

Credit Name: CSE 2920 - CSE Project C

Assignment: TugOfWar

How has your program changed from planning to coding to now? Please Explain

```
public static void main(String[] args) throws Exception{

    //Create
    DigitalInput redButton = new DigitalInput();
    DigitalOutput redLED = new DigitalOutput();
    DigitalInput greenButton = new DigitalInput();
    DigitalOutput greenLED = new DigitalOutput();

    //Address
    redButton.setHubPort(0);
    redButton.setIsHubPortDevice(true);
    redLED.setHubPort(1);
    redLED.setIsHubPortDevice(true);
    greenButton.setHubPort(5);
    greenButton.setIsHubPortDevice(true);
    greenLED.setHubPort(4);
    greenLED.setIsHubPortDevice(true);

    //Open
    redButton.open(1000);
    redLED.open(1000);
    greenButton.open(1000);
    greenLED.open(1000);
}
```

I firstly created, addressed and opened all buttons and LEDs as in previous lessons.

```
int player1 = 0;
int player2 = 0;

System.out.println("Player 1 is Red");
System.out.println("Player 2 is Green");
```

I then declared and initialized both players' scores. Then I printed a statement to indicate which player uses which button.

```

while(player1 < 10 && player2 < 10)
{
    if(redButton.getState())
    {
        player1++;

        System.out.println("Player 1: " + player1);

        Thread.sleep(150);
    }

    if(greenButton.getState())
    {
        player2++;

        System.out.println("Player 2: " + player2);

        Thread.sleep(150);
    }
    Thread.sleep(100);
}

```

Then i set up a while loop which is for the main portion of the game where player1 and player2 race to press the buttons. If the red button is pressed (its state is checked by code), player1's score increases by 1 and their score is displayed. The exact same process is repeated with player2. There is a lag of 150 ms between each players presses to ensure the sensitivity of the button is not abused within the game. This is repeated until either player 1 or player 2 reaches a score of 10.

```

redLED.setState(true);
greenLED.setState(true);
Thread.sleep(1000);

redLED.setState(false);
greenLED.setState(false);
Thread.sleep(1000);

```

Once the game ends, both LEDs flash for a second.

```

    if(player1 == 10)
    {
        System.out.print("Player 1 Wins!");

        for(int i = 0; i<5; i++)
        {
            redLED.setState(true);
            Thread.sleep(1000);
            redLED.setState(false);
            Thread.sleep(1000);
        }
    }

    if (player2 == 10)
    {
        System.out.print("Player 2 Wins!");

        for(int i = 0; i<5; i++)
        {
            greenLED.setState(true);
            Thread.sleep(1000);
            greenLED.setState(false);
            Thread.sleep(1000);
        }
    }
}

```

Lastly, if player 1 reaches 10, a statement is shown which states that they won. Then a for loop which repeats 5 times flashes the red LED (for player 1) on and off for a second.

The same is done for player 2's case of winning but the green LED flashes instead.