Programming Project #5
CpSc 8700: OO Software Development
Computer Science Department
Clemson University
(1) Object Pooling, (2) Collisions,
(3) Explosions, and (4) Projectiles
Brian Malloy, PhD
November 24, 2015

Due Date:

In order to receive credit for this assignment your solution must be submitted, using `web handin`, by 8 AM, Monday, November 23rd, 2015. You may receive 90% of the grade if you submit the project within one week of the deadline.

Project Specifications:

The goal of this project is to incorporate more action and interaction into your animation so that it includes projectiles, collisions, and explosions. Also, you should begin to develop a consistent theme that threads your animation, and to begin to enable the player and your game to reach a conclusion. When considering your project goals, keep in mind that video games are typically built by large teams of programmers, artists, and many others, and that there are only about *two weeks remaining in the semester*. With this in mind, your goal should be to keep the scope reasonable and produce a robust game that reaches a conclusion.

For this project, you are not required to draw your own sprites; also, you **may not** use any sprites that I have provided. however, if you use sprites that you did not build yourself, the sprites must be made available under the open source license and you must specify, in your **ascii** README, the site where you got the sprites. Also, your game must be data driven so that you read game constants from an XML file.

Your goal should be to establish a consistent theme so that your game begins to tell a story. The requirements listed below should guide this fifth project; however, if one or more of the requirements is inconsistent with the theme that you are trying to establish, stop by during office hours or send an email with times that we might meet to discuss your game idea and to negotiate a trade for the inconsistent requirement. However, one requirement that is not negotiable is that your fifth project must include object pooling and your information HUD should demonstrate this use.

For example, Figure **??** illustrates, in the upper right corner, the instructor's use of an object pool for bullets, with **3** bullets active and **2** bullets in the pool. This figure is intended only for illustration and should not be interpreted to mean that you must have bullets and they must be pooled in your game. The point is that you must implement an object pool and you must demonstrate your use of the pool in your information HUD. You can pool bullets, explosions, or some other dynamically created object. We will discuss object pooling in lecture and there is a lot of internet information about object pooling; you should consult this information if you need additional help in understanding object pooling.

A summary of the requirements for this project include:

1. Demonstrate object pooling and show pool contents in your information HUD that appears at program start, toggles with F1, and shows information about how to move the player, how many objects are

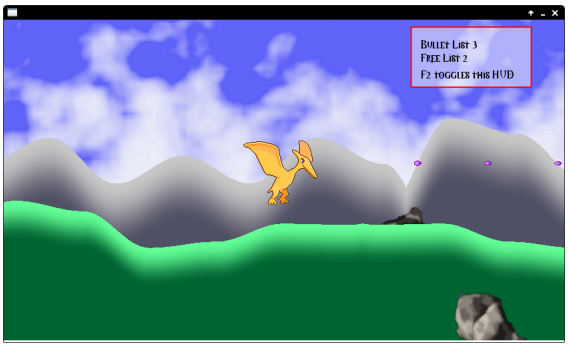

Figure 1: An illustration of object pooling, with three active bullets and one bullet in the "pool"

active and how many objects are pooled, and any other information you wish to display.

2. Your player should explode and, after the explosion completes, should re-appear. Use OO design.
3. Implement a reset function that restarts your game; toggle reset with "r" key.
4. Projectiles: you don't have to shoot bullets, but you need projectiles.
5. Collision detection and NPC explosions: use chunks and/or frames.
6. We should be able to generate frames for your game using the F4 key.
7. Include a well-controlled player object, create depth, and provide an information HUD.

Your assignment will be tested on a Linux platform using gcc or clang, however you should test your project on several different platforms and it should be independent of platform and language implementation. (Key summary: F1 ⇒ help, F4 ⇒ frames, and r ⇒ restart)