## Lesson Plan for Even semester Govt. Polytechnic, Ambala City

**Faculty:** SANJEEV KUMAR

**Discipline:** Computer Engineering Semester**:** IV **Subject:** DATA STRUCTURES USING ‘C’

**Lesson Plan Duration:** 16 weeks

|  |  |  |
| --- | --- | --- |
| Week | Theory | Practical |
| 1st | L-1 Introduction to d a t a Structure (Linear, Non-Linear, Primitive, Non-Primitive, Contiguous, Non-contiguous data structures) | [P-1] Operations on Arrays (Traversing, insertion, deletion) |
| L-2 Problem solving concept, top down and bottom- up design | [P-17] Operations on Arrays (Searching- Linear Search) |
| L-3 Structured programming concepts |
| 2nd | L-4 Concept of data types, variables, constants. concept of data- information | [P-16] Operations on Arrays (Searching- Binary Search) |
| L-5 Concept of pointer variables and constants. Arrays and pointers, pointers to structures. | [P-2] The addition of two matrices using functions |
| L-6 Concept of Arrays: Single dimensional array Two- dimensional array |
| 3rd | L-7 Representation of Two-dimensional Array (Base Address, LB, UB) | [P-3] The multiplication of two matrices using function |
| L-8 Storage representation of multi-dimensional arrays (Row major, column major order) | [P-\*] Creation of arrays using dynamic memory allocation |
| L-9 Operations on Arrays (Traversing, Insertion, Deletion) |
| 4th | L-10 Operations on Arrays (Searching – Linear Search) | [P-\*] Creation of structures using dynamic memory allocation |
| L-11 Operations on Arrays (Searching – Binary Search) | [P-7] Creation of linked lists using static and dynamic memory allocation |
| L-12 Introduction to linked list. Representation of linked lists in Memory, Comparison between Linked List and  Array |
| 5th | **L-13,14 Ist sessional** | **Ist sessional** |
| L-15 Traversing a linked list Searching an item in a linked list | [P-7] Insertion of elements in linked list at the beginning, at the last and at the desired location |
| 6th | L-16 Insertion and deletion into linked list (At first Node, Specified Position, Last node Application of linked lists | [P-7] Deletion of an item from a linked list |
| L-17 Doubly linked lists Traversing a doubly linked lists Insertion and deletion into doubly linked lists |
| L-18 Applications of linked lists. Stacks, queues | [P-8] Insertion of elements in Doubly linked list at the desired location |
| 7th | L-19 Introduction to stacks. Representation of stacks with array and Linked Lists | [P-8] Deletion of an item from Doubly linked list |
| L-20 Application of stacks-Postfix expression evaluation |
| L-21 Transforming infix expression into postfix expression | [P-4] Push and Pop operations in stacks using linked lists. |
| 8th | L-22 Quick Sort | [P-4] Push and Pop operations in stacks using Arrays |
| L-23 Concept and Comparison between recursion and Iteration factorial of a no with and without recursion | [P-5] Inserting and deleting elements in queue using arrays. |
| L-24 Fobonacii series problem using recursion and without recursion |
| 9th | L-25 Solving Tower of Hanoi problem using recursion and without recursion | [P-5] Inserting and deleting elements in queue using linked lists |
| L-26 Introduction to Queues Implementation of Queues using arrays |
| L-27 Implementation of Queues using linked lists | [P-6] Inserting and deleting elements in circular queue using arrays. |
| 10th | L-28 Circular Queues, De-queues, Application of Queues | [P-6] Inserting and deleting elements in circular queue using linked lists. |
| **L-29,30 IInd sessional** | **IInd sessional** |
| 11th | L-31 Concept of Trees | [P-9] The Factorial of a given number with recursion and without recursion |
| L-32 Representation of Binary tree in memory | [P-10] Fibonacii series with recursion and without recursion |
| L-33 Preorder Traversal (Non-recursive) |

|  |  |  |
| --- | --- | --- |
| Week | Theory | Practical |
| 12th | L-34 In order Traversal (Non-recursive) | [P-11] Program for binary search tree operation- inserting/deleting a node into a binary search tree |
| L-35 Post order Traversal (Non-recursive) | [P-11] Program for binary search tree operation- preorder, in order, post order traversal |
| L-36 Concept of Binary Search Trees (BST) |
| 13th | L-37 Searching and Inserting nodes into BSTs | [P-12] The selection sort technique |
| L-38 Deleting a node from a BST | [P-13] The bubble sort technique |
| L-39 Introduction to Heap |
| 14th | L-40 How to insert Item into a Heap | [P-14] The quick sort technique |
| L-41 How to delete an Item from a Heap & Heapsort | [P-14] The quick sort technique |
| L-42 Selection sort |
| 15th | L-43 Insertion Sort | [P-15] The merge sort technique |
| L-44 Merging | [P-15] The merge sort technique |
| L-45 Merge Sort |
| 16th | L-46 Revision | **IIIrd Sessional** |
| **L-47-48 IIIrd Sessional** |
| Revision |