

# CENG 111 ALGORİTMALAR VE PROGRAMLAMA

Doç. Dr. Tufan TURACI

tturaci@pau.edu.tr

- Pamukkale Üniversitesi
- Mühendislik Fakültesi
- Bilgisayar Mühendisliği Bölümü
- Hafta 7
- 1 Kasım 2022

# DİZİLER

## (Tek boyutlu\_İndisli Değişkenler)

- Dizi nedir?
- Dizilerin Bildirimi
- Dizilere Başlangıç Değerinin Verilmesi

# Dizi Nedir?

- Diziler bir veri yapısı türüdür. (Array, Struct, Pointer, Class, Graphs ...).
- Bir dizinin içerisinde birden fazla aynı tipte ve aynı isimde veri bulunur, ve bu verileri bellekte saklayabilen **değişkenlerdir**.
- **Örneğin**; bir sınıftaki öğrencilerin aldığı notlar bir **Dizide** saklanabilir.
- Tek boyutlu indisli değişkenler olarakda ifade edilirler.

# Dizilerin Bildirimi

- Bir dizi çok sayıda değişken değeri saklayabildiğinden, bunlar indis adı verilen bilgiler yardımıyla birbirlerinden ayrılırlar.
- C Programlama Dili'nde, bir dizi hangi tipte tanımlanmış olursa olsun **başlangıç indisi** her zaman **0**' dır.
- Diziler tanımlanırken;
  - dizinin **adı**,
  - dizinin **boyutu**,
  - dizi elemanlarının hangi **tipte**olacağı belirtilmelidir.

A[0]	A[1]	A[2]	A[3]	A[4]	A[5]
65	90	100	45	35	80

*6 elemanlı A dizisi*

- Bir dizinin aşağıdaki şekilde tanımlanır:

**Dizinin\_Veri\_tipi    dizi\_adı[eleman\_sayısı] ;**

- Örneğin; integer türündeki 6 adet öğrencinin notu bellekte tutmak için aşağıdaki gibi bir dizi tanımlanır

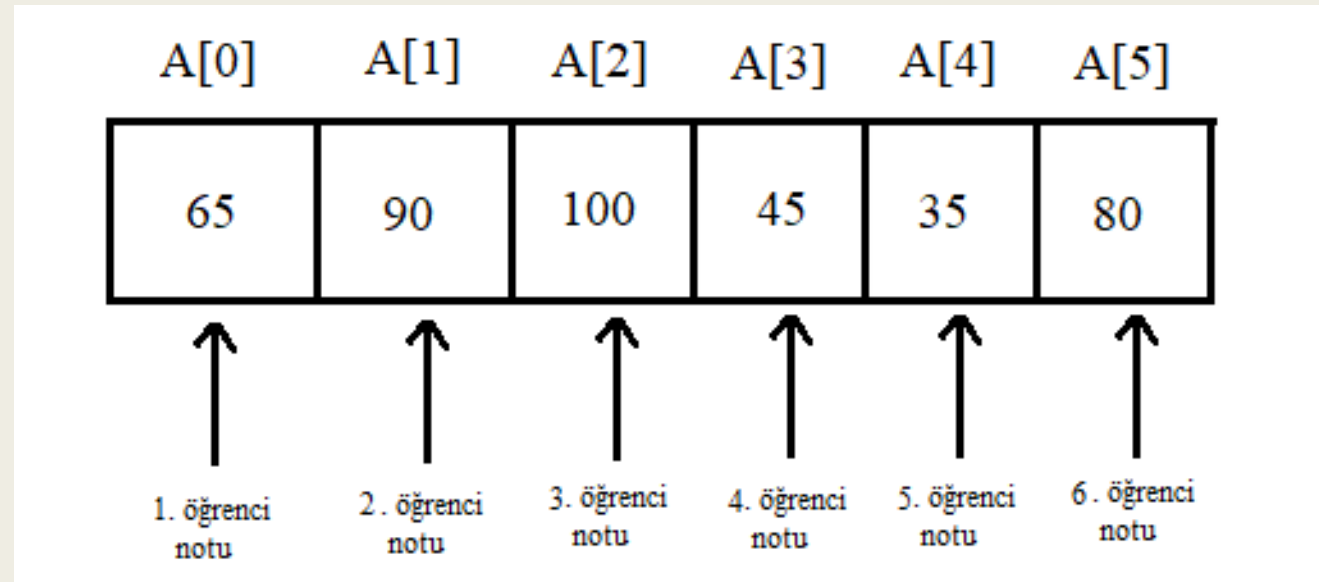
**int A[6] ;**

--- Şu andaki bilgilerimizle bunu nasıl yapabiliriz?

```
int n1,n2,n3,n4,n5,n6;
```

Şeklinde 6 farklı değişken tanımlamalıydık.

--- Dizi yardımıyla tek bir değişken (indisli) ile bu tanımlama yapıldı.



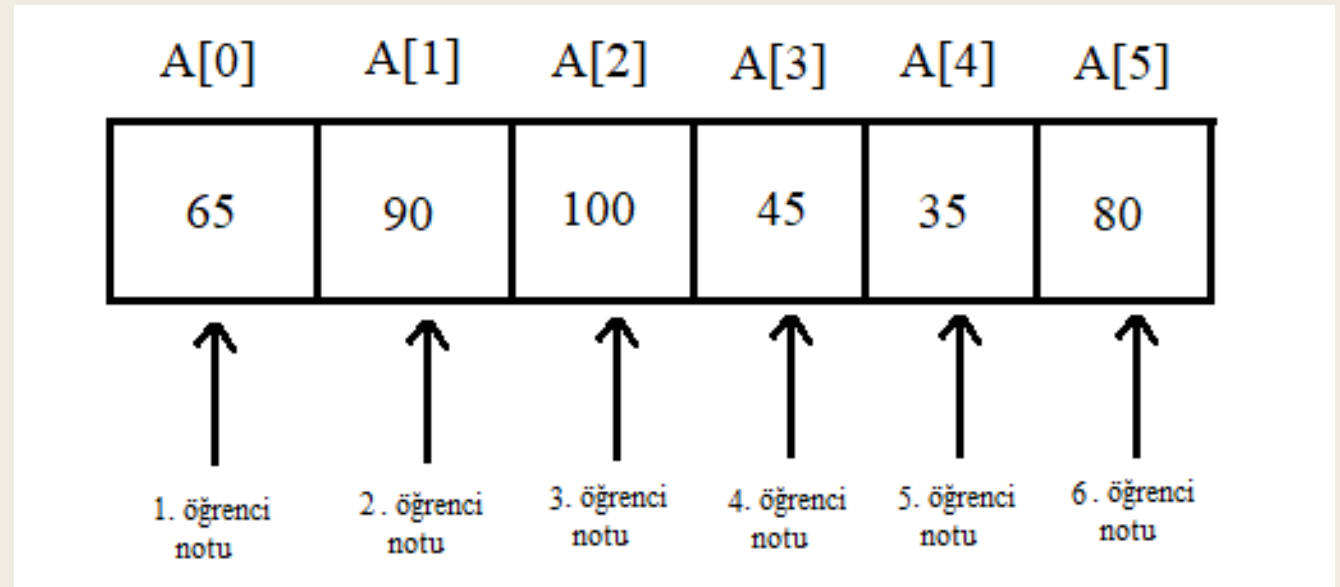
- Dizinin ismi **A** dır.

- Dizinin 6 elemanı

- $A[0], A[1], \dots, A[5]$  şeklindedir.

- Bu dizinin ilk üç elemanının içinde tutulan değerlerin toplamının sonucu:

- `printf("Sonuc: %d", A[0] + A[1] + A[2] );`
  - Sonuç: 255



# Uyarı:

Bu dizinin üçüncü elemanının değerini ikiye bölüp, oluşan sonucu s değişkenine atamak istersek aşağıdakişekilde yapmalıyız:

$$s = A[2] / 2;$$

Dizinin ” üçüncü elemanı” ile ”üçüncü dizi elemanı” arasındaki **farkı önemlidir.**

Dizi indisleri **0'dan başladığı** için “dizinin üçüncü elemanı” **2 indisine sahiptir.**

“**Üçüncü dizi elemanı**“ ise 3 indisine sahiptir ve aslında dizinin **dördüncü** elemanıdır.



# Dizilere Başlangıç Değeri Verme

--- Atama şeklinde:

```
int n[5] = { 1, 2, 3, 4, 5 };
```

```
double A[3]= { 1.756, 2.567, 3.123 };
```

```
float ort[6]= { 1.1, 2.1, 3.1, 4.1, 5.1, 6.1 };
```

- Küme parantezleri sonlandırıcı ; karakteri ile bitmektedir.

--- Eğer dizinin belirtilen boyutu kadar eleman yoksa geriye kalanlar sıfır olarak atanır.

`int A[5] = {1,2};`



A[0]	A[1]	A[2]	A[3]	A[4]
1	2	0	0	0

`int A[5] = {0};`



A[0]	A[1]	A[2]	A[3]	A[4]
0	0	0	0	0

- Metin tipindeki dizi elemanlarına NULL değeri atanır.

- Bir dizinin uzunluğu belirtilmeden de başlangıç değeri atamak mümkündür.
- **Örnek:**
  - `int A[ ] = { 10, 20, 30, 40, 50, 60, 70 };`
  - `float ort[ ] = { 9.1, 11.4, 7.2, 0.3, 12.6, 20.3};`
- Derleyici bu şekilde bir atama ile karşılaştığında, küme parantezi içindeki eleman sayısını hesaplar ve dizinin o uzunlukta açıldığını varsayar.
- Yukarıdaki örnekte, **A dizisinin 7** ve **ort dizisinin 6** elemanlı olduğu varsayılır.

# Dizilere Değer Atama ve Okuma:

## Örnek 1:

```
#include <stdio.h>
#include <conio.h>
int main()
{
    int n[ 10 ], i;
    for ( i = 0; i <= 9; i++ )
        n[ i ] = 0;
    printf( "%s%13s\n", "Eleman", "Deger" );
    for ( i = 0; i <= 9; i++ )
        printf( "%7d%13d\n", i, n[ i ] );
    getch();
    return 0;
}
```

Eleman	Deger
0	0
1	0
2	0
3	0
4	0
5	0
6	0
7	0
8	0
9	0

```
#include <stdio.h>
#include <conio.h>
int main()
{
    int n[ 10 ] = { 32, 27, 64, 18, 95, 14, 90, 70, 60, 37 };
    int i;
    printf( "%s%13s\n", "Eleman", "Deger" );
    for ( i = 0; i <= 9; i++ )
        printf( "%7d%13d\n", i, n[ i ] );
    getch();
    return 0;
}
```

Eleman	Deger
0	32
1	27
2	64
3	18
4	95
5	14
6	90
7	70
8	60
9	37

## Örnek 2:

int A[10]; // 10 elemanlı integer tipli A dizisi

int i = 3;

- A[0] = 20; // Geçerli atama
- A[3.5] = 5; // Geçersiz atama
- A[2\*i - 1] = 8; // Geçerli atama, A[5] dizi elemanına 8 değerini atar.
- A[i++]; // Önce A[3] dizi elemanına erişilir daha sonra değişkenine 4 değeri atanır.
- A[A[5]]; // A[8] dizi elemanına erişilir.

## Örnek 3:

```
#include<stdio.h>
#include<conio.h>
int main ()
{ int A[11],i,j;
for (i=0;i<=10;i++)
{ A[i]=i;}
j = 3;
A[10] = 20;
A[2*j-1]=9;
A[++j]=20;
A[A[5]]=0;
printf( "%s%14s\n", "indis", "Degeri" );
for (i=0;i<11;i++)
{printf( "%3d%16d\n", i, A[i] );
}
getch();
return 0;
}
```

```
indis      Degeri
0          0
1          1
2          2
3          3
4         20
5          9
6          6
7          7
8          8
9          0
10         20

-----
Process exited after 9.025 seconds with return value 0
Press any key to continue . . .
```

## Örnek 4:

Klavyeden sayısal değer atamak:

Bir sınıftaki 6 öğrencinin notunu A dizisinde saklayan C programını yazınız.

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
int main()
```

```
{ int A[6],i;
```

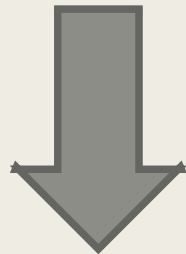
```
for (i=0;i<6;i++)
```

```
{
```

```
printf ("%d. ci ogrencinin notunu giriniz: ", i+1);
```

```
scanf("%d",&A[i]);
```

```
}
```



```
printf( "%s%14s\n", "Ogrenci", "Notu" );
```

```
for (i=0;i<6;i++)
```

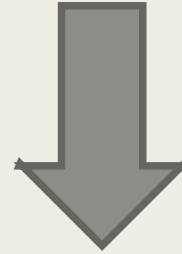
```
{printf( "%3d%16d\n", i+1, A[i] );
```

```
}
```

```
getch();
```

```
return 0;
```

```
}
```



```
1. ci ogrencinin notunu giriniz: 70
2. ci ogrencinin notunu giriniz: 90
3. ci ogrencinin notunu giriniz: 100
4. ci ogrencinin notunu giriniz: 55
5. ci ogrencinin notunu giriniz: 45
6. ci ogrencinin notunu giriniz: 85
```

Ogrenci	Notu
1	70
2	90
3	100
4	55
5	45
6	85

-----

```
Process exited after 15.14 seconds with return value 0
Press any key to continue . . .
```



Atama aşağıdaki şekilde yapıldı:

```
for (i=0;i<6;i++)
{
    printf ("%d. ci ogrencinin notunu giriniz: ", i+1);
    scanf("%d",&A[i]);
}
```

Atama aşağıdaki şekilde de yapılabilirdi:

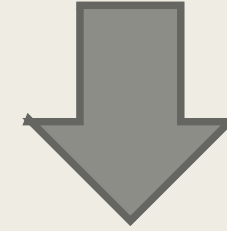
```
int x;
for (i=0;i<6;i++)
{
    printf ("%d. ci ogrencinin notunu giriniz: ", i+1);
    scanf("%d", &x);
    A[i]=x;
}
```

# Dizi Kullanımlarında Dikkat Edilmesi Gereken En Önemli Durumlar

- Dizi boyunca döngü kullanırken dizi indisi asla 0'ın altına inmemelidir ve her zaman dizideki toplam eleman sayısından az olmalıdır.
- Döngü devam şartının bu aralığın dışındaki elemanlara ulaşılmasını engellediğinden emin olmamız gereklidir.
- Dizi sınırlarının dışındaki elemanları kullanmanın yaratacağı hatalar (genelde ciddi hata olarak görülürler) sistemden sisteme farklılık gösterir.

## Örnekler:

1-) 1 den 50 ye kadar olan tek sayıları A isimli diziye çift sayıları B isimli diziye atayan bir C programı yazınız.



```
#include <stdio.h>
```

```
#include <conio.h>
```

```
int main()
```

```
{ int A[25],B[25],x,i=-1,j=-1,k;
```

```
for (k=1;k<=50;k++)
```

```
{
```

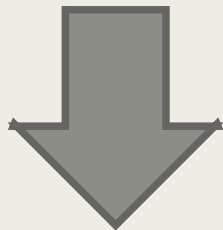
```
if (k%2==1) { i++;
```

```
    A[i]=k;}
```

```
else {j++;
```

```
    B[j]=k;}
```

```
}
```



```
printf ("Tek Sayilar    Cift Sayilar  \n");
```

```
for (i=0;i<25;i++)
```

```
{
```

```
printf ("%4d%16d\n",A[i],B[i]);
```

```
}
```

```
getch();
```

```
return 0;
```

```
}
```

```
Tek Sayilar      Cift Sayilar
1                2
3                4
5                6
7                8
9                10
11               12
13               14
15               16
17               18
19               20
21               22
23               24
25               26
27               28
29               30
31               32
33               34
35               36
37               38
39               40
41               42
43               44
45               46
47               48
49               50

-----
Process exited after 3.202 seconds with return value 0
Press any key to continue . . .
```

2-) 5 elemanlı bir A dizisi oluşturunuz. A dizisinin ortalamasını bulunuz.

```
#include<stdio.h>
#include<conio.h>
int main ()
{ float A[5],x,top=0,ort; int i;

for (i=0;i<=4;i++)
{
printf ("%d. sayiyi giriniz:",i+1);
scanf ("%f",&A[i]);
top=top+A[i];
}

ort=top/5.0;
printf( "%s%13s\n", "Eleman", "Deger A" );
    for ( i = 0; i <= 4; i++ )
        printf( "%4d%13.0f\n", i+1, A[i] );
printf ("A dizisinin elemanlarinin ortalamasi=%.2f", ort);

getch();
return 0;
}
```

```
1. sayiyi giriniz:45
2. sayiyi giriniz:25
3. sayiyi giriniz:40
4. sayiyi giriniz:12
5. sayiyi giriniz:90
Eleman      Deger A
   1         45
   2         25
   3         40
   4         12
   5         90
A dizisinin elemanlarinin ortalamasi=42.40
-----
Process exited after 19.95 seconds with return value 0
Press any key to continue . . .
```

3-) Klavyeden 5 adet tamsayı giriliyor.  
Klavyeden girilen sayının adeti  
kadar \* simgesi koyan bir  
C programı yazınız.

```
1. degeri giriniz: 4
2. degeri giriniz: 8
3. degeri giriniz: 13
4. degeri giriniz: 1
5. degeri giriniz: 15
Eleman      Deger      Histogram
1           4          ****
2           8          ****
3          13          ****
4           1          *
5          15          ****

-----
Process exited after 21.47 seconds with return value 0
Press any key to continue . . .
```

```
#include <stdio.h>
#include <conio.h>
#define BOYUT 5
int main()
{int n[BOYUT],i,j ;
  for ( i = 0; i <= BOYUT-1; i++ ) {
    printf( "%d. degeri giriniz: ",i+1) ;
    scanf("%d",&n[i]);}
  printf( "%s%13s%17s\n", "Eleman", "Deger",
"Histogram" );
  for ( i = 0; i <= BOYUT-1; i++ )
  {
    printf( "%3d%14d      ", i+1, n[i]) ;
    for ( j = 1; j <= n[i]; j++ )
      printf( "%c", '*' );
    printf( "\n" );
  }
  getch();
  return 0;
}
```

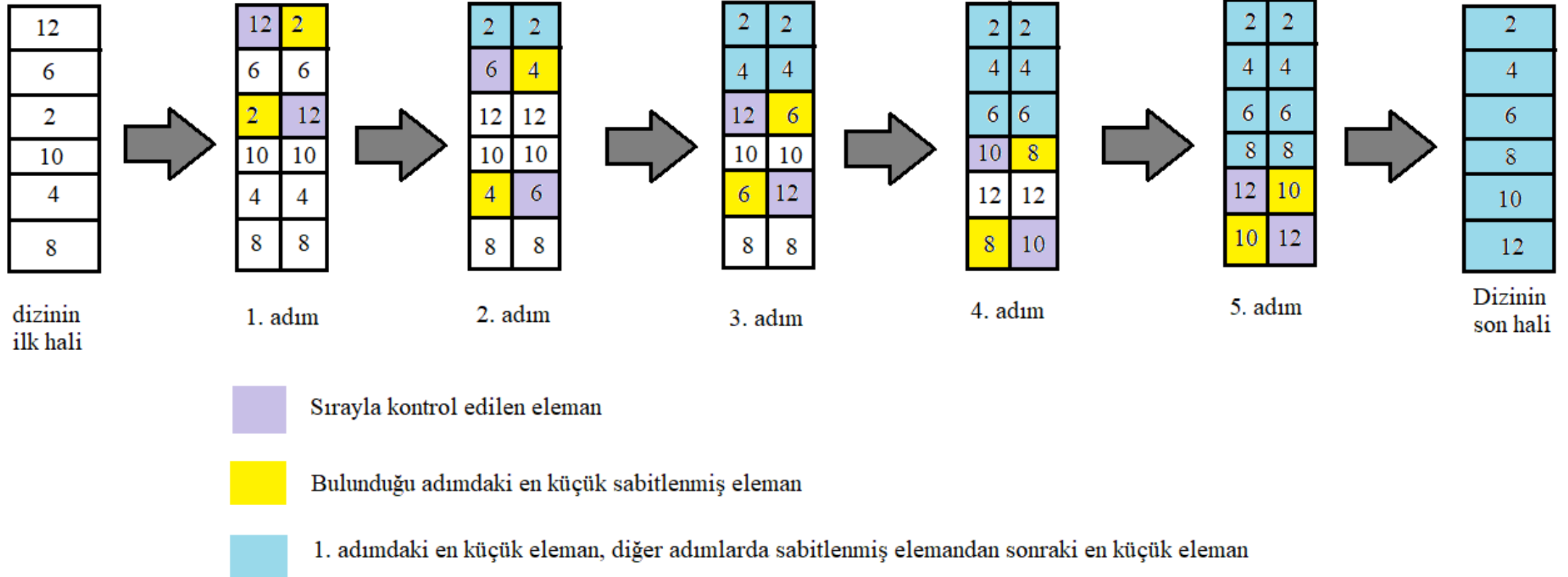
# Dizileri Sıralamak

- Veri sıralamak önemli bilgisayar uygulamalarından biridir.
- Bir çok günlük hayat probleminde bazı verileri sıralamak zorunda kalabiliriz.
- En çok bilinen sıralama algoritmalarında biri **Seçmeli Sıralamadır.**

## Seçmeli sıralama

1. Listedeki en küçük değerli veri bulunur.
2. İlk konumdaki öğeyle bulunan en küçük değerli öğenin yeri değiştirilir.
3. Yukarıdaki adımlar listenin ilk elemanından sonrası için (ikinci elemandan başlayarak) yinelenir.


# Seçmeli Sıralama





## Örnek:

```
#include <stdio.h>
#include <conio.h>
#define BOYUT 4
int main()
{
    int n[BOYUT],y[BOYUT],i,j,a;
    for ( i = 0; i <= BOYUT-1; i++ )
    {printf ("%d. sayiyi giriniz: ",i+1);
      scanf ("%d",&n[i]);
    }
    for ( i = 0; i <= BOYUT-1; i++ )
        {y[i]=n[i];}
    printf ("dizi elemanlari\n");
    for ( i = 0; i <= BOYUT-1; i++ )
        printf ("%d ",n[i]);
    printf("\n\n");
```



```
printf ("dizi elemanlari buyukten kucuge sirali\n");
for (i = 0; i < BOYUT-1; i++)
    {for ( j = i + 1; j < BOYUT; j++)
        if (n[i] < n[j])
            {a = n[i]; n[i]=n[j]; n[j]=a;}
    }
for ( i = 0; i <= BOYUT-1; i++ )
    printf ("%d ",n[i]);
printf("\n\n");
printf ("dizi elemanlari kucukten buyuge sirali\n");
for (i = 0; i < BOYUT-1; i++)
    {for ( j = i + 1; j < BOYUT; j++)
        if (n[i] > n[j])
            {a = n[i]; n[i]=n[j]; n[j]=a;}
    }
for ( i = 0; i <= BOYUT-1; i++ )
    printf ("%d ",n[i]);
printf("\n");

return 0;
getch();
}
```

1. sayiyi giriniz: 45

2. sayiyi giriniz: 35

3. sayiyi giriniz: 20

4. sayiyi giriniz: 90

dizi elemanlari

45 35 20 90

dizi elemanlari buyukten kucuge sirali

90 45 35 20

dizi elemanlari kucukten buyuge sirali

20 35 45 90

-----

Process exited after 9.278 seconds with return value 0

Press any key to continue . . .

# Dizilerde Arama Yapmak

Bir dizide belirli bir arama değerine eşit bir değer olup olmadığının belirlenmesidir.

## Lineer arama

- *Basit*
- *Dizinin her elemanını arama değeriyle karşılaştırır.*
- *Küçük ya da sıralanmamış dizilerde iyi çalışır.*

**Örnek-)** 0 ile 200 arasındaki çift sayılar a dizisinde saklanıyor. Klavyeden girilen bir tamsayının bu dizide yer alıp almadığını bulan ve sonucu ekrana yazdıran bir C programı yazınız.

```
#include <stdio.h>
#include <conio.h>
#define BOYUT 101
int main()
{int a[BOYUT],i,key,s=0;
for(i=0;i<BOYUT;i++)
    a[i]=2*i;
printf("Aranacak degeri giriniz: ");
scanf("%d",&key);

    for(i=0;i<BOYUT;i++)
        {if(a[i]==key) {printf("Bu deger bulundu, dizinin %d.
elemanidir, indisi= %d dir. \n",i+1,i);
            s=1;
            break;}}
    if (s==0) printf("Bu deger bulunamadi\n");
    getch();
    return 0;
}
```

```
Aranacak degeri giriniz: 35
Bu deger bulunamadi

-----
Process exited after 11.87 seconds with return value 0
Press any key to continue . . .
```

```
Aranacak degeri giriniz: 50
Bu deger bulundu, dizinin 26. elemanidir, indisi= 25 dir.

-----
Process exited after 4.868 seconds with return value 0
Press any key to continue . . .
```

```
Aranacak degeri giriniz: 200
Bu deger bulundu, dizinin 101. elemanidir, indisi= 100 dir.

-----
Process exited after 4.082 seconds with return value 0
Press any key to continue . . .
```

```
Aranacak degeri giriniz: 0
Bu deger bulundu, dizinin 1. elemanidir, indisi= 0 dir.

-----
Process exited after 2.705 seconds with return value 0
Press any key to continue . . .
```

# ÇEŞİTLİ ÖRNEKLER

**Örnek-1)** Klavyeden girilen 5 elemanlı A ve B dizilerini toplayarak C dizisini oluşturan ve yazdıran C programını yazınız.

- $A = (a_1, a_2, \dots, a_n)$

- $B = (b_1, b_2, \dots, b_n)$

- $C = A + B = (c_1, c_2, \dots, c_n) = (a_1 + b_1, a_2 + b_2, \dots, a_n + b_n)$

```

#include <stdio.h>
#include <conio.h>
int main()
{int A[5],B[5],C[5],i;
 printf("A dizisinin elemanlarini giriniz:\n");
 for(i=0;i<5;i++)
 {printf("%d. elemani giriniz: ",i+1);
  scanf("%d",&A[i]);
 }

 printf("B dizisinin elemanlarini giriniz: \n");
 for(i=0;i<5;i++)
 {printf("%d. elemani giriniz: ",i+1);
  scanf("%d",&B[i]);
 }

 for(i=0;i<5;i++)
  C[i]=A[i]+B[i];

 printf("C dizisinin elemanlari asagidadir: \n");
 for(i=0;i<5;i++)
  printf("C dizisinin %d. elemani: %d \n",i+1,C[i]);
 getch();
 return 0;
}

```

```

A dizisinin elemanlarini giriniz:
1. elemani giriniz: 2
2. elemani giriniz: 5
3. elemani giriniz: 8
4. elemani giriniz: 9
5. elemani giriniz: 3
B dizisinin elemanlarini giriniz:
1. elemani giriniz: 2
2. elemani giriniz: 6
3. elemani giriniz: 8
4. elemani giriniz: 9
5. elemani giriniz: 1
C dizisinin elemanlari asagidadir:
C dizisinin 1. elemani: 4
C dizisinin 2. elemani: 11
C dizisinin 3. elemani: 16
C dizisinin 4. elemani: 18
C dizisinin 5. elemani: 4

-----
Process exited after 25.83 seconds with return value 0
Press any key to continue . . .

```

**Örnek-2)** Klavyeden float tipinde 10 adet değer giriliyor. Bu değerlerin ortalamasını bulan, ve kaç adet elemanın ortalamasının altında kaldığı ve kaç adet elemanın ortalamasının üstünde olduğunu belirleyen bir program hazırlayınız.

```
#include<stdio.h>
#include<conio.h>
int main()
{ float dizi[ 10 ];
  float ortalama, toplam = 0;
  int xu = 0;
  int xa = 0, xe=0;
  int i;
  for( i = 0; i < 10; i++ ) {
    printf( "%d. elemani giriniz >", (i+1) );
    scanf( "%f", &dizi[ i ] );
    toplam += dizi[ i ];}
  ortalama = toplam / 10.0;
```





```
for( i = 0; i < 10; i++ ) {  
    if ( dizi[ i ] < ortalama )  
        xa ++;  
    else if ( dizi[ i ] > ortalama )  
        xu ++;  
    else xe++;}  
printf( "Ortalama: %.2f\n", ortalama );  
printf( "Ortalamadan dusuk %d eleman vardır.\n", xa );  
printf( "Ortalamadan yuksek %d eleman vardır.\n", xu );  
printf( "Ortalamaya esit %d eleman vardır.\n", xe );  
getch();  
return 0; }
```



```
1. elemani giriniz >5
2. elemani giriniz >5
3. elemani giriniz >5
4. elemani giriniz >5
5. elemani giriniz >5
6. elemani giriniz >6
7. elemani giriniz >7
8. elemani giriniz >2
9. elemani giriniz >8
10. elemani giriniz >2
Ortalama: 5.00
Ortalamadan dusuk 2 eleman vardır.
Ortalamadan yuksek 3 eleman vardır.
Ortalamaya esit 5 eleman vardır.

-----
Process exited after 29.21 seconds with return value 0
Press any key to continue . . .
```

```
1. elemani giriniz >3
2. elemani giriniz >6
3. elemani giriniz >7.8
4. elemani giriniz >9
5. elemani giriniz >3.8
6. elemani giriniz >5.1
7. elemani giriniz >8.7
8. elemani giriniz >3
9. elemani giriniz >9.6
10. elemani giriniz >18.4
Ortalama: 7.44
Ortalamadan dusuk 5 eleman vardır.
Ortalamadan yuksek 5 eleman vardır.
Ortalamaya esit 0 eleman vardır.

-----
Process exited after 43.13 seconds with return value 0
Press any key to continue . . .
```

**Örnek-3)** Bir zar 10 kez atılıyor. Her bir rakamın gelme yüzdeleri ekrana yazdıran bir C programı yazınız.

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
int main ()
```

```
{ int A[6],x,i,y;
```

```
for (i=0;i<6;i++)
```

```
{ A[i]=0;}
```

```
for (i=0;i<10;i++)
```

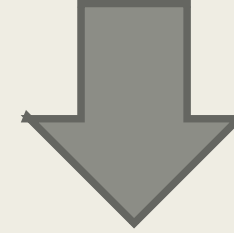
```
{
```

```
printf ("%d. atistaki degeri giriniz: ",i+1);
```

```
scanf("%d",&x);
```

```
A[x-1]=A[x-1]+1;
```

```
}
```



```
printf ("Zardaki sayilarin gelme yuzdeleri: \n");
```

```
for (i=0;i<6;i++)
```

```
{
```

```
y=(A[i]*100)/10;
```

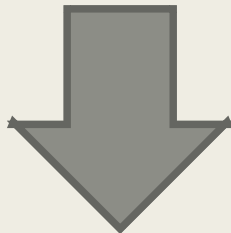
```
printf ("%d. degerinin gelme yuzdesi= %% %d \n",i+1,y);
```

```
}
```

```
getch();
```

```
return 0;
```

```
}
```



```
1. atistaki degeri giriniz: 1
2. atistaki degeri giriniz: 3
3. atistaki degeri giriniz: 2
4. atistaki degeri giriniz: 4
5. atistaki degeri giriniz: 5
6. atistaki degeri giriniz: 4
7. atistaki degeri giriniz: 3
8. atistaki degeri giriniz: 2
9. atistaki degeri giriniz: 2
10. atistaki degeri giriniz: 2
Zardaki sayilarin gelme yuzdeleri:
1. degerinin gelme yuzdesi= % 10
2. degerinin gelme yuzdesi= % 40
3. degerinin gelme yuzdesi= % 20
4. degerinin gelme yuzdesi= % 20
5. degerinin gelme yuzdesi= % 10
6. degerinin gelme yuzdesi= % 0
```

-----

```
Process exited after 12.48 seconds with return value 0
Press any key to continue . . .
```

# Kaynaklar

- C: How to Program Third Edition Harvey M. Deitel ; Paul J. Deitel.
- C Programlama Dili Dr. Rıfat Çölkesen Papatya Yayıncılık.
- Problem Solving and Program Design in C, 7/E Jeri R. Hanly; Elliot B. Koffman.
- C Programlama dili; İbrahim Güney; Nobel Yayıncılık.
- Algoritma Geliştirme ve Programlamaya Giriş, Fahri Vatansever, Seçkin yayıncılık
- C Programlama Ders Notları, A. Kadir YALDIR, Pamukkale Üniversitesi ders notları.