Course Specifications

Course Title:	Design and Analysis of Algorithms
Course Code:	CSI 321
Program:	Computer Science and Information Technology
Department:	Department of Computer Science and Information
College:	College of Science
Institution:	Majmaah University

Table of Contents

A. Course Identification

1. Credit hours:	3 **credit hours (2 lecture + 2 Exercise)**
2. Course type **a.** University □ College □ Department ☑ Others □ **b.** Required ☑ Elective □	
3. Level/year at which this course is offered:	**6th level**
4. Pre-requisites for this course (if any)**:**	**Data Structures (CSI 312)**
5. Co-requisites for this course (if any)**:**	**NA**

6. Mode of Instruction (mark all that apply)

No	Mode of Instruction	Contact Hours	Percentage
1	Traditional classroom	48	80 %
2	Blended	6	10%
3	E-learning		0 %
4	Correspondence		0 %
5	Other	6	10%

7. Actual Learning Hours (based on academic semester)

No	Activity	Learning Hours
Contact Hours		
1	**Lecture**	30
2	**Laboratory/Studio**	30
3	**Tutorial**	
4	**Others** (specify)	
	Total	60
Other Learning Hours*		
1	**Study**	45
2	**Assignments**	15
3	**Library**	5
4	**Projects/Research Essays/Theses**	5
5	**Others** (specify)	0
	Total	70

* The length of time that a learner takes to complete learning activities that lead to achievement of course learning outcomes, such as study time, homework assignments, projects, preparing presentations, library times

B. Course Objectives and Learning Outcomes

1. Course Description

Algorithms are fundamental to computer science and software engineering. The real-world performance of any software system depends on two things: (1) the algorithms chosen, and (2) the suitability and efficiency of the various layers of implementation. Good algorithm design is therefore crucial for the performance of all software systems. Moreover, the study of algorithms provides insight into the intrinsic nature of a problem as well as possible solution techniques independent of programming languages, programming paradigms, computer hardware, and other implementation aspects. Course Description: This course will cover the following topics: Basic Definitions, Solving Recursions, O(n^2) Sorting Algorithms, Divide and Conquer Paradigm, Searching Algorithms, Graph Algorithms, Advanced data structures, Dynamic Programming Paradigm, Greedy Algorithms Paradigm.

2. Course Main Objective

1. To provide students with the ability to select algorithms appropriate to a particular purpose and to apply them recognizing the possibility that no suitable algorithm may exist.
2. To acquire students with the range of algorithms that address an important set of well-defined problems, recognizing their strengths and weaknesses, and their suitability in particular contexts.
3. To introduce students to a new range of paradigms and techniques to design algorithms and to solve problems.
4. To enable students to be efficient in their work.

3. Course Learning Outcomes

	CLOs	Aligned PLOs
1	**Knowledge:**	
1.1	Recognize the role of algorithms relative to other technologies used in computer science.	k1
1.2	Name the key algorithmic design paradigms including Brute force, Divide and conquer, Decrease and conquer, Transform and conquer, Greedy Algorithms, Dynamic programming.	k2
1.3	Define the language, notation, and concepts of algorithmic design.	k3
2	**Skills :**	
2.1	Predict the resources that the algorithm requires.	s1
2.2	Develop, analyze and compare existing algorithms for a wide variety of problems including sorting, searching, graphs, and binary search tree.	s2
2.3	Justify and analyze algorithmic tradeoffs: time vs. space, deterministic vs. randomized, and exact vs. approximate.	s3
2.4	Write efficient algorithms of certain selected problems.	s4
3	**Competence:**	
3.1	Work cooperatively in a small group environment.	c1
3.2	Save time and space in each task.	c2
3.3		
3...		

C. Course Content

No	List of Topics	Contact Hours
1	**Basic Definitions:** Definition of an algorithm, Time and space tradeoffs in algorithms, Algorithms strategies, Asymptotic analysis of upper and average complexity bounds, Identifying differences among best, average and worst case behaviors, Big oh, omega, and theta notations.	**8**
2	**Solving Recursions:** Using recurrence relations to analyze recursive algorithms, Substitution method, Recursion-tree method, Master theorem method.	**8**
3	**$O(n^2)$ Sorting Algorithms.** Insertion, Selection, Bubble sort.	**8**
4	**Divide and Conquer Paradigm:** Elements of the divide and conquer technique, Merge sort, and Quick sort.	**8**
5	**Searching Algorithms.** Linear and Binary search.	**4**
6	**Graph Algorithms:** Representation of graphs (adjacency list, adjacency matrix), Depth- and Breadth-first traversals. Minimum spanning tree (Kruskal's and Prim's algorithms). Dijkstra's algorithm.	**12**
7	**Advanced data structures:** Binary search tree.	**4**
8	**Dynamic Programming Paradigm:** Elements of dynamic programming, Matrix chain algorithm.	**4**
9	**Greedy Algorithms Paradigm:** Elements of greedy algorithm, optimal binary search tree.	**4**
Total		

D. Teaching and Assessment

1. Alignment of Course Learning Outcomes with Teaching Strategies and Assessment Methods

Code	Course Learning Outcomes	Teaching Strategies	Assessment Methods
1.0	**Knowledge**		
1.1	Recognize the role of algorithms relative to other technologies used in computer science.	Lectures, Lab demonstrations, Case studies, Individual presentations	Written Exam, Homework assignments, Lab assignments, Class Activities, Quizzes
1.2	Name the key algorithmic design paradigms including Brute force, Divide and conquer, Decrease and conquer, Transform and conquer, Greedy Algorithms, Dynamic programming.		
1.3	Define the language, notation, and concepts of algorithmic design.		
2.0	**Skills**		
2.1	Predict the resources that the algorithm requires.	Lectures, Lab demonstrations, Case studies, Individual	Written Exam, Homework ,assignments, Lab assignments,
2.2	Develop, analyze and compare existing algorithms for a wide variety of		

Code	Course Learning Outcomes	Teaching Strategies	Assessment Methods
	problems including sorting, searching, graphs, and binary search tree.	presentations, Brainstorming	Class Activities, Quizzes, Observations
2.3	Justify and analyze algorithmic tradeoffs: time vs. space, deterministic vs. randomized, and exact vs. approximate.		
2.4	Write efficient algorithms of certain selected problems.		
3.0	**Competence**		
3.1	Work cooperatively in a small group environment.	Small group discussions, Whole group discussions, Brainstorming Presentation	Observations, Homework assignments, Lab assignments. Class Activities
3.2	Save time and space in each task.		

2. Assessment Tasks for Students

#	Assessment task*	Week Due	Percentage of Total Assessment Score
1	First written mid-term exam	6	10%
2	Second written mid-term exam	12	10%
3	Presentation, class activities, and group discussion	Every week	10%
4	Homework assignments	After each chapter	10%
5	Implementation of presented algorithms	Every two weeks	10%
6	Electronic Quizzes	Every chapter	10%
7	Final written exam	16	40%
8	Total		100%

***Assessment task** (i.e., written test, oral test, oral presentation, group project, essay, etc.)

E. Student Academic Counseling and Support

Arrangements for availability of faculty and teaching staff for individual student consultations and academic advice :

Office hours: Sun: 10-12, Mon. 10-12, Thru. 8-10
Office call: Sun. 12-1 and Wed 12-1

Email: h.haly@mu.edu.sa
Mobile: 0538231332

F. Learning Resources and Facilities

1.Learning Resources

Required Textbooks	**Thomas Cormen, Charles Leiserson, Ronald Rivest, and Clifford Stein, Introduction to Algorithms, Third Edition. MIT Press, 2009**
Essential References Materials	**Michael T. Goodrich, Roberto Tamassia, and Divid Mount, Data Structures and Algorithms in C++, John Wiley & Sons Inc, 2011.**
Electronic Materials	MIT courseware, videos of the algorithms course - **http://mitpress.mit.edu/catalog/item/default.asp?ttype=2&tid=3440** - **http://www-rohan.sdsu.edu/faculty/baase/algortext.html#slideshttp://en.wikipedia.org/wiki/Genetic_disorders**
Other Learning Materials	None

2. Facilities Required

Item	Resources
Accommodation (Classrooms, laboratories, demonstration rooms/labs, etc.)	Classroom and Labe available at College of science in Zulfi.
Technology Resources (AV, data show, Smart Board, software, etc.)	All resource are available in the halls
Other Resources (Specify, e.g. if specific laboratory equipment is required, list requirements or attach a list)	

G. Course Quality Evaluation

Evaluation Areas/Issues	Evaluators	Evaluation Methods
Effectiveness of teaching and assessment	Students Reviewers	Questionnaires (course evaluation) filled by the students and electronically organized by the university. Student-faculty and management meetings.
Quality of learning resources	Program Leaders	Direct/indirect

Evaluation areas (e.g., Effectiveness of teaching and assessment, Extent of achievement of course learning outcomes, Quality of learning resources, etc.)

Evaluators (Students, Faculty, Program Leaders, Peer Reviewer, Others (specify)

Assessment Methods (Direct, Indirect)

H. Specification Approval Data

Council / Committee	**Dr Fayez AlFayez** **Dr. Hassan Aly**
Reference No.	4134
Date	13-10-2019