Single Number

solved by	Senan
Platform	GeeksForGeeks
difficulty	Easy
tags	Bit Manipulation
language	C++
solved on	@18/10/2024
link	https://www.geeksforgeeks.org/problems/single-number1014/1
Completion	☑

Intuition

In this problem, we need to identify the element that appears an odd number of times in an array where all other elements appear an even number of times. XOR (`^`) is particularly useful here because it has the property that:

- $a \oplus a = 0$ (XORing a number with itself cancels it out)
- $a \oplus 0 = a$(XORing a number with 0 results in the number itself)

By XORing all elements in the array, the pairs of numbers (that occur an even number of times) will cancel each other out, and only the number that appears an odd number of times will remain.

Approach

1. **Traverse the array**:

   Iterate through each element in the array and XOR it with an initially set result variable (`answer`). Since pairs of numbers cancel out with XOR, the result will end up being the number that occurs an odd number of times.

2. **Properties of XOR**:

   - $a \oplus a = 0$ cancels out all numbers that appear an even number of times.
   - The final result will be the number that appears an odd number of times.

Complexity

Time Complexity:

- **O(n)**, where `n` is the number of elements in the array. We traverse the array once, and each XOR operation is constant time, so the total time complexity is linear.

Space Complexity:

- **O(1)**. We only use a constant amount of extra space to store the result.

Code

```
class Solution {
  public:
    int getSingle(vector<int>& arr) {
```

```
        int answer = 0;
        // XOR all elements
        for(auto elem: arr) {
            answer ^= elem;
        }
        return answer;
    }
};
```