**Jogo de Memória em C com Raylib**

**Descrição do Projeto**

Este é um jogo de memória desenvolvido em C utilizando a biblioteca Raylib para gráficos. O objetivo do jogo é memorizar a posição de quadrados azuis exibidos brevemente em uma grade e depois clicar nas posições corretas. O jogo possui sistema de cadastro de jogadores, ranking de pontuações e fases progressivamente mais desafiadoras.

**Autores:** Felipe Gonçalves, Guilherme Barretto e Geovanna Sargaço  
**Data:** 06/2025  
**Versão:** 0.1  
**Licença:** GPL

**Características Principais**

* Interface gráfica com Raylib
* Sistema de cadastro de jogadores
* Múltiplas fases com dificuldade progressiva
* Sistema de pontuação e ranking
* Temporizador visual para memorização
* Feedback visual dos acertos e erros

**Estrutura do Código**

**Bibliotecas Utilizadas**

c

#include <raylib.h> // Gráficos e interface

#include <stdio.h> // Entrada/saída

#include <stdlib.h> // Funções gerais

#include <time.h> // Tempo e aleatoriedade

#include <string.h> // Manipulação de strings

**Estrutura de Dados**

c

typedef struct registo\_de\_jogadores {

char nick[10]; // Nome do jogador (max 10 caracteres)

int pontos; // Pontuação acumulada

int tempo; // Tempo de jogo

int fase; // Fase atual

} registro;

**Variáveis Globais**

c

registro jogador[4]; // Vetor de jogadores (max 4)

int contador\_de\_jogadores = 0; // Contador de jogadores

int indice\_do\_jogador = 0; // Índice do jogador atual

const int largura\_da\_tela = 500; // Dimensões da janela

const int altura\_da\_tela = 500;

int cliques = 0; // Contador de cliques

int estado\_do\_jogo = 0; // Estado atual do jogo

**Fluxo do Jogo**

**1. Tela de Início (**tela\_de\_inicio**)**

* Exibe mensagem de boas-vindas
* Aguarda clique do mouse para iniciar

**2. Cadastro do Jogador (**tela\_de\_cadastro**)**

* Solicita nome do jogador (3-10 caracteres)
* Valida entrada e armazena no registro

**3. Jogo Principal (**tela\_do\_tabuleiro**)**

* **Fase de Memorização:** Exibe quadrados azuis por 5 segundos
* **Fase de Interação:** Jogador clica nos quadrados memorizados
* **Fase de Resultado:** Mostra acertos (verde), erros (vermelho) e omissões (azul)

**4. Sistema de Pontuação**

* +100 pontos por acerto
* Avança para próxima fase ao acertar todas as posições
* Reinicia ou mostra ranking em caso de erro

**5. Ranking (**mostrar\_ranking**)**

* Exibe os 4 melhores jogadores ordenados por pontuação
* Opção para reiniciar o jogo

**Funções Principais**

**Controle do Jogo**

c

void execucao\_do\_jogo() // Controla o fluxo principal

void tela\_de\_inicio() // Tela inicial

void tela\_de\_cadastro() // Cadastro de jogador

int tela\_do\_tabuleiro(int fase)// Jogo principal

void mostrar\_ranking() // Exibe ranking

**Lógica do Jogo**

c

void aleatorizar\_tabuleiro() // Gera posições aleatórias

void calcular\_pontos() // Calcula pontuação

int definir\_tamanho() // Define tamanho do tabuleiro

int definir\_jogador() // Seleciona jogador

**Interface Gráfica**

c

void desenhar\_gabarito() // Desenha tabuleiro de memorização

void desenhar\_interacao() // Desenha tabuleiro interativo

void desenhar\_resultado() // Desenha resultados

void desenhar\_contador() // Desenha temporizador

**Utilitários**

c

int gerador\_de\_numeros() // Gera números aleatórios

void limpar\_matriz() // Zera matriz

void temporizador() // Controla tempo

**Sistema de Fases**

**Progressão de Dificuldade**

* **Fases 1-5:** Tabuleiro 4x4
* **Fases 6-10:** Tabuleiro 5x5
* **Fases 11+:** Tabuleiro 6x6

**Mecânica de Pontuação**

* Cada fase exige memorizar mais quadrados
* +100 pontos por acerto  
  -必须 acertar todos os quadrados para avançar

**Compilação e Execução**

**Requisitos**

* Biblioteca Raylib instalada
* Compilador C (GCC recomendado)

**Comando de Compilação**

bash

gcc -o jogo\_memoria main.c -lraylib -lm -lpthread -ldl

**Execução**

bash

./jogo\_memoria

**Recursos Gráficos**

* Ícone personalizado da janela (icon.png)
* Fundo da tela de cadastro (fundoInicio.png)
* Cores distintas para diferentes estados:
  + Azul: Quadrados para memorizar
  + Verde: Acertos
  + Vermelho: Erros
  + Cinza: Quadrados neutros

**Controles**

* **Mouse Esquerdo:** Selecionar quadrado
* **Mouse Direito:** Desselecionar quadrado
* **Teclado:** Digitar nome durante cadastro
* **Enter:** Confirmar nome
* **Backspace:** Corrigir nome

**Possíveis Melhorias**

1. Adicionar efeitos sonoros
2. Implementar sistema de salvamento de scores
3. Adicionar mais efeitos visuais
4. Criar fases temáticas
5. Implementar modo multiplayer
6. Adicionar dificuldades personalizadas

**Observações Técnicas**

* Uso de matrizes dinâmicas VLA (Variable Length Arrays)
* Gerenciamento manual de memória para texturas
* Sistema de estados para controle de fluxo
* Ponteiros para passagem de parâmetros por referência