جامعة الزيتونة الأردنية

Al-Zaytoonah University of Jordan

Course Syllabus

According to JORDAN National Qualification Framework (JNQF)

Course Name: Data Structures

Course Number: 0130221

General Course Information:

Course Title	Data Structures
Course Number	0130221
Credit Hours	3 credit hours
Education Type	Traditional learning
Prerequisites/Co-requisites	Applied Programming (0130231)
Academic Program	Computer Science
Program Code	130
Faculty	Faculty of Science and Information Technology
Department	Computer Science
Level of Course	2
Academic Year /Semester	2024/2025 2nd Semester
Awarded Qualification	BSc
Other Department(s) Involved in Teaching the Course	-
Language of Instruction	English
Date of Production	October 2024
Date of Revision	October 2025

Course Coordinator:

Coordinator's Name	
Office No.	
Office Phone Extension Number	
Office Hours	
E-mail	

Other Instructors:

Instructor Name	
Office No.	
Office Phone Extension Number	
Office Hours	
Email	

Course Description *(English/Arabic)*:

English	An overview of data structure concepts, arrays, stack, queues, trees, and graphs. Discussion of various implementations of these data objects, programming styles, and run-time representations. Applying data structures in problem solving.
Arabic	نظرة عامة على مفاهيم تراكيب البيانات والرزم والطوابير والأشجار والرسوم. مناقشة التطبيقات المختلفة لتراكيب البيانات وأنماط البرمجة وتمثيلات وقت التشغيل. تطبيق تراكيب البيانات في حل المسائل.

Textbook: *Author(s), Title, Publisher, Edition, Year, Book website.*

John Peterson, Data Structures and Algorithms in Java: A Comprehensive Guide, 1st Ed, Amazon Digital Services, 2023.

References: *Author(s), Title, Publisher, Edition, Year, Book website.*

1. Frank Carrano and Timothy Henry, Data Structures and Abstractions with Java: What's New in Computer Science, 5th Ed, Pearson, 2018.
2. Jay Wengrow, A Common-Sense Guide to Data Structures and Algorithms: Level Up Your Core Programming Skills, 2nd Ed, Pragmatic Bookshelf, 2020.
3. Y. Liang and Y. Daniel Liang, Introduction to Java Programming and Data Structures: Comprehensive Version, 12th Ed, Pearson, 2019.
4. Nell Dale and Daniel Joyce, Chip Weems, Object-Oriented Data Structures Using Java, 4th Ed, Jones & Bartlett, 2016.
5. Lucien Sina, Algorithms and Data Structures, epubli, 2025.

Course Educational Objectives (CEOs):

CEO1	Analyze, design and implement efficient and reliable computer programs.
CEO2	Recognize the concept of an Abstract Data Type (ADT).
CEO3	Design and implement some user-defined data structures (lists, stacks, queues, linked lists, binary trees, etc.) as Java generic classes.

Intended Learning Outcomes (ILO's):

Intended learning outcomes (ILOs)		Relationship to CEOs	Contribution to PLOs	Bloom Taxonomy Levels*	JNQF Descriptors **
K	Knowledge and Understanding				
ILO1-K	Understand the design of well-known generic data structures	CEO2	PLO2-K	Understanding	K
ILO2-K	Understand the relationship between data structures and algorithm design	CEO3	PLO2-K	Understanding	K
S	Skills				
ILO3-S	Determine how an ADT is designed and implemented as a class of an object-oriented programming language	CEO2	PLO4-S	Analyzing	S
ILO4-S	To be able to build and use object-oriented classes and computer programs for data structures	CEO3	PLO4-S	Applying	S
ILO5-S	To use data structures in providing solutions to different problems	CEO3	PLO3-S	Applying	S
C	Competencies				
ILO6-C	Ability to design new efficient and reliable data structures to solve problems	CEO1	PLO5-C	Creating	C
ILO7-C	Ability to implement different data structures in different programming languages	CEO3	PLO6-C	Creating, Evaluating	C

***Bloom Taxonomy Levels:**

Level #	1	2	3	4	5	6
Level Name	Remembering	Understanding	Applying	Analyzing	Evaluating	Creating

**** Descriptor (National Qualification Framework Descriptors): K: Knowledge, S: Skill, C: Competency.**

Program Learning Outcome (PLOs):

(PLOs)		JNQF Descriptors**		
		K	S	C
PLO1-K	Knowledge of professional ethics, social responsibility, and the regulations governing them.	X		
PLO2-K	Understanding various programming techniques, the stages of software development, and the fundamental principles of security.	X		
PLO3-S	Skill in applying mathematical concepts to analyze and design algorithms and verify their correctness		X	
PLO4-S	Skill in using different programming languages and applying them to develop software and computer applications.		X	
PLO5-C	The ability to analyze, design, and develop effective and reliable computer programs that meet user requirements and adhere to professional ethics.			X
PLO6-C	The ability to keep up with continuous advancements in computer science, innovate, and work independently or as part of a team.			X
PLO7-D	The ability to work collaboratively, communicate effectively, and demonstrate teamwork spirit.			X

**** Descriptors according to the national qualifications framework (K: knowledge, S: skill, C: Competency)**

Weekly Schedule *(please choose the type of teaching)*

- ☒ **Face to Face (F2F)**
- ☐ **Hybrid** *(One - To - One)*
- ☐ **Online**

Schedule of Simultaneous and their Topics:

Week	First Lecture	Second Lecture	ILOs	PLOs	JNQF Descriptors*
1	**Concept of an ADT**: Definition, representation, and operations	Abstract Data Type. Big-O Analysis	ILO1-K	PLO2-K	K
2	**Linked Lists**: Array vs. Linked List, simple operations	Simple Linked List Implementation	ILO1-K	PLO2-K	K
3	**Stack**: Stack ADT definition, array & linked stack implementation	Applications of stacks	ILO3-S	PLO4-S	S
4	**Recursion**: Recursive definitions, how recursion works.	Classic examples of recursion	ILO2-K	PLO2-K	K
5	**Queue**: Queue ADT	Queue array implementations	ILO3-S	PLO4-S	S
6	Queue implementation as a linked structure	Applications of queues	ILO4-S	PLO4-S	S
7	**List**: List ADT specifications, Sorted and Unsorted Lists	Array implementation of lists	ILO6-C	PLO5-C	C
Midterm Exam (30%)					
9	Binary Search algorithm, recursive binary search	Linked List Implementation	ILO5-S	PLO3-S	S

10	Circular linked lists, doubly linked lists, linked lists with headers	Operations on linked lists	ILO7-C	PLO6-C	C
11	**Graphs & Trees**: Basic concepts and definitions of graph theory	Basic types of Graphs & Trees	ILO1-K	PLO2-K	K
12	**Binary Trees**: BT ADT, array & linked representations of trees	Binary Tree Traversal	ILO5-S	PLO3-S	S
13	**Binary Search Trees**: Specifications & Examples	Binary Search Tree implementation	ILO1-K	PLO2-K	K
14	Recursive Binary Search Tree operations	Binary Search Tree Constructions	ILO6-C	PLO5-C	C
15	Adding and removing nodes of binary search trees	Tree Complexity Analysis	ILO7-C	PLO6-C	C
16	**Final Exam**				

* **K: Knowledge, S: Skills, C: Competency**

Teaching Methods and Assignments:

Development of ILOs is promoted through the following teaching and learning methods:

- Lecture.
- Learning through projects.
- Learning through problem solving.
- Participatory learning

Course Policies:

A- Attendance policies:
The maximum allowed absences is 15% of the lectures.
B- Absences from exams and handing in assignments on time:
Midterm exam can be retaken based on approval of excuse by the instructor's discretion.
Not handing assignment on time will incur penalties.
C- Academic Health and safety procedures
D- Honesty policy regarding cheating, plagiarism, and misbehaviour:
Cheating, plagiarism, misbehaviour will result in zero grade and further disciplinary actions may be taken.
E- Grading policy:

- All homework is to be posted online through the e-learning system.
- Exams will be marked within 72 hours and the marked exam papers will be handed to the students.
- Activities (Course Videos, Homework, Quizzes, Project) **20%**
- Midterm 30**%**
- Final Exam 50**%**

F- Available university services that support achievement in the course: **E-Learning Platform, Labs, Library.**

Required Equipment:

- PC / Laptop with webcam and mic
- Internet Connection
- Access to the ZUJ E-Learning Platform
- E-learning plan
- Satisfaction questionnaires for online and face-to-face learning
- Software for e-learning
- Training

Assessment Tools Implemented in the Course:

- Final Exam
- Midterm Exam
- Quizzes (Optional)
- Homework
- Programming Project

Responsible Persons and their Signatures:

Course Coordinator		**Completed Date**	/ /
		Signature	

Received by (Department Head)		**Received Date**	/ /
		Signature	