HW6 – Polymorphism

CSS 501 – Data Structures & Object-Oriented Programming
By: Hansel Ong

Summary

This homework assignment focuses on inheritance and polymorphism

Skills Expected

- All previous skills up to now
- Abstract classes
- Inheritance/Polymorphism
- Friend keyword

Assignment Description

Design and create an abstract "Parent" class, two non-abstract "Child" classes (each inherits from the Parent), and one non-abstract "Grandchild" class. In other words, attempt the "Hybrid" (or "Diamond") inheritance scheme[1].

For example, you could have a "Creature" abstract class that are inherited by "Robot" and "Bear" classes and subsequently a "RoboBear[2]" class.

[1] https://www.tutorialcup.com/cplusplus/inheritance.htm#hybrid-inheritance
[2] http://www.robobear.org/

Grading Criteria

- One abstract "Parent" object (e.g. "Creature")
  - **[1 Point]** All instance variables private
  - **[1 Point]** Getters/setters as appropriate (enforce invariants)
  - **[1 Point]** At least one overloaded constructor (call setter to enforce invariants)
  - **[1 Point]** At least one "pure virtual" function
  - **[1 Point]** At least one virtual function that is implemented in "Parent"
- Two separate non-abstract "Child" objects (e.g. "Robot" and "Bear")
  - **[1 Point]** All instance variables private
  - **[1 Point]** Getters/setters as appropriate (enforce invariants)
  - **[2 Point]** Properly pass data to parent constructor(s)
  - **[2 Point]** Make one Child a "friend class" of the other Child and have a function that makes use of the friend class' private instance variables
  - **[2 Point]** At least one Child must have at least one "friend function"
  - **[1 Point]** Proper use of override
- One non-abstract "Grandchild" Object (e.g. "Robobear")
  - **[1 Point]** All instance variables private
  - **[1 Point]** Getters/setters as appropriate (enforce invariants)
  - **[2 Point]** Properly pass data to parent constructor(s)
  - **[2 Point]** Override at least one function
  - **[2 Point]** Have at least one function that calls a specific parent's function
- Driver
  - **[1 Point]** Functional Positive Testing: Create an instance of each Child and Grandchild objects given valid values (passes invariant check)
  - **[1 Point]** Functional Negative Testing: Attempt to create an instance of each Child and Grandchild objects given invalid values (does not pass invariant check – what should it do?)
  - **[6 Points]** Driver should call each public method (including any public getter/setter)