IT230 Lab 6
Score: ______/10 v1.1

The purpose of this lab is to get practice writing C++ code on your own.

Specifications

C++ is a great language to demonstrate linked lists. This is because C++ has built-in pointers, which makes node traversal natural. Implement the LinkedListBase, the SinglyLinkedList, and DoublyLinkedList classes.

Because there will be an abstract class, utilize pure virtual functions, similar to the last C++ lab with inheritance.

Make sure all three classes allow generic types.

Constraints

- Test all of your methods in your main the same way you did in C#.
- Because C++ does not have properties, utilize getter methods.
- You must not have memory leaks. **Every node in the linked list must be freed.**
- All methods defined in your source file must have its header in the header file (.h).
- C++ uses camelCase. Name variables and methods using camelCase. Classes should be named with PascelCase. See the tips for help.
- If a method will not affect the state of the object, mark the method as constant.

Submission

The submission for this lab is the commit ID you want to be graded. You will submit the commit ID via the Canvas assignment. If you need help finding the commit ID, ask for help.

10 points will be assigned by correctness and your observable understanding as reflected in your code.

Remember to write efficient code. Optimize your code wherever possible.

Tips

- There is a line in the C# `ToString()` override that has this snippet:
  `this[i]`
  It can be translated to:
  `this->operator[](i)`
  - Make sure the indexing operator is marked constant. It can be constant and a pure virtual method.
- When the head or tail’s next or previous pointer should be null, make sure you assign it `nullptr`.