Name : Aisha Muhammad Nawaz
Roll Number: 20L-0921
Section : BSCS - 4A
Course : Design and Analysis of Algorithms
Instructor : Dr. Maryam Bashir
Semester : Spring 2022 ##hw
Due: 07-03-22 (Monday)
Date pages = 5

- Maximum Subarray Sum Problem -

TASK # 1

i	1	2	3	4	5	6	7	8	9
A[i]	3	-5	1	5	-4	5	-6	7	-7

for • i = 1
and
j = 1 → subArraySum = 3
Max Sum = 3

MaxSum[1] = 3
• i = 2
and
j = 2 → subArraySum = -5
MaxSum = 3

j = 1 → subArraySum = -2
Max Sum = 3

MaxSum[2] = -2
• i = 3 and j = 3 → subArraySum = 1
Max Sum = 3

j = 2 → subArraySum = -4
Max Sum = 3

j = 1 → subArraySum = -1
Max Sum = 3

MaxSum[3] = 1
• i = 4
and
j = 4 → subArraySum = 5
Max Sum = 5

j = 3 → Sub Array Sum = 6
Max Sum = 6

j = 2 → Sub Array Sum = 1
Max Sum = 6

j = 1 → Sub Array Sum = 4
Max Sum = 6

Max Sum[4] = 6
i = 5

and

j = 5 → Sub Array Sum = -4
Max Sum = 6

j = 4 → Sub Array Sum = 1
Max Sum = 6

j = 3 → Sub Array Sum = 2
Max Sum = 6

j = 2 → Sub Array Sum = -3
Max Sum = 6

j = 1 → Sub Array Sum = 0
Max Sum = 6

Max Sum[5] = 2
i = 6

and

j = 6 → Sub Array Sum = 5
Max Sum = 6

j = 5 → Sub Array Sum = 1
Max Sum = 6

j = 4 → Sub Array Sum = 6
Max Sum = 6

j = 3 → Sub Array Sum = 7
Max Sum = 7

j = 2 → Sub Array Sum = 2
Max Sum = 7

j = 1 → Sub Array Sum = 5
Max Sum = 7

Max Sum[6] = 7

- i = 7
  and
  j = 7 → SubArraySum = −6, MaxSum = 7
  j = 6 → SubArraySum = −1, MaxSum = 7
  j = 5 → SubArraySum = 5, MaxSum = 7
  j = 4 → SubArraySum = 0, MaxSum = 7
  j = 3 → SubArraySum = 1, MaxSum = 7
  j = 2 → SubArraySum = −4, MaxSum = 7
  j = 1 → SubArraySum = −1, MaxSum = 7

maxSum[7] = 1

- i = 8
  and
  j = 8 → SubArraySum = 7, MaxSum = 7
  j = 7 → SubArraySum = 1, MaxSum = 7
  j = 6 → SubArraySum = 6, MaxSum = 7
  j = 5 → SubArraySum = 2, MaxSum = 7
  j = 4 → SubArraySum = 7, MaxSum = 7
  j = 3 → SubArraySum = 8, MaxSum = 8
  j = 2 → SubArraySum = 3, MaxSum = 8
  j = 1 → SubArraySum = 6, MaxSum = 8

- i = 9 maxSum[8] = 8
  and
  j = 9 → SubArraySum = −2, MaxSum = 8
  j = 8 → SubArraySum = 5, MaxSum = 8
  j = 7 → SubArraySum = −1, MaxSum = 8
  j = 6 → SubArraySum = 4, MaxSum = 8
  j = 5 → SubArraySum = 0, MaxSum = 8

continued

j = 4 → SubArraySum = 5, MaxSum = 8

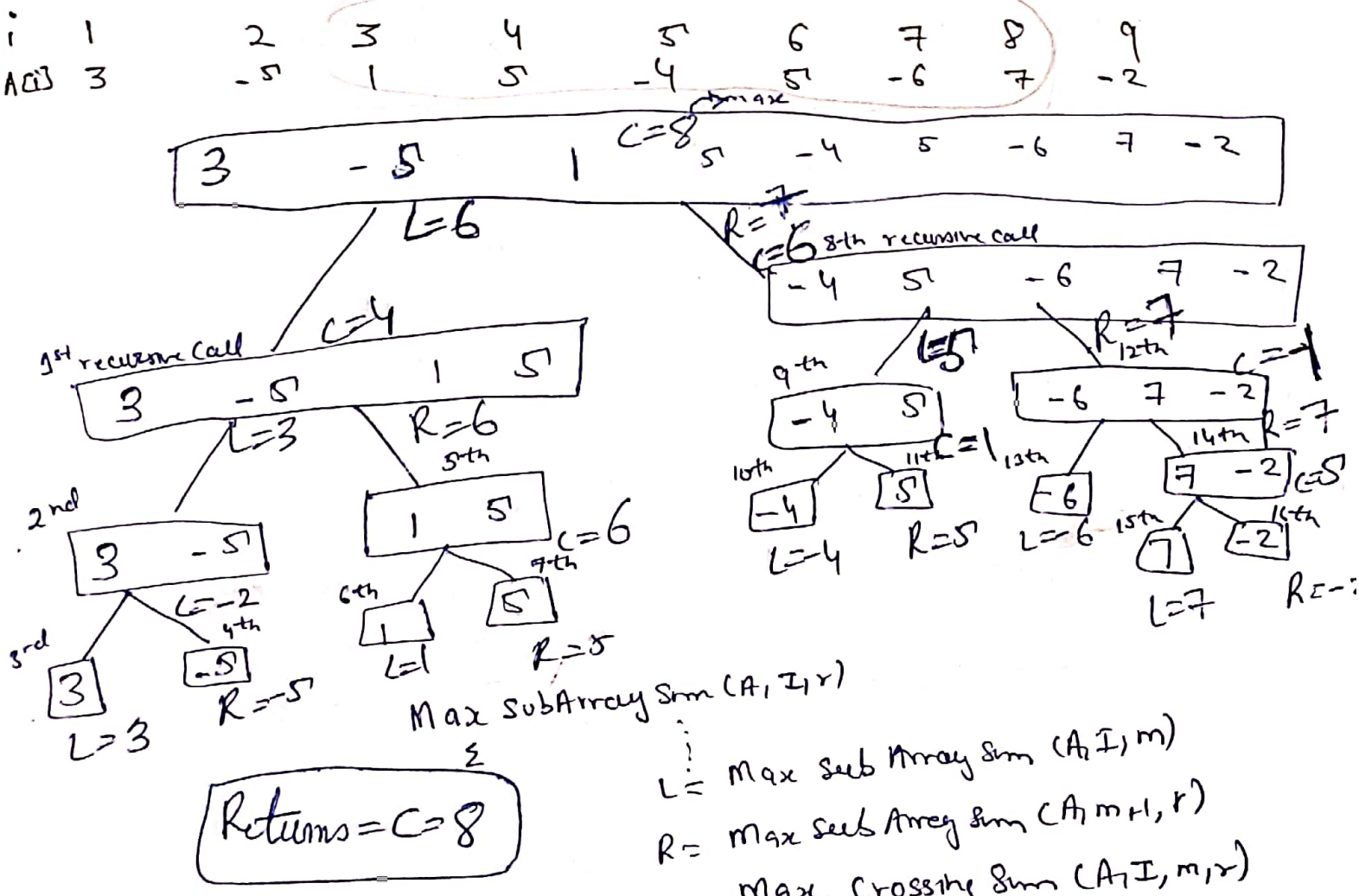
j = 3 → SubArraySum = 6, MaxSum = 8

j = 2 → SubArraySum = 1, MaxSum = 8

j = 1 → SubArraySum = 4, MaxSum = 8 is returned value

MaxSum[9] = 6

TASK # 2



Max SubArray Sum (A, I, r)

Returns = C = 8

L = Max Sub Array Sum (A, I, m)

R = Max Sub Array Sum (A, m+1, r)

C = Max Crossing Sum (A, I, m, r)

return max(L, R, C);

TASK #3

```
Max Sub Array Sum (A, n)
{
    int start = 0, end = 0;
    Max Sum = -infinity

    for (i=1 to n)
    {
        Sub Array Sum = 0
        for (j = i to 1)
        {
            Sub Array Sum += A[j]
            if ( max Sum < Sub Array Sum)
            {
                start = j;
                end = i;
            }
            Max Sum = Max (Max Sum, Sub Array Sum)
        }
    }
    return Max Sum
}
```

TASK #4

```
Max Crossing Sum (A, l, m, r)
{
    Sum = 0
    int start = 0, end = 0;
    for (i = m to 1)
    {
        sum = sum + A[i]
        if (left sum < sum)
            start = i;
        left sum = Max (left sum, sum)
    }
    sum = 0
    for (i = m+1 to r)
    {
        sum = sum + A[i]
        if (right sum < sum)
            end = i;
        right Sum = max (right sum, sum)
    }
    return left Sum + right Sum
}
```