VIT
Vellore Institute of Technology

Continuous Assessment Test I - May 2023

Programme	B Tech.(CSE)	Semester	Fall Inter 2023-24
Course	Design and Analysis of Algorithms	Code	BCSE 204L
Faculty	Dr M Janaki Meena, Dr. Joe Dhanith P R, Dr. Pandiyaraju V, Dr. Senthil Kumar A M, Dr. Suguna M, Dr. Dominic Savio M	Slot/Class No.	D2 + TD2 CH2022232500782, CH2022232500783, CH2022232500784, CH2022232500785, CH2022232500786, CH2022232500781
Time	90 Minutes	Max. Marks	50

Instructions:

- Answer all the FIVE questions. Use of intelligence is highly appreciated.
- If any assumptions are required, assume the same and mention those assumptions in the answer script.
- 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.

---

1. Consider the logarithmic series

$$\log_e(1+x) = 1 + x + \frac{x^2}{2} + \frac{x^3}{3} + \ldots.$$

Design a pseudocode to compute the $\lfloor \log_e n \rfloor$, using the above series. Here $\lfloor . \rfloor$ is the usual floor operator. For designing the algorithm, you are required to consider ten terms of the above series. [10 Marks]

[**Rubrics**: Logic: 2 marks, Illustration :2 marks, Pseudocode : 4 marks,Time-complexity :2 marks]

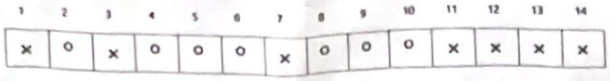
2. Given an array A of 5-digit integers, design two different recursive algorithms, say, algorithm X, algorithm Y, to check if middle digit in all the elements of A are odd. For example, given the array A =[12345, 56178, 23579, 11321], your algorithm should return True. For the array A = [12345, 56278, 23579, 11321], your algorithm should return False as middle digit of the second element in A is even. Please note that both the Algorithms X and Y, should be recursive algorithms but with a different logic. [10 marks]

[**Rubrics**: Logic for pseudocodes X and Y: 2 marks, Illustration for pseudocodes X and Y : 3 marks, Pseudocodes X and Y : 3 marks, Time-complexities of X and Y :2 marks]

3. Given an integer array A of size n and an integer i, nonBreakRightOdd(A, i) returns the number of odd numbers in A that immediately follows the number at index i before an even integer occurs. For example, if A = [4, 3, 4, 5, 1, 7, 8] nonBreakRightOdd(A,1) = 1 since 3 is the only odd number that immediately follows the number 4 at index 1 before an even digit occurs and nonBreakRightOdd(A,3) = 3 since there are three odd numbers 5, 1, 7 that immediately follows the number 4 at position 3 before an even digit occurs. nonBreakRightOdd(A,1)= 0 if length of A is 1. Given an array of integers A, design a 'Divide-Conquer-Combine' algorithm to determine the position of the digit that has got maximum nonBreakRightOdd value. For the array A given above the algorithm should return 3

[**Rubrics**: Logic (2 marks), Illustration(2 marks), Pseudocode(2 marks), Proof-of-correctness(2 marks), running-time and the time-complexity (2 marks)]

In a board game, two type of coins Dought (o) and cross (X) are placed linearly. In a move, a player shall pick any number of coins as per his choice but placed contiguous in the board. Total number of points given to the player for a move is the sum of value of the coins picked by the player in that particular move. Dought is given a value of +1 and cross is given a value of -1. For the board configuration given below, if the player decides to pick coins from index 4 to 7 then he will be given 2 points and if he decides to pick coins from position 1 to 3 then he will be given -1 points. Given a board configuration (letter 'o' will be used for Doughts and letter 'x' will be used for crosses), develop an algorithm to determine the coins to be picked by the player such that he can score maximum points. If there is more than one possibilty to get maximum points then the player would prefer to take the coins placed in rightmost contiguous position



[**Rubrics**: Logic (2 marks), Illustration(2 marks), Pseudocode(2 marks), Proof-of-correctness(2 marks), running-time and the time-complexity (2 marks)]

5. Consider the following algorithm.

Algorithm 1 XXXX(A, w,h)

```
0: Input : (A,w,h)
1: if h-w<2 then
2:    return -1
3: end if
4: m1=⌊(w + h)/3⌋
5: m2 = 2*m1
6: a = YYYY(A, w, m1)
7: b = YYYY(A, m1+1, m2)
8: c = YYYY(A, m2+1, h)
9: return a+b+c
```

Algorithm 2 YYYY(A, w,h)

```
0: Input : (A,w,h)
1: if  w=h then
2:    return A[w]
3: end if
4: m=⌊(w + h)/2⌋
5: n1 = YYYY(A,l,m)
6: n2 = YYYY(A, m-1, h)
7: if n1 > n2 then
8:    return n1
9: end if
10: return n2
```

Understand the above algorithm and answer the following.

(a) Compute the output of the algorithm XXXX if the input array is [7, 2, 9, 8, 1, 3, 5, 7, 2] and when input array is [2, 5] [3 Marks]

(b) Describe the functionality of the Algorithm XXXX [2 Marks]

(c) Compute the time-complexity of the Algorithm XXXX [2 Marks]

(d) Modify the Algorithm XXXX into another Algorithm BB such that the functionality of Algorithms XXXX and BB remains the same but the time-complexity of the Algorithms XXXX and BB are not same. [3 Marks]