Module [302]

Elementary Data Structures

Module Overview

This module introduces several elementary data structures and their implementation. It will allow us to review topics such as dynamic memory allocation and pointers in a very applied context. We will also consider recursion as a design approach to handle elegantly self-referencing data structures.

As usual, our learning activities will be divided in two groups. The “discovery week” will focus on the reading assignments and understanding the main Concepts. The “apply week” will help you put this newly acquired knowledge into practice and evaluate your understanding of the entire module through graded assessments. Refer to the Syllabus for more details.

Learning Outcomes

By the end of this module, you will gain the following knowledge:

Programming Concepts

- Self-referencing data structures
- Recursion applications to self-referencing data structures
- Linked lists and derived abstract data types
- Hash Tables principles

Designing Programs

- Self referencing data structures
- Recursion
- Re-using abstract data types to implement other abstract data types (e.g. using linked lists to implement queues and stacks)

Implementing Programs (in C)

- Manipulating data structures with pointers
- Dynamically allocating structures
- Implementing abstract data types based on linked lists (e.g. queues, stacks, binary trees)
- Understanding the implementation of a simple Hash Table

Troubleshooting

- Using memory diagrams and stack diagrams to trace through the execution of programs using pointers to link data structures