Ques I/P 1 2 3 4 5 ⇒ O/P 12345 ⇒ 54321

1*10+2 = 12*10+3 = 123*10+4 = 1234*10+5
onum = 12345

rnum = 54321

5*10+4 = 54*10+3 = 543*10+2
= 5432*10+1
= 54321

```
public class Solution {

    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);
        int n = scn.nextInt();
        int i=1;
        int onum =0;
        while(i<=n){
            int digit = scn.nextInt();
            onum = onum*10+digit;
            i++;
        }
        System.out.println(onum);
        int rnum =0;
        while(onum>0){
            int rem = onum %10;
            rnum = rnum*10+rem;
            onum /=10;
        }
        System.out.println(rnum);
        /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your clas
```

n = 3 → 4, 3, 2
i = 1 2 3
digit = 4 3 2
onum = 0*10+4 = 4*10+3 = 43*10+2 = 432
432 → reverse → 234
rnum = 0*10+2 = 2*10+3 = 23*10+4 = 234
rem = 2 3 4

Qus Armstrong number:-

n=(153) # of digit = 3

$= 1^3 + 5^3 + 3^3 = 1 + 125 + 27$

$\# = 153$

↳ find digits
↳ Do power operation
↳ Sum operation - (num)
↳ if num == onum → true (Armstrong)
→ false (# is not)

```
    int n = scn.nextInt();
    int i=1;
    while(i<=n){
      int onum = scn.nextInt();
      int countDigits = countNoDigits(onum);
      boolean ans = armstrong(onum,countDigits);
      System.out.println(ans);
      i++;
    }
}

public static int countNoDigits(int val){
   int count=0;

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

   return count;
}

public static boolean armstrong(int val, int noOfDigits){
  int sum =0;
  int temp=val;
  while(val>0){
     int rem = val%10;
     sum += Math.pow(rem,noOfDigits);
     val /=10;
  }

  return sum == temp;
}
```

onum 153 ✓

153 153 153

[153 == 153]
Sum == temp

armstrong =

val = 153 15 1 0
# of's = 3
Sum = 0+27 = 27+125 = 152+1
temp = 153
rem = 3 5 1

$5^3 = 27+125 = 152+1^3$
$= 153$

onum = 153
countdigits = 3
ans = true

main

true

true

Qu val = 1234567

int lastdigit = val % 1000; 567

int first4digits = val / 1000 = 1234

567 * 10000 + 1234

5670000 + 1234

= 5671234

Pass

```
public static void main(String[] args) {

    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    int i =1;
    while(i<=n){
        int onum = scn.nextInt();
        int last3Digits= onum%1000;
        int first4Digits = onum/1000;
        int ans = last3Digits*10000+first4Digits;
        System.out.println(ans);
        i++;
    }
    /* Enter your code here. Read input from STDIN. Print output to
}
```

Que GCD

25, 35 = 5

Largest common factor

i = 2 to 40

if(a%i==0 && b%i==0){
commonfactor = i
}

= 40 , 20 =

1, 2, 4, 5, 8, 10, 20, 40

1, 2, 4, 5, 10, 20

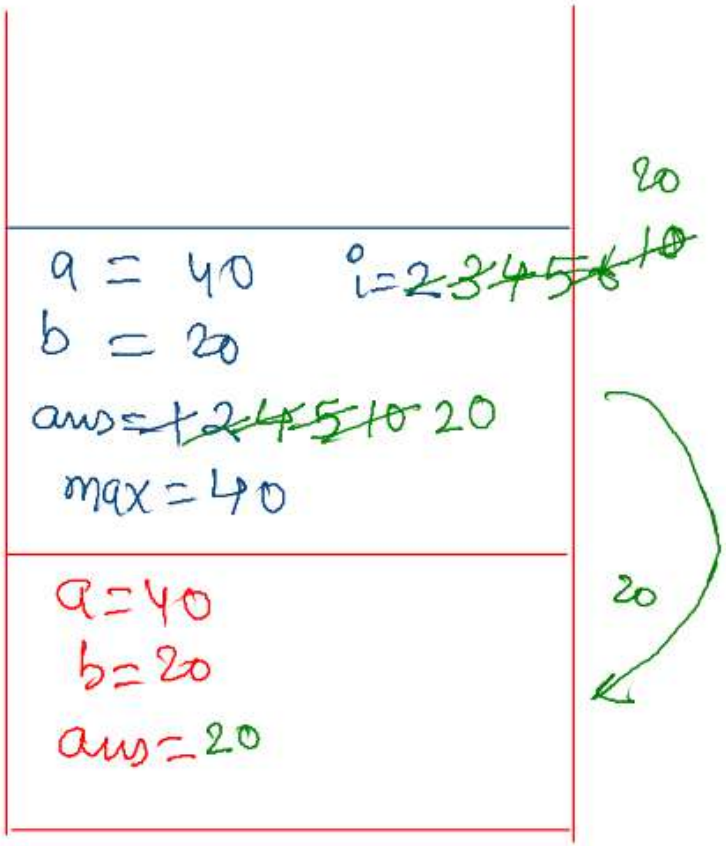
⇒ Any GCD = 20

(40, 20)

```java
public class Solution {

    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);
        int n = scn.nextInt();
        int i=1;
        while(i<=n){
            int a = scn.nextInt();
            int b = scn.nextInt();
            int ans = GCD(a,b); 20
            System.out.println(ans);
            i++;
        }
        /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your GCD
    }

    public static int GCD(int a , int b){
      int ans =1;
        int max = Math.max(a,b);
        for(int i=2;i<=max;i++){
          if(a%i==0 && b%i==0){
                ans =i;
            }
        }

        return ans;20
    }
```

20

main



Q4 n = 27

↳ Prime ← Yes / No

2 ↔ 26

1) Sudo Code

```
for( i=2; i<n; i++){
    if(n%i ==0){
        System.out.print("No");
    }
}
else
    sys(Yes);
```

```java
public static void main(String[] args) {

    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    int i=1;
    while(i<=n){
        int val = scn.nextInt();
        boolean ans = isPrime(val);
        if(ans){
            System.out.println("Yes");
        }else{
            System.out.println("No");
        }
        i++;
    }
    /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your cl
}

public static boolean isPrime(int val){
    boolean  ans = true;

    for(int i =2;i<val;i++){
        if(val%i==0){
            ans = false;
            break;
        }
    }

    return ans;
}
```

val = 9

No