TTG Team - Still No Name Yet: (Insert Undecided Game Name Here)

Connor Blakey
C.Blakey@2014.hull.ac.uk

Matthew Burling
M.Burling@2014.hull.ac.uk

Marc Mardare
M.Mardare@2014.hull.ac.uk

Richard Smith
Richard.Smith@2014.hull.ac.uk

October 22, 2014

Contents

1 Instructions for Compilation

2 Gameplay Instructions

3 Interesting Bits of the Internal Workings

3.1 Life System

The player is to receive 3 lives to play with when the game starts. The `Lives` variable is initialised in the preamble and in the `LoadGame` method. The Life functionality is provided in the `Update` method and is triggered by an `if` statement when the object is exiting the bounding box or 'Arena':

```
            }
```

We make use of the Boolean `LeftArena` to make sure that this code is somehow no run without us foreseeing and to enable a simple while loop for the two case scenarios of the amount of lives left:

```
            {
                spriteBatch.Draw(ResumeButton, ResumeButtonPOS,
                    Color.White);
            }
            spriteBatch.End();

            base.Draw(gameTime);
        }
        void LoadGame()
        {
            sprite = Content.Load<Texture2D>("Images/Orb");
            PauseButton = Content.Load<Texture2D>("Images/123");
            ResumeButton = Content.Load<Texture2D>("Images/1234"
                );
            ResumeButtonPOS = new Vector2((GraphicsDevice.
                Viewport.Width / 2) - (ResumeButton.Width / 2), (
                GraphicsDevice.Viewport.Height / 2) - (
                ResumeButton.Height / 2));
            SpritePOS = new Vector2((GraphicsDevice.Viewport.
                Width / 2) - (SpriteWidth / 2), (GraphicsDevice.
                Viewport.Height / 2) - (SpriteHeight / 2));

            // Why not yolo
            Thread.Sleep(3000);

            gameStates = GameStates.Playing;
            isLoading = false;

        }
        void MouseClicked(int x, int y)
```

and to keep evaluating an `if` to check if we lost a life and reset the game with our `ResetGame` method:

```
            gameStates = GameStates.Playing;
            isLoading = false;

        }
```

3.1.1 Bugs

It was found that we actually interpreted the `Lives` variable in a binary format (0, 1, 2, 3, 4 etc., where we count from zero first) instead of base 10 so when `Lives` was originally set to the value of 3, the user was able to have 4 goes of the game instead. This was simply fixed by initialising `lives` to 2 instead since 0 is counted as a number. We left the algorithm as-is as that's how Mathematics and algorithms should be done.

4 Idea Documentation

5 Tips/Mistakes When Using C#

Name	Mistake	Solution	Time Taken to Resolve
Matt	Incorrectly nested `if` statements	Think about the ordered sequence of your program (because Rob Miles reminded us)	2 : 00