Course program and reading list

Semester 2 Year 2024

School: Efi Arazi School of Computer Science B.Sc

Data Structures

Lecturer:

Prof. Yael Moses yael@runi.ac.il

Tutors:

Mr. Itamar Azulay itamar.azulay@post.runi.ac.il

Ms. Yaheli Neemani yaheli.neemani@post.runi.ac.il

Mr. Yaniv Nikankin yaniv.nikankin@post.runi.ac.il

Teaching Assistant:

Mr. Itamar Azulay itamar.azulay@post.runi.ac.il

Ms. Noa Kochav kochav.noa@post.runi.ac.il

Ms. Maayan Yesharim maayan.yesharim@post.runi.ac.il

Mr. Haytham Hammud haytham.hammud@post.runi.ac.il

Ms. Shiri Ron shiri.ron01@post.runi.ac.il

Mr. Ben Jaim Catalan benjaim.catalan@post.runi.ac.il

Mr. Yaniv Nikankin yaniv.nikankin@post.runi.ac.il

Course No.:	**Course Type :**	**Weekly Hours :**	**Credit:**
59	Lecture	5	5

Course Requirements :	**Group Code :**	**Language:**
Final Exam	242005901	Hebrew

Prerequisites

Prerequisite:

417 - Introduction To Computer Science

Course Description

As the saying goes, "good data structures and bad code are much better than the other way around". Indeed, in order to solve challenging problems in an efficient and elegant manner, one must organize one's data in a suitable form, called data structure. Data structures are an essential area of study and practice for computer scientists and serious software developers. The course presents and analyzes classical data structures and related algorithms. In particular, we will learn how to assess the performance of various algorithms for searching, sorting, and manipulating data. We will learn widely-used data structures such as lists, stacks, queues, various trees, and hash tables. In addition, we will then learn the relations between the data structures used and the efficiency of the algorithms using them.

List of subjects to be covered :

Time & space complexity: Best case, worst case, average case, and amortized analysis. We will use O, Ω, Q bounds.

List of Data structures: linked list, stack, queue, binary heap, binary search tree, binary search tree with augmented data (e.g., interval tree), AVL trees, B-trees, hash table, upside-down forest (disjoint union / find)

Algorithms: In addition to the algorithms related to the data structures, we will also learn comparison based sorting algorithms (heap sort, quick sort, merge sort), linear sorting algorithms (radix sort, counting sort and bucket sort), ordered statistics algorithms, and the median-of-median algorithm.

Lower bounds: we will prove a lower bound of $\Omega(n \log n)$ for comparison-based sorting algorithms.

Note: because the semester is shorter than usual, we may not cover all of these topics.

Course Goals

Be familiar with basic data structure and classic sorting algorithms.

Know how to formal analyze the complexity of various algorithms

Know how to choose the right data structure for a given task.

Grading

There will be 6 theoretical (dry) assignments and 2 programming (wet) assignments.

The wet assignments can be submitted in pairs.

The HW grade is the average of ALL assignments.

Final grade: if final_exam_grade < 60 then fail (sorry)

else, final_grade = 0.84 final_exam_grade + 0.16 HW_grade

Learning Outcomes

The students will master the folllowing topics and gain the following capabilities:

- Abstract data types
- Basic data structures
- Basic sorting algorithms
- Space & time analysis of algorithms
- Lower bound proof
- How to choose a data structure for a given task
- How to modify a data structure for a given task
- Implement data structures

Lecturer Office Hours

Office hours will be posted on the course's website.

Tutor Office Hours

Office hours will be posted on the course's website.

Additional Notes

Generative AI policy: It is prohibited to use AI tools to produce solution to homework problems.

Reading List

Textbook
Introduction to Algorithms by Cormen, Leiserson, Rivest and Stein. (CLRS)
A Hebrew translation exists (by Open univ).