Divide and Conquer Algorithm Analysis

```
WHATDOIDO(integer left, integer right):
   if left==right:
       if A[left]<0 return (0, 0, 0, A[left])
       else return (A[left], A[left], A[left], A[left])
   if left<right:
       m = (left+right)/2 (rounded down)
       (lmaxsum, llmaxsum, lrmaxsum, lsum) = WHATDOIDO(left, m)
       (rmaxsum, rlmaxsum, rrmaxsum, rsum) = WHATDOIDO(m+1, right)
       maxsum = max{lmaxsum, rmaxsum, lrmaxsum+rlmaxsum}
       leftalignedmaxsum = max{llmaxsum, lsum+rlmaxsum}
       rightalignedmaxsum = max{rrmaxsum, lrmaxsum+rsum}
       sum = lsum+rsum
       return (maxsum, leftalignedmaxsum, rightalignedmaxsum, sum)
```

State the recurrence for T(n) that captures the running time of the algorithm as closely as possible.

```
WHATDOIDO(integer left, integer right):
   if left==right:
       if A[left]<0 return (0, 0, 0, A[left]) # O(1)
       else return (A[left], A[left], A[left], A[left]) # O(1)
   if left<right:
       m = (left+right)/2 (rounded down) # O(1)
       (lmaxsum, llmaxsum, lrmaxsum, lsum) = WHATDOIDO(left, m) # T(n/2)
       (rmaxsum, rlmaxsum, rrmaxsum, rsum) = WHATDOIDO(m+1, right) #
T(n/2)
       maxsum = max{lmaxsum, rmaxsum, lrmaxsum+rlmaxsum}
       leftalignedmaxsum = max{llmaxsum, lsum+rlmaxsum}
       rightalignedmaxsum = max{rrmaxsum, lrmaxsum+rsum}
       sum = lsum+rsum
       return (maxsum, leftalignedmaxsum, rightalignedmaxsum, sum)
```

$T(n) = 2T(n/2) \text{ if } n > 1 = O(1) \text{ if } n = 1$

Use the unrolling the recurrence or the mathematical induction to find a tight bound on $T(n)$.

$T(n) = 2T(\frac{n}{2}) = 2(2T(\frac{n}{4})) = 4(2T(\frac{n}{8})) = 2^k T(\frac{n}{2^k})$ so when $k = \log n$ $T(n) = 2^{\log n} \cdot T(1) = nT(1) = n$ So $T(n) = O(n)$

What does the algorithm do?

For an input in A and integers left and right, succinctly describe the meaning of the return variables maxsum, leftalignedmaxsum, rightalignedmaxsum, and sum.

maxsum

The maximum subarray sum anywhere inside A

leftalignedmaxsum

The maximum sum of a subarray that starts at left and ends somewhere between left and right.

rightallignedmaxsum

The maximum sum of a subarray that ends at right and starts somewhere between left and right.

sum

The total sum of all elements in $A[\text{left...right}]$

Recall that we run WHATDOIDO(1,n) and output the rst of the four returned values. Succinctly describe the meaning of this output what does it correspond to in terms of the input?

The first returned value (maxsum) is the maximum subarray sum of the entire array. It correspondes to the largest possible sum of any contiguous subsequence.