UNITED INTERNATIONAL UNIVERSITY

Department of Computer Science and Engineering (CSE)

Course Syllabus

<table>
<tr><td></td><td></td><td></td></tr>
<tr><td>1</td><td><b>Course Title</b></td><td>Data Structure and Algorithms II</td></tr>
<tr><td>2</td><td><b>Course Code</b></td><td>CSE 2217</td></tr>
<tr><td>3</td><td><b>Trimester and Year</b></td><td>Summer 2022</td></tr>
<tr><td>4</td><td><b>Pre-requisites</b></td><td>CSE 2215: Data Structure and Algorithms I, CSE 2213: Discrete Mathematics</td></tr>
<tr><td>5</td><td><b>Credit Hours</b></td><td>3.00</td></tr>
<tr><td>6</td><td><b>Section</b></td><td>A</td></tr>
<tr><td>7</td><td><b>Class Hours</b></td><td>Saturday & Tuesday: 10:05 AM – 11:35 AM</td></tr>
<tr><td>8</td><td><b>Class Room</b></td><td>Room: 0423</td></tr>
<tr><td>9</td><td><b>Instructor's Name</b></td><td>Md. Tarek Hasan (MdTH)</td></tr>
<tr><td>10</td><td><b>Email</b></td><td>tarek@cse.uiu.ac.bd</td></tr>
<tr><td>11</td><td><b>Office</b></td><td>319 (A)</td></tr>
<tr><td>12</td><td><b>Counselling Hours</b></td><td>
<table>
<tr><th>Day</th><th>Time</th></tr>
<tr><td>Saturday</td><td>11:50 AM – 4:15 PM</td></tr>
<tr><td>Sunday</td><td>08:30 AM – 11:00 AM</td></tr>
<tr><td>Tuesday</td><td>11:50 AM – 4:15 PM</td></tr>
<tr><td>Wednesday</td><td>08:30 AM – 11:00 AM</td></tr>
</table>
</td></tr>
<tr><td>13</td><td><b>Text Book</b></td><td>Introduction to Algorithms (3rd edition) by Cormen, Leiserson, Rivest and Stein</td></tr>
<tr><td>14</td><td><b>Reference</b></td><td>https://www.geeksforgeeks.org/</td></tr>
<tr><td>15</td><td><b>Course Contents (approved by UGC)</b></td><td>Techniques for analysis of algorithms, Methods for the design of efficient algorithms: divide and conquer, greedy method, dynamic programming, back tracking, branch and bound, Basic search and traversal techniques, graph algorithms, Algebraic simplification and transformations, lower bound theory, NP-hard and NP-complete problems.</td></tr>
<tr><td>16</td><td><b>Course Outcomes (COs)</b></td><td>
<table>
<tr><th>Cos</th><th>Description</th></tr>
<tr><td>CO1</td><td>Analyze worst-case running times of algorithms using asymptotic analysis. Explain what complexity classes are. Be familiar with the complexity classes and conversion, relation, and reduction between them.</td></tr>
<tr><td>CO2</td><td>Describe different algorithm paradigms and explain when algorithmic design situations call for them. Recite algorithms that employ these paradigms. Synthesize such algorithms.</td></tr>
</table>
</td></tr>
</table>

		CO3	Apply appropriate data structures to design algorithms for solving various problems.

CO	Statement	Bloom's Domain	Program Outcome	Knowledge Profile	Complex Problem	Engineering Activities
CO1	Analyze best-case, average-case and worst-case running times of algorithms using asymptotic analysis. Explain what complexity classes are. Be familiar with the complexity classes and conversion, relation and reduction between them.	C	a Engineering Knowledge	Engineering fundamentals (K3)	Depth of Knowledge (P1)	-
CO2	Describe different algorithm paradigms and explain when algorithmic design situations call for them. Recite algorithms that employ these paradigms. Synthesize such algorithms. Derive and solve problems describing the performance of the algorithms.	C			Depth of Knowledge (P1) Range of conflicting requirements (P2)	
CO3	Compare between different data structures. Pick an appropriate data structure for a design situation.	C				

17	**Teaching Methods**	Lecture, Case Studies.
18	**CO with Assessment Methods**	

CO	Assessment Method	(%)
-	Attendance	5
-	Assignments	5
-	Class Tests	20
CO1, CO2	Midterm exam	30
CO1, CO2, CO3	Final exam	40

19	**Lecture Outline**

Class	Topics/Assignments	COs	Reading Reference	Lecture Outcomes/Activities
1	Analyzing Algorithms: Worst-Case and Best-Case Analysis	1, 2	Lecture	Lecture, Assignment
2	Analyzing Algorithms: Worst-Case and Best-Case Analysis	1, 2	Lecture	Lecture, Assignment
3	Asymptotic Notation	2	3.1	Lecture, Assignment
4	Class Test; The Divide-and-Conquer Approach; Analyzing Divide-and-Conquer Algorithms	2, 3	2.3.1, 2.3.2	Lecture, Test
5	The Maximum-Subarray Problem; The Recursion-Tree Method for Solving Recurrences	2, 3	4.1, 4.4	Lecture, Assignment
6	The Recursion-Tree Method for Solving Recurrences	2	4.1, 4.4	Lecture, Assignment
7	An Activity-Selection Problem; Elements of the Greedy Strategy	2, 3	16.1, 16.2	Lecture, Assignment
8	Class Test; Some Legacy Greedy Problems	2, 3	Lecture	Lecture, Test
9	Fractional Knapsack Problem, Coin Change Problem	2, 3	Lecture	Lecture, Assignment
10	Dynamic Programming Basics, The Rod Cutting Problem	2, 3	15.1	Lecture, Assignment
11	Coin Change Problem, Elements of Dynamic Programming	3	15.3	Lecture, Assignment
12	0/1 Knapsack Problem, Review	3	Lecture	Lecture, Assignment
	MIDTERM EXAM			
13	Applications of BFS, DFS	2, 3	Lecture	Lecture, Assignment
14	Disjoint-Set Operations; Disjoint-Set Forests	2	21.1, 21.3	Lecture, Assignment
15	Growing a Minimum Spanning Tree	3	23.1	Lecture, Assignment
16	Class Test; Kruskal’s Algorithm	2, 3	23.2	Lecture, Test
17	Single-Source Shortest Path Variants, Optimal Substructure of a Shortest Path, Negative-weight Edges, Cycles, Relaxation	3	24	Lecture, Assignment
18	The Bellman-Ford Algorithm	2, 3	24.1	Lecture, Assignment
19	Dijkstra’s Algorithm	2, 3	24.3	Lecture, Assignment

		20	Class Test; Direct-Address Tables, Hash Tables	3	11.1, 11.2	Lecture, Test	
		21	Hash Functions; Open Addressing	3	11.3, 11.4	Lecture, Assignment	
		22	The Naive String-Matching Algorithm; The Rabin-Karp Algorithm	2, 3	32.1, 32.2	Lecture, Assignment	
		23	Class Test; Polynomial Time; Polynomial-Time Verification; NP-Completeness	1	34.1, 34.2, 34.3	Lecture, Test	
		24	NP-Hard, Reducibility, Review	1	34.3	Lecture	

Appendix 1: Assessment Methods

Assessment Method	(%)
Attendance	5
Assignments	5
Class Tests	20
Midterm exam	30
Final exam	40

Appendix 2: Grading Policy

Letter Grade	Marks %	Grade Point	Letter Grade	Marks%	Grade Point
A (Plain)	90-100	4.00	C+ (Plus)	70-73	2.33
A- (Minus)	86-89	3.67	C (Plain)	66-69	2.00
B+ (Plus)	82-85	3.33	C- (Minus)	62-65	1.67
B (Plain)	78-81	3.00	D+ (Plus)	58-61	1.33
B- (Minus)	74-77	2.67	D (Plain)	55-57	1.00
			F (Fail)	<55	0.00

Appendix-3: Program outcomes

POs	Program Outcomes
PO1	An ability to apply knowledge of mathematics, science, and engineering
PO2	An ability to identify, formulate, and solve engineering problems
PO3	An ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability
PO4	An ability to design and conduct experiments, as well as to analyze and interpret data
PO5	An ability to use the techniques, skills, and modern engineering tools necessary for engineering practice
PO6	The broad education necessary to understand the impact of engineering solutions in a global, economic, environmental, and societal context
PO7	A knowledge of contemporary issues
PO8	An understanding of professional and ethical responsibility
PO9	An ability to function on multidisciplinary teams

PO10	An ability to communicate effectively
PO11	Project Management and Finance
PO12	A recognition of the need for, and an ability to engage in life-long learning