CSCI 235 – Software Design & Analysis II
Assignment 4

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 the two List types – ArrayList and LinkedList. The program must compile and run on the server eniac.cs.hunter.cuny.edu or one of the machines in G-Lab. You must work on this assignment individually.

Your grade will be based on the following:
55% = Correctness (conformance to the requirements below)
The program must compile and run on one of the G-Lab machines. In addition, it must perform the functions outlined in the "Assignment" section.
10% = Design
The program must show a reasonable object-oriented decomposition of the assignment into classes.
20% = Performance
The implementation must be as efficient as possible in terms of the amount of memory used and in terms of the number of computational cycles used.
15% = Documentation and style
The implementation must have good comments; variables and functions must have reasonable names, and the submissions must have instructions on how to compile the classes.
There is a 10-point penalty per day after the first day.

Background

This program will have you implement two List types discussed in class. As such, you will demonstrate your knowledge of templates and classes in a parent-child (i.e. (abstract) base class and derived class) relationship.

Assignment

Attached to this assignment is the file A4.tar.gz, which contains 4 C++ source files:

- driver.cpp = A driver file, containing "main" and supporting include statements
- ListTester (ListTester.hpp, ListTester.cpp) = A class for testing List instances
- List.hpp = The Abstract Base Class for List which is the parent class for ArrayList and LinkedList

Write C++ classes and statements sufficient to complete the attached program. Specifically, you must write implementations for ArrayList (in ArrayList.hpp) and LinkedList (in LinkedList.hpp). You are allowed to add any classes you wish (eg. exception classes), but you are not allowed to modify any existing code. When complete, the implementation should provide a suitable test – though be aware that some of the prompts may be misleading.

Submission

Submit your source code on Blackboard. Only include ArrayList.hpp, LinkedList.hpp and any exception classes. Submit all your code and documentation as one "tar.gz" file. A tar file concatenates a bunch of different files (without compressing them). A gz (gzip) file compresses a single file. You can create a tar file (named "a4.tar") in the same directory from three files (main.cpp This.hpp This.cpp) with the following command:

```
tar -cvf ./a4.tar ArrayList.hpp LinkedList.hpp ListException.hpp
```

You can gzip the tar file above with the following command:

```
gzip a4.tar
```

This creates a file in the same directory called "a4.tar.gz", which is what you should submit on Blackboard.