Homework Assignment 2

- Use C++ language (C++23 or before). Your code must compile on the latest version of GCC.
- The source and header files should be named as **學號_hw2.cpp** and **學號_hw2.h**, respectively.
- Please provide a **Makefile** that supports **make all** to compile each source code file into a standalone executable.
- Please include a brief **README** describing what's special have been done.
- Compress all files into a **學號_hw2.rar** archive. All letters in the file name are in lower cases. Do *not* include any executables in the archive!
- Upload your archival file to HW2 entry of E3. Submission deadline: **9/22, 23:59**.

Homework Assignment 2 (Cont'd)

- Linked Lists (implemented in **學號_hw2.h** and **學號_hw2.cpp**)
  - Implement a **`Node`** class. The only field in a Node is an **`int k`**, the unique key.
  - Implement two variants of linked lists (**singly** and **XOR**). Name your classes as **SingleList** and **XORList**, respectively.
  - Each variant should support the following methods:
    - **`list_walk()`**: Print "List: ", then print all keys in order, separated by a comma.
    - **`list_insert(int k)`**: Print "Inserted k\n".
    - **`list_search(int k)`**: Print "Found k\n" or "Not found k\n".
    - **`list_delete(Node* n)`**: Print "Deleted k\n".
    - **`list_ins_del(int k)`**: Insert key *k* if it does not exist (Print "Inserted k\n"), delete it otherwise (Print "Deleted k\n").
    - **`list_reverse()`**: reverse the entire list.

Homework Assignment 2 (Cont'd)

- In the source file (**學號_hw2.cpp**), include driver functions **slist_test()** and **xlist_test()** to test the linked lists.
  - The driver functions are invoked by your **main()** in order; **SingleList** then **XORList**.
  - Insert 2×10^5 unique and uniformly-random integers in $[1, 10^6]$ into each list, and insert-or-delete 2×10^5 random integers in the same range.
  - Then, call **list_walk()** to print the content of the list.
  - Call **list_reverse()**, and then **list_walk()** again.
  - Use **std::chrono::high_resolution_clock** to measure the time elapsed.