International Islamic University Chittagong

Department of Computer Science & Engineering

Course No.: CSE –2303	**Course Title: Data Structures**
Credit Hours: 3	**Session: Spring - 2013**

Instructor: Md. Mahiuddin, Assistant Professor, Department of CSE.

Syllabus: Internal data representation; Abstract data types; Elementary data structures: arrays, lists, stacks, queue, trees, graphs; Advanced data structures: heaps, B-trees; Recursion; Sorting; Searching; Hashing; Storage management.

Textbook: I will follow the following book as core textbook. Try to get a personal copy of this book.

1. Seymour Lipschutz: (*Schaum's Outline Series) Theory and Problems of Data Structures*, Adapted by: G A V Pai, Tata McGraw-Hill, Indian adapted edition - 2006.

Reference Books: There are also some other very standard texts on data structure in our library. We will consult these books from time to time. These are:

1. D. Samanta: *Classic Data Structures*, Prentice Hall of India, 2003.
2. Edward M. Reingold, Wilfred J. Hansen: *Data Structures*, CBS Publishers and Distributors.
3. Mark Allen Weiss: *Data Structures and Algorithm Analysis in C*, 2/e, Addison Wesley Longman Inc, 2001.
4. Robert L. Kruse, Bruce P. Leung, Clovis L. Tondo*: Data Structures and Program Design in C*, Prentice-Hall of India.
5. Yedidyah Langsam, Moshe J. Augenstein, Aron M. Tenenbaum: *Data Stuuctures using C and C++*, 2/e, Prentice Hall India.
6. Ellis horowitz, Sartaj Sahni, Dinesh Mehta: *Fundamentals of Data Structures in C++*, Galgotia PublicationsPvt. Ltd., 2000.
7. N. S. Kutti, P. Y. Padhye: Data Structures in C++, Prentice Hall of India, 2001.
8. Nell Dale, Susan C Lilly: *Pascal plus Data Structures, Algorithms and Advanced Programming*, 3/e, Galgotia Publications, India.
9. Robert Sedgewick, *Algorithms in C*, 3/e, Pearson Education, Inc, 2001.

Prerequisites: CSE-1201 & 1202 Structured Programming (& Sessional), which was covered in your second semester.

Class Schedule: We will have three lectures each of one hour duration and one sessional of two hours duration in a week.

Grading Policy: *CSE 2303:* Class Attendance & Performance 10, Class Tests & Assignments 10, Mid Term Examination 30 and Final Examination 50.

CSE 2304: Class Attendance & Performance 10, Lab Assignments 30, ACM Programming 10, Mid Term Examination 20 and Final Examination 30.

Homework & Programming Assignments: You have to follow the deadline to submit your homework and assignments. No late submission will be accepted. Copying will be considered as an offence and will be penalized.

8-Stanza Syllabus

1.	**Introduction:** Elementary Data organization, Information; Data types; Data Structure, Data Structure operations; Algorithm; Time-Space tradeoff of Algorithms
	b) Mathematical notation & Functions; Algorithmic Notation; Control structures; Sub-algorithms
	c) 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
	b) 2D Array & its representation in memory; Matrices; Algebra of matrices; Sparse matrices
3.	**Stack -** its representation & applications; PUSH and POP operation on stack
	b) Polish Notation, reverse polish notation; Evaluation of a postfix expression; Transforming infix expression into postfix expression
4.	**Queue** – its representation; Insertion & deletion in Queue; Deques; Priority Queues.
	b) 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
6.	Complexity of algorithms, Rate of growth: Big O notation; Complexity of Linear Search, Binary search & Bubble sort algorithm
	b) 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
8.	**Graph** – graph terminology; representation of graphs – adjacency matrix, path matrix, adjacency list; Traversing a graph – BFS & DFS

Detailed Lecture Schedule

Lecture 1	Introduction
Lecture 2 to 3	Chapter-1: Elementary Data organization, Information; Data types; Data Structure, Data Structure operations; Algorithm; Time-Space tradeoff of Algorithms
Lecture 4	Chapter-2 [except 2.5]: Mathematical notation & Functions; Algorithmic Notation; Control structures; Sub-algorithms
Lecture 5 to 7	Chapter-4 [up to 4.8]: Linear Array & its representation in memory; Traversing LA, Insertion & Deletion in LA, Bubble Sort, Linear Search & binary Search
Lecture 8 to 9	Chapter 3: String; String operations; Pattern matching algorithms
Lecture 10	**Class test-1**
Lecture-11	Chapter-2 [only 2.5]: Complexity of algorithms, Rate of growth: Big O notation; Complexity of Linear Search, Binary search & Bubble sort algorithm
Lecture 12 to 13	Chapter-4 [4.9 – 4.14]: 2D Array & its representation in memory; Matrices; Algebra of matrices; Sparse matrices
Lecture 14 to 15	Chapter-6 [6.1-6.3]: Stack - its representation & applications; PUSH and POP operation on stack.
Lecture 16 to 17	Chapter-6 [6.4]: Polish Notation, reverse polish notation; Evaluation of a postfix expression; Transforming infix expression into postfix expression
Lecture 18	**Class test-2 / Review Class**
	Mid Term Examination
Lecture 19 to 21	Chapter 6 [6.9-6.11]: Queue – its representation; Insertion & deletion in Queue; Deques; Priority Queues.
Lecture 22 to 23	Chapter- 6 [6.6]: Recursion [Factorial function, Fibonacci sequence, Ackermann function, Towers of Hanoi]
Lecture 24 to 27	Chapter 5: Linked list & its representation in memory; Traversing, Searching, Insertion & Deletion operation on Linked list; Header linked lists; Two way lists
Lecture 28	**Class test-3**
Lecture 29 to 31	Chapter 9: Sorting- Insertion sort, selection sort, quick sort, merge sort; Searching & data modification; Hashing: Hash function, collision resolution
Lecture 32 to 36	Chapter 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
Lecture 37 to 40	Chapter 8: Graph – graph terminology; representation of graphs – adjacency matrix, path matrix, adjacency list; Traversing a graph – BFS & DFS
Lecture 41	**Class Test-4**
Lecture 42	Review Class

International Islamic University Chittagong

Department of Computer Science & Engineering

Course No.: CSE –2304	**Course Title: Data Structures Sessional**
Credit Hours: 1	**Session: Spring - 2013**

Instructor: Md. Mahiuddin, Assistant Professor, Department of CSE.

Detailed Lab Course Outline:

Lab 1	1. Write a program to create an array of n elements and then display all the elements of the list. 2. Write a program to find the largest number from a given list of integers. 3. Write a program to calculate the roots of the quadratic equation $ax^2 + bx + c = 0$ where a, b and c are known.
Lab 2	4. Write a program to create an array of n elements and then separately write the odd and even elements of the list. 5. Write a program to create an array of n elements and then insert an element to the list. 6. Write a program to create an array of n elements and then delete an element from the list. 7. Write a program to sort n numbers using Bubble Sort algorithm.
Lab 3	8. Write a program to search an element from a list of n numbers using Linear Search algorithm. 9. Write a program to search an element from a list of n numbers using Binary Search algorithm. 10. Write a program to determine whether a number n is prime or not where $1< n < 2^{15}$ by using sieve method. 11. Write a program to write 100 randomly generated integer to a file called RAND.DAT. And then read the contents of the file and display them on the screen.
	Project: Write a menu driven program to Process a *sorted* Array. Your program should have the following menu: *Insert, Delete, Search, Display, and Exit*.
Lab 4	12. Write a program to insert a string S into a text T so that S begins in position K of T. 13. A text T and a pattern P are in memory. Write a program to delete first occurrence of P in T. 14. Write a program that will read a string (S) and find the index of the first occurrence of a pattern (P) in the string S. 15. Write a program which calculates the no. of occurrence of each letter of an input text. 16. Write a program to implement the following string operation without using any built in functions related to string. a) Find the length of a string S b) Copy string S2 to S1. c) Concatenate string S2 to S1. d) Compare two strings S1 and S2 e) Reverse a string S. [H]
Lab 5	17. Write a program to interchange the row and column of a matrix. 18. Write a program to add two matrices. 19. Write a program to calculate the multiplication of two matrices. 20. Write a program to calculate the row sum and column sum of a matrix.

	21. Write a program that will read a positive integer in base ***b*** (2<=b<=16) and convert it into base ***d*** (2<=d<=16).
Lab 6	22. Write a program to implement the push and pop operation of a stack 23. Write a program to evaluate a Postfix expression. 24. Write a program to convert an Infix expression into its equivalent Postfix expression. 25. Write a program to implement the *Euclidean Algorithm* for finding the Greatest Common Divisor (GCD) of two given positive integers.
Lab 7	**Mid Term Examination**
Lab 8	26. Write a program to show the insert and delete operations of a circular queue. 27. Write a program to show the insert and delete operations of a priority queue. 28. Write a program to calculate the Factorial of a number using recursive and non-recursive method 29. Write a program to find the nth term F_n of the Fibonacci sequence using recursive and non-recursive method. 30. Write a program to move n disks for Tower of Hanoi problem.
Lab 9	31. Write a program to create a Linked List of n elements and then display the list. 32. Write a program to create a Linked List of n elements and then search an element from the list. 33. Write a program to create a Linked List of n elements and then insert an element to the list. 34. Write a program to create a Linked List of n elements and then delete an element from the list. 35. Write a program to create a Circular Header Linked List of n elements and then display the list. 36. Write a program to create a Two way Linked List of n elements and then display the list.
Lab 10	37. Write a program to sort n numbers using Insertion Sort algorithm. 38. Write a program to sort n numbers using Selection Sort algorithm. 39. Write a program to sort n numbers using Quick Sort algorithm. 40. Write a program to sort n numbers using Merge sort algorithm.
Lab 11	41. Write a program to create a Binary Search Tree of n elements and then display the elements (preorder, inorder and postorder) of the tree. 42. Write a program to create a Binary Search Tree of n elements and then search an element from the tree. 43. Write a program to create a Binary Search Tree of n elements and then delete an element from the tree. 44. Write a program to create a Maxheap of n elements and then display the elements of the heap. 45. Write a program to create a Maxheap of n elements and then delete an element from the heap. 46. Write a program to sort n numbers using Heap sort algorithm.
Lab 12	47. Write a program to display the adjacency matrix of a graph. 48. Write a program to display the adjacency list of a graph. 49. Write a program to traverse a graph using Breadth First Search. 50. Write a program to traverse a graph using Depth First Search.
Lab 13	51. Write a program to find the 100! 52. Write a program to determine the value of the ***n***th Fibonacci number F_n where $Fn = F_{n-1} + F_{n-2}$ and $F_1 = F_2 = 1$ and n<=500.
Lab 14	Review class