Laboratory Session Worksheet. Week 4, Semester 2, 2007

Individual Exercises.

1. Take as a starting point a copy of your Beast class from the earlier laboratory session.

2. Remove from class Beast the field for the number of wings. Be sure to remove any access to this data member from within the constructor and the ListAttributes() methods.

3. Change the data member scariness to a private data member bodyWeight.

4. Write a function setBodyWeight() that checks that its only parameter is a valid weight and sets the Beast’s bodyWeight to that value. Write a function getBodyWeight() that returns the current value of the Beast’s bodyWeight.

5. Write a function getTotalWeight() that returns the total weight of the Beast calculated using the following formula:

   totalWeight = bodyWeight + numberOfLegs

6. Change the function Frightens() to compare the total weight of the Beasts. The heavier Beast always frightens the lighter Beast.

7. Derive two new classes from class Beast: class WingedBeast and class FinnedBeast.

8. Add to class WingedBeast a data member for the wing-span and another for its number of wings. The number of wings must always be an even number. How will you ensure this is the case?

9. Write constructors for the two new classes that call the constructors of their parent classes (super classes) as well as initialising all of the fields unique to themselves.

10. Overload the class Beast’s getTotalWeight() function and calculate the total weight of a WingedBeast using the formula:

    totalWeight = bodyWeight + numberOfWings * (wingSpan / 2.0) + numberOfLegs

11. Add to class FinnedBeast a data member for the number of fins.

12. Overload the class Beast’s getTotalWeight() function and calculate the total weight of a FinnedBeast using the formula:

    totalWeight = bodyWeight + numberOfFins + numberOfLegs

13. Overload the Winged and Finned Beasts’ ListAttributes() functions. The derived classes ListAttributes() function should only print out the extra features of these two Beast types and then call the base class Beast’s ListAttributes() function to list the remaining features.

14. Store your Zoo of Beasts in a generic container type of your choice. Allow the user of the software to add and remove Beasts from the Zoo.