```
// Define LED pins for 4017 output
int led1 = 13;  // Red
int led2 = 12;  // Green
int led3 = 11;  // Yellow
int led4 = 10;  // Red
int led5 = 9;   // Green
int led6 = 8;   // Yellow

const int button = 4;     // Button pin
const int buzzer = 2;     // Buzzer pin

// Game variables
int score = 0;
long randColour;
int x = 0;
int initialMessage = 0;

void setup() {
  Serial.begin(9600);

  // Set LED pins as INPUT (controlled by 4017)
  pinMode(led1, INPUT);
  pinMode(led2, INPUT);
  pinMode(led3, INPUT);
  pinMode(led4, INPUT);
  pinMode(led5, INPUT);
  pinMode(led6, INPUT);

  pinMode(button, INPUT_PULLUP);  // Button (active LOW)
  pinMode(buzzer, OUTPUT);        // Buzzer output

  randomSeed(analogRead(0));  // Seed randomness

  // Display initial message once
  while (initialMessage == 0) {
    Serial.println("Welcome to the LED chaser game.");
    Serial.println("Click on the button on the color you are given by the computer to
score 1 point.");
    Serial.println("Click incorrectly, you lose 1 point.");
    Serial.println("WIN = 5 Points || LOSE = -5 Points");
    Serial.println("Change potentiometer value to make the game faster or slower.
Good Luck Player!!");
    initialMessage++;
    delay(2000);
```

```
  }
}

void loop() {
  int buttonState = digitalRead(button);

  if (buttonState == LOW) {
    randColour = random(1, 4);

    // Turn off all LEDs
    digitalWrite(led1, LOW);
    digitalWrite(led2, LOW);
    digitalWrite(led3, LOW);
    digitalWrite(led4, LOW);
    digitalWrite(led5, LOW);
    digitalWrite(led6, LOW);

    // Light up random color
    if (randColour == 1) {
      Serial.println("The color you need to stop on is Red");
      digitalWrite(led1, HIGH);
      digitalWrite(led4, HIGH);
    } else if (randColour == 2) {
      Serial.println("The color you need to stop on is Green");
      digitalWrite(led2, HIGH);
      digitalWrite(led5, HIGH);
    } else if (randColour == 3) {
      Serial.println("The color you need to stop on is Yellow");
      digitalWrite(led3, HIGH);
      digitalWrite(led6, HIGH);
    }

    delay(100);

    while (digitalRead(button) == LOW);  // Wait for release

    // Check player response
    if (buttonState == LOW) {
      if ((randColour == 1 && (digitalRead(led1) == HIGH || digitalRead(led4) ==
HIGH)) ||
          (randColour == 2 && (digitalRead(led2) == HIGH || digitalRead(led5) ==
HIGH)) ||
          (randColour == 3 && (digitalRead(led3) == HIGH || digitalRead(led6) ==
HIGH))) {
```

```
        score++;
        Serial.print("Correct! Your score is: ");
        Serial.println(score);
      } else {
        score--;
        Serial.print("Wrong! Your score is: ");
        Serial.println(score);
      }
    }

    //  WIN
    if (score == 5) {
      Serial.println("You win!");
      playVictoryTune();
      score = 0;
    }

    //  LOSE
    else if (score == -5) {
      Serial.println("You lose!!");
      playLoseTune();
      score = 0;
    }

    delay(500);
  }
}

// 🎶 Victory Tune
void playVictoryTune() {
  tone(buzzer, 523); delay(200);  // C5
  tone(buzzer, 659); delay(200);  // E5
  tone(buzzer, 783); delay(200);  // G5
  tone(buzzer, 1046); delay(300); // C6
  noTone(buzzer);
}

// 🎶 Lose Tune
void playLoseTune() {
  tone(buzzer, 349); delay(300);  // F4
  tone(buzzer, 330); delay(300);  // E4
  tone(buzzer, 311); delay(300);  // D#4
  tone(buzzer, 262); delay(500);  // C4
  noTone(buzzer);
```

```
}
```