Veri Yapıları

**Proje-1**

**Ali Yeşilkanat** – 05120000223

**İçindekiler**

İçindekiler

1)Program 1.1**3**

1) Gerçekleştirilen Platform3

2) Proje Tanımı3  
3) Kullanılan Metotlar ve Bileşik Veri Tipleri ............................................................................. 3  
 3.1) Kullanılan Metotlar.................................................................................................3  
 3.2) Kullanılan Bileşik Veri Tipleri...................................................................................3  
4) Örnek Sonuçlar.......................................................................................................................4  
5) Yazılım Geliştirme İçin Harcanan Süre....................................................................................4

2)Program 1.2**5**

1) Gerçekleştirilen Platform5  
2) Proje Tanımı5  
3) Kullanılan Metotlar ve Bileşik Veri Tipleri.............................................................................. 5  
 3.1) Kullanılan Metotlar.................................................................................................5  
 3.2) Kullanılan Bileşik Veri Tipleri...................................................................................5  
4) Örnek Sonuçlar.......................................................................................................................6  
5) Yazılım Geliştirme İçin Harcanan Süre....................................................................................6  
**Ek:** Program 1.2’nin c seçeneğinin Sözdekodu / Algoritması......................................................7

# Program 1.1

1. **Gerçekleştirilen Platform:** Visual Studio 2010 , .Net Framework 4
2. **Proje Tanımı:** Rastgele secilmis 23 kişi veya daha fazlalık bir grupta, en az iki kişinin doğum gününün aynı olma olasılığı %50'dir. Aynı şekilde 60 veya daha fazla insan için bu olasılık yaklaşık %99'dur. 366 kişiye kadar bu olasılık %99'dan fazladır, ancak 366 kişide bu olasılık %100 olur. Projede istenilen, belli koşullara göre rastgele doğum günleri üretip bunların eşleşme sayılarının istatistiklerinin bulunmasıdır.
3. **Kullanılan Metotlar ve Bileşik Veri Tipleri:**

**3.1) Kullanılan Metotlar**

**private** **static** **int** **SayiAl**()  
**private** **static** **int** **SayiAl**(**int** a,**int** u)  
Sayıların kullanıcıdan alınmasını sağlar. Eğer kullanıcı sayıdan başka bir girdi girerse kullanıcıya hata mesajı verip tekrar girmesini ister. Argüman isteyen aşırı yüklenmiş versiyonu ise, kullanıcının üst ve alt sınır değerleri arasında bir sayı girmesini sağlar.

**private** **static** **int** **AnaMenu**()  
Ana menüyü listeler ve seçilen menü numarasını geri döndürür.

**private** **static** **void** **HerDeneyiGerceklestir**(Random r, **int** yil1, **int** yil2)  
Belirtilen yıllar arasında deneyleri gerçekleştirir, her birindeki doğum günü sayılarını ve deneyler sonundaki istatistikleri ekrana yazdırır.

**private** **static** **void** **DeneyGerceklestir**(Random r, **int** yil1, **int** yil2)

Belirtilen yıllar arasında deneyleri gerçekleştirip sadece istatistikleri ekrana yazdırır.

**private** **static** DateTime[] **nRandomBDayGenerator**(Random r, **int** yil1, **int** yil2, **int** n)  
Belirtilen yıllar arasında n rastgele, n tane doğum günü üretip DateTime dizisine atar ve diziyi geri döndürür.

**private** **static** DateTime **randomBDayGeneratorBetweenTwoYears**(Random r, **int** yil1, **int** yil2)  
Belirtilen 2 yıl arasında bir doğum günü üretip tarihi DateTime sınıfı nesnesi olarak geri döndürür.

**3.2) Kullanılan Bileşik Veri Tipleri**

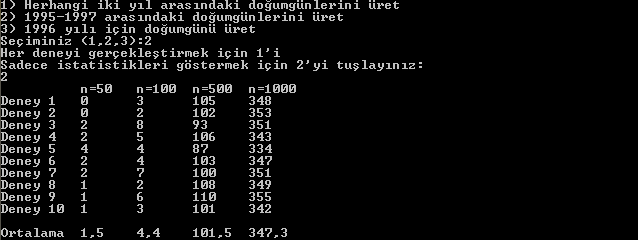
**int** [,] cakismaSayilari= **new** **int**[10,4];   
Çakışma-Eşleşme sayılarını tutan matristir. 4 n için(50,100,500,1000) 10 deney sonucunda çıkan değerleri tutar.

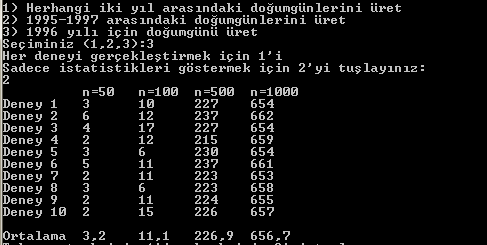
**int**[] nDizisi = { 50, 100, 500, 1000 };  
Deneylerde kullanılacak kişi sayısını tutar.

DateTime[] bDays = nRandomBDayGenerator(r, yil1, yil2, n);  
DateTime[] bDays = **new** DateTime[n];  
Üretilen n doğum günü sayısını tutar.

**int**[][] yildakiDGunleri = **new** **int**[12][];  
Belirtilen yıldaki günleri aylara göre tutan düzensiz dizidir.

1. **Örnek Sonuçlar**





1. **Yazılım Geliştirme İçin Harcanan Süre:** 6 saat

# Program 1.2

1. **Gerçekleştirilen Platform:** Visual Studio 2010 , .Net Framework 4
2. **Proje Tanımı:** Öğrencilerin 9 ülkedeki üniversitelerden birisine 3. sınıfta 1 seneliğine okumak için gideceği düşünülmekte ve bizden yerleştirmeyi yapan yazılımı geliştirmeniz istenmektedir  
   Bu yerleştirmeyi yaparken de ülkelerdeki doluluk oranının eşit olmasının sağlanması istenmektedir.
3. **Kullanılan Metotlar ve Bileşik Veri Tipleri:**

**3.1) Kullanılan Metotlar:**

**private** **static** **int** **SayiAl**()  
**private** **static** **int** **SayiAl**(**int** a,**int** u)  
Sayıların kullanıcıdan alınmasını sağlar. Eğer kullanıcı sayıdan başka bir girdi girerse kullanıcıya hata mesajı verip tekrar girmesini ister. Argüman isteyen aşırı yüklenmiş versiyonu ise, kullanıcının üst ve alt sınır değerleri arasında bir sayı girmesini sağlar.

**private** **static** **void** **ogrencileriYerlestir**(List<Ulke> ulkeListesi, List<Ogrenci> ogrenciListesi)  
Öğrencieri ülkelere yerleştirir.

**private** **static** **void** **ogrenciOlusturma**(Random r, List<Ogrenci> ogrenciListesi)  
Rastgele isim ve soyisimleriyle öğrenci oluşturur. Rastgele bir değerlendirme puanı vererek eğer barajı geçmişsse öğrenciyi öğrenci listesine atar.

**private** **static** **void** **kontenjanSayilariniAl**(Random r,List<Ulke> ulkeListesi)

Ülkelerin kontenjan sayılarını istenilen duruma göre ya rastgele yaratır ya da kullanıcıdan alır

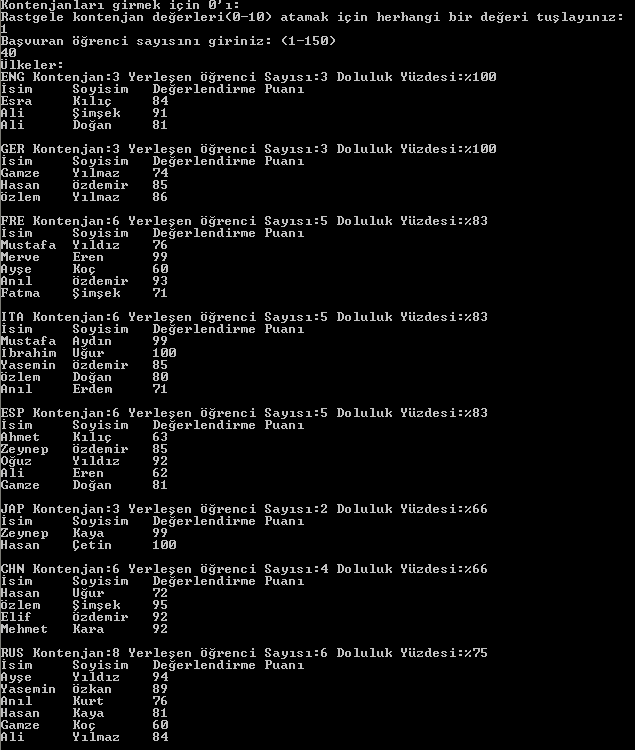
**3.2) Kullanılan Bileşik Veri Tipleri:**  
List<Ulke> ulkeListesi = **new** List<Ulke>();  
 Programda kullanılacak ülkeleri tutmak için oluşturulmuştur.  
   
List<Ogrenci> ogrenciListesi = **new** List<Ogrenci>();  
Programda kullanılacak öğrencileri tutmak için oluşturulmuştur.

**string**[] isimler = { "Ali", "Hasan", "Fatma", "Oğuz", "Ayşe", "Zeynep", "Elif", "Merve", "Ece", "Esra", "Kaan", "Özlem", "Yasemin", "Anıl", "Mehmet", "Mustafa", "Ahmet", "Gamze", "Hüseyin", "İbrahim" };

**string**[] soyisimler = { "Yıldız", "Yıldırım", "Avcı", "Öztürk", "Kaya", "Erdem", "Aydın", "Özdemir", "Arslan", "Doğan", "Kılıç", "Çetin", "Kara", "Koç", "Uğur", "Kurt", "Özkan", "Eren", "Şimşek", "Yılmaz" };  
Öğrencileri yaratırken kullanılan örnek isim ve soyisimler.  
  
**string**[] ulkeler = { "ENG", "GER", "FRE", "ITA", "ESP", "USA", "JAP", "CHN", "RUS" };  
Ülkelerin isimlerinin tutulmasını sağlayan dizi.

**public** List<Ogrenci> Ogrenciler;  
 Ülkeler sınıfında yerleşen öğrencileri tutmak için oluşturulmuştur.

1. **Örnek Sonuçlar**



1. **Yazılım Geliştirme İçin Harcanan Süre:** 4 saat

**Ek:** Program 1.2’nin c seçeneğinin Sözdekodu / Algoritması

**void ogrencileriYerlestir(List<Ulke> ulkeListesi, List<Ogrenci> ogrenciListesi)**

totalKontenjan ← 0

Repeat foreach Ulke item in ulkeListesi

item.Ogrenciler ← new List<Ogrenci>()

totalKontenjan ← item.Kontenjan+totalKontenjan

End Repeat

If totalKontenjan < ogrenciListesi.Count

ogrenciListesi.Sort()

ogrenciListesi.Reverse()

End If

bosKontenjan ← totalKontenjan;

totalOgrenciSayisi ← ogrenciListesi.Count;

Repeat for i ←0 to totalOgrenciSayisi-1

Ulke atanacakUlke ← ulkeListesi.OrderBy(item => item.DolulukYuzdesi()).First()

atanacakUlke.Ogrenciler.Add(new Ogrenci(ogrenciListesi[0]))

ogrenciListesi.RemoveAt(0)

ogrenciListesi.RemoveAllNullElements  
 if --bosKontenjan = 0

exitloop

End If

End Repeat

If ogrenciListesi.Count > 0

Write"Yerleştirilemeyen öğrencilerin sayısı:" , ogrenciListesi.Count;

Write "İsim Soyisim Değerlendirme Puanı"

Repeat foreach Ogrenci item in ogrenciListesi

Write item.Ad, item.Soyad, item.DegerlendirmeNotu

End Repeat  
 End If

Write "Ülkeler:"

Repeat foreach Ulke item in ulkeListesi

Write item.Isim, " Kontenjan:" ,item.Kontenjan, " Yerleşen Öğrenci Sayısı:", item.Ogrenciler.Count," Doluluk Yüzdesi:%",item.DolulukYuzdesi()

Write "İsim Soyisim Değerlendirme Puanı"

Repeat foreach Ogrenci o in item.Ogrenciler

Write o.Ad, o.Soyad, o.DegerlendirmeNotu

End Repat

End Repat