# “Paddleball hit”

## A program to assist with your entertainment.

**Elisha Olabode**
**20753895**

CS171 Computer Systems I

B.Sc.

Computer Science and Software Engineering

Department of Computer Science

Maynooth University

Maynooth, Co. Kildare

Ireland

## Declaration

I hereby certify that this material, which I now submit for assessment as part of CS171 Computer Systems module, is entirely my own work and has not been taken from the work of others - save and to the extent that such work has been cited and acknowledged within the text of my work.

Signed: Elisha Olabode Date: 18/12/2020

## Acknowledgements

I would like to thank all my lectures on all their effort towards my project it hasn't been easy getting through to them because of the pandemic but they have contributed their time an effort into helping me. Also want to appreciate the demonstrator for always replying to my issues and problem I wouldn't have gotten anywhere if it hadn't been for them much love to all of them.

## Abstract

This report describes the development of a program that's for entertainment in a form of game sense to help you feel entertained while your board at home. There is colour graphic inverted into the paddleball hit with various of different colour and two different paddles to initiate the different players. My first thought of idea was a basketball game but noticed through the process that it was too complicated and needed more time to make but my other idea which was on my mind is a game like pong because it wasn't too complicated.

## 1 Introduction

While sitting at home thinking of what games you can play that isn't to complicate or hard and still fun then paddleball hit is the right game for you because this is not to complicated it's a game you can enjoy with your friends and family. The aim of the work is to help kids and others to see that life isn't too much of a rush by taking your time in life you're most likely to reach your goal and if you are down you could also start over and also to full you with entertainment with enjoyment while playing the game.

## 2 Research

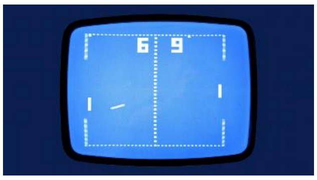
There are lots of website that makes games like paddleball hit some of this relate to the pong games played in this generation. There are lots of ways of making a game like pong and requires focus and skills set to accomplish a successful game. Most of the research focusses on getting the sprite associated with paddle and ball.

Creating the Paddle. To create a paddle, create a sprite that is a horizontal line.

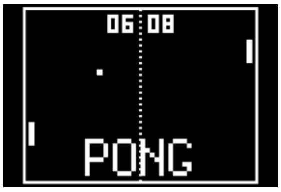
Creating the Ball. Create a script that is a small circle to represent the ball.

Pong was invented as an arcade video game in a small bar in Sunnyvale, California. This was such a hit that the machine broke down – not because there was something wrong with the programming, but because it was stuffed with quarters from the bar's patrons. Pong game is a simple game but full of complicated history.

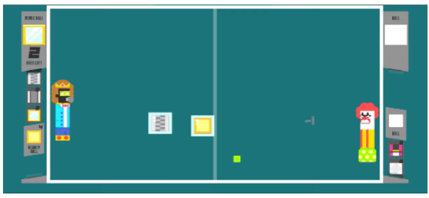
Example of other pong games



**Coded pong**



**Sprite pong**



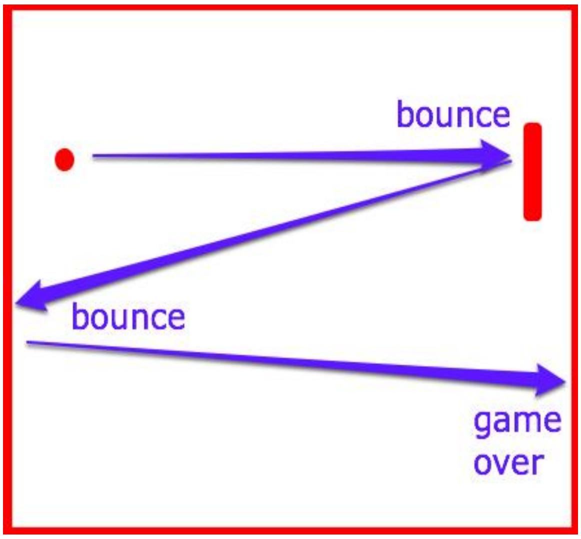
co

**Sprite pong**

There are 3 other ways of making pong but this image gives a basic idea of what they are supposed to look like, a coded pong is made from scratch without any sprite applied but the ones which as been invoked with sprite are these two that are more complicated because there is more coding involved. The sprited pong is usually the ones that is played most of time because there is more functionality attached.

## 3 Technical background and problem statement

The issues of ball hitting the paddle was quite complicated but can be solved using Boolean and a update statement. In this case there is a ball bouncing off a paddle then landing on a different angle which ends the game for a restart.



There different ways of getting the ball to hit the paddle but Boolean was a quick and fast way to accomplish these issues.

Creating a Boolean and setting values to true

```
Boolean shoot=true;

Boolean shoot1=true;

Boolean shoot2=true;
```

Setting Boolean to false for other direction

```
 shoot=false;

 shoot1=false;

shoot2=false;
```

Creating an integer called bladder for the update statement.

```
int baddle1=0;

int baddle2=0;
```

the Boolean is setting value of the ball movement while baddle is updating motion of ball after sliding off the side to count the point and restart the process.

Creating a float and giving them variables name to store the width and height.

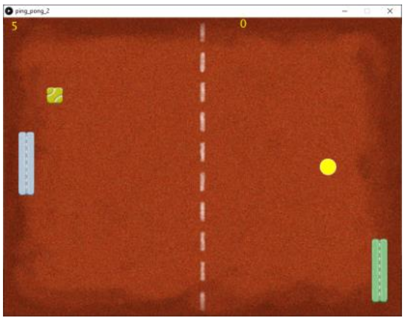
```
day=height/2-50;

  bax=width/2;

  bay=height/2;

  baxs=-2;

  bays=1;

  vex=width-60;

  vey=height/2-50;
```

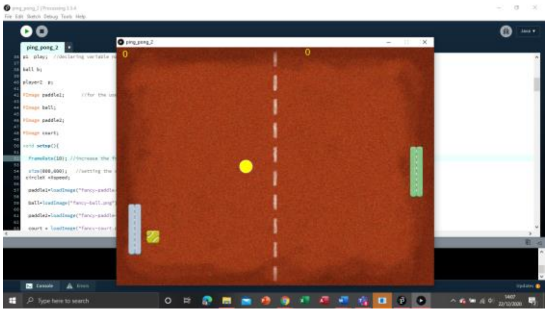
## 4 Solution

This program requires 3 steps to getting the ball touching paddle and bouncing off the side to count the point. The paddle and ball are defined by six floating point variables specifying its height and width.  There is an integer variable that update the statement after the ball goes out of place and increase the score board to 1 every time paddle doesn't hit the ball. The six floating point defines height and width which means the sprite invoked will be set at the heigh and width calculate in the code. If ball tend to scroll out of the width and height set then ball is out of bounce game resumes but if paddle hit the ball game continues.

Boolean values study motion of the paddle and also the ball, if the 1 paddle is set to true then ball moves and also paddle with the height set plus width but when ball goes out of angle set from height and width Boolean become false then score increase by 1 depending on which player side ball goes out of bounce.

The ball hit the paddle games continuous.




ping_pong_2 | Processing 3.5.4
File Edit Sketch Debug Tools Help
ping_pong_2
ping_pong_2
0
0
Console
Errors
Type here to search

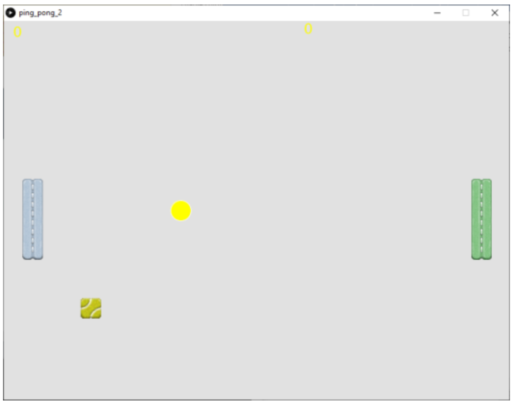
```
p1  play;  //declaring variable
ball b;
player2  p;
PImage paddle1;      //for the us
PImage ball;
PImage paddle2;
PImage court;
void setup(){
 frameRate(10); //increase the f
 size(800,800);   //setting the
circleX +=speed;
 paddle1=loadImage("fancy-paddle
 ball=loadImage("fancy-ball.png"
 paddle2=loadImage("fancy-paddle
 court = loadImage("fancy-court
```

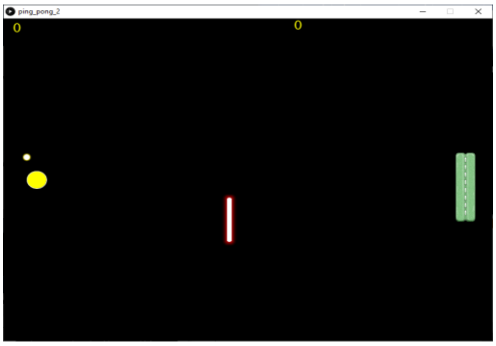
The score point is viewed at the top for both players. The yellow ball is there for distraction purpose to give a challenge to both players. This are sprite images that coded at a good height and width also, images show the movement of the ball, if this reaches the angle of the paddle the Boolean is set to false and then the integer increases by 1 and game starts over but if paddle hits ball bounces off and moves to the next paddle until ball goes out of angle range.

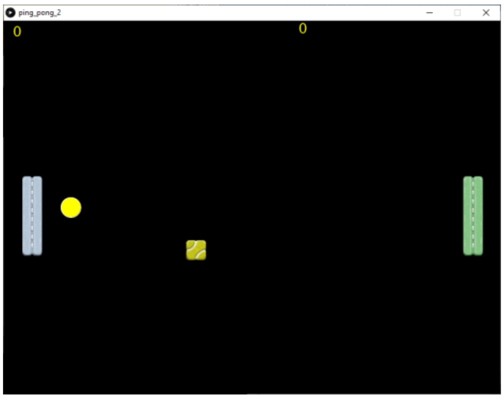
A full listing of the final program is included in section A at the end of this report.

## 5 Evaluation

This program is tested using different sprite style, different audio sounds to see which ones fits the game perfectly. Heights and width were changed to test for the different sizes for the paddle. Certain background was tested before a sprite background was attached. The key and mouse function were tested to view the easier option, mouse was too slow and complicated while key function is fast and quick for two players.







This are test case tested some include different background and some use of another sprite, most sprite tested didn't seem to work as seen in second picture the height of the 1 paddle hasn't been placed in correct position.

## 6 Conclusion

A program has been invented to help others get through stress referring back a case when entertain is needed to seduce stress.  This program also helps you test the different sprite of pong in case image changing is needed like your background or paddle colour. Happy with the work I have presented there might be some changes which still needs to be applied but that is future work. Code can be readjusted to just 1 paddle or four paddles but with the help of sprite coding.

## References

Website - https://en.scratch-wiki.info/wiki/Pong

Website - https://www.instructables.com/-Pong-Paddle

Website-  https://opengameart.org/

Website- https://www.maynoothuniversity.ie/library

Sample of project report

Lecture notes

Past coded activities in 171

Myself

## **Section 1** Source code listing

```
// Program to assist with entertainment.

//  E.olabode

// Last edit 17/12/2020
```

```
import ddf.minim.*;

//importing music into the game


Minim minim;

//using processing libray( Processing libray)

AudioPlayer groove;





float dax;

// declaring a floating variable names for the height and width of the game


float day;


float bax;


float bay;


float baxs;


float bays;


float vex;


float vey;


float circleX;

//declaring for a obstacle ball to spin forward and backward in same direction


float Xspeed=7;

// setting the ball speed to 50 after reaching the far end.


int baddle1=0;
```

```
//interger names for the movement of the ball and paddle


int baddle2=0;

//int p1 = 0;

//int p2 = 0;


boolean shoot=true;

//setting boolean values(lecture codes)


boolean shoot1=true;


boolean shoot2=true;


p1  play;

//declaring variable names for the 3 material needed


ball b;


player2  p;


PImage paddle1;

//for the used of sprite the images needs to be declared first.


PImage ball;


PImage paddle2;


PImage court;


void setup(){


  minim = new Minim(this);

  groove = minim.loadFile("gunna topfloor.mp3");
```

```
 // applying your music file.mp3
 groove.loop();
 frameRate(130);
 //increase the frameCount to determine the speed of the ball

 size(800,600,P3D);
 //setting the size 800 because the background needs to be fitted.


circleX =Xspeed;


 paddle1=loadImage("fancy-paddle-blue.png");
 //inventing sprite into the code (https://opengameart.org)

 ball=loadImage("fancy-ball.png");

 paddle2=loadImage("fancy-paddle-green.png");

 court = loadImage("fancy-court.png");

 dax=30;

 day=height/2-50;
 //setting the height and the width to be subtracted and divided by 2

 bax=width/2;

 bay=height/2;

 baxs=-2;

 bays=1;
```

```
  vex=width-60;


  vey=height/2-50;
}
void draw(){
  //display my draw block


  background(court);
  //setting background to the sprite image


  fill(255,255,0);


  stroke(255);


  ellipse(circleX,height/2,32,32);


  circleX = circleX + Xspeed;
  //Setting the speed for the obstacle ball


  if (circleX > width){
    // Using a statement to position the obstacle ball and making sure it rotates left and right at same position.


    Xspeed = -15;
}
if (circleX<0)


{


  Xspeed = 15;
}


  p1 play = new p1();
  player2 p = new player2();
  //declaring the values for the player to view their count

  ball b = new ball();

  play.view();play.input();play.update();

  b.view();b.update();

  p.view();p.update();

  count();
}



void reset(){

  dax=30;

  day=height/2-50;
  //Setting a reset counter when a ball goes out its range

  bax=width/2;

  bay=height/2;

  baxs=-2;

  bays=1;

  vex=width-60;
  vey=height/2-50;

  shoot=true;

  shoot1=true;

  shoot2=true;
}
void count(){
  //creating a void count

  textSize(22);

  text(""+baddle1,15,25);

  text(""+baddle2,500-25,20);



 //Setting the counter up so when a player scores a point it increases their value
  if (baddle1>=7){

    shoot=false;

    shoot1=false;

    shoot2=false;

    baxs=0;bays=0;
    text("Your the champion : Player2 \n Press Space key \n to start over ",400/2-40,300/2-10);   //A statement to determine the winner


    if (keyPressed){
     //placing a keypressed function for the first paddle


     if (key==' '){


      reset();baddle1=0;baddle2=0;
     }
    }
  }
  if (baddle2>=7){
    //placing a keypressed function for the second paddle


    shoot=false;


    shoot1=false;


    shoot2=false;


    baxs=0;bays=0;


    text("Your the champion : Player1 \n Press Space key \n to start over ",500/2-50,400/2-10);


    if (keyPressed){


     if (key==' '){


      reset();baddle1=0;baddle2=0;
     }
    }
  }
```

```
}

class p1

{

  void view(){
    //declaring a view block and a update statement

    if (shoot){


      image(paddle1,dax,day);
      // viewing the image of the paddle and setting the height

    }
  }
  void update(){

    if (day<0){

      day+=10;

  } else if (day+64>height){

    day-=10;

  }

}

  void input(){
    // using a key pressed function to move the first paddle (https://processing.org/)
    if (keyPressed){

      if (key=='n')

       {day+=5;}

      if (key== 'm')

     {day-=5;}

   }
    }
  }


class ball

{

  void view(){

    //view the ball
    if (shoot1){

      image(ball,bax,bay);
    }
  }

  void update(){
    //update statement

    bax = bax + baxs;
```

```
bay = bay + bays;

//Vertical

if (bax>vex-1){

  if (bay< vey+75&&bay>vey-1){

    baxs=-baxs+1;
  } else {

    reset();

    baddle2++;
  }
}

if (bax<dax+25){

  if (bay<day+75&&bay>day-1){

    baxs=-baxs+1;
  } else {

    reset();

    baddle1++;
  }
}
//horizontal

if (bay+16>height){
      bays=-bays;
    } else if (bay<0){

      bays=-bays;
    }
  }
}
class player2
{

  void view(){
    //viewing the second paddle

    if (shoot2){

      image(paddle2,vex,vey);
      //System.out.print(p2);

     }
   }

   void update(){
     //Also using a keypressed function of up and down  (https://processing.org/)


     if (keyPressed){

       if (keyCode==UP){

         vey-=5;
       }
       else if (keyCode==DOWN){
```

```
      vey+=5;
    }
  }

  if (vey+80>height){

    vey-=10;

} else if (vey<0){

  vey+=10;
  }
}
}
```