiz, gerçekçi ve içtenlikle cevap vermeniz, bu araştırmanın amacına ulaşmasına katkıda bulunacaktır.

Sorulara vereceğiniz cevaplarla yapacağınız değerli yardım ve katkılarınız için şimdiden

teşekkür ederiz. Kişisel bilgileriniz gizli tutulacaktır. Saygılarımızla

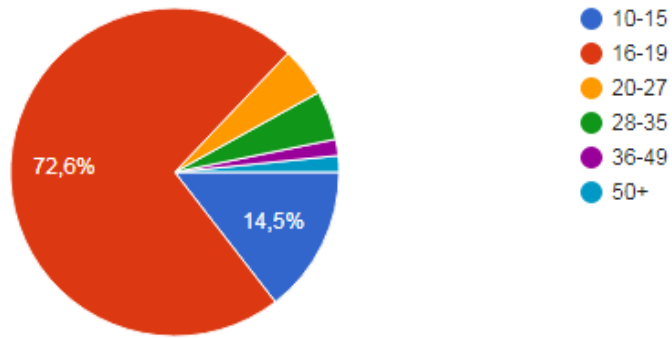
1.) Soru

İlk soruda katılımların proje kapsamında bilgilerinin kullanımı konusunda onayı alınmıştır.

2.) Soru

Yaş aralığınız nedir. Veriler sadece araştırma amaçlı kullanılacaktır.

62 yanıt

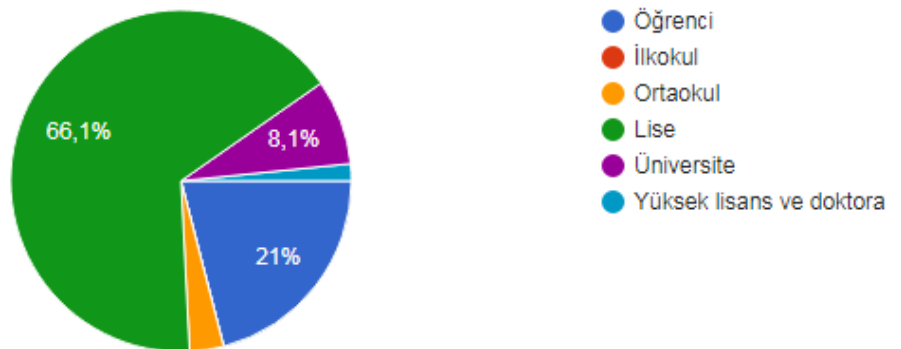


Not :20-27 ve 28-35 yaş aralıkları oranları %4.8 dir

3.) Soru

Eğitim seviyeniz nedir ?

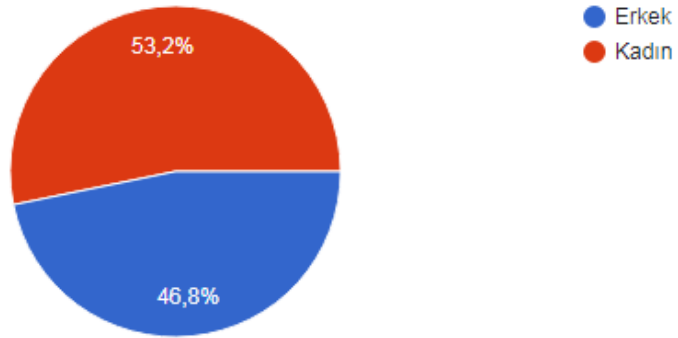
62 yanıt



4.) Soru

Cinsiyetiniz nedir. Veriler sadece araştırma amaçlı kullanılacaktır.

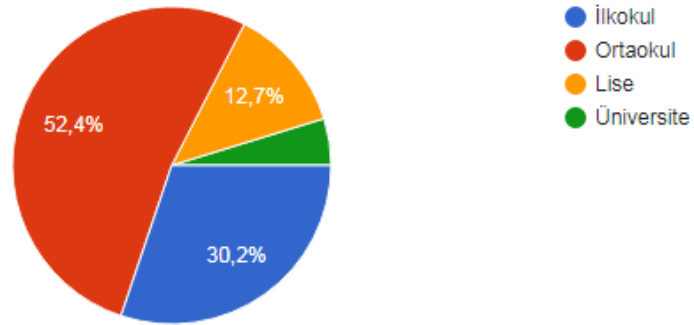
62 yanıt



5.) Soru

Sizce kodlama eğitimi için en uygun yaş aralığını nedir? (itibaren)

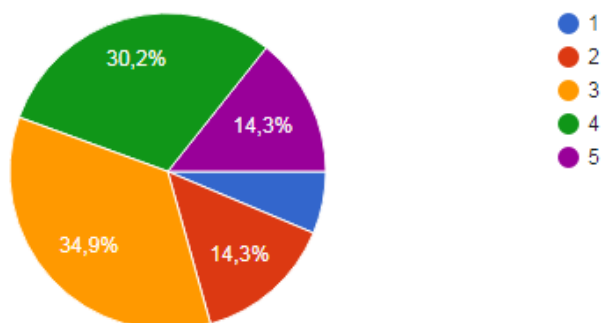
63 yanıt



6.) Soru

Bu cümleye ne kadar katılıyorsunuz "Herkes belli bir seviyede yazılım ve kodlama bilmelidir." 5 üzerinden

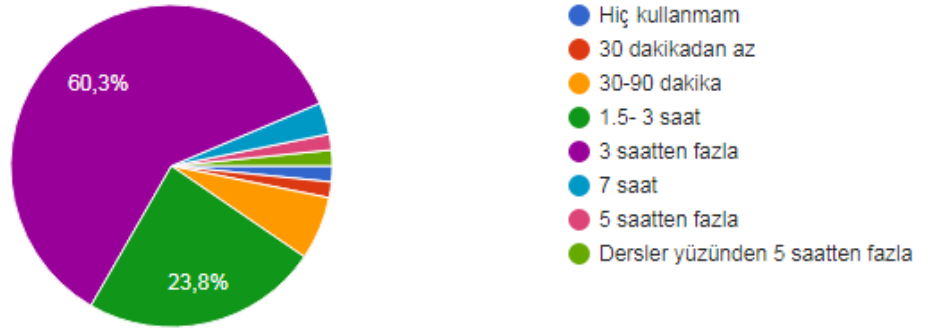
63 yanıt



7.) Soru

Bilgisayar Telefon gibi teknolojik aletlerle bir günde ortalama ne kadar vakit harcıyorsunuz ?

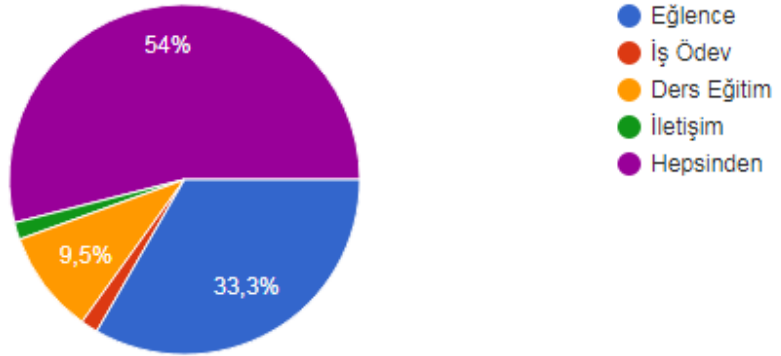
63 yanıt



8.) Soru

Genelde sanal ortamlarda ne için vakit geçirirsiniz

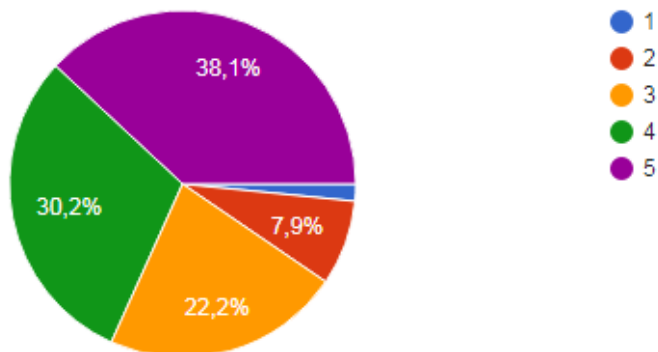
63 yanıt



9.) Soru

Bu cümleye ne kadar katılıyorsunuz "Yazılım bilmek önemlidir" 5 üzerinden

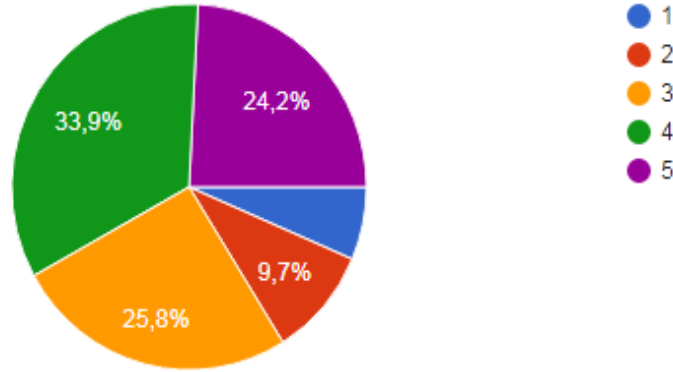
63 yanıt



10.) Soru

Bu cümleye ne kadar katılıyorsunuz "Bilgisayarlar eğitim öğretim sürecinin en önemli unsurlarından biridir" 5 üzerinden

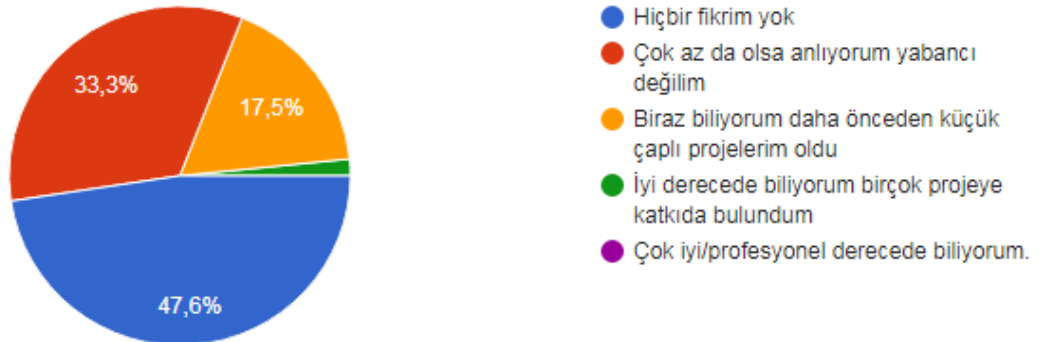
62 yanıt



11.) Soru

Nesne tabanlı programlama dilleri (C C++ C# Python Java..) hakkında bilgi seviyeniz nedir

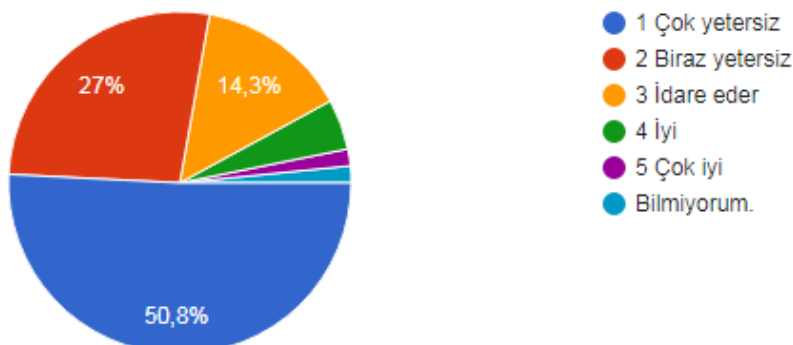
63 yanıt



12.) Soru

Günümüzde ülkemizde bilişim teknolojileri eğitimini 5 üzerinden puanlasaydınız kaç puan verirdiniz?

63 yanıt



13.) Soru

Web tasarımı uygulama oluşturma yada hazır kod ile program oluşturma konusunda neler öğrendiniz.

63 yanıt



14.) Soru

14. soruda “Ülkemizde yazılım ve kodlama eğitiminin kalıcı ve eğlenceli olması sizce neye bağlıdır ?” sorusu sorulmuştur. Gelen başlıca bazı cevaplar şu şekildedir:

- “Küçük yaşlarda eğitime başlanmalı, ders olarak verilip sonunda zorunlu bi sınav yapılmazsa rahat ve kalıcı bir eğitim olacağını düşünüyorum.”
- “Öğretim biçimine dogmatik ve ezberci bir eğitimdense uygulama tabanlı bir eğitim biçimi olmalı
- Yazılımın eğlenceli ve herkese hitap eder olmasını bir gereklilik olarak görmüyorum. "Teknoloji Üretmek" genellikle ziyade özellik, sıkı bir tedrisat liyakat gerektiren bir iştir fikrimce.
- Bence esas olan, eğitimin ilgi alanlarına bakılarak titizlikle verilmesi ve belirli aralıklarla teşvik edici etkinliklerin yapılmasıdır.
- “Bence sınavsız seçmeli ders şeklinde olmalıdır çünkü işin içine sınav girince derse çok ilgisi olmayanlar sadece sınava kadar çalışıp öğrendiklerini aklında tutacak. Ayrıca belli bir yaş grubunun öğrenmek ya da yapmak zorunda olduğu şeyleri sevsen (yapacak olsa) bile soğumaya (yapmak istememeye) başladığını düşünüyorum. Özetle gençler, yazılım ve kodlamaya yönlendirilmeli ancak gönüllülük esas olmalı çünkü zorunluluğun kalıcılığı ve eğlenceyi engellediğini düşünüyorum.”

Bu değerli cevaplar projemizin temel prensiplerine uymaktadır ve birçok insanın da aynı şekilde düşündüğünü göstermektedir. Hedefimiz bu projeye ülkemizde geleceğe daha hazırlıklı ve bilinçli bireyler yetiştirilmesine yardımcı olmaktır.

EK 2: Örnek başarı belgesi

Yazılım ve Kodlama Başarı Belgesi	
Öğrencinin	Belge Numarası:
T. C. Kimlik Numarası:	Tarihi:
Sınıfı-Numarası:	
Ders Yılı-Dönemi:	
Yukarıda kimliği yazılı yazılım ve kodlama derslerinde gösterdiği gayreti ile yeteneğinden dolayı bu belgeyi almaya hak kazanmıştır.	
OKUL MÜDÜRÜ	

EK3: Örnek Yazılım ve Kodlama Ders Notu

PYTHON

4 Temel Tür

İnt(integer yani tamsayı) → örnek: 1 , 13 , 1056

Float(ondalıklı sayı) → örnek: 1.5 , 15.0 , 156.3

Str(string yani yazı) → örnek: "Bu bir yazıdır" , 'Bu da bir yazıdır' (tek veya çift tırnak ifadeyi string türüne çevirir.)

True - False → Bu değerlere Bool denir. True doğru anlamına gelirken False yanlış anlamına gelir.

Yorum Satırı

```
1 ''' Kod yazarken not almaya ya da bir kodun çalışmasını önlemeye olanak
   veren aralıklara denir. Yorum satırları tarayıcılar tarafından okunmaz'''
```

-Bu kutucuklar kod satırlarını temsilen oluşturulmuştur.

Yorum Satırı Oluşturma: Bu işlem için 3 basit yol vardır ;

- 1) # işareti ile bu işaretten sonraki satırları yorum satırına dönüştürebilirsiniz.
-Yorum satırından çıkarmak için # işaretini silmeniz gerekir.
- 2) Metinleri üç tırnak('') içine alarak yorum satırına dönüştürebilirsiniz.
-Yorum satırından çıkarmak için üç tırnakları silmeniz gerekir.
- 3) Yorum satırı yapmak istediğiniz bölümü imleç ile seçerek Ctrl+K+C tuşlayabilirsiniz.
-Yorum satırından çıkarmak için Ctrl+K+U tuşlamanız gerekir.

İşlemler ve İşaretleri

Çıkarma işareti → -

Toplama işareti → +

Çarpma işareti → *

Bölme işareti → /

Kalansız bölme işareti → //

Üssünü alma işareti → **

Mod işareti → %

```
1 say11 = 2
2 say12 = 3
3 print(number1+number2)
4 #print girilen değeri yazdırmayı sağlar.
```

SONUÇ:

5

Proje raporu 20 sayfayı geçeceğinden buraya örnek bir sayfası konulan “5. Sınıf Yazılım ve Kodlama Haftalık Ders Müfredatı Tablosu” adlı EK4, ekler kısmına eklenmiştir.