

"حيث تصبح الرؤية واقعاً"
"When Vision Becomes Reality"

"عراقة وجودة"
"Tradition and Quality"

| | |
|---|---------------|
| نموذج خطة المادة الدراسية - إجراءات إعداد الخطة الدراسية وتحديثها / قسم علم الحاسوب | QF01/0408-4.0 |
|---|---------------|

| | | | |
|---|---|---|----------------------|
| علم الحاسوب | التخصص | 2022/2021 | رقم الخطة الدراسية |
| Data Structures | اسم المادة الدراسية | 0112212 | رقم المادة الدراسية |
| تراكيب البيانات | المتطلب السابق للمادة | 3 | عدد الساعات المعتمدة |
| Object Oriented Programming | برمجة كينونية | | |
| <input type="checkbox"/> متطلب تخصص اختياري | <input type="checkbox"/> متطلب كلية اجباري | <input type="checkbox"/> متطلب جامعة اجباري | نوع المادة الدراسية |
| <input type="checkbox"/> متطلب اجباري | <input checked="" type="checkbox"/> متطلب عائلة تخصص/ مساند | <input type="checkbox"/> تعلم الكتروني كامل | نمط تدريس المادة |
| <input type="checkbox"/> تعلم وجاهي | <input checked="" type="checkbox"/> تعلم مدمج | <input type="checkbox"/> (1 متزامن: 1 غير متزامن) | النموذج التدريسي |
| <input type="checkbox"/> 2 وجاهي | <input checked="" type="checkbox"/> (2 وجاهي: 1 غير متزامن) | | |

معلومات عضو هيئة التدريس والشعب الدراسية (تعباً في كل فصل دراسي من قبل مدرس المادة)

| | | | | |
|---------------------------------|-------------------|------------|------------|-------------------|
| الاسم | الرتبة الأكاديمية | رقم المكتب | رقم الهاتف | البريد الالكتروني |
| د. أيمن محمود عبد الله | أ. مشارك | 09/330 | 391 | ayman@zuj.edu.jo |
| الساعات المكتبية (اليوم/الساعة) | | | | |
| رقم الشعبة | وقتها | مكانها | عدد الطلبة | نمط تدريسها |
| | | | | النموذج المعتمد |
| | | | | 1:2 |
| | | | | مدمج |
| | | | | 1:2 |
| | | | | مدمج |

الوصف المختصر للمادة الدراسية

Principles of data design. Data types and structures. Abstract data types (ADTs) and encapsulation. Unsorted List and Sorted List ADTs. Stack and Queue ADTs. Linked structures. Implementing Unsorted Lists, Sorted Lists, Stacks and Queues as linked structures. Programming with recursion. Binary Search Trees.

مصادر التعلم

| | |
|---|---|
| Nell Dale, Daniel Joyce, Chip Weems, <i>Object-Oriented Data Structures Using Java</i> , 4 th Ed., Jones & Bartlett, 2016 | معلومات الكتاب المقرر |
| 1. Frank Carrano, Timothy Henry, <i>Data Structures and Abstractions with Java: What's New in Computer Science</i> , 5 th Ed., Pearson, 2018. 2. Jay Wengrow, <i>A Common-Sense Guide to Data Structures and Algorithms: Level Up Your Core Programming Skills</i> , 2 nd Ed., Pragmatic Bookshelf, 2020. 3. Y. Liang, Y. Daniel Liang, <i>Introduction to Java Programming and Data Structures: Comprehensive Version</i> , 12 th Ed., Pearson, 2019. | مصادر التعلم المساندة |
| https://elearning.zuj.edu.jo | المواقع الالكترونية المساندة |
| <input type="checkbox"/> أخرى | <input checked="" type="checkbox"/> منصة تعليمية افتراضية |
| <input type="checkbox"/> مختبر/ مشغل | <input checked="" type="checkbox"/> قاعة دراسية |
| Java Beans or any other compiler or language such as C++ or Python | التجهيزات والبرمجيات اللازمة |
| | دعم ذوي الاحتياجات الخاصة |
| | للحصول على الدعم الفني/التقني |

(K= Knowledge, S= Skills, C= Competences)

مخرجات تعلم المادة الدراسية

| الرقم | مخرجات تعلم المادة | رمز مخرج تعلم البرنامج المرتبط |
|-----------------|--|--------------------------------|
| المعارف | | |
| K1 | Comprehend the concept of an Abstract Data Type (ADT) | MK2 |
| K2 | Understand the design of well-known generic data structures | MK2, MK4 |
| K3 | Understand the applications of different data structures | MK2, MK4 |
| K4 | Understand the relationship between data structures and algorithm design | MK2, MK4 |
| المهارات | | |
| S1 | Determine how an ADT is designed and implemented as a class of an object-oriented programming language | MS1, MS2 |
| S2 | To be able to build and use object-oriented classes and computer programs | MS2 |
| S3 | To design and implement some user-defined data structures as generic classes | MS2 |
| S4 | To use data structures in providing solutions to different problems | MS2, MS3 |
| الكفايات | | |
| C1 | Ability to design new data structures to solve problems | MC2 |
| C2 | Ability to design, analyze, and implement efficient and reliable computer programs | MC3 |
| C3 | Ability to implement different data structures in different programming languages | MC4 |

آليات التقييم المباشر لنتائج التعلم

| نوع التقييم/ نمط التعلم | التعلم الإلكتروني | التعلم المدمج | التعلم الوجاهي (مواد نظرية) | التعلم الوجاهي (مواد عملية) |
|----------------------------------|-------------------|---------------|-----------------------------|-----------------------------|
| امتحان منتصف الفصل | %30 | %30 | %40 | %30 |
| المشاركة/ التطبيقات العملية | 0 | 0 | %10 | %30 |
| النشاطات التفاعلية غير المتزامنة | %30 | %30 | 0 | 0 |
| الامتحان النهائي | %40 | %40 | %50 | %40 |

ملاحظة 1: النشاطات التفاعلية غير المتزامنة هي النشاطات والمهام والمشاريع والواجبات والأبحاث والدراسات والمشاريع والعمل ضمن مجموعات طلابية... الخ، والتي ينفذها الطالب ذاتياً، بواسطة المنصة الافتراضية دون لقاء مباشر مع مدرس المادة.

جدول اللقاءات المتزامنة/ الوجاهية وموضوعاتها

| الأسبوع | الموضوع | أسلوب التعلم* | المرجع** |
|---------|--|----------------------|-------------------------------|
| 1 | Introduction to Data Structures: What is a data structure? References, arrays, Big-O Analysis. Concept of an ADT: Definition of an Abstract Data Type, representation of objects, implementation of operations. | Lecture & Discussion | Text: 27-77 |
| 2 | Introduction to Linked Lists: Array vs. Linked List, simple operations on Linked Lists. Stack ADT: Stack ADT definition and array implementation. | Lecture & Discussion | Text: 111-115 Text: 78-100 |
| 3 | Stack ADT: Linked stack implementation, applications of stacks. | Lecture & Discussion | Text: 121-141 |
| 4 | Recursion: Recursive definitions, how recursion works, classic examples, recursive processing of linked lists, deciding when to use recursion. | Lecture & Discussion | Text: 161-181 |

| المرجع ** | أسلوب التعلم * | الموضوع | الأسبوع |
|---------------|----------------------|--|---------|
| Text: 217-236 | Lecture & Discussion | Queue ADT: Queue ADT definition and its linear array implementation, circular queue implementation. | 5 |
| Text: 237-247 | Lecture & Discussion | Queue ADT: Queue implementation as a linked structure, applications of queues. | 6 |
| Text: 345-354 | Lecture & Discussion | List ADT: List ADT specifications, array implementation of sorted and unsorted lists. | 7 |
| Text: 355-360 | Lecture & Discussion | List ADT: Binary Search algorithm, recursive binary search, implementing List ADT as a linked structure. | 8 |
| --- | Discussion | Review + Midterm Exam | 9 |
| Text: 380-385 | Lecture & Discussion | List ADT: Circular linked lists, doubly linked lists, linked lists with headers and trailers, operations on linked lists. | 10 |
| Text: 421-425 | Lecture & Discussion | Graphs & Trees: Basic concepts and definitions of graph theory, common types of trees. | 11 |
| Text: 429-434 | Lecture & Discussion | Binary Trees: Binary tree ADT, array & linked representations of trees, binary tree traversal. | 12 |
| Text: 435-442 | Lecture & Discussion | Binary Search Trees: Binary search tree specification, binary search tree implementation. | 13 |
| Text: 443-448 | Lecture & Discussion | Binary Search Trees: Recursive binary search tree operations, adding and removing nodes. | 14 |
| --- | Discussion | Review: General review of the course topics. | 15 |
| --- | --- | Final Exam | 16 |

* اساليب التعلم: محاضرة، تعلم معكوس، تعلم من خلال المشاريع، تعلم من خلال حل المشكلات، تعلم تشاركي ... الخ.
 ** المرجع: صفحات في كتاب، قاعدة بيانات، محاضرة مسجلة، محتوى على منصة التعلم الالكتروني، فيديو، موقع... الخ

جدول النشاطات التفاعلية غير المتزامنة (في حال التعلم الالكتروني والتعلم المدمج)

| النتائج المتوقعة | المرجع | المهمة / النشاط | الأسبوع |
|------------------|---------------|--|---------|
| Submitted Work | Text: 27-77 | Define an Abstract Data Type | 1 |
| Submitted Work | Text: 111-115 | Introduction to linked lists | 2 |
| Submitted Work | Text: 78-100 | Array implementation of the stack | 3 |
| Submitted Work | Text: 121-141 | Linked stack implementation | 4 |
| Submitted Work | Text: 121-141 | Applications of stacks | 5 |
| Submitted Work | Text: 217-236 | Circular queue implementation | 6 |
| Submitted Work | Text: 237-247 | Queue implementation as a linked structure | 7 |
| Submitted Work | Text: 237-247 | Applications of queues | 8 |
| Discussions | --- | Midterm Exam Review | 9 |
| Submitted Work | Text: 345-354 | Array implementation of lists | 10 |
| Submitted Work | Text: 355-360 | Linked list implementation | 11 |
| Submitted Work | Text: 380-385 | Operations on linked lists | 12 |
| Submitted Work | Text: 429-434 | Binary tree traversal | 13 |
| Submitted Work | Text: 435-442 | Binary search tree | 14 |
| Submitted Work | --- | Review questions | 15 |