

You are given a number  $n$ , Write a program to **reverse** digits of  $n$ .

#### Input Format

The first line of input will contain an integer  $T$  — the number of test cases. The description of  $T$  test cases follows.

For each test case, you will get an integer input.

#### Constraints

$$1 \leq T \leq 10^4$$

$$1 \leq n \leq 10^4$$

#### Output Format

Print the reverse.

#### Sample Input 0

```
4
1234
5678
4567
89
```

#### Sample Output 0

```
4321
8765
7654
98
```

#### Explanation 0

Print the reverse number .

1 2 3 4

4 3 2 1

Integer

∴

1 2 3 4  
• • • •

int store = 4 3 2 1

method

1 2 3 4

1 2 3 4 ∴ 10 ⇒ 4  
∴

store  
4 3 2 1

1 2 3 ∴ 10 ⇒ 3

1 2 ∴ 10 ⇒ 2

1 ∴ 10 ⇒ 1

store = 0

store = store \* 10 + rem

store = 4

store \* 10 + rem

40 + 3 ⇒ 43 \* 10 + 2 ⇒ 432 \* 10 + 1 ⇒ 4321

#### Submitted Code

Language: Java 15

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

Write a Program to check whether a given number **n** is a **Palindrome** or not.

**Definition of Palindrome:-** A number which is equal to the reverse of the number.

**Input Format**

For each test case, you will get an positive integer input.

**Constraints**

10 <= n <= 10<sup>4</sup>

**Output Format**

If number is a Palindrome then Print "YES"

If number is not a Palindrome number then Print "NO"

**Sample Input 0**

3  
121  
123  
333

**Sample Output 0**

YES  
NO  
YES

**Explanation 0**

Number 121 is equal to its **reverse** so, the answer will be YES

121  
121

MADAM  
MADAM

DAD  
DAD

RACECAR  
RACECAR

YES

NO

121

121

123  
321

333  
333

NAYAN

NAYAN

NITIN

NITIN

EYE

EYE

Submitted Code

Language: java 15

```
1 import java.io.*;
2 import java.util.*;
3
4 public class Solution {
5
6     public static void main(String[] args) {
7         Scanner sc = new Scanner(System.in);
8         int t = sc.nextInt();
9
10        for(int i=0;i<t;i++){
11            System.out.println(reverse(sc.nextInt()));
12        }
13    }
14    public static String reverse(int n){
15        int val =n;
16        int result=0;
17        while(n>0){
18            int rem = n%10;
19            result=result*10+rem;
20            n/=10;
21        }
22        if(result == val)return "YES";
23        else return "NO";
24    }
25 }
```

Reversing

val = n

n = 1234 / 10

n = 123 / 10

n = 12 / 10

n = 1 / 10

n / 10 . n = ...

n

result

(result == val) YES  
NO

You are given a 6 digit number  $n$  , you have to pick the **last 2** digits of the number of and put them in the starting.

Your task is to write a Program for the above problem and Print the Transformed number.

Input Format

For each test case, you will given a number  $n$  as an integer input.

Constraints

Given  $n$  should be 6 digits Number .

Output Format

Print the transformed number.

Sample Input 0

123456

Sample Output 0

561234

Explanation 0

123456 is given, then this number should transform to 561234.

123456

56

561234

2368

6823

$n \% 100$

= 56

$n / 100$

561234

while

561234

1234

1234

$n = 123456$

$rem = n \% 100$

$rem = 56$

$n /= 100$

$n = 1234$

$n = 1234$

$rem = 560000$   
 $1234 \Rightarrow 561234$

$rem + n \rightarrow 561234$  Gan2

$qnt = n$  while( $n > 0$ ) {  
 $n = 1234$   $n /= 10$ ;

4

56

2 (4)

560000

Submitted Code

```
Language: Java 15
1 import java.io.*;
2 import java.util.*;
3
4 public class Solution {
5
6     public static void main(String[] args) {
7         Scanner sc = new Scanner(System.in);
8
9
10        System.out.print(transform(sc.nextInt()));
11    }
12
13    public static int transform(int n){//123456
14        int rem = n%100; //56
15        int qnt = n/100;//1234
16        //561234 rem*qnt
17        //560000+1234=561234
18
19        int copy = qnt;
20        int count=0;
21        while(copy>0){
22            count++;
23            copy/=10;
24        }
25        //count=4
26        while(count>0){
27            rem*=10;
28            count--;
29        }
30        //rem =560000;
31
32        int ans = rem+qnt;
33        //560000+1234 = 561234
34        return ans;
35
36    }
37 }
38 }
```