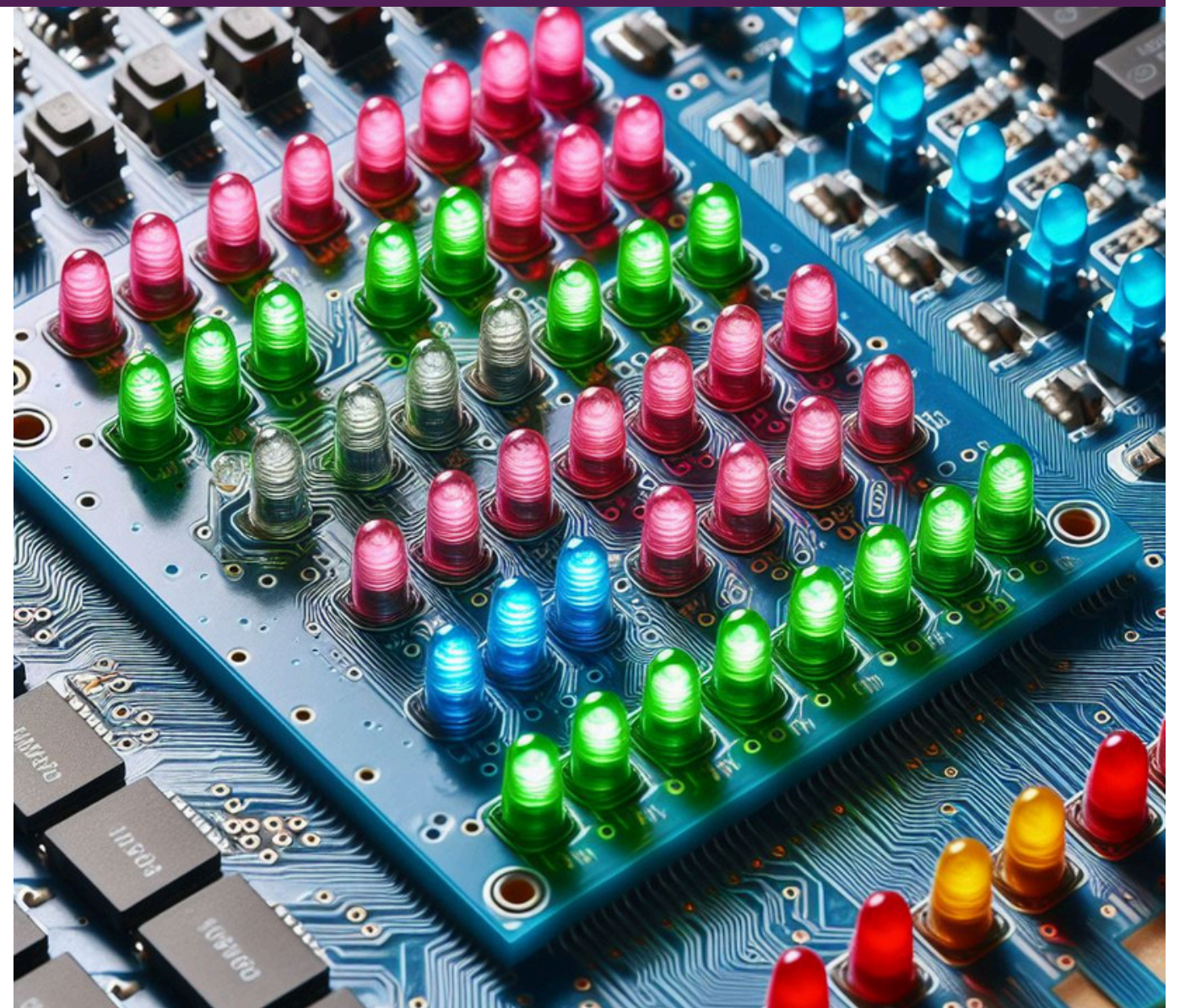


# Arduino Jogo Genius (Jogo da memória)

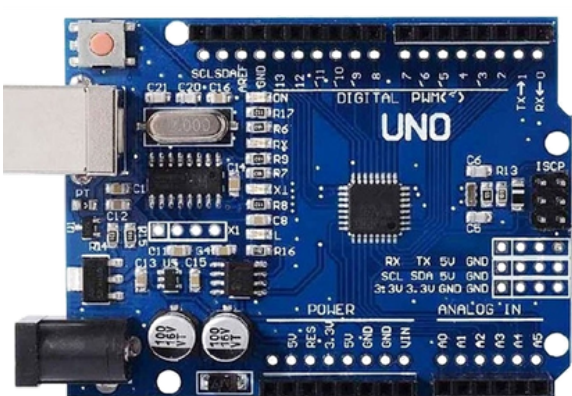


## Objetivos

- 🔧 Desenvolvimento de habilidades de programação
- 🔧 Introduzir conceitos da eletrônica básica
- 🔧 Estímulo ao raciocínio lógico e memória

### MATERIAIS NECESSARIOS

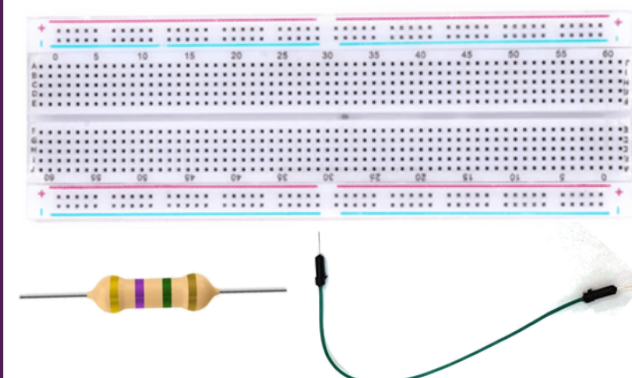
1 Arduino Uno R3



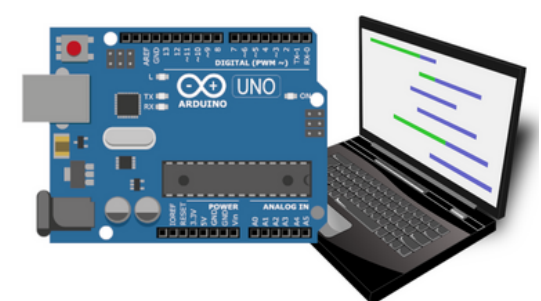
LED e botões



Protoboard, cabos e Resistências



Computador/ Código



## RESUMO

Neste projeto vamos apresentar o "Jogo da Memória" também chamado de "Genius", jogo de grande sucesso na década de 80. O jogo apresenta 4 botões e 4 leds que acenderão em uma sequência aleatória, com sons definidos para cada um dos leds. O jogador deverá clicar nos botões de acordo com a sequência gerada pelo Arduino. A cada jogada, a sequência é acrescida de um novo item. Ganha quem conseguir repetir a maior sequência possível. O jogo é limitado a 100 jogadas

### REALIZADORES

Rodrigo, Kaique, Yan, Mariana e Gustavo

### APOIO

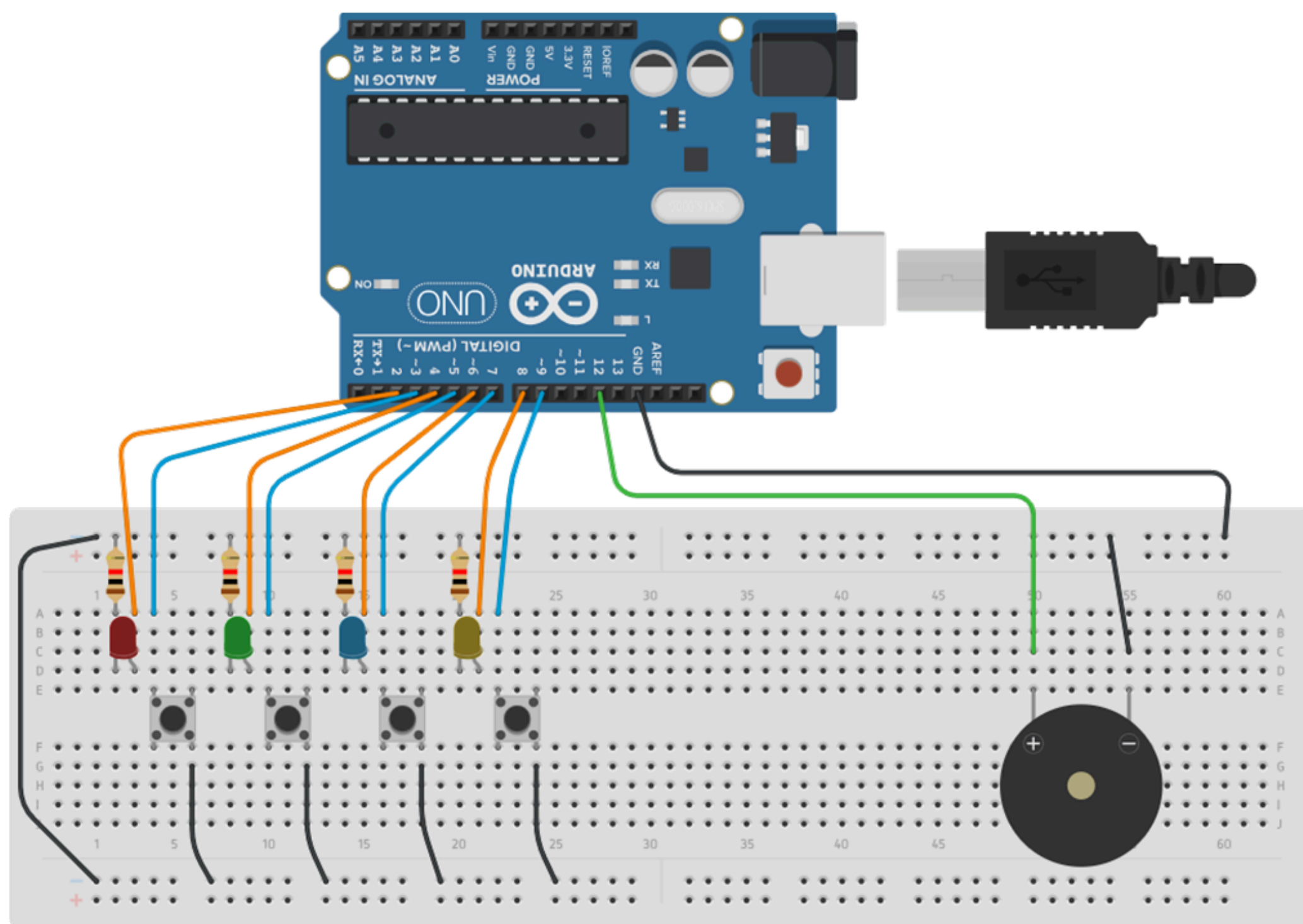
# Jogo Genius (Jogo da memória)

## COMO FUNCIONA O PROJETO

- Ao executar o programa, todos os leds acenderão e é dado um sinal sonoro.
- Aleatoriamente, acenderá o primeiro led entre os 4 leds do jogo. O programa irá aguardar que o jogador clique no botão correspondente.
- Cada led aceso tem um som correspondente.
- Se o jogador acertar o botão, repete-se a sequência com o acréscimo aleatório do segundo led.
- Se o jogador errar, acenderá um sinal sonoro indicando o fim do jogo. Novo led se acenderá....
- Ganha o jogador que acertar mais jogadas, ou seja, memorizar a maior sequência de sons.

## MONTAGEM DO CIRCUITO

Conecte os componentes no Protoboard como mostra a figura abaixo. Verifique cuidadosamente os cabos de ligação antes de ligar seu Arduino. Lembre-se que o Arduino deve estar desconectado da força enquanto você monta o circuito.



### REALIZADORES

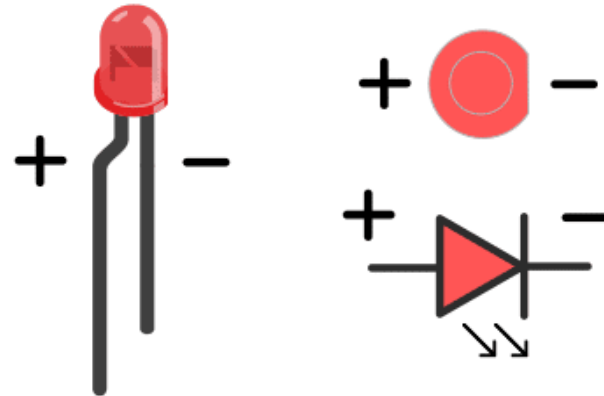
Rodrigo, Kaique, Yan, Mariana e Gustavo

### APOIO

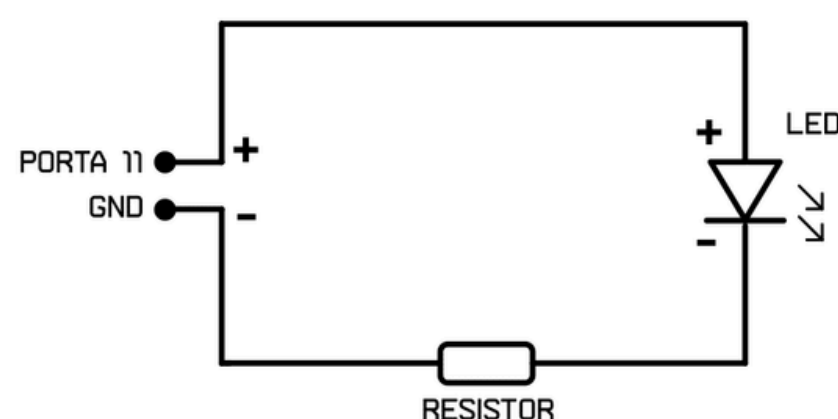
# Jogo Genius (Jogo da memória)

## ATENÇÃO

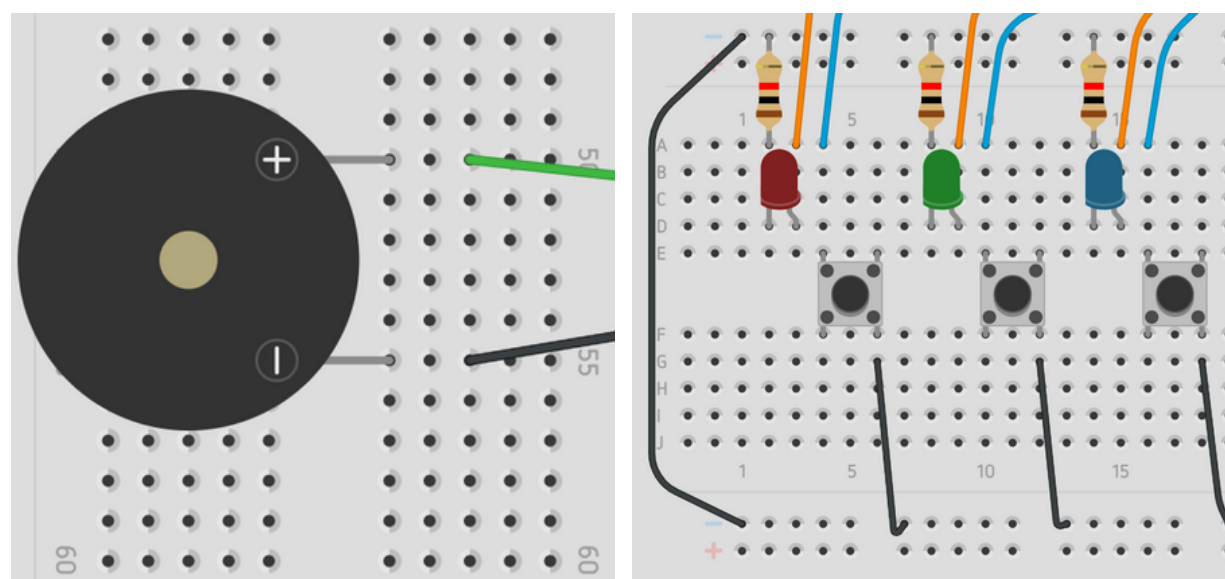
- Lembre-se que o LED tem polaridade: O terminal maior tem polaridade positiva e o lado do chanfro tem polaridade negativa.



- Utilizando um resistor reduzimos a corrente do circuito para trabalhar nos valores que o LED suporta. Sem o resistor, a corrente que passa pelo LED seria muito grande e acabaria queimando ele.



- O buzzer tem polaridade. Portanto, cuidado para não ligar o buzzer invertido. Se você retirar o adesivo superior do buzzer poderá ver um sinal de positivo (+). Este sinal mostra onde está o pino positivo do componente que deverá estar conectado a uma porta digital do Arduino e o polo negativo ao GND.



- Monte o botão (push button) sem o resistor, pois através da programação vamos habilitar o resistor pull-up interno do arduino. Desta forma, quando o botão estiver pressionado, o Arduino retornará "LOW".

## RÊFERENCIA

Adaptado de:

<https://meetarduino.wordpress.com/2012/05/27/arduino-genius-jogo-da-memoria/>

### REALIZADORES

Rodrigo, Kaique, Yan, Mariana e Gustavo

### APOIO

# Jogo Genius (Jogo da memória)

## CÓDIGO

```

#define NOTE_D4 294
#define NOTE_G4 392
#define NOTE_A4 440
#define NOTE_A5 880

int tons[4] = { NOTE_A5, NOTE_A4,
NOTE_G4, NOTE_D4 };
int sequencia[100] = {};
int rodada_atual = 0;
int passo_atual_na_sequencia = 0;
int pinoAudio = 12;
int pinosLeds[4] = { 2, 4, 6, 8 };
int pinosBotoes[4] = { 3, 5, 7, 9 };
int botao_pressionado = 0;
int perdeu_o_jogo = false;

void setup() {

for (int i = 0; i <= 3; i++) {
pinMode(pinosLeds[i], OUTPUT);
}

for (int i = 0; i <= 3; i++) {
pinMode(pinosBotoes[i], INPUT_PULLUP);
}

pinMode(pinoAudio, OUTPUT);
randomSeed(analogRead(0));
}

void loop() {

if (perdeu_o_jogo) {
int sequencia[100] = {};
rodada_atual = 0;
passo_atual_na_sequencia = 0;
perdeu_o_jogo = false;
}

if (rodada_atual == 0) {
tocarSomDeInicio();
delay(500);
}

proximaRodada();
reproduzirSequencia();
aguardarJogador();

delay(1000);
}

void proximaRodada() {
int numero_sorteado = random(0, 4);
sequencia[rodada_atual++] =
numero_sorteado;
}

void reproduzirSequencia() {
for (int i = 0; i < rodada_atual; i++) {
tone(pinoAudio, tons[sequencia[i]]);
digitalWrite(pinosLeds[sequencia[i]],
HIGH);
delay(500);
noTone(pinoAudio);
digitalWrite(pinosLeds[sequencia[i]], LOW);
delay(100);
}
noTone(pinoAudio);
}

void aguardarJogador() {
for (int i = 0; i < rodada_atual; i++) {
aguardarJogada();

if (sequencia[passo_atual_na_sequencia] !=
botao_pressionado) {
gameOver();
}

if (perdeu_o_jogo) {
break;
}

passo_atual_na_sequencia++;
}

passo_atual_na_sequencia = 0;
}

void aguardarJogada() {
boolean jogada_efetuada = false;
while (!jogada_efetuada) {
for (int i = 0; i <= 3; i++) {
if (!digitalRead(pinosBotoes[i])) {
// Dizendo qual foi o botao pressionado.
botao_pressionado = i;

tone(pinoAudio, tons[i]);
digitalWrite(pinosLeds[i], HIGH);
delay(300);
digitalWrite(pinosLeds[i], LOW);
noTone(pinoAudio);

jogada_efetuada = true;
}
}
}
}

void gameOver() {
// GAME OVER.
for (int i = 0; i <= 3; i++) {
tone(pinoAudio, tons[i]);
digitalWrite(pinosLeds[i], HIGH);
delay(200);
digitalWrite(pinosLeds[i], LOW);
noTone(pinoAudio);
}

tone(pinoAudio, tons[3]);
for (int i = 0; i <= 3; i++) {
digitalWrite(pinosLeds[0], HIGH);
digitalWrite(pinosLeds[1], HIGH);
digitalWrite(pinosLeds[2], HIGH);
digitalWrite(pinosLeds[3], HIGH);
delay(100);
digitalWrite(pinosLeds[0], LOW);
digitalWrite(pinosLeds[1], LOW);
digitalWrite(pinosLeds[2], LOW);
digitalWrite(pinosLeds[3], LOW);
delay(100);
}

noTone(pinoAudio);

perdeu_o_jogo = true;
}

void tocarSomDeInicio() {
tone(pinoAudio, tons[0]);
digitalWrite(pinosLeds[0], HIGH);
digitalWrite(pinosLeds[1], HIGH);
digitalWrite(pinosLeds[2], HIGH);
digitalWrite(pinosLeds[3], HIGH);
delay(500);
digitalWrite(pinosLeds[0], LOW);
digitalWrite(pinosLeds[1], LOW);
digitalWrite(pinosLeds[2], LOW);
digitalWrite(pinosLeds[3], LOW);
delay(500);
noTone(pinoAudio);
}

```

## REALIZADORES

Rodrigo, Kaique, Yan, Mariana e  
Gustavo

## APOIO