

KADANE ALGORITHM TO FIND MAXIMUM/MINIMUM SUBARRAY SUM IN AN ARRAY

```
class Solution {
public int maxAbsoluteSum(int[] nums) {
    int maxSum = Integer.MIN_VALUE;
    int minSum = Integer.MAX_VALUE;
    int currSum = 0;
    // Kadane's Algorithm to find maximum subarray sum
    for(int i = 0; i < nums.length; i++){
        currSum += nums[i];
        maxSum = Math.max(maxSum, currSum);
        currSum = (currSum < 0) ? 0 : currSum;
    }

    // Kadane's Algorithm to find minimum subarray sum
    currSum = 0;
    for(int i = 0; i < nums.length; i++){
        currSum += nums[i];
        minSum = Math.min(minSum, currSum);
        if(currSum > 0) currSum = 0;
    }
    return Math.max(Math.abs(maxSum), Math.abs(minSum));
}
}
```