

CENG 111 ALGORİTMALAR VE PROGRAMLAMA

Doç. Dr. Tufan TURACI

tturaci@pau.edu.tr

- Pamukkale Üniversitesi
- Mühendislik Fakültesi
- Bilgisayar Mühendisliği Bölümü
- Hafta 10
- 22 Kasım 2022

KARAKTER TIPLİ DİZİLER

- Karakter Tipli Diziler
 - Karakter Tipli Dizileri Okumak ve Yazmak
 - Karakter Tipli Dizilerin Uzunluğunun Bulunması
 - Karakter Tipli Dizileri Birleştirmek
 - Karakter Tipli Dizileri Kopyalamak
 - Karakter Tipli Dizileri Karşılaştırmak

Karakter Tipli Dizileri Okumak ve Yazmak

- Bazı programlama dillerinde karakter dizilerini tutmak için özel veri türleri (string, vb.) bulunmaktadır. Fakat C programlama dilinde böyle bir veri türü olmadığı için yerine karakterlerden oluşan bir boyutlu diziler kullanılır.
- Karakter dizilerine özel olarak, karakter dizilerinin sonuna sonlandırıcı karakter olarak adlandırılan bir simge eklenir.
- Sonlandırıcı karakter:
 - *Dizinin bittiği yeri gösterir.*
 - *ASCII tablosunun sıfır numaralı (**\0**) karakteridir.*
 - *Nul (Null) olarakta ifade edilir.*

- ASCII, American Standard Code for Information Interchange'in kısaltmasıdır.
- 7 bitlik bir karakter seti olan ASCII, Latin alfabesi üzerine kuruludur.
- İlk olarak ANSI tarafından 1963 yılında standart olarak sunulmuştur.
- Örneğin, büyük A harfinin ASCII kodu 65' dir.
- Artı (+) nın ASCII kodu 43' dür.

Karakter Tipli Dizilere Başlangıç Değeri Verilmesi:

- Karakter tipli dizilere iki şekilde başlangıç değeri verilebilir:

`char A[10]= {'a','l','g','o','r','i','t','m','a','\0'},`

`B[11]= {"bilgisayar"},`

`C[10]= {"pamukkale"};`

- A dizisindeki durumda sonlandırıcı karakter (`\0`) konmalıdır.
- Diğer durumlarda buna gerek yoktur. Çünkü, sonlandırıcı karakter derleyici tarafından eklenir.

Karakter Tipli Dizilere Klavyeden Değer Verilmesi ve Ekrana Yazdırılması:

1. yol:

--- Bir karakter dizisini klavyeden okumak için C'nin standart gets() fonksiyonu kullanılabilir.

--- gets() fonksiyonu için «stdio.h» dışında yeni bir kitaplığa ihtiyaç yoktur.

Kullanımı: gets(dizi_adı)

--- Bu fonksiyon herhangi bir indeks tanımlamadan karakter dizilerinin okunmasını sağlar.

--- printf() fonksiyonu ile direk olarak dizideki string ifade yazdırılabilir.

Kullanımı: printf(dizi_adı)

Örnek: Klavyeden girilen bir metni ekrana yazdıran C programını yazınız.

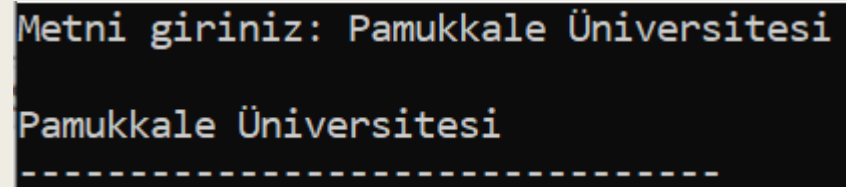
```
#include <stdio.h>
#include <conio.h>
```

```
int main()
{ char A[50];

printf("Metni giriniz: ");
gets(A);

printf("\n");
printf(A);

getch();
return 0;
}
```



```
Metni giriniz: Pamukkale Üniversitesi
Pamukkale Üniversitesi
-----
```

Klavyeden girilen bir metni ekrana farklı şekillerde yazdırılması:

```
#include <stdio.h>
#include <conio.h>
int main()
{ char A[50]; int i;
printf("Metni giriniz: ");
gets(A);
printf("\nprintf kullanarak:\n");
printf(A);
printf("\nfor kullanarak(1.durum):\n");
for(i=0;A[i]!='\0';i++)
{ printf ("%c",A[i]); }
printf("\nfor kullanarak(2.durum):\n"); // türkçe karakter
kullanıldığında tüm metni yazdırmayabilir.
for (i=0;i<=A[i];i++)
{ printf ("%c",A[i]); }
getch();
return 0;
}
```

```
Metni giriniz: Bilgisayar Muhendisligi

printf kullanarak:
Bilgisayar Muhendisligi
for kullanarak(1.durum):
Bilgisayar Muhendisligi
for kullanarak(2.durum):
Bilgisayar Muhendisligi
-----
```

```
Metni giriniz: Bilgisayar Mühendisliği

printf kullanarak:
Bilgisayar Mühendisliği
for kullanarak(1.durum):
Bilgisayar Mühendisliği
for kullanarak(2.durum):
Bilgisayar M
-----
```


2. yol:

--- scanf komutu ile bir kelime karakter tipli diziye aktarılır.

Örnek:

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

```
int main()
{ char A[50]; int i;
```

```
printf("Metni giriniz: ");
scanf ("%s",A);
```

```
printf("scanf ile girilen karakter tipli dizi:\n");
printf(A);
```

```
getch();
return 0;
}
```

```
Metni giriniz: Bilgisayar
scanf ile girilen karakter tipli dizi:
Bilgisayar
-----
```

```
Metni giriniz: Bilgisayar Mühendisliği
scanf ile girilen karakter tipli dizi:
Bilgisayar
-----
```

Tek kelimeyi yazdırdı...

Matris Kullanarak birden farklı kelime saklanabilir.

Örnek: Birden fazla kelimenin `scanf` ile matrise aktarılması:

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

```
int main()
{char isim[4][30];
  int i;
```

```
  for (i=0; i<4; i++)
    {printf ("%d. ismi giriniz:",i+1);
      scanf ("%s", isim[i]);}
```

```
  for (i=0; i<4; i++)
    { printf ("%s ", isim[i]); }
```

```
  getch();
  return 0;
}
```

```
1. ismi giriniz:Pamukkale
2. ismi giriniz:Üniversitesi
3. ismi giriniz:Bilgisayar
4. ismi giriniz:Mühendisliği
Pamukkale Üniversitesi Bilgisayar Mühendisliği
-----
```

Karakter Tipli Dizilerin Uzunluğunun Bulunması

1. yol:

--- strlen() fonksiyonu kullanarak dizinin uzunluğu bulunabilir.

Kullanımı: strlen(dizi_adı)

--- Uzunluk bulunurken, içerdiği en son karakter olan **NULL** (boş karakter) hesaplanmaz.

--- **#include <string.h>** ile kullanılır.

Örnek:

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

```
int main()
{ char A[50];
```

```
printf("Metni giriniz: ");
gets(A);
```

```
printf("Girilen karakterin uzunlugu: %d",strlen(A));
```

```
getch();
return 0;
}
```

```
Metni giriniz: Bilgisayar
Girilen metnin uzunlugu: 10
-----
```

```
Metni giriniz: Bilgisayar Mühendisliği
Girilen metnin uzunlugu: 23
-----
```

2. yol:

--- Sayaç koyarak karakterleri sayabiliriz.

Örnek:

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

```
int main()
{ char A[50];int i,top=0;
```

```
printf("Metni giriniz: ");
gets(A);
```

```
for(i=0;A[i]!='\0';i++)
{ top++; }
```

```
printf("Girilen metnin uzunlugu: %d",top);
```

```
getch();
return 0;
}
```

```
Metni giriniz: Bilgisayar
Girilen metnin uzunlugu: 10
-----
```

```
Metni giriniz: Bilgisayar Mühendisliği
Girilen metnin uzunlugu: 23
-----
```

Örnek: Klavyeden girilen bir kelimeyi tersten yazdıran bir C programı yazınız.

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

```
int main()
{ char A[50];int i,top=0;

printf("Metni giriniz: ");
gets(A);

for(i=0;A[i]!='\0';i++)
{ top++; }

printf("Girilen metnin tersi: ");

for(i=top-1;i>=0;i--)
{ printf("%c",A[i]);}

getch();
return 0;
}
```

```
Metni giriniz: PAMUKKALE
Girilen metnin tersi: ELAKKUMAP
-----
```

```
Metni giriniz: bilgisayar mühendisliği
Girilen metnin tersi: iğilsidnehüm rayasıglıb
-----
```

Örnek: Klavyeden girilen bir kelimeyi dikey olarak yazdıran bir C programı yazınız.

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

```
int main()
{ char A[50];int i,top=0;

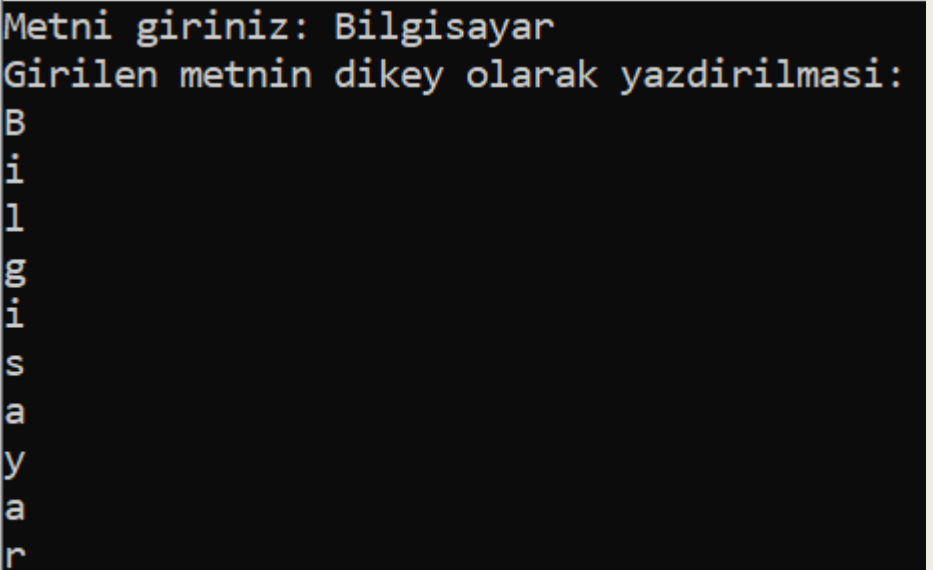
printf("Metni giriniz: ");
gets(A);

for(i=0;A[i]!='\0';i++)
{ top++; }

printf("Girilen metnin dikey olarak yazdirilmesi: \n");

for(i=0;i<top;i++)
{ printf("%c \n",A[i]);}

getch();
return 0;
}
```



```
Metni giriniz: Bilgisayar
Girilen metnin dikey olarak yazdirilmesi:
B
i
l
g
i
s
a
y
a
r
```

Karakter Tipli Dizileri Birleştirmek

---İki tane karakter tipli diziyi birleştirilerek tek bir karakter tipli dizi haline dönüştürülebilir.

---Birleştirme işlemi C' nin `strcat()` fonksiyonu ile yapılır.

---`strcat()` fonksiyonu, var olan bir karakter tipli dizinin sonuna başka bir karakter tipli dizi eklenmesini sağlar.

--- Örneğin "bilgisayar" karakter dizisinin sonuna "mühendisliği" karakter dizisi `strcat()` fonksiyonu kullanılarak eklenebilir.

--- `#include <string.h>` ile kullanılır.

Örnek: Klavyeden girilen iki farklı karakter tipli diziyi birleştiren bir C programı yazınız.

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

```
int main()
{ char A[50],B[50];

printf("1. Metni giriniz: ");
gets(A);

printf("2. Metni giriniz: ");
gets(B);

strcat(A, B);
printf("Birlestirilmis Metin: ");
printf(A);

getch();
return 0;
}
```

```
1. Metni giriniz: Pamukkale
2. Metni giriniz: Üniversitesi
Birlestirilmis Metin: Pamukkale Üniversitesi
-----
```

Örnek: Klavyeden girilen üç farklı karakter tipli diziyi birleştiren bir C programı yazınız.

```
#include <stdio.h>
#include <conio.h>
#include <string.h>
int main()
{ char A[50],B[50],C[50];
printf("1. Metni giriniz: ");
gets(A);
printf("2. Metni giriniz: ");
gets(B);
printf("3. Metni giriniz: ");
gets(C);
strcat(B,C);
strcat(A,B);
printf("Birlestirilmis Metin: ");
printf(A);
getch();
return 0;
}
```

```
1. Metni giriniz: Denizli
2. Metni giriniz: ege bölgesinin
3. Metni giriniz: bir ilidir.
Birlestirilmis Metin: Denizli ege bölgesinin bir ilidir.
-----
```

Karakter Tipli Dizileri Kopyalamak

---Karakter dizilerine direk atama yapılamamaktadır.

---Aşağıdaki şekilde atama ifadesi yapılamaz!

```
char dizi[100];  
dizi = "algoritma";
```

---Çünkü bu atama göstergeye yapılan atamadır.

--- C'nin standart strcpy() fonksiyonu kullanılarak, karakter dizisine başlangıç değeri ataması yapılabilir.

--- `#include <string.h>` ile kullanılır.

Örnek: strcpy fonksiyonunu kullanan bir C programı yazınız.

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

```
int main()
{ char A[50],B[50];
strcpy(A, "Pamukkale ");
strcpy(B, "Universitesi");
printf("A dizisi: ");
printf(A);
printf("\n");
printf("B dizisi: ");
printf(B);
printf("\n");
strcat(A,B);
printf("Birlestirilmis Metin: ");
printf(A);
getch();
return 0;
}
```

```
A dizisi: Pamukkale
B dizisi: Universitesi
Birlestirilmis Metin: Pamukkale Universitesi
-----
```

Karakter Tipli Dizileri Karşılaştırmak

--- İki karakter tipli diziyi birbirleriyle karşılaştırarak, içerdiği karakterlerin aynı olup olmadıkları **strcmp()** fonksiyonu kullanılarak kontrol edilebilir.

--- Karşılaştırma sonucunda, her iki karakter dizisi birbirinin aynı ise "0"; birinci dizi ikinciden küçükse " negatif bir değer" ve ikinci dizi birinciden küçükse " pozitif bir değer" üretilir.

--- `#include <string.h>` ile kullanılır.

Örnek: strcmp() fonksiyonunu kullanan bir C programı yazınız.

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

```
int main()
{ char A[50],B[50]; int x;
printf("1. Metni giriniz: ");
gets(A);
printf("2. Metni giriniz: ");
gets(B);
x=strcmp(A,B);
printf("x= %d\n",x);
if (x==0) printf("2 metin birbirinin aynisidir...\n");
    else printf("2 metin birbirinden farklıdır...\n");
getch();
return 0;
}
```

```
1. Metni giriniz: Algoritma
2. Metni giriniz: Algoritma
x= 0
2 metin birbirinin aynisidir...
```

```
1. Metni giriniz: Bilgisayar
2. Metni giriniz: Algoritma
x= 1
2 metin birbirinden farklıdır...
```

```
1. Metni giriniz: Algoritma
2. Metni giriniz: Bilgisayar
x= -1
2 metin birbirinden farklıdır...
```

strrev() fonsiyonu

--- Karakter tipli bir diziyi tersten yazdırmak için kullanılır.

--- `#include <string.h>` ile kullanılır.

Örnek:

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

```
int main()
{ char A[50],B[50];
printf("Metni giriniz: ");
gets(A);
printf("Girilen metnin tersi: %s\n",strrev(A));
getch();
return 0;
}
```

```
Metni giriniz: bilgisayar
Girilen metnin tersi: rayasiglib
```

```
-----
```

toupper() ve tolower () fonksiyonları

--- Karakter tipli bir dizilerde toupper() fonksiyonu küçük harfleri büyük harfe çevirir.

tolower() fonksiyonu büyük harfleri küçük harfe çevirir.

---Türkçe karakterler için çevirme işlemi yapılmıyor...

--- `#include <ctype.h>` ile kullanılır. (Standart C kütüphanesi `ctype.h` başlık dosyasında, **Karakterle** ilgili fonksiyonlar yer alır.)

Örnek: toupper() fonksiyonunu kullanan bir C programı yazınız.

```
#include <stdio.h>
#include <conio.h>
#include <string.h>
#include <ctype.h>
```

```
int main()
{ char A[50],B[50]; int i;
printf("Metni giriniz: ");
gets(A);
printf("Girilen metin buyuk harflerle yazildi: ");
```

```
for (i=0;i<strlen(A);i++)
    B[i]=toupper(A[i]);
```

```
printf(B);
getch();
return 0;
}
```

```
Metni giriniz: pamukkale
Girilen metin buyuk harflerle yazildi: PAMUKKALE
-----
```

```
Metni giriniz: çarşı
Girilen metin buyuk harflerle yazildi: ÇARŞI
-----
```

Örnek: tolower() fonksiyonunu kullanan bir C programı yazınız.

```
#include <stdio.h>
#include <conio.h>
#include <string.h>
#include <ctype.h>
```

```
int main()
{ char A[50],B[50]; int i;
printf("Metni giriniz: ");
gets(A);
printf("Girilen metin kucuk harflerle yazildi: ");

for (i=0;i<strlen(A);i++)
    B[i]=tolower(A[i]);

printf(B);
getch();
return 0;
}
```

```
Metni giriniz: PAMUKKALE
Girilen metin kucuk harflerle yazildi: pamukkale
-----
```

```
Metni giriniz: ÇARŞI
Girilen metin kucuk harflerle yazildi: Çarşı
-----
```

Örnek: Bir sınıftaki 5 öğrencinin isimleri, numaraları ve vize sınavından aldığı notlar A, B ve C isimli dizilerde saklanıyor.

--- Öğrencileri numaralarına göre küçükten büyüğe listeleyen,

--- Öğrenci isimlerini **alfabetik** olarak listeleyen

bir C programı yazınız.

```
1. ogrencinin ismini giriniz: can
1. ogrencinin numarasini giriniz: 2053
1. ogrencinin notunu giriniz: 80
2. ogrencinin ismini giriniz: kemal
2. ogrencinin numarasini giriniz: 2078
2. ogrencinin notunu giriniz: 68
3. ogrencinin ismini giriniz: veli
3. ogrencinin numarasini giriniz: 2044
3. ogrencinin notunu giriniz: 55
4. ogrencinin ismini giriniz: inci
4. ogrencinin numarasini giriniz: 2012
4. ogrencinin notunu giriniz: 95
5. ogrencinin ismini giriniz: hayat
5. ogrencinin numarasini giriniz: 2018
5. ogrencinin notunu giriniz: 90
```

Isim	Numara	Not

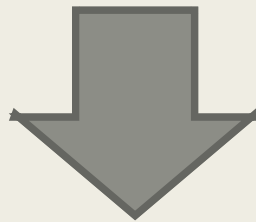
inci	2012	95
hayat	2018	90
veli	2044	55
can	2053	80
kemal	2078	68
Isimler Alfabetik Olarak Sirali		
Isim	Numara	Not

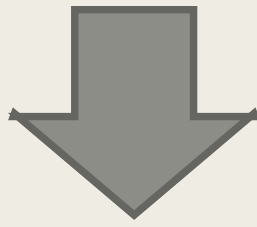
can	2053	80
hayat	2018	90
inci	2012	95
kemal	2078	68
veli	2044	55

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

```
int main()
{ int    B[5],C[5],i,t,m,j; char A[5][10],k[10];

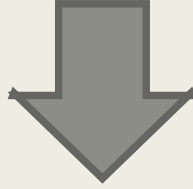
for (i=0;i<5;i++)
{ printf ("%d. ogrencinin ismini giriniz: ", i+1);
  scanf("%s",A[i]);
  printf ("%d. ogrencinin numarasini giriniz: ", i+1);
  scanf("%d",&B[i]);
  printf ("%d. ogrencinin notunu giriniz: ", i+1);
  scanf("%d",&C[i]);
}
```





```
for (i=0;i<4;i++)
{ for (j=i+1;j<5;j++)
  { if (B[i]>B[j]) {      strcpy(k, A[i]);
                        strcpy(A[i], A[j]);
                        strcpy(A[j], k);
                        m=B[i]; B[i]=B[j]; B[j]=m;
                        t=C[i]; C[i]=C[j]; C[j]=t;
                        }
  }
}

printf("Numara Sirasina Gore Kucukten Buyuge
Sirali\n");
printf("Isim          Numara          Not");
printf("\n-----\n");
for (i=0;i<5;i++)
{ printf("%s\t\t\t%d\t\t\t%d\n",A[i],B[i],C[i]);
}
```



```
for (i=0;i<4;i++)
{ for (j=i+1;j<5;j++)
    { if (strcmp(A[i], A[j]) > 0)
        { strcpy(k, A[i]);
          strcpy(A[i], A[j]);
          strcpy(A[j], k);
          m=B[i]; B[i]=B[j]; B[j]=m;
          t=C[i]; C[i]=C[j]; C[j]=t;
        }
    }
}
printf("\nIsimler Alfabetik Olarak Sirali\n");
printf("Isim          Numara          Not");
printf("\n-----\n");
for (i=0;i<5;i++)
{ printf("%s\t\t\t%d\t\t\t%d\n",A[i],B[i],C[i]);
}
getch ();
return 0;
}
```

Rastgele Sayılar Üretme

--- rand fonksiyonu ile rastgele sayılar üretilebilir.

--- 1 ile n arasında rasgele sayı üretebilmek için

$1 + (\text{rand}() \% n)$

- ✓ $\text{Rand}() \% n$ 0 ile $n-1$ arasında bir sayı döndürür.
- ✓ Buna 1 eklemek 1 ile n arasında bir rasgele sayı üretir.

Örnek:

$1 + (\text{rand}() \% 6)$

1 ile 6 arasında sayı üretir.

$\text{rand}() \% 6$

0 ile 5 arasında sayı üretir.

Örnek: 1 ile 20 arasında 5 kere 10 elemanlı sayı üreten bir C programı yazınız.

```
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
int main()
{ int A[10], i, x, j;
  for(j=0;j<5;j++)
  {
    for (i=0;i<10;i++)
    {x=1+(rand()%20);
     A[i]=x;}
    printf ("%d. adimda uretilen Sayilar\n",j+1);
    for (i=0;i<10;i++)
    printf ("%d  ",A[i]);
    printf ("\n\n");
  }
  getch();
  return 0;
}
```

Her çalıştırmada aynı sayılar üretilir

1. Çalıştırma sonucu

```
1. adimda uretilen Sayilar
2  8 15  1 10  5 19 19  3  5

2. adimda uretilen Sayilar
6  6  2  8  2 12 16  3  8 17

3. adimda uretilen Sayilar
12  5  3 14 13  3  2 17 19 16

4. adimda uretilen Sayilar
8  7 12 19 10 13  8 20 16 15

5. adimda uretilen Sayilar
4 12  3 14 14  5  2 12 14  9

-----
```

2. Çalıştırma sonucu

```
1. adimda uretilen Sayilar
2  8 15  1 10  5 19 19  3  5

2. adimda uretilen Sayilar
6  6  2  8  2 12 16  3  8 17

3. adimda uretilen Sayilar
12  5  3 14 13  3  2 17 19 16

4. adimda uretilen Sayilar
8  7 12 19 10 13  8 20 16 15

5. adimda uretilen Sayilar
4 12  3 14 14  5  2 12 14  9

-----
```


- Her çalışmada aynı sayıların üretilmesini engellemek için **srand()** fonksiyonu kullanılır.
- Programın hataları tamamen ayıklandığında, her çalıştırılmada rasgele sayıların farklı bir dizisinin üretilmesi sağlatılabilir. Buna, **rasallaştırma** denir ve **srand()** fonksiyonu sayesinde bu işlem yapılır.

Örnek: srand() fonkiyonu ile, 1 - 20
arasında 5 kere 10 elemanlı sayı üreten
bir C programı yazınız.

Her çalıştırmada farklı sayılar üretilir.

```
1. adimda uretilen Sayilar
1 19 19 16 2 20 12 19 12 6
2. adimda uretilen Sayilar
9 13 2 4 18 4 8 16 12 11
3. adimda uretilen Sayilar
11 14 7 17 20 16 20 1 17 19
4. adimda uretilen Sayilar
2 13 13 10 19 9 6 5 14 17
5. adimda uretilen Sayilar
3 1 18 1 6 14 6 13 10 5
-----
```

➡ 1. çalıştırma

```
1. adimda uretilen Sayilar
9 8 13 13 15 10 11 3 7 8
2. adimda uretilen Sayilar
8 6 8 17 4 17 6 17 9 13
3. adimda uretilen Sayilar
15 5 16 3 18 20 14 2 17 6
4. adimda uretilen Sayilar
9 17 5 13 8 19 9 13 9 17
5. adimda uretilen Sayilar
3 4 8 4 18 18 17 16 17 13
-----
```

➡ 2. çalıştırma

```
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
#include <time.h>
```

```
int main()
{ int A[10], i, x, j;
  srand(time(NULL));
```

```
  for(j=0;j<5;j++)
  {
    for (i=0;i<10;i++)
    {x=1+(rand()%20);
     A[i]=x;}
    printf ("%d. adimda uretilen Sayilar\n",j+1);
    for (i=0;i<10;i++)
    printf ("%d ",A[i]);
    printf ("\n\n");
  }
  getch();
  return 0;
}
```

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.