PERSONAL HOMEWORK 2: LINKED LIST - STACK - QUEUE RESEARCH

1 Introduction

In this Homework, you are required to analyze the complexity of some basic operations in Linked List, Stack, Queue.

You must analyze the complexity in two ways:

- Using theory. In this way, you present the algorithm by drawing a figure step by step and then conclude the complexity.
- Using experiment. In this way, you need to implement basic operations. Then, you count the number of assignments and comparisons to draw conclusions about complexity.

2 Exercise

2.1 Exercise 1 (3-4 points) : Linked List

We know that both array and Linked List can store list of values. We already have array. Why Linked List?

Consider operation `"Remove first item in list of 10.000.000 values"`:

- If you use array, you may fall asleep due to long long long time to wait.
- If you use linked list, the operation completes immediately.

Some problems require ultra fast `"remove first item in list"`. You may consider Linked List instead of array.

In this exercise, you are required to analyze the complexity of the basic Linked List operations, including:

1. `InsertHead (InsertFirst, AddFirst)`.
2. `RemoveTail (RemoveLast)`.
3. `RemoveAtPosition`.

Note: If you just analyze the complexity of these basic operations in Linked List, you get 3 points. If you compare with arrays, you get the full 4 points.

2.2 Exercise 2 (3 points) : Queue

When go to checkout in supermarket, everyone must be arranged in a queue.

- First person in queue will be served first.
- Last person in queue will be served last.

Queue obeys FIFO rule (FIFO = "First In, First Out").

In this exercise, you are required to analyze the complexity of the basic Queue operations (using Linked List for implementation), including:

1. `Enqueue`.
2. `Dequeue`.
3. `GetFront (Peek)`.

2.3 Exercise 3 (3 points) : Stack

Stack is a linear data structure which follows a particular order in which the operations are performed. The order may be LIFO (Last In First Out) or FILO(First In Last Out).

There are many applications of a stack, such as:

- Expression evaluation.
- Conversion from one form of expression to another.
- Memory Management.
- Backtracking problems.
- ...

In this exercise, you are required to analyze the complexity of the basic Stack operations (using Linked List for implementation), including:

1. `Push`.
2. `Pop`.
3. `IsEmpty`.

3 Submission

- The submission file must be in the following format: [**Student_ID.zip**], is the compression of the [**Student_ID**] folder. This folder contains:
  - The report file must be presented as a document [**Student_ID.pdf**]. This file presented your research from 3 exercises.
    - Information (Names, Student IDs) must be declared clearly on the first page of your report. Your working progress (How much work have you completed?) should be demonstrated on this page, too.
    - The report file should be **structured, logical, clear** and **coherent**. The length of the submission should not exceed 15 pages for the document file.
    - All links and books related to your submission must be mentioned.
  - The programming file must be a single file [**Student_ID.cpp**]. The code fragment must be clear, logical and commented.

- Submission with wrong regulation will result in a "0" (zero).
- Plagiarism and Cheating will result in a "0" (zero) for the entire course and will be subject to appropriate referral to the Management Board of the CLC program for further action.

THE END