1. Write your name below.

2. Trace the execution of the FIND-MAX-CROSSING-SUBARRAY algorithm on the array `A` depicted below, with the arguments `low`, `mid`, and `high` equal to 1, 5, and 10, respectively.

`i`:	1	2	3	4	5	6	7	8	9	10
`A[i]`:	88	19	9	-66	-2	116	-56	-12	87	101

   List the values of *sum* and *left-sum* after each iteration of the first for-loop of the algorithm. Similarly, list the values of *sum* and *right-sum* after each iteration of the second for-loop.

3. Depict the recursion tree that outlines the recursive calls made by the FIND-MAXIMUM-SUBARRAY algorithm when invoked on the array of Question 2 (repeated below), with `low` and `high` equal to 1 and 10, respectively. The nodes of the tree should be labeled with the function invoked (FIND-MAXIMUM-SUBARRAY or FIND-MAX-CROSSING-SUBARRAY and the edges should connect each function's node to the node of its invoker.

i:	1	2	3	4	5	6	7	8	9	10
A[i]:	88	19	9	-66	-2	116	-56	-12	87	101

4. Demonstrate the *recursion tree* method on the recurrence $T(n) = 4T(n/3) + 5n$.

5. Demonstrate the application of the substitution method with guess $T(n) = cn^{\log_3 4}$ to the recurrence of Question 4. Explain where the proof breaks down.

6. Modify the guess of Question 5 to allow the use of the substitution method to prove that $T(n) = O(n^c)$, for a suitable constant c.

7. Prove the result of Question 6 using the *master method*.