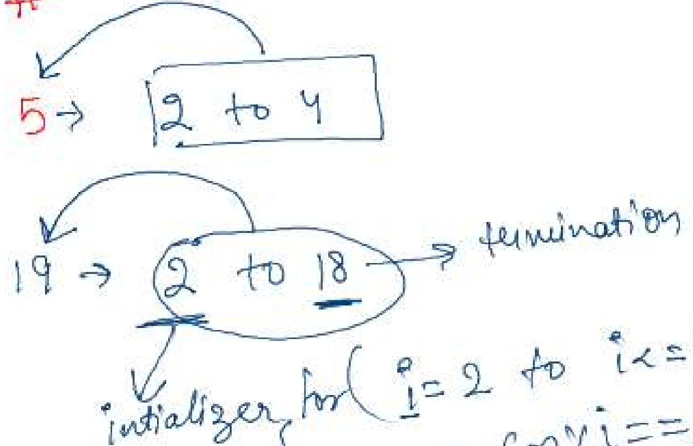
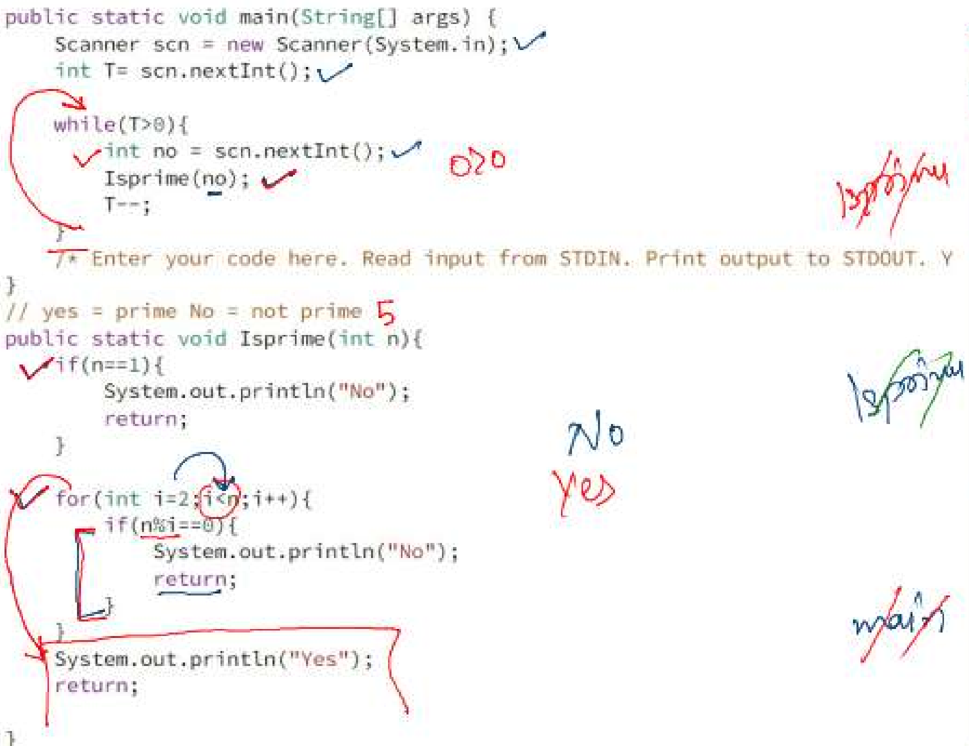
Q Prime



```
for( i=2 to i<=n-1){
    if (n%i==0){
        Syso("not prime");
    }
}
Syso("prime");
```



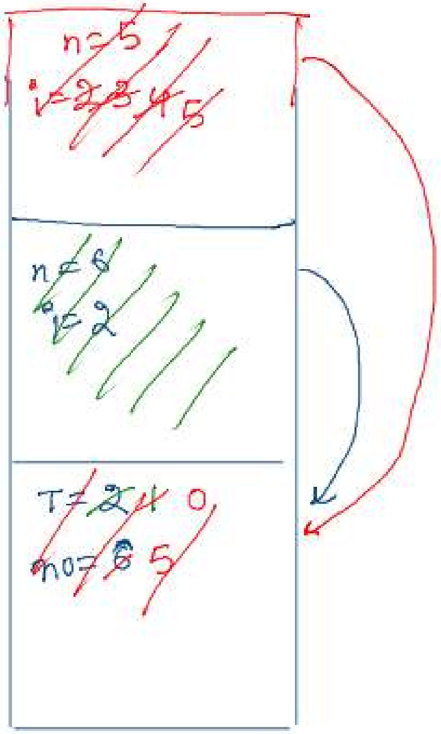
```
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int T= scn.nextInt();

    while(T>0){
      int no = scn.nextInt();
      Isprime(no);
      T--;
    }
    /* Enter your code here. Read input from STDIN. Print output to STDOUT. Y
}
// yes = prime No = not prime
public static void Isprime(int n){
    if(n==1){
        System.out.println("No");
        return;
    }

    for(int i=2;i<n;i++){
        if(n%i==0){
            System.out.println("No");
            return;
        }
    }
    System.out.println("Yes");
    return;

}
```

0>0

5

No

Yes



Q Unique prime factors.

n = 12

2	12
2	6
3	3
	1

$\underbrace{2 * 2} * 3 \Rightarrow 2 * 3$

2	36
2	18
3	9
3	3
	1

$\Rightarrow 2*2*3*3$

$\Downarrow$

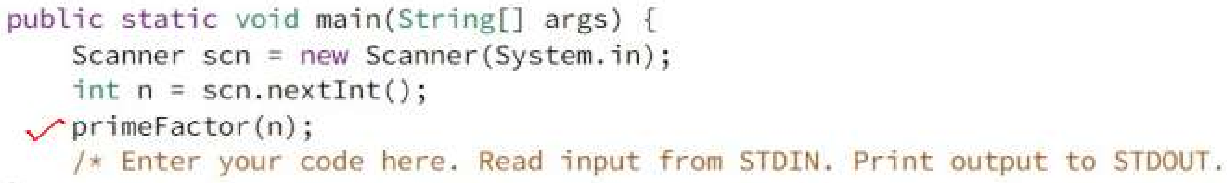
$\boxed{2*3}$

2	240 ✓
2	120 ✓
2	60 ✓
2	30 ✓
3	15
5	5
	1

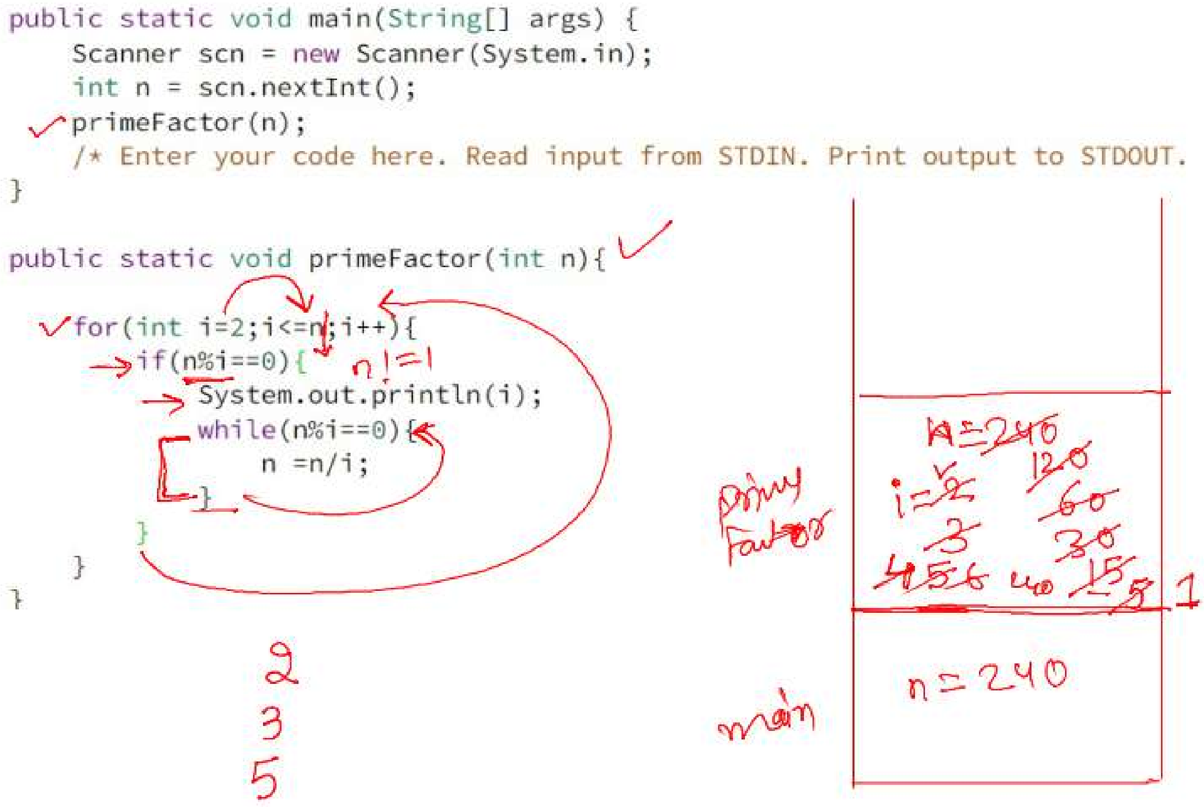
$\Rightarrow 2 \times 2 \times 2 \times 2 \times 3 \times 5$

$\Downarrow$

$\boxed{2 \times 3 \times 5}$



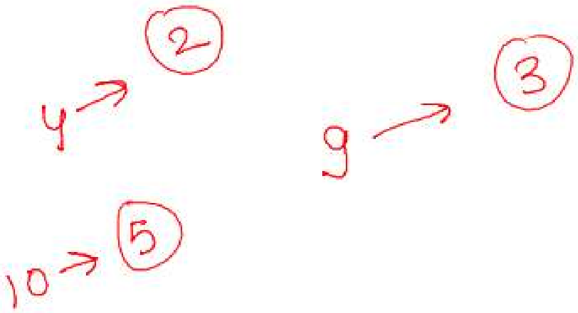
```
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    primeFactor(n);
    /* Enter your code here. Read input from STDIN. Print output to STDOUT.
```

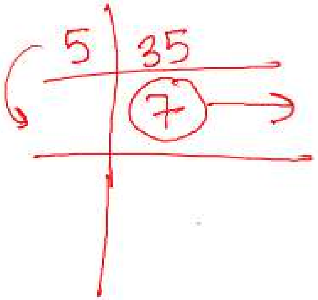


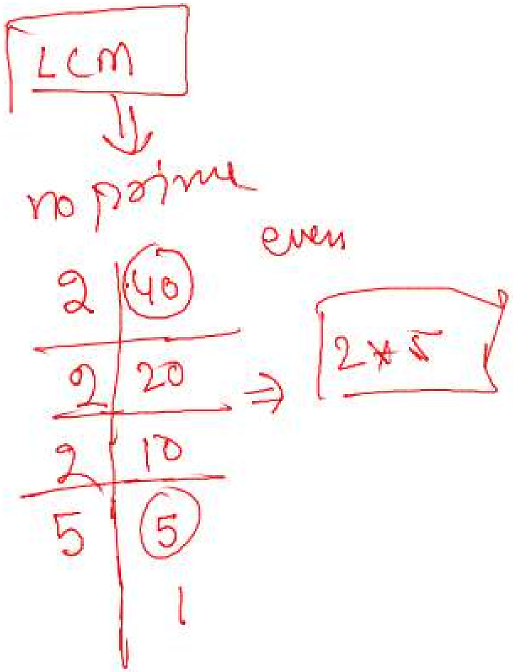
```
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    primeFactor(n);
    /* Enter your code here. Read input from STDIN. Print output to STDOUT.
}

public static void primeFactor(int n){

  for(int i=2;i<=n;i++){
      if(n%i==0){
          System.out.println(i);
          while(n%i==0){
              n =n/i;
          }
      }

  }
}
```

240%2==0
n

1%5==0
1







Q print all factors.

$1 \to n$

4 → 1, 2, 4

12 → 1, 2, 3, 4, 6, 12

--n

⇓

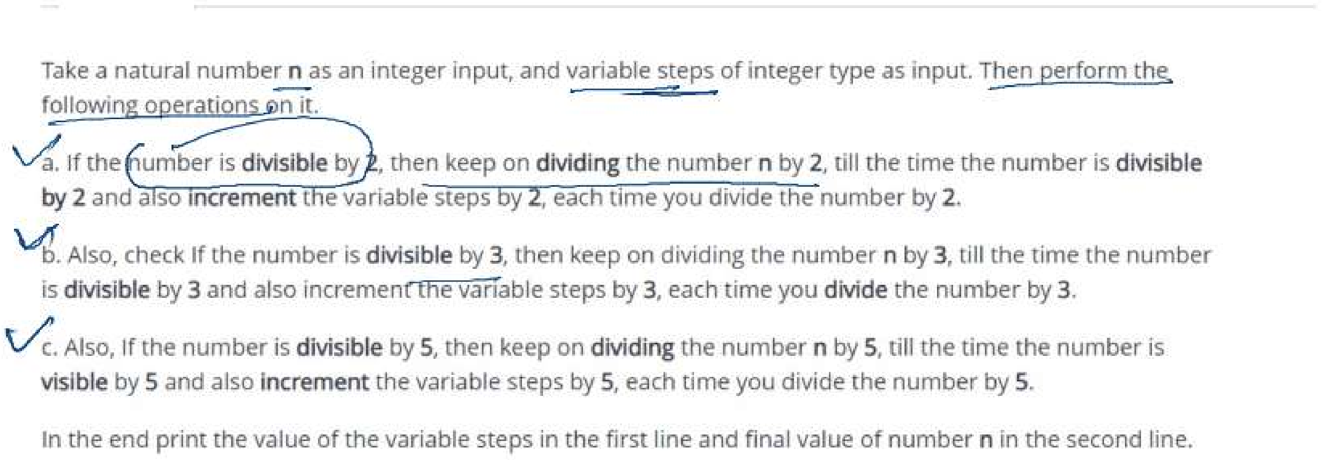
pre decrement

n--

post decrement

same response

Q



Take a natural number **n** as an integer input, and variable steps of integer type as input. Then perform the following operations on it.

a. If the number is **divisible** by **2**, then keep on **dividing** the number **n** by **2**, till the time the number is **divisible by 2** and also **increment** the variable steps by **2**, each time you divide the number by **2**.

b. Also, check If the number is **divisible** by **3**, then keep on dividing the number **n** by **3**, till the time the number is **divisible** by **3** and also increment the variable steps by **3**, each time you **divide** the number by **3**.

c. Also, If the number is **divisible** by **5**, then keep on **dividing** the number **n** by **5**, till the time the number is **visible** by **5** and also **increment** the variable steps by **5**, each time you divide the number by **5**.

In the end print the value of the variable steps in the first line and final value of number **n** in the second line.

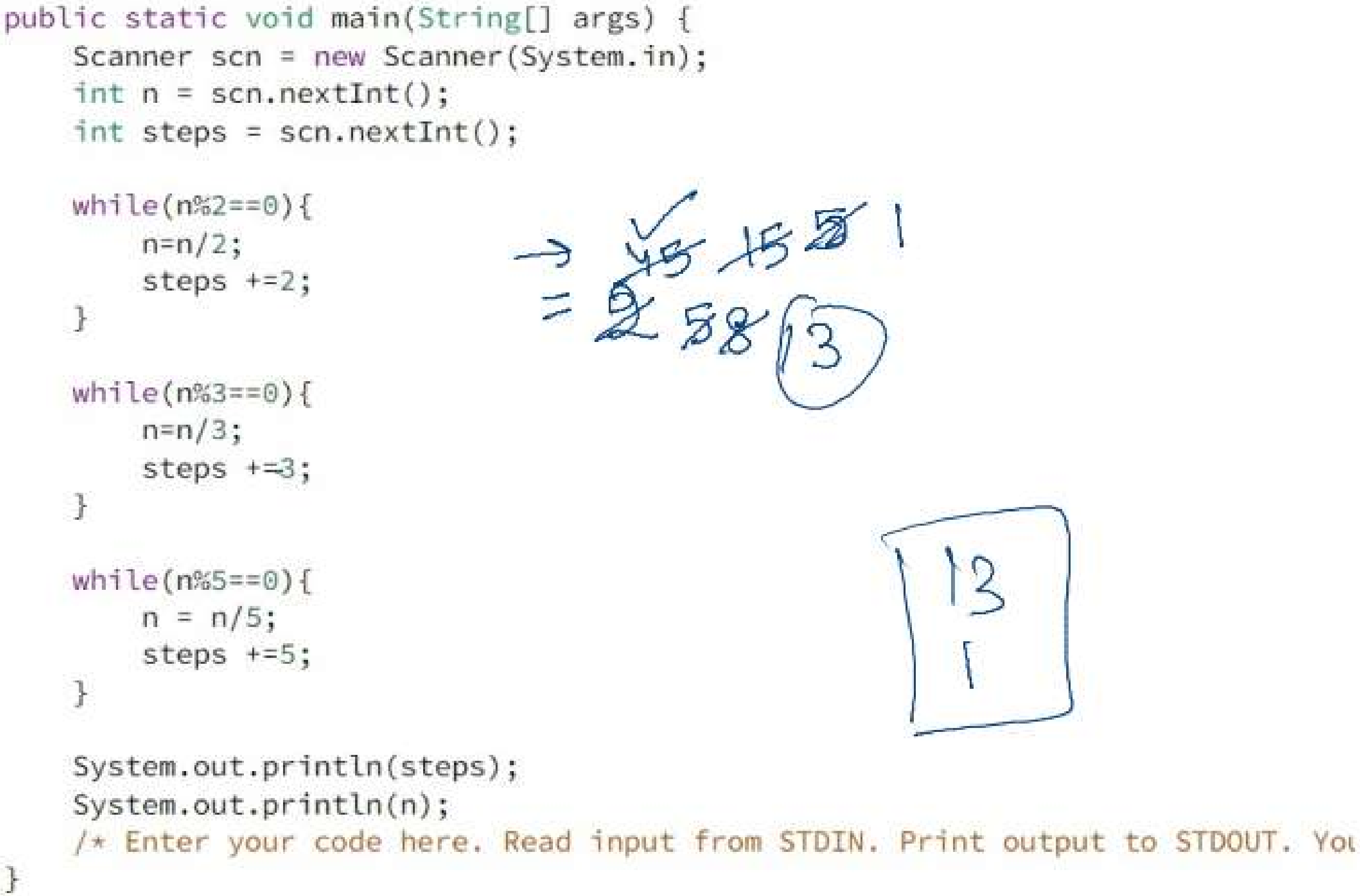
n = 45

steps = 2

if 2 → divisible.
    step += 2;

if 3 – divisible
    step += 3;

if 5 – divisible;
    step += 5;



```
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    int steps = scn.nextInt();

    while(n%2==0){
        n=n/2;
        steps +=2;
    }

    while(n%3==0){
        n=n/3;
        steps +=3;
    }

    while(n%5==0){
        n = n/5;
        steps +=5;
    }

    System.out.println(steps);
    System.out.println(n);
    /* Enter your code here. Read input from STDIN. Print output to STDOUT. You
}
```

→ ~~45~~ ~~15~~ ~~5~~ 1

= ~~2~~ ~~5~~ ~~8~~ 13

13
1