CS 311 - HW 6 - 100 points

Inherited Lined list

We will use the **Linked List** from HW5 and expand it into a Searchable Linked List by adding more functions and inheritance.

Steps:

1. One of the programs at HW5demos folder is used for HW6: equalstest.cpp
   Try to compile the file using g++ equalstest.cpp rlist.o
   You will see that it does not compile. The reason is operator== has not been defined in C++. In HW6, you will implement it, so that when you compile HW6client.cpp, it will work.
2. You will use your completed llist class from HW5 in HW6.
3. Refer to Lecture 11 part 2 and Lecture 12 for instructions of implementation.
4. Implement the given slist class: complete the attached **slist.h** and also create and complete your **slist.cpp. Update** header file **llist.h (from HW5).** Inherit all member functions from the llist class (from HW5). Note that the client of Searchable List will need to be able to call these functions. Pick the access type (public, protected or private) properly.
5. Searchable List class (slist) must have access to the data members of the base class (llist). But the client program should not access to these data members. Thus, data members should be in which part of the Linked List header file? Think.
6. Now, put only the following 4 functions in slist.h and slist.cpp:
   - **slist();**
     constructor which does nothing (note: we are not creating a destructor)
   - **int search(el_t Key)**
     search through the list to see if any node contains Key.
     If so, return its position (>=1). Otherwise, return 0.
   - **void replace(el_t Elem, int I)**
     go to the Ith node (I is between 1 and Count) and replace the element there with Elem.
     If I was an illegal value, throw an exception (OutOfRange).
   - **bool operator==(cost slist&)**
     compare two slist object to see if they look the same.
7. Complete the attached client program **HW6client.cpp**
8. Test your **HW6client.cpp** program using the following **Test1** and enter the results to your **test1.txt**. These test cases are also implemented in HW6client.cpp

Test1:

Case 1: Search and Replace

1. add to front once (element 4)
2. add to rear 3 times (elements 6,7,8)
3. displayAll - 4 6 7 8
4. search for 6, report the result - found in pos 2
5. replace the 6 with 0 using the search result
6. displayAll - 4 0 7 8
7. search for 8 and report the result - found in pos 4
8. replace the 8 with 2 using the search result
9. displayAll - 4 0 7 2
10. search for 5 and report the result - not found
11. replace position 7 with 10 - error

Case 2: Comparison

1. L1 is empty and L2 is empty - true
2. L1 is empty and L2 has 4, 5 - false
3. L1 has 4, 5 and L2 is empty - false
4. L1 has 1,2,3 and L2 has 1,2,3 - true
5. L1 has 1,2,3 and L2 has 1,2 - false
6. L1 has 1,2,3 and L2 has 1,2,3,4 - false
7. L1 has 1,2,3 and L2 has 1,2,4 - false

Submission

Submit a zip file containing the following files.

1. llist.h (5 points) -- llist class header file (update it)
2. llist.cpp (required) -- llist class implementation from last HW
3. slist.h (5 points) -- slist class header file
4. slist.cpp (60 points) -- class implementation file
5. HW5client.cpp (25 points) -- the implemented application
6. test1.txt (5 points) -- results of Test1

Important note1: You will miss up to 10 points if you don't comment your programs.

Important Note2: Always make sure the files you submit can be compiled on **empress.csusm.edu** with no error. We will compile and test your files on empress.