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

Summary

- C/C++ programming

- Implementing a unidirectional linked-list in pure 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 `Register`*, are composed of two fields:*

- *a 32-bit unsigned integer, representing a student number;*

- *a string, used to store the student's name.*

(a) *Files* `ull.{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 understand them.*

- *The support data structures (`Register` and `Node`) are declared in the .cpp file, not in the header file. Can you figure out why?*
- *Pointer* `head` *in file* `ull.cpp` *is declared* `static`*. What's the consequence of this?*
- *The interface functions are defined within namespace* `ull`*. 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.*

(d) *Some extra work:*

- *Add the possibility to store the list to a file.*
- *Add the possibility to operate in non-interactive mode, for instance, allowing to input a file and store the registers sorted in a different way.*