code armstrong in range

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

    for (int i = x; i <= y; i++) {
        if (solve(i) == true) {
            System.out.println(i);
        }
    }
}

public static boolean solve(int num) {
    int count = noOfDigits(num);

    int ans = 0;
    int temp = num;
    while (num > 0) {
        int rem = num % 10;
        ans = ans + (int)Math.pow(rem, count);

        num = num / 10;
    }
    return (temp == ans);
}

public static int noOfDigits(int num) {
    int count = 0;

    while (num > 0) {
        num = num / 10;
        count++;
    }

    return count;
}
```

false == true ×

1234 → count = 4

ans = 0
temp = 1234

rem = 4
ans = 0 + (4)^4

n = 123
rem = 3
ans = (4)^4 + (3)^4

n = 12
rem = 2
ans = 4^4 + 3^4 + 2^4

n = 1
rem = 1
ans = 4^4 + 3^4 + 2^4 + 1^4

n = 0

= Math.max(4, 3) = 4

(int) Math.pow(a, b) = a^b

2, 4 = 2^4 = 16.0

Find GCD 3

↳ greatest common division

greatest no. which going to divide both values

$x = 100, \quad y = 35$

3, 5

```
for( int i=1; i<=x && i<=y ; i++){
        if( x%i==0 && y%i==0){

        }
}
```

Find GCD 3

```
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int x = scn.nextInt();
    int y = scn.nextInt();
    int ans = findGCD(x, y);
    System.out.println(ans);
}
public static int findGCD(int x, int y) {
    int ans = 0;
    for (int i = 1; i <= x && i <= y ;i++) {
        if ( x % i == 0 && y % i == 0 ) {
            ans = i;
        }
    }
    return ans;
}
```

x = 100, y = 35

i = 1, ans = 1

i = 2,

i = 3,

= 4,

i = 5, ans = 5

= 6

= 7

:

:

i =

= 35

Prime checker 2

(a no. which is divisible by either 1 or itself)

- t = 5
  - 17 → Yes
  - 8 → No
  - 3 → Yes
  - 1 → No
  - 22 → No

n = 17

2, 3, 4, 5, ----, 16

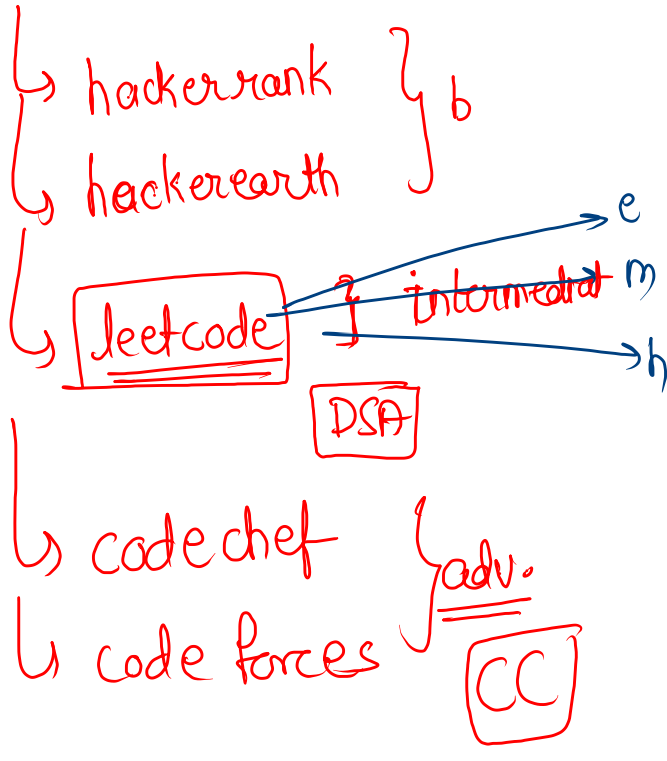
n

2, ⟶ n-1

code

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

    for (int i = 0; i < t; i++) {
        int n = scn.nextInt();
        boolean ans = checkPrime(n);
        if ( ans == true ) {
            System.out.println("Yes");
        } else {
            System.out.println("No");
        }
    }
}
public static boolean checkPrime(int num) {
    for (int i = 2; i <= num - 1; i++) {
        if ( num % i == 0 ) {
            return false;
        }
    }
    return true;
}
```



Divide n by 2 3 5 and tell steps

steps

2	2472
2	1236
2	618
	309

steps = 3

3	309
	103

steps = 1

103

steps = 0

code

```java
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);
}
```