

Bilgisayar ve Programlamaya Giriş



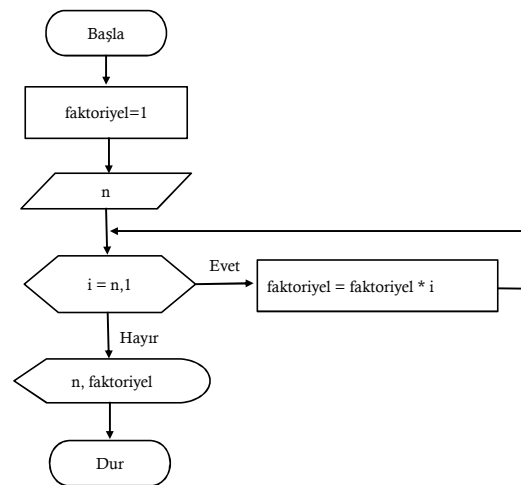
Hafta 11

Faktöriyel Hesabı

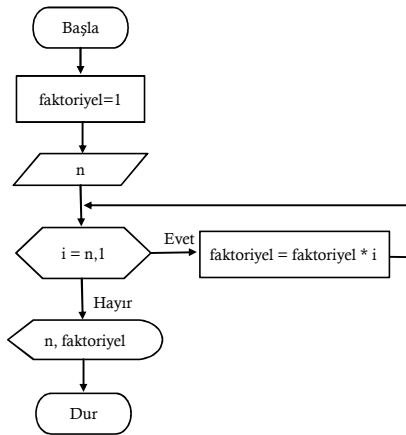
Faktöriyel fonksiyonu verilen pozitif tam sayının kendisinden önceki bütün pozitif tamsayılarla 1'e inilinceye kadar çarpılması sonucunda elde edilen çarpımı gösterir:

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

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



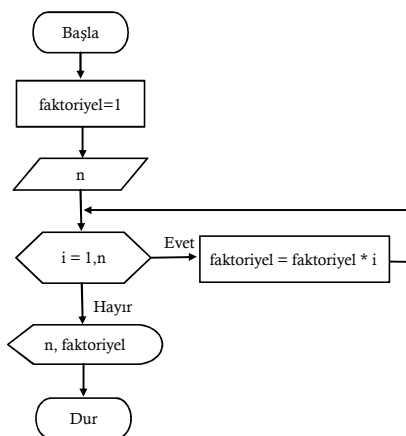
Faktöriyel Hesabı (n'den 1'ye)



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

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

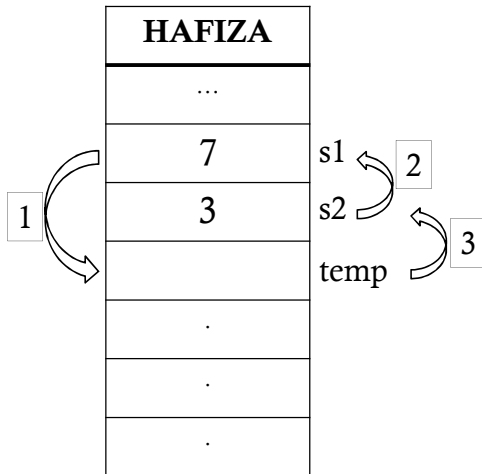
Faktöriyel Hesabı (1'den n'ye)



```
/* 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;
}
```

NOT: İki sayının yer değişimi



```
/* iki sayının yer deęiřimi */
#include<stdio.h>

int main()
{
    int s1=7,s2=3,temp;
    printf("%d,%d\\n",s1,s2);

    temp=s1;
    s1=s2;
    s2=temp;

    printf("%d,%d",s1,s2);

    return 0;
}
```

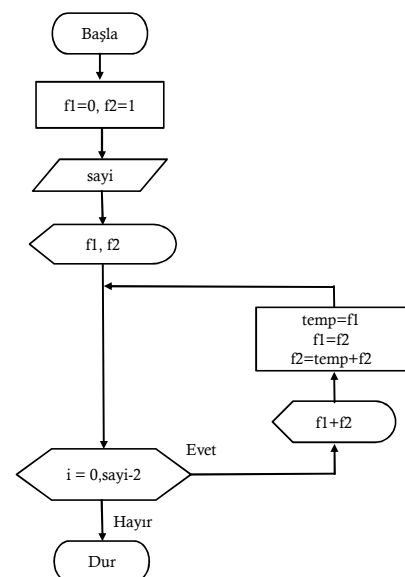
Fibonacci Serisi

- **Fibonacci Serisi**, her sayının kendisinden bir önceki sayı ile toplanması ile elde edilen sayılar serisidir.

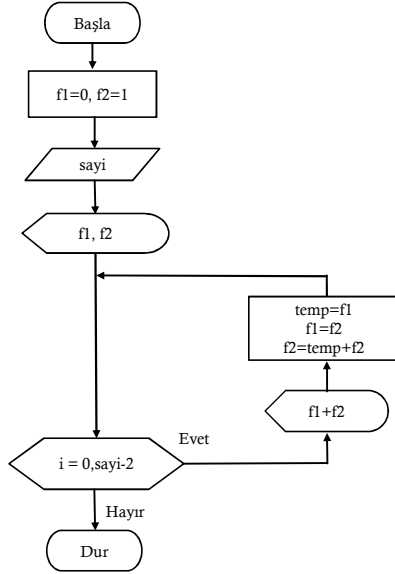
$$F_n = \begin{cases} 0 & \text{if } n = 0; \\ 1 & \text{if } n = 1; \\ F_{n-1} + F_{n-2} & \text{if } n > 1. \end{cases}$$

0 1 1 2 3 5 8 13 21 34 55 89 144 233 ...

- Bu dizinin ileri elemanlarında, bir sonraki elemanın bir öncekine oranı Altın oran adı verilen ve yaklaşık 1,618 (1:0,618) değerine eşit bir sayıyı verir.



Fibonacci Serisi



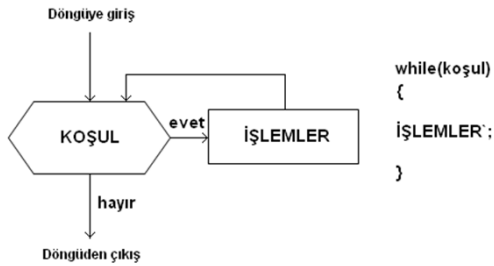
```
/* fibonacci serisi */
#include<stdio.h>
int main()
{
    int f1=0, f2=1, sayi, i, temp;
    printf("Kac fibonacci sayisi yazilsin?");
    scanf("%d", &sayi);

    printf("%d ", f1);
    printf("%d ", f2);
    for(i=0; i<sayi-2; ++i)
    {
        printf("%d ", f1+f2);
        temp=f1;
        f1=f2;
        f2=temp+f2;
    }
    return 0; }
```

while döngüsü

while döngüsü for döngüsü gibi aynı işlemleri birçok kez tekrarlamak için kullanılır. Bu döngüde de koşul sınaması çevrime girmeden yapılır.

Koşul tek bir karşılaştırmadan oluşabileceği gibi birden çok koşulun mantıksal operatörler ile birleştirilmesi ile de oluşturulabilir.



```
/* while(1) while(false)
while(true) */
#include<stdio.h>
int main()
{
    while (0)
        printf("kutahya\n");
    return 0;
}
```

```
/* while döngüsü*/
#include<stdio.h>
int main()
{
    int a=0;
    while(a<3)
    {
        printf("DPU\n");
        a++;
    }
    return 0;
}
```

while döngüsü program örnekleri

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

```
/* while sonsuz döngü */
#include<stdio.h>
int main()
{
  while(1) //sonsuz döngü
    printf("S"); /*döngüden çıkmak için Ctrl+Break ya da Ctrl+C */
  return 0;
}
```

```
/* bir tuşa basınca çıkalım */
#include <stdio.h>
int main()
{
  char c;
  printf("\nTuslara basin, cikmak icin 'q' ya basiniz \n");
  c=getchar();
  while(c!='q') //q'ya basana kadar okuma devam eder
  {
    c=getchar(); // bir sonraki karakteri okuyor
  }
  printf("\ngüle gülee ;) \n");
  return 0;
}
```

while döngüsü ile program örnekleri

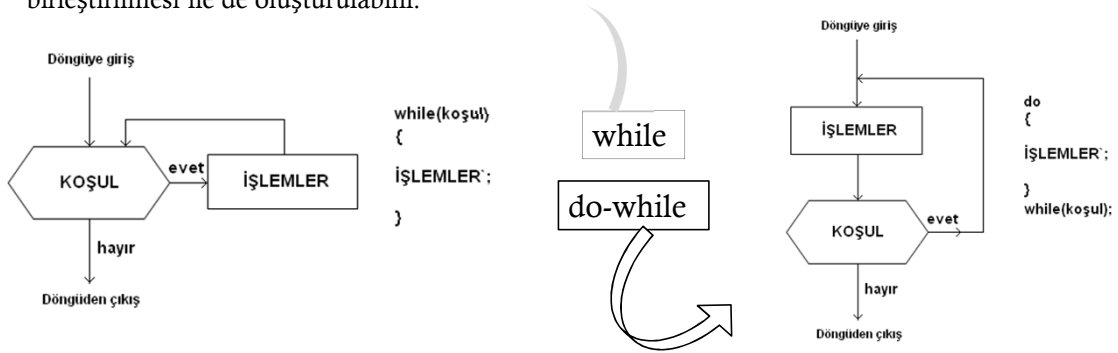
```
/* sayının basamak değerleri toplamını bulma */
#include <stdio.h>
int main()
{
  int sayi, toplam = 0, kalan;
  printf("bir sayi giriniz\n");
  scanf("%d",&sayi);
  while(sayi != 0) /* n 0'a esit degilse*/
  {
    kalan = sayi % 10;
    toplam = toplam + kalan;
    sayi = sayi / 10;
  }
  printf("basamaklar toplami = %d\n",toplam);
  return 0;
}
```

```
/*tek satır cümle içindeki boşlukları sayduralım */
#include <stdio.h>
int main()
{
  int c, bosluk=0;
  printf("\nBir cümle giriniz:\n");
  c=getchar(); //tek karakter okur
  while(c!='\n') //enter basan kadar okuma devam eder
  {
    if(c==' ') bosluk++;
    c=getchar(); // bir sonraki karakteri okuyor
  }
  printf("boşluk sayısı: %d", bosluk);
  return 0;
}
```

do-while döngüsü

do - while döngüsü diğer döngüler gibi aynı işlemleri birçok kez tekrarlamak için kullanılır. Farklı olarak, bu döngüde koşul sınaması yapılmadan çevrime girilir ve işlem kümesi en az bir kere işletilir. Bu deyim yapısında da koşul sağlandığı sürece çevrim tekrarlanır.

Koşul tek bir karşılaştırmadan oluşabileceği gibi birden çok koşulun mantıksal operatörler ile birleştirilmesi ile de oluşturulabilir.



do-while döngüsü program örnekleri

```
#include <stdio.h>
int main()
{ int x=6;
  do{
    x=x-2;
    printf("%d\n",x);
  }while(x>=0);
return 0; }
```

```
#include<stdio.h>
int main()
{
  int i=1;
  do{
    printf("i'nin degeri : %d\n",i);
    i++;
  }while(i<=4 && i>=2);
return 0; }
```

```
// Çıkmak için bir tuşa basma örneği
#include<stdio.h>
int main()
{
  char c;
  printf("Herhangi tuşa basın, çıkmak için c :\n ");
  do {
    scanf("%c", &c);

  } while (c!= 'c');
  return 0;
}
```

while

vs

do-while

```
/*tek satır cümle içindeki boşlukları sayduralım */
#include <stdio.h>
int main()
{
    int c, bosluk=0;
    printf("\nBir cümle giriniz:\n");
    c=getchar(); //tek karakter okur
    while(c!='\n') //enter basana kadar okuma devam eder
    {
        if(c==' ') bosluk++;
        c=getchar(); // bir sonraki karakteri okuyor
    }
    printf("boşluk sayısı: %d", bosluk);
    return 0;
}
```

```
/* tek satır cümle içindeki boşlukları sayduralım */
#include <stdio.h>
int main()
{
    int c, bosluk=0;
    printf("\nBir cümle giriniz:\n");
    do{
        c=getchar(); // bir sonraki karakteri okuyor
        if(c==' ') bosluk++;
    }while(c!='\n') //enter basana kadar okuma devam eder

    printf("boşluk sayısı: %d", bosluk);
    return 0;
}
```

break ve continue 1

```
#include <stdio.h>
int main()
{
    for (int i = 1; i<10; i++)
    {
        printf("%d\n", i);
        if (i == 3)
            //i=3 olursa break devreye girerek for'dan çıkar
            break;
        printf("dongunun altı\n");
    }
    printf("dongunun disi");
    return 0;
}
```

```
#include<stdio.h>
int main()
{
    for (int i = 1; i <= 6; i++)
    {
        if (i % 2)
            continue; //döngüye devam
        else
            printf("%d\n",i);

        printf("dongunun altı\n");
    }
    return 0;
}
```

break ve continue 2

```
#include <stdio.h>
int main()
{
    int a = 0;
    while (a<5)
    {
        if (a<3)
        {
            a += 2;
            printf("%d\n", a);
            continue;
        }
        else
        {
            printf("%d\n", ++a); // a++ yazarsak 4 olur
            break; //break ile while'dan çıkıyoruz
        }
        printf("dongunun alti\n"); //hiç uğramıyor
    }
    return 0;
}
```