127. Valid Palindrome

```
class Solution {
    func isPalindrome(_ s: String) -> Bool {
        let clean = s.lowercased().filter{
$0.isLetter || $0.isNumber}

        return clean ==
String(clean.reversed())

    }
}
```

15. 3Sum

```
class Solution {
    func threeSum(_ nums: [Int]) -> [[Int]]
{
        var result = [[Int]]()
        let nums = nums.sorted()

        for i in 0..<nums.count{
            if(nums[i] > 0)
            {
                break
            }

            if i > 0 && nums[i] == nums[i-1]
            {
                continue
            }

            var left = i+1
            var right = nums.count - 1
            while left < right
            {
                let sum = nums[i] +
nums[left] + nums[right]
                if sum == 0{
                    result.append([nums[i],
nums[left], nums[right]])
                    while left < right &&
nums[left] == nums[left + 1] {
            left += 1
        }
```

```
       while left < right && nums[right] ==
nums[right - 1] {
           right -= 1
       }
                   left += 1
                   right -= 1
               }

               else if sum < 0 {
                   left += 1
               }
               else
               {
                   right -= 1

               }



           }
       }
```

```
        return result



    }
}
```

11. Container With Most Water

```
class Solution {
    func maxArea(_ height: [Int]) -> Int {
        var left = 0
        var right = height.count - 1
        var maxArea = 0

        while left < right {
            var area = min(height[left],
height[right]) * (right - left)

            maxArea = max(area, maxArea)

            if(height[left] < height[right])
            {
                left += 1
            }
            else
            {
                right -= 1
            }
        }
        return maxArea
    }
}
```