

Question 6

```
// Note that the recursive case is when j < k
middle = (j + k) / 2
sum = 0
left = -infinity
for(i = middle; i >= j; i--)
    sum = sum + A[i]
    if(sum > left)
        left = sum
sum = 0
right = -infinity
for(i = middle + 1; i <= k; i++)
    sum = sum + A[i]
    if(sum > right)
        right = sum
curr_max = left + right
left_max = maxSubArrayRecursive(A, j, middle)
right_max = maxSubArrayRecursive(A, m + 1, k)
return max(curr_max, left_max, right_max)
```