MAXIMUM PRODUCT SUBARRAY

☆ In this problem, we are supposed to return the subarray which has the maximum product.

Brute force solution is same as with all subarray questions, to find all subarrays using 3 loops and keeping a track of maximum product.

There are two optimal solutions to this problem out of which one is a modification to the Kadane's algo and the other has been discussed

In an array, we can have either all positives, or even no. of negatives or odd number of negatives or some zeroes as well. We start by splitting the array on zeroes and then just finding maximum of product of all subarrays formed.

Pseudocode :

```
maximumProductSubArray(arr, N) {
    pre, suff = 1, 1
    maxSum = INT_MIN
    for(i = 0 → N-1) {
        pre = pre * arr[i]
        suff = suff * arr[N-i-1]
        if (pre == 0) pre = 1
```

```
        if (suff == 0)  suff = 1
        maxSum = max(pre, suff, maxSum)
    }
    return maxSum
}
```