Victoria University of Wellington
School of Engineering and Computer Science

SWEN221: Software Development

Lab Handout 3 (worth ≈1.3% of overall mark)

Outline

The purpose of this lab is to get more experience with inheritance and polymorphism. Firstly, you will complete some simple exercises which test your understanding of polymorphism. Secondly, you will use polymorphism in a more realistic setting, namely that of a text-based adventure game.

By the end of the lab, the tutor should have given you a grade based on your performance. **Be sure that you have been given a grade before you leave, since attendance at labs is mandatory**.

NOTE: it is also recommended that you submit your final lab code via the online submission system, which can be found on the SWEN221 homepage. This is useful in case your lab grade is lost, or you believe you have been given the incorrect grade.

Activity 1: Polymorphism Quiz (30 minutes)

This is a short, fun activity designed to get you thinking about polymorphism. The aim of the activity is simply to answer as many questions correctly as possible! You should begin by downloading the polyquiz.jar file from the lecture schedule on the course homepage.

From the command-line (or from within Eclipse), simply run the program as follows:

```
java -jar polyquiz.jar
```

Figure 1 (overleaf) illustrates the user interface presented by the polyquiz program. The program puts up a simple test class, and asks you what it will print. The program will then continue until all questions are answered.

The aim is simple: answer as many questions as possible correctly! If you are unsure about why you got a particular question wrong, ask the tutor. You should find that there are many easy questions in the quiz. However, some are definitely harder than others! If you believe the quiz has got one wrong, try running the test code in question for yourself to see what it does.

When you have completed activity 1, move on to activity 2.

```
========================================
package swen221.polyquiz;

public class Test4 {
   static class Cat {
      public String name;
      public Cat(String name) { this.name = name; }

      public void fight(Cat target) {
         System.out.println(name + " claws " + target.name);
         target.fightBack(this);
      }

      public void fightBack(Cat target) {
         System.out.println(name + " claws back " + target.name);
      }
   }

   static class RoughCat extends Cat {
      public RoughCat(String name) { super(name); }
      public void fight(Cat target) {
         System.out.println(name + " bites " + target.name);
         target.fightBack(this);
      }
   }

   public static void main(String[] args) {
      Cat jim = new Cat("Jim");
      Cat bob = new RoughCat("Bob");
      bob.fight(jim);
   }
}
========================================
1) "Bob bites Jim\n Jim claws back Bob"
2) "Bob claws Jim\n Jim claws back Bob"
3) "Bob bites Jim\n Jim bites back Bob"

What is your answer?
```

Figure 1: Illustrating the interface for the polyquiz game

Activity 2: Extending a simple game (60 minutes)

The aim of this activity is to extend a simple text-adventure game using inheritance and polymorphism. The source code for the game provides a more realistic illustration of inheritance and polymorphism than seen in lectures.

The text adventure game provides a simple, text-based adventure world in which you can move around and interact with objects. You should begin by spending some time playing the game and getting a feel for it.

To run the program, first download the `textadventure.jar` from the lecture schedule on the course homepage. Then, run the game from the command line as follows (or by loading it into Eclipse):

```
java -jar textadventure.jar
```

You should see a simple interface pop-up, asking you some questions and then letting you play the game (see Figure 2 for an illustration). Using various keys, you can navigate the haunted house, interacting with objects and trying to solve clues.

The aim of this activity is to extend the game with more kinds of object. Specifically, you should extend the game with at least the following objects:

1. **A Key**. This item can be picked up, but obviously not walked through and prodding does nothing.
2. **A fake door**. This item looks like a door. But, trying to walk through it results in bumping your head!
3. **A Book**. This item can be picked up, but obviously not walked through. Prodding it reveals some interesting clues (if you have enough smarts).
4. **A special bookcase**. This item is just like a book case, except that prodding it reveals a book which you can then pick up!

In implementing these extensions, you should endeavour to use polymorphism and inheritance wherever possible.

CHALLENGE: If you feel like a challenge, then other interesting objects you might try and implement are:

1. **A locked door**. This door can only be passed through if the player is holding the key.
2. **Heavy items**. These can only be picked up by sufficiently strong players.
3. **Magic Items**. These can be used by the wizard to do useful things.

```
Welcome to the Haunted House Adventure!!!!!
-------------------------------------------
Written by David J. Pearce (in a rush)


What is your name?
Dave
Who do you want to be?

1) The strong warrior
2) The wiley wizard
3) The average archer
Enter 1-3:
3
================================
Dave, you are in The Hallway.  You see:

A dimly lit room, littered with strange paintings and grotesque statues.

You can see the following things in The Hallway:
        1) North Exit.  It connects The Hallway to The Kitchen
        2) East Exit.  It connects The Hallway to The Sitting Room
        3) South Exit.  It connects The Hallway to The Study

Do you want to go (E)ast, (W)est, (N)orth, (S)outh, (P)ick up something,
 P(R)od something or (L)ook at something? (E/W/N/S/P/R/L):
```

Figure 2: Illustrating a simple session of the text adventure game

Marking Guide

Each lab is worth just under 1% of your overall mark for SWEN221. The lab should be marked during the lab sessions, according to the following grade scale:

- **0**: Student didn't attend lab.
- **E**: Student did not really participate in the lab.
- **D**: Student's participation was *poor*. For example, he/she made some attempt to work on the lab, but did not complete any activities.
- **C**: Student's participation was *satisfactory*. That is, he/she completed at least one activity (e.g. Polymorphism Quiz).
- **B**: Student's participation was *good*. That is, he/she has made reasonable progress on activity 2.
- **A**: Student's participation was *excellent*. That is, he/she completed all activities. Note, completing the *challenge* is not required.