

Write a function to check if an Armstrong number or not

Take **n** as an integer input.

Then write a function that takes in this **n** as an integer parameter and returns if **n** is an **Armstrong number** or

Sample Input 0

2
153
234

Sample Output 0

true
false



Armstrong Number

$$n = \underline{\underline{153}}$$

$$1^3 + 5^3 + 3^3 = 1 + 125 + 27 = \underline{\underline{153}}$$

```

1 import java.io.*;
2 import java.util.*;
3
4 public class Solution {
5
6     public static boolean isArmstrong(int n){
7         int originalNumber = n;
8         int newNumber = 0;
9         while(n > 0){
10             int d = n % 10;
11             newNumber += d*d*d;
12             n /= 10;
13         }
14         return originalNumber==newNumber;
15     }
16
17     public static void main(String[] args) {
18         Scanner scn = new Scanner(System.in);
19         int t = scn.nextInt();
20         for(int i = 1; i <= t; i++){
21             int n = scn.nextInt();
22             boolean ans = isArmstrong(n);
23             System.out.println(ans);
24         }
25     }
26 }

```

$$\begin{array}{lcl}
 \text{Or Num} = \cancel{153} & 15 & \\
 \text{new Num} = \cancel{0} \cancel{27} & 15/2 & \textcircled{53} \\
 n = 15/3 \times 6 \times 0 & 153 > 6 &
 \end{array}$$

$$d = 3$$

$$15 > 0$$

$$\underline{153 = 153.}$$

$$d = 5$$

$$1 > 0$$

$$\textcircled{070} \times$$

$$nN \neq nN + 125$$

$$153$$

$$d = 1$$

Find GCD 3

GCD/ HCF

12, 36.

(12) → 1 2 3 4 6 (12)
(36) → 1 2 3 4 6 9 (12) 18 36

highest number
that which can divide both numbers

4, 6

4 → 1 2 4

6 → 1 2 3 6

100, 35

100 → 1 2 (5) 10 20 25 50 100

35 → 1 (5) 7 35

$$\underline{\underline{12, 36}}$$

$$\underline{x, y}$$

$$12 \rightarrow 1 \quad 2 \quad 3 \quad 4 \quad 6 \quad 12$$

$$36 \rightarrow 1 \quad 2 \quad 3 \quad 4 \quad 6 \quad 9 \quad 12 \quad 18 \quad 36$$

$$\text{gcd.} \rightarrow \text{min} \rightarrow (12)$$

$$\begin{matrix} x \\ y \end{matrix} \rightarrow \underline{\text{min.}}$$

$$\left[\begin{matrix} i \\ 1 \end{matrix} \right] \xrightarrow{\quad \quad \quad} \text{min}$$

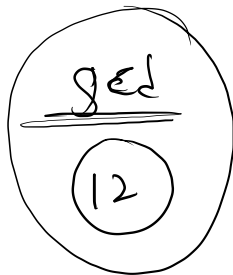
$$\underbrace{x \% i == 0} \quad \& \quad \underbrace{y \% i == 0}$$

$$\underline{\underline{\text{gcd} = i}}$$

gcd

12, 36

12 $\overline{) 36}$



[① ————— min]

5 → ① 5

7 → ① 7

```

1 import java.io.*;
2 import java.util.*;
3
4 public class Solution {
5     public static int gcd(int x, int y){
6         int min = Math.min(x,y);
7         int ans = 0;
8         for(int i = 1; i <= min; i++){
9             if(x % i == 0 && y % i == 0){
10                 ans = i;
11             }
12         }
13         return ans;
14     }
15
16     public static void main(String[] args) {
17         Scanner scn = new Scanner(System.in);
18         int t = scn.nextInt();
19         for(int i = 1; i <= t; i++){
20             int x = scn.nextInt();
21             int y = scn.nextInt();
22
23             int ans = gcd(x, y);
24             System.out.println(ans);
25         }
26     }
27 }

```

4 6

$$x = 12$$

$$y = 36$$

$$\min = 12$$

$$\text{ans} = \cancel{0} \cancel{1} \cancel{2} \cancel{3} \cancel{4} \cancel{6} 12$$

$$\begin{array}{ll}
 i=1 & 1 \leq 12 \\
 =2 & 2 \leq 12 \\
 3 & 3 \leq 12
 \end{array}$$

4
5
6
⋮

$$(12)$$

$$(12 \leq 12) \checkmark$$

$$12 \% 12 == 0$$

T

$$36 \% 12 == 0$$

T

```

1 import java.io.*;
2 import java.util.*;
3
4 public class Solution {
5     public static int gcd(int x, int y){
6         int min = Math.min(x,y);
7         int ans = .0;
8         for(int i = 1; i <= min; i++){
9             if(x % i == 0 && y % i == 0){
10                 ans = i;
11             }
12         }
13         return ans;
14     }
15
16     public static void main(String[] args) {
17         Scanner scn = new Scanner(System.in);
18         int t = scn.nextInt();
19         for(int i = 1; i <= t; i++){
20             int x = scn.nextInt();
21             int y = scn.nextInt();
22
23             int ans = gcd(x, y);
24             System.out.println(ans);
25         }
26     }
27 }

```

$$x = 4$$

$$y = 6$$

lmao

$$\min = 4$$

$$\text{ans} = 4 \times 2$$

$$i = 1$$

$$2$$

$$1 \leq 4$$

$$2 \leq 4$$

$$3$$

$$4$$

$$x \cdot 1 = 0$$

$$6 \cdot 1 = 0$$

$$x \cdot 2 = 0$$

$$6 \cdot 2 = 0$$

Prime checker 2

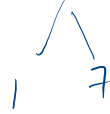
Sample Input 0

1
5

Sample Output 0

Yes

$n = 7 \rightarrow \text{prime}$



$n = 7$



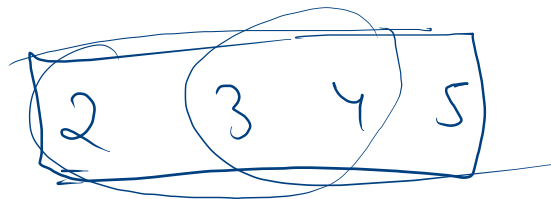
$$7 \% i == 0$$

$\rightarrow \text{Not prime}$

eg.

$n=6$

[1]



[6]

~~$6 \div 2 == 0$~~ \rightarrow not

```
1 import java.io.*;
2 import java.util.*;
3
4 public class Solution {
5
6     public static boolean isPrime(int n){
7         for(int i = 2; i < n; i++){
8             if(n % i == 0){
9                 return false;
10            }
11        }
12        return true;
13    }
14
15
16    public static void main(String[] args){
17        Scanner scn=new Scanner(System.in);
18        int t=scn.nextInt();
19        for(int i=1;i<=t;i++){
20            int n=scn.nextInt();
21            boolean ans = isPrime(n);    //true is number is prime otherwise false
22            if(ans == true){
23                System.out.println("Yes");
24            }else{
25                System.out.println("No");
26            }
27        }
28    }
29 }
```

$i = 1 \xrightarrow{\hspace{10em}} \leq n$

$n \% i \neq 0$ → print

10

7

①

①

2

7

S



6

12

$\rightarrow n=6.$

```
1 import java.io.*;
2 import java.util.*;
3
4 public class Solution {
5
6     public static void main(String[] args) {
7         Scanner scn = new Scanner(System.in);
8         int n = scn.nextInt();
9         for(int i = 1; i <= n; i++){
10             if(n % i == 0){
11                 System.out.println(i);
12             }
13         }
14     }
15 }
```

Divide n by 2 3 5 and tell steps

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.

$n = 30$ 15 5 1

steps = ~~0~~ ~~2~~ ~~5~~ 10

$n = 7$

step = 3

$n = 18$ 9 3 1

steps = ~~0~~ ~~3~~ ~~10~~ 13

13

1

i/p { n
(int) steps

o/p { steps ✓ 10
n 1

Sample Input 0

30
0

Sample Output 0

10
1

```
1 import java.io.*;
2 import java.util.*;
3
4 public class Solution {
5
6     public static void main(String[] args) {
7         Scanner scn = new Scanner(System.in);
8         int n = scn.nextInt();
9         int steps = scn.nextInt();
10
11         //try diving by 2 again nd again
12         while(n%2 == 0){
13             n /= 2;
14             steps += 2;
15         }
16         //with 3
17         while(n % 3 == 0){
18             n /= 3;
19             steps += 3;
20         }
21         //with 5
22         while(n % 5 == 0){
23             n /= 5;
24             steps += 5;
25         }
26
27         System.out.println(steps);
28         System.out.println(n);
29     }
30 }
```

Arrays

→ First D.S.

problem.

Class.

→ Test.

1 wrong Question.

8 students.

⇒

8 variable
to store

8 student
marks.

marks₀ = 12

marks₁ = 16

⋮ = 14

⋮ = 7

marks₇ = 9

mark₀ + 2

marks₁ + 2

marks₂ + 2

Array: \rightarrow DS. which store similar kind of data at contiguous memory location.

marks

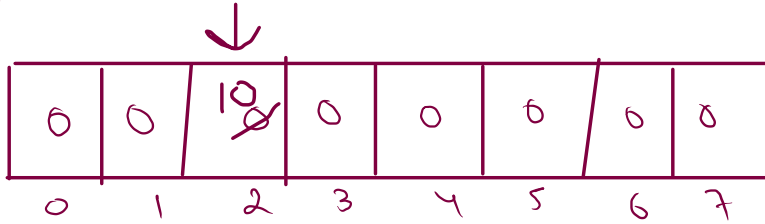
12	14	9	7	12	16	21	8
0	1	2	3	4	5	6	7


```
//Initialize array  
int [] marks = new int[8];  
}
```

data
type

name of
array

marks



int \rightarrow 0

```
1 public class Main
2 {
3
4     public static int sum(int x, int y){
5
6     }
7     public static void main(String[] args) {
8         int age = 52;
9
10        sum(10,20);
11        |
12        //Initialize array
13
14        int [] marks = new int[8];
15        System.out.println(marks[2]);
16
17        marks[2] = 18;
18
19
20        System.out.println(marks[2]);
21
22    }
23 }
```

