الجامعة الأهلية
AHLIA UNIVERSITY
BAHRAIN

COLLEGE OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION TECHNOLOGY

COURSE SYLLABUS/SPECIFICATION

CODE & TITLE:	**ITCS 303 – Design and Analysis of Algorithms**
WEIGHT:	**(2 - 2 - 3)**
PREREQUISITE:	**ITCS 224 & ITCS 209**
NQF Level Allocated:	**Level 7**
NQF Notional Hours / Credits:	**120 notional hours/ 12 NQF credit**
DESCRIPTION:	The course covers classical techniques and paradigms used in the design and analysis of algorithms. Some of the covered techniques are induction and recursion, divide and conquer, dynamic programming, and greedy approach. Techniques like backtracking and randomization are also introduced to deal with NP-Complete problems. Students will be able to practice their skills on many well-known algorithms and data structures designed to solve practical problems.
OBJECTIVES:	1. To introduce the specialist theories, concepts and principles of problems and algorithms design. 2. To design efficient algorithms for solving different kinds of problems using various classical techniques and paradigms. 3. To analyze the time and space complexities of algorithms. 4. To introduce computability and complexity theories, and study a number of techniques for solving hard problems.

SEMESTER: **ACADEMIC YEAR:**

INSTRUCTOR:
OFFICE TEL.:
EMAIL:

INTENDED LEARNING OUTCOMES (ILOS)

Upon successful completion of the course, students should be able to:

A. Knowledge and Understanding		**NQF Descriptor/ Level**
A1	Concepts and Theories: Demonstrate advanced knowledge and understanding of the concepts and specialist theories of algorithmic design and analysis, algorithm design paradigms, optimal algorithms, complexity theory, P and NP problems, etc.	Knowledge: theoretical understanding [Level 7]
A2	Contemporary Trends, Problems and Research: N/A	N/A
A3	Professional Responsibility: N/A	N/A
B. Subject-Specific Skills		**NQF Descriptor/ Level**
B1	Problem Solving: Identify real world problems and solve them by designing efficient algorithms.	Knowledge: Practical Application [Level 7] Skills: Communication, ICT & Numeracy [Level 7]
B2	Modeling and Design: Develop formal definitions of real world problems, and design their efficient algorithmic solutions using different techniques, such as, divide and conquer, dynamic programming, and the greedy approach.	Knowledge: Practical Application [Level 7]
B3	Application of Methods and Tools: Apply a computer programming language to implement algorithms designed for solving real world problems.	Knowledge: Practical Application [Level 7] Skills: Communication, ICT & Numeracy [Level 7]
C. Thinking Skills		**NQF Descriptor/ Level**
C1	Analytic: Critically analyze and evaluate the asymptotic performance of different algorithms, and find the best and optimal solution for a problem.	Generic Problem Solving & Analytical skills [Level 7]
C2	Synthetic: N/A	N/A
C3	Creative: Demonstrate ability to analyze and design efficient algorithm for solving new problems using different algorithmic techniques.	Generic Problem Solving & Analytical skills [Level 7]
D. General and Transferable Skills (Other Skills Relevant to Employability and Personal		**NQF Descriptor/ Level**
D1	Communication: Show ability to convey ideas and describe processes of designing efficient algorithms in appropriate oral and written forms.	Communication, ICT and Numeracy Skills [Level 7]
D2	Teamwork and Leadership: Work effectively as a member/leader of a team who may plan, design, and	Competence: Autonomy, Responsibility and Context [Level 7]
D3	Organizational and Developmental Skills: Demonstrate ability to utilize ideas of classical algorithms to develop procedures and processes to solve real world	Competence: Autonomy, Responsibility and Context [Level 7]
D4	Ethical and Social Responsibility: N/A	

Course Structure (Outline)						
Week	**Hours**		**ILOs**	**Unit/Module or Topic Title**	**Teaching Method**	**Assessment Method**
	Lec.	**Lab**				
1	2	2	A1	**Introduction to problems:** Types and sizes of problems, problem search space and instances.	Lecture/ In-Class Supervised Work	
2	2	2	A1, B2	**Problem Formulation and Modeling**	Lecture/ Lab Demonstration/ In-Class Supervised Work	In-Class Exercises
3	2	2	A1	**Algorithm Analysis:** Best, worst and average cases, asymptotic analysis.	Lecture/ Lab Demonstration	
4	2	2	A1, C1	**Algorithm Analysis:** Asymptotic notations.	Lecture/ In-Class Supervised Work/ Lab Demonstration (program execution time)	In-Class Exercises/ Quiz 1
5-6	4	4	A1, B1, B2, B3, C1	**Divide and Conquer:** Problem solving such as sorting problems.	Lecture/ In-Class Supervised Work / In-Lab Supervised Work	In-Lab Exercises/ In-Class Exercises/ Quiz 2
7-8	4	4	A1, B1, B2, B3, C1	**Data Structures:** Heaps, operations on heaps and Heap Sort.	Lecture/ In-Class Supervised Work/ In-Lab Supervised Work	Major Test / In-Lab Exercises/ In- Class Exercises
9-10	4	4	A1, B1, B2, B3, C1	**Greedy Approach:** Problem solving such as MST, Prim's and Kruskal's Algorithms.	Lecture/ In-Class Supervised Work/ In-Lab Supervised Work	Quiz 2/ In-Lab Exercises/ In-Class Exercises
11-12	4	4	A1, B1, B2, B3, C1	**Dynamic Programming:** Problem solving, such as shortest paths problem and knapsack problem.	Lecture/ In-Class Supervised Work/ In-Lab Supervised Work	Quiz 3/ In-Lab Exercises/ In-Class Exercises

13	2	2	A1, B1, B2, B3, C1	**Complexity Theory:** P, NP, NP-complete problems and Exhaustive search.	Lecture/ Lab Demonstration	In-Lab Exercises
14	2	2	A1, B1, B2, B3, C1	**Coping with NP-complete:** Backtracking, branch, bound and randomization, parallelization.	Lecture/ In-Lab Supervised Work	In-Lab Exercises
15	2	2	B1, B2, B3, C1, C3, D1, D2, D3	Student Projects	Project Supervision	Evaluation of Project Presentations and Reports
16	2	-	A1, B1, B2, C1, C3	All Topics		Final Exam

TEACHING MATERIALS:

TEXTBOOK(S):

1. Sandeep Sen , Amit Kumar. (2019) *Design and Analysis of Algorithms: A Contemporary Perspective*, 1st Edition, Cambridge University Press, ISBN: 978-1108496827.

HANDOUT(S): Available on Moodle i.e. http://www.ahlia.edu.bh/moodle

REFERENCE(S):

1. Puntambekar A.A. (2010) *Design and Analysis of Algorithms*, Technical Publications.
2. Dormehl L. (2015) *The Formula: How Algorithms Solve All Our Problems-And Create More*, WH Allen.
3. Weiss M. A. (2012) *Data Structures and Algorithm Analysis in Java*, 3rd Edition, Pearson Education.
4. Wayne K. and Sedgewick R. (2014) *Algorithms*, 4th Edition, Addison-Wesley Professional.
5. Sedgewick R. and Flajolet P. (2013) *An Introduction to the Analysis of Algorithms*, 2nd Edition, Addison-Wesley.
6. Levitin A. (2014) *Introduction to the Design & Analysis of Algorithms*, 3rd Edition, Pearson Education.

ASSESSMENTS:

Type of Assessment	Description	ILOs	Weighting
Quizzes	Three written quizzes to be conducted. The quizzes consist mainly of problem solving and analysis questions. Average of best two will be considered.	A1, B1, B2, C1	10%
Project	Student will work as groups of 2-4 members; they will choose a real world problem, analyze it, design an algorithm, implement it and evaluate the performance.	B1, B2, B3, C1, C3, D1, D2, D3	20%
In-Class Exercises	Exercises cover problem solving and analysis questions to help the students in differentiating between the various algorithm designs.	B1, B2, C1	Formative
In-Lab Exercises	Implement the algorithms for solving problems.	B3	Formative
Major Test	In-class One test that will consist of short-answer, essay, problem solving, and algorithm analysis and design questions.	A1, B1, B2, C1	30 %
Final Exam	The final exam is comprehensive and will be of two hours duration. It will consist of short-answer, essay, problem-solving, algorithm analysis and design questions.	A1, B1, B2, C1, C3	40%
Overall			**100%**

Admissions	
Minimum number of students	**5**
Maximum number of students	**25**

Ahlia University values academic integrity. Therefore, all students must understand the meaning and consequences of cheating, plagiarism and other academic offences under the Code of Student Conduct and Disciplinary Procedures (see www.ahlia.edu.bh/integrity for more information).