Department of Computer Science and Engineering
University of Dhaka

Course Outline

CSE2101 - Data Structures and Algorithms

---

1. General Information

Course Title:	CSE2101 - Data Structures and Algorithms
Credit:	3
Semester:	January, 2023
Google Classroom Code:	mzqm6ng
Instructor:	Md. Tanvir Alam
Contact:	tanvir@cse.du.ac.bd

2. Course Contents

Introduction: Introduction to Data Structures, idea of abstract data type, preliminary idea of algorithm runtime complexity (Big Oh notation), preliminary idea of data structure space complexity. **LinkedList:** Singly/doubly/circular linked lists, basic operations on linked list (insertion, deletion and traverse), dynamic array and its application. **Stack and Queue:** Basic stack operations (push/pop/peek), stack-class implementation using Array and

linked list, in-fix to post-fix expressions conversion and evaluation, balancing parentheses using stack, basic queue operations(enqueue, dequeue), circular queue/ dequeue, queue-class implementation using array and linked list, application- Josephous problem, palindrome checker using stack and queue. **Recursion:** Basic idea of recursion (3 laws-base case, call itself, move towards base case by state change), tracing output of a recursive function, applications- merge sort, permutation, combination. **Sorting:** Insertion sort, selection sort, bubble sort, merge sort, quick sort (randomized quick sort), distribution sort (counting sort, radix sort, bucket sort), lower bounds for sorting, external sort. **Binary Tree:** Binary tree representation using array and pointers, traversal of Binary Tree (in-order, pre-order and post-order). **Binary Search Tree:** BST representation, basic operations on BST (creation, insertion, deletion, querying and traversing), application- searching, sets. **Searching:** Linear search, binary Search, application of Binary Search- finding element in a sorted array, finding n^{th} root of a real number, solving equations. **Heap:** Min-heap, max-heap, Fibonacci-heap, applications-priority queue, heap sort. **General Tree:** Implementation, application of general tree- file system**. Disjoint Set:** Union find, path compression. **Huffman Coding:** Implementation, application- Compression. **Graph:** Graph representation (adjacency matrix/adjacency list), basic operations on graph (node/edge insertion and deletion), traversing a graph: breadth-first search (BFS), depth-first search (DFS), graph-bicoloring. **Self-balancing Binary Search Tree:** AVL tree (rotation, insertion). **Set Operations:** Set representation using bitmask, set/clear bit, querying the status of a bit, toggling bit values, LSB, application of set operations. **String ADT:** The concatenation of two strings, the extraction of substrings, searching a string for a matching substring, parsing.

3. Course Materials

- **Textbook:** Introduction to algorithms by Thomas H. Cormen[et al.].—3rd ed.
- **Powerpoint slides** and other additional materials will be shared on Google Classroom.

4. Lecture Plan

Lecture	Topics
1	**Introduction:** Introduction to Data Structures, idea of abstract data type.
2	**Complexity:** preliminary idea of algorithm runtime complexity (Big O

	notation), preliminary idea of data structure space complexity.
3	**LinkedList:** Singly/doubly/circular linked lists, basic operations on linked list (insertion, deletion and traverse), dynamic array and its application.
4	**Stack:** Basic stack operations (push/pop/peek), stack-class implementation using Array and linked list, in-fix to post-fix expressions conversion and evaluation, balancing parentheses using stack.
5	**Queue:** Basic queue operations(enqueue, dequeue), circular queue/ dequeue, queue-class implementation using array and linked list, application- Josephous problem, palindrome checker using stack and queue.
6	**Sorting I:** Insertion sort, selection sort, bubble sort.
7	**Sorting II:** distribution sort (counting sort, radix sort, bucket sort), lower bounds for sorting, external sort.
8	**Searching:** Linear search, binary Search, application of Binary Search- finding element in a sorted array, finding n^{th} root of a real number, solving equations.
9	**Recursion:** Basic idea of recursion (3 laws-base case, call itself, move towards base case by state change), tracing output of a recursive function,
10	**Recursion II:** Applications- merge sort, quick sort.
11	**Recursion III:** Applications-permutation, combination.
12	**Binary Tree:** Binary tree representation using array and pointers, traversal of Binary Tree (in-order, pre-order and post-order).
13	**Binary Search Tree:** BST representation, basic operations on BST (creation, insertion, deletion, querying and traversing), application- searching, sets.
Incourse	
14	**Heap:** Min-heap, max-heap.
15	**Heap II:** Fibonacci-heap.

16	**Heap III:** applications-priority queue, heap sort.
17	**Self-balancing Binary Search Tree:** AVL tree (rotation, insertion).
18	**Self-balancing Binary Search Tree II:** Red–black tree
19	**Disjoint Set:** Union find, path compression.
20	**Huffman Coding:** Implementation, application- Compression.
21	**Graph I:** Graph representation (adjacency matrix/adjacency list), basic operations on graph (node/edge insertion and deletion).
22	**Graph II:** Traversing a graph: breadth-first search (BFS).
23	**Graph III:** Traversing a graph: depth-first search (DFS).
24	**Graph IV:** Applications of graph - graph bicoloring.
25	**Set Operations:** Set representation using bitmask, set/clear bit, querying the status of a bit, toggling bit values, LSB, application of set operations.
26	**String ADT:** The concatenation of two strings, the extraction of substrings, searching a string for a matching substring, parsing, suffix tree, trie.
27	**Advanced Tree I:** Segment Tree
28	**Advanced Tree II:** Binary Indexed Tree

5. Evaluation and Grading

This course will be evaluated out of 100 marks including continuous assessments and final examinations following the grading policy of the University of Dhaka for regular undergraduate and graduate degree programs. Below are the tentative marks distribution.

Component	**Marks**
Incourse	15
Quiz	10
Assignment	5

Final Exam	70

6. Academic Dishonesty

Any act of academic dishonesty including the adoption of unfair means in the examinations, copying from others, and submission of plagiarized term paper or case presentation or any designated report exercised by a student will result in an 'F' grade in the concerned course subject to the determination of the instructors.