Pooja Singh

Final Project
Brick Breaker

For my final project, I created a Brick Breaker game, which works very similarly to the Apple version of the game.

Small details- At the top of the game, I have labeled what game the user is playing. Using Processing to insert the text's, color and size were meant to compliment the layout of the game. This game contains 30 bricks to start with (this could be manipulated in other versions using the array). I have imprinted a "brick" image on them, to make them a bit more visually interesting. The bricks are laid over a sky colored background (also an image that was imported) to make it seem more realistic rather than abstract. At the bottom, the paddle, which the player will use to ensure the ball bounces, has a wood type image, which was imported. I decided to have a black outline of it so the player can see the paddle more clearly. The ball is meant to signify a wrecking ball, hence its black color.

First, a ball starts at the center of the screen. The ball is programed to move down initially and the player must use the paddle at the bottom of the screen to make the ball bounce back up. The user should use the **left** and **right** arrow keys to move the paddle at the bottom of the screen. The direction in which the ball moves is dependent on where the ball hits on the paddle: if it hits the left half of the paddle, the ball will bounce back to the left, if it hits the right side of the paddle, the ball will bounce right. In the project proposal, I had initially wrote that the ball direction would be random as long as the player hits the ball with the paddle; however I realized after I wrote this code that the randomization makes it hard for the player to strategically play the game and get all the bricks.

The ball then moves at a continuous pace towards the bricks above. If the ball comes in contact with the edge of the brick, the brick disappears and the ball will bounce back towards the paddle, which the player should move. In this game, the player is allowed to hit more than one brick in one move.

The player continues moving the paddle to block the ball from falling of the screen until all the blocks are gone. Notice, if the ball hits the side of the paddle, this does not count as the top of the paddle. Therefore, if the side of the paddle is hit, the ball will fall through and the player will have to reset. If the ball disappears beneath the paddle, then the player loses and they can push **enter** to have the game "reset". The reset method ensures that the bricks are re-drawn, the ball goes back to the center of the screen and they can restart from the beginning.

In the project proposal, I had written that I would use the backspace key as the reset button. I discovered on my laptop that it is much easier to use the enter key because on my personal keyboard, the backspace bar was starting to glitch. The methodology behind it was the same and this could easily be changed should someone have an alternative preference. I also added a few extra details, which were not outlined in my project proposal just because I thought it would help it be more visually appealing.