Anton Guaman

Professor Hogan

Engineering 1020

October 1, 2020

**Prelab 5**

1. **Selection**

I will do the speed work project.

1. **Design**

The speed work project will result in a competitive game between two players that have to input data via the button or the touch sensor. The first player to digitally input will win. We are aware that the code created will also have the option of a draw if the players digital input occurs at the same time. With this information we will precede to describe the algorithm.

For this algorithm we will not take in consideration the time module as stated in the prelab 5 description.

We will begin the game by explaining how the game works using the print command. The explanation will include how the users will know when the game starts. Which digital input represents Player 1 and Player 2.

In order to make the game run multiple times without having the users run the code multiple times we can implement a will True: loop that will allow users to play as many times as they wish and it will also give them the opportunity to quit when they wish.

There will be two variables one storing the input from Player 1 and Player 2. For Player 1 it will be b1 and for Player 2 it will be b2.

After this we will manipulate the LCD color of the screen store in variable x with the command lcd\_rgb(225, 255, 255)for white and then it will change to red with the command lcd\_rgb(0,225,0) stored in variable y. This will show the users that the game has started.

Then we will proceed to check which Players input was first pressed using a loop. We will compare which player pressed the button pressed or if it’s a draw with if and elif statements. The variable storing the player who pressed the button first will be winner which can store b1 or b2. The variable winner can also store a tie.

When a button is pressed, we will change the color of the LCD screen with the command lcd\_rgb(green) demonstrating that a button has been pressed.

After, we will store the value of the duration of the game in the variable “duration” this will be calculated through the random module time which we will implement in the lab.

Lastly, the game is complete we will give the users the option to play again or quit depending on the input the loop will run again or stop.

1. **Testing**

|  |  |  |  |
| --- | --- | --- | --- |
| **Testing Matrix** | | | |
| **Test number** | 1st Test | 2nd Test | 3rd test |
| **Digital input state comparison** | b1>b2 | b1< b2 | b1 == b2 |
| **Winner?(Expected output)** | b1 player wins. Store in winner variable = b1. Additionally, print(“Player 1 wins”) | b2 player wins. Store in winner variable = b2. Additionally, print(“Player 1 wins”) | b1 and b2 tie. Leave the variable winner empty. Additionally, print(“Good job, it’s a draw”) |
| b2 player loses. | b1 player loses. | b1 and b2 tie. |

b1(Player 1) = button input

b2(Player 2) = touch sensor

* For the first test I will press the button first and verify that my expected output matches.
* For the second test I will press the touch sensor first and verify that my expected output matches.
* For the third test I will press the button and touch sensor simultaneously and verify that my expected output matches.

\*For all the verifications of this test it is key to print so we can verify the winner.