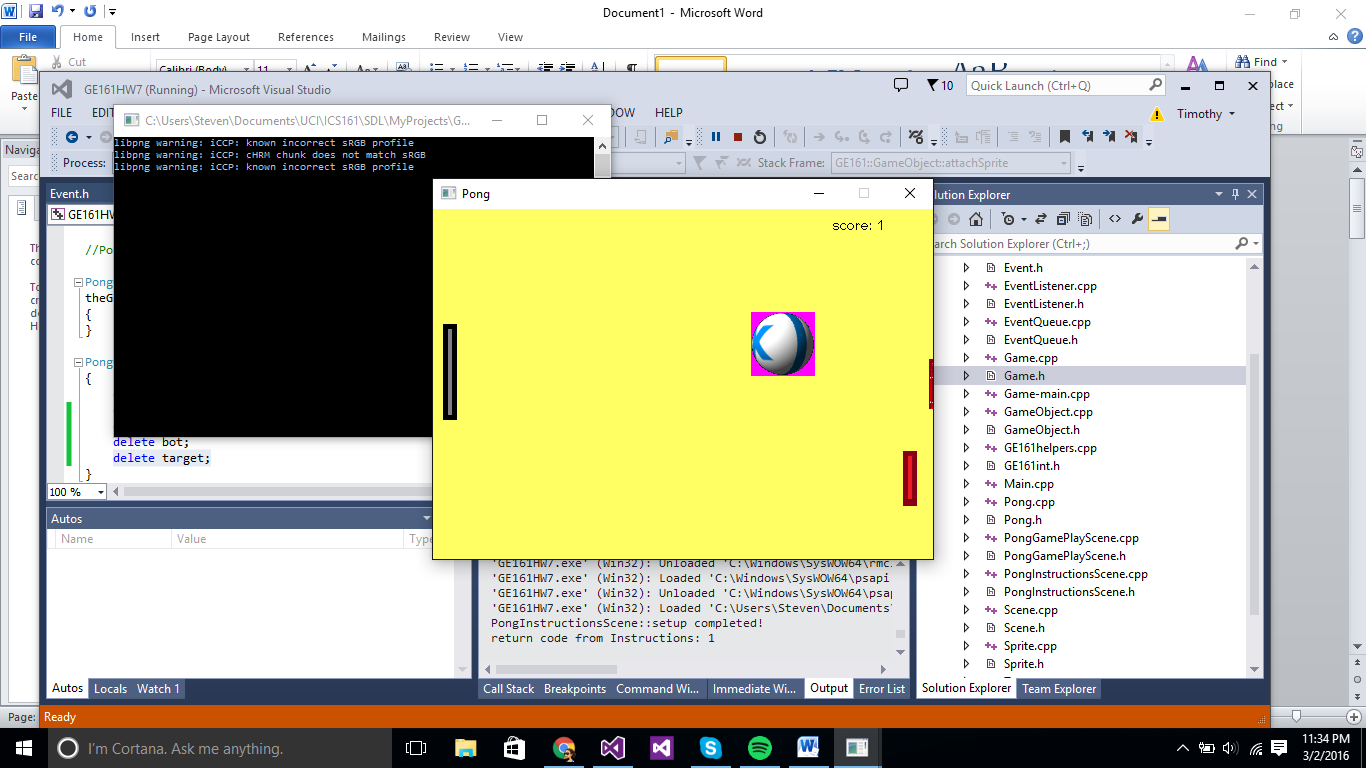
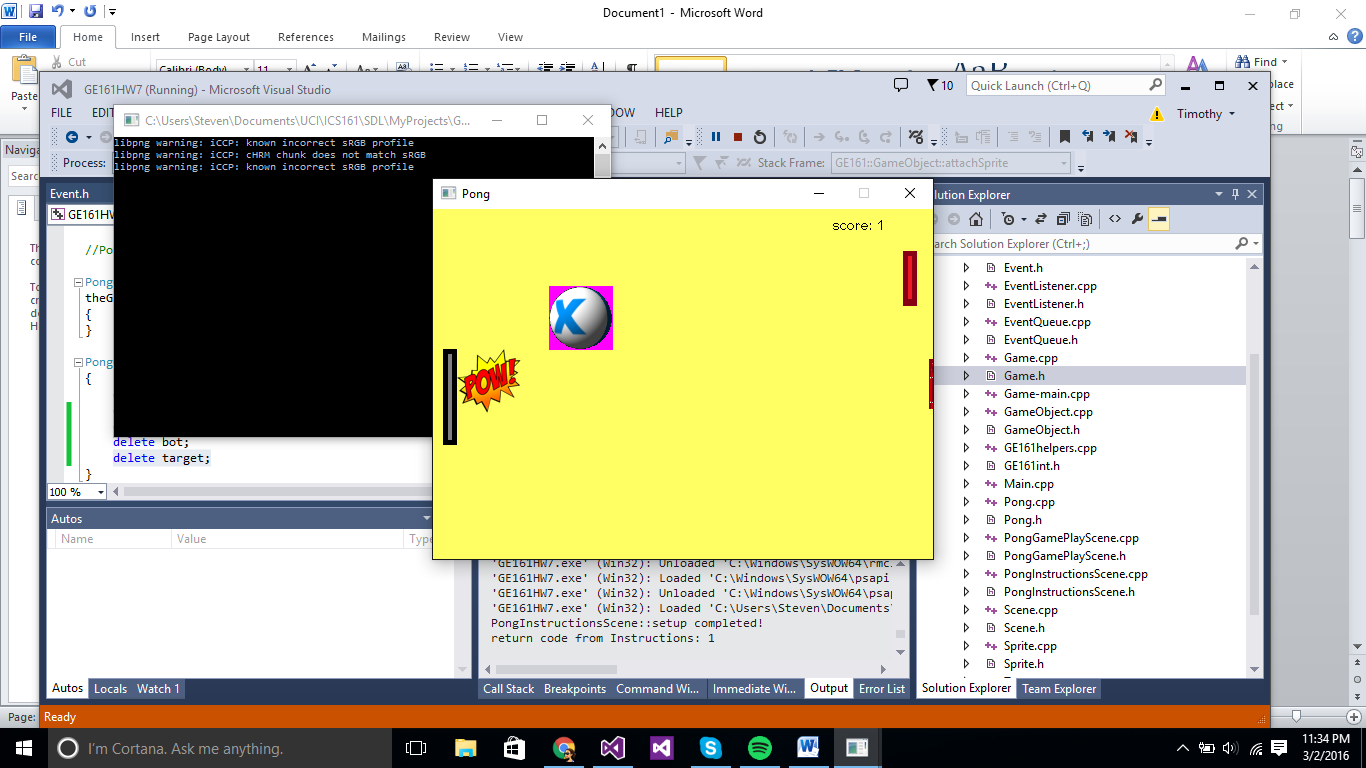
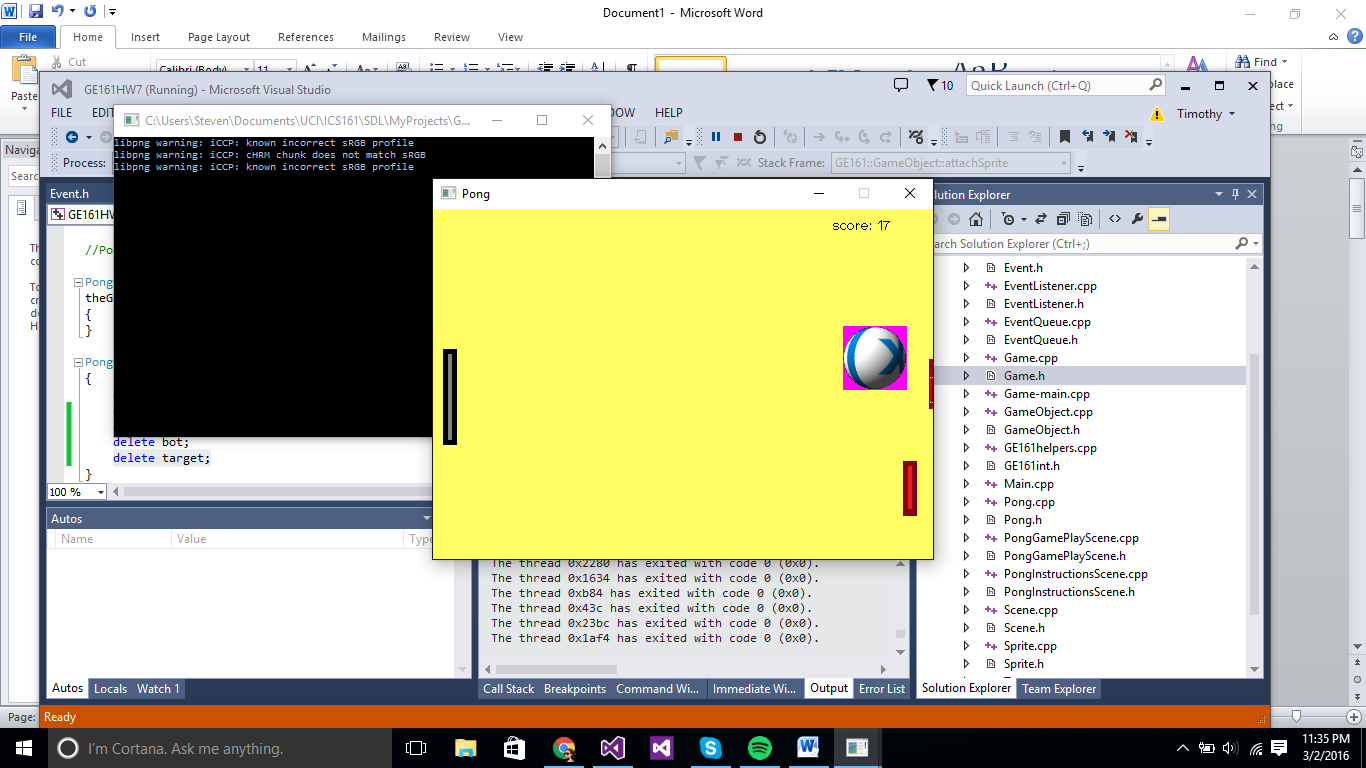
Timothy Pranoto  
38964311  
ICS161 HW8







Part1

I added this function to Game.h

static void sendEvent(Event& e);

and the definition in Game.cpp

void GE161::Game::sendEvent(GE161::Event& e)

{

theGame->eventQueue\_->eventQueue\_.push(&e);

}

Part2

I added an event type in the Event.h

static const int COLLIDE = 0;

In the PongGamePlayScene.h

I added private variables:

GE161::GameObject\* pow;

bool colliding;

int start, end;

int powW;

int powH;

In the PongGamePlayScene.cpp,

In the bool PongGamePlayScene::setup() I added:

pow = new GE161::GameObject(24, 0);

powW = 64;

powH = 64;

GE161::Sprite\* funnyPow = new GE161::Sprite(powW, powH);

int q = funnyPow->makeFrame(GE161::Game::basePath() + "pow.png", 0, 0);  
funnyPow->addFrameToSequence("Pow", q);  
pow->attachSprite(funnyPow);  
colliding = false;

GE161::Game::registerAsListener(GE161::Event::COLLIDE, this);

In the int PongGamePlayScene::draw(),

Inside the if (paddle->overlapsWith(ball)) loop, I added :

GE161::Event\* coll = new GE161::Event();

coll->type = GE161::Event::COLLIDE;

start = GE161::Game::getTicks();

theGame->sendEvent(\*coll);

After the loop, I added:

if (colliding){

pow->draw("Pow");

end = GE161::Game::getTicks();

if (end - start >= 1500)

colliding = false;

}

In the void PongGamePlayScene::onEvent(GE161::Event & e), I added:

if (e.type == GE161::Event::COLLIDE)

colliding = true;

In the PongGamePlayScene::~PongGamePlayScene(), I added:

delete pow;

Part3:

1. I added a bot paddle on the right side of the window, and if the ball hit the bot paddle it will bounce off and player will not receive any score as it would if it hits the right side of the window.

In the Event.h, I added an event type:

static const int BOTHIT = 1;

In the PongGamePlayScene.h, I added private variables:

GE161::GameObject\* bot;  
int botW, botH;  
int ty\_delta;  
bool collBot;

In the PongGamePlayScene.cpp,

In the bool PongGamePlayScene::setup(), I added:

bot = new GE161::GameObject(470, 0);

botW = 14;

botH = 55;

GE161::Sprite\* botSprite = new GE161::Sprite(botW, botH);

int a = botSprite->makeFrame(GE161::Game::basePath() + "botpaddle.png", 0, 0);

botSprite->addFrameToSequence("Bot", a);

bot->attachSprite(botSprite);

collBot = false;

GE161::Game::registerAsListener(GE161::Event::BOTHIT, this);

ty\_delta = 3;

In the int PongGamePlayScene::draw(), I added:

bot->moveY(ty\_delta);

bot->draw("Bot");

if (bot->getY() < 0)

{

ty\_delta = rand() % 3 + 3;

}

if (bot->getY() + botH >= theGame->window()->clientHeight())

{

ty\_delta = -(rand() % 3 + 3);

}

if (bot->overlapsWith(ball)){

GE161::Event\* coll = new GE161::Event();

coll->type = GE161::Event::BOTHIT;

theGame->sendEvent(\*coll);

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

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

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

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

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

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

}

if (collBot){

collBot = false;

}

In the void PongGamePlayScene::onEvent(GE161::Event & e), I added:

if (e.type == GE161::Event::BOTHIT)

collBot = true;

In the PongGamePlayScene::~PongGamePlayScene(), I added:

delete bot;

2. I made a target placed in the middle of the right side of the window, if the ball managed to hit that area, the +10 will be added to the player score.

In the Event.h, I added an event type:

static const int TARGET = 2;

In the PongGamePlayScene.h, I added private variables:

GE161::GameObject\* target;  
int tarW, tarH;  
bool colltar;

In the PongGamePlayScene.cpp,

In the bool PongGamePlayScene::setup(), I added:

tarW = 4;

tarH = 50;

target = new GE161::GameObject(theGame->window()->clientWidth() - tarW, theGame->window()->clientHeight() / 2-(tarH/2));

GE161::Sprite\* targetSprite = new GE161::Sprite(tarW, tarH);

int ct = targetSprite->makeFrame(GE161::Game::basePath() + "target.png", 0, 0);

targetSprite->addFrameToSequence("Target", ct);

target->attachSprite(targetSprite);

colltar = false;

GE161::Game::registerAsListener(GE161::Event::TARGET, this);

In the int PongGamePlayScene::draw(), I added:

target->draw("Target");

if (target->overlapsWith(ball)){

GE161::Event\* coll = new GE161::Event();

coll->type = GE161::Event::TARGET;

theGame->sendEvent(\*coll);

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

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

}

if (colltar){

score += 10;

colltar = false;

}

In the void PongGamePlayScene::onEvent(GE161::Event & e), I added:

if (e.type == GE161::Event::TARGET)

colltar = true;

In the PongGamePlayScene::~PongGamePlayScene(), I added:

delete target;