

Name: Aisha Muhammad Nawaz

Roll# 20L-0921

Section :- BSCS 4A1 DAA

Homework # 6 (Dr. Maryam Basir)
Kadane's AlgorithmTASK # 1

```

Max Sub Array Sum (A, n)
    global start = 1;
    global sum = A[1]
    Max sum [1] = A[1]
    for (i = 2 to n)
        if (Max sum [i-1] + A[i] > A[i])
            Max sum [i] = Max sum [i-1] + A[i]
        else
            Max sum [i] = A[i]
            global start = i
    if (global sum < max sum [i])
        global sum = max sum [i]
        global End = i
    return global sum

```

3

TASK # 2 Dry run Brute force Algorithm

starts here

i = 1 → j = 1 → Sub Array Sum = 2
 → Max sum = 2

i = 2 → j = 2 → Sub Array Sum = -4
 Max sum = 2

j = 1 → Sub Array Sum = -2
 Max sum = 2

max sum [2] = -2

Assumption :-
 This variable is equal to store max sum of all sub arrays.

These are different variables and arrays.

↳ store max sum out of all sub arrays ending at index i

continued

i = 3 → j = 3 → Sub Array Sum = 3
 Max sum = 3

j = 2 → Sub Array Sum = -1
 Max sum = 3

j = 1 → Sub Array Sum = 1
 Max sum = 3

max sum [3] = 3

$i = 4$
 $j = 4$
 Sub Array Sum = 4
 max sum = 4

 $j = 3$
 Sub Array Sum = 7
 max sum = 7

 $j = 2$
 Sub Array Sum = 3
 max sum = 7

 $j = 1$
 Sub Array Sum = 5
 max sum = 7
 max sum [4] = 7

 $i = 5$
 $j = 5$
 Sub Array Sum = -3
 max sum = 7

 $j = 4$
 Sub Array Sum = 1
 max sum = 7

 $j = 3$
 Sub Array Sum = 4
 max sum = 7

 $j = 2$
 Sub Array Sum = 0
 max sum = 7

 $j = 1$
 Sub Array Sum = 2
 max sum = 7
 max sum [5] = 4

 $i = 6$
 $j = 6$
 Sub Array Sum = 5
 max sum = 7

 $j = 5$
 Sub Array Sum = 2
 max sum = 7

 continue

$j = 4$
 Sub Array Sum = 6
 max sum = 7

 $j = 3$
 Sub Array Sum = 9
 max sum = 9

 $j = 2$
 Sub Array Sum = 5
 max sum = 9

 $j = 1$
 Sub Array Sum = 7
 max sum = 9
 max sum [6] = 9

 $i = 7$
 $j = 7$
 Sub Array Sum = -5
 max sum = 9

 $j = 6$
 Sub Array Sum = 0
 max sum = 9

 $j = 5$
 Sub Array Sum = -3
 max sum = 9

 $j = 4$
 Sub Array Sum = +1
 max sum = 9

 $j = 3$
 Sub Array Sum = 4
 max sum = 9

 $j = 2$
 Sub Array Sum = 0
 max sum = 9

 $j = 1$
 Sub Array Sum = 2
 max sum = 9
 max sum [7] = 4

 $i = 8$
 $j = 8$
 Sub Array Sum = 6
 max sum = 9

 $j = 7$
 Sub Array Sum = 1
 max sum = 9

 $j = 6$
 Sub Array Sum = 6
 max sum = 9

$$j=5 \rightarrow \text{SubArray Sum} = 3 \\ \text{max sum} = 9$$

$$j=4 \rightarrow \text{SubArray Sum} = 7 \\ \text{max sum} = 9$$

$$j=3 \rightarrow \text{SubArray Sum} = 10 \\ \text{max sum} = 10$$

$$j=2 \rightarrow \text{SubArray Sum} = 6 \\ \text{max sum} = 10$$

$$j=1 \rightarrow \text{SubArray Sum} = 8 \\ \text{max sum} = 10 \\ \text{max sum} \text{ is } 10$$

$$i=9 \rightarrow j=9 \\ \text{SubArray Sum} = -1 \\ \text{max sum} = 10$$

$$j=8 \rightarrow \text{SubArray Sum} = 5 \\ \text{max sum} = 10$$

$$j=7 \rightarrow \text{SubArray Sum} = 0 \\ \text{max sum} = 10$$

$$j=6 \rightarrow \text{SubArray Sum} = 5 \\ \text{max sum} = 10$$

$$j=5 \rightarrow \text{SubArray Sum} = 2 \\ \text{max sum} = 10$$

$$j=4 \rightarrow \text{SubArray Sum} = 6 \\ \text{max sum} = 10$$

Continued

$$j=3 \rightarrow \text{SubArray Sum} = 3+6=9 \\ \text{max sum} = 10$$

$$j=2 \rightarrow \text{SubArray Sum} = 5 \\ \text{max sum} = 10$$

$$j=1 \rightarrow \text{SubArray Sum} = 7 \\ \text{max sum} = 10 \\ \text{max sum is } 10$$

return max sum
value of
max sum that
will be
returned is
10

TASK #3 → $globalSum = 2$
 $maxSum[1] = 2$

• $i = 2$ →
 $maxSum[2] = -2$

$globalSum = 2$

• $i = 3$ →
 $maxSum[3] = 3$
 $globalSum = 3$

• $i = 4$ →
 $maxSum[4] = 7$
 $globalSum = 7$

• $i = 5$ →
 $maxSum[5] = 4$
 $globalSum = 7$

• $i = 6$ →
 $maxSum[6] = 9$
 $globalSum = 9$

• $i = 7$ →
 $maxSum[7] = 4$
 $globalSum = 9$

• $i = 8$ →
 $maxSum[8] = 10$
 $globalSum = 10$

• $i = 9$ →
 $maxSum[9] = 9$
 $globalSum = 10$

return globalSum

↳ 10 will be returned in globalSum.

Max Sub Array Sum (A, n)

{

global sum = A[1]

Cmaxsum =

$-\infty$

Newmaxsum =

A[1]

for (i = 2 to n)

Newmaxsum = Newmaxsum + A[i]

{

if (Cmaxsum < Newmaxsum)

Cmaxsum = Newmaxsum;

if (Newmaxsum < 0)

Newmaxsum = 0;

return

Cmaxsum;

Max Sub Array Sum (A, n)

{

global sum = A[1]

maxsum = A[1]

for (i = 2 to n)

if (maxsum + A[i] > A[i])

maxsum = maxsum + A[i]

else

maxsum = A[i]

if (globalsum < maxsum)

globalsum = maxsum

return globalsum