(2) → Palindrome number -

1. Question -

Given an integer (x), return true if x is palindrome integer

2. Examples -

121 → true

-121 → false

10 → false

121

<121->

<01> ⇒ 1

follow up → Could you solve it without converting the integer to a string

3. Intuition / Concept -

- convert integer to string
- check if string is palindrome

4. Brute force solution -

x = 12321 ⇒ 5 digits

0 1 2 3 4

left, right

x = 66 1 2 3 2 1 22

left, right, right

Time complexity = $O(\log_{10} x)$

Space complexity = $O(\log_{10} x)$? we created a string

x = 99999 <$\log_{10} x$> spaces → 5

int → string

5. Optimized sol^n ? -

Yes, there is.

In terms of time, we can't do much.

Space → ?? ✓

- when number is 0, return true
- when negatives, false
- When number is divisible by 10, return false

1. 0 ✓

2. - ; x-
-x ; y-

10, 100, 1000, 90

01 ✗ 001 ✗ 0001 ✗ 09 ✗

300

3.

$x = 12321$; $x = 12$; $y = 123$; $x == y/10$ ✓

- reverse the last half, and compare it with original's first half.

odd digits = 12321 → 123 (rev) | even digits | 123321 = x ; y = 123 ; x = 123

x = 12

odd no. of digits

x = 12321

rev = 0

→ rev = rev × 10 + d = 1

⇒ rev = rev × 10 + d = 12

⇒ rev = rev × 10 + d = 12 × 10 + 3 = 123

x = 1232

x = 123 → 12

even -

rev = rev × 10 + d

x = 12321 > rev (0)

rev = 0

x = 12321

rev = 0 × 10 + 1 } 1

x = x/10 ; 1232

rev = 1 × 10 + 2 = 12

x = x/10 = 123

x > rev

rev = 12 × 10 + 3 = 123

x = x/10 = 12

⟨x == (rev/10)⟩

⟨x == rev⟩

$T = O(\log_{10} x)$

$S = O(1)$

6 → Time and space Complexity -

Brute force =

$T = O(\log_{10} x)$

$S = O(\log_{10} x)$

Optimized solⁿ

$T = O(\log_{10} x)$

$S = O(1)$

7 → Code walkthrough -

```
public boolean isPalindrome (int x)
    if (x == 0)
        return true
    if x < 0
        return false
    if (x % 10 == 0)
        return false

    int rev = 0
    while (x > rev)
        rev = rev × 10 + (x % 10)
        x = x / 10
    return (x == rev) || (x == (rev / 10))
```

-x, y -

90 → 09

x = 12 21

x = 12, rev = 12

x = 12 3 21

rev = 123

x = 12

x == rev/10