Sistemas de Operação / Fundamentos de Sistemas Operativos

(Ano letivo de 2023-2024)

Guiões das aulas práticas

Quiz #CPP/01 — A brief revision on C++ and linked-lists

Summary

- C/C++ programming
- Implementing a unidirectional linked-list in pure C/C++

Exercice 1 *Implementing a linked-list as a C++ module*

The objective of this exercice is to implement a simple unidirectional linked-list in C++. The elements of the list, called `Registers`*, are composed of two fields:*

- *a 32-bit unsigned integer, representing a student number;*
- *a string, used to store the student's name (in C a string is implemented as a memory address of a sequence of characters terminating with a zero value).*

(a) *Files* `linked-list.{h,cpp}` *partially implement the unidirectional linked-list as a C++ module. The support data structure and the interface are defined, but the bodies of the manipulation functions are to be implemented. Read these files carefully and try to the following questions.*

- **What is the purpose of the pattern `#ifndef #define` in `linked-list.h` file?**
- *All of linked-list module functions have a* `Node*` *first argument.* **Why?**
- *The linked-list module functions are defined within a namespace, named* `base`*.* **Is there any advantage in doing so?**

(b) *File* `main.cpp` *is the main program which implements a menu driven application. It is also only partially implemented. Read it carefully and try to understand it.*

(c) *Complete both the linked-list module and the main program, starting with module functions (and correspondent main functions):* `append`*, and* `print`*.*

(d) *Implement all functions.*
