

**Problem-1.2. En Büyük Ortak Bölen Bulma Problemi****(Öklid Algoritması)**

$m, n > 0$  ve  $m > n$  olacak biçimde iki tamsayı olsun. Bu iki tamsayının en büyük ortak bölenini bulmak için kullanılan Öklid Algoritmasını inceleyelim. İki tamsayının en büyük ortak böleni (EBOB), verilen her iki sayıyı tam olarak bölen en büyük tamsayıdır. Öklid Algoritması ile EBOB'ü bulmak için verilen sayılardan büyüğü küçüğüne bölünür ve kalan hesaplanır. Eğer kalan sıfır ise küçük olan sayı EBOB'dir ve işlem sonlandırılır. Değilse bölüm yerine küçük olan sayı, bölen yerine de kalan sayı alınarak aynı işlemler tekrarlanır. 1380 ile 38 sayılarının en büyük ortak bölenini bu yöntemle bulalım;

$$1380/38 \text{ kalan } 12$$

$$38/12 \text{ kalan } 2$$

$$12/2 \text{ kalan } 0$$

En son bölen sayı 2 en büyük ortak bölendir. Buna göre Öklid Algoritmasının adımları aşağıdaki gibi verilir.

**A1.**  $m, n$  tamsayılarını  $m > n$  olacak biçimde GİR/OKU

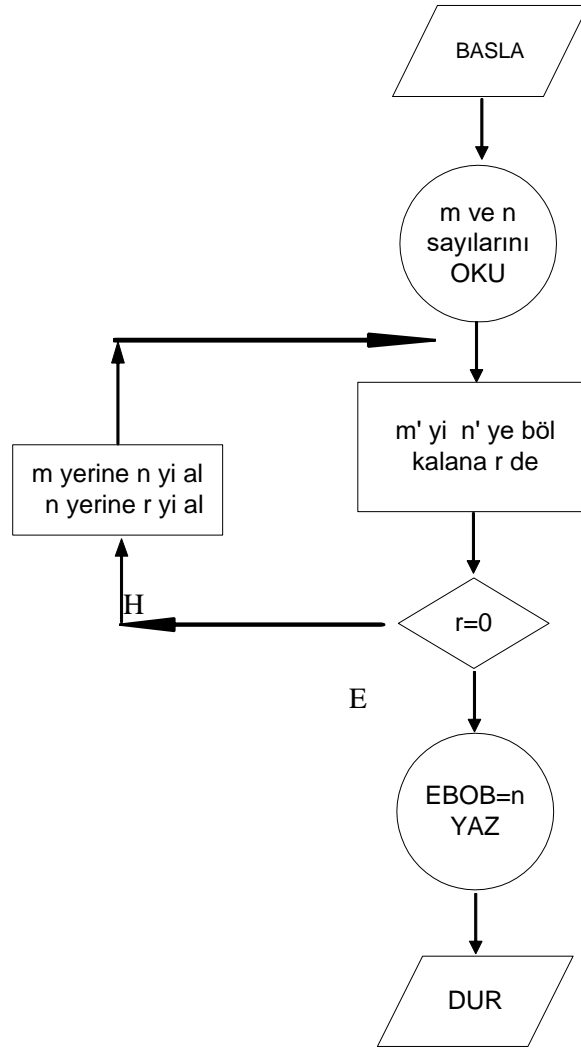
**A2.**  $m$ 'yi  $n$ 'ye böl ve kalana  $r$  de

**A3.** EĞER  $r = 0$  İSE “EBOB( $m, n$ )= $n$ ” YAZ, DUR

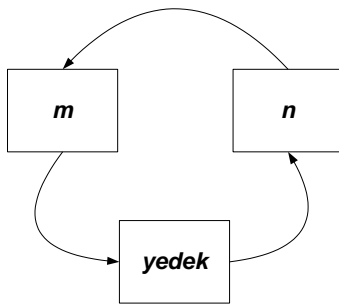
**A4.**  $m$ 'nin yerine  $n$ 'yi,  $n$ 'nin yerine  $r$ 'yi al ve A2'ye GİT

*Algoritmanın ikinci adımından sonra  $m = bn + r$  kolayca yazılabilir. Burada  $b$  bölüm,  $r$  de algoritmada verildiği gibi  $m$ 'nin  $n$ 'ye bölünmesinden kalandır. Eğer  $r = 0$  ise  $m, n$ 'ye tam bölünüyor demektir ki, bu durumda en büyük ortak bölen gerçekten  $n$ 'dir. Eğer  $r$  sıfırdan farklı ise,  $m$ 'nin ve  $n$ 'nin herhangi bir ortak böleni  $r = m - bn$ 'yi de böler. Öte yandan hem  $n$ 'yi hem de  $r$ 'yi bölen bir sayı  $m = bn - r$ 'yi de böler. Şu halde  $m$  ile  $n$ 'nin ortak bölenlerinin kümesi ile  $n$  ile  $r$ 'nin ortak bölenlerinin kümesi aynıdır; öyleyse en büyük ortak bölenler de aynıdır. Bu nedenle 4'üncü adımdaki yer değiştirme, problemin yanıtını değiştirmez (Tepedelenlioğlu 1993, Penrose 1997).*

Yukarıda verilen algoritma adımlarının akış şeması aşağıdaki gibidir.



$n > m$  olduğunda algoritmanın çalışabilmesi için bu iki sayının yerlerinin değiştirilmesi gerekir. Bunun için bir yedek değişkene ihtiyaç duyulur. Buna göre,  $m$  değişkeni  $yedek$  değişkenine,  $n$  değişkeni  $m$  değişkenine ve  $yedek$  değişkeni  $n$  değişkenine atanarak yer değiştirme işlemi yapılır. Bu yapılan atama işlemi kısaca aşağıdaki gibi gösterilir.



**Atama işlemi:**

$yedek = m$

$m = n$

$n = yedek$

Bu durumda Öklid algoritması aşağıdaki şekilde yeniden yazılabilir.

**A1.**  $m, n$  tamsayılarını GİR/OKU

**A2.** EĞER  $m < n$  İSE

$yedek = m$

$m = n$

$n = yedek$

al

**A3.**  $m$ 'yi  $n$ 'ye böl ve kalana  $r$  de

**A4.** EĞER  $r = 0$  İSE

“EBOB( $m, n$ )= $n$ ” YAZ

DUR

**A5.**  $m=n$

$n=r$  al ve A3' e GİT

Burada dikkat edilmesi gereken kalanın nasıl bulunacağı sorunudur. Bunun için,  $m$  ve  $n$  gibi iki tamsayı verildiğinde kalan bulma algoritmasını geliştirelim. Örnek olarak,  $m$  ve  $n$  sayıları yerine sırasıyla 1380 ve 38 sayılarını alalım. Öncelikle 1380 sayısı 38'e bölüldüğünde 36.32 elde edilir. Buna göre, kalan  $1380-38*36=12$  olarak bulunur. Dolayısıyla, kalana  $r$  dediğimizde,  $r = m - \text{tamkısım}(m/n) \times n$  eşitliği ile elde edilir. Buna göre algoritmayı aşağıdaki gibi tekrar yazalım.

**A1.**  $m, n$  tamsayılarını GİR/OKU

**A2.** EĞER  $m < n$  İSE

$yedek = m$

$m = n$

$n = yedek$

al

**A3.**  $a = \text{INT}(m/n)$

**A4.**  $r = m - n * a$

**A5.** EĞER  $r = 0$  İSE

“EBOB( $m, n$ )= $n$ ” YAZ

DUR

**A6.**  $m=n$

$n=r$  al ve A3'e GİT

Basic programında bir sayının tam kısmını almak için INT(.) matematiksel fonksiyonu kullanılır.

**Tablo 1.2.** Kullanılan bazı hazır matematiksel fonksiyonlar ve tanımları.

Matematiksel Fonksiyonlar	Tanımları
ABS(.)	Parantez içinde verilen sayısal ifadenin mutlak değerini bulur.
ATN(.)	Parantez içinde radyan cinsinden verilen sayısal ifadenin ark tanjantını bulur.
COS(.)	Parantez içinde radyan cinsinden verilen sayısal ifadenin kosinüsünü bulur.
EXP(.)	Parantez içinde verilen sayısal ifadenin eksponansiyel değerini bulur.
FIX(.)	Parantez içinde verilen sayısal ifadenin tam kısmını bulur.
INT(.)	Parantez içinde verilen sayısal ifadenin tamsayı değerini bulur.
LOG(.) LOG10(.)	Parantez içinde verilen sayısal ifadelerin (e ve onluk tabanda) logaritmik değerlerini bulurlar.
SGN(.)	Parantez içinde verilen sayısal ifadenin işaretini bulur.

SIN(.)	Parantez içinde radyan cinsinden verilen sayısal ifadenin sinüsünü bulur.
SQR(.)	Parantez içinde verilen sayısal ifadenin karekökünü bulur.
TAN(.)	Parantez içinde radyan cinsinden verilen sayısal ifadenin tanjantını bulur.

INT(.) fonksiyonu kullanımı ile Öklid algoritması için program aşağıdaki şekilde yazılabilir.

INPUT "m DEĞERİNİ GİRİNİZ=",m

INPUT "n DEĞERİNİ GİRİNİZ =",n

IF n>m THEN

    yedeK=m

    m=n

    n=yedeK

END IF

A3:

a=INT(m/n)

r=m-n\*a

IF r=0 THEN

    PRINT "EBOB(m,n)=",n

    END

END IF

m=n

n=r

GO TO A3

INT(.) fonksiyonunu kullanmadan KALAN'ı bulan algoritma tasarlanabilir. Bunun için çıkarma işleminden yararlanılır. Algoritma adımları aşağıdaki gibidir.

**A1.**  $m, n$  tam sayılarını GİR/OKU

**A2.** EĞER  $m < n$  İSE

    yedeK = m

    m = n

    n = yedeK

**A3.**  $m = m - n$  al

**A4.** EĞER  $n < m$  İSE A3'e GİT

**A5.**  $r = m$  al

**A6.** EĞER  $r = 0$  İSE "EBOB(m, n)=n" YAZ, DUR

**A7.**  $m = n$

    n = r al

**A8.** A3'e GİT

Yukarıda verilen algoritmaya göre program aşağıdaki şekilde yazılır.

```
INPUT "m=", m
INPUT "n=", n
IF m < n THEN
    yedek = m
    m = n
    n = yedek
END
END IF
A3:
m = m - n
IF n < m THEN GOTO A3
r = m
IF r = 0 THEN
    PRINT "EBOB(m,n)=", n
END
END IF
m = n
n = r
GOTO A3
```

MOD operatörü yardımı ile KALAN'ı bulmak mümkündür. MOD operatörünün kullanımı *sayı1 MOD sayı2* biçimindedir. MOD operatörü yardımı ile Öklid Algoritması adımları program aşağıdaki gibidir.

- A1.  $m, n$  tam sayılarını GİR/OKU
- A2. EĞER  $n > m$  İSE  
yedek = m  
m = n  
n = yedek al
- A3.  $r = m \text{ MOD } n$  hesapla
- A6. EĞER  $r = 0$  İSE “EBOB( $m, n$ )= $n$ ” YAZ, DUR
- A7.  $m = n$   
 $n = r$  al
- A8. A3'e GİT

```
INPUT "m=", m
INPUT "n=", n
IF n > m THEN
    yedek = m
```

```
    m = n
    n = yedek
END IF
A3:
r = m MOD n
IF r = 0 THEN
    PRINT "EBOB(m,n)=", n
    END
END IF
m = n
n = r
GOTO A3
```