21BCE1846

VIT
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

## Continuous Assessment Test I - January 2023

| | | | |
|---|---|---|---|
| Programme | B.Tech.(CSE) | Semester | Winter 2022-23 |
| Course | Design and Analysis of Algorithms | Code | BCSE 204L |
| Faculty | Dr.L.Jeganathan, Dr B Jayaram, Dr M Janaki Meena, Dr Muthukumaran, Dr Rajkumar Arul, Dr R Ramesh | Slot/Class No. | B1/CH2022235000277 /CH2022235000279 /CH2022235000280 /CH2022235000283 /CH2022235000284 /CH2022235000285 |
| Time | 90 Minutes | Max. Marks | 50 |

Instructions:

- Answer all the FIVE questions.
- If any assumptions are required, assume the same and mention those assumptions in the answer script.
- Use of intelligence is highly appreciated.
- Your answer for all the questions should have both the 'design' component and the 'analysis component'
- The 'Design' component should consist: understanding of the problem, logic to develop the pseudocode, illustration, pseudocode.
- The 'Analysis' component should consist: Proof-of-correctness, Computation of T(n), Time-complexity.
- For question numbers. 1 & 2, rubric is : Logic (2 marks), Illustration(2 marks), Pseudocode(2 marks), Proof-of-correctness(2 marks), running-time and the time-complexity (2 marks).
- For question no. 3, rubric is : logic of the three pseudocodes (3), illustrations for the three pseudocodes (2), three pseudocodes (2) time-complexities of the three pseudocodes (2) and the conclusion on the efficient pseudocode (1).
- For question No.5, rubric is : Proposal of the problem (1 marks), logic of the pseudocode A(1 mark), Illustration for A(1 mark), pseudocode A(1 mark ), logic for the recursive pseudocode B (2 mark),pseudocode B(1 marks ), Illustration for B(1 marks), Time-complexities of A and B and the conclusion (2 marks),

---

1. The symbols $a, e, i, o, u$ are called as vowels and the remaining symbols of English alphabet are called consonants. A word of length 6 (a word with 6 symbols) $w_1$ is said to be $vowel - greater$ than another word $w_1$ of lenght 6 if the number of distinct vowels is greater than or equal to the number of distinct vowels in $w_2$. If two words $w_1$, $w_2$ have the same number of vowels, the word which comes later in an alphabetical order, is said to be $vowel - greater$ than the other. The number of distinct vowels in *abbaae* is 2. The word *aaeoub* is said to be $vowel - greater$ than the word *abedfg* since the number of distinct vowels in the first word is 4 and the number of distinct vowels in the second word is 2. Similarly, the word *deoudd* is $vowel - greater$ than *cedodu*. Here, both the words have the same number of vowels and the first word *deoudd* comes later in the alphabetical order when compared with the second word *cedodu*. Given the three words *dedodu*, *aeiouu*, *befghs*, they are arranged in an ascending order with respect to the operator $vowel - greater$, as *befghs*, *dedodu*, *aeiouu*. Consider an array of n distinct words of same length (ie., no two words are same), design a pseudocode to sort the array in an ascending order with respect to the relational operator $vowel - greater$. Your design component should contain all the required components. Analyse the pseudocode with all the required steps. [10]

2. Given an n-digit positive integer $m$ (with digits from 0 to 9 ) design a pseudocode with the 'Divide-Conquer-Combine' strategy to generate all n-digit possible integers $m'$ derived from $m$ such that all digits of $m$ are present in $m'$ and the the first digit of $m'$ is same as the last digit of $m'$. The pseudocode

should return all the possible words $m'$ in an ascending order. For example, if m=12323, your pseudocode should return 21332, 23132, 23312, 31223, 32123, 32213. If $m = 1234$, it is clear that no $m'$ is possible since there are no duplicate digits . Your design component should contain all the required steps. Analyse the algorithm with all the required steps. [10]

3. Given a matrix $A$

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ . & . & . & . \\ . & . & . & . \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix}$$

where $a'_{ij}s$ are the integers in the position $(i, j)$. Design three different pseudocodes using the 'Divide-Conquer-Combine' strategy, in three different ways to compute $M$, where

$$M = Max\{a_{ij}, 1 \leq i \leq n, 1 \leq j \leq n\}.$$

Based on the time-complexities of the three pseudocodes, identify the efficient pseudocode. Besides the three different pseudocodes, your answer should contain the logic of the three pseudocodes, illustrations for the three pseudocodes, time-complexities of the three pseudocodes and your conclusion on the efficient pseudocode. [10]

4. Consider the following algorithm.

- Algorithm YYY(n)
- 1. Read n
- 2. Initialise sum to 0
- 3. Initialise digit to 1
- 4. Repeat
- 5. digit= Mod(n,10)
- 6. n=n/10
- 7. sum=sum*10+digit
- 8. Until n==0
- 9. Write sum
- 10. Stop.

Understand the functionality of the above algorithm and answer the following.

(a) Write the output of the algorithm when (i) n = 231, (ii) n= 5 [2]

(b) Describe the functionality of the above algorithm. [3]

(c) Compute the time-complexity of the algorithm. [2]

(d) Modify the above algorithm in such a way that the time-complexity of the modified algorithm remains the same. [3]

5. Propose a problem in detail (of your choice), which is not discussed in your class-room as well as in the lab sessions. Write a brute-force pseudocode $A$ to solve the problem proposed by you. Compute the time-complexity of the pseudocode $A$. Transform your pseudocode $A$ into an equivalent pseudocode $B$ which uses divide-conquer-combine strategy. Compute the time-complexity of both the pseudocodes $A$ and $B$ and identify the efficient one. Note that you are required to propose a problem for which a divide-conquer-combine based pseudocode is possible.
[10]