Tinkercad link: https://www.tinkercad.com/things/lozV8434PFG-simon-v1

Code for making buttons light up like in Whack a Mole:

```
 const int redLED = 2;
const int blueLED = 3;
const int yellowLED = 4;
const int greenLED = 5;

const int redButton = 6;
const int blueButton = 7;
const int yellowButton = 8;
const int greenButton = 9;

int waitTime = 2000; //loop cycle for button press before timeout

// the setup function runs once when you press reset or power the board
void setup() {
 // initialize digital pin LED_BUILTIN as an output.
 pinMode(redLED, OUTPUT);
 pinMode(blueLED, OUTPUT);
 pinMode(yellowLED, OUTPUT);
 pinMode(greenLED, OUTPUT);

 //use "PULLUP" to set open/unpressed to HIGH
 pinMode(redButton, INPUT_PULLUP);
 pinMode(blueButton, INPUT_PULLUP);
 pinMode(yellowButton, INPUT_PULLUP);
 pinMode(greenButton, INPUT_PULLUP);
  Serial.begin(9600);      // open the serial port at 9600 bps:
 randomSeed(analogRead(0));
}

// the loop function runs over and over again forever
void loop() {
 //pick a new color
 int newColor = pickNewColor();

 //show the color
 ledOn(newColor);

 //check to see if whacked or not
 if (isWhacked(newColor)) {
```

```
    Serial.println("You whacked the mole!");
    ledOff(newColor);
    delay(500); //pause briefly before showing a new mole
    waitTime = 0.9*waitTime;
  } else {
    Serial.println("You failed to whack the mole! Game over.");
    gameover();
    waitTime = 2000;
  }
  // challenge - how to speed up the game at next mole? (and reset after game over?)
}

//^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
boolean isWhacked(int newColor) {
 int i = 0;
 int chkButton;
 boolean whacked = false;
 boolean buttonPressed = false;

 //start the counter and wait for botton to be pressed..or timeout
 Serial.print("Wait Time: ");
 Serial.print(waitTime);
 while ((i < waitTime) and (!buttonPressed)) {

    chkButton = isButtonPressed2(); //0 if nothing is pressed
    if (newColor == chkButton) { //correct button is pressed
      whacked = true;
      buttonPressed = true;
    } else if (chkButton > 0) { //incorrect button is pressed
      whacked = false;
      buttonPressed = true;
    }



    i++;
    //Serial.println("."); //Comment out later. This slows down the loop
    delay(1); //make each loop about 1ms
 }
 if (buttonPressed) {
    Serial.println("..button detected.");
 } else {
```

```
   Serial.println("..time out");
 }

 return whacked;
}

//function that monitor all buttons and returns a integer
// 0 = nothing pressed
// 2-5 = button prssed
// variation #2 -
// do not turn off led if button is NOT pressed
int isButtonPressed2() {
    int buttonPressed = 0;
    //2 = red, 3 = blue, 4 = yellow, 5=green
   //Set to LOW when a button is pressed

   if (digitalRead(redButton) == LOW) {
     ledOn(redLED);
     buttonPressed = redLED;
   } /*else {
     ledOff(redLED);;
   }*/
    if (digitalRead(blueButton) == LOW) {
     ledOn(blueLED);
     buttonPressed = blueLED;
   } /*else {
     ledOff(blueLED);
   } */
    if (digitalRead(yellowButton) == LOW) {
     ledOn(yellowLED);
     buttonPressed = yellowLED;
   } /* else {
     ledOff(yellowLED);
   } */
    if (digitalRead(greenButton) == LOW) {
     ledOn(greenLED);
     buttonPressed = greenLED;
   } /*else {
     ledOff(greenLED);
   } */

   return buttonPressed;
```

```
}

void gameover() {
   for (int i=0; i <= 2; i++){
    ledOn(redLED);
    ledOn(blueLED);
    ledOn(yellowLED);
    ledOn(greenLED);
    delay(200);
    ledOff(redLED);
    ledOff(blueLED);
    ledOff(yellowLED);
    ledOff(greenLED);
    delay(200);
  }
 delay(2000);
}
//^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^


//function that monitor all buttons and returns a integer
// 0 = nothing pressed
// 2-5 = button prssed
int isButtonPressed() {
    int buttonPressed = 0;
    //2 = red, 3 = blue, 4 = yellow, 5=green
   //Set to LOW when a button is pressed

   if (digitalRead(redButton) == LOW) {
     ledOn(redLED);
     buttonPressed = redLED;
   } else {
     ledOff(redLED);;
   }
    if (digitalRead(blueButton) == LOW) {
     ledOn(blueLED);
     buttonPressed = blueLED;
   } else {
     ledOff(blueLED);
   }
    if (digitalRead(yellowButton) == LOW) {
     ledOn(yellowLED);
```

```
      buttonPressed = yellowLED;
    } else {
      ledOff(yellowLED);
    }
     if (digitalRead(greenButton) == LOW) {
      ledOn(greenLED);
      buttonPressed = greenLED;
    } else {
      ledOff(greenLED);
    }

    return buttonPressed;
}

//function to randomly pick a new color
int pickNewColor () {
  int randomColor;
 //random(min, max)
 //Parameters
 //min - lower bound of the random value, inclusive (optional)
 //max - upper bound of the random value, exclusive
  //generate random # from 2 to 5 that matches led Pins
 //2 = red, 3 = blue, 4 = yellow, 5=green
  randomColor = random(2,6);
 return randomColor;

}

void boomerang(int speed) {
 chaseL2R(speed);
 chaseR2L(speed);
}

void chaseL2R(int speed) {
 blink(redLED,speed);
 blink(blueLED,speed);
 blink(yellowLED,speed);
 blink(greenLED,speed);
}
 void chaseR2L(int speed) {
 blink(greenLED,speed);
 blink(yellowLED,speed);
```

```
 blink(blueLED,speed);
 blink(redLED,speed);
}

void blink(int color, int blinkTime) {
 ledOn(color);
 delay(blinkTime);
 ledOff(color);
 delay(blinkTime);
}

void ledOffAll() {

 digitalWrite(redLED, LOW);
 digitalWrite(blueLED, LOW);
 digitalWrite(yellowLED, LOW);
 digitalWrite(greenLED, LOW);
}

void ledOn(int colorON) {

 if (colorON == redLED) {
   digitalWrite(redLED, HIGH);
 }
 else if (colorON == blueLED) {
   digitalWrite(blueLED, HIGH);
 }
 else if (colorON == yellowLED) {
   digitalWrite(yellowLED, HIGH);
 }
 else if (colorON == greenLED) {
   digitalWrite(greenLED, HIGH);
 }

}

void ledOff(int colorOFF) {

 if (colorOFF == redLED) {
   digitalWrite(redLED, LOW);
 }
 else if (colorOFF == blueLED) {
```

```
    digitalWrite(blueLED, LOW);
  }
  else if (colorOFF == yellowLED) {
    digitalWrite(yellowLED, LOW);
  }
  else if (colorOFF == greenLED) {
    digitalWrite(greenLED, LOW);
  }

}
```