

C# Intro

Konsol ile Algoritma ve Yapısal Programlama Örnekleri

NBUY Alfa Yazılım Ekibi
BİLGEADAM

9 UYGULAMALAR

9.1 SAYI TAHMİN OYUNU

Oyunun 3 zorluk seviyesi vardır. Zorluk seviyeleri oyunun açılışında kullanıcıya sunulur ve kullanıcı 1, 2 veya 3 olarak ifade edilen zorluk seviyesini seçer.

Lütfen aşağıdaki zorluk seviyelerinden birisini seçiniz:

1. Kolay seviye için 1'i tuşlayın
2. Orta seviye için 2'i tuşlayın
3. Zor seviye için 3'i tuşlayın

Seçilen zorluk seviyesine göre tahmin edilecek sayının üretilme aralığı değişkenlik gösterir.

- 1: 1 - 10 arasında
- 2: 1 - 100 arasında
- 3: 1 - 1000 arasında

Eğer ki kullanıcı yanlış bir giriş yaparsa program kullanıcıya uyarı verir ve zorluk seviyesi tekrar sorulur:

A

Yanlış bir giriş yaptınız!

Lütfen aşağıdaki zorluk seviyelerinden birisini seçiniz:

1. Kolay seviye için 1'i tuşlayın
2. Orta seviye için 2'i tuşlayın
3. Zor seviye için 3'i tuşlayın

Doğru seçenek işaretlendikten sonra oyun başlar ve kullanıcının tahmin yapması beklenir.

Tahmininizi yazınız: 18

Kullanıcı sayı tahmin eder. Yapılan tahmine göre kullanıcıya bir dönüş sağlanır. Çok yakın tahminlerde “sıcak”, orta yakınlıktaki tahminlerde “ılık”, uzak tahminlerde ise kullanıcıya “soğuk” kelimesi ile dönüş yapılır.

Sayı doğru tahmin edilirse ise oyun sonlanır, yanlış ise oyun devam eder ve kullanıcıdan yeni sayı girmesi beklenir.

Her tahmin adımı bir değişkende sayılır. Oyun yanlış tahminlerden dolayı sonlanmaz, ancak doğru tahmin edildiğinde oyun son bulur. Ancak oyun sonunda oyuncunun doğru tahmini kaç adımda yaptığı ekrana yansıtılır.

Tahmininizi yazınız: 64

Tebrikler!! 64 doğru tahmin!
Oyunu 6 tahminde bitirdiniz.

Kullanıcının sonsuz deneme hakkı vardır.

BONUS: Kullanıcı ne kadar kısa denemede doğru tahmin ederse, 1000'lik sistemde o kadar çok puan alır. Tahmin ne kadar uzun sürerse süsün kullanıcı oyunu asla 0 puanla tamamlamamalıdır.

SÜPER BONUS: Oyun tamamlandığında kullanıcıya tekrar oynayıp oynamayacağı sorulur E/H

9.2 SAYI ÇÖZÜMLEME

Bir tamsayı alan ve tamsayı basamaklarını çözümleyen bir program yazınız.

Girilen ifadenin doğru bir tamsayı olup olmadığı kontrol edilmelidir. Eğer ki kullanıcının giriş yaptığı değer bir tamsayı değilse, kullanıcıdan doğru formatta bir sayı istenmelidir. Ayrıca yapılacak işlemler kesinlikle tamsayı tipindeki değişkenler kullanılarak yapılmalıdır. Bu örnekte string metotları kullanılmayacaktır.

Programı sonlandırmak için sayı giriş aşamasında “exit” yazılması gereklidir.

2837 sayısı için sonuç aşağıdaki gibi görünmelidir:

```
Lütfen bir sayı giriniz: 2837
*****

7 x 1 = 7
3 x 10 = 30
8 x 100 = 800
2 x 1000 = 2000

Lütfen bir sayı giriniz: asd
Doğru formatta bir sayı girmediniz!
*****

Lütfen bir sayı giriniz: 94
*****

4 x 1 = 4
9 x 10 = 90

Lütfen bir sayı giriniz: exit
```

9.3 PALİNDROME

Tersten ve düzden okunduğunda aynı okunan cümlelere, sözcüklere veya sayılara palindrom denir. Örneğin İLAÇ İÇ ALİ cümlesi, AĞA, NAZAN, KÜÇÜK sözcükleri ve 363, 2442, 98789 sayıları birer palindromdur. Kullanıcının ekrandan gireceği tam sayının bir palindrom olup olmadığını kontrol eden ve sonucu ekrana yazdıran bir program yazınız.

Girilen ifadenin doğru bir tamsayı olup olmadığı kontrol edilmelidir. Eğer ki kullanıcının giriş yaptığı değer bir tamsayı değilse, kullanıcıdan doğru formatta bir sayı istenmelidir. Ayrıca yapılacak işlemler kesinlikle tamsayı tipindeki değişkenler kullanılarak yapılmalıdır. Bu örnekte string metotları kullanılmayacaktır.

Programı sonlandırmak için sayı giriş aşamasında “exit” yazılması gereklidir.

Program örnek çıktısı aşağıdaki gibidir:

```
Bir tamsayı giriniz: yirmibeş
Lütfen doğru formatta bir sayı yazınız!

Bir tamsayı giriniz: 25
Girdiğiniz sayı bir palindrom değildir.

Bir tamsayı giriniz: 545
Girdiğiniz sayı bir palindromdur.

Bir tamsayı giriniz: 12345
Girdiğiniz sayı bir palindrom değildir.

Bir tamsayı giriniz: 9449
Girdiğiniz sayı bir palindromdur.

Bir tamsayı giriniz: exit
```

9.4 OTOPARK ÜCRET HESAPLAMA

Bu programda bir otoparkın ücretlendirme sistemi geliştirilecektir. Otopark ücretleri araç tipine göre değişkenlik göstermektedir.

Program ilk açıldığında ekranda kullanıcıya araç tipleri seçeneği gelmelidir:

```
1. Otomobil için 1'e basın
2. Motosiklet için 2'ye basın
3. Minibüs için 3'e basın
4. Kamyon (ve diğer ticari araçlar) için 4'e basın
```

Kullanıcı bu seçeneklerden birisini seçmelidir. Eğer ki yanlış bir giriş yaptıysa program bunu dikkate almayacak ve seçenekler ekranda kalacaktır.

Araç tipi seçimi yapıldıktan sonra kullanıcıya saat bilgisi sorulmalıdır.

Kullanıcı saat bilgisini ondalık sayı tipinde girecektir. Eğer ki geçersiz bir giriş yapıldıysa sistem kullanıcıya uyarı döndürür ve saat bilgisi yeniden sorulur.

```
Araç kaç saat boyunca park alanında kaldı? Dört
Lütfen geçerli bir sayı yazınız!
Araç kaç saat boyunca park alanında kaldı? 4
```

Araçın park yerini işgal süresi girildikten sonra hesaplama yapılacaktır. Araçların park yeri tarifesi aşağıdaki gibidir:

- Otomobil
 - 0 – 2 saat: 5TL
 - 2 – 6 saat: 10TL
 - 6 – 12 saat: 20TL
 - 12 – 24 saat: 35TL
 - 24 saat sonrası her ilave gün için (saate bakılmaz): +20TL
- Motosiklet
 - 0 – 2 saat: Ücretsiz
 - 2 – 6 saat: 3TL
 - 6 – 12 saat: 6TL
 - 12 – 24 saat: 12TL
 - 12 – 24 saat: 20TL
 - 24 saat sonrası her ilave gün için (saate bakılmaz): +10TL
- Minibüs
 - 0 – 2 saat: 8TL
 - 2 – 6 saat: 16TL
 - 6 – 12 saat: 32TL
 - 12 – 24 saat: 45TL
 - 24 saat sonrası, her ilave gün için (saate bakılmaz): +25TL
- Kamyon ve ticari araç
 - 0 – 2 saat: 15TL
 - 2 – 6 saat: 30TL

- 6 – 12 saat: 60TL
- 12 – 24 saat: 100TL
- 24 saat sonrası, her ilave gün için (saate bakılmaz): +55TL

Girilen değere göre ücret hesaplaması yaptırılır ve ekrana yazdırılır

Ödenmesi gereken tutar 10TL

Eğer kullanıcı sınırdaki bir tam sayı değeri girerse, ücret alt tarifieden hesaplanacaktır. Örneğin 2 saatlik otopark ücreti 0 – 2 saat aralığındaki tarife göre hesaplanacaktır. Ancak ondalık sayı tipindeki bir değer için bir üst tarife dahil olur. Örneğin 2.5 saatlik otopark ücreti 2 – 6 saatlik otopark ücret tarifi üzerinden hesaplanır.

9.5 HAVAYOLLARI REZERVASYON SİSTEMİ

Küçük bir havayolu şirketi rezervasyon kayıtlarını tutmak için bir program yazmanızı istemiştir. Havayolu şirketinin sahip olduğu uçaklar küçük boyutlarda ve toplamda 20 koltuk kapasitelidir. 20 koltuktan 8 tanesi Business Class, 12 tanesi Economy Class koltuklardır.

Rezervasyonlar sınıf, koltuk numarası ve kişi ismi alınarak yapılmaktadır. Program açıldığında kullanıcıya iki seçenek sunulmalıdır:

1. Business Class bölümü için 1 tuşuna basın
2. Economy Class bölümü için 2 tuşuna basın

Kullanıcı bu iki seçenekten birisini seçmelidir. Eğer ki yanlış bir giriş yaptıysa program bunu dikkate almayacak ve seçenekler ekranda kalacaktır.

Sınıf seçimi yapıldıktan sonra kullanıcıya boş koltuk numaraları listelenecektir. Örneğin Economy Class seçimi yapıldıktan sonra:

```
Economy Class bölümünde kalan boş koltuklar:  
- 9  
- 12  
- 13  
- 19  
- 20
```

Kullanıcı boştaki koltuk numaralarından birisini ekrana yazar ve Enter tuşuna basar. Eğer kullanıcı boş olmayan bir koltuk seçtiyse, ilgili koltuk kimin adına rezerve edildiye isimle birlikte uyarı verilmelidir:

```
Seçilen koltuk numarası: 11  
11 Numaralı koltuğu daha önce "Tsubasa Ozora" isimli yolcuya rezerve ettiniz!  
Lütfen boş koltuklardan birisini seçiniz.  
  
Economy Class bölümünde kalan boş koltuklar:  
- 9  
- 12  
- 13  
- 19  
- 20
```

Kullanıcı uygun koltuklardan birisini seçtikten sonra yolcu adı ve soyadı istenecektir:

```
Seçilen koltuk numarası: 13  
Lütfen yolcunun Adı ve Soyadını yazın: Misaki Taro
```

İsim yazılıp Enter tuşuna basıldıktan sonra seçilen koltuk kişiye rezerve edilecek ve ekrana sonuç yazdırılacaktır:

```
Economy Class bölümündeki 13 Numaralı koltuğu Misaki Taro isimli yolcuya rezerve ettiniz.  
Devam etmek için bir tuşa basın...
```

Kullanıcı bir tuşa bastığında program başa döner ve ilk ekrana gelen sınıf seçenekleri görüntülenir.

Eğer ki sınıf seçildikten sonra o sınıfta boş koltuk kalmadıysa kullanıcıya şu uyarı döndürülmelidir:

```
Seçtiğiniz Business Class bölümünde boş koltuk kalmamıştır.  
Economy Class bölümündeki boş koltukları görmek ister misiniz? E/H
```

Kullanıcı E seçeneğini seçerse, sınıf seçimi işleminden sonraki akış aynen uygulanacaktır. Eğer kullanıcı H bölümünü seçerse ekrana şu uyarı döndürülecektir:

```
Bir sonraki uçuş kayıtları 4 saat sonradır.
```

Bir sonraki uçuş uyarısı ekrana getirildikten sonra program ilk duruma döner ve sınıf seçeneklerini ekrana getirir.

Örnek, tek boyutlu string tipinde değerler alan bir dizi ile çözülmelidir. Koltuk numaraları ile index değerleri uygun şekilde eşleştirilerek program geliştirilmelidir.

9.6 ŞİFRE GİZLEME

Bu uygulamada konsol üzerinde yazılmış bir projenin kullanıcı giriş ekranı tekrar geliştirilecektir. Konsol ekranına yazılan her şey görünür olduğu için kullanıcılar yazılıma olan güvenini kaybetmiştir. Giriş yaparken görünen ekran aşağıdaki gibidir:

```
////////////////\////////////////
||  Lorem Ipsum CRM System  ||
\\////////////////////////
Lütfen kullanıcı adınızı ve parolanızı giriniz.
Kullanıcı Adı: Tsubasa
Parola: N@nkat5u123
```

Giriş aşamasında yazılan şifreler konsol ekranında açık olarak görüldüğü için kullanıcılar yazılımı kullanmama kararı almıştır. Müşteri firma yetkilileri ile yapılan görüşmeler sonrasında Proje Yöneticisi sizden giriş ekranının aşağıdaki gibi tekrar geliştirilmesini istemiştir:

```
////////////////\////////////////
||  Lorem Ipsum CRM System  ||
\\////////////////////////
Lütfen kullanıcı adınızı ve parolanızı giriniz.
Kullanıcı Adı: Tsubasa
Parola: *****
```

Örnekte görüldüğü gibi kullanıcı giriş ekranında yazılan şifreler, klavyeden her bir karaktere basıldığında ekrana o karakter yerine * karakteri yazılarak gizli hale getirilecektir.

Girilen şifre sadece ekranda * karakterleri ile değiştirilmemeli, aynı zamanda arka planda enter tuşuna basılana kadar bekletilmelidir. Yazılımı test etmek için girilen şifreyi enter tuşundan sonra ekrana yansıtınız.

```
////////////////\////////////////
||  Lorem Ipsum CRM System  ||
\\////////////////////////
Lütfen kullanıcı adınızı ve parolanızı giriniz.
Kullanıcı Adı: Tsubasa
Parola: *****
-----
PAROLA TEST: N@nkat5u123
```