

# Bilgisayar ve Programlamaya Giriş



Hafta 8-9

## Döngüler

Tekrar (döngü) yapıları belirli bir kod parçasının farklı değerler üzerinde tekrar tekrar çalıştırılmasını sağlar. Örneğin **dizi** ve **matrise** dayalı uygulamalar genel olarak döngü deyimlerini kullanır.

Döngü deyimleri kendi içerisinde iki sınıfa ayrılır. Birisi koşulu başta sınavan, diğeri koşulu sonda sınavan.

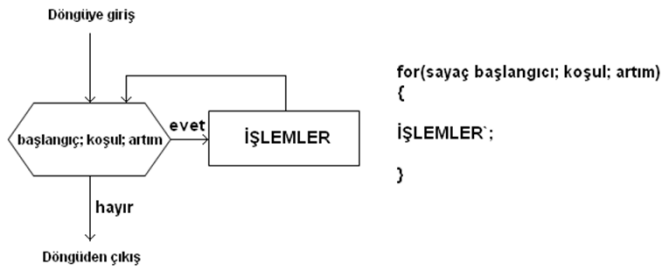
Koşulu başta sınavması, daha çevrime girmeden döngü koşuluna bakılması ve koşul olumlu ise çevrime girilmesi, koşul olumsuz ise çevrime girilmeden sonraki adımlara geçilmesi anlamına gelir.

Koşulun sonda sınavması ise, çevrim içerisinde kodun en az bir kere işletilmesi ve eğer koşul sağlanıyorsa çevrime devam edilmesi, sağlanmıyor ise çevrimden çıkılması anlamına gelir.

## for döngüsü

**for** döngüsü aynı işlemleri birçok kez tekrarlamak için kullanılır. Koşul sınaması çevrime girmeden yapılır.

**for** döngüsünde koşul sayacı ve “*for*” koşul satırında üç parametre vardır. Birinci parametre döngü sayacına başlangıç değeri verilmesi, ikinci parametre koşulu ve üçüncü parametre her çevrimde sayacın nasıl artacağını/eksileceğini ifade eder.



Döngüye girmeden önce sayaç başlangıç değeri alır ve daha sonra koşula bakılır. Döngü içerisindeki işlemler yapıldıktan sonra sayaç üçüncü parametrenin durumuna göre değiştirilir (artırılır/eksiltir).

```
for(sayaç başlangıcı; koşul; artım)
{
    İŞLEMLER;
}
```

## for döngüsü

Örnekler:

```
for (k=1; k<50; k+=2)
```

```
for (k=5; k<=n; k++)
```

```
for (x=50; x>10; x--)
```

```
for ( ; x<10; x++) /* başlangıç değeri daha önce atanmış olmalı */
```

```
for (x=2; x<n; ) /* x döngü sayacı döngü içinde değiştirilmeli */
```

```
for ( ; ) /* sonsuz döngü */
```

## for döngüsü program örnekleri

```
/* for */
#include<stdio.h>
int main()
{
    int sayac;
    for(sayac=1; sayac<=10; sayac++)
        printf("%d\n ", sayac);
    return 0;
}
```

```
/* harfleri yazdıralım */
#include <stdio.h>
int main()
{
    char harf;
    for( harf = 'A'; harf <= 'Z'; harf++ )
        printf("%c ", harf);
    return 0;
}
```

```
/* for sonsuz döngü */
#include<stdio.h>
int main()
{
    for(;;) //sonsuz döngü
        printf("S"); /*döngüden çıkmak için Ctrl+Break ya da Ctrl+C */
    return 0;
}
```

```
/* kodu oynatalım*/
#include <stdio.h>
int main()
{
    int a=5;
    for( ; a>0 ; a--) printf("%d \n", a);
    printf("Döngü Bitti"); /*for ile alakası yok*/
    return 0;
}
```

## for döngüsü program örnekleri

```
/* for içi */
#include<stdio.h>
int main()
{
    int a,b;
    for(a=0, b=10; a<3 && b<16 ; a++, b+=5)
        printf("%d, %d\n", a,b);
    return 0;
}
```

```
/* for içini oynatalım */
#include<stdio.h>
int main()
{
    int a,b,x;
    for(a=0, b=10; a<3 && b<16 ; a++, b+=5, scanf("%d", &x), printf("%d\n", x))
        printf("%d, %d\n", a,b);
    return 0;
}
```

## for döngüsü program örnekleri

```
/* iç içe for */
#include <stdio.h>
int main()
{
    for(int i = 0; i < 3; i++ )
        for(int j = 0; j < 3; j++ )
            printf("\ni=%d, j=%d", i,j);
            printf("Merhaba"); //for ile alakası yok
    return 0;
}
```

```
/* iç içe for */
#include <stdio.h>
int main()
{
    for(int i = 0; i < 2; i++ )
        for(int j = 0; j < 2; j++ )
        {
            for(int k = 0; k < 2; k++ )
                printf("\ni=%d, j=%d, k=%d ", i,j,k);
        }
    printf("\nBitti");
    return 0;
}
```

## for döngüsü program örnekleri

```
*
* *
* * *
* * * *
* * * * *
```

```
Z
Z y
Z y x
Z y x w
Z y x w v
```

```
      *
    * * *
  * * * * *
* * * * * *
* * * * * * *
```

```
#include <stdio.h>
int main()
{ int i, j;
  for( i = 0 ; i < 5; i++ ) {
    for( j = 0; j < i+1; j++ ) {
      printf(" " );
    }
    printf("\n");
  }
  return 0; }
```

```
#include <stdio.h>
int main()
{ int sat;
  char ch;
  for (sat = 1; sat <= 5; sat++)
  {
    for (ch = 'z'; ch > 'z' - sat; ch=ch-1)
      printf("%c ",ch);
    printf("\n");
  }
  return 0; }
```

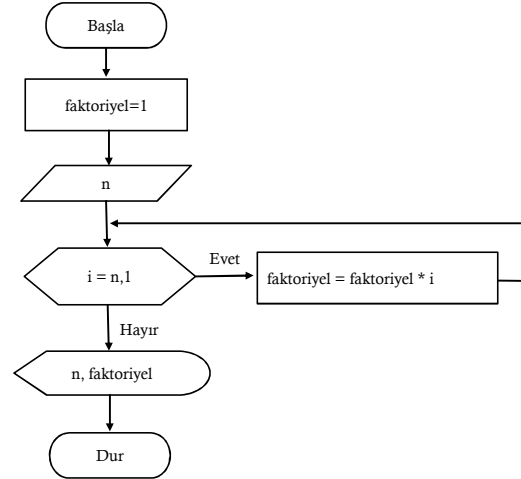
```
#include <stdio.h>
int main()
{ int i,k,j;
  for(i=1;i<=9;i+=2)
  {
    for(k=0;k<(9-i)/2;k++)
      printf(" ");
    for(j=0;j<i;j++)
      printf("*");
    printf("\n");
  } return 0; }
```

# Faktöriyel Hesabı

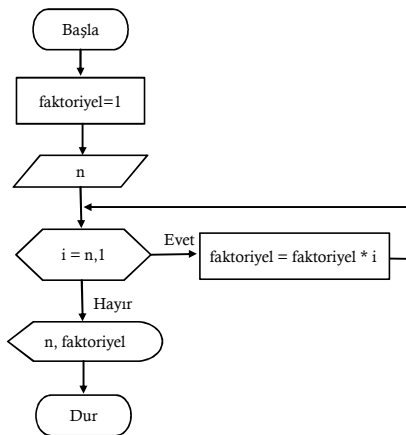
**Faktöriyel** fonksiyonu verilen pozitif tam sayının kendisinden önceki bütün pozitif tamsayılarla 1'e inilinceye kadar çarpılması sonucunda elde edilen çarpımı gösterir:

- $1! = 1 \cdot 1 = 1$
- $2! = 2 \cdot 1 = 2$
- $3! = 3 \cdot 2 \cdot 1 = 6$
- $4! = 4 \cdot 3 \cdot 2 \cdot 1 = 24$

$$\begin{aligned} n! &= \prod_{k=1}^n k \\ &= 1 \cdot 2 \cdot 3 \cdots (n-2) \cdot (n-1) \cdot n \\ &= n(n-1)(n-2) \cdots (2)(1) \\ n! &= n \cdot (n-1)!. \end{aligned}$$



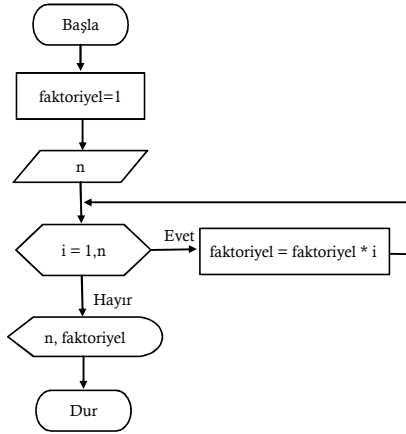
# Faktöriyel Hesabı (n'den 1'ye)



```
/* faktöriyel (n'den 1'e) */
#include <stdio.h>
int main()
{
    int i, n, faktoriyel=1;

    printf("faktöriyeli alınacak sayiyi giriniz: ");
    scanf("%d",&n);
    for(i=n;i>=1;i--)
        faktoriyel*=i;
    printf("f(%d) = %d\n", n,faktoriyel);
    return 0;
}
```

# Faktöriyel Hesabı (1'den n'ye)



```
/* faktöriyel (1'den n'e) */  
#include <stdio.h>  
int main()  
{  
    int i, n, faktoriyel=1;  
  
    printf("faktoriyeli alınacak sayiyi giriniz: ");  
    scanf("%d",&n);  
    for(i=1;i<=n;i++)  
        faktoriyel*=i;  
    printf("f(%d) = %d\n", n,faktoriyel);  
    return 0;  
}
```