**CS311 Yoshii HW3 Part 3 - Linked List Extended (based on Week 7)**

**===================================================================**

**DUE: Week 9 Tuesday at the beginning of the class**

**TOTAL: 30 pts Your score is:**

**\*Your Name: Thomas Griffin**

**\*Date Turned In: 10/27/15**

**\*Did you do the extra credit?[5pts] <Yes>**

**---------------------------------------------------------------**

**Purpose: To be able to implement and use an overloaded operator and a copy constructor to complete the linked list class.**

**---------------------------------------------------------------**

**PROGRAMMING: Finalizing Linked List [30pts]**

**===================================**

Finally, add the following to your Searchable List (slist) class from HW3P2.

Pseudo code was already provided for you.

**overload operator = (see Week7 Notes-7A notes)**

**create a copy constructor (see Week7 Notes-7A notes)**

**Client:**

**Your client file will have two functions: Main() and CopyTest**

**Main will**

1. Create a 5 element list with 1,2,3,4,5 (L1)
2. Pass the list to a **client function** called CopyTest to

test your copy constructor.

* 1. Copytest will receive the list **passed by value** from main() and
  2. Simply 1) add a node to its rear with 6 in it (should not affect the original)

2) display it (6 elements 1,2,3,4,5,6)

1. Display L1 (this should still be a 5 element list)
2. Do L1 = L1;
3. Display L1 (this should still be 1 2 3 4 5)
4. Create a 4 element list L2 with 7,8,9,10.
5. Display L2
6. Do L2 = L1; (L2 becomes 5 elements 1,2,3,4,5)
7. Display L2.
8. Remove a rear node from L1. (This should not affect L2).
9. Display L1. (L1 is 1,2,3,4)
10. Display L1 again. (4 elements)
11. Display L2 again. (still 5 elements 1,2,3,4,5)

**It is very important to complete this linked list class and make sure the = and copy constructor work perfectly. You will use this to implement HW6 and HW7 graphs.**

**Q) State of the program [2pts] <answer here>**

* **Does your program compile without errors?**

**Yes**

* **List any bugs you are aware of, or state “No bugs”:**

**No bugs**

**Submit these files:**

1. **this assignment sheet with your answers**
2. **slist.h**
3. **slist.C**
4. **the client file**
5. **test results (one run)**

**Whether working or not, test result must include the lines for compiling your files or we will not grade your program i.e. 0 points for the program.**

**Did you check your comments and style against CS311 How To Comment.doc??**

**EXTRA credit [5pts] – highly recommended to do**

**Overload operator == by adding it to the above slist.h and slist.C**

**Must work perfectly to receive any points. No partial credit. So, if it does not work perfectly, do not submit it.**

**Create a separate client file hw3ec.cpp to test ==.**

**Test cases – right before displaying the result of each comparison, must display the following description of the case:**

* 1. **L1 is empty and L2 is empty**
  2. **L1 is empty and L2 has 2 elements**
  3. **L1 has 2 elements and L2 is empty**
  4. **L1 has 1,2,3 and L2 has 1,2,3**
  5. **L1 has 1,2,3 and L2 has 1,2**
  6. **L1 has 1,2,3 and L2 has 1,2,3,4**
  7. **L1 has 1,2,3 and L2 has 1,2,4**

**Submit these additional files:**

1. **Client for extra credit – hw3ec.cpp (you can have the == in the original header and implementation)**
2. **Test results for extra credit - TestEC**