**As Per the University Syllabus**

**Subject Name (Code) : Data Structure using C ( NCS-301)**

**Assignment 1**

**Q1.** Implement a data structure that supports the following operations insert, findMin, findMax, deleteMin, deleteMax, is Empty, makeEmpty.

You must use the following algorithm:

* Maintain a sorted array.
* Insert new items into the correct position in the array, sliding elements over one position to the right, as needed.
* findMin and findMax, and deleteMax are trivial.
* For deleteMin remove the item in position 0, and slide over all the other items one position to the left.

The program should display the following menu:

Welcome!

Press 1 to insert a new item

Press 2 to find minimum Element

Press 3 to find maximum Element

Press 4 to delete minimum Element

Press 5 to delete maximum Element

Press 6 to determine whether the array is empty or not.

Press 7 to empty the array.

Press 9 to quit

**OBJECTIVE**

* To make students understand representation, implementation and Basic operations that can be performed on Arrays.

**OUTCOME**

* Student would be able to understand and use Arrays.
* Student will get more practice on defining and manipulating arrays.

**Matrix for Mapping between POs and Questions:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| Q1 | 1 | 2 | 3 | 3 | - | - | - | - | - | - | - | 2 |

|  |  |  |
| --- | --- | --- |
| **Qus.** | **CO** | **PSO** |
| Q.1 | 2 | 1, 2, 3 |

**Assignment 2**

Q1.Evaluate the following postfix expression using a stack and show the contents of stack after execution of each operation:

            50, 40, +, 18, 14,-, \*, +

Q2. Write a procedure to implement a queue using two stacks.

Q3. Covert the given expression into prefix using stack.

(a+b)+c ↑ (d+e)+f ↑ (g+h)

**OBJECTIVE**

* To make students understand representation, implementation and Basic operations that can be performed on Stacks.
* To make students understand the differences between the various tree Data Structures.

**OUTCOME**

* Student would be able to understand and use Stacks and Queue.
* Student will get more practice on the operations that can be performed on Stack.

**Matrix for Mapping between POs and Questions:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| Q1 | 1 | 2 | 3 | 3 | - | - | - | - | - | - | - | 2 |
| Q2 | 1 | 2 | 3 | 3 | - | - | - | - | - | - | - | 2 |
| Q3 | 1 | 2 | 3 | 3 | - | - | - | - | - | - | - | 2 |

|  |  |  |
| --- | --- | --- |
| **Qus.** | **CO** | **PSO** |
| Q.1 | 4 | 1, 2, 3 |
| Q.2 | 4 | 1, 2, 3 |
| Q.3 | 4 | 1, 2, 3 |

**Assignment 3**

Q1. Construct an AVL search tree by inserting the following elements in the order of their occurrence. 64, 1, 44 , 26, 13, 110, 98, 85.

Q2. Draw a binary tree T with given the inorder and postorder traversal

Inorder: D K I B A E G H J F C

Post order: K D I E A G B F C J H

Q3. Suppose characters a, b, c, d, e, f have probabilities 0.07, 0.09, 0.12, 0.22, 0.23, 0.27 respectively. Find an optimal Huffman code and draw the Huffman tree

**OBJECTIVE**

* To make students understand the differences between the various tree Data Structures.

**OUTCOME**

* To understand the concept of tree traversing, Height Balance tree and data compression.

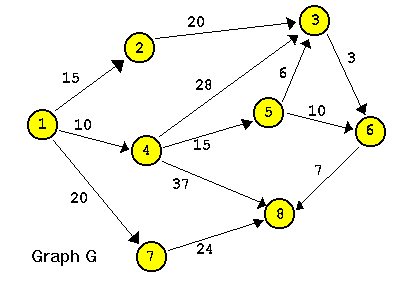
**Matrix for Mapping between POs and Questions:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| Q1 | 1 | 2 | 3 | 3 | - | - | - | - | - | - | - | 2 |
| Q2 | 1 | 2 | 3 | 3 | - | - | - | - | - | - | - | 2 |
| Q3 | 1 | 2 | 3 | 3 | - | - | - | - | - | - | - | 2 |

|  |  |  |
| --- | --- | --- |
| **Qus.** | **CO** | **PSO** |
| Q.1 | 4 | 1, 2, 3 |
| Q.2 | 4 | 1, 2, 3 |
| Q.3 | 4 | 1, 2, 3 |

**Assignment 4**

Q1. Find Minimum spanning tree using Kruskal’s Algorithm of given graph.



|  |  |
| --- | --- |
| Q2. Trace the DFS, BFS on the following graph  problem |  |

**OBJECTIVE**

* To get more practice on Linked Lists and Graphs.

**OUTCOME**

* Students get a hand on Practice on Linked List & Graph, BFS and DFS.

**Matrix for Mapping between POs and Questions:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| Q1 | 1 | 2 | 3 | 3 | - | - | - | - | - | - | - | 2 |
| Q2 | 1 | 2 | 3 | 3 | - | - | - | - | - | - | - | 2 |

|  |  |  |
| --- | --- | --- |
| **Qus.** | **CO** | **PSO** |
| Q.1 | 5 | 1, 2, 3 |
| Q.2 | 5 | 1, 2, 3 |

**Assignment 5**

1. Translate insertion sort into a subprogram INSERTCOUNT (A, N, NUMB) which sorts the array A with N elements and which also counts the number NUMB of comparisons.
2. Suppose an array contains N elements. Given a number X that may occur several times in the array. Find
   1. The number of occurrence of X in the array.
   2. The position of first occurrence of X in the array.

**OBJECTIVE**

* To enable the students to learn the various sorting and searching algorithms and their concepts.

**OUTCOME**

* The students will get a clear idea on some sorting techniques.
* Students will get more practice on the looping concepts.

**Matrix for Mapping between POs and Questions:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| Q1 | 1 | 2 | 3 | 3 | - | - | - | - | - | - | - | 2 |
| Q2 | 1 | 2 | 3 | 3 | - | - | - | - | - | - | - | 2 |

|  |  |  |
| --- | --- | --- |
| **Qus.** | **CO** | **PSO** |
| Q.1 | 6 | 1, 2, 3 |
| Q.2 | 2, 6 | 1, 2, 3 |

**Assignment 6**

Q1. Suppose a Binary Tree T is in memory and an ITEM of information is given.

a. Write a procedure which finds the location LOC of ITEM in T (assuming the elements of T are distinct).

b. Write a procedure which finds the location LOC of the ITEM and the location PAR of the ITEM in T.

c. Write a procedure which finds the number NUM of the ITEM appears in T(assuming the elements of T are not necessarily distinct).

**OBJECTIVE**

* Understand the concept of trees and the operations that can be performed on trees.
* Understand the concept of Graph and BFS.

**OUTCOME**

* Students get a clear concept of the trees and their operations.
* Get a clear idea of the graph and its implementation.
* Students get an idea of Breadth First Search on a directed graph.

**Matrix for Mapping between POs and Questions:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| Q1 | 1 | 2 | 3 | 3 | - | - | - | - | - | - | - | 2 |
| Q2 | 1 | 2 | 3 | 3 | - | - | - | - | - | - | - | 2 |

|  |  |  |
| --- | --- | --- |
| **Qus.** | **CO** | **PSO** |
| Q.1 | 4 | 1, 2, 3 |

**MINI PROJECT**

**Subject Code: NCS 301 Subject Name: Data Structure**

**Problem Statement:** Binary Tree Class (Graphical Implementation).

**Objectives:** This project contains the implementation of all tree related concepts and writes an application that will implement all basic tree related concepts and operations and will show the trees graphically.

**Expected Results & Outcomes:** As outcome of this project we will get graphical representation of various trees. Students may understand the concept and operations of Trees and successfully implement them by using any programming language.

**Concepts learned:** Trees.

**Tools & Techniques are used:** C / C++Programming Language.

**Justify the addressed Pos:** 1, 2, 3, 4, 5, 8, 9, 10, 11, and 12.

1. During implementation, students need to understand the basic computer engineering knowledge. (Mapping with 1).
2. Student analyzes the given problem. (Mapping with 2).
3. Student designs solutions for the given problem. (Mapping with 3).
4. They conduct the investigations and experiments, analyze the outcome of experiments. (Mapping with 4)
5. For implementation, they require advance drawing algorithms in C/C++. (Mapping with 5).
6. While working in a team they come to know about the professional ethics which should followed by team members. (Mapping with 8).
7. They must learn how to work in a team. (Mapping with 9).
8. While discussing with team and project supervisor they will learn how to communicate their views and ideas. (Mapping with 10).
9. This project can act as a prototype to learn about real time project handlings in industries. (Mapping with 11).
10. It is life long learning. (Mapping with 12)

|  |  |
| --- | --- |
| **CO** | **PSO** |
| 4 | 1, 2, 3 |
| 4 | 1, 2, 3 |
| 4 | 1, 2, 3 |

**Content beyond Syllabus**

**Assignment 1**

Q1. Implement a database of student’s information for a department. The program should be able to dynamically allocate or deallocate storage for the student’s records using linked lists.

The database should have the following fields: the first and last names, a course code, and a grade for a student.

The program should display the following menu:

Welcome to the database menu!

Press 1 to insert a new record

Press 2 to delete a record

Press 3 to search the database (by last name)

Press 4 to find the class average for a course

Press 9 to quit

Q2. Suppose A and B are sorted linked lists with distinct elements, both maintained in INFO and LINK. Write a Procedure which combines the lists into a single sorted Linked list.

**OBJECTIVE**

* Understand and use link lists for various operations like insert, delete, append and concatenate etc.
* Understand applications of linked lists in real world applications.

**OUTCOME**

* Students will get a clear concept of use of link lists for various operations like insert, delete, append and concatenate etc.
* Students will get idea of the applications of the linked list in various scenarios.

**Matrix for Mapping between POs and Questions:**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
| Q1 | 1 | 2 | 3 | 3 | - | - | - | - | - | - | - | 2 |
| Q2 | 1 | 2 | 3 | 3 | - | - | - | - | - | - | - | 2 |

|  |  |  |
| --- | --- | --- |
| **Qus.** | **CO** | **PSO** |
| Q.1 | 2, 3 | 1, 2, 3 |

**Text Book &References**

* Aaron M. Tenenbaum,YedidyahLangsam and Moshe J. Augenstein “Data Structures Using C and C+”, PHI Learning Private Limited, Delhi India
* Horowitz and Sahani, “Fundamentals of Data Structures”, Galgotia Publications Pvt Ltd Delhi India.
* A.K. Sharma ,Data Structure Using C, Pearson Education India.
* Rajesh K. Shukla, “Data Structure Using C and C+” Wiley Dreamtech Publication.
* Lipschutz, “Data Structures” Schaum’s Outline Series, Tata Mcgraw-hil Education (India) Pvt. Ltd . 6. Michael T. Godrich, Roberto Tamasia, David M. Mount “Data Structures and Algorithms in C+”, Wiley India.
* P.S. Deshpandey, “C and Datastructure”, Wiley Dreamtech Publication.
* R. Kruse etal, “Data Structures and Program Design in C”, Pearson Education
* Berztis, A.T.: Data structures, Theory and Practice :, Academic Pres.
* Jean Paul Trembley and Paul G. Sorenson, “An Introduction to Data Structures with aplications”, McGraw Hil.

**Suggested Web Links**

* <http://nptel.ac.in/>
* <http://ocw.mit.edu/index.htm/>
* [www.eecs.wsu.edu](http://www.eecs.wsu.edu)
* [www.cse.chalmers.se/edu](http://www.cse.chalmers.se/edu)