→ Inbuilt f^n

= $\text{Math.pow}(\overset{\checkmark}{a}, \overset{\checkmark}{b});$

= a^b

= Math.pow(2, 3); // 8 $\boxed{2^3}$

---

int ans (double)

ans = ans + (int) Math.pow(rem, digits);

Rotate 7-digit number to right by three

$n = \underbrace{1234}\underbrace{567}$

```
a = n % 1000     // 567
n = n / 1000     // 1234
```

$ans = \boxed{567}\,1234$

```
ans = (a * 10000) + n
```

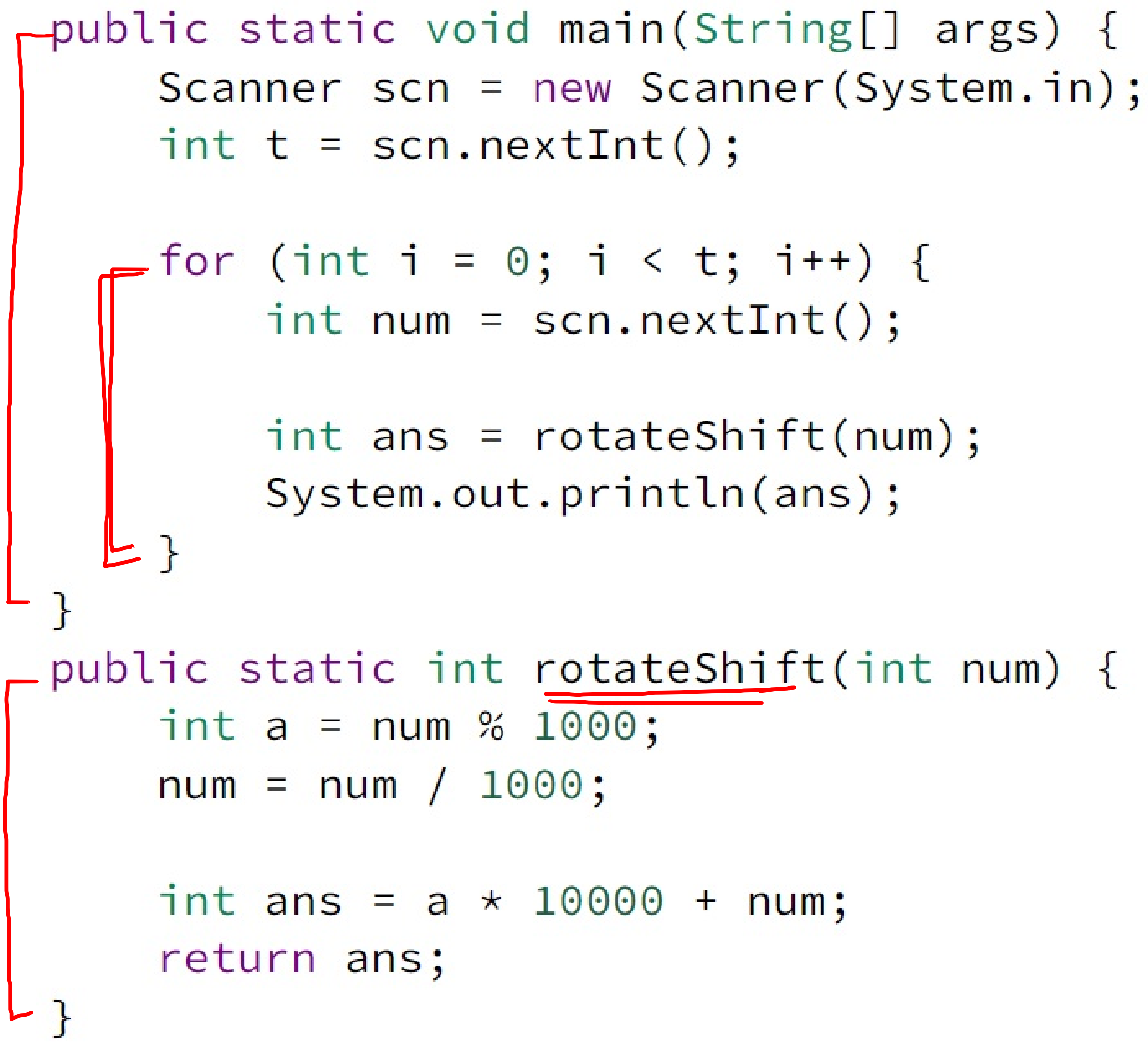
$\Rightarrow 567 * 10000$

$\Rightarrow 5670000$

$+ \quad 1234$

$\Rightarrow 5671234$

Code



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

    for (int i = 0; i < t; i++) {
        int num = scn.nextInt();

        int ans = rotateShift(num);
        System.out.println(ans);
    }
}
```

logic

```
public static int rotateShift(int num) {
    int a = num % 1000;
    num = num / 1000;

    int ans = a * 10000 + num;
    return ans;
}
```

Find GCD / HCF

$t = 2$

$x = 100, \quad y = 35$

$x = 20, \quad y = 7$

$$\begin{array}{r|l} 5 & 100, 35 \\ \hline & 20, 7 \end{array}$$

GCD :-

greatest no. which can divide both no.'s entirely

$$\begin{array}{r|l} 1 & 20, 7 \\ \hline & 20, 7 \end{array}$$

x = 100 , y = 15

x % i == 0 && y % i == 0 ans = 1

✓ i = 1 , (100 % 1 == 0 && 15 % 1 == 0) ✓ ans = 1

i = 2 , (100 % 2 == 0 && 15 % 2 == 0) ✗

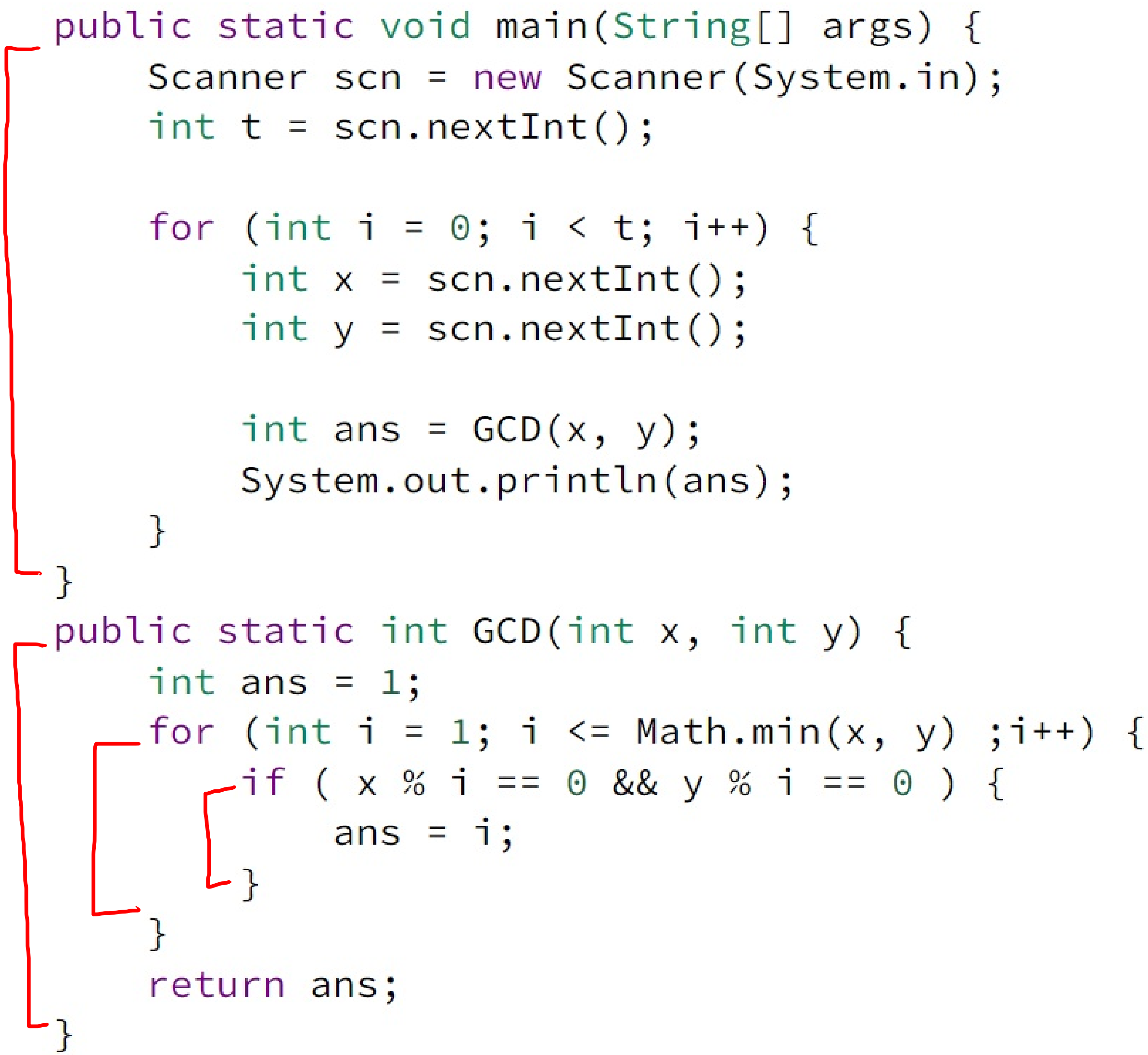
i = 3 , (100 % 3 == 0 && 15 % 3 == 0) ✗

i = 4 , (100 % 4 == 0 && 15 % 4 == 0) ✗

✓ i = 5 , (100 % 5 == 0 && 15 % 5 == 0) ✓ ans = 5

smallest value

i <= Math.min(x, y)



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

    for (int i = 0; i < t; i++) {
        int x = scn.nextInt();
        int y = scn.nextInt();

        int ans = GCD(x, y);
        System.out.println(ans);
    }
}
public static int GCD(int x, int y) {
    int ans = 1;
    for (int i = 1; i <= Math.min(x, y) ;i++) {
        if ( x % i == 0 && y % i == 0 ) {
            ans = i;
        }
    }
    return ans;
}
```

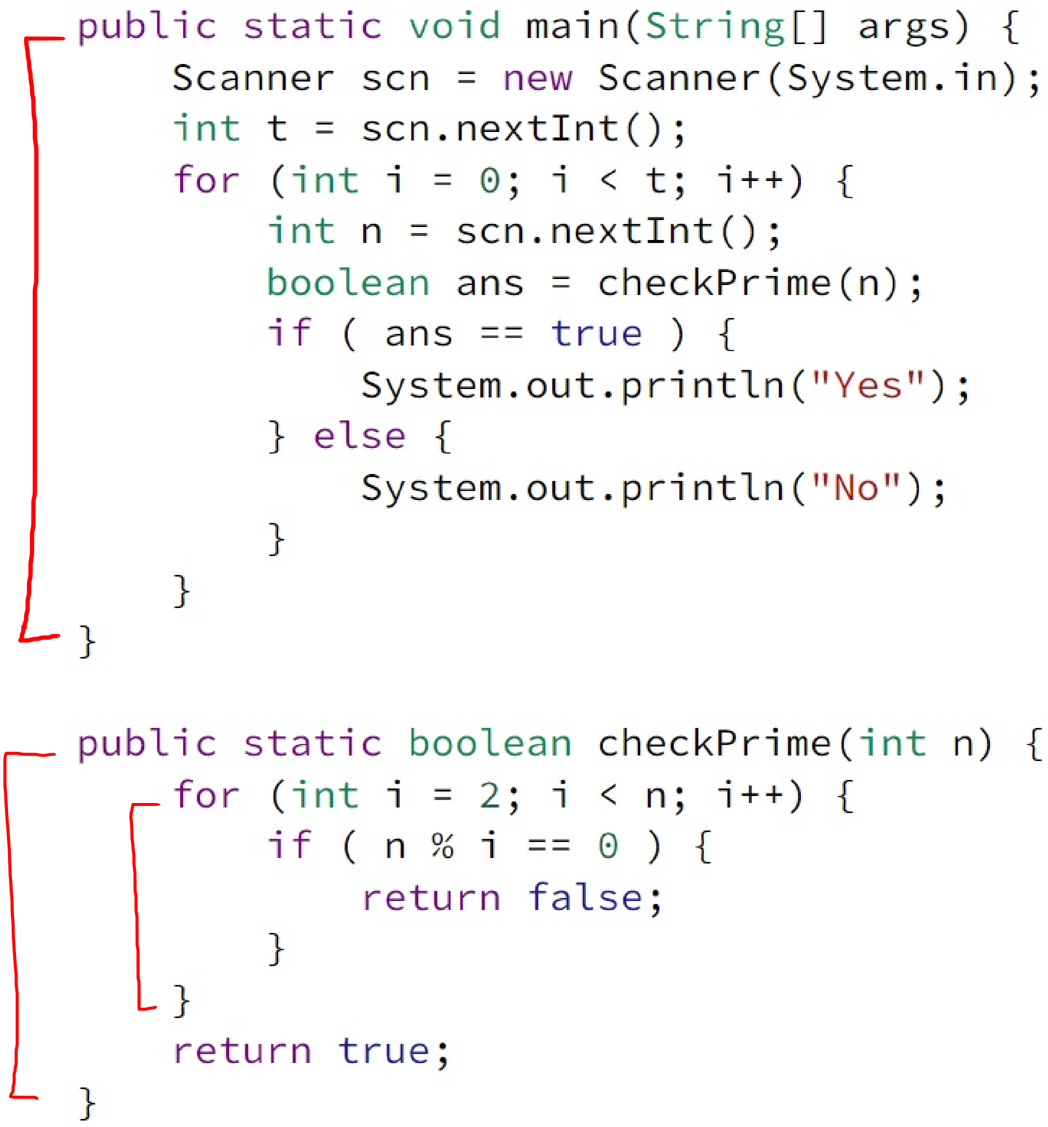
Prime checker

(no. which can only be divisible by 1 or itself)

t = 5

n = 20	false		i = 2, ... , 19
n = 13	true	→	i = 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12
n = 93	false		
n = 8	false		
n = 1	false		

Code



```
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 n) {
    for (int i = 2; i < n; i++) {
        if ( n % i == 0 ) {
            return false;
        }
    }
    return true;
}
```