

$$a = [5, -4, -2, 6, -1]$$

DATE / /  
PAGE

Q Find longest sum contiguous Subarray

A computer science algorithm that finds the max sum of a contiguous subarray in an array of numbers.

### Algorithms

① Initialize two variables:

currentSum and maxSum

② Iterate through the array.

③ For each element, calculate the currentSum adding the element to the previous currentSum.

④ If the currentSum becomes negative, reset it to zero.

⑤ Update maxSum if the currentSum is greater than maxSum.

⑥ The value of maxSum is the maximum sum of a contiguous subarray.

Boundary case:

```
for (int i=0; i<n; i++)
```

$O(n^2)$

int sum = 0

```
    for (int j=i; j<n; j++) {
```

sum = sum + arr[j];

```
    } // action
```

maxSum;

Kadane Algo

Sum = 0

maxi = -2

arr[0]

a)  $\rightarrow \text{sum} = \text{sum} + \text{arr}[i]$

b)  $\rightarrow \text{maxUpdate} \Rightarrow \text{maxi} = \max(\text{maxi}, \text{sum})$

c)  $\rightarrow \text{if } (\text{sum} < 0)$   
 $\quad \quad \quad \text{sum} = 0$

```
class Solution {
public:
    int maxSubArray (int arr[]) {
        int sum=0;
        int maxi=arr[0];
        for( int i=0 ; i<arr.size(); i++ ) {
            sum = sum + arr[i];
            maxi = max(maxi, sum);
            if( sum < 0 ) { sum = 0; }
        }
        return maxi;
    }
}
```