CSCI 235 – Software Design & Analysis II
(Homework) Assignment 2

Introduction

Before starting this assignment, read the following programming rules:
http://www.compsci.hunter.cuny.edu/~sweiss/course_materials/csci235/programming_rules.pdf

This assignment tests your ability to implement Linked Lists and Array-Based Lists.. The program must compile and run on the computers in 1000G HN (G-lab). As with all assignments, you should feel free to discuss the assignment with anyone or to get parts of code from others. (If you do get code from others, you must cite their contribution. Although this is a programming assignment, it will count as a homework assignment and must be performed individually.

Assignment

Attached to this assignment, you will find three files (`test.cpp`, `ABCList.hpp` and `ABList.hpp`). Details for these three files:

- `test.cpp` – This file contains "main" and other supporting global functions. It allows the user to select an implementation (Linked List or Array-Based List) and allows the user to enter data, etc. to test the implementations.
- `ABCList.hpp` – This file contains the *abstract base class* declaration for the List.
- `ABList.hpp` – This file contains a complete declaration and incomplete definition for the Array-Based implementation of the List. Two functions have been completed (the constructor, `ABList`, and the function, `isEmpty`.

Your assignment is to perform the following:

1) Complete the definition of the functions in `ABList.hpp`, specifically: `getLength`, `insert`, `remove` and `retrieve`. With the exception that your implementation must throw instances of integers when it encounters an error, your implementation should not require any changes to the class declaration, but should only require that you add these function definitions.
2) Create a file, `LinkedList.hpp` with declarations and definitions similar to those found in `ABList.hpp`, complete with the potential of throwing errors, but which implements a pointer-based List or a Linked List (i.e. the "private" section of the class declaration must be different).

Your changes to `ABList.hpp` and your new `LinkedList.hpp` must work with an unaltered version of `test.cpp`, with the exception of throwing errors where necessary.

Submission

Submit only your `ABList.hpp` and `LinkedList.hpp` on Blackboard. Submit these as one file tar.gz file with your first initial + last name (eg. dbrizan.tar.gz).

Grading

Your grade will be based on how the implementations perform with the `test.cpp` file

- 45% = ABList implementation
- 55% = Linked List declaration and implementation

There is a 10% late penalty per day after the first day.