Reverse Integer

Easy

49577520Add to ListShare

Given a signed 32-bit integer `x`, return `x` *with its digits reversed*. If reversing `x` causes the value to go outside the signed 32-bit integer range `[-2^31, 2^31 - 1]`, then return `0`.

Assume the environment does not allow you to store 64-bit integers (signed or unsigned).

Example 1:

```
Input: x = 123

Output: 321
```

Example 2:

```
Input: x = -123

Output: -321
```

Example 3:

```
Input: x = 120

Output: 21
```

Example 4:

```
Input: x = 0

Output: 0
```

Constraints:

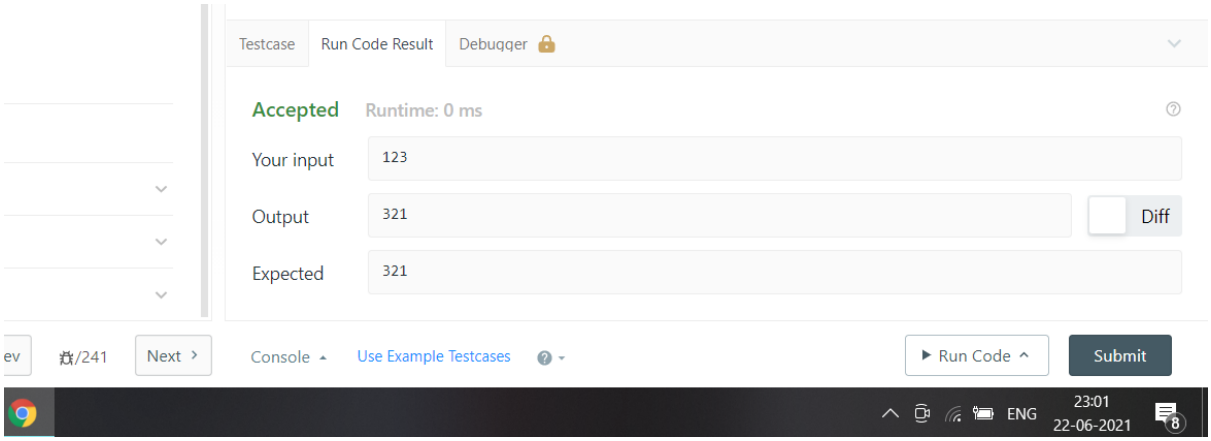
- `-2^31 <= x <= 2^31 - 1`

SOLUTION

```
class Solution {

   public int reverse(int x) {

      boolean flag = false;

      if(x < 0){

         flag = true;

         x = -x;
```

```
        }
        int prev_rev = 0, rev = 0;
        while(x != 0){
            int curr_digit = x%10;

            rev = rev*10 + curr_digit;

            if((rev - curr_digit)/10 != prev_rev){
                return 0;
            }

            prev_rev = rev;
            x = x/10;
        }
        if(flag){
            return -rev;
        }
        else{
            return rev;
        }


    }
}
```

OUTPUT


Testcase
Run Code Result
Debugger
Accepted
Runtime: 0 ms
Your input
123
Output
321
Diff
Expected
321
ev
/241
Next
Console
Use Example Testcases
Run Code
Submit
ENG
23:01
22-06-2021
8