41. First Missing Positive

Solved

Hard | Topics | Companies | Hint

Given an unsorted integer array `nums`. Return the *smallest positive integer* that is *not present* in `nums`.

You must implement an algorithm that runs in `O(n)` time and uses `O(1)` auxiliary space.

Example 1:

```
Input: nums = [1,2,0]
Output: 3
Explanation: The numbers in the range [1,2] are all in the array.
```

Example 2:

```
Input: nums = [3,4,-1,1]
Output: 2
Explanation: 1 is in the array but 2 is missing.
```

```swift
class Solution {
  func firstMissingPositive(_ nums: [Int]) -> Int {
      let n = nums.count
      var arr = Array(repeating:false, count:n+1)

      for i in nums {
          if (0 < i && i <= n) {
              arr[i] = true
          }
      }

      for i in (1..<arr.count) {
          if(arr[i] == false) {
              return i
          }
      }

      return n+1
  }
}
```