Is it perfect?

sum of its proper divisors
(excluding the number itself).

6 ⟶ 1 . 2 , 3 ~~6~~

sum = 1 + 2 + 3 = 6

Yes

1 2 3 4 5

```
int n = scn.nextInt();

int sum = 0;

for (int i = 1; i < n; i++) {

        if (n % i == 0) {

            sum = sum + i;
        }

}

if (sum == n) {
    S.O.P ("Yes");

} else {
    S.O.P ("No");
 }
```

$n = 6$

$Sum = 0$

$6 \% 1 == 0$

i	i < 6	6 % i == 0	sum	i++
1	true	true	0+1 = 1	2
2	true	true	1+2 = 3	3
3	true	true	3+3 = 6	4
4	true	false	—	5
5	true	false	—	6
6	false	→ Break		

$Sum == n$

$6 == 6 \Rightarrow true \Longrightarrow Yes$

handle multiple test case

```
int t = scn.nextInt();
while (t>0) {
    // work
```



```
    t--;
}
```

Easy Power

two integers A and B → $(A)^B$

A = 5
B = 3

$(5)^3 \Rightarrow 125$

5 x 5 x 5

A = 2
B = 5

$(2)^5 \Rightarrow 32$

2 x 2 * 2 * 2 * 2

$(A)^B$ → multiplying A, B number of times

A = 4 B = 5 ⇒ 4 x 4 x 4 x 4 x 4

```
int ans = 1 ;

for ( int i=1 ; i<=B ; i++) {

    ans = ans * A;

}
```

B=5

A=3

ans = 1

i = 1 ans = 1*3 = 3 ⇒ $(3)^1$

i = 2 ans = 3*3 ⇒ $(3)^2$

i = 3 ans = 3*3*3 ⇒ $(3)^3$

⋮

i = B ans ⇒ $(3)^B$

LCM → (Least common multiple)

A = 2

→ 2 4 6 8 10 ... - -

B = 3

→ 3 6 9 12 15 ... -

6 ⇐

2	2, 3
3	1, 3
	1, 1

20 ⇐

2	4, 10
2	2, 5
5	1, 5
	1, 1

A = 4

→ 4 8 12 16 20 24 ... ... - -

B = 10

→ 10 20 30 40 50 60 ...

L.C.M is the smallest number which is completely divisible by both A & B.

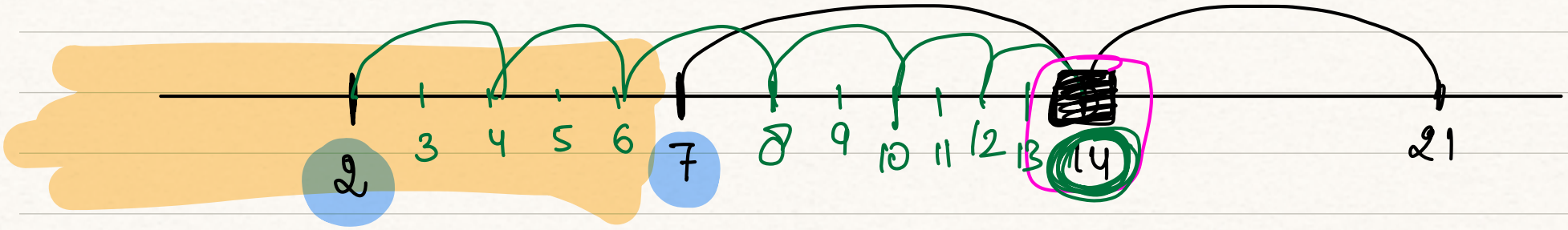
A = 8 B = 40 max = 40

8 x 5 = 40 = 40 x 1

12 ⇐

2	4 , 6
2	2 , 3
3	1 , 3
	1 , 1

A = 4 B = 6 ⇒ 12



① find the maximum between A & B

② move forward and try each number whether it is divisible by both or not

(3) stop loop as soon as you find first common multiple. [break statement]

```
int a = scn.nextInt();
int b = scn.nextInt();

int max = 0;
if (a>b) {
    max = a;
} else {
    max = b;
}
int ans = 0;
while ( true ) {
    if (max%a == 0 && max%b == 0) {
        break;
    }
    max++;
}
```

S.O.P (max);

Print the primes

N

print all prime numbers from 1 to N

N = 10 2 3 5 7

① check all numbers from 1 to N

② Count of factors == 2

↳ print the number

```
int n = scn.nextInt();

for(int i=1; i<=n; i++){

    // check and print if 'i' is prime

    int count = 0;
    for(int num=1; num<=i; num++){

        if(i%num==0){

            count++;
        }

        if(count>2){
            break;
        }

    }
```

```
        if (count == 2) {
            S.O.Pln (i);
        }
    }
```

Output →

2
3

N = 4 ⟶ 2 3

i = 1 count = 0

num	num <= i	i % num == 0	count++	count > 2	num++
1	true	true	1	false	2
2	false		break		

count == 2 → false

i = 2 count = 0

num	num <= i	i % num == 0	count++	count > 2	num++
1	true	true	1	false	2
2	true	true	2	false	3
3	false	→	→	break	

count == 2 (true)

i = 3 count = 0

num	num <= i	i % num == 0	count++	count > 2	num++
1	true	true	1	false	2
2	true	false	-	false	3
3	true	true	2	false	4
4	false	→	→	break	

count == 2 (true)

i = 4

count = 0

num	num <= i	i % num == 0	count++	count > 2	num++
1	true	true	1	false	2
2	true	true	2	false	3
3	true	false	–	false	4
4	true	true	3	true →	break

count == 2 (false)

```
int i = 0;
for( i= 0;  i < 5;  i++) {

    if (i < 4) {

        S.O.P ("Hello! ");
        break;

    }

}
```

i	i < 5	i < 4	Output	i++
0	true	true	Hello!	→ break [exit the loop]

```
int a=0;
for (int i=1; i<=10; i++) {
    if (i%2!=0){
        continue;
    }
    a++;
}
S.O.P (a);
```

⇒ 5

a=0

i	i<=10	i%2!=0	a	i++
1	true	true →		2
2	true	false	1	3
3	true	true →		4
4	true	false	2	5
5	true	true →		6
6	true	false	3	7
7	true	true →		8
8	true	false	4	9
9	true	true →		10
10	true	false	5	11
11	false → Break			

Doubts

```
for (int i=1; i<=5; i++){
    S.O.P (i);
    if (i==3){
        break
    }
}
```

i	i<=5	Output	i==3	i++
1	✓	1	X	2
2	✓	2	X	3
3	✓	3	✓ →	break exit the loop

HCF

a = 24

b = 36

12

1 → min

last answer will be the highest

1 → 24

1 2 3 4 5 6 7 8 9 10 11 12

✓ ✓ ✓ ✓ ✗ ✓ ✗

ans = i

min → 1

first ans will be the highest

break