UNIVERSIDADE FEDERAL DO AMAZONAS-UFAM

INSTITUTO DE CIÊNCIAS EXATAS E TECNOLOGIA-ICET

CURSO ENGENHARIA DE SOFTWARE

WALTER JONAS DE SOUSA VIANA

INTRODUÇÃO A PROGRAMAÇÃO ESTRUTURADA 2

ITACOATIARA

2020

WALTER JONAS DE SOUSA VIANA

INTRODUÇÃO A PROGRAMAÇÃO ESTRUTURADA 2

TRABALHO PARA OBTENÇÃO DE NOTA PARCIAL NA DISCIPLINA DE INTRODUÇÃO A PROGRAMAÇÃO ESTRUTURADA 2 DO CURSO BACHAREL EM ENGENHARIA DE SOFTWARE DA UNIVERSIDADE FEDERAL DO AMAZONAS-UFAM.

PROFESSOR: ALTERNEI BRITO

ITACOATIARA

2020

Introdução

O presente trabalho consiste em um jogo da velha programado em C, utilizando conceitos básicos de condição, repetição e outras ferramentas da linguagem, incluindo as estruturas de dados, matrizes e funções.

Objetivo

Estimular a prática da programação na linguagem C utilizando os conceitos e técnicas vistos nas aulas de programação.

Detalhes da Implementação

O algoritmo tem as seguintes funções:

**Menu():** Onde terá as opções para jogar e sair do programa.

**Zerar():** Toda vez que uma partida terminar essa função será chamada, o tabuleiro deve zerar para uma nova partida começar.

**Exibe():** Essa função irá exibir o tabuleiro em cada jogada na tela, o tabuleiro exibido é semelhante a uma matriz 3x3.

**Jogar():** Essa função é chamada quando o usuário escolhe a opção de jogar, dentro desta função será chamada a função zerar(), pois para uma nova partida começar, o tabuleiro deve estar zerado, sem nenhum símbolo nas posições. Na função Jogar(), as funções Exibe() e Jogada() serão repetidos até a função ChecaTermino() retornar o valor “1” (quando o algoritmo encerrar a partida).

**ChecaLocal():** Essa função será chamada ao decorrer de uma jogada, ou seja, quando um jogador tentar inserir um valor (um símbolo) no tabuleiro, essa função irá verificar se a posição escolhida pelo jogador é válida.

**ChechaLinha():** Essa função irá verificar se o jogador enfileirou seus símbolos em uma **linha**, essa função será chamada dentro da função ChecaTermino().

**ChecaColuna():** Essa função irá verificar se o jogador enfileirou seus símbolos em uma **coluna**, essa função será chamada dentro da função ChecaTermino().

**ChecaDiagonal():** Essa função irá verificar se o jogador enfileirou seus símbolos na **diagonal**, essa função será chamada dentro da função ChecaTermino().

**ChecaEmpate():** Essa função irá verificar se a partida resultou em um **empate**, essa função será chamada dentro da função ChecaTermino().

**ChecaTermino():** Essa função está dentro de uma condição na função Jogar(), ela será chamada em cada jogada, para verificar se o jogo terminou ou vai continuar, as funções para checar as linhas, colunas, diagonais e os empates estão dentro dessa função e serão checados também.

**Jogada():** É a função que entra em ação no momento da jogada, nela o programa ler a linha e a coluna digitada pelo jogador se referindo a posição que deseja inserir seu símbolo, dentro dessa função será chamada o ChecaLocal() para verificar se a posição é válida.

Variáveis principais:

**Int vez:** Para controlar as jogadas dos jogadores.

**Int quant\_partidas:** Para contar quantas partidas foram jogadas.

**Int vitoria\_jog1:** Para contar as vezes que o jogador 1 venceu. Será incrementado +1 vitória toda vez que a seguinte condição for verdadeira: if ((vez%2)+1)==1). Essa condição é executada toda vez que a função checaTermino() é chamada.

**Int vitoria\_jog2:** Para contar as vezes que o jogador 2 venceu. Será incrementado +1 vitória toda vez que a seguinte condição for verdadeira: if ((vez%2)+1)==2). Essa condição é executada toda vez que a função checaTermino() é chamada.

**Float porcentagem1:** Para armazenar o cálculo da porcentagem de vitórias do jogador 1.

**Float porcentagem2:** Para armazenar o cálculo da porcentagem de vitórias do jogador 2.

**Int l, c:** Variáveis para ler o valor da posição da matriz onde o símbolo será inserido.

Constante:

**#define tam 3**.

Matriz:

**Tabuleiro [tam] [tam**]. Essa matriz será utilizada em todas as partes do algoritmo.

Struct:

**Typedef struct jogad**

**{  
 int l;**

**int c;**

**}**

Ideias gerais da aplicação

O jogo da velha é um passatempo com intenção de fazer as pessoas se distraírem em momentos estressantes. É jogado por dois jogadores utilizando um tabuleiro, seja ele real ou virtual, esse tabuleiro equivale a uma matriz 3 por 3. Cada jogador deve escolher um símbolo, os mais comuns são “X” e “O”. Em Cada rodada, cada jogador deve inserir seu símbolo apenas uma vez no tabuleiro, o jogador vence quando existir uma fileira completa do seu símbolo, seja na vertical, horizontal ou na diagonal principal. Existem várias estratégias para fazer uma fileira antes do seu oponente.

O algoritmo do jogo da velha foi feito utilizando algumas funções especificas para uma determinada funcionalidade, como exibir tabuleiro, zerar tabuleiro, jogar, processar jogada, checar cada parte do tabuleiro no momento da jogada e analisar o tabuleiro depois do término da partida para imprimir o vencedor ou empate. Dentro dessas funções existem os conceitos básicos de vetores e matrizes que são recursos uteis para esse programa.

O programa começa com o menu, com as opções: 1- Jogar e 0 - Sair. Se o usuário digitar “1” o programa irá continuar. Pedindo o valor da linha e da coluna da matriz onde o jogador deseja inserir seu símbolo. Em cada rodada os jogadores iram inserir seu símbolo, até um determinado jogador conseguir enfileirar o próprio símbolo na vertical, horizontal ou diagonal, ou quando não tiver mais posição para inserir os símbolos (quando o jogo “velhar”). Os usuários podem jogar quantas partidas quiserem. Se o usuário digitar “0”, o programa irá encerrar e mostrar a quantidade de partidas jogadas, a quantidade de partidas vencidas por cada jogador e a porcentagem de vitórias de cada jogador