**CENTRO UNIVERSITÁRIO DE JOÃO PESSOA**

**TÉCNICAS DE DESENVOLVIMENTO DE ALGORITMOS**

**Grupo:** Andrews Carneiro Gomes - RGM: 27505618

Jaciara Ferreira dos Santos - RGM: 28217292

Júlio Cezar Nascimento dos Santos Silva - RGM: 27396207

Lauro Alves Barbalho Júnior - RGM: 27860264

**PROJETO DE CONCLUSÃO DE DISCIPLINA**

**João Pessoa, PB**

**Dezembro de 2021**

**SUMÁRIO**

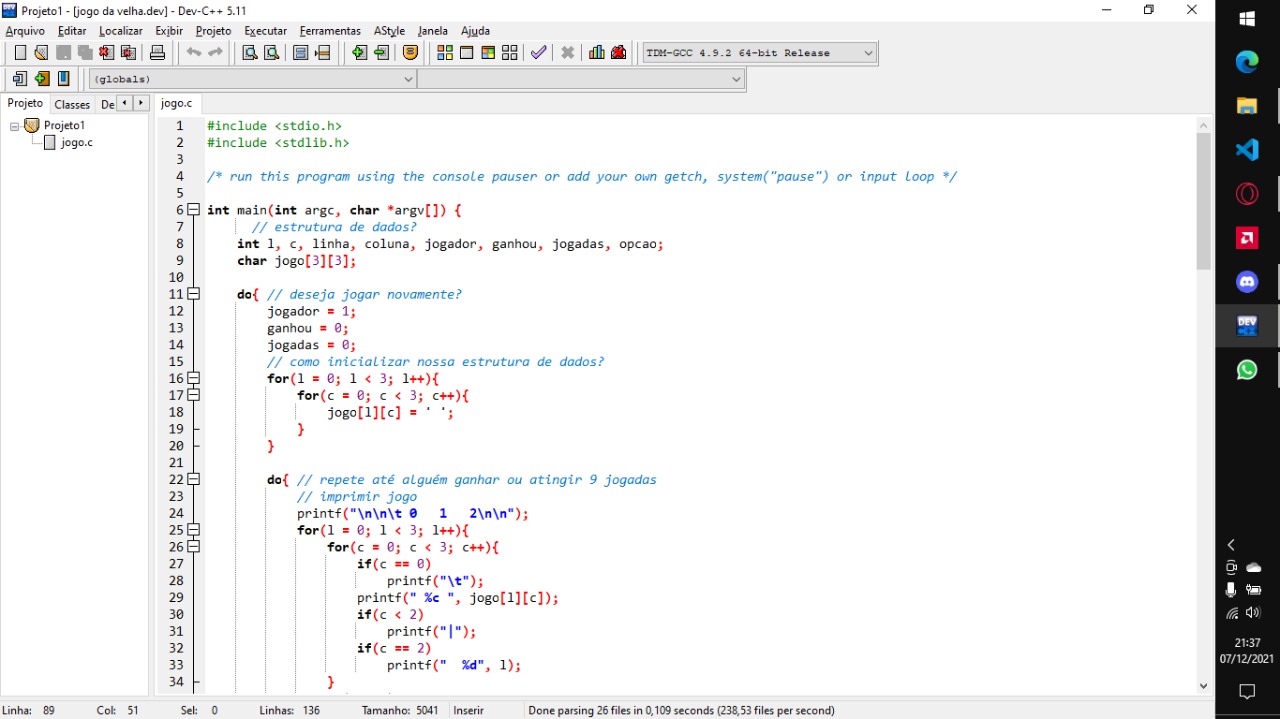
1. **INTRODUÇÃO**...........................................................................................................
2. **RESULTADOS**.............................................................................................................
3. **APÊNDICE**........................................................................................
4. **REFERÊNCIAS**...........................................................................................................
5. **INTRODUÇÃO**

O seguinte projeto tem o intuito de criar um algoritmo que execute um Jogo da Velha para dois jogadores por meio de programação usando a Linguagem C.

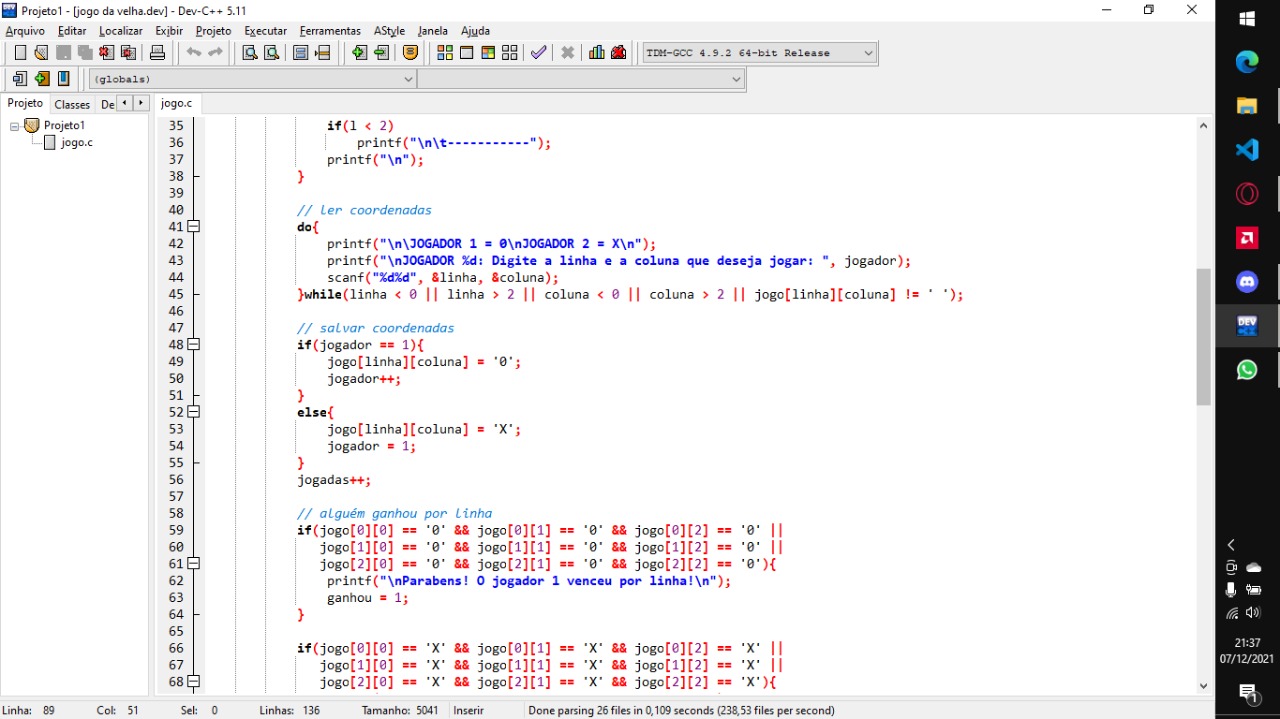
O funcionamento do jogo se dará de forma que cada jogador fornecerá uma posição válida dentro do tabuleiro atribuído como uma matriz 3x3 em que a cada turno será solicitado uma linha e uma coluna, que serão preenchidas . O jogador ‘X’ começará primeiro e logo após o jogador ‘O’. Assim que acontecer de algum dos dois completarem uma linha, uma coluna, ou uma diagonal o programa declara o vencedor ou, caso haja empate, o programa declara que ‘deu velha’.

1. **RESULTADOS**

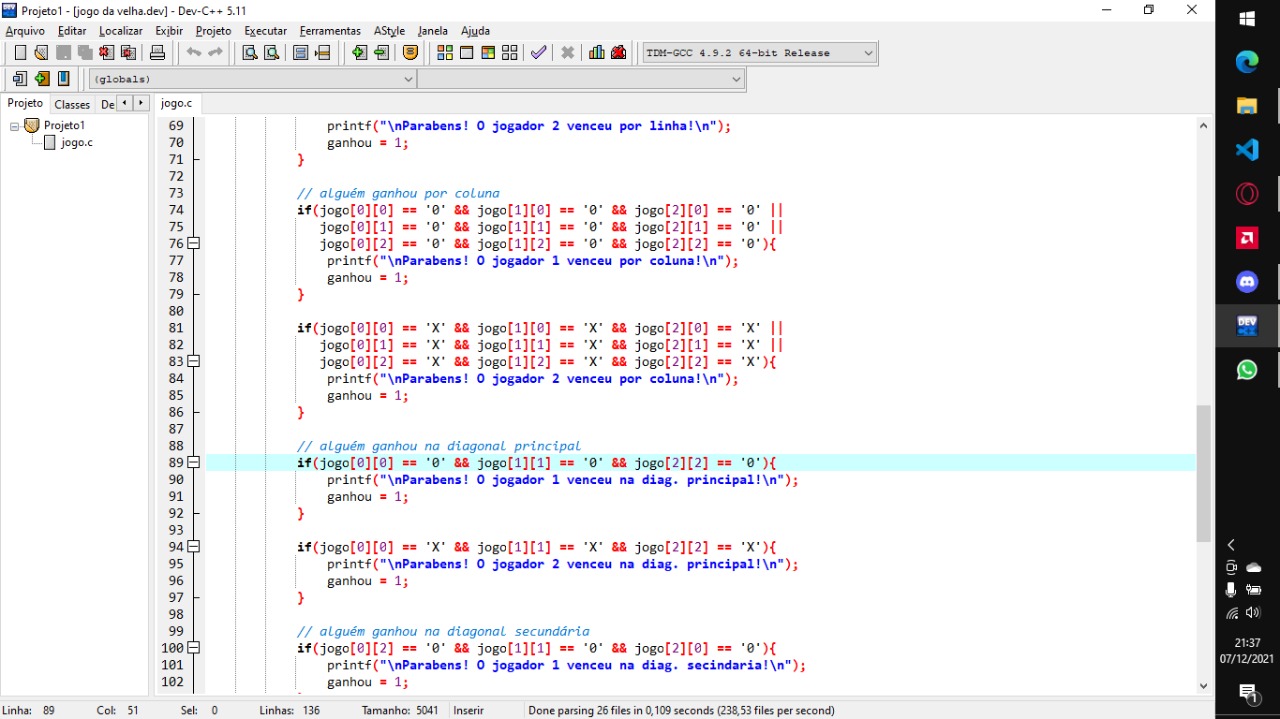
Para a realização do Jogo da Velha foram usadas as variáveis ‘int’ e ‘char’ para a criação do código. Foi utilizado um método ‘do’ para definir o jogador número 1 e 2 e um método ‘for’ para a inicialização do jogo, e para a transformação de cada espaço preenchido pelos jogadores, conforme demonstrado no print abaixo:



Em seguida foi usado um método para realizar a leitura das coordenadas possíveis, e o método ‘if’ e ‘else’ para que sempre que alguém digitar uma coordenada, ela seja gravada no código, assim não permitindo que os dois jogadores marquem a mesma coordenada, conforme demonstrado na imagem abaixo:



Após a recepção das coordenadas das jogadas e sua impressão na tela, são utilizados métodos ‘if’ com a função de definir qual foi a estratégia utilizada pelo jogador para atingir a vitória da partida, seja ela por linha, coluna, diagonal principal ou secundária. Sendo assim, cada um dos ‘if’ lêem um determinado grupo de coordenadas, imprimindo a mensagem de vencedor quando todas tiverem sido preenchidas, conforme a imagem abaixo:



1. **BIBLIOGRAFIA**
2. **APÊNDICE**

int main(int argc, char \*argv[]) {

// estrutura de dados?

int l, c, linha, coluna, jogador, ganhou, jogadas, opcao;

char jogo[3][3];

do{ // deseja jogar novamente?

jogador = 1;

ganhou = 0;

jogadas = 0;

// como inicializar nossa estrutura de dados?

for(l = 0; l < 3; l++){

for(c = 0; c < 3; c++){

jogo[l][c] = ' ';

}

}

do{ // repete até alguém ganhar ou atingir 9 jogadas

// imprimir jogo

printf("\n\n\t 0 1 2\n\n");

for(l = 0; l < 3; l++){

for(c = 0; c < 3; c++){

if(c == 0)

printf("\t");

printf(" %c ", jogo[l][c]);

if(c < 2)

printf("|");

if(c == 2)

printf(" %d", l);

}

if(l < 2)

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

printf("\n");

}

// ler coordenadas

do{

printf("\n\JOGADOR 1 = 0\nJOGADOR 2 = X\n");

printf("\nJOGADOR %d: Digite a linha e a coluna que deseja jogar: ", jogador);

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

}while(linha < 0 || linha > 2 || coluna < 0 || coluna > 2 || jogo[linha][coluna] != ' ');

// salvar coordenadas

if(jogador == 1){

jogo[linha][coluna] = '0';

jogador++;

}

else{

jogo[linha][coluna] = 'X';

jogador = 1;

}

jogadas++;

// alguém ganhou por linha

if(jogo[0][0] == '0' && jogo[0][1] == '0' && jogo[0][2] == '0' ||

jogo[1][0] == '0' && jogo[1][1] == '0' && jogo[1][2] == '0' ||

jogo[2][0] == '0' && jogo[2][1] == '0' && jogo[2][2] == '0'){

printf("\nParabens! O jogador 1 venceu por linha!\n");

ganhou = 1;

}

if(jogo[0][0] == 'X' && jogo[0][1] == 'X' && jogo[0][2] == 'X' ||

jogo[1][0] == 'X' && jogo[1][1] == 'X' && jogo[1][2] == 'X' ||

jogo[2][0] == 'X' && jogo[2][1] == 'X' && jogo[2][2] == 'X'){

printf("\nParabens! O jogador 2 venceu por linha!\n");

ganhou = 1;

}

// alguém ganhou por coluna

if(jogo[0][0] == '0' && jogo[1][0] == '0' && jogo[2][0] == '0' ||

jogo[0][1] == '0' && jogo[1][1] == '0' && jogo[2][1] == '0' ||

jogo[0][2] == '0' && jogo[1][2] == '0' && jogo[2][2] == '0'){

printf("\nParabens! O jogador 1 venceu por coluna!\n");

ganhou = 1;

}

if(jogo[0][0] == 'X' && jogo[1][0] == 'X' && jogo[2][0] == 'X' ||

jogo[0][1] == 'X' && jogo[1][1] == 'X' && jogo[2][1] == 'X' ||

jogo[0][2] == 'X' && jogo[1][2] == 'X' && jogo[2][2] == 'X'){

printf("\nParabens! O jogador 2 venceu por coluna!\n");

ganhou = 1;

}

// alguém ganhou na diagonal principal

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

printf("\nParabens! O jogador 1 venceu na diag. principal!\n");

ganhou = 1;

}

if(jogo[0][0] == 'X' && jogo[1][1] == 'X' && jogo[2][2] == 'X'){

printf("\nParabens! O jogador 2 venceu na diag. principal!\n");

ganhou = 1;

}

// alguém ganhou na diagonal secundária

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

printf("\nParabens! O jogador 1 venceu na diag. secindaria!\n");

ganhou = 1;

}

if(jogo[0][2] == 'X' && jogo[1][1] == 'X' && jogo[2][0] == 'X'){

printf("\nParabens! O jogador 2 venceu na diag. secundaria!\n");

ganhou = 1;

}

}while(ganhou == 0 && jogadas < 9);

// imprimir jogo

printf("\n\n\t 0 1 2\n\n");

for(l = 0; l < 3; l++){

for(c = 0; c < 3; c++){

if(c == 0)

printf("\t");

printf(" %c ", jogo[l][c]);

if(c < 2)

printf("|");

if(c == 2)

printf(" %d", l);

}

if(l < 2)

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

printf("\n");

}

if(ganhou == 0)

printf("\nO jogo finalizou sem gamhador!\n");

printf("\nDigite 1 para jogar novamente: \n");

scanf("%d", &opcao);

}while(opcao == 1);

return 0;

}