BRAC
UNIVERSITY
Inspiring Excellence

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CSE 221: ALGORITHMS

To understand the meaning of computation and analysis of a problem solving process is the main objective of this course

Fall 2018
Sec 05: Mon, Wed 8:00 AM - 9:20 AM, Room UB30501
Sec 07: Mon, Wed 9:30 AM - 10:50 AM, Room UB30501

Instructors:
Najeefa Nikhat Choudhury, UB - 50410 najeefa.chy@bracu.ac.bd

I. Rationale:

Algorithm is a process or set of rules to be followed in calculations or other problem-solving operations, especially by a computer. The reason why algorithms should be studied is because it does not rely specifically on technology. Algorithms do not require you to know Java or C++ or whatever technology comes out tomorrow. It requires that you have an understanding of the concept of the steps that are involved to solve a problem for a given set of programming languages with common features. The topic of algorithms is important in computer science because it allows for analysis on different ways to compute things and ultimately come up with the best way to solve a particular problem. By best we mean one that consumes the least amount of resources or has the fastest running time.

II. Course Aims and Outcomes:

Aims

From this course, students will learn about some basic types of algorithms used in computer science. They will mainly learn different sorting and search algorithms, graph algorithms, greedy and dynamic algorithms. In addition, they will gather knowledge about analyzing the complexity of any problem. This course will help the students to develop an understanding of how to solve a problem by first mapping it to any known algorithm or problem domain.

Specific Learning Outcomes:

By the end of this course, students will be able to know about:

a) Complexity analysis.
b) Sorting and searching algorithms.
c) Graph algorithms.
d) Greedy and Dynamic algorithms.

e) Geometric Computation based problems.
f) Advanced Data structures.
g) And so on.

III. Format and Procedures:

The course is structured as two lectures per week, each of duration 80 minutes. In addition, there is a weekly 3 hour lab session that is mandatory for students, and contributes towards 25% of their final grade.

Students are expected to be punctual in class, and participate actively through questions and discussion. Bear in mind that participation will be a big portion of your final grade, and simply attending the lectures without any visible engagement will not be of much help. All students are expected to be civil and ensure an environment where everyone feels safe to voice their questions and comments.

IV. My Stance

Given the analytical nature of the course, most of the classes will be modeled as lectures given by the instructor, juxtaposed with questions and clarifications from the students. This course will include a compulsory 3-hour laboratory work each week. Students will be required to do a significant amount of reading. For best results, students should read the textbook prior to coming to class, participate actively in the lecture, and revise the topics once they go back from class. A list of topics to be covered, along with an expected timeline, will be provided in class in order to facilitate this. Details about the textbook will be discussed in class.

V. Course Requirements:

1. Class attendance and participation policy: While attending lectures and being punctual is mandatory, just passively sitting in class will not be conductive to learning. Students are expected to ask questions and are encouraged to have discussions in class about the material being covered. This will be done a lot more productively if students read the textbook prior to coming to class, and also review material already covered in class once they are back home.

2. Course readings:

a) *Reference Books:*

(1) **CLRS -** *Introduction to Algorithms*, By Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein. ISBN: 9780262033848

(2) **HSR -** *Computer Algorithms*, By Ellis Horowitz , Sartaj Sahni, Sanguthevar Rajasekaran. ISBN: 9780929306414

b) *Lecture Slides:* Lecture slides will be periodically posted on TSR.

VI. Grading Procedures:

Theory (75%)					Lab (25%)		Total (100%)
Attendance (Active)	Assignments	Quiz	Mid Term	Final	Assignments	Quiz	
5%	5%	10%	20%	35%	50%	50%	

VII. Academic Integrity

Each student in this course is expected to abide by the BRAC University Code of Academic Integrity. Any work submitted by a student in this course for academic credit will be the student's own work.

You are encouraged to study together and to discuss information and concepts covered in lecture with other students. You can help out or receive help from other students in the form of consultation and guidance. However, this permissible cooperation should never involve one student having possession of a copy of all or part of work done by someone else, in the form of an e-mail, an e-mail attachment file, or any form electronic or hard copy.

If copying occurs, both the student who copied work and the student who gave material to be copied will both automatically receive a zero for the assignment. Penalty for violation of this Code can also be extended to include failure of the course and University disciplinary action.

During examinations, you must do your own work. Talking or discussion is not permitted during the examinations, and you may not compare answers, copy from others, or collaborate in any way. Any collaborative behavior during the examinations will result in failure of the exam, and may lead to failure of the course and University disciplinary action.

VIII. Accommodations for students with disabilities

I am available to discuss appropriate academic accommodations that may be required for student with disabilities. Requests for academic accommodations are to be made during the first three weeks of the semester, except for unusual circumstances, so arrangements can be made.

IX. Tentative Course Schedule

Week No.	Topic of Lab	Topic of Theory Class 1	Topic of Theory Class 2	Assignments For Theory
1	Learning Graph	Introduction & Algorithm Analysis	Time Complexity Analysis	Analysing space complexity of Algorithm
2	Implementing Merge Sort/Quick Sort/HeapSort	Sorting Algorithms	Sorting + Searching Algorithms	
3	Graph + Sorting Mixed Problem	Searching + Sorting Techniques	Quiz 1	
4	Obstacle avoiding BFS	Graph Basics+BFS	DFS	BFS Complexity Analysis
5	Gold mining DFS	Minimum spanning Tree	Topological Sort	
6	MidTerm			
7	Kruskal Implementation	Strongly Connected Components	Quiz 2	
8	Prim Implementation	Huffman Encoding /Decoding	Hashing	Tarjan's Strongly Connected Component Algorithm Analysis
9	Implementing multiple LCS	Dynamic Programming basics	LCS,Coin Change	
10	Implementing Hashing	Job Scheduling	Quiz 3	
11	Implementing Bellman Ford	Shortest Path Dijekstra	BellmanFord	Binary Indexed Tree Complexity
12	Implementing 0/1 Knapsack	Quiz 4	Review 1	
13	Final exam	Review 2	Quiz 5	

GOOD LUCK!!!