Course program and reading list
Semester 2 Year 2022

School:  Efi Arazi School of Computer Science B.Sc

Data Structures

Lecturer:
Dr. Ben Lee Volk  benlee.volk@idc.ac.il

Tutors:
Ms. Yael Hitron  yael.hitron@post.idc.ac.il

Teaching Assistant:
Ms. Yael Hitron  yael.hitron@post.idc.ac.il
Mr. Guy Kornowski  guy.kornowski@post.idc.ac.il
Mr. Gilad Battat  gilad.battat@post.idc.ac.il
Mr. Aviad Baron  Aviad.Baron@post.idc.ac.il
Mr. Elad Tzalic  elad.tzalic@post.idc.ac.il

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

Course Requirements: Final Exam  Group Code: 222005903  Language: English

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 amortize case. We will use $O$, $\Omega$, $\Theta$ 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, KD-tree, 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 the lower bound of comparison based algorithm.

Course Goals

The goal of the course is to learn basic data structure and classic sorting algorithms. We will learn formal analysis of the complexity of various algorithms. In addition, we will learn how to choose a data structure for a given task.

Grading

If final exam < 60 then
grade = fail
else

\[ \text{grade} = 0.1 \times \text{HW} + 0.9 \times \text{final exam} \]

**Note:** all assignments will be considered.

---

## Learning Outcomes

Learn:

- Abstract data type
- 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 structure

---

## Lecturer Office Hours

TBA

---

## Tutor Office Hours

TBA

---

## Teaching Assistant

International classes: Yael Hitron

Hebrew classes: Guy Kornowski, Elad Tzalic, Maya Goldner

---

## Reading List

**Textbook**

*Introduction to Algorithms* by Cormen, Leiserson, Rivest and Stein. (CLRS)

A Hebrew translation exists (by Open univ.)