BOP IT!

LAB 06

ARYAN RAO

SECTION 5

DATE-10/20/2021

SUBMISSION DATE-10/26/2021

PROBLEM

Create a game where the user is asked to push a button on the DualShock4 and the game continues till either the time runs out or a wrong button is pressed.

ANALYSIS

Use the ds4rd.exe file provided. We also need to provide a menu and give the user instructions on what to do. The button that is to be pressed will be determined randomly by the computer.

DESIGN

- Download the ds4rd.exe
- Create a menu
- Keep track of time
- End if time runs out or a wrong button is pressed

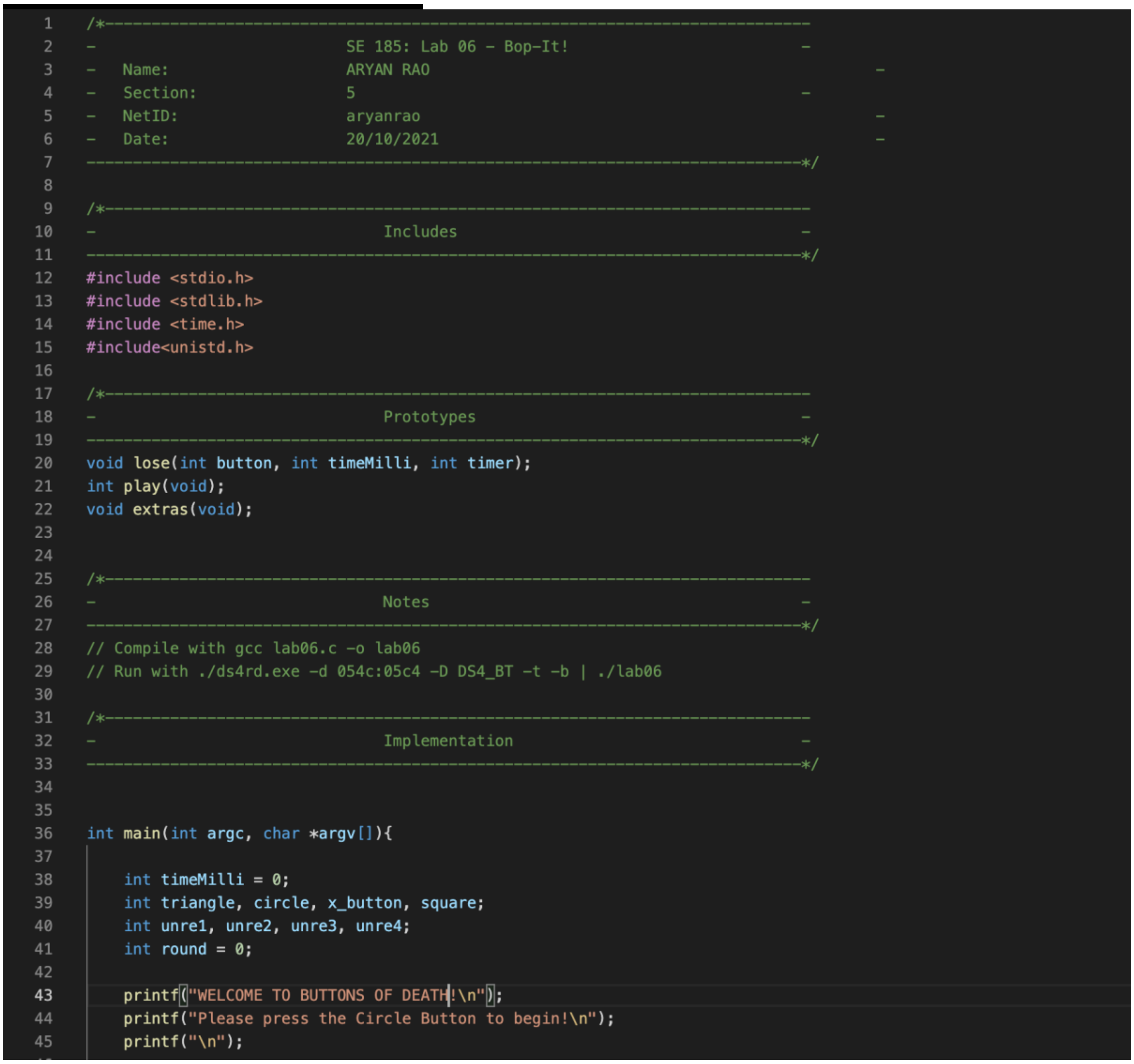
TESTING

The program should end when time runs out. Manually keep track of time and compare with the code.

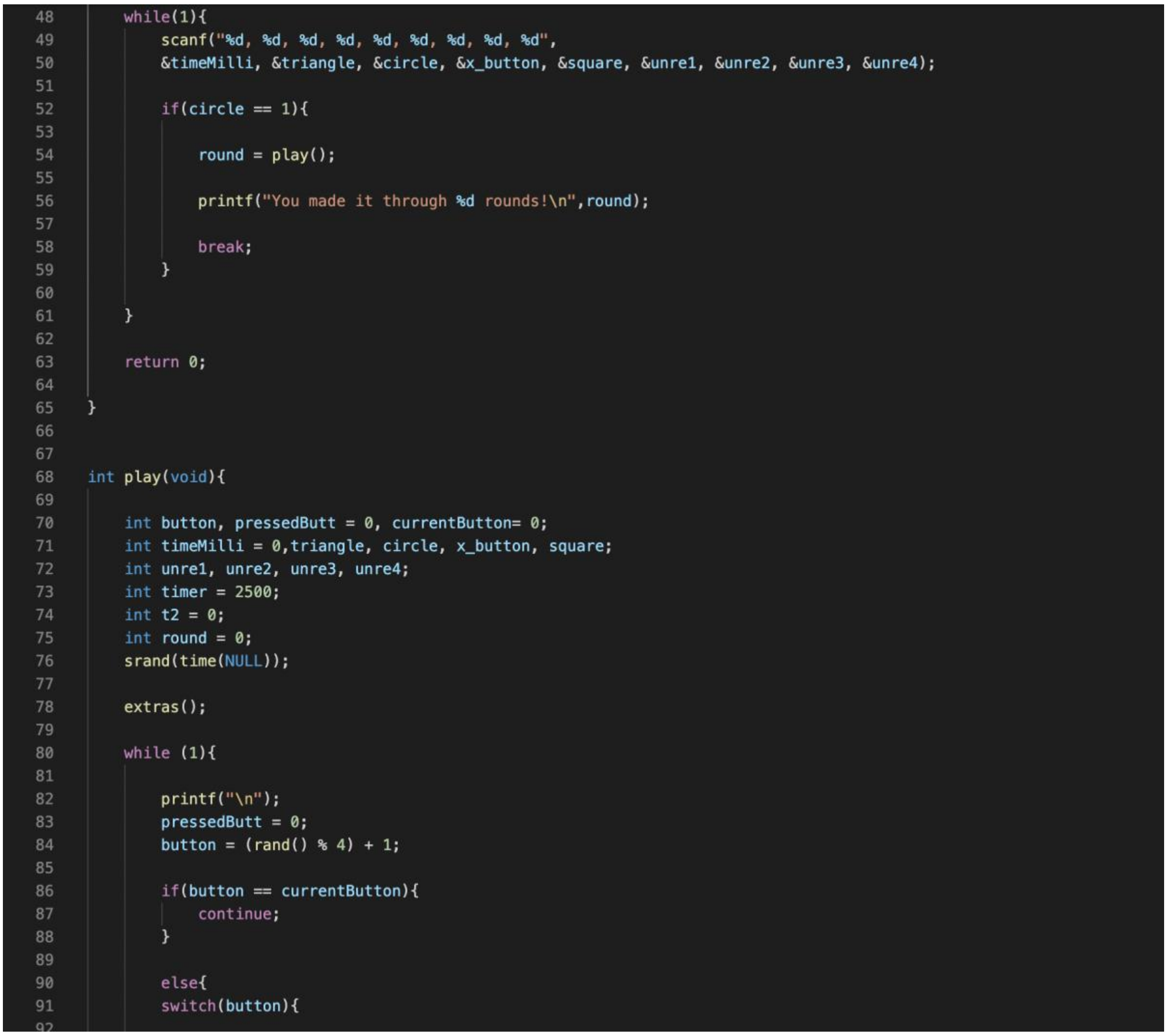
COMMENTS

Make sure the code only registers one press of a button on the DS4. Also be sure to have the correct mechanism to reduce time.

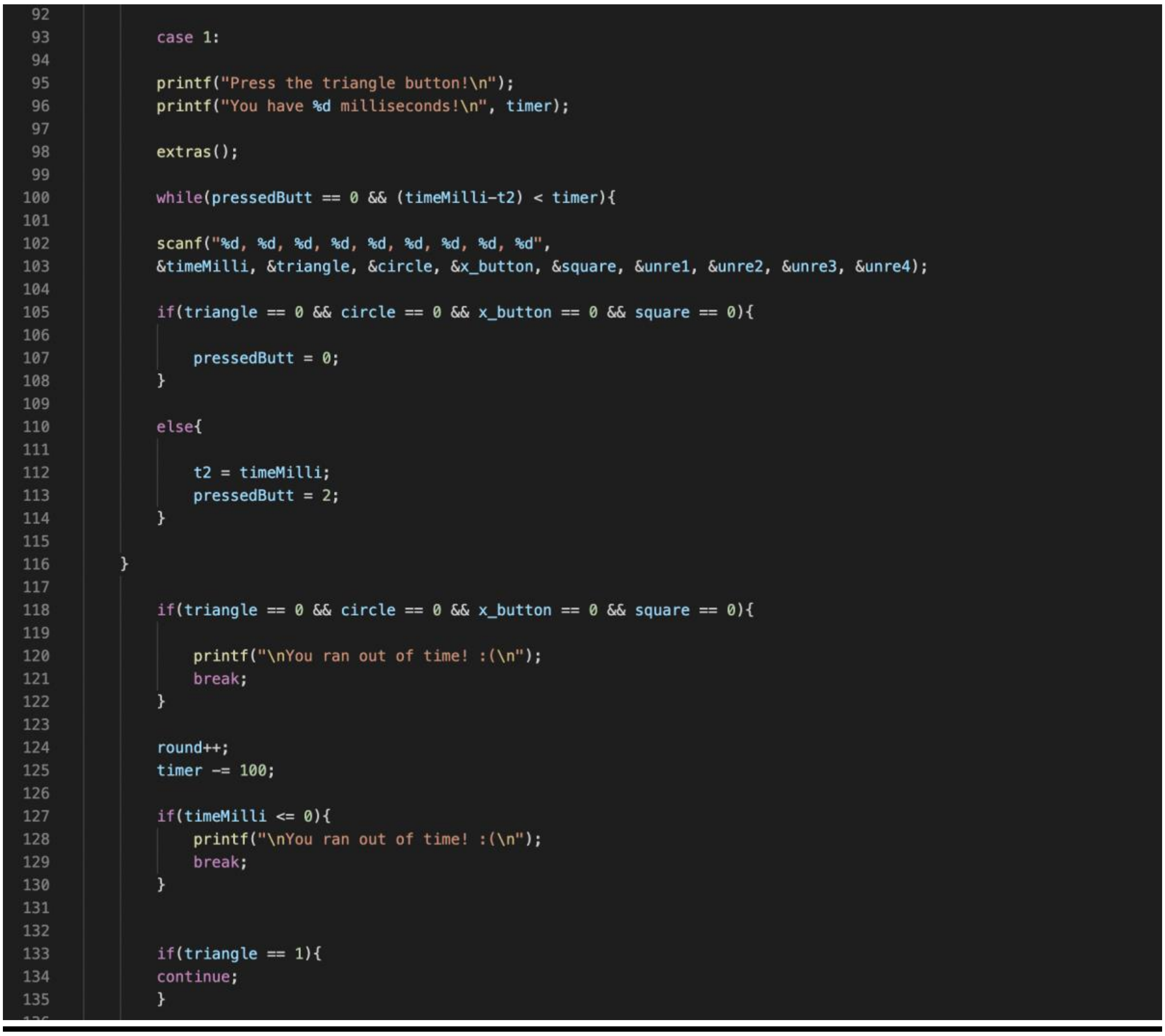
SOURCE CODE:



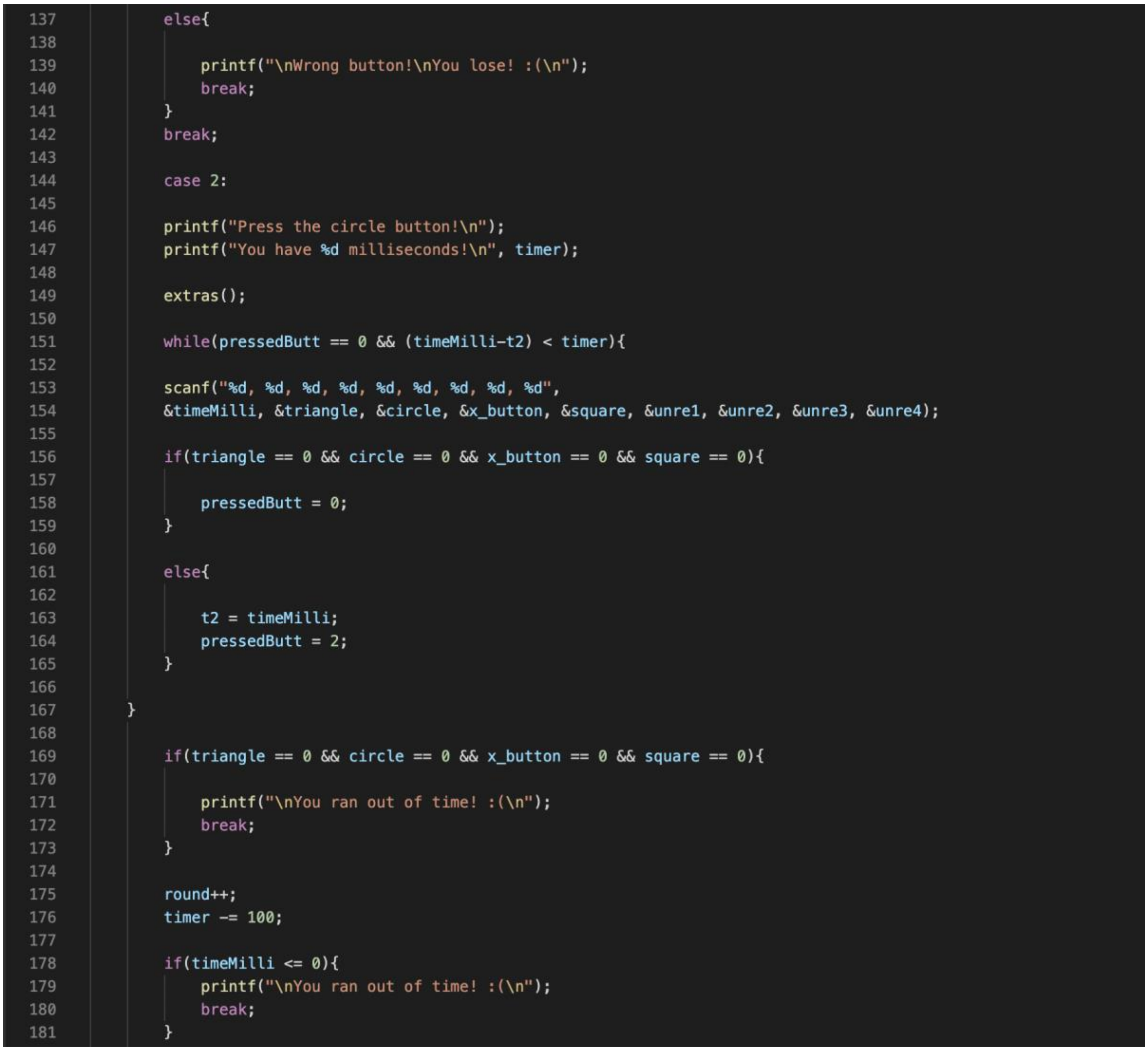
```c
/*----------------------------------------------------------------------------
-                           SE 185: Lab 06 - Bop-It!                         -
-   Name:                   ARYAN RAO                                                -
-   Section:                5                                                -
-   NetID:                  aryanrao                                                 -
-   Date:                   20/10/2021                                               -
-----------------------------------------------------------------------------*/

/*----------------------------------------------------------------------------
-                               Includes                                     -
-----------------------------------------------------------------------------*/
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include<unistd.h>

/*----------------------------------------------------------------------------
-                               Prototypes                                   -
-----------------------------------------------------------------------------*/
void lose(int button, int timeMilli, int timer);
int play(void);
void extras(void);


/*----------------------------------------------------------------------------
-                               Notes                                        -
-----------------------------------------------------------------------------*/
// Compile with gcc lab06.c -o lab06
// Run with ./ds4rd.exe -d 054c:05c4 -D DS4_BT -t -b | ./lab06

/*----------------------------------------------------------------------------
-                               Implementation                               -
-----------------------------------------------------------------------------*/


int main(int argc, char *argv[]){

    int timeMilli = 0;
    int triangle, circle, x_button, square;
    int unre1, unre2, unre3, unre4;
    int round = 0;

    printf("WELCOME TO BUTTONS OF DEATH!\n");
    printf("Please press the Circle Button to begin!\n");
    printf("\n");
```



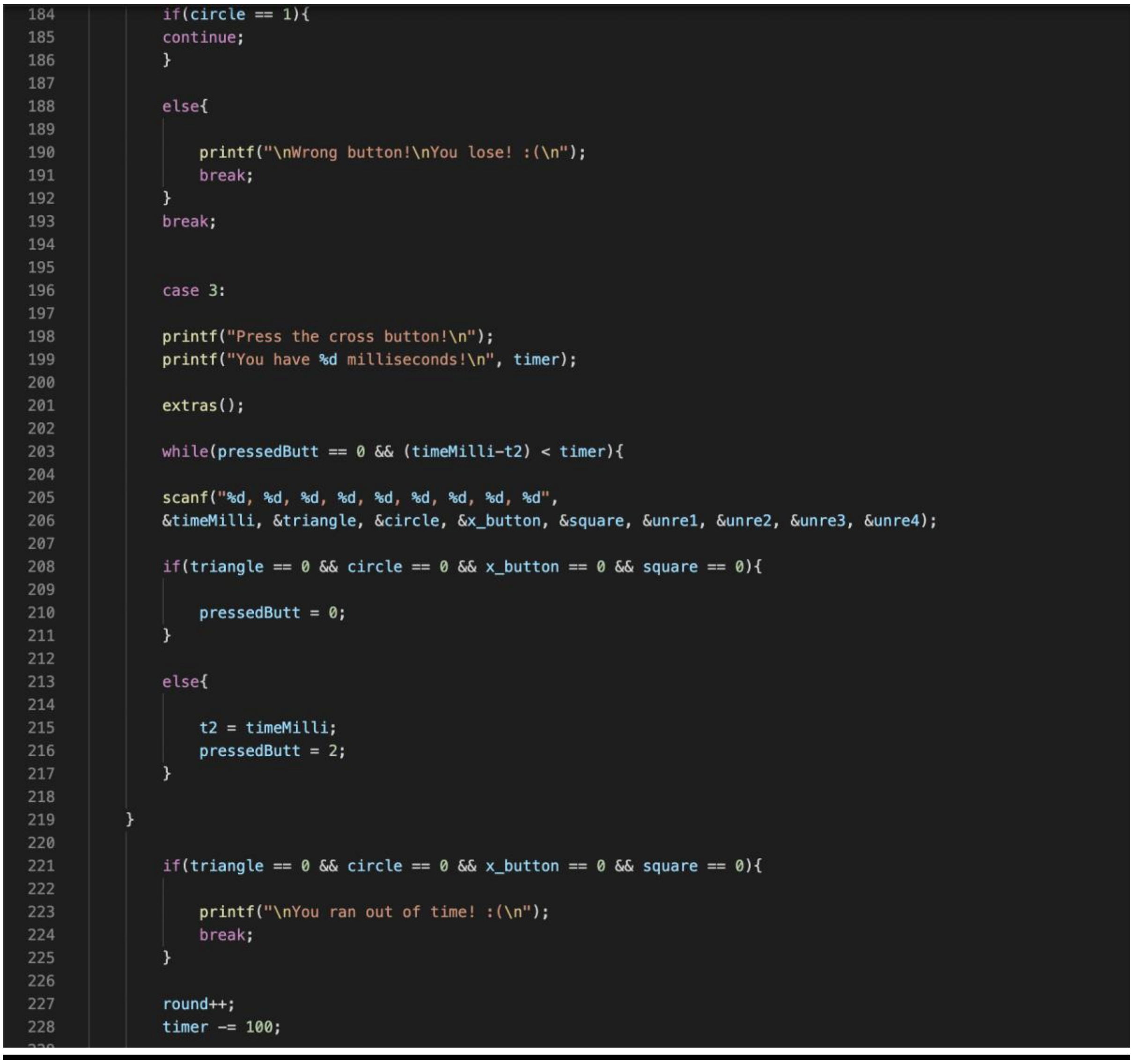
```c
    while(1){
        scanf("%d, %d, %d, %d, %d, %d, %d, %d, %d",
        &timeMilli, &triangle, &circle, &x_button, &square, &unre1, &unre2, &unre3, &unre4);

        if(circle == 1){

            round = play();

            printf("You made it through %d rounds!\n",round);

            break;
        }

    }

    return 0;

}


int play(void){

    int button, pressedButt = 0, currentButton= 0;
    int timeMilli = 0,triangle, circle, x_button, square;
    int unre1, unre2, unre3, unre4;
    int timer = 2500;
    int t2 = 0;
    int round = 0;
    srand(time(NULL));

    extras();

    while (1){

        printf("\n");
        pressedButt = 0;
        button = (rand() % 4) + 1;

        if(button == currentButton){
            continue;
        }

        else{
        switch(button){
```



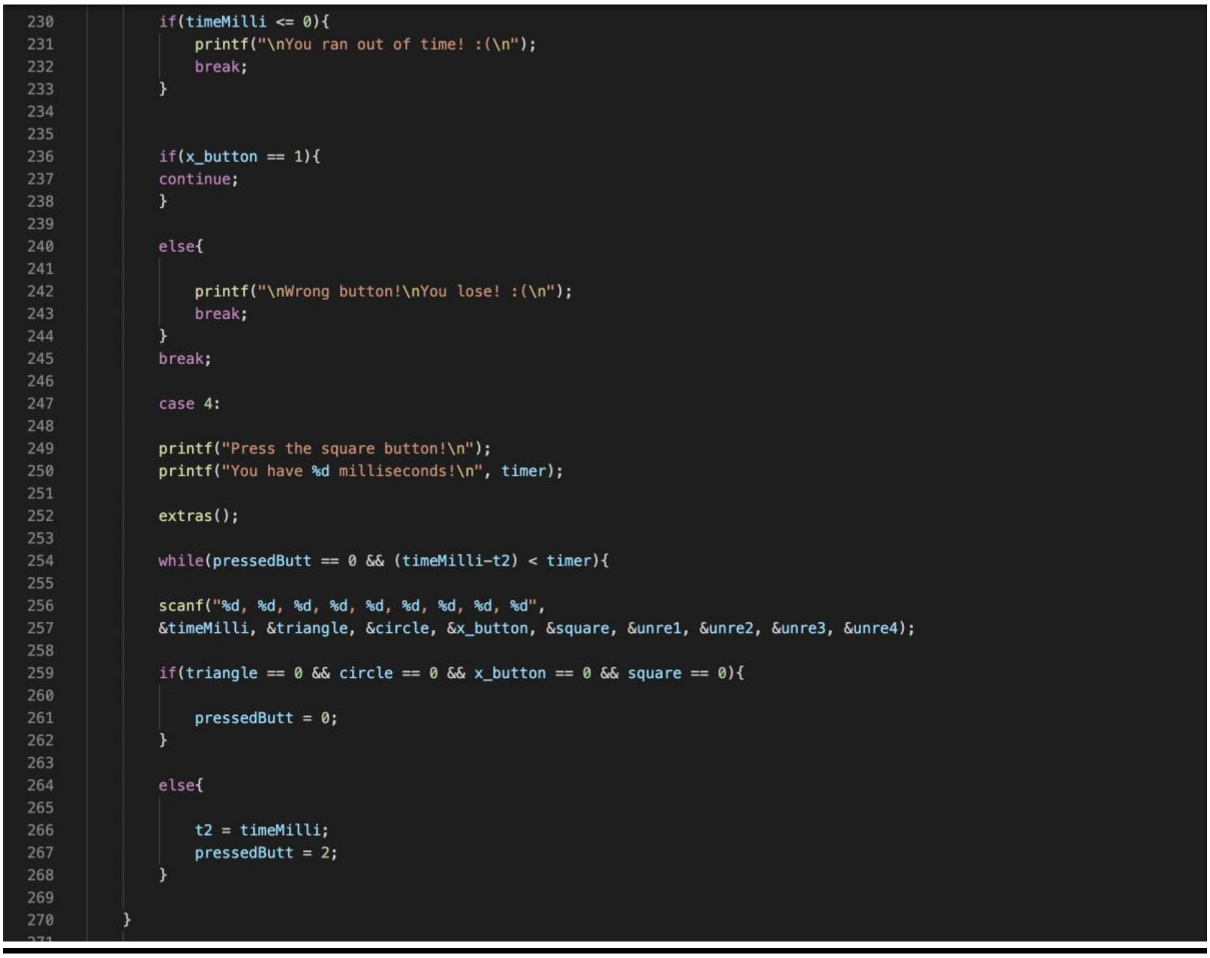
```
        case 1:

        printf("Press the triangle button!\n");
        printf("You have %d milliseconds!\n", timer);

        extras();

        while(pressedButt == 0 && (timeMilli-t2) < timer){

        scanf("%d, %d, %d, %d, %d, %d, %d, %d, %d",
        &timeMilli, &triangle, &circle, &x_button, &square, &unre1, &unre2, &unre3, &unre4);

        if(triangle == 0 && circle == 0 && x_button == 0 && square == 0){

            pressedButt = 0;
        }

        else{

            t2 = timeMilli;
            pressedButt = 2;
        }

    }

        if(triangle == 0 && circle == 0 && x_button == 0 && square == 0){

            printf("\nYou ran out of time! :(\n");
            break;
        }

        round++;
        timer -= 100;

        if(timeMilli <= 0){
            printf("\nYou ran out of time! :(\n");
            break;
        }


        if(triangle == 1){
        continue;
        }
```



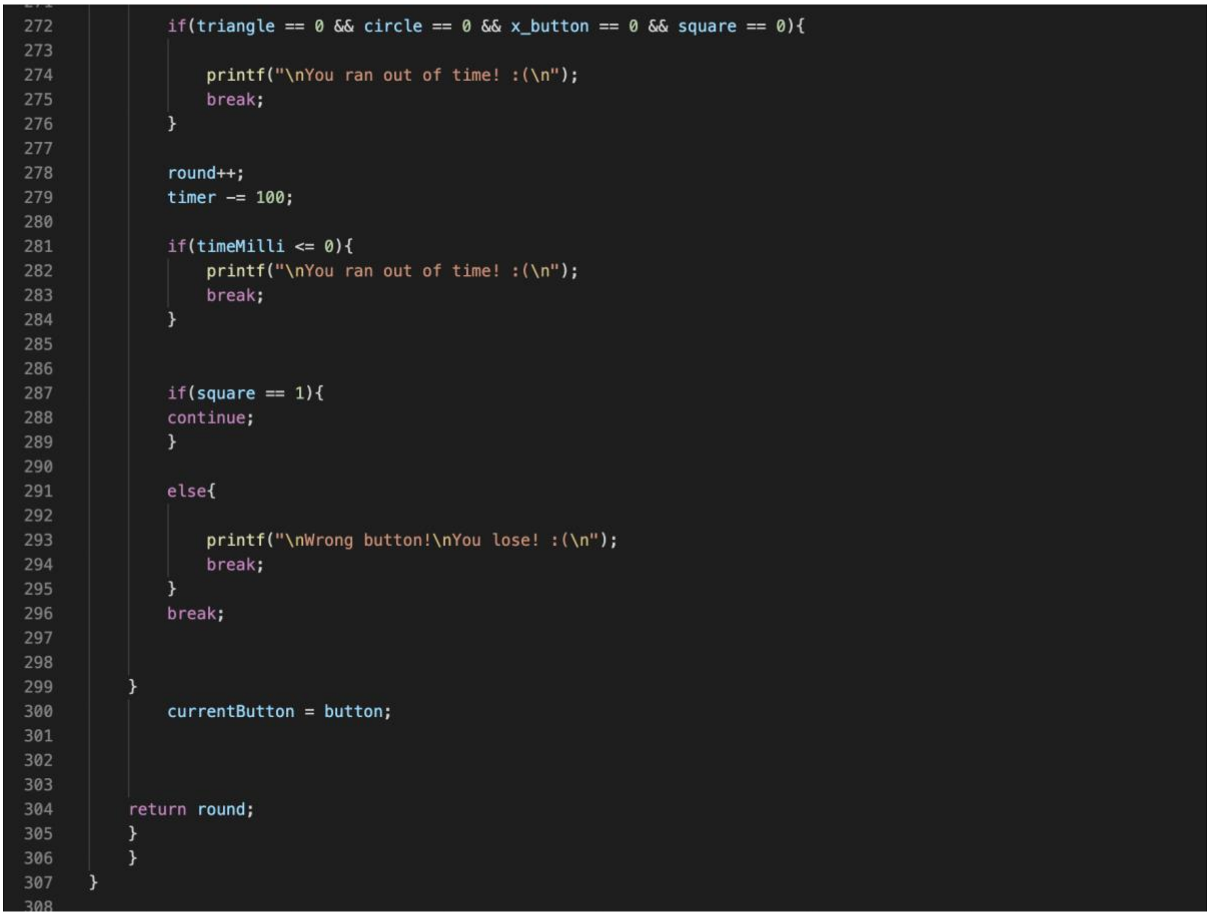
```
        else{

            printf("\nWrong button!\nYou lose! :(\n");
            break;
        }
        break;

        case 2:

        printf("Press the circle button!\n");
        printf("You have %d milliseconds!\n", timer);

        extras();

        while(pressedButt == 0 && (timeMilli-t2) < timer){

        scanf("%d, %d, %d, %d, %d, %d, %d, %d, %d",
        &timeMilli, &triangle, &circle, &x_button, &square, &unre1, &unre2, &unre3, &unre4);

        if(triangle == 0 && circle == 0 && x_button == 0 && square == 0){

            pressedButt = 0;
        }

        else{

            t2 = timeMilli;
            pressedButt = 2;
        }

    }

        if(triangle == 0 && circle == 0 && x_button == 0 && square == 0){

            printf("\nYou ran out of time! :(\n");
            break;
        }

        round++;
        timer -= 100;

        if(timeMilli <= 0){
            printf("\nYou ran out of time! :(\n");
            break;
        }
```



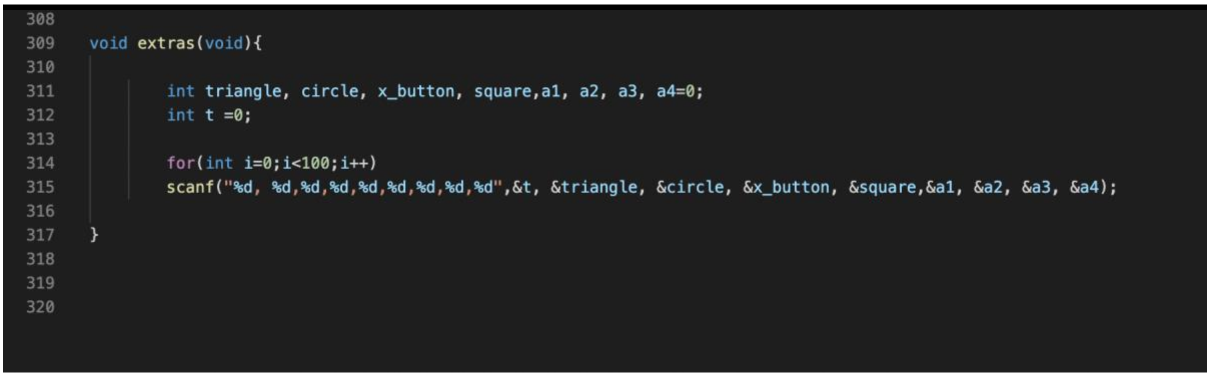
```
        if(circle == 1){
        continue;
        }

        else{

            printf("\nWrong button!\nYou lose! :(\n");
            break;
        }
        break;


        case 3:

        printf("Press the cross button!\n");
        printf("You have %d milliseconds!\n", timer);

        extras();

        while(pressedButt == 0 && (timeMilli-t2) < timer){

        scanf("%d, %d, %d, %d, %d, %d, %d, %d, %d",
        &timeMilli, &triangle, &circle, &x_button, &square, &unre1, &unre2, &unre3, &unre4);

        if(triangle == 0 && circle == 0 && x_button == 0 && square == 0){

            pressedButt = 0;
        }

        else{

            t2 = timeMilli;
            pressedButt = 2;
        }

    }

        if(triangle == 0 && circle == 0 && x_button == 0 && square == 0){

            printf("\nYou ran out of time! :(\n");
            break;
        }

        round++;
        timer -= 100;
```



```
            if(timeMilli <= 0){
                printf("\nYou ran out of time! :(\n");
                break;
            }


            if(x_button == 1){
            continue;
            }

            else{

                printf("\nWrong button!\nYou lose! :(\n");
                break;
            }
            break;

            case 4:

            printf("Press the square button!\n");
            printf("You have %d milliseconds!\n", timer);

            extras();

            while(pressedButt == 0 && (timeMilli-t2) < timer){

            scanf("%d, %d, %d, %d, %d, %d, %d, %d, %d",
            &timeMilli, &triangle, &circle, &x_button, &square, &unre1, &unre2, &unre3, &unre4);

            if(triangle == 0 && circle == 0 && x_button == 0 && square == 0){

                pressedButt = 0;
            }

            else{

                t2 = timeMilli;
                pressedButt = 2;
            }

        }
```

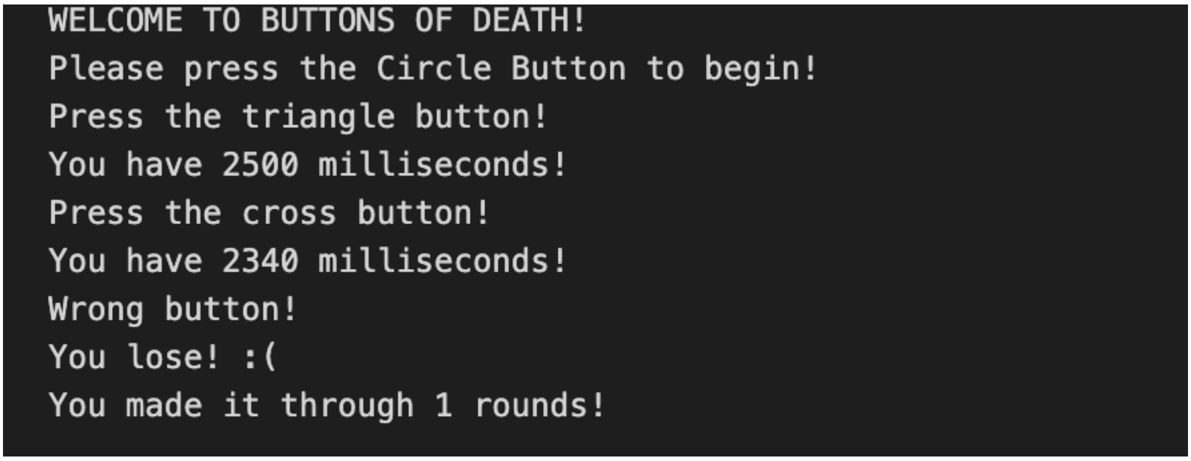


```c
        if(triangle == 0 && circle == 0 && x_button == 0 && square == 0){

            printf("\nYou ran out of time! :(\n");
            break;
        }

        round++;
        timer -= 100;

        if(timeMilli <= 0){
            printf("\nYou ran out of time! :(\n");
            break;
        }


        if(square == 1){
        continue;
        }

        else{

            printf("\nWrong button!\nYou lose! :(\n");
            break;
        }
        break;


    }
        currentButton = button;



    return round;
    }
    }
}
```

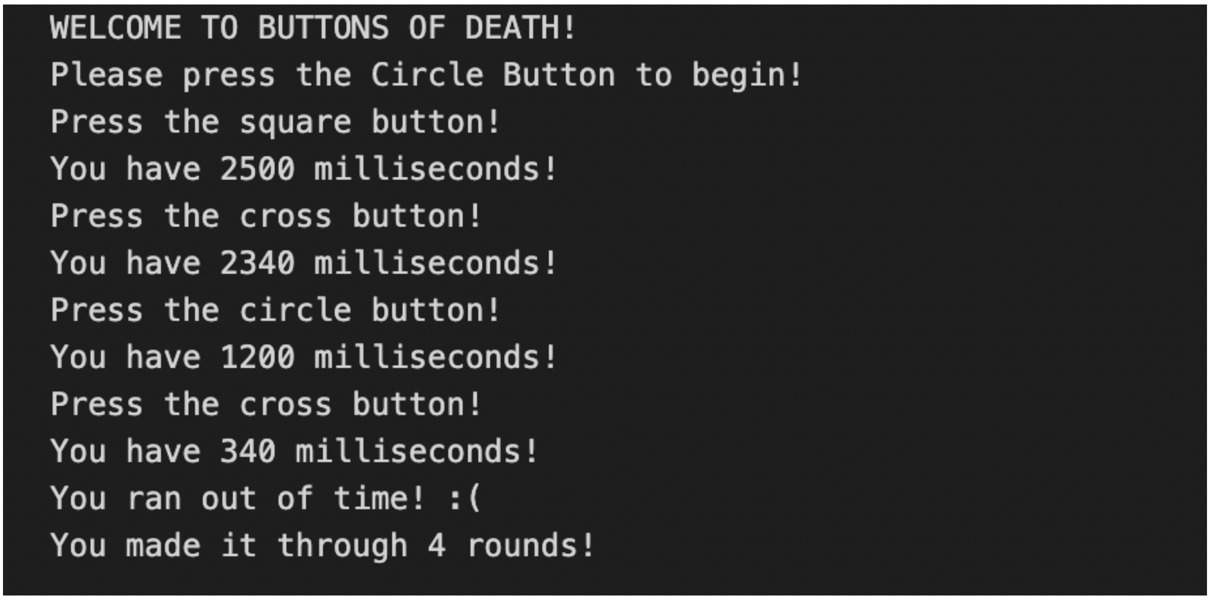


```c
void extras(void){

        int triangle, circle, x_button, square,a1, a2, a3, a4=0;
        int t =0;

        for(int i=0;i<100;i++)
        scanf("%d, %d,%d,%d,%d,%d,%d,%d,%d",&t, &triangle, &circle, &x_button, &square,&a1, &a2, &a3, &a4);

}
```

OUTPUT:



```
WELCOME TO BUTTONS OF DEATH!
Please press the Circle Button to begin!
Press the triangle button!
You have 2500 milliseconds!
Press the cross button!
You have 2340 milliseconds!
Wrong button!
You lose! :(
You made it through 1 rounds!
```



```
WELCOME TO BUTTONS OF DEATH!
Please press the Circle Button to begin!
Press the square button!
You have 2500 milliseconds!
Press the cross button!
You have 2340 milliseconds!
Press the circle button!
You have 1200 milliseconds!
Press the cross button!
You have 340 milliseconds!
You ran out of time! :(
You made it through 4 rounds!
```

Questions:

1. How did you randomize the buttons that needed to be pressed?

I used the rand function in and assigned any value from 1-4 to cases of switch. So, for example if we get 3 that will be case 3 and hence x button will be required.

2. What game states, if any, did you keep track of?

I kept track of the time as the game should end when the time runs out or a wrong button is pressed.

3. What mechanism did you use to make sure extraneous button presses were not registered?

Scanf statement and I used a function called extra() to make sure no extra button is pressed.