

```

#define RED 7

#define YELLOW 8

#define GREEN 9

#define BLUE 10

#define RPUSH 3

#define YPUSH 4

#define GPUSH 5

#define BPUSH 6

String difficulty;

int Time = 1000;

int Round[20];

int Level = 1;

int Index = 0;

int Score = 0;

const int timeDuration = 5000;

void setup() {

    pinMode(RED, OUTPUT);

    pinMode(YELLOW, OUTPUT);

    pinMode(GREEN, OUTPUT);

    pinMode(BLUE, OUTPUT);

    pinMode(RPUSH, INPUT_PULLUP);

    pinMode(YPUSH, INPUT_PULLUP);

    pinMode(GPUSH, INPUT_PULLUP);

    pinMode(BPUSH, INPUT_PULLUP);

    Serial.begin(9600);

    Serial.println("Enter difficulty level:\nEasy - Moderate - Hard ");

    while (true) {

        if (Serial.available() > 0) {

            difficulty = Serial.readString();

            difficulty.trim();

            if (difficulty == "Easy") {

                Time = 2500;

                break;

            } else if (difficulty == "Moderate") {

                Time = 1000;

```

```

        break;

    } else if (difficulty == "Hard") {

        Time = 500;

        break;

    } else {

        Serial.println("Invalid input. Please enter difficulty level:\nEasy - Moderate - Hard ");

    }

}

}

randomSeed(millis());

}

void loop() {

    Round[Index] = random(4);

    for (int i = 0; i < Level; i++) {

        ON(Round[i]);

        delay(Time);

        OFF();

        delay(1000);

    }

    Serial.println();

    for (int i = 0; i < Level; i++) {

        int playerInput = waitForInput();

        Serial.print("Player Input: ");

        Serial.println(playerInput);

        if (playerInput == -1 || playerInput != Round[i]) {

            Serial.print("Game Over! Score: ");

            Serial.println(Score);

            while (true);

        }

    }

    Index++;

    Score++;

    Level++;

}

void ON(int led) {

```

```
switch (led) {

  case 0:

    digitalWrite(REDA, HIGH);

    break;

  case 1:

    digitalWrite(YELLOW, HIGH);

    break;

  case 2:

    digitalWrite(GREEN, HIGH);

    break;

  case 3:

    digitalWrite(BLUE, HIGH);

    break;

}

}

void OFF() {

  digitalWrite(REDA, LOW);

  digitalWrite(YELLOW, LOW);

  digitalWrite(GREEN, LOW);

  digitalWrite(BLUE, LOW);

}

int waitForInput() {

  unsigned long startTime = millis();

  while (millis() - startTime < timeDuration) {

    if (digitalRead(RPUSH) == LOW) {

      delay(50);

      if (digitalRead(RPUSH) == LOW) {

        while (digitalRead(RPUSH) == LOW);

        Serial.println("Red Button Pressed");

        return 0;

      }

    } else if (digitalRead(YPUSH) == LOW) {

      delay(50);

      if (digitalRead(YPUSH) == LOW) {

        while (digitalRead(YPUSH) == LOW);

      }

    }

  }

}
```

```
    Serial.println("Yellow Button Pressed");  
    return 1;  
}  
} else if (digitalRead(GPUSH) == LOW) {  
    delay(50); // Debounce delay  
    if (digitalRead(GPUSH) == LOW) {  
        while (digitalRead(GPUSH) == LOW);  
        Serial.println("Green Button Pressed");  
        return 2;  
    }  
} else if (digitalRead(BPUSH) == LOW) {  
    delay(50); // Debounce delay  
    if (digitalRead(BPUSH) == LOW) {  
        while (digitalRead(BPUSH) == LOW);  
        Serial.println("Blue Button Pressed");  
        return 3;  
    }  
}  
}  
return -1;  
}
```