Day-19 Python DSA

Leetcode 152 Maximum Product Subarray

https://leetcode.com/problems/maximum-product-subarray/description/

Bruteforce

```
def maxProduct(nums):
    n= len(nums)
    maxi= float("-inf")
    for i in range(0,n):
        product=1
        for j in range(i,n):
            product *= nums[j]
            maxi= max(maxi, product)
    return maxi
nums = [2,3,-2,4]
maxProduct(nums)
```

TC – O(N^2)

SC- O(1)

Optimal

```
class Solution:
    def maxProduct(self, nums: List[int]) -> int:
        ans= nums[0]
        dpMin = nums[0]
        dpMax= nums[0]

        for i in range(1, len(nums)):
            num = nums[i]
            prevMin = dpMin
            prevMax= dpMax
            if num <0:
                dpMin = min(prevMax * num, num)
                dpMax= max(prevMin * num , num)
            else:
                dpMin = min(prevMin* num, num)
                dpMax= max(prevMax* num, num)

            ans = max(ans , dpMax)
        return ans
```

TC – O(N)

SC- O(1)