/\*

The Game Project 4

Midterm

by:Swen Chan

\*/

var gameCha\_x;

var gameCha\_y;

var floorP\_y;

var isL;

var isR;

var isF;

var isP;

var cany;

var isF\_cany;

var collectable;

var trees\_x;

var clouds;

var mountains;

var cameraPosX;

function setup()

{

createCanvas(1024, 576);

floorP\_y = height \* 3/4;

gameCha\_x = width/2;

gameCha\_y = floorP\_y;

isL = false;

isR = false;

isF = false;

isP = false;

isF\_cany = false;

//start

cany = { x\_pos: 200, width: 100 };

collectable = {x\_pos: 140, y\_pos: floorP\_y - 50, size: 50, isFound:false};

//initialise underground variables

trees\_x = [100, 150, 180];

clouds = {x\_pos:[200, 400, 700], y\_pos:[100, 80, 50],

width:[145, 120, 100], height:[45, 30, 20]};

mountains = {x\_pos:[620, 400, 150], height:[250, 200, 130],

width:[100, 80, 130]};

cameraPosX = 0;

}

function draw()

{

//modify camera's position

cameraPosX = gameCha\_x - width / 2;

//do not allow second jump when the character is Plummeting

if (gameCha\_y <= 285){

isP = false;

}

//do not allow move when the player has fallen in the cany

if (isF\_cany == true){

isL = false;

isR = false;

}

//end

///////////DRAWING CODE//////////

background(100,155,255);

noStroke();

fill(0,155,0);

rect(0, floorP\_y, width, height - floorP\_y); //draw some green ground

//start

//save state

push();

translate(-cameraPosX, 0);

//draw the cany

noStroke();

fill(20,20,10);

rect(cany.x\_pos, floorP\_y, cany.width, 144);

//draw trees

noStroke();

fill(150,100,100);

for(i in trees\_x.length){

rect(trees\_x[i], treePos\_y + 58, 40, 92);

}

//draw clouds

fill(240);

for(var i = 0; i < clouds.x\_pos.length; i++) {

ellipse(clouds.x\_pos[i], clouds.y\_pos[i],

clouds.width[i], clouds.height[i]);

ellipse(clouds.x\_pos[i] - 15, clouds.y\_pos[i],

clouds.width[i] / 2, 4 \* clouds.height[i] / 3);

ellipse(clouds.x\_pos[i] + 15,clouds.y\_pos[i],

clouds.width[i] / 2, 4 \* clouds.height[i] / 3);

}

//draw mountains

fill(100,100,0);

for(var i = 0; i < mountains.x\_pos.length; i++){

triangle(mountains.x\_pos[i], floorP\_y - mountains.height[i],

mountains.x\_pos[i] - mountains.width[i] / 2, floorP\_y,

mountains.x\_pos[i] + mountains.width[i] / 2, floorP\_y

);

triangle(mountains.x\_pos[i] - mountains.width[i] / 4, floorP\_y - mountains.height[i] \* 0.75,

mountains.x\_pos[i] - 3 \* mountains.width[i] / 4, floorP\_y,

mountains.x\_pos[i] + mountains.width[i] / 4, floorP\_y);

triangle(mountains.x\_pos[i] + mountains.width[i] / 4, floorP\_y - mountains.height[i] \* 0.75,

mountains.x\_pos[i] + 3 \* mountains.width[i] / 20, floorP\_y,

mountains.x\_pos[i] + 3 \* mountains.width[i] / 4, floorP\_y);

}

//if the character on the cany,fall into

if(isF\_cany){

gameCha\_y += 10;

}

//decide the isF variable

if(gameCha\_y < floorP\_y){

isF = true;

}

//change the 'isFound' variable if the collectable is found

if(dist(gameCha\_x, gameCha\_y, collectable.x\_pos, collectable.y\_pos + 50) < collectable.size ){

collectable.isFound = true;

}

//draw the collectable

if(!collectable.isFound){

noStroke();

fill(255,215,0);

rect(collectable.x\_pos, collectable.y\_pos + collectable.size / 4,

collectable.size / 2, collectable.size / 2);

rect(collectable.x\_pos - collectable.size / 4, collectable.y\_pos + collectable.size / 2,

collectable.size, collectable.size / 2)

}

//the game character

if(isL && isF)

{

// add your jumping-left code

fill(100,0,200);//head

ellipse(gameCha\_x - 11, gameCha\_y - 60, 24, 24);

fill(200,0,0);//body

rect(gameCha\_x - 20, gameCha\_y - 48, 18, 20);

fill(0);//right foot

rect(gameCha\_x - 4, gameCha\_y - 28, 10, 10);

rect(gameCha\_x - 22, gameCha\_y - 28, 10, 10);//left foot

rect(gameCha\_x - 2, gameCha\_y - 40, 10, 3);//arm

}

else if(isR && isF)

{

// add your jumping-right code

fill(100,0,200);//head

ellipse(gameCha\_x + 11, gameCha\_y - 60, 24, 24);

fill(200,0,0);//body

rect(gameCha\_x + 2, gameCha\_y - 48, 18, 20);

fill(0);//right foot

rect(gameCha\_x + 12, gameCha\_y - 28, 10, 10);

rect(gameCha\_x - 6, gameCha\_y - 28, 10, 10);//left foot

rect(gameCha\_x - 8, gameCha\_y - 43, 10, 3);

}

else if(isL)

{

// add your walking left code

fill(100,0,200);//head

ellipse(gameCha\_x - 11, gameCha\_y - 50, 24, 24);

fill(200,0,0);//body

rect(gameCha\_x - 20, gameCha\_y - 38, 18, 30);

fill(0);//right foot

rect(gameCha\_x - 4, gameCha\_y - 8, 10, 10);

rect(gameCha\_x - 22, gameCha\_y - 8, 10, 10);//left foot

}

else if(isR)

{

// add your walking right code

fill(100,0,200);//head

ellipse(gameCha\_x + 11, gameCha\_y - 50, 24, 24);

fill(200,0,0);//body

rect(gameCha\_x + 2, gameCha\_y - 38, 18, 30);

fill(0);//right foot

rect(gameCha\_x + 12, gameCha\_y - 8, 10, 10);

rect(gameCha\_x - 6, gameCha\_y - 8, 10, 10);//left foot

}

else if(isF || isP)

{

// add your jumping facing forwards code

fill(100,0,200);//head

ellipse(gameCha\_x, gameCha\_y - 60, 24, 24);

fill(200,0,0);//body

rect(gameCha\_x - 9, gameCha\_y - 48, 18, 20);

fill(0);//right foot

rect(gameCha\_x + 4, gameCha\_y - 28, 10, 10);

rect(gameCha\_x - 14, gameCha\_y - 28, 10, 10);//left foot

rect(gameCha\_x + 9, gameCha\_y - 40, 10, 3);//right arm

rect(gameCha\_x - 19, gameCha\_y - 40, 10, 3);//left arm

}

else

{

// add your standing front facing code

fill(100,0,200);//head

ellipse(gameCha\_x, gameCha\_y - 50, 24, 24);

fill(200,0,0);//body

rect(gameCha\_x - 9, gameCha\_y - 38, 18, 30);

fill(0);//right foot

rect(gameCha\_x + 4, gameCha\_y - 8, 10, 10);

rect(gameCha\_x - 14, gameCha\_y - 8, 10, 10);//left foot

}

///////////INTERACTION CODE//////////

//Put conditional statements to move the game character below here

if(isL){

gameCha\_x -= 5;

}

if(isR){

gameCha\_x += 5;

}

if(isP){

gameCha\_y -= 25;

if(gameCha\_y < 280){

gameCha\_y = 280;

}

}

if(isF){

gameCha\_y += 5;

if(gameCha\_y >= floorP\_y){

gameCha\_y = floorP\_y;

isF = false;

}

}

if(gameCha\_x < cany.x\_pos + cany.width - 10

&& gameCha\_x > cany.x\_pos + 10

&& gameCha\_y == floorP\_y){

isF\_cany = true;

}

//restore state

pop();

}

function keyPressed()

{

// if statements to control the animation of the character when

// keys are pressed.

if(!isF\_cany){

if(key == 'a'){

isL = true;

}

if(key == 'd'){

isR = true;

}

if(key == 'w'){

if(isP){

isF = true;

}

if(!isF){

isP = true;

}

}

}

}

function keyReleased()

{

// if statements to control the animation of the character when

// keys are released.

if(key == 'a'){

isL = false;

}

if(key == 'd'){

isR = false;

}

if(key == 'w'){

isP = false;

}

}

//end