Homework 2: Divide-and-Conquer and Heapsort

Note: ^ means “raise to the power”.

1. (2 points) What is the Big-O of 3T(n/3) + 3n^4?
a. O(n^4)
b. O(n^3)
c. O(n!)
d. O(n^⅓)

Answer: a. O(n^4)

2. (3 points) Given an input array A = [4, 1, 3, 2, 16, 9, 10, 14, 17, 6], how is the array transformed by build_min_heap?
a. [4, 2, 3, 1, 14, 9, 10, 17, 6, 16]
b. [1, 2, 3, 6, 14, 9, 10, 17, 4, 16]
c. [1, 2, 3, 14, 4, 9, 10, 17, 16, 6]
d. [1, 4, 3, 2, 14, 9, 10, 17, 6, 16]

Answer: c. [1, 2, 3, 14, 4, 9, 10, 17, 16, 6]

3. (1 point) All recurrences can be handled by the master theorem.
Answer: False

4. (4 points) Trace through the heapsort of [6, 3, 8, 12, 1, 2, 34, 5, 7]. Show the array after buildheap and after each heapify.

Answer: Following all the steps of Build Heap
6, 3, 34, 12, 1, 2, 8, 5, 7
34, 3, 6, 12, 1, 2, 8, 5, 7
34, 3, 8,12, 1, 2, 6, 5, 7
34, 12, 8, 3, 1, 2, 6, 5, 7
34, 12, 8, 7, 1, 2, 6, 5, 3

After This we swap the max element at A[1] position with last element is the heaps that shortens after each iteration. This way elements after A[i] are sorted after each iteration and then run a Max Heapify so that we have maximum element at position 1 in the array
So, the heap after every for loop iteration will be:

12, 7, 8,5, 1, 2, 6, 3, 34
8, 7, 6, 5, 1, 2, 3, 12, 34
7, 5, 6, 3, 1, 2, 8, 12, 34
6, 5, 2, 3, 1, 7, 8, 12, 34

5, 3, 2,1, 6, 7, 8, 12, 34
3, 1, 2, 5, 6, 7, 8,12, 34
2, 1, 3, 5, 6, 7, 8, 12, 34

5. (4 points) Give at least two recurrences for each:
1 - f(n) = Theta(lg n)
2 - f(n) = Theta(n)
3 - f(n) = Theta(n ^2^ lg n)
And for each one, choose a recurrence you wrote and list the work done at the first few levels of the recurrence tree.

Answer:
1.T(n) = T(n/2)+ 1
T(n) = 2T(n/4) + 1
For T(n) = T(n/2) + 1
root : [1]
level 1 : [(1/2) (1/2)]
level 2 : [(1/4) (1/4) (1/4) (1/4)]

2.f(n) = Theta(n)
T(n) = T(n/2) + n
T(n) = T(n/4) + n
Recurrence Tree for T(n) = T(n/2) + n
root : n
level 1 : [(n/2) (n/2)]
level 2 : [(n/4) (n/4) (n/4) (n/4)]

3.f(n) = Theta(n^2 lg n)
1. T(n) = 4T(n/2) + n^2
2. T(n) = 9T(n/3) + n^2
Recursion Tree for T(n) = 4T(n/2) + n^2 :
Root : n^2
level 1 : [(n/2)^2, (n/2)^2, (n/2)^2, (n/2)^2]
level 2 : [(n/4)^2, n/4)^2, n/4)^2, n/4)^2, n/4)^2, n/4)^2, n/4)^2, n/4)^2, n/4)^2, n/4)^2, n/4)^2, n/4)^2, n/4)^2, n/4)^2, n/4)^2, n/4)^2]

6. (2 points) According to the master method, the recurrence T(n) = 4T(n / 2) + n^2 is
a. Big-Theta(n^2)
b. Big-Theta((n^2) / 3)
c. Big-Theta(n^2 lg n)
d. The Master method not applicable

Answer: c. Big-Theta(n^2 lg n)

7. (2 points) According to the master method, the recurrence T(n) = 3T(n / 3) + n lg n is
a. Big-Theta(n)
b. Big-Theta(n / 3)
c. Big-Theta(n lg n)
*d. The Master method not applicable

Answer d: The Master method not applicable

8. (3 points) What is the reason for decrementing loop index i in line 2 of BUILD-MAX-HEAP from A.length / 2 to 1 rather than increasing it from 1 to A.length / 2?

Answer: BUILD-MAX-HEAP calls Max-heapify after every iteration. MAX-HEAPIFY assumes that the heaps below it already satisfy the **max heap property.** So, after satisfying the Max Heap property for the smaller leafs the algorithm percolates upwards.
If we iterate from 1, It will only check for first three elements. Then, it will check for the heaps (sub heaps) below it. If the max heap property is not satisfied, then after swapping the sub - heap becomes a max-heap but then the resulting heap may be a max heap or not as the swapped element may be greater than the parent.
So, the BUILD-MAX-HEAP satisfying the MAX HEAP PROPERTY in the sub leafs and then percolates upwards conquering the rest of the heap.

9. (2 points) What is the time complexity of finding the minimum item in a max heap of size n?
a. O(1)
b. O(lg n)
c. O(n)
d. None of the above

Answer: c. O(n)

10. (1 point) Which of the following algorithms is NOT a divide & conquer algorithm?
a. Binary search
b. Heap Sort
c. Quick Sort
d. Strassen's matrix multiplication
Answer: b. Heap Sort

12. (2 points) The time complexity of heapsort is O(n lg n) because:
a. We do a O(lg n) operation n times.

b. We do a O(n) operation lg n times.
c. The master theorem gives that answer.
d. None of the above.

Answer: a. We do a O(lg n) operation n times.

13. (2 points) Consider a max heap, represented by the array: [40, 30, 20, 10, 15, 16, 17, 8, 4]
The value 37 is inserted into this heap. After insertion and heapification, the new heap is:
a. [40, 30, 20, 10, 15, 16, 17, 8, 4, 37]
b. [40, 30, 20, 10, 37, 16, 17, 8, 4, 15]
c. [40, 37, 20, 10, 30, 16, 17, 8, 4, 15]
d. [40, 37, 20, 10, 15, 16, 17, 8, 4, 30]

Answer: c. [40, 37, 20, 10, 30, 16, 17, 8, 4, 15]

14. (2 points) According to the master method, the recurrence T(n) = .5T(n / 2) + n^2 is
a. Big-Theta(n)
b. Big-Theta(n^2)
c. Big-Theta(n lg n)
d. The Master method not applicable

Answer: d. The Master method not applicable