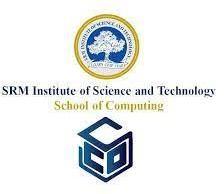
SRM INSTITUTE OF SCIENCE AND TECHNOLOGY FACULTY OF ENGINERING AND TECHNOLOGY SCHOOL OF COMPUTING



COURSE PLAN

21CSC204J DESIGN AND ANALYSIS OF ALGORITHMS

JANUARY – MAY 2024

## Revision History:

| **Date** | **Version** | **Modification**  **done** | **Modified by** | **Reviewed by** | **Signature** |
| --- | --- | --- | --- | --- | --- |
| 24-12-2023 | 1.0 | Initial Release | Dr.K.Alice | Dr.M.Pushpalatha  Dr.G.Niranjana |  |
|  |  |  |  |  |  |

**Table of Contents**

[1.0 General Details 3](#_30j0zll)

[2.0 Reference Books 3](#_3znysh7)

[3.0 Prerequisites 3](#_tyjcwt)

[4.0 Instructional Objectives 3](#_1t3h5sf)

[5.0 Overall Assessment Plan 4](#_2s8eyo1)

[6.0 Tentative Test Schedule 4](#_26in1rg)

[7.0 Detailed Test Plan 5](#_35nkun2)

8.0 Lifelong Learning 7

9.0 Detailed Session Plan ………………………………………………………………………………………………………….…8

10.0 Detailed Execution Plan………………………………………………………………………………………………………….10

# 1.0 General Details

Course Code: 21CSC204J

Course Title: DESIGN AND ANALYSIS OF ALGORITHMS

Semester: IV

Course Time: JAN – MAY 2024

Slot: B

| Day | Batch | | | |
| --- | --- | --- | --- | --- |
| Batch 1 | | Batch 2 | |
| Hour | Timing | Hour | Timing |
| Day order 1 | - | - | - | - |
| Day order 2 | 6,7 | 12.30pm – 2:15pm | 1,2 | 8:00am - 9:40am |
| Day order 3 | 5 | 11:35am - 12:25pm | 10 | 4:00pm - 4:50pm |
| Day order 4 | 8 | 2:20pm - 3:10pm | 3 | 9:45am - 10:35am |
| Day order 5 | - | - | - | - |

**Location**: University Building, Tech Park

# 2.0 Reference Books

1. Thomas H Cormen, Charles E Leiserson, Ronald L Revest, Clifford Stein, Introduction to Algorithms, 3rd ed., The MIT Press Cambridge, 2014
2. Mark Allen Weiss, Data Structures and Algorithm Analysis in C, 2nd ed., Pearson Education, 2006

3. Ellis Horowitz, Sartajsahni, Sanguthevar, Rajesekaran, Fundamentals of Computer Algorithms, Galgotia Publication, 2010

4. S. Sridhar, Design and Analysis of Algorithms, Oxford University Press, 2015

# 3.0 Prerequisites

Nil

# 4.0 Instructional Objectives

1. Develop programs using data types like structures, pointers and arrays supported by C programming language.
2. Analyze the complexity of algorithm and if needed, modify it to improve its efficiency.
3. Identify and use appropriate data structure for devising solution.
4. Describe and use tree structure while developing programs.
5. Implement the Graph structure and use it whenever deemed necessary for providing better solution
6. Decide and use appropriate searching and sorting algorithms while developing solutions for specific problem

# 5.0 Overall Assessment Plan

| **Sl.No.** | **Assessment** | **Component** | **Type** | **Marks** |
| --- | --- | --- | --- | --- |
| 1. | Formative /  CLA-1(T1, T2, T3,T4)  (45%) | CLA 1– T1 (10) | Written Test | **10** |
| CLA 1 – T2 (10) | Lab (Weekly Experiment (7 Marks) +  E-Lab (3 Marks) ) | **10** |
| CLA 1 – T3 (15) | Written Test | **15** |
| CLA1- T4 (10) | Lab (Weekly Experiment (7 Marks) +  E-Lab (3 Marks) ) | **10** |
| 2. | Life Long Learning  CLA-2(T1, T2, T3,T4)  (15%) | Higher Order Thinking Skills ( Co teaching) (CLA-2 T1) | Group Discussion or Assessment | **5** |
| Lab Assessment | Assessment | **3** |
| Analyzing on Real World Applications - VR Demo/ Simulated Demo / PPT  (CLA-2 – T3) | Demo/Presentation | **5** |
| Portfolio Creation  (CLA-2 – T4) | Presentation | **2** |
|  | | **Total Marks** | | **60** |

# 6.0 Tentative Test Schedule

| S.NO. | Tentative date | Test | Marks | Portion | Duration |
| --- | --- | --- | --- | --- | --- |
| 1 | 01-03-2024 | CLA1 T1 | 10 | Unit 1  and 2 | 100 minutes |
| 2 | 24.04.2024 | CLA1 T3 | 15 | Unit 3,4  and 5 | 100 minutes |

# 7.0 Detailed Test Plan

| **Test** | **Tentative**  **Date** | **Type** | **Marks** | **Mode** |
| --- | --- | --- | --- | --- |
|  |  |  | **Total: 50 Marks (Converted to 10)** |  |
| CLA 1 – T1  (UNIT- 1 & UNIT-2) | 01.03.24 | Written Test | Exam Pattern: MCQs –10 (10\*1=10) **(10marks)**  Concept Understanding Questions – 4  (4\*5=20) **(20 marks)**  Scenario based / HOTs Questions – 2  (2\*10=20) **(20 marks)** | Physical Exam |
| CLA 1 – T2 | 01.03.24 | Lab Experiments | **Total: 10 marks**  Elab – **3 Marks**  Weekly lab Experiments –**7 Marks** | Physical mode |
| CLA 1 – T3  (UNIT-3, UNIT-4 & UNIT-5) | 02.05.24 | Written Test | **Total: 50 Marks (Converted to 15)**  Exam Pattern: – MCQs 8 (8\*1=8) **(8 marks)**  Concept Understanding Questions – 3  (3\*5=15) **(15 marks)**  Scenario based / HOTs Questions – 3  (3\*9=27) **(27 marks)** | Physical Exam |
| CLA 1 – T4 | 02.05.24 | Lab Experiments | **Total: 10 marks**  Elab -**3 Marks**  Weekly lab Experiments –**7 Marks** | Physical mode |

# 7.1 Weekly Lab Experiments and E lab Plan

| **TEST** | **LIST OF EXPERIMENTS** | **RUBRICS** |
| --- | --- | --- |
| CLA1-T2 | **Weekly Experiments:**  Lab 1a: Simple Algorithm-Insertion sort  Lab 1b: Bubble Sort  Lab 2: Linear search, Binary search  Lab 3: Merge sort  Lab 4 : Quick sort  Lab 5: Strassen’s Matrix multiplication  Lab 6: Finding Maximum and Minimum in an array, Convex Hull problem  **Elab Sessions:**   1. Searching 2. Sorting 3. Divide & conquer | **Total Marks -10**  **Weekly Experiments:**   * Viva to be conducted for every lab session with respect to the elab practice questions and lab practice questions assigned for each unit. * Observation notebook for lab practice questions should be maintained. * Each experiment in the observation should have the following components: * Aim & Algorithm (1 Marks). * Program implementation (10 Marks) * Basic Solution (2 Marks) * Modularity ( 2.5 Marks) * Readability (2.5 Marks) * Validation (2 Marks) * Scalability (1 Marks) * Time complexity analysis(3 Marks) * Dry run with sample input and output and Result (1 Marks) * Viva –(5 Marks)   Total Marks – 20 **(converted to 1 Marks)**  for each experiment  **Total Marks for Lab Experiments : 7 Marks**  **Elab Sessions:**  3 Programs per session  Total- 9 Programs  **Total Marks for E lab : 3 Marks** |
| CLA1-T4 | **Weekly Experiments:**  Lab 7a: Huffman coding  Lab 7b:Knapsack using greedy  Lab 8: Longest common subsequence  Lab 9: N queen’s problem  Lab 10: Travelling salesman problem  Lab 11: Randomized quick sort  Lab 12: String matching algorithms  **Elab:**   1. Greedy and Dynamic 2. Backtracking and Branch and Bound 3. Randomization |

**8. Lifelong Learning**

**8.1 Presentation on Real World Applications - VR Demo/ Simulated Demo / PPT**

| **Test** | **Tentative Date of final evaluation** | **Artifacts** | **Total Marks** | **Split-up** |
| --- | --- | --- | --- | --- |
| CLA2- T3 | 01.04.24 | PPT/VR Demo/ Simulated Demo | 5 | Final marks to be evaluated based on presentation and demo  Problem selection – 2 Marks  Demo and Presentation – 1 Marks  Subject knowledge – 2 Marks |

**8.2 Higher order thinking Skills through Co teaching**

| **Test** | **Tentative date of**  **Evaluation** | **Total Marks** | **Split-up** |
| --- | --- | --- | --- |
| CLA2- T1 | 24.04.24 | 5 | Medium / hard HOTS questions from Design and Analysis of Algorithm through Co Teaching Method  Tutorial hours to be conducted to evaluate students solving the questions |

# 

**8.3 Lab Assessment**

| **Test** | **Tentative date of**  **Evaluation** | **Total Marks** | **Split-up** |
| --- | --- | --- | --- |
| CLA2- T2 | 02.05.24 | 3 | Simple applications problems can be given based on lab experiments done also complexity Analysis should be tested |

**8.4 Portfolio Creation**

| **Test** | **Tentative Date of final evaluation** | **Artifacts** | **Total Marks** | **Split-up** |
| --- | --- | --- | --- | --- |
| CLA2- T4 | 02.05.24 | Portfolio Creation | 2 | Final marks can be evaluated based on the student’s performance projected on the portfolio web page. |

# 9.0 Detailed Session Plan

| **Sl.No** | **Topics to be covered** | **Hours** | **Teaching Method** | **Testing Method** |
| --- | --- | --- | --- | --- |
| **UNIT-1** | | | | |
| 1 | Introduction-Algorithm Design, Fundamentals of Algorithms | 1 | PPT, BB | Illustration using example |
| 2 | Correctness of algorithm, Time Complexity analysis | 1 | PPT, BB | Illustration using example |
| 3 | Insertion Sort - line count, Operation Count, Algorithm Design Paradigms | 1 | PPT, BB | Illustration using example |
| 4 | Design an Algorithm, And its anlysis - Best. Worst and Average case | 1 | PPT, BB | Illustration using example |
| 5 | Asymptotic Notations Based on Growth Functions., O,Ө, ω, Ω | 1 | PPT, BB | Illustration using example |
| 6 | Mathematical Analysis, Induction, Recurrence Relations | 1 | PPT, BB | Illustration using example, Group Discussion |
| 7 | Solution of Recurrence Relations, Substitution Methods | 1 | PPT, BB | Illustration using example |
| 8 | Solution of Recurrence Relations, Recurrence Tree | 1 | PPT, BB | Illustration using example |
| 9 | Solution of Recurrence Relations, Examples | 1 | PPT, BB | Illustration using example |
| **UNIT-2** | | | | |
| 10 | Introduction-Divide and Conquer, Maximum Subarray Problem | 1 | PPT, BB | Illustration using example |
| 11 | Binary Search, Complexity of binary search | 1 | PPT, BB | Illustration using example |
| 12 | Merge sort, Time complexity analysis | 1 | PPT, BB | Illustration using example |
| 13 | Quick sort and its Time complexity analysis;Best case, Worst case, Average case analysis | 1 | PPT, BB | Illustration using example |
| 14 | Strassen's Matrix multiplication and its recurrence relation;Time complexity analysis | 1 | PPT, BB | Illustration using example |
| 15 | Largest sub-array sum;Time complexity analysis of Largest sub-array sum | 1 | PPT, BB | Illustration using example |
| 16 | Master Theorem Proof; Master theorem examples | 1 | PPT, BB | Illustration using example |
| 17 | Finding Maximum and Minimum in an array; Time complexity analysis-Examples | 1 | PPT, BB | Illustration using example |
| 18 | Algorithm for finding closest pair problem; Convex Hull problem | 1 | PPT, BB | Illustration using example, Activity |
| **UNIT- 3** | | | | |
| 19 | Introduction-Greedy and Dynamic Programming, Examples of problems that can be solved usinggreedy and dynamic approach | 1 | PPT, BB | Illustration using example |
| 20 | Huffman coding using greedy approach,Comparison of brute force and Huffman method | 1 | PPT, BB | Illustration using example |
| 21 | Knapsack problem using greedy approach, Complexity derivation of knapsack using greedy | 1 | PPT, BB | Illustration using example |
| 22 | Tree traversals, Minimum spanning tree - greedy, Kruskal's algorithm - greedy | 1 | PPT, BB | Illustration using example, Activity |
| 23 | Minimum spanning tree - Prims algorithm, Introduction to dynamic programming | 1 | PPT, BB | Illustration using example, Quiz |
| 24 | 0/1 knapsack problem, Complexity calculation of knapsack problem | 1 | PPT, BB | Illustration using example |
| 25 | Matrix chain multiplication using dynamic programming, Complexity of matrix chain | 1 | PPT, BB | Illustration using example |
| 26 | Longest common subsequence using dynamic programming, Explanation of LCS with | 1 | PPT, BB | Illustration using example |
| 27 | Optimal binary search tree (OBST)using dynamic programming, Explanation of OBST | 1 | PPT, BB | Illustration using example |
| **UNIT-4** | | | | |
| 28 | Introduction to Trees, Tree traversals | 1 | PPT, BB | Illustration using example |
| 29 | Complete Binary Tree and its height | 1 | PPT, BB | Illustration using example |
| 30 | Binary Search Trees | 1 | PPT, BB | Illustration using example, Group Discussions |
| 31 | Need for Balance- Rotation | 1 | PPT, BB | Illustration using example |
| 32 | AVL trees | 1 | PPT, BB | Illustration using example, Activity |
| 33 | B Trees | 1 | PPT, BB | Illustration using example |
| 34 | Heaps | 1 | PPT, BB | Illustration using example |
| 35 | Trees and array implementations and applications | 1 | PPT, BB | Illustration using example |
| 36 | Hash functions - Introduction, functions, Collision avoidance, Separate chaining, | 1 | PPT, BB | Illustration using example, Activity |
| **UNIT 5** | | | | |
| 37 | Introduction to randomization and approximation algorithm ; Randomized hiring problem | 1 | PPT, BB | Illustration using example |
| 38 | Randomized quick sort; Complexity analysis | 1 | PPT, BB | Illustration using example |
| 39 | String matching algorithm; Examples | 1 | PPT, BB | Illustration using example |
| 40 | Rabin Karp algorithm for string matching; Example discussion | 1 | PPT, BB | Illustration using example |
| 41 | Approximation algorithm; Vertex covering | 1 | PPT, BB | Illustration using example |
| 42 | Introduction Complexity classes; P type problems | 1 | PPT, BB | Illustration using example |
| 43 | Introduction to NP type problems; Hamiltonian cycle problem | 1 | PPT, BB | Illustration using example |
| 44 | NP complete problem introduction;Satisfiability Problem | 1 | PPT, BB | Illustration using example |
| 45 | NP hard problems;Examples | 1 | PPT, BB | Illustration using example |

# 10. Overall Execution Plan

| # | Activity | Execution |
| --- | --- | --- |
| 1 | Attainment Level - Setting up of target | The target is 2.5 |
| 2 | Lecture handling – Theory | The lecture will be handled through PPT/ Chalk & Board. |
| 3 | Practical Sessions | The practical sessions will be handled using C and C++. |
| 4 | Question Paper Scrutiny | The CT questions are to be taken wrt Blooms Taxonomy based on Course Learning Assessment (CLA), Performance indicators (PIs), COs and Pos. The same will be validated and verified by audit professor. |
| 5 | Conduct of Test | CLA1: CLA1-T1 (10m), CLA1-T2 (10), CLA1-T3 (10m), CLA1-T4 (15m) –Formative Assessment  CLA2- CLA2-T1 (5m), CLA2-T2 (3m), CLA2-T3 (5m), CLA2-T4 (2m) – Lifelong Learning |
| 6 | Mark Entry | The marks will be entered in academia after the CLA components evaluation |
| 7 | Course File Preparation | The course file will be prepared simultaneously from the very first phase of commencement of course based on the SOC & Dept. checklist |
| 8 | COPO mapping | The COPO mapping will be done for every CLA Components and for the end semester result as well. |
| 9 | Co Teaching | Co Teaching has to be organized for students (max.120 per team) and the same will be tested based on Assignments and HOTS which will be considered for CLA2-T1 Component. |
| 10 | Presentation on real world applications | Presentation on Real World Applications - VR Demo/ Simulated Demo / PPT will be conducted for course students to showcase their skills in subject |
| 11 | Conduct of meeting with Course Audit Professor | Before and after the commencement of every CT meeting will be conducted with audit professor to discuss the progress of course |
| 12 | Feedback collection | The feedback will be collected from students (mid sem and end sem) |

Prepared by Verified by

[Dr. K.Alice, AP/ CTECH] [Dr. M.Pushpalatha, Prof.&HOD CTECH]

Date: 09-01-2024 [Dr.G.Niranjana Prof CTECH]