VIT

Vellore Institute of Technology

Continuous Assessment Test II - March 2023

Programme	B.Tech (CSE)		
Course	Design and Analysis of Algorithms	Semester	Winter 2022-23
Faculty	Dr L Jeganathan, Dr B Jayaram, Dr M Janaki Meena, Dr Muthukumaran, Dr Rajakumar Arul, Dr R Ramesh	Code	BCSE 204L
		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' illustration, pseudocode.
- The 'Design' component should consist: understanding of the problem, logic to develop the pseudocode,
- The 'Analysis' component should consist: Computation of T(n), Time-complexity.
- For question Nos. 1,2 & 3, rubric is : Logic (2 marks), Illustration(3 marks), Pseudocode(3 marks), running-time and the time-complexity (2 marks)
- 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 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. Given the sequence $X : ABBCDDGFFD$, $Y : ABBCDDFF$, the **Longest Common Pair Subsequences(LCPS)** of X and Y is $BBDDFF$. As an illustration, the characters of $BBDDFF$ occurs in both X and Y in such a way that BB, DD, FF occur in pairs. So in any LCPS of X and Y, the first two characters occur in both X and Y as adjacent to each other, the second pair of characters occur in both X and Y as adjacent to each other and so on. Given two sequences $< X_m, Y_n >$ (X_m is a sequence of m characters, Y_n is a sequence of n characters) m, n are both even integers, design a dynamic programming based pseudocode to compute the Longest Common Pair Subsequence (LCPS) of X_m and Y_n. Your design component should contain all the required steps. Analyse the algorithm with all the required steps.

[10 Marks]

2. Given a square board B of size $n \times n$, design a back-tracking based pseudocode to place all the given n integers: $1, 2, 3, ...n$ in B such that (i) no successive numbers are placed in both the leading diagonals (in leading diagonals: line drawn from the left-most top cell to the right-most bottom cell as well as the line drawn from the left-most bottom cell to the right-most top cell) (ii) non-consecutive numbers can be placed in the same row or column of the board B. Your design component should contain all the required steps. Analyse the algorithm with all the required steps.

[10 Marks]

3. Given a set of courses $S = \{C_1, C_2, ..., C_n\}$, where each C_i is represented by a pair of numbers , $C_i =< start - No., End - No. >$. The courses in S are the courses in which students are interested to take if offered by the school. $C_4 =< 1234, 1267 >$ means that the students with roll numbers 1234, 1235,

1236,...1266, 1267 are interested in taking the course. Assume that there are enough slots to offer all the n courses. Given S and a positive integer k, design a greedy based algorithm to identify the courses that can be offered to the students in such a way that (i) no course can have more than k students. (ii) one student can take only one course (iii) Maximum number of students are offered with the courses. Your design component should contain all the required steps. Analyse the algorithm with all the required steps. [10 Marks]

4. Consider the following algorithm.

Algorithm 1 XXXX

```
0: Input : X, Y
1: n=X.length
2: m= Y.length
3: MCh=" "
4: Mc=0
5: for i= 1 to m do
6:    c=0
7:    for j= 1 to n do
8:       if X[j]==Y[i] then
9:          c++
10:      end if
11:      if c > Mc then
12:         Mc=c
13:         MCh=Y[i]
14:      end if
15:   end for
16: end for
17: Return Mc, MCh
```

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

(a) Write the output of the algorithm when X=123235, Y = 137 [2]

(b) Write the output of the algorithm, when X= ABCDDCEADBDG, Y= ADDG, [2]

(c) Describe the functionality of the above algorithm with the description for each line of the algorithm .

(d) Compute the time-complexity of the algorithm. [4]

5. Propose a problem in detail (of your choice)which is not discussed in the classroom or in the lab sessions. Design a pseudocode A to solve the problem proposed by you that uses a non-recursive brute-force strategy. Note that A should not involve recursive functions. Transform your pseudocode A into an equivalent pseudocode B that uses dynamic programming 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 dynamic programming based pseudocode is possible. [10 Marks]