

Sınav Dağılım & IMKB Endeks

Ege Üniversitesi Bilgisayar Mühendisliği

Veri Yapıları

Proje-1

Hüseyin YAŞAR 05-06-7657

Didem KAYALI 05-06-7669

Umut BENZER 05-06-7670

Özlem GÜRSES 05-07-8496

Sürüm: 0.2



Bölüm 1: Sınav Dağılımı

Genel Bilgi

Proje yerleştirilmesi uygun sınıflar verilmesi halinde rastgele öğrenci isim ve numaraları üretmek bunları sınıflara yerleştirir.

Proje kolaylık açısından fonksiyonlara parçalanmıştır. Her fonksiyonun algoritmaları pseudocode olarak verilmiştir. Ayrıca fonksiyonlarda birçok açıklama mevcuttur.

Aşağıda fonksiyon listesini ve giriş/çıkış değerlerini tek bir bakışta görebilirsiniz.

```
/* function main
 * Author: Hüseyin Yaşar
 */
public static void main(String[] args) {}

/* function sirala
 * Author: Didem Kayalı
 *
 * @param: Sıralanacak integer dizisi
 * @return: Sıralanmış integer dizisi (büyükten küçüğe)
 *
 * Dizi referans olarak gönderileceğinden void döndürmesi
 * yeterlidir. Asıl değişiklikler asıl diziyeye yansıtılır.
 */
private void sirala(int[] sinifKapasite) {}

/* function dagilimBul
 * Author: Hüseyin Yaşar
 *
 * @param: Sıralanmış sınıf kapasiteleri, öğrenci sayısı, boş sınıf mevcudu dizisi
 * @return: Sınıflara kaç kişi koyulacağına ilişkin bilgisi
 */
private void dagilimBul(int[] sinifKapasite, int ogrSayi, int sinifMevcut[]) {}

/* function dagit
 * Author: Özlem Gürses
 *
 * @param: Sıralanmış sınıf kapasiteleri, içerilerine yerleştirilecek öğrenci sayısı
 * @return: Ekranaya çıktı
 */
private void dagit(int[] sinifKapasite, int sinifMevcut[]) {}

/* function kacSinirKullanilacak
 * Author: Hüseyin Yaşar
 *
 * @param: Öğrenci sayısı ve sınıf kapasiteleri
 * @return: Öğrencilerin kaç sınıfa yerleştirileceği
 */
private int kacSinifKullanilacak(int ogrSayi, int sinifKapasite[]) {}
```

Pseudocode

main()

```
kntrl=0;

read str
do{
    top :=0
    if kntrl=1 then
        write "Hatali giris"
    endif
    write "Yerlestirilecek ogrenci sayisini giriniz: "
    read ogrSayi

    if ogrSayi<100 or ogrSayi>200 then
        kntrl := 1
        continue
    endif

    repeat for i:=0 to 5
        write (i+1)
        write ". sinifin kapasitesini giriniz: "
        read sinifKapasite[i]
        if sinifKapasite[i]<0 then break
        endif
        top :=top + sinifKapasite[i]
    end repeat

    if top<ogrSayi then kntrl :=1
    else kntrl :=0
    endif

} while kntrl=1
Call sirala(sinifKapasite)
Call dagilimBul(sinifKapasite, ogrSayi, sinifMevcut)
Call dagit(sinifKapasite, sinifMevcut)
```

dagilimBul(sinifKapasite[], ogrSayi, sinifMevcut[])

```
kapasitelerToplami := 0
a := kacSinifKullanilacak(ogrSayi,sinifKapasite)
kusuratlarToplami := 0

repeat for i := 0 to a
    kapasitelerToplami := kapasitelerToplami + sinifKapasite[i]
end repeat

repeat for j :=0 to <a
    sinifMevcutOndalikli[j] := sinifKapasite[j]*(ogrSayi/kapasitelerToplami)
    sinifMevcut[j] := (int)sinifMevcutOndalikli[j]
    kusuratlarToplami := kusuratlarToplami + sinifMevcutOndalikli[j]
sinifMevcut[j]
end repeat

if kusuratlarToplami>0 then
    repeat for i=0 to a
        sinifMevcut[i] := sinifMevcut[i] + 1
```

```

        kusuratlarToplami := kusuratlarToplami - 1

        if kusuratlarToplami<=0 then break
        endif
    end repeat
endif

/* Son kısım küsüratların toplamı olan tamsayıyı soldan sağa birer birer bitene
kadar dağıtır. Küsüratlar toplamı daima tamsayı gelmesi gereken bir sayıdır, örneğin
0.2 ve 0.8 gibi artanlar olur, bunların toplamı daima tamsayıdır, ama bazen
bölümlerde 0.19999999 ve 0.7999999999 gibi sayılar gelebilir, bunların toplamı
0.999999 gibi sayılardır, dolayısıyla 1 eksildiği zaman 0'ın altına düşer, bu yüzden
if satırında küçüktür kusuratlarToplami<=0 ibaresi kullanılmıştır. */

```

kacSinifKullanilacak(ogrSayi, sinifKapasite[])

```

kapasitelerToplami := 0
kullanilacakSinifSayisi := 0
Repeat for i := 0 to 4
    kapasitelerToplami := kapasitelerToplami + sinifKapasite[i]
    kullanilacakSinifSayisi := kullanilacakSinifSayisi + 1

    if kapasitelerToplami>=ogrSayi then break
    endif

end repeat

return kullanilacakSinifSayisi

```

sirala(sinifKapasite[])

```

repeat for i:=0 to length(sinifKapasite)-1
    repeat for j=i+1 to length(sinifKapasite)
        if sinifKapasite[j]>sinifKapasite[i] then
            tmp := sinifKapasite[j]
            sinifKapasite[j] := sinifKapasite[i]
            sinifKapasite[i] := tmp
        endif
    end repeat
end repeat

```

dagit(sinifKapasite[],sinifMevcut[])

```

ad[13]={umut,didem,özlem,hüseyin,hülya,banu,ajdar,ayşe,mehmet,hüsnü,canan,metin,okşan}
soyad[13]={benzer,kayalı,gürses,yaşar,avşar,alkan,anık,sert,andıç,yılmaz,şahin,kurt,çelik,boya,hüyük,güler,sever,ceylan}

```

```

ekleAd[4] = {can,gül,nur,su}
ekleSoyad[3] = {oğlu,gil,öz}
repeat for i=0 to sinifKapasite.length-1
    if sinifKapasite[i]=0
        write(i+1, '. sınıf boş sınıftır.')
    else if sinifMevcut[i]=0
        write(sinifKapasite[i], 'kişilik sınıfa öğrenci yerleştirilmemiştir.')
    else
        ort = 100 * sinifMevcut[i] / sinifKapasite[i]
        write(sinifKapasite[i], 'kişilik sınıfa', sinifMevcut[i], 'kişi yerleştirildi. sınıfın doluluk oranı: 'ort, ' %dır.')

        repeat for k=0 to sinifMevcut[i]

```

```
    yeniIsim = ad[(int) (Math.random() *
    ad.length)].concat(ekleAd[(int) (Math.random()*
    ekleAd.length)])

    yeniSoyisim = soyad[(int) (Math.random() * soyad.length)]
    .concat(ekleSoyad[(int) (Math.random() *
    ekleSoyad.length)])

    yil=(int) (Math.random() * 8)

    write((k+1),'. 05-0',yil,'-',(int) (Math.random() * 9),(int)
    (Math.random() * 9),(int) (Math.random() * 9),(int)
    (Math.random() * 9),yeniIsim,yeniSoyisim)
    end repeat
  end if
end repeat
```

Bölüm 2: IMKB Endeksi

Genel Bilgi

Proje sinüs fonksiyonu ve çeşitli parçalı fonksiyonlar ile üretilen rastgele endeks değerlerini kullanarak bir yıla ait endeks değerlerinin değişim oranlarını, ortalamalarını ve en yüksek/düşük endeks değerlerini hesaplayarak ekrana yazdırır.

Proje kolaylık açısından fonksiyonlara parçalanmıştır. Her fonksiyonun algoritmaları pseudocode olarak verilmiştir. Ayrıca fonksiyonlarda birçok açıklama mevcuttur.

Aşağıda fonksiyon listesini ve giriş/çıkış değerlerini tek bir bakışta görebilirsiniz.

```
/* function main
 * Author: Umut Benzer
 */
public static void main(String[] args) {}

/* function hangiGun
 * Author: Didem Kayalı
 *
 * @param: Seans numarası
 * @return: Ayın bir günü (1-31) Seansın ayın hangi
 * gününe ait olduğunu döndürür.
 */
private static int hangiGun(int seansno) {}

/* function hangiAy
 * Author: Didem Kayalı
 *
 * @param: Seans numarası
 * @return: Yılın bir ayı (1-12) Seansın ayın hangi
 * aya ait olduğunu döndürür.
 */
private static int hangiAy(int seansno) {}

/* function degisimOrani
 * Author: Didem Kayalı
 *
 * @param: İlk seansın değeri
 * @param: İkinci seansın değeri
 * @return: Seanslar arası değişim oranı
 *
 * Seanslar arasındaki değişim oranını %
 * cinsinden döndürür.
 */
private static float degisimOrani(float ilkyuzde, float sonyuzde) {}

/* function tatilMi
 * Author: Özlem Gürses
 *
 * @param: Seans numarası
 * @return: Seansın olması gerektiği günün tatil olup
 * olmadığı
 */
private static boolean tatilMi(int no) {}

/* function deger
 * Author: Hüseyin Yaşar
 *
 * @param: Seans numarası
 * @return: Eğer seans tatil gününe denk gelmiyorsa
 * rastgele bir seans değeri üretir. Tatil günleri için -1 döndürür.
 */
private static float deger(int seansno) {}
private static float randomFonksiyon(int x) {}
private static float parcaliFonksiyon(int a) {}
```

Pseudocode

main

Ay = 1

IMKB Değerleri Toplamı = 0

IMKB Değerleri Sayısı = 0

Bir Önceki Ayın Son Seansının IMKB Değeri = *deger*(0)

Bir Önceki Seansın Değeri = *deger*(0)

Bir Önceki Seansın Numarası = 0

En Yüksek Değerler = { -1, -1, -1 };

En Yüksek Değerlerin Seans Numaraları = { -1, -1, -1 };

En Düşük Değerler = { -1, -1, -1 };

En Düşük Değerlerin Seans Numaraları = { -1, -1, -1 };

for sayac = 1 **to** sayac = 730

```
    if (deger(sayac) != -1) /* Eğer tatil gününe denk gelmediyse */
        /* Değişim oranını ilgili fonksiyonda hesapla ve ekrana yazdır. */
        print degisimOrani(Bir Önceki Seansın Değeri, deger(sayac))
```

Bir Önceki Seansın Değeri = *deger*(sayac)

Bir Önceki Seansın Numarası = sayac

IMKB Değerleri Toplamı += *deger*(sayac);

IMKB Değerleri Sayısı++;

if En Düşük Değerler Listesi Boşsa ya da *deger*(sayac) En Düşük Değerler Listesindeki elemanlardan daha düşük bir değere sahipse

deger(sayac) kendisinden büyük ilk elemanla kendisinden küçük ilk eleman arasına geçer, listenin en büyük elemanı listeden silinir.

En düşük değerler seans numarasının uygun yerine sayac değeri yazılır.

endif

if En Yüksek Değerler Listesi Boşsa ya da *deger*(sayac) En Yüksek Değerler Listesindeki elemanlardan daha yüksek bir değere sahipse

deger(sayac) kendisinden küçük ilk elemanla kendisinden büyük ilk eleman arasına geçer, listenin en küçük elemanı listeden silinir.

En büyük değerler seans numarasının uygun yerine sayac değeri yazılır.

endif

if (*hangiAy*(sayac + 1) != ay) /* Ay sonu */

/* Ay için endeks ortalaması ve değişim oranı */

print IMKB Değerleri Toplamı / IMKB Değerleri Sayısı

print *degisimOrani* Bir Önceki Ayın Son Seansının IMKB Değeri, *deger*(sayac))

Bir Önceki Ayın Son Seansının IMKB Değeri = *deger*(sayac);

IMKB Değerleri Toplamı = 0

IMKB Değerleri Sayısı = 0

Ay++;

endif

endif

endfor

En Düşük Seans ve En Yüksek Seans bilgileri kullanılarak en yüksek ve en düşük seans bilgileri ekrana yazdırılır.

deger (seansno)

```
if seansno is tatil then
    return -1
else
    IMKBdegeri = 30000 + 20000 * randomFonksiyon(seansno)
    return IMKBdegeri
endif
```

randomFonksiyon (x)

```
y = sin(x) + 1
y = y + sin(x/2) + 1
y = y + parcaliFonksiyon(x)
y = y / 4
return y;
```

/* Yukarıda y değerinin randomlaşması aşama aşama gösterilmiştir, kodlamada hepsi tek satırda yazılabilir. */

parçalıFonksiyon (x)

```
if 0<=x and x<70
then y:=0
else if 70<=x and x<315
then y:=-x/180+1
else if 315<=x and x<450
then y:=1
else if 450<=x and x<720
then y=x/360+2
else if 720<=x
then y=0
else
    write "Random hatası"
    y:=-1
endif
endif
endif
endif
endif
return y;
```

tatilMi(no)

```
yil[12]={0,31,28,31,30,31,30,31,31,30,31,30,31}
ay<--hangiAy(no)
gun<--0
Repeat for i<--0 to ay
    gun<--gun+yil[i]
End Repeat
gun<--gun+hangiGun(no)
if (gun mod 7 = 0) OR (gun mod 7 =6) then
    return true
else
    return false
endif
```


hangiAy(seansno)

```
if (seansno>=1) AND (seansno<=62)
    1 döndür
else if (seansno>=63) AND (seansno<=118)
    2 döndür
else if (seansno>=119) AND (seansno<=180)
    3 döndür
else if (seansno>=181) AND (seansno<=240)
    4 döndür
else if (seansno>=241) AND (seansno<=302)
    5 döndür
else if (seansno>=303) AND (seansno<=362)
    6 döndür
else if (seansno>=363) AND (seansno<=424)
    7 döndür
else if (seansno>=425) AND (seansno<=486)
    8 döndür
else if (seansno>=487) AND (seansno<=546)
    9 döndür
else if (seansno>=547) AND (seansno<=608)
    10 döndür
else if (seansno>=609) AND (seansno<=668)
    11 döndür
else if (seansno>=669) AND (seansno<=730)
    12 döndür
else
    -1 döndür
endif
```

/* Her gün için iki seans bulunduğundan 730 seans numarası vardır. Fonksiyon seans numarasını alarak yılın hangi ayı içerisinde olduğunu döndürür. */

degisimOrani(ilk deęer, son deęer)

$(100 * (\text{sonyuzde} - \text{ilkuyuzde})) / \text{ilkuyuzde}$ deęerini döndür

hangiGun(seansno)

```
int kalan = seansno( mod) 2
if ((seansno >= 1) AND (seansno <= 62))
    if (kalan == 1)
        ((seansno / 2) + 1) döndür
    else
        (seansno / 2) döndür
else if ((seansno >= 63) AND (seansno <= 118))
    if (kalan == 1)
        (((seansno - 62) / 2) + 1) döndür
    else
        ((seansno - 62) / 2) döndür

else if ((seansno >= 119) AND (seansno <= 180))
    if (kalan == 1)
        (((seansno - 118) / 2) + 1) döndür
    else
        ((seansno - 118) / 2) döndür
else if ((seansno >= 181) AND (seansno <= 240))
```

```

    if (kalan == 1)
        (((seansno - 180) / 2) + 1) döndür
    else
        ((seansno - 180) / 2) döndür
else if ((seansno >= 241) AND (seansno <= 302))
    if (kalan == 1)
        (((seansno - 240) / 2) + 1) döndür
    else
        ((seansno - 240) / 2) döndür
else if ((seansno >= 303) AND (seansno <= 362))
    if (kalan == 1)
        (((seansno - 302) / 2) + 1) döndür
    else
        ((seansno - 302) / 2) döndür
else if ((seansno >= 363) AND (seansno <= 424))
    if (kalan == 1)
        (((seansno - 362) / 2) + 1) döndür
    else
        ((seansno - 362) / 2) döndür
else if ((seansno >= 425) AND (seansno <= 486))
    if (kalan == 1)
        (((seansno - 424) / 2) + 1) döndür
    else
        ((seansno - 424) / 2) döndür
else if ((seansno >= 487) AND (seansno <= 546))
    if (kalan == 1)
        (((seansno - 486) / 2) + 1) döndür
    else
        ((seansno - 486) / 2) döndür
else if ((seansno >= 547) AND (seansno <= 608))
    if (kalan == 1)
        (((seansno - 546) / 2) + 1) döndür
    else
        ((seansno - 546) / 2) döndür
else if ((seansno >= 609) AND (seansno <= 668))
    if (kalan == 1)
        (((seansno - 608) / 2) + 1) döndür
    else
        ((seansno - 608) / 2) döndür
else if ((seansno >= 669) AND (seansno <= 730))
    if (kalan == 1)
        (((seansno - 668) / 2) + 1) döndür
    else
        ((seansno - 668) / 2) döndür
else
    -1 döndür

```