Started on	Sunday, 26 March 2023, 1:03 PM
State	Finished
Completed on	Sunday, 26 March 2023, 1:20 PM
Time taken	16 mins 32 secs
Marks	2.67/7.00
Grade	**3.81** out of 10.00 (**38.1**%)

Question **1**

Incorrect

Mark 0.00 out of 1.00

Select the asymptotic upper and lower bounds for T(n) in the following recurrence. Assume that T(n) is constant for $n \leq 3$. Make your bounds as tight as possible.

T(n) = T(n-2) + log(n)

Select one:

○ a. T(n) = Θ(log(n))

● b. T(n) = Θ(n) ✗

○ c. T(n) = $\Theta(n^2)$

○ d. T(n) = Θ(nlog(n))

Your answer is incorrect.

The correct answer is:
T(n) = Θ(nlog(n))

Question **2**

Incorrect

Mark 0.00 out of 1.00

Given a set 'S' of n integers and another integer x, an algorithm should determine whether or not there exists two elements in S whose sum is exactly x . A possible algorithm for this task is described below.

1) Sort the elements in S using any efficient sorting algorithm.

2) Remove the last element from S. Let y be the value of the removed element.

3) If S is non-empty, look whether an element z exist in S where z=x-y

4) If S contains such an element z, then stop, since we have found y and z such that x=y+z; otherwise repeat Step 2.

5) If S is empty, then no two elements in S sum to x.

Select the correct statement(s) regarding above approach.

☑ a. Step 1 can be acheived through merge sort with Θ(nlg n) time complexity. ✓

□ b. Best time complexity to do Step 3 is Θ(n) .

☑ c. There are algorithms which can solve this task with better time complexity than above described algorithm ✗

□ d. Time complexity of this algorithm is Θ(nlg n).

The correct answers are: Step 1 can be acheived through merge sort with Θ(nlg n) time complexity., Time complexity of this algorithm is Θ(nlg n).

Question **3**

Correct

Mark 1.00 out of 1.00

Which is not a method for analyzing time complexity of recurrences?

○ a. Master Method

○ b. Substitution Method

○ c. Recurrence Tree Method

● d. Amortized Method ✓

The correct answer is: Amortized Method

Question **4**

Correct

Mark 1.00 out of 1.00

There are n people born between the year 1000 A.D and 2018 A.D, is it possible to sort them by birthdate in O(nlog(n)) time.

Select one:

● True ✓

○ False

Birthdate can be represented in a number format in a database, then we can use **Merge Sort** to sort them by birthdate.

The correct answer is 'True'.

Question 5

Incorrect

Mark 0.00 out of 1.00

For the following recurrence, select the correct expression for runtime T(n) if the recurrence can be solved using Master Theorem, Otherwise, indicate that the Master Theorem does not apply.

T(n) =T(n/2) + 2^n

○ a. T(n) =nlog(n)

○ b. T(n)=Θ(2^n)

● c. Master Theorem does not apply. ✗

○ d. T(n) = Θ(n^2log(n))

The correct answer is: T(n)=Θ(2^n)

Question 6

Incorrect

Mark 0.00 out of 1.00

Find the solution to following recurrence equation:

$f\left(n\right) = \{ \begin{smallmatrix} 1\ if\ n=1 \\ 1+f(\lfloor n/2 \rfloor)\ if\ n\geq 2 \end{smallmatrix}$

● a. $f(n) = log(n)$ ✗

○ b. $f(n) = \lfloor log(n) \rfloor + 1$

○ c. $f(n) = \lfloor log(n+1) \rfloor$

○ d. $f(n) = nlog(n)$

The correct answer is: $f(n) = \lfloor log(n) \rfloor + 1$

Question 7

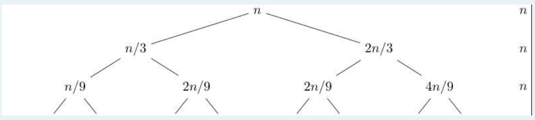
Partially correct

Mark 0.67 out of 1.00

Recursion Tree is one way to analyze recursive functions. Consider a function with following time complexity.

T(n) = T(n/3) + T(2n/3) + n

Following figure shows the first 3 levels of the recursion tree.



What is/are the number(s) which can not be appear in the next (4th) level in this recursion tree?

☐ a. 2n/27

☐ b. 8n/27

☑ c. n/27 ✗

☑ d. 16n/27 ✓

The correct answer is: 16n/27