Title: Design and Analysis of Algorithms (ICT and CS) SEM III core
Course code: IT1216
Structure: 3-0-2-4
Course contents: (Course content will be pruned dynamically in order to adhere to the institute’s guideline of reduced duration for this semester)

1. Introduction
   (a) What is an algorithm?
   (b) Notation for programs
   (c) Proof techniques
   (d) Basics review: Sets - Functions - Limits - Simple series

2. Fundamentals
   (a) Instances and problems - Elementary operations.
   (b) Efficiency
   (c) Average and worst-case analysis
   (d) Examples

3. Asymptotic notation
   (a) Introduction
   (b) A notation for “the order of”
   (c) The omega notation, the ‘oh’ (O) notation, the theta notation

4. Analysis of algorithms
   (a) Solving recurrences
   (b) Data structures
   (c) Arrays, stacks and queues
   (d) Records and pointers
   (e) Lists, graphs, trees and associative tables
   (f) Heaps
   (g) Disjoint set structures

5. Greedy algorithms
   (a) Making change
   (b) General characteristics of Greedy algorithms
   (c) Graphs MST - Kruskal’s and Prims’s algorithms
   (d) Graphs: shortest paths
   (e) Knapsack problem
   (f) Scheduling

6. Divide - and - Conquer
   (a) Multiplying large integers
(b) Binary search
(c) Sorting by: merging and quicksort
(d) Finding the median
(e) Matrix multiplication
(f) Exponentiation

7. Dynamic programming
   (a) Basics of dynamic programming
   (b) Rod cutting problem
   (c) Chained matrix multiplication
   (d) The knapsack problem

8. Introduction to probabilistic algorithms - Parallel algorithms
9. Introduction to computational complexity
10. Approximation Algorithms

Suggested Textbook/references:
1. Introduction to Algorithms - Cormen, Leiserson, Rivest and Stein.
2. Algorithm Design - Kleinberg and Tardos
3. Algorithms - Robert Sedgewick and Kevin Wayne

Evaluation process: Each student has to undertake In-semester and final examinations. At the end of the course he will be given a grade based on his performance in these examinations. Also there are 4-5 Lab Assignments which tests their programming as well as algorithm design skills.

Mechanisms/modalities for online delivery of lecture/lab/tutorial sessions
I shall take the class during regular timetable. Sometimes recorded the videos and uploaded well before the class. During class time, I shall clarify their doubts and solve problems based on uploaded lectures.

Objectives: Through this course students can develop ability to

- design and analysis of the major classes of algorithms.
- to develop their own versions for a given computational task and to compare and contrast their performance.
- understand mathematical structures such as graphs, trees and learn their uses.

<table>
<thead>
<tr>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
<th>P6</th>
<th>P7</th>
<th>P8</th>
<th>P9</th>
<th>P10</th>
<th>P11</th>
<th>P12</th>
</tr>
</thead>
<tbody>
<tr>
<td>❌</td>
<td>❌</td>
<td>❌</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>❌</td>
</tr>
</tbody>
</table>