Assignment 7 – an ADT Front List

Due Date: 11:59 pm, November 17, Monday

Objectives

- Students will gain an experience to use public inheritance to derive a class
- Students will gain an experience to use private inheritance to derive a class
- Students will gain an experience to use containment relationship to define a class

Problem

Consider an ADT `FrontList`, which restricts insertions, removals, and retrievals to the first item in the list. These operations can be defined as

```
bool insert(const ItemType& newEntry);
bool remove();
Itemtype retrieve() const;
```

Define an abstract class (interface) for the ADT `FrontList`, and then implement the class `FrontList` in the following three different ways:

1. Store the list's entries in an instance of `LinkedList`
   [Hint: This is similar to the class `SortedListHasA.` Please see the definition, implementation and its tester of `SortedListHasA` on D2L.]

2. Derive the class `FrontList` from `LinkedList` using public inheritance.
   [Hint: This is similar to the class `SortedListIsA.` Please see the definition, implementation and its tester of `SortedListIsA` on D2L.]

3. Derive the class `FrontList` from `LinkedList` using private inheritance.
   [Hint: This is similar to the class `SortedListAsA.` Please see the definition, implementation and its tester of `SortedListAsA` on D2L.]

Other requirements

- Add default and copy constructors following `LinkedList` class.
- Add a destructor `LinkedList` class.
- Disable the following member functions when using public inheritance because they may cause problems.

  ```
  bool insert(int newPosition, const ItemType& newEntry);
  bool remove(int position);
  void replace(int position, const ItemType& newEntry)
                              throw(PrecondViolatedExcep);
  ```

- It is noticed that if `remove(position)` is disabled in the case of public inheritance, it would cause a problem for `clear()` to function incorrectly. So you also need to override `clear()` function so that it can function correctly.

- Write a tester program (driver) for each of the three different classes for the ADT `FrontList` by following the programs that test `SortedList` class on D2L.

What to Hand In

- Submit all source programs to your class account in **GitHub** system and test them.
- Submit the following documents to the drop box **Project7** on D2:
  - all source programs
  - a script file for test runs on **GitHub**
  - a word file that contains a user document for this project. No other documents are needed for this project.

Grading

Requirements	**points**
Comments in the program	5
Program correctness for HasA	20
Program correctness for IsA	20
Program correctness for AsA	20
Script file from several test runs on GitHub (10 points for each case)	30
User document	5
TOTAL POINTS	**100**