

## 1- Değişkenler

- a- Ekrandan girilen değerin string bir değişkene atanması
- b- Ekrandan girilen değerin tamsayı tipinde bir değişkene atanması
- c- Ekrandan girilen değerin ondalık sayı tipinde değişkene atanması
- d- Ekrandan girilen değerin tarih tipinde bir değişkene atanması
- e- long tipinde bir sayının int tipinde bir değişkene atanması.
- f- İnt tipinde bir sayının string tipinde bir değişkene atanması

## 2- Temel Arimetik İşlemler

- a- Ekrandan okunan iki tamsayının toplamını ekrana yazınız.
- b- Ekrandan okunan iki tamsayının farkını ekrana yazınız.
- c- Ekrandan okunan iki tamsayının çarpımını ekrana yazınız.
- d- Ekrandan okunan iki tamsayının bölümünü ekrana yazınız.
- e- Ekrandan okunan iki ondalıklı sayı üzerinde abcd maddelerinde belirtilen 4 işlemin yapılması.
- f- Ekrandan okunan bir tamsayı ve bir ondalıklı sayı arasından abcd maddelerinde belirtilen 4 işlemin yapılması.

## 3- Karşılaştırma(Metot kurmak için kendinizi zorlamaya başlayın)

- a- Ekrandan alınan iki tamsayının karşılaştırılması ve eşitlik durumunun ekrana yazılması. ("eşit" yada "eşit değil" yazmalı)
- b- Kullanıcıdan alınan iki string tipindeki verinin eşitlik durumunun ekrana yazılması.
- c- Kullanıcıdan alınan iki karakter tipindeki verinin eşitlik durumunun ekrana yazılması.

## 4- Mantıksal İşlemler(Metot kurmak için kendinizi zorlamaya başlayın)

### a- Uygulama 1 – Sayı karşılaştırma:

Kullanıcıdan alınan 3 adet sayının karşılaştırılarak sonucun bool tipinde bir değişkene atanması işleminin yapılması. Bool değişkeni işlem sonunda ekranda gösterilmelidir. Programda 45, 3, 100 sayılarının giriş yapıldığını düşünürsek console ekranındaki çıktı alttaki gibi olmalıdır:

```
01. 45, 3 ve 100 sayıları birbirine eşittir: false
02. 45, 3 ve 100 sayıları birbirine eşit değildir: true
03. 45 sayısı 3 ve 100 sayılarından büyüktür: false
04. 45 sayısı 3 ve 100 sayılarından büyüktür veya sayılara eşittir: false
05. 45 sayısı 3 ve 100 sayılarından küçüktür: false
06. 45 sayısı 3 ve 100 sayılarından küçüktür veya sayılara eşittir: false
07. 3 sayısı 45 ve 100 sayılarından büyüktür: false
08. 3 sayısı 45 ve 100 sayılarından büyüktür veya sayılara eşittir: false
09. 3 sayısı 45 ve 100 sayılarından küçüktür: true
10. 3 sayısı 45 ve 100 sayılarından küçüktür veya sayılara eşittir: true
11. 100 sayısı 45 ve 3 sayılarından büyüktür: true
12. 100 sayısı 45 ve 3 sayılarından büyüktür veya sayılara eşittir: true
13. 100 sayısı 45 ve 3 sayılarından küçüktür: false
14. 100 sayısı 45 ve 3 sayılarından küçüktür veya sayılara eşittir: false
```

#### b- Uygulama 2 - Yaş Karşılaştırma:

Ekrandan giriş yapılacak üç isim ve yaşlarının karşılaştırması yapılacak. Yapılan karşılaştırmalar bool tipinde değişkenlere atanacak ve sonuçlar ekrana yazdırılacak. Programda Osman 17, Emre 15 ve Alper 21 değerlerinin giriş yapıldığını düşünürsek çıktı alttaki gibi olmalıdır.

```
01. Osman, Emre ve Alper isimli kişilerin yaşları birbirine eşittir: false
02. Osman, Emre ve Alper isimli kişilerin yaşları birbirine eşit değildir: true
03. Osman isimli kişinin yaşı Emre ve Alper isimli kişilerin yaşlarından büyüktür: false
04. Osman isimli kişinin yaşı Emre ve Alper isimli kişilerin yaşlarından büyüktür veya yaşlarına eşittir: false
05. Osman isimli kişinin yaşı Emre ve Alper isimli kişilerin yaşlarından küçüktür: false
06. Osman isimli kişinin yaşı Emre ve Alper isimli kişilerin yaşlarından küçüktür veya yaşlarına eşittir: false
07. Emre isimli kişinin yaşı Osman ve Alper isimli kişilerin yaşlarından büyüktür: false
08. Emre isimli kişinin yaşı Osman ve Alper isimli kişilerin yaşlarından büyüktür veya yaşlarına eşittir: false
09. Emre isimli kişinin yaşı Osman ve Alper isimli kişilerin yaşlarından küçüktür: true
10. Emre isimli kişinin yaşı Osman ve Alper isimli kişilerin yaşlarından küçüktür veya yaşlarına eşittir: true
11. Alper isimli kişinin yaşı Osman ve Emre isimli kişilerin yaşlarından büyüktür: true
12. Alper isimli kişinin yaşı Osman ve Emre isimli kişilerin yaşlarından büyüktür veya yaşlarına eşittir: true
13. Alper isimli kişinin yaşı Osman ve Emre isimli kişilerin yaşlarından küçüktür: false
14. Alper isimli kişinin yaşı Osman ve Emre isimli kişilerin yaşlarından küçüktür veya yaşlarına eşittir: false
```

## 5- Karar yapıları

#### a- Kullanıcı adı ve parola karşılaştırma:

Kullanıcıdan sırayla alınacak “Kullanıcı adı” ve “Şifre” değerleri birer string değişkene atanır. If-Else yapısı içerisinde girilen değerler olması gereken değerler ile karşılaştırılıp sonuç ekrana yazdırılacak.

#### b- Büyük/küçük sayı tespit etme:

Ekrandan girilecek 3 adet sayıdan en büyüğünü ve/veya en küçüğünü tespit eden program yazın. Sonucu ekrana yazdırın.

#### c- Tek mi Çift mi?:

Ekrandan girilecek 1 tam sayının tek sayı mı yoksa çift sayı mı olduğunu tespit eden ve sonucu ekrana yazdıran bir program geliştiriniz.

**d- Tam katı tespit etme:**

Ekrandan girilecek 2 tam sayıdan, büyük sayının küçük olanın tam katı olup olmadığını tespit edip ekrana yazdıran bir program geliştiriniz.

**e- Üçgen midir? (Kenar uzunlukları ile):**

Ekrandan girilecek 3 ondalık sayı değeri ile bir üçgen oluşturup oluşturulamayacağının kararını veren bir program yazın. Girilen değerler bir üçgenin kenar uzunluğu olarak kullanılmalıdır. Sonucu ekrana yazdırın.

**f- Üçgen midir? (Açı değerleri ile):**

Ekrandan girilecek 3 tam sayı değeri ile bir üçgen oluşturup oluşturulamayacağının kararını veren bir program yazın. Girilen değerler bir üçgenin açı değerleri olarak kullanılmalıdır. Sonucu ekrana yazdırın.

**g- Dik üçgen midir? (Kenar uzunlukları ile):**

Ekrandan girilecek 3 tam sayı değeri ile bir dik üçgen çizilip çizilemeyeceğinin kararını veren programı yazın. Girilen değerler bir üçgenin kenar uzunlukları olarak kullanılmalıdır. Sonucu ekrana yazdırın.

**h- Dik üçgen midir? (Açı değerleri ile):**

Ekrandan girilecek 3 tam sayı değeri ile bir dik üçgen çizilip çizilemeyeceğinin kararını veren programı yazın. Girilen değerler bir üçgenin açı değerleri olarak kullanılmalıdır. Sonucu ekrana yazdırın.

## **6- Döngü Yapısı**

**a- Girilen 10 sayıdan en küçük/büyük olanı tespit etme:**

Kullanıcıdan istenecek 10 adet sayının içerisinde en büyük ve en küçük olanları tespit edip ekrana sonucu yazan programı geliştirin. Kullanıcının yazdığı değerlerin sayı olup olmadığının kontrolü yapılmalıdır. Kullanıcı uygun formatta sayı yazmazsa ("Yirmi beş", "üç", "Muhittin") uyarı döndürülecek ve girilen değer geçersiz sayılacaktır. Programı geliştirirken minimum sayıda değişken kullanın, her giriş yapılan değer için değişken oluşturmayın. Ayrıca kodlamada dizi kullanılmayacaktır. Program bir kez hesap yaptıktan sonra herhangi bir tuşa basıldığında sonlanacaktır.

**b- Girilen belirsiz miktardaki sayıdan en küçük/büyük olanı tespit etme:**

Kullanıcıdan istenecek belirsiz miktardaki sayı içerisinden en büyük ve en küçük olanları tespit edip ekrana sonucu yazan programı geliştirin. Kullanıcıdan toplamda kaç adet sayı alınacağı kullanıcıya bağlı olacak, kullanıcı "end" yazdığında sayı alma işlemi durdurulacak ve sonuçlar ekrana yazdırılacak. Ancak kullanıcı minimum 2 sayı yazmak zorundadır. Kullanıcının yazdığı değerlerin sayı olup olmadığının kontrolü yapılmalıdır. Kullanıcı uygun formatta sayı yazmazsa ("Yirmi beş", "üç", "Muhittin") uyarı döndürülecek ve girilen değer geçersiz sayılacaktır. Programı geliştirirken minimum sayıda değişken kullanın, her giriş yapılan değer için değişken oluşturmayın. Ayrıca kodlamada dizi kullanılmayacaktır. Program hesaplama ve tespit işlemlerini yaptıktan sonra başa dönecek ve tekrar sayı girişi bekleyecektir. Programı sonlandırmak, kullanıcının "exit" yazmasına bağlı olacaktır. NOT: "end" komutu, eğer ki toplamda 2 sayıya ulaşılmadıysa işlevsiz olacaktır ve uyarı döndürecektir. Ancak "exit" komutu herhangi bir anda doğrudan çalışabilir. Ayrıca

kullanıcı uygun formatta sayı yazmazsa (“Yirmi beş”, “üç”, “Muhittin”) uyarı döndürülecek ve girilen değer geçersiz sayılacaktır.

**c- Girilen aralıktaki sayıların kare ve küp hesapları:**

Kullanıcıdan alınacak iki sayı arasında kalan tüm sayıların kare ve küp değerleri hesaplanacaktır. Kullanıcı iki sayıyı birincisi küçük, ikincisi büyük olacak şekilde veya birincisi büyük, ikincisi küçük olacak şekilde sıraya bağlı kalmaksızın girebilir. Kullanıcının yazdığı değerlerin sayı olup olmadığının kontrolü yapılmalıdır. Kullanıcı uygun formatta sayı yazmazsa (“Yirmi beş”, “üç”, “Muhittin”) uyarı döndürülecek ve girilen değer geçersiz sayılacaktır. Yapılan hesaplamalar ekrana üç kolonlu yapıda yazdırılacaktır: Sayı {TAB} Karesi {TAB} Küpü Program hesaplama ve tespit işlemlerini yaptıktan sonra başa dönecek ve tekrar sayı girişi bekleyecektir. Programı sonlandırmak, kullanıcının “exit” yazmasına bağlı olacaktır.

## 7- Diziler

**a. Dizi Elemanlarını alt alta yazdırma:**

İlk aşamada dizi içerisindeki elemanlar bir döngü içerisinde alt alta yazdırılır. WriteLine metodu veya Write metodu içerisinde “\n” çıkış karakteri ile uygulama yaptırılabilir.

**b. Dizi elemanlarını alt alta tersten yazdırma:**

İkinci aşamada dizi elemanlarının yerleri değiştirilmeden, dizi tersten ekrana yazdırılır. Bu örnekte öğrenciler, azalan döngü kurarak algoritmayı yazmalıdır.

**c. Dizi elemanlarını yan yana yazma:**

Üçüncü aşamada dizi içerisindeki elemanlar 45 – 928 – 78 – 4 – 1007 – 83 şeklinde yan yana yazdırılır. Elemanlar arasında mutlaka bir ayraç karakter olmalıdır. İlk elemandan önce veya son elemandan sonra ekranda – (veya diğer bir ayraç) karakterinin olmaması gerektiği söylenir. Aynı örnek, dizi elemanlarını ters sırada soldan sağa yazdırarak da yapılabilir.

**d. Dizi elemanlarını iki kolonda yazdırma:**

Dördüncü uygulamada dizi elemanları her satırda iki eleman, arada bir sekme boşluk olacak şekilde ekrana yazdırılır. Çıktı elde edildiğinde ekranda sanal iki kolonlu bir yapı görülmektedir:

```
**1**      **2**
dizi[0]      dizi[1]
dizi[2]      dizi[3]
dizi[4]      dizi[5]
dizi[6]      dizi[7]
```

**e. Dizi elemanlarını üç kolonda ters sırada yazdırma:**

Dördüncü olarak dizi elemanlarını 3 kolonlu sanal bir tabloya yazdırır. Ancak kolonların Right to Left formatta, sağdan sola dizilmiş şekilde olması gerekmektedir. Örnek çıktı aşağıdaki gibidir:

<b>**3**</b>	<b>**2**</b>	<b>**1**</b>
dizi[2]	dizi[1]	dizi[0]
dizi[5]	dizi[4]	dizi[3]
dizi[8]	dizi[7]	dizi[6]
		dizi[9]

**f. İkinci bir dizi yardımıyla dizi elemanlarını ters sıralamak:**

Öncelikle birinci dizinin boyutunda yeni bir dizi oluşturarak, tek bir döngü içerisinde birinci dizideki her bir elemanı ikinci dizideki elemanlara ters sırada atamak ile yeni dizi elde edilir.

**g. Aynı dizi üzerinde elemanları ters sıralamak:**

Aynı dizi üzerinde, dizi eleman sayısının yarısı kadar yürütülecek bir döngü içerisinde, klasik yer değiştirme algoritması kullanılarak dizi elemanları ters çevrilebilir.