Programming Assignment – Pong – Design

The idea was to create a cooperative arcade video game inspired by the 1972 game. A Tennis based game that features two-dimensional graphics.
The aim of the game is to work with the other player to get as many rallies as possible while the ball is increasing in speed.
The players control the paddles on each side, moving them vertically across the left and right.

The JFrame will be 800 pixels by 600 pixels, and will be able to exit on close and be visible. Antialiasing will cause the movements of the ball and paddles to be much smoother.

The players will be able to control the paddles using keyboard inputs; I will do this by implementing KeyListeners to give each player the ability to press and release keys. They will also be given different key bindings to control their specific paddle.

The ball will be colliding with the sides of the frame and the paddles themselves. When the ball does interact with them, the ball will bounce off them, restricting the ball from leaving the frame.
The ball and paddle is planned to be the colour black and the background to be grey to simulate the 1972 black and white look and feel.

I will create a rectangle to be placed in the middle to simulate the border between each player; the colour of the rectangle will be black. The ball will not be able to collide with this rectangle.

The paddles should not be able to move in the x-directions. We are restricting their movement because moving closer to the ball could make affect the experience of the opposing player, causing it to become easier or harder for them.

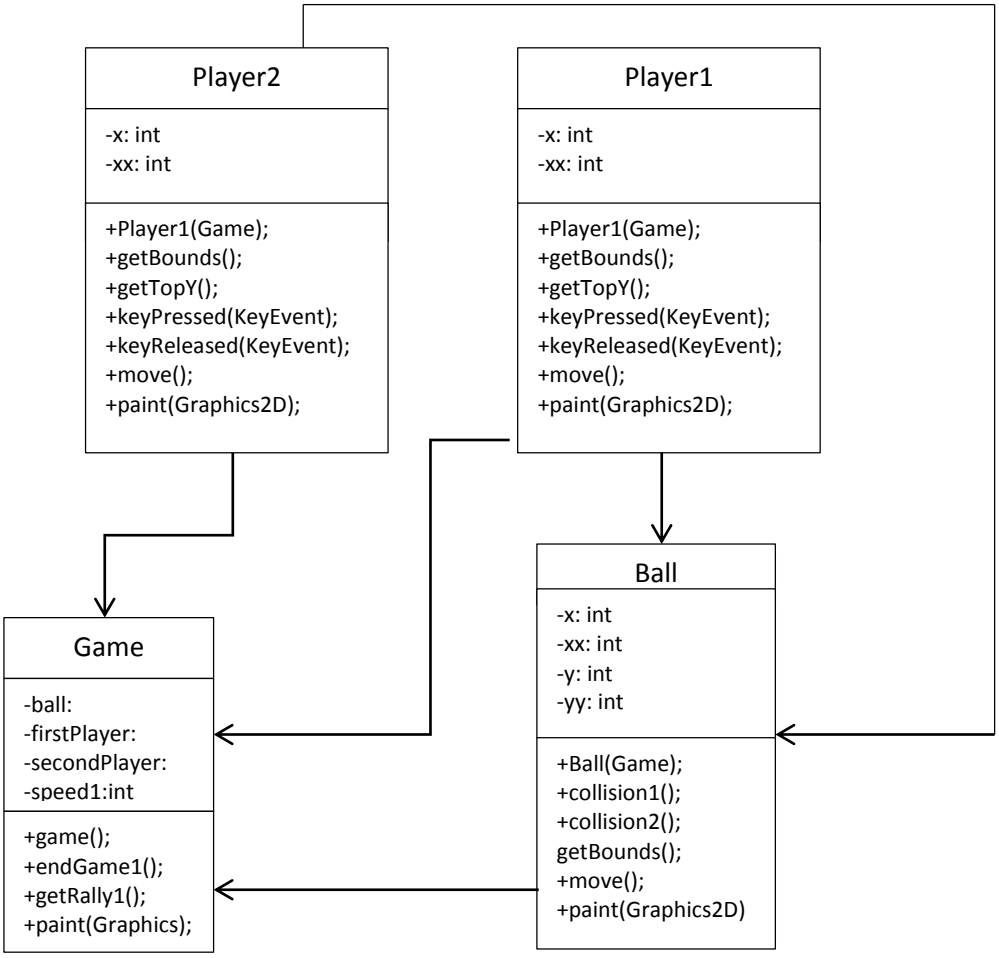
Once the players have lost the game, a JOptionPanel will appear giving the user relevant information regarding how well they played. The game will then terminate afterwards.

The ball and the paddles will be able to move due to repainting them every time the object moves position.

I will also implement a rally scoreboard on the top left of the frame, this will show how many times the paddles have interacted with the ball during the round.

I will also make the ball increase in speed every time it collides with the paddles, this will increase the difficulty of the game.

Class Diagram



Here we have the class diagram for my Pong video game created in Java. The class diagram contains the attributes and methods used to create the program for each class.

Game – Attributes:

- The Ball attribute contains data provided by the Ball class, this includes the movement, collisions of the ball and the bounds.

- The first and second player attributes gives the information about the paddles shown in their respective class, this includes movements of the paddle and collisions with the ball.

- The speed1 attribute will be initialised as 1 and will increase every time the ball hits a paddle.

Game – Methods:

- The game() method is the main method in the Game class, contains all the other methods in the class.

- The endgame() method will be active once the paddle interacts with the left and right side of the panel. Once it has, it will

- The getRally1() method will return the value of speed1 by -1

- The paint(Graphics) method will display each of the objects in the game effectively, repainting them as the objects change position.

Player1 – Attributes:

- The x attribute acts as the movement in the x-axis, the integer is a constant number so the paddle is unable to move from that x-axis.

- The xx attribute represents the speed of the paddle, similar to the ball. It also makes sure that the sprite doesn't go out of it's frame.

Player1 – Methods:

- The getBounds() method ensures that the paddle cannot leave the frame of the game when playing.

- The getTopY() method is used to give the position of the racket in the y-axis of the upper part of the racket.

- The keyPressed() method makes sure that the user can press the keys to move the paddle in the game.

- The keyReleased() method allows the user to release the key to stop the paddle from moving up or down in the game.

- The move() method allows the paddle to move in the game effectively

- The paint(Graphics2D) method will display the paddle for the game, also repaints the paddle when movement has occurred.

Player2 – Attributes:

- The x attribute acts as the movement in the x-axis, the integer is a constant number so the paddle is unable to move from that x-axis.

- The xx attribute represents the speed of the paddle, similar to the ball. It also makes sure that the sprite doesn't go out of its frame.

Player2 – Methods:

- The getBounds() method ensures that the paddle cannot leave the frame of the game when playing.
- The getTopY() method is used to give the position of the racket in the y-axis of the upper part of the racket.
- The keyPressed() method makes sure that the user can press the keys to move the paddle in the game.
- The keyReleased() method allows the user to release the key to stop the paddle from moving up or down in the game.
- The move() method allows the paddle to move in the game effectively
- The paint(Graphics2D) method will display the paddle for the game, also repaints the paddle when movement has occurred.

Ball – Attributes:

- The x attribute is used as the movement of the ball in the x-direction
- The xx attribute is used to increase and change the speed and direction of the ball when collisions occur.
- The y attribute is used as the movement of the ball in the y-direction
- The yy attribute is used to increase and change the speed and direction of the ball when collisions occur.

Ball – Methods:

- The collision1() method is used once the ball has collided with the paddle of the first player, causing the ball to reflect back towards the second player.
- The collision2() method is used once the ball has collided with the paddle of the first player, causing the ball to reflect back towards the second player.
    - getBounds() method is used to stop the ball from leaving the frame of the game and bounce rebound from the borders.
- Move() method is used to get the ball moving, the ball will stay stationary without this method.
- Paint(Graphics2D) used paint the ball so it appears when it is moving and when it's stationary.

Screen Designs

Below I have produced basic designs of what I'd like the program to look like ingame. The rectangle in the middle is there to separate the two players. The rectangle, paddles and ball to appear black with a grey background to give the retro appearance.

There will have a score on the top left of the game showing the players their rally score.

Once the players have let the ball get past them, a JOptionPane should appear displaying the score and will terminate the project once the user has clicked the "OK" button.