Roll Number:_____

Thapar Institute of Engineering & Technology Department of Computer Science and Engineering **MID SEMESTER EXAMINATION**	
B. E. **(2nd Yr. COE/CSE, 3rd Yr. EM)** 22nd April, 2023 Saturday, Time– 10:30 To 12:30 PM Time: 2 Hours, Max Marks: 25	Course Code: UCS415 Course Name: Design and Analysis of Algorithms Name of Faculty: Rajesh Mehta, Tarunpreet Bhatia, Randheer Bagi, Anil Singh, Vaibhav Pandey, Manisha Panjeta, Shruti Aggarwal
Note: Attempt all Questions in sequence. Answer all sub-parts of each question at one place. Do mention Page No. of your attempt at front page of your answer sheet. Assume missing data (if any). Pencil-based answers won't be taken into consideration.	

Q.1	Solve the following recurrence relation using recursive tree method. $$T(n)=\begin{cases}1 & \text{if } n=1\\ 3T\left(\left\lfloor\frac{n}{4}\right\rfloor\right)+cn^2 & \text{otherwise}\end{cases}$$ (3)
Q.2	Given a one dimensional array $(n=8), A=\{-2,-3,4,-1,-2,1,5,-3\}$ that contained both positive and negative integers. Objective is to find the sum of contiguous sub-array of numbers which has the largest sum using Brute force and Divide and Conquer approach. Explain both the algorithms and deduce their time complexity with the help of recurrence relation. (5)
Q.3	Compute the time complexity of the following recurrence relation by applying Master Theorem: (i) $T(n)=\sqrt{2}\,T\left(\frac{n}{2}\right)+\log n$ (ii) $T(n)=T\left(\sqrt{n}\right)+\log n$ (iii) $T(n)=2T\left(\frac{n}{2}\right)+\frac{n^2}{\log n}$ (iv) $T(n)=2T\left(\frac{n}{2}\right)+\frac{1}{n}$ (v) $T(n)=T\left(\frac{n}{2}\right)+\sin n$ (5)
Q.5	Consider 4 matrix A, B, C, D with dimension 2×1, 1×3, 3×4, and 4×5 respectively. Apply the efficient algorithmic design strategy to multiply these four matrices with least number of multiplications. (Show all the intermediate steps and recursive formula). Also explain the recursive procedure (algorithm) to print the optimal parenthesization of given four matrices. (6)

Q. Consider C be the set of n characters and each character $c \in C$ is an object with an attribute $c.freq$ giving its frequency. Q be the min-priority queue. Compute the time complexity of each statement from line 1 to line 9 individually and finally explain the overall time complexity of the algorithm given below. Also design the Huffman tree of the following data:

Keys	a	b	c	d	e	F
frequency	5	9	16	12	13	45

```
HUFFMAN(C)
1  n = |C|
2  Q = C
3  for i = 1 to n − 1
4      allocate a new node z
5      z.left = x = EXTRACT-MIN(Q)
6      z.right = y = EXTRACT-MIN(Q)
7      z.freq = x.freq + y.freq
8      INSERT(Q, z)
9  return EXTRACT-MIN(Q)
```

(6)