CS32 Project3 Report

For GhostRacer project, I designed the inheritance hierarchy of class Actor as following:

GraphObject

Actor

BorderLine

HolyWaterProjectile

Agent

GhostRacer

Pedestrian

HumanPedestrian

ZombiePedestrian

ZombieCab

GhostRacerActivatedObject

OilSlick

HealingGoodie

HolyWaterGoodie

SoulGoodie

In Actor class, I first define getter and setter functions for alive, horizontal speed, vertical speed, and collidance avoidance worthy. I define a pure virtual version of doSomething() since all children class of Actor needs to define doSomething() in a tick. I define a virtual version of getHit() since different actor reacts differently when hit by holy water projectile. I define move(), isOverlap() and isInScreen() as helper functions for Actor children classes to use in doSomething(). I define isOilSlick() to check if this is an OilSlick for StudentWorld to avoid spawning goodies on oil slick.

In Agent class, I added getter and setter functions for HP and movement plan in addition to parent class Actor.

In GhostRacer class, I define getter and setter functions for holy water. I define getHit(), getHealed() and spin() for ghost racer to respond to other actor. In doSomething(), I define ghost racer to respond to key input and out of the borders.

For the following children classes, I implement doSomething() and getHit() for virtual functions of Actor class.

In doSomething() of Pedestrain, I give random horizontal speed and set movement plan in each click for children classes of pedestrian to use.

In doSomething() of HumanPedestrain, I set racer died when overlapped with human pedestrian. In getHit() of HumanPedestrain, I set new horizontal speed and direction.

In doSomething() of ZombiePedestrain, I lower racer HP when overlapped and make it grunt and move towards racer when close. In getHit() of ZombiePedestrain, I lower is HP and drop heal goodie when died.

In doSomething() of ZombieCab, I set it to another direction when overlapped (crashed) with racer, and move alone the lane based on other avoidance worthy actors on the lane. In getHit() of ZombieCab, I lower its HP when hit and drop oil slick when died.

In doSomething() of ActivatedObject, I move it in each click for children classes of ActivatedObject to use.

In doSomething() of OilSlick, I make racer spin when overlapped.

In doSomething() of HealGoodie, I heal racer when overlapped. In getHit() of HealGoodie, I set it died when hit.

In doSomething() of HolyWaterGoodie, I add holy water to racer when overlapped. In getHit() of HolyWaterGoodie, I set it died when hit.

In doSomething() of SoulGoodie, I decrease soul2save in StudentWorld when overlapped.

In StudentWorld class, I first define three basic functions: init(), move(), and cleanUp(). Then I define getter and setter functions for soul2Save and bonus, which are used in stats line of the game. I also define getter functions for racer and actors for actors to call on other actors if in need. I define spawn() and dropActor() for adding actors in move() and define getLane(), getCloestActor(), and isOilSlickOverlap() as helper functions for spawn().

In init(), I initializes racer, bonus, and soul2Save. I draw initial borders

In move(), I draw new border lines, spawn new actors, let each actor do something, remove dead actors, and update game status line. I check status of game to see if racer died, level finished, or game over.

In cleanUp(), I remove all actors.