* Single number :-

→ We are given a non-empty integer array where every element appears twice except for one.

→ Return that single element.

* Test Cases :-

Input :- [2, 2, 1]
Output :- 1

Input :- [4, 1, 2, 1, 2]
Output :- 4

Input :- [1]
Output :- 1

* Constraints :-

→ Linear runtime required

→ Constant extra space required

* Solution :-

→ If there were no space constraints, we could have used a frequency map approach.

→ A better approach can be using bitwise XOR operator.

XOR

a	b	result
0	0	0
0	1	1
1	0	1
1	1	0

If we XOR two same values → 0,

XOR with 0 returns the same value.

$4 \wedge 0 = 4$

arr = [4, 1, 2, 1, 3, 2, 3]

→ If we XOR all elements $(4 \wedge 1 \wedge 2 \wedge 1 \wedge 3 \wedge 2 \wedge 3)$, the elements which are repeating twice will return 0.

→ Repeating elements will cancel out each other.

→ Ultimately, we will be left with a single element.

$TC = O(n)$
$SC = O(1)$