Timothy Pranoto  
38964311  
ICS161 HW7

Design issue: My checking for collision function just doing the check if there are any collisions between 2 related sprites. So, in order to use it in the paddle and ball, I need to add extra code to check which part of the paddle collide which part of the ball, and add the bounce code respectively.

I think it would make sense to create such an event after every collision, because to make an event appear in your program, you need some sort of trigger. I can make that trigger from checking if there are any collisions happen between sprites. I can also make multiple different events for every type of collisions by checking collision.

After searching around the internet, I found that I can use SDL\_intersectRect or SDL\_hasIntersection to check if each SDL\_Rect of sprites intersect with each other. I don’t think there are any difference between my implementation of overlapsWith and SDL functions for intersect because they are both only check for a Boolean value if sprites intersect with each other.

For part 2, I added this function as available function in the GameObject.h:

bool overlapsWith(GameObject\* otherGameObject);

And in GameObject.cpp I specified the definition:

bool GE161::GameObject::overlapsWith(GE161::GameObject\* otherGameObject)

{

if (x\_>=otherGameObject->getX()+otherGameObject->sprite\_->frameWidth\_){

return false;

}

if (y\_ >= otherGameObject->getY() + otherGameObject->sprite\_->frameHeight\_){

return false;

}

if (x\_ + sprite\_->frameWidth\_ <= otherGameObject->getX()){

return false;

}

if (y\_ + sprite\_->frameHeight\_ <= otherGameObject->getY()){

return false;

}

return true;

}

And I changed the collision part in the PongGamePlayScene.cpp to use the new function I added in GameObject:

if (paddle->overlapsWith(ball)){

if (paddle->getX() <= ball->getX()+frameWidth\_)

x\_delta = rand() % 3 + 1;

if (paddle->getY() <= ball->getY()+frameHeight\_)

y\_delta = rand() % 3 + 1;

if (paddle->getY() + paddleH >= ball->getY())

y\_delta = -(rand() % 3 + 1);

}