A picture containing candle, halloween, pumpkin

Description automatically generated

**MANIPAL UNIVERSITY JAIPUR**

School of Computer Science Engineering

Department of Computer Science and Engineering

Design & Analysis of Algorithm | AI3130| 1 Credits | 0 0 2 1

**Session: Aug 23 – Dec 23**

Mr. Vivek Sharma | Dr. Vivek Bharadwaj | Dr. Sunil Kumar | Dr. Satya Prakash Maurya | Mr. Sunil Kumar Patel

Class: V AI & ML

**Course Coordinator: Dr. Sunil Kumar**

1. **INTRODUCTION:** This course is offered by the Department of Computer Science and Engineering, targeting students who wish to learn new technologies, ideas, and research in industries or higher studies in the field of Computer Science, IT, and Communication Engineering. This course is designed to develop analytical skills to enable students to design algorithms for various applications and to analyze the algorithms. The mathematical analysis of algorithms is also discussed.
2. **COURSE OUTCOMES:** At the end of the course, students will be able to

## [AI3130.1] Illustrate basic concepts of various algorithm and their complexities.

[AI3130.2] Select and/or apply the appropriate algorithm to solve real-life problems and assess the trade-offs involved in the design choices and calculate the running time complexity.

## [AI3130.3] Demonstrate and analyze various paradigms such as greedy, divide, and conquers approach to enhance their skills, also used in sorting algorithms like merge sort, heap sort, and quick sort, etc., and analyses of different cases.

[AI3130.4] Demonstrating dynamic programming, backtracking, and graph-based techniques to enhance entrepreneurship skills.

[AI3130.5] Developing employability skills to solve various applications based on different designing approaches.

1. **PROGRAM OUTCOMES AND PROGRAM-SPECIFIC OUTCOMES**

## **[PO.1] Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

**[PO.2] Problem Analysis**: Identity, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

**[PO.3] Design/development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

**[PO.4] Conduct investigations of complex problems**: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

**[PO.5] Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

**[PO.6] The engineer and society**: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

**[PO.7] Environment and sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

**[PO.8] Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practices

**[PO.9] Individual and team work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings **[PO.10] Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions **[PO.11] Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one’s own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

**[PO.12] Life-long learning**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

1. **PROGRAM SPECIFIC OUTCOMES (PSOS):** At the end of the B Tech CSE program, the student:

**[PSO.1]** Graduates will be able to examine the applications of Artificial Intelligence and Machine Learning in real-life problems.

**[PSO.2]** Graduates will be able to design and implement intelligent systems for multidisciplinary problems.

1. **ASSESSMENT PLAN:**
2. **SYLLABUS**

Sorting & Searching Algorithm: insertion sort, selection sort, binary search. Basic data structures: stacks and queues, graphs and trees, binary trees. Algorithmic paradigms: Recursion, divide-and-conquer, Merge sort, Quick sort. Greedy: Knapsack, Huffman encoding, dynamic programming, lower bounds and optimal algorithms. Heaps: Heaps, priority queues, min-max heaps, heap sort. Dynamic search structures: Binary search trees, height balancing, B-trees. Algorithms on arrays: Linear-time median finding, sorting in linear time (counting sort, radix sort, bucket sort), String matching (Rabin-Karp and Knuth-Morris-Pratt algorithms). Graph algorithms Traversal: (BFS, DFS, topological sort), Minimum spanning trees (Prim and Kruskal algorithms), shortest paths (Dijkstra’s and Floyd-Warshal algorithms). Mini-Projects & Case Studies.

1. **Text Books:**
2. E. Horowitz, S. Sahni and S. Rajasekaran, *“Computer Algorithms”*, 2nd Edition, University Press, 2008.
3. T. H. Cormen, C. E. Leiserson, R.L.Rivest, and C. Stein, *"Introduction to Algorithms*", 3rd Edition, MIT press, 2010.
4. **Reference Book:**

A. V. Aho, J. E. Hopcroft and J. D. Ullman, "*The Design and Analysis of Computer Algorithms"*, 1st Edition, Pearson Education, 2002.

1. **Web Reference:**

## Share with lab sessions.

1. **LAB PLAN**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Lab No | **Topics** | **Session Outcome** | **Mode of Delivery** | **Corresponding CO** | **Mode of Assessing the Outcome** |
| 1 | **Searching** | Programs based on Iterative Binary Search & Recursive Binary Search. | Lab | AI3130.1 AI3130.3 | Internal Evaluation |
| Home Assignments External Evaluation |
| 2 | **Sorting** | Programs to implement Insertion Sort & Selection Sort. | Lab | AI3130.1 AI3130.3 | Internal Evaluation |
| Home Assignments External Evaluation |
| 3 | **Linear Sorting** | Programs to implement Counting Sort, |  |  | Internal Evaluation and Home Assignments External Evaluation |
| Radix Sort & Bucket Sort. |
| 4 | **DAC Sorting** | Programs to implement Merge Sort & Quick Sort. | Lab | AI3130.1 AI3130.3 | Internal Evaluation |
| Home Assignments External Evaluation |
| 5 | **Heap & Priority** | Programs to implement sorting a given list of elements in ascending order using the following sorting methods. | Lab | AI3130.1 AI3130.3 | Internal Evaluation Home Assignments External Evaluation |
| **Queue** | Heapsort – MAX Heap and MIN Heap & |
|  | programs based on Priority Queue. |
| 6 | **Greedy method** | Programs to implement knapsack problem and Huffman encoding using greedy method. | Lab | AI3130.4 | Internal Evaluation Home Assignments |
| External Evaluation |
| 7 | **Greedy method-** | Programs to implement the shortest path problem using greedy method. (Dijkstra’s and Floyd-Warshal algorithms). | Lab | AI3130.4 | Internal Evaluation Home Assignments |
| **Graph** | External Evaluation |
| 8 | **Greedy method-** | Programs to implement Prim’s & Kruskal’s | Lab | AI3130.4 AI3130.2 | Internal Evaluation Home Assignments |
| **Spanning Trees** | algorithms. | External Evaluation |
| 9 | **Graph Traversal** | Programs to implement Breadth first | Lab | AI3130.4 AI3130.2 | Internal Evaluation Home Assignments External Evaluation |
| search ,Depth first search traversal & Topological sorting. |
| 10 | **Dynamic** | Write a program to implement longest common subsequence. | Lab | AI3130.4 | Internal Evaluation Home Assignments |
| **Programming** | External Evaluation |
| 11 | **Dynamic** | Write a program to implement Matrix chain multiplication and Knapsack using Dynamic Programming. | Lab | AI3130.4 | Internal Evaluation |
| **Programming** | Home Assignments External Evaluation |
| 12 | **String Matching** | Write a program to implement Rabin karp and KMP algorithm. | Lab | AI3130.4 AI3130.5 | Internal Evaluation Home Assignments |
| External Evaluation |

1. Course Evaluation (Tentative): As per DOA guidelines

|  |  |  |  |
| --- | --- | --- | --- |
| **Criteria** | **Description** | **Date** | **Maximum Marks** |
| Internal Assessment (Summative) | Continuous Assessments   1. Internal Viva + Lab Program Execution (30) 2. Lab File (10) 3. Coding Competition (20) | 22, 23, 26 Sept. 2023 (MTE) | 60 |
| End Term Exam  (Summative) | End Term Exam  (Writeup Execution (20), Viva (20)) | 15 Nov. – 21 Nov. 2023 | 40 |
|  | **Total** |  | 100 |
| Attendance (Formative) | A minimum of 75% Attendance is required to be maintained by a student to be qualified for taking up the End Semester examination. The allowance of 25% includes all types of leaves including medical leaves. | | |

1. **Course Articulation Matrix: (Mapping of COs with POs)**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **CO** | **STATEMENT** | CORRELATION WITH PROGRAM OUTCOMES | | | | | | | | | | | | | CORRELATION WITH PROGRAM SPECIFIC  OUTCOMES | | | |
| PO  1 | PO 2 | PO 3 | PO 4 | PO 5 | PO 6 | PO 7 | PO 8 | PO 9 | PO 10 | PO  11 | PO  12 | PSO  1 | | PSO  2 | PSO 3 |
| [AI3130.1] | Illustrate basic concepts of various algorithm and their complexities. | 3 | 1 |  |  |  |  |  |  |  |  |  | 2 | 1 | |  |  |
| [AI3130.2] | Select and/or apply appropriate algorithm to solve real life problems and assess the trade-offs involved in the design choices also calculate the running time complexity. |  | 1 | 2 |  |  |  |  |  |  |  |  | 2 | 3 | |  |  |
| [AI3130.3] | Demonstrate and analyze various paradigms such as greedy, divide and conquers approach to enhance their skills, also used in sorting algorithms like merge sort, heap sort and quick sort etc and analyses of different  cases. |  | 1 | 2 |  |  |  |  |  |  |  |  | 2 | 2 | |  |  |
| [AI3130.4] | Demonstrating dynamic programming, backtracking and graph- based techniques to enhance entrepreneurship  skills. |  | 1 | 2 |  |  |  |  |  |  |  |  | 2 | 3 | |  |  |
| [AI3130.5] | Developing employability skills to solve various application based on different designing approach |  | 1 | 2 |  |  |  |  |  |  |  |  | 2 | 3 | |  |  |