CSC 2431 Assignment 5

Assigned: 5/26/16

Due: 6/7/16 at the start of the final, no late turn-in allowed

No textbook exercises are assigned, but they are good practice for the final exam!

Part 1 – Programming Assignment

Derive a class petInventoryTree from bSearchTreeType to store information for pets in a pet store. The petInventoryTree class must have the following methods, which are essentially wrappers to the bSearchTreeType methods so clients can query using names (strings) instead of petType objects:

```
printPets
-Print all the pets in the inventory, and the number of pets
-This method is a wrapper for bSearchTreeType inorderTraversal

findPet
-Find a pet in the inventory by name
-Return true if pet is in the inventory, false otherwise
-This method is a wrapper for bSearchTreeType search

addPet
-Add a pet to the inventory (if not present) by name
-This method is a wrapper for bSearchTreeType insert

removePet
-Remove a pet from the inventory (if present) by name
-This method is a wrapper for bSearchTreeType deleteNode
```

And a single private member (an unsigned int) with an appropriate accessor to keep track of the number of pets currently in the inventory. Create a default constructor that sets this member, and update it as needed in the class methods.

Some notes and rules:

- This class is *not templated,* the tree will always store petType objects.
- You are not allowed to modify the bSearchTreeType or binaryTreeType classes, or override or overload any of their methods.
    - o With these rules, you will have to ensure that all required operations on a petType in the base classes are implemented via operator overloading.
- All methods in bSearchTreeType and binaryTreeType must be inaccessible to clients that have created petInventoryTree objects.

Testing/Client

Create a simple interface in your client with menu options to allow the user to add, remove, find, and list pets by name. An example listing for a run of my program is copied at the end of this document. Your client should loop, asking for menu choices until the user chooses to quit.

What to Turn In

1. All header (.h) and source (.cpp) files **including the unmodified parent classes.**

2. Your output after running with the tests in the example below.

   a. You should thoroughly test your code to ensure it works for all other tests cases.

3. All documents should be uploaded to Blackboard following the assignment submission instructions (linked on Blackboard).

Extra credit:

1. Turn in your solution following the requirements and rules described above.
2. Extend the code you wrote so that it also works with `dogType` objects as defined in the `dogType` class earlier in the quarter. Recall dogs have a `breed` private member as well as a `name` inherited from `petType`. Make sure all operations work on dogs **and** pets. You may modify any of the base classes for this extra credit portion. Turn in a *separate, modified version* of your original solution with these new features.
3. This extra credit is worth the same # of points as the one in Homework 4.

Example run of Dr. Dingler’s code:

```
1. Add pet to inventory
2. Remove pet from inventory
3. Search for pet in inventory
4. Print list of pets
5. Quit
Enter menu choice: 1
Enter pet name to be added: Rocky

1. Add pet to inventory
2. Remove pet from inventory
3. Search for pet in inventory
4. Print list of pets
5. Quit
Enter menu choice: 1
Enter pet name to be added: Otis

1. Add pet to inventory
2. Remove pet from inventory
3. Search for pet in inventory
4. Print list of pets
5. Quit
Enter menu choice: 4
Here is a list of the 2 pet(s) in the database: Otis Rocky

1. Add pet to inventory
2. Remove pet from inventory
3. Search for pet in inventory
4. Print list of pets
5. Quit
Enter menu choice: 2
Enter pet name to be removed: Otis

1. Add pet to inventory
2. Remove pet from inventory
3. Search for pet in inventory
4. Print list of pets
5. Quit
Enter menu choice: 4
Here is a list of the 1 pet(s) in the database: Rocky

1. Add pet to inventory
2. Remove pet from inventory
3. Search for pet in inventory
4. Print list of pets
5. Quit
Enter menu choice: 3
Enter pet name to search for: Otis
Otis is not in our database!

1. Add pet to inventory
2. Remove pet from inventory
3. Search for pet in inventory
4. Print list of pets
5. Quit
Enter menu choice: 5
Goodbye!
Press any key to continue . . .
```