1. **Course Code:** **CSE-2321**
2. **Course Title:** **Data Structures**
3. **Credit Hours:** **3**
4. **Contact Hours:** **3 lecture hours per week**
5. **Type:** **Core, Engineering**
6. **Prerequisite:** **CSE-1121 (Computer Programming I)**
7. **Co-requisite:** **CSE-2322 (Data Structures Lab)**
8. **Instructor's and Class Schedule and Locations**
   Instructor: Zinnia Sultana
   Office Location: Room # 307, Female Academic Building
   Class Hours: Saturday (12.00– 1:00), Sunday (11:00 – 1:00)
   Email: zinniacse@iiuc.ac.bd
9. **Course Rationale / Summary:**
   This course is a continuation to the introduction to computer science and is a study of the different types of data structures, their design, implementation, efficiency and effective use in solving problems. It introduces students to new types of data structures such as arrays, linked list, trees, graphs, heaps, stacks and queues. Students will also learn how to design algorithms for each new data structure studied, create and perform simple operations such as insertion, deletion, merging, sorting, and traversing on data structures. It describes and implements common algorithms for working with advanced data structures and recognizes which data structure is the best to use to solve a particular problem. To take this course, students should be able to program in a standard programming language preferably in C/C++. Some mathematical maturity will also be helpful for the students.

10. **Course Objective:** The objectives of this course are to
    a. Impart a thorough understanding of linear data structures such as linked list, arrays, stacks, queues and their applications.
    b. Learn a thorough understanding of non-linear data structures such as trees, graphs and their applications.
    c. Familiarize with various sorting, searching and hashing techniques and their performance comparison.
    d. design and analyze recursive algorithms in data structures

11. **Course Outcomes (COs):**
    Upon successful completion of this course, students will be able to:

#	CO Description	Weightage (%)
1.	**Write** code in pseudocode for the implementation of various data structures and algorithms.	20%
2.	**Analyze** how the choices of data structure & algorithm methods impact the performance of program.	20%
3.	**Demonstrate** appropriate data structure & algorithmic methods in solving problem.	30%
4.	**Apply** important static and dynamic data structures, and compare and contrast among them.	10%
5.	**Synthesize** the computational efficiency of the principal algorithms for insertion, deletion, sorting, searching, and hashing.	20%

12. Mapping of CO-PO:

#	CO Description	POs	Bloom's Taxonomy Domain/Level	Delivery Methods and activities	Assessment Tools
CO1	**Describe** the basics of various data structures and related algorithms.	PO1	Cognitive/Understand	Lecture, Class discussion, Assignment, Note	Exam, Quiz, Assignment
CO2	**Analyze** performance of various data structure & related algorithms.	PO1 PO2	Cognitive/Analyze	Lecture, Class discussion, Assignment, Note, Competitive Problem Solving	Exam, Quiz, Assignment, Competitive Problem Solving
CO3	**Apply** important static and dynamic data structures.	PO1 PO2	Cognitive/Apply	Lecture, Class discussion, Assignment, Lab work, Note, Competitive Problem Solving	Exam, Quiz, Assignment, Competitive Problem Solving
CO4	**Choose** appropriate data structure and algorithm methods in solving problems and compare among them.	PO2	Cognitive/Evaluate	Lecture, Class discussion, Assignment, Lab work, Note, Competitive Problem Solving	Exam, Quiz, Assignment, Competitive Problem Solving

13. Resources:

Text Books:

#	Name of Authors	Title of Book	Edition	Publisher's Name	Year	ISBN
1.	Seymour Lipschutz	Data Structures	Special Indian Edition	Tata McGraw-Hill	2014	ISBN-13: 978-0-07-060168-0

Reference Books:

#	Name of Authors	Title of Book	Edition	Publisher's Name	Year	ISBN
1.	Y. Langsam, Augenstein, A. M. Tanenbaum	Data Structures Using C and C++	2nd Edition	Prentice Hall India	2014	ISBN:13 978-0387202778
2.	Edward M. Reingold, Wilfred J. Hansen	Data Structures	1st Edition	CBS Publishers and Distributors	1983	ISBN-13: 978-0316739511
3.	Robert Sedgewick	Algorithms in C	3rd	Pearson Education, Inc	2001	ISBN-13: 978-0321573513
4.	D. Samanta	Classic Data Structures	2nd	Prentice Hall of India	2003	ISBN-10: 8120318749

Online Resources:

I) **https://www.topcoder.com/community/competitive-programming/tutorials/**

14. Weightage Distribution among Assessment Tools:

Assessment Tools	Weightage (%)
Class Attendance	10
Class Tests and Assignments	10
Midterm examination	30
Final Examinations	50

15. Grading Policy:

As per IIUC grading policy

16. Course Content

Section-A (Mid-term: 30 Marks)

1. Introduction: Elementary Data organization, Information; Data types; Data Structure, Data Structure operations; Algorithm; Time-Space tradeoff of Algorithms. Mathematical notation & Functions; Algorithmic Notation; Control structures; Sub-algorithms. String; String operations; Pattern matching algorithms

2. Linear Array: Linear Array & its representation in memory; Traversing LA, Insertion & Deletion in LA, Bubble Sort, Linear Search & binary Search. 2D Array & its representation in memory; Matrices; Algebra of matrices; sparse matrices

3. Stack: its representation & applications; PUSH and POP operation on stack. Polish Notation, reverse polish notation; Evaluation of a postfix expression; Transforming infix expression into postfix expression.

Section-B (Final Exam: 50 Marks)

Group-A (20 Marks)

4. **Queue** – its representation; Insertion & deletion in Queue; Deques; Priority Queues. Recursion [Factorial function, Fibonacci sequence, Ackermann function, Towers of Hanoi]

5. **Linked list** - Linked list & its representation in memory; Traversing, Searching, Insertion & Deletion operation on Linked list; Header linked lists; two way lists.

Group-B (30 Marks)

6. Complexity of algorithms, Rate of growth: Big O, Ω and Θ notations; Complexity of Linear Search, Binary search & Bubble sort algorithm. **Sorting** - Insertion sort, selection sort, quick sort, merge sort; **Searching** & data modification; Hashing: Hash function, collision resolution

7. **Tree**- Tree terminology; representation of binary trees in memory; Traversing binary tree; Binary search tree; Insertion & deletion on binary search tree; Heap; Insertion & deletion on heap; Heapsort; B trees; General tree; Balanced binary search tree (AVL tree, red-black tree)

8. **Graph** – graph terminology; representation of graphs – adjacency matrix, path matrix, adjacency list; Traversing a graph – BFS & DFS

17. Weekly Activity Plan

Week	Activities	Introduction
1	Lecture	Chapter-1: Elementary Data organization, Information; Data types; Data Structure, Data Structure operations; Algorithm; Time-Space tradeoff of Algorithms
2	Lecture	Chapter-2 [except 2.5]: Mathematical notation & Functions; Algorithmic Notation; Control structures; Sub-algorithms
3	Lecture	Chapter 3: String; String operations; Pattern matching algorithms
4	Lecture	Chapter-4 [up to 4.8]: Linear Array & its representation in memory; Traversing LA, Insertion & Deletion in LA, Bubble Sort, Linear Search & binary Search
5	**Lecture**	Chapter-4 [4.9 – 4.14]: 2D Array & its representation in memory; Matrices; Algebra of matrices; Sparse matrices
6	Lecture	Chapter-6 [6.1-6.3]: Stack - its representation & applications; PUSH and POP operation on stack Chapter-6 [6.4]: Polish Notation, reverse polish notation; Evaluation of a postfix expression; Transforming infix expression into postfix expression
7	Lecture	Chapter 6 [6.9-6.11]: Queue – its representation; Insertion & deletion in Queue; Deques; Priority Queues.
8	Lecture	Chapter- 6 [6.6]: Recursion [Factorial function, Fibonacci sequence, Ackermann function, Towers of Hanoi]
9	Lecture	Chapter 5: Linked list & its representation in memory; Traversing, Searching, Insertion & Deletion operation on Linked list; Header linked lists; Two way lists
10	**Lecture**	Chapter-2 [only 2.5]: Complexity of algorithms, Rate of growth: Big O notation; Complexity of Linear Search, Binary search & Bubble sort algorithm
11	Lecture	Chapter 9: Sorting- Insertion sort, selection sort, quick sort, merge sort; Searching & data modification; Hashing: Hash function, collision resolution

12	Lecture	Chapter 7: Tree- Tree terminology; representation of binary trees in memory; Traversing binary tree; Binary search tree; Insertion & deletion on binary search tree;
13	Lecture	Heap; Insertion & deletion on heap; Heapsort; B trees; General tree
14	Lecture	Chapter 8: Graph – graph terminology; representation of graphs – adjacency matrix, path matrix, adjacency list; Traversing a graph – BFS & DFS
15	Lecture	Review Class