

|                         |  |
|-------------------------|--|
| <b>Programme Name/s</b> | <b>: Cloud Computing and Big Data/ Computer Technology/ Computer Engineering/ Computer Science &amp; Engineering/ Computer Hardware &amp; Maintenance/ Information Technology/ Computer Science &amp; Information Technology</b> |
| <b>Programme Code</b>   | <b>: BD/ CM/ CO/ CW/ HA/ IF/ IH</b>  |
| <b>Semester</b>         | <b>: Third</b>   |
| <b>Course Title</b>     | <b>: DATA STRUCTURE USING C</b>  |
| <b>Course Code</b>      | <b>: 313301</b>  |

**I. RATIONALE**

One of the most important courses in information and communication technology is data structures. Data organization or structuring is essential for developing effective algorithms and programs. Students will get the ability to develop logic to solve problem using principles of data structure with the aid of this course.

**II. INDUSTRY / EMPLOYER EXPECTED OUTCOME**

Implement algorithm using relevant Data Structures.

**III. COURSE LEVEL LEARNING OUTCOMES (COS)**

Students will be able to achieve & demonstrate the following COs on completion of course based learning

- CO1 - Perform basic operations on Arrays.
- CO2 - Apply different Searching and Sorting methods.
- CO3 - Implement basic operations on Linked List.
- CO4 - Perform operations on Stack using Array and Linked List Implementations.
- CO5 - Perform operations on Queue using Array and Linked List Implementations.
- CO6 - Create and Traverse Tree to solve problems.

**IV. TEACHING-LEARNING & ASSESSMENT SCHEME**

| Course Code | Course Title           | Abbr | Course Category/s | Learning Scheme          |    |    |     |     |                | Credits | Assessment Scheme |     |       |     |                  |     |       |     |             |     |             |       |
|-------------|------------------------|------|-------------------|--------------------------|----|----|-----|-----|----------------|---------|-------------------|-----|-------|-----|------------------|-----|-------|-----|-------------|-----|-------------|-------|
|             |                        |      |                   | Actual Contact Hrs./Week |    |    | SLH | NLH | Paper Duration |         | Theory            |     |       |     | Based on LL & TL |     |       |     | Based on SL |     | Total Marks |       |
|             |                        |      |                   |                          |    |    |     |     |                |         |                   |     |       |     | Practical        |     |       |     |             |     |             |       |
|             |                        |      |                   | CL                       | TL | LL |     |     |                |         | FA-TH             |     | SA-TH |     | Total            |     | FA-PR |     |             |     |             | SA-PR |
|             |                        |      |                   |                          |    |    | Max | Min |                |         | Max               | Min | Max   | Min | Max              | Min | Max   | Min | Max         | Min |             |       |
| 313301      | DATA STRUCTURE USING C | DSU  | DSC               | 3                        | 1  | 4  | -   | 8   |                | 4       | 3                 | 30  | 70    | 100 | 40               | 50  | 20    | 25# | 10          | -   |             | -     |

**Total IKS Hrs for Sem. : 0 Hrs**

Abbreviations: CL- ClassRoom Learning , TL- Tutorial Learning, LL-Laboratory Learning, SLH-Self Learning Hours, NLH-Notional Learning Hours, FA - Formative Assessment, SA -Summative assessment, IKS - Indian Knowledge System, SLA - Self Learning Assessment

Legends: @ Internal Assessment, # External Assessment, \*# On Line Examination , @\$ Internal Online Examination

Note :

1. FA-TH represents average of two class tests of 30 marks each conducted during the semester.
2. If candidate is not securing minimum passing marks in FA-PR of any course then the candidate shall be declared as "Detained" in that semester.
3. If candidate is not securing minimum passing marks in SLA of any course then the candidate shall be declared as fail and will have to repeat and resubmit SLA work.
4. Notional Learning hours for the semester are (CL+LL+TL+SL)hrs.\* 15 Weeks
5. 1 credit is equivalent to 30 Notional hrs.
6. \* Self learning hours shall not be reflected in the Time Table.
7. \* Self learning includes micro project / assignment / other activities.

## V. THEORY LEARNING OUTCOMES AND ALIGNED COURSE CONTENT

| Sr.No | Theory Learning Outcomes (TLO's) aligned to CO's.  | Learning content mapped with Theory Learning Outcomes (TLO's) and CO's.   | Suggested Learning Pedagogies.                                 |
|-------|--|---|--|
| 1     | TLO 1.1 Classify the given type of Data Structures based on their characteristics and space.<br>TLO 1.2 Perform operations on the given type of Data Structure.  | <b>Unit - I Introduction to Data Structures</b><br>1.1 Introduction: Concept and Need of Data Structure, Definition, Abstract Data Type<br>1.2 Types of Data Structures: (i) Linear Data Structures (ii) Non-Linear Data Structures<br>1.3 Operations on Data Structures: (i) Traversing (ii) Insertion (iii) Deletion  | Lecture Using Chalk-Board Presentations                        |
| 2     | TLO 2.1 Develop algorithm to search the given key using different Searching Techniques.<br>TLO 2.2 Create algorithm to sort data using a given method.   | <b>Unit - II Searching and Sorting</b><br>2.1 Searching: Searching for an item in a data set using the following methods: (i) Linear Search (ii) Binary Search<br>2.2 Sorting: Sorting of data set in an order using the following methods: (i) Bubble Sort (ii) Selection Sort (iii) Insertion Sort (iv) Quick Sort (v) Merge Sort   | Lecture Using Chalk-Board Demonstration Presentations Hands-on |
| 3     | TLO 3.1 Differentiate between Static and Dynamic Memory Allocation.<br>TLO 3.2 Create a suitable structure using a Linked List to represent a Node.<br>TLO 3.3 Create Algorithm to add or remove a specified item from a Linear Linked List. | <b>Unit - III Linked List</b><br>3.1 Difference between Static and Dynamic Memory Allocation.<br>3.2 Introduction to Linked List, Terminologies: Node, Address, Pointer, Information field / Data field, Next pointer, Null Pointer, Empty List.<br>3.3 Type of Lists: Linear List, Circular List, Representation of Doubly Linked List.<br>3.4 Operations on a Singly Linked List: Creating a Linked List, Inserting a new node in a Linked List, Deleting a node from a Linked List, Searching a key in Linked List, Traversing a Singly Linked List.<br>3.5 Applications of Linked List. | Lecture Using Chalk-Board Demonstration Presentations Hands-on |

| Sr.No | Theory Learning Outcomes (TLO's) aligned to CO's.   | Learning content mapped with Theory Learning Outcomes (TLO's) and CO's.   | Suggested Learning Pedagogies.  |
|-------|---|---|---|
| 4     | TLO 4.1 Represent Stack using Array and Linked List.<br>TLO 4.2 Create Algorithm to carry out the PUSH and POP operations in a Stack.<br>TLO 4.3 Use Stack to transform the given expression from Infix to Postfix.<br>TLO 4.4 Evaluate Postfix Expression. | <b>Unit - IV Stack</b><br>4.1 Introduction to Stack: Definition, Stack as an ADT, Operations on Stack-(Push, Pop), Stack Operation Conditions – Stack Full / Stack Overflow, Stack Empty / Stack Underflow.<br>4.2 Stack Implementation using Array and representation using Linked List.<br>4.3 Applications of Stack: Reversing a List, Polish Notations, Conversion of Infix to Postfix Expression, Evaluation of Postfix Expression.<br>4.4 Recursion: Definition and Applications. | Lecture Using Chalk-Board<br>Demonstration<br>Presentations<br>Hands-on |
| 5     | TLO 5.1 Represent Queue using Array and Linked List.<br>TLO 5.2 Explain the characteristics of different types of Queue.<br>TLO 5.3 Create Algorithm to carry out the INSERT and DELETE Operations on a Queue.  | <b>Unit - V Queue</b><br>5.1 Introduction to Queue: Queue as an ADT, Queue representation in memory using Array and representation using a Linked List.<br>5.2 Types of Queues: Linear Queue, Circular Queue, Concept of Priority Queue, Double-Ended Queue.<br>5.3 Queue Operations: INSERT, DELETE, Queue Operation Conditions: Queue Full, Queue Empty.<br>5.4 Applications of Queue.  | Lecture Using Chalk-Board<br>Demonstration<br>Presentations<br>Hands-on |
| 6     | TLO 6.1 Describe the given Tree Terminology.<br>TLO 6.2 Create a Binary Search Tree based on the provided data.<br>TLO 6.3 Create Algorithms to Traverse the Tree using the given method.<br>TLO 6.4 Create an Expression Tree.<br>TLO 6.5 Create Heap.     | <b>Unit - VI Tree</b><br>6.1 Introduction to Trees Terminologies: Tree, Degree of a Node, Degree of a Tree, Level of a node, Leaf Node, Depth / Height of a Tree, In-Degree and Out-Degree, Path, Ancestor and Descendant Nodes.<br>6.2 Tree Types and Traversal methods, Types of Trees: General Tree, Binary Tree, Binary Search Tree (BST).<br>Binary Tree Traversal: In-Order Traversal, Preorder Traversal, Post-Order Traversal.<br>6.3 Expression Tree, Heap                     | Lecture Using Chalk-Board<br>Demonstration<br>Presentations<br>Hands-on |

#### VI. LABORATORY LEARNING OUTCOME AND ALIGNED PRACTICAL / TUTORIAL EXPERIENCES.

| Practical / Tutorial / Laboratory Learning Outcome (LLO) | Sr No | Laboratory Experiment / Practical Titles / Tutorial Titles  | Number of hrs. | Relevant COs |
|--|-------|---|----------------|--------------|
| LLO 1.1 Implement Array Operations.                      | 1     | * Write a 'C' program to perform following Operations on Array: Create, Insert, Delete, Display.              | 4              | CO1          |
| LLO 2.1 Implement Linear Search Method on Numbers.       | 2     | Write a 'C' Program to Search a particular data from the given Array of numbers using: Linear Search Method.  | 2              | CO2          |
| LLO 3.1 Implement Linear Search Method on Strings.       | 3     | * Write a 'C' Program to Search a particular data from the given Array of Strings using Linear Search Method. | 2              | CO2          |
| LLO 4.1 Implement Binary Search Method on Numbers.       | 4     | * Write a 'C' program to Search a particular data from the given Array of numbers using Binary Search Method. | 2              | CO2          |

| Practical / Tutorial / Laboratory Learning Outcome (LLO)              | Sr No | Laboratory Experiment / Practical Titles / Tutorial Titles   | Number of hrs. | Relevant COs |
|---|-------|--|----------------|--------------|
| LLO 5.1 Implement Binary Search Method on Strings.                    | 5     | Write a 'C' Program to Search a particular data from the given Array of Strings using Binary Search Method.                          | 2              | CO2          |
| LLO 6.1 Apply Bubble Sort method for Sorting Numbers.                 | 6     | * Write a 'C' Program to Sort an Array of numbers using Bubble Sort Method.  | 2              | CO2          |
| LLO 7.1 Apply Bubble Sort method for Sorting Strings.                 | 7     | Write a 'C' Program to Sort an Array of Strings using Bubble Sort Method.  | 2              | CO2          |
| LLO 8.1 Apply Selection Sort for Sorting Numbers.                     | 8     | * Write a 'C' Program to Sort an Array of numbers using Selection Sort Method.   | 2              | CO2          |
| LLO 9.1 Apply Selection Sort for Sorting Strings.                     | 9     | Write a 'C' Program to Sort an Array of Strings using Selection Sort Method.   | 2              | CO2          |
| LLO 10.1 Apply Insertion Sort for Sorting Numbers.                    | 10    | * Write a 'C' Program to Sort an Array of numbers using Insertion Sort Method.   | 2              | CO2          |
| LLO 11.1 Apply Insertion Sort for Sorting Strings.                    | 11    | Write a 'C' Program to Sort an Array of Strings using Insertion Sort Method.   | 2              | CO2          |
| LLO 12.1 Create Singly Linked List.                                   | 12    | * Write a 'C' Program to Implement Singly Linked List with Operations: (i) Insert at beginning, (ii) Search, (iii) Display           | 2              | CO3          |
| LLO 13.1 Perform given Operations on Singly Linked List.              | 13    | * Write a C Program to Implement Singly Linked List with Operations: (i) Insert at end, (ii) Insert After, (iii) Delete (iv) Display | 2              | CO3          |
| LLO 14.1 Create Polynomials using Linked List.                        | 14    | Write a C Program to Create Two Polynomials using a Linked List.   | 2              | CO3          |
| LLO 15.1 Perform the Addition of Two Polynomials using a Linked List. | 15    | * Write a 'C' Program to add Two Polynomials using a Linked List.  | 2              | CO3          |
| LLO 16.1 Perform Operations on the Stack using the Array.             | 16    | * Write a 'C' Program to perform PUSH and POP Operations on Stack using an Array.  | 2              | CO4          |
| LLO 17.1 Perform Operations on the Stack using a Linked List.         | 17    | * Write a 'C' Program to perform PUSH and POP Operations on a Stack using a Linked List.   | 2              | CO4          |
| LLO 18.1 Apply recursive procedure to multiply two numbers.           | 18    | * Write a 'C' program to perform multiplication of two numbers using recursion.  | 2              | CO4          |
| LLO 19.1 Apply recursive procedure to reverse the string.             | 19    | Write a 'C' program to print given string in reverse using recursion.  | 2              | CO4          |
| LLO 20.1 Apply recursive procedure to display linked list in reverse. | 20    | Write a 'C' program to create a Singly Linked List and traverse in reverse order using recursion.                                    | 4              | CO3<br>CO4   |
| LLO 21.1 Perform Operations on Linear Queue using Array.              | 21    | * Write a 'C' Program to perform INSERT and DELETE Operations on Linear Queue using an Array.  | 2              | CO5          |
| LLO 22.1 Perform Operations on Linear Queue using Linked List.        | 22    | * Write a 'C' Program to perform INSERT and DELETE operations on Linear Queue using a Linked List.                                   | 2              | CO5          |

| Practical / Tutorial / Laboratory Learning Outcome (LLO)  | Sr No | Laboratory Experiment / Practical Titles / Tutorial Titles   | Number of hrs. | Relevant COs |
|---|-------|--|----------------|--------------|
| LLO 23.1 Perform Operations on Circular Queue using Array.  | 23    | * Write a 'C' Program to perform INSERT and DELETE operations on Circular Queue using an Array.    | 2              | CO5          |
| LLO 24.1 Perform Operations on Circular Queue using a Linked List.  | 24    | Write a 'C' Program to perform INSERT and DELETE operations on Circular Queue using a Linked List. | 2              | CO5          |
| LLO 25.1 Implement Priority Queue using Linked List.  | 25    | Write a 'C' Program to Create a Priority Queue using a Linked List.                                | 4              | CO5          |
| LLO 26.1 Implement Binary Search Tree and perform In-Order Traversal.   | 26    | * Write a 'C' Program to Implement BST (Binary Search Tree) and Traverse in In-Order.              | 2              | CO6          |
| LLO 27.1 Implement Tree Traversal Operations.   | 27    | Write a 'C' Program to Traverse BST in Preorder, and Post-Order.                                   | 2              | CO6          |
| <b>Note : Out of above suggestive LLOs -</b> <ul style="list-style-type: none"> <li>'*' Marked Practicals (LLOs) Are mandatory.</li> <li>Minimum 80% of above list of lab experiment are to be performed.</li> <li>Judicial mix of LLOs are to be performed to achieve desired outcomes.</li> </ul> |       |  |                |              |

#### VII. SUGGESTED MICRO PROJECT / ASSIGNMENT/ ACTIVITIES FOR SPECIFIC LEARNING / SKILLS DEVELOPMENT (SELF LEARNING) : NOT APPLICABLE

#### VIII. LABORATORY EQUIPMENT / INSTRUMENTS / TOOLS / SOFTWARE REQUIRED

| Sr.No | Equipment Name with Broad Specifications  | Relevant LLO Number |
|-------|---|---------------------|
| 1     | Computer System with all necessary Peripherals and Internet Connectivity.<br>'C' Compiler / GCC Compiler/ Online 'C' Compiler | All                 |

#### IX. SUGGESTED WEIGHTAGE TO LEARNING EFFORTS & ASSESSMENT PURPOSE (Specification Table)

| Sr.No              | Unit | Unit Title                      | Aligned COs | Learning Hours | R-Level   | U-Level   | A-Level   | Total Marks |
|--------------------|------|---------------------------------|-------------|----------------|-----------|-----------|-----------|-------------|
| 1                  | I    | Introduction to Data Structures | CO1         | 2              | 2         | 2         | 0         | 4           |
| 2                  | II   | Searching and Sorting           | CO2         | 8              | 2         | 2         | 8         | 12          |
| 3                  | III  | Linked List                     | CO3         | 12             | 2         | 4         | 10        | 16          |
| 4                  | IV   | Stack                           | CO4         | 8              | 2         | 4         | 6         | 12          |
| 5                  | V    | Queue                           | CO5         | 6              | 2         | 2         | 6         | 10          |
| 6                  | VI   | Tree                            | CO6         | 9              | 2         | 4         | 10        | 16          |
| <b>Grand Total</b> |      |                                 |             | <b>45</b>      | <b>12</b> | <b>18</b> | <b>40</b> | <b>70</b>   |

#### X. ASSESSMENT METHODOLOGIES/TOOLS

##### Formative assessment (Assessment for Learning)

- Continuous Assessment based on Process and Product related Performance Indicators. Each practical will be assessed considering 60% weightage to Process and 40% weightage to Product

##### Summative Assessment (Assessment of Learning)



- End semester Examination, Lab performance, Viva-Voce

**XI. SUGGESTED COS - POS MATRIX FORM**

| Course Outcomes (COs)  | Programme Outcomes (POs)                     |                       |                                       |                        |  |                         |                         | Programme Specific Outcomes* (PSOs) |       |       |
|--|--|-----------------------|---------------------------------------|------------------------|--|-------------------------|-------------------------|-------------------------------------|-------|-------|
|  | PO-1 Basic and Discipline Specific Knowledge | PO-2 Problem Analysis | PO-3 Design/ Development of Solutions | PO-4 Engineering Tools | PO-5 Engineering Practices for Society, Sustainability and Environment | PO-6 Project Management | PO-7 Life Long Learning | PSO-1                               | PSO-2 | PSO-3 |
| CO1  | 2  | -                     | -                                     | 1                      | -  | -                       | 1                       |                                     |       |       |
| CO2  | 2  | 2                     | 2                                     | 1                      | -  | -                       | 1                       |                                     |       |       |
| CO3  | 2  | 2                     | 2                                     | 1                      | 1  | 1                       | 1                       |                                     |       |       |
| CO4  | 2  | 2                     | 2                                     | 1                      | -  | 1                       | 1                       |                                     |       |       |
| CO5  | 2  | 2                     | 2                                     | 1                      | -  | 1                       | 1                       |                                     |       |       |
| CO6  | 2  | 2                     | 2                                     | 1                      | -  | 1                       | 1                       |                                     |       |       |
| Legends :- High:03, Medium:02,Low:01, No Mapping: -<br>*PSOs are to be formulated at institute level |  |                       |                                       |                        |  |                         |                         |                                     |       |       |

**XII. SUGGESTED LEARNING MATERIALS / BOOKS**

| Sr.No | Author            | Title  | Publisher with ISBN Number                                   |
|-------|-------------------|--|--|
| 1     | Lipschutz         | Data Structures with 'C' (SIE) (Schaum's Outline Series) | McGraw Hill Education, New Delhi ISBN: 978-0070701984        |
| 2     | Balgurusamy, E.   | Data Structures using 'C'                                | McGraw Hill Education, New Delhi 2013, ISBN: 978-1259029547  |
| 3     | ISRD Group        | Data Structures using 'C'                                | McGraw Hill Education, New Delhi 2013, ISBN: 978-12590006401 |
| 4     | Yashwant Kanetkar | Understanding Pointers in C                              | BPB ISBN 8170298911  |

**XIII . LEARNING WEBSITES & PORTALS**

| Sr.No | Link / Portal   | Description     |
|-------|---|-----------------|
| 1     | <a href="https://www.javatpoint.com/data-structure-introduction">https://www.javatpoint.com/data-structure-introduction</a>   | For All Content |
| 2     | <a href="https://www.geeksforgeeks.org/introduction-to-data-structures/">https://www.geeksforgeeks.org/introduction-to-data-structures/</a>                           | For All Content |
| 3     | <a href="https://studytoday.com/data-structures/">https://studytoday.com/data-structures/</a>   | For All Content |
| 4     | <a href="https://www.tutorialspoint.com/data_structures_algorithms/">https://www.tutorialspoint.com/data_structures_algorithms/</a>                                   | For All Content |
| 5     | <a href="https://www.w3schools.in/data-structures/">https://www.w3schools.in/data-structures/</a>   | For All Content |
| 6     | <a href="https://www.mygreatlearning.com/blog/data-structure-tutorial-for-beginners/">https://www.mygreatlearning.com/blog/data-structure-tutorial-for-beginners/</a> | For All Content |
| 7     | <a href="https://byjus.com/gate/introduction-to-data-structure-notes/">https://byjus.com/gate/introduction-to-data-structure-notes/</a>                               | For All Content |

| Sr.No  | Link / Portal | Description |
|--|---------------|-------------|
| <b>Note :</b> <ul style="list-style-type: none"><li>Teachers are requested to check the creative common license status/financial implications of the suggested online educational resources before use by the students</li></ul> |               |             |