CS 132 - Intro to Computer Science II - Spring 2005
Week #5 Lab Exercise

Week #5 Lab Exercise due: Wednesday, February 16th, END of lab

Week #5 Lab Exercise

INDIVIDUAL exercise:

* On the public course web page, you will find copies of two files, **bag.h** and **bag.cpp**. These implement the **bag** ADT discussed in lecture and in the course text's Chapter 4 (although remember that the version discussed in class, and implemented here, has some different method names from the version discussed in Savitch and Main Ch. 4).

  * create copies of these two files **bag.h** and **bag.cpp** in a new directory on your cs-server account.

* To what type does **bag.h** currently have **value_type** set?

Change this type to **char**.

Then, write a "baby" **tryBag** program, in a file **tryBag.cpp**, that tries out this bag class for elements of that value_type.

The following opening comment block will suffice, for this little less-than-a-tester:

```
// File: tryBag.cpp
// Name:
// last modified:
//
// Purpose: try the bag class out a little
```

Program **tryBag** should:

1. create at least two bag instances, appropriately trying out both constructors;
2. insert at least 5 items into at least one of the bag instances, with at least two items having at least two copies;
3. print to the screen a message nicely displaying each bag instance's size (both expected and actual values), using the appropriate accessor to obtain the bags' sizes;
4. print to the screen a message nicely displaying each bag instance's capacity (both expected and actual values), using the appropriate accessor to obtain the bags' capacities;
5. remove one copy of one of the elements that has 2 copies in the bag; remove all copies of another (different) element that has at least 2 copies in the bag.
6. print to the screen a message nicely displaying the current count of the element that had one copy removed in step #5 (both expected and actual values), and displaying the current count of the element that had all copies removed in step #5 (both expected and actual values), using the appropriate accessor to obtain these quantities for these elements.
7. change the capacity of either bag instance to a different capacity of your choice;
8. print to the screen a message nicely displaying the capacity of the bag involved in step #7 (both expected and actual values), using the appropriate accessor to obtain the bag's capacities.

Run, test, and debug **tryBag.cpp** until you are satisfied with it. Then put your name on the Next: list on the board, so that your program can be checked. Note that I will expect to check and run your program from your cs-server account.

To receive credit for this lab exercise, the above must be completed by the end of the lab period.