**Jogo Da Velha 3x3**

**Integrantes do Grupo:**

Erick de Lucena Chianca  
Isaac Adryan Cartaxo Rodrigues Menezes  
Matheus Ramalho de Freitas Crispim  
Ryan de Souza Sebadelhe

**Introdução ao Jogo e suas Regras:**

O jogo da velha consiste em um tabuleiro 3x3 onde dois jogadores se enfrentam para tentar formar uma fileira de 3 marcações, cada jogador vai ter sua marca, sendo o jogador 1 o "X" e jogador 2 o "O". Quem conseguir colocar três marcas em uma mesma linha, coluna ou diagonal é o vencedor.

**As regras são:**

1. Somente o jogador da vez pode fazer a jogada, isto é, o outro jogador deve esperar seu oponente jogar para fazer sua escolha
2. As posições só podem ser escolhidas caso ela esteja vazia
3. Uma partida na qual, todas as posições estão preenchidas e não há um vencedor, significa empate, o famoso "Deu Velha"

**Descrição Geral:**

O código começa com um Struct, que é uma estrutura de dados que armazena, o nome, pontuação e símbolo. Essa estrutura recebe duas instancias, jogador1 e jogador2. Seus dados vão ser armazenados ao decorrer do código.  
O programa começa perguntando o nome dos dois jogadores. O símbolo X e O são atribuídos respectivamente aos jogadores 1 e 2, as pontuações são armazenadas quando alguém vence uma partida.  
Após isso aparece um menu, onde será escolhido entre:

1- Jogar  
2- Ver Ranking (pontuação dos jogadores)  
3- Créditos do Projeto  
4- Sair

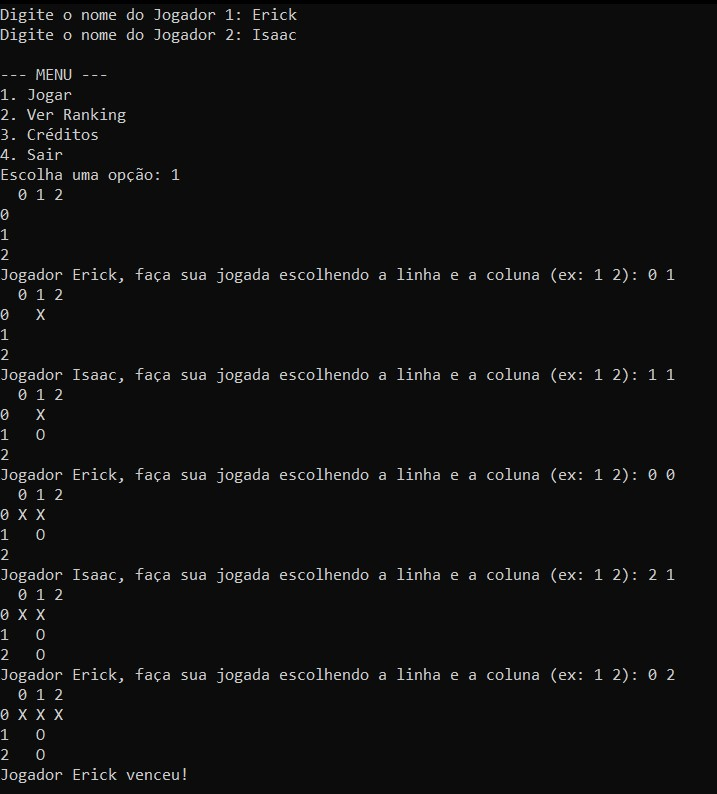
Esse menu vai aparecer pelo menos uma vez, e enquanto o número escolhido não for 4 (Sair), o código continuará rodando de acordo com o número que foi escolhido.

Em caso de 1:   
1- O código vai chamar o void mostrarTabuleiro (), que mostra o estado atual do tabuleiro. A esse ponto, outro void já havia sido chamado, o void inicializarTabuleiro (), que preenche todos as posições do tabuleiro com espaços em branco.

2- Um printf vai pedir uma posição, um scanf vai receber essa posição e vai verificar se há espaço vazio, se estiver preenchido, vai ser necessário colocar uma nova posição.

3- Após cada jogado, será chamado o void verificarVitoria (), que verifica todas as linhas, colunas e diagonais do tabuleiro.

4- Em caso de vitória, o código mostra quem ganhou, e vai contabilizar os pontos para o jogador.  
Se não houver vencedor e o tabuleiro ainda tiver espaço em branco, o jogo continua.  
Se não houver vencedor e o tabuleiro estiver todo preenchido, deu velha.

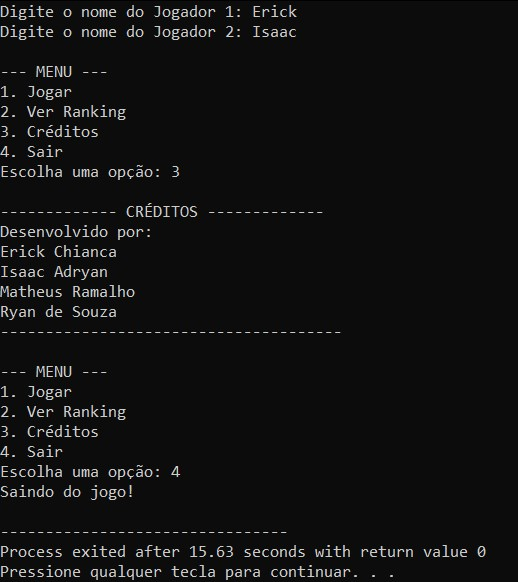


Em caso de 2:

O void mostrarRanking () é chamado:  
Nele há o nome e a pontuação de cada jogador

Em caso de 3:  
Os créditos são mostrados

Em caso de 4:   
O código é finalizado



**Dificuldades enfrentadas:**A princípio, a sintaxe da linguagem C foi o maior problema, a lógica em si não foi o maior dos problemas pois já tínhamos experiência com um código parecido, porem em outra linguagem.

A maior dificuldade foi provavelmente na função de iniciar o tabuleiro, fazer esse tabuleiro na linguagem C é bem complicado.

**Código:**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <locale.h>

struct Jogador {

char nome[50];

char simbolo;

int pontuacao;

};

struct Jogador jogador1, jogador2;

char \*\*tabuleiro;

void inicializarTabuleiro() {

tabuleiro = (char \*\*)malloc(3 \* sizeof(char \*));

for (int i = 0; i < 3; ++i) {

tabuleiro[i] = (char \*)malloc(3 \* sizeof(char));

for (int j = 0; j < 3; ++j) {

tabuleiro[i][j] = ' ';

}

}

}

void preencherJogadores() {

printf("Digite o nome do Jogador 1: ");

scanf("%s", jogador1.nome);

jogador1.simbolo = 'X';

jogador1.pontuacao = 0;

printf("Digite o nome do Jogador 2: ");

scanf("%s", jogador2.nome);

jogador2.simbolo = 'O';

jogador2.pontuacao = 0;

}

void mostrarTabuleiro() {

printf(" 0 1 2\n");

for (int i = 0; i < 3; ++i) {

printf("%d ", i);

for (int j = 0; j < 3; ++j) {

printf("%c ", tabuleiro[i][j]);

}

printf("\n");

}

}

int verificarVitoria() {

for (int i = 0; i < 3; ++i) {

if (tabuleiro[i][0] == tabuleiro[i][1] && tabuleiro[i][1] == tabuleiro[i][2] && tabuleiro[i][0] != ' ') {

return 1;

}

if (tabuleiro[0][i] == tabuleiro[1][i] && tabuleiro[1][i] == tabuleiro[2][i] && tabuleiro[0][i] != ' ') {

return 1;

}

}

if (tabuleiro[0][0] == tabuleiro[1][1] && tabuleiro[1][1] == tabuleiro[2][2] && tabuleiro[0][0] != ' ') {

return 1;

}

if (tabuleiro[0][2] == tabuleiro[1][1] && tabuleiro[1][1] == tabuleiro[2][0] && tabuleiro[0][2] != ' ') {

return 1;

}

for (int i = 0; i < 3; ++i) {

for (int j = 0; j < 3; ++j) {

if (tabuleiro[i][j] == ' ') {

return 0;

}

}

}

return -1;

}

void mostrarRanking() {

printf("\n--- RANKING ---\n");

printf("%s: %d ponto(s)\n", jogador1.nome, jogador1.pontuacao);

printf("%s: %d ponto(s)\n", jogador2.nome, jogador2.pontuacao);

printf("---------------\n");

}

int main() {

setlocale(0, "Portuguese");

int jogadorAtual = 1;

int linha, coluna;

inicializarTabuleiro();

preencherJogadores();

int opcao;

do {

printf("\n--- MENU ---\n");

printf("1. Jogar\n");

printf("2. Ver Ranking\n");

printf("3. Créditos\n");

printf("4. Sair\n");

printf("Escolha uma opção: ");

scanf("%d", &opcao);

switch (opcao) {

case 1:

do {

mostrarTabuleiro();

printf("Jogador %s, faça sua jogada escolhendo a linha e a coluna (ex: 1 2): ", (jogadorAtual == 1) ? jogador1.nome : jogador2.nome);

scanf("%d %d", &linha, &coluna);

if (linha < 0 || linha >= 3 || coluna < 0 || coluna >= 3 || tabuleiro[linha][coluna] != ' ') {

printf("Esse local está ocupado ou não existe. Por favor, tente de novo.\n");

continue;

}

tabuleiro[linha][coluna] = (jogadorAtual == 1) ? 'X' : 'O';

int result = verificarVitoria();

if (result == 1) {

mostrarTabuleiro();

printf("Jogador %s venceu!\n", (jogadorAtual == 1) ? jogador1.nome : jogador2.nome);

if (jogadorAtual == 1) {

jogador1.pontuacao++;

} else {

jogador2.pontuacao++;

}

inicializarTabuleiro();

break;

} else if (result == -1) {

mostrarTabuleiro();

printf("Deu Velha :(\n");

inicializarTabuleiro();

break;

}

jogadorAtual = (jogadorAtual == 1) ? 2 : 1;

} while (1);

break;

case 2:

mostrarRanking();

break;

case 3:

printf("\n------------- CRÉDITOS -------------\n");

printf("Desenvolvido por Erick, Isaac, Matheus e Ryan\n");

printf("--------------------------------------\n");

break;

case 4:

printf("Saindo do jogo!\n");

break;

default:

printf("Opção inválida.\n");

}

} while (opcao != 4);

return 0;

}