Assembly Project: Breakout

Jack Liu, 1008012124 and Natalie Duarte, 1008009986

April 3, 2023

1 Instruction and Summary

1. Which milestones were implemented? All milestones were implemented.

   Easy Features

   (a) Player "health" aka multiple attempts - there are three hearts displayed at the top of the screen; after three losses, the game is over and the player loses. Note that health is not the same as score.

   (b) When the player has used up their 3 attempts, display a Game Over screen. Restart the game if the retry option is selected, and go back to the main menu if the home option is selected.

   (c) The user can pause the game by pressing the keyboard key p.

   (d) The player can launch the ball at the beginning of each attempt.

   Hard Features

   (a) When the ball bounces off the paddle, the angle at which it bounces changes based on which side of the paddle the ball comes into contact with. If the ball hits the middle of the paddle, it bounces straight back up, and the closer to either edge it hits, the wider the angle. This implementation allows the ball to bounce 30, 60, 120, or 150 degrees, depending on where on the paddle it has been hit (and technically also 45 degrees because of the way the ball's horizontal and vertical velocity change in 0.5 increments).

   (b) Bricks must be hit multiple times before breaking and their "health" is shown with a change in colour. There are three health levels for the bricks.

   (c) The player's score is tracked and displayed at the top left of the screen

   (d) There are three different levels with different brick layouts (rainbow, diamond, etc., based on Pink Floyd's Dark Side of the Moon album).

   (e) There is a main menu in which you can select the level to play, which you see when you press quit.

2. How to view the game:

   (a) Open the code in MARS (or any alternative assembly IDE)

   (b) Open bitmap display, set pixel width/height to 8, set display width/height to 256. You can also set it to other values as long as the screen is 32 x 32 pixels, which is what the game is built to work on. Connect display to MARS.

   (c) Open keyboard simulator and connect it to MARS.

   (d) Compile code and press RUN.

3. Game Summary:

   - Brick breaker game

- Main Menu. Use A and D to select the level of the game to play, then press the space bar to start playing at that level.
- Use A and D to move paddle
- Ball follows paddle until space is clicked, which launches the ball straight upwards
- Ball will bounce on bricks, all bricks have 3 health and will grow dimmer based on remaining health.
- When a brick goes to 0 health, they disappear and no longer performs collision check with the ball.
- When ball bounces on paddle, it changes direction based on where it landed on the paddle. There are many different angles it could bounce in; the farther to the edge the ball is hit, the greater the bounce angle.
- If the ball falls off the screen, the player loses a life, and the ball returns to an unlaunched position. Press space to launch again.
- If the player has used up all three lives, they can use A and D to select whether they want to restart the level or go back to the main menu. Press space to "click" the option.
- To pause the game, press the p key; to quit and return to the main menu, press the q key.

2 Attribution Table

Student 1 Jack Liu 1008012124	Natalie Duarte 1008009986
Report (Summary, View Game)	Report (Summary, Milestones description)
Draw walls, ball	Draw bricks, paddle
Move ball	Redraw screen
Ball bouncing	Brick health progression
Health tracker	Collision check
Main Menu	Game Over Screen
Diamond Level	Another Brick in the Wall Level
Score tracking, calculation	Score display