

Diofantoszi egyenlet

Adott két pozitív egész szám, A és B . Eldöntendő, hogy adott Z pozitív egész szám előállítható-e valahány A és valahány B összegeként. Azaz, hogy van-e olyan x és y természetes szám, hogy $Z = A * x + B * y$

Feladat

Ijunk olyan programot, amely megadott A és B -re és adott számokra kiszámítja, hogy azok előállíthatók-e A és B többszöröseinek összegeként !

Bemenet

A standard bemenet első sorában három egész szám van, az előállításban használható A és B szám ($1 \leq A, B \leq 1000$), és az előállítandó számok N darabszáma ($1 \leq N \leq 10\,000$) van. A következő N sor mindegyike egy előállítandó számot ($1 \leq Z_i \leq 1\,000\,000\,000$) tartalmaz.

Kimenet

A standard kimenet pontosan N sort tartalmazzon, az i . sorban az **Igen** szó legyen, ha Z_i előállítható A és B többszöröseinek összegeként, egyébként a **Nem** szó!

Példa

Bemenet	Kimenet
5 7 8	Igen
12	Igen
10	Nem
11	Igen
19	Igen
22	Igen
102	Nem
8	Nem
9	

Korlátok

Időlimit: 0.1 mp.

Memórilimit: 32 MiB

Pontozás: a tesztesetek 40%-ában $Z_i < 1\,000\,000$

Megoldás

Minden $0 < m < A$ számra jelölje $Am[m]$ azt a legkisebb Z természetes számot, amely előállítható A és B

többszörösének összegeként és $Z \% A = m$. ($A \%$ jel az osztás maradéka műveletet jelöli.) Ha nincs ilyen szám, akkor $Am[m]$ legyen ∞ . Hasonlóképpen, minden $0 < m < B$ számra jelölje $Bm[m]$ azt a legkisebb Z természetes számot, amely előállítható A és B többszörösének összegeként és $Z \% B = m$. Ha nincs ilyen szám, akkor $Bm[m]$ legyen ∞ .

Bármely Z szám akkor és csak akkor állítható első A és B többszörösének összegeként, ha

$$Am[Z \% A] \leq Z \text{ vagy } Bm[Z \% B] \leq Z$$

Ha $Am[m]$ illetve $Bm[m]$ létezik, akkor az nem nagyobb, mint $A * B$. Először számítsuk ki minden $z \leq A * B$ számra, hogy előállítható-e A és B többszörösének összegeként. A dinamikus programozás módszere ad megoldást. Jelölje $E(z)$ azt a logikai függvényt, amely akkor és csak akkor igaz, ha z előállítható. $E(0)$ igaz, és $z > 0$ esetén $E(z)$ akkor és csak akkor igaz, ha $A \leq z$ és $E(z - A) = \text{igaz}$ vagy $B \leq z$ és $E(z - B) = \text{igaz}$. Tehát az alábbi algoritmussal meg tudjuk határozni az $Am[m]$ illetve $Bm[m]$ értékeket.

```

Be:A, B, N
Inf:=A*B+1
ciklus x:=0-tól A-1 -ig
    Am[x]:=Inf
ciklus vége
ciklus x:=0-tól B-1 -ig
    Bm[x]:=Inf
ciklus vége
E[0]:=igaz Am[0]:=A Bm[0]:=B
ciklus z:=1-től A*B-ig
    E[z]:=hamis
    ha z>=A és E[z-A] igaz akkor E[z]:=igaz
    ha z>=B és E[z-B] igaz akkor E[z]:=igaz
    ha E[z]=igaz akkor
        ha z<Am[z % A] akkor Am[z % A]:=z
        ha z<Bm[z % B] akkor Bm[z % B]:=z
    elágazás vége
ciklus vége

ciklus i:=1-től N-ig
    Be:z
    ha Am[z % A]<=z vagy Bm[z % B]<=z akkor
        Ki:Igen
    egyébként
        Ki:Nem
    elágazás vége
ciklus vége

```

Megvalósítás C++ nyelven

```

1  #include <iostream>
2  #define maxAB 1000
3  #define INF maxAB*maxAB+1
4  using namespace std;
5
6  int A,B,N;
7  int Am[maxAB];
8  int Bm[maxAB];
9  bool E[maxAB*maxAB];
10
11 int main() {
12     cin>>A>>B>>N;
13     long long x;
14     for(int x=1;x<A;x++) Am[x]=INF;
15     for(int x=1;x<B;x++) Bm[x]=INF;
16     Am[0]=A; Bm[0]=B;
17     E[0]=true;
18     for(int x=1;x<=A*B;x++){
19         E[x]=(x>=A && E[x-A]) || (x>=B && E[x-B]);
20         if (E[x]){
21             if(x<Am[x % A]) Am[x % A]=x;
22             if(x<Bm[x % B]) Bm[x % B]=x;
23         }
24     }
25     for(int i=0;i<N;i++){

```

```
26     cin>>x;
27     if( x>=Am[x%A] || x>=Bm[x%B])
28         cout<<"Igen"<<endl;
29     else
30         cout<<"Nem"<<endl;
31 }
32 return 0;
33 }
```