|  |
| --- |
| Técnica e Desenvolvimento de Algoritmos |

Grupo e RGM:

- Renan Monteiro Firmino (27822851)

- Tibério de Lima (28325141)

- Samuel Estevam Alves de Lima (27699633)

- Risiveldo Dias Maciel (27902790)

Introdução

**Descrição do Jogo:** Esse é um jogo de xadrez desenvolvido na linguagem de programação C. Seguindo os mesmos parâmetros de um jogo de xadrez convencional, portando digital. Esse jogo é bastante semelhante aqueles jogos antigos de xadrez digital, no qual era feito de forma digitada a movimentação de cada peça do jogo.

**Regras do jogo:** Como segue a mesma base de um jogo de xadrez convencional, as regras são as mesmas, apenas algumas adequações para o meio digital.

- O movimento das peças é feita pelo jogador, de forma que ele selecione qual linha e coluna a peça escolhida esteja, e logo em seguida, indique, da mesma forma, onde quer que a peça selecionada vá.

- Todas as peças, exceto o cavalo, movem-se em linha reta – horizontalmente, verticalmente ou diagonalmente.

- Os **peões (P, p),** de maneira geral, se movem somente para a frente, uma casa por vez. E também, é a única peça que não pode se mover para trás. Os peões capturam a peça do adversário se movendo nas diagonais. E, não podem promover.

- O **Bispo (B, b)** se move numa linha reta diagonalmente no tabuleiro. Ele pode mover-se por tantas casas quantas quiser, até encontrar o final do tabuleiro ou outra peça. Ele captura no mesmo caminho que se move.

- A **Torre (T, t)** se move em linha reta horizontalmente e verticalmente pelo número de casas não ocupadas, até atingir o final do tabuleiro ou ser bloqueado por outra peça. Ela captura no mesmo caminho que se move.

- O **Cavalo (C, c)** é a única peça do tabuleiro que pode pular sobre as outras peças. O cavalo move-se por duas casas horizontalmente ou verticalmente e então uma casa a mais em um ângulo reto. O movimento do cavalo forma um “L”. O cavalo captura quando termina seu movimento na casa do adversário, mas não captura quando passa por cima delas.

- A **Rainha (Q, q)** pode mover-se qualquer número de casas em linha reta – verticalmente, horizontalmente ou diagonalmente. Ela não pode pular outras peças e captura na mesma direção que se move, ocupando o lugar da peça do adversário.

- O **Rei (K, k)** pode se mover apenas para uma única casa adjacente. Assim, podendo mover-se para qualquer direção. Ele não pode se mover para uma casa ocupada por uma peça da mesma cor. O rei também pode capturar ocupando a casa do adversário.

**OBS:** Com exceção do cavalo, nenhuma outra peça pode pular outra.

Resultados

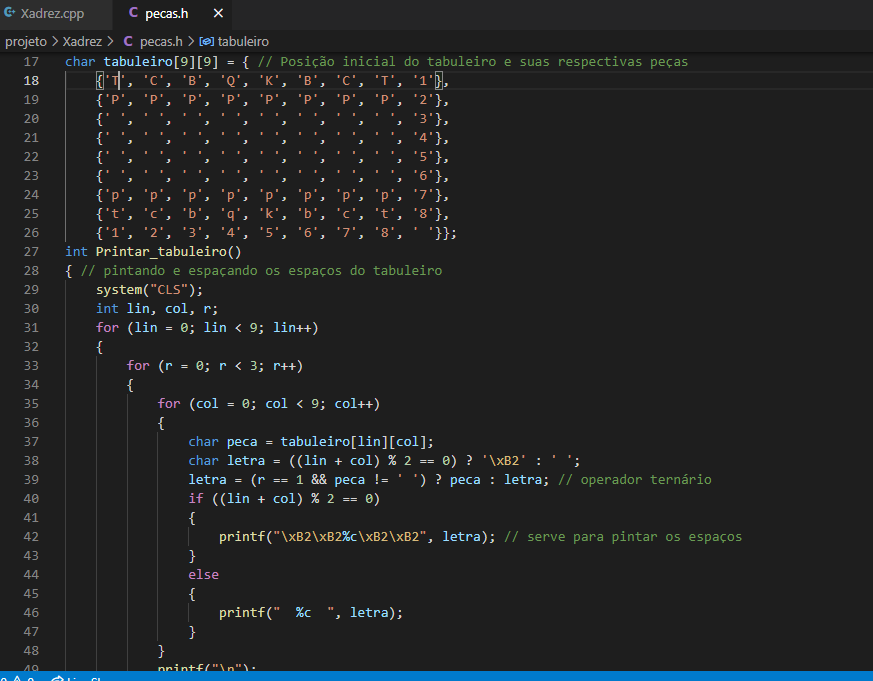
**Descrição do jogo**

Para iniciar o programação do jogo, foram incluídas as seguintes bibliotecas em C: #include <stdio.h; #include <stdlib.h>; #include <stdbool.h>; #include <locale.h>; #include string.h>;#include”pecas.h”

Em seguida, na biblioteca **pecas.h** foram definidas o espaçamento de cada casa do jogo, e as suas peças em cada lugar no início do jogo. Para facilitar a identificação, nomeamos as peças com letras maiúsculas e minúsculas. Menos as peças Rei e Rainha, pois ficariam dois R’ s, dificultando a discrepância de ambas as peças. Nomeamos a peça Rei com K e Rainha com Q. Ao todo, o tabuleiro foi feito como uma Matriz, para identificar as peças e facilitar na hora da movimentação. A matriz é formada por 9 linhas e 9 colunas.

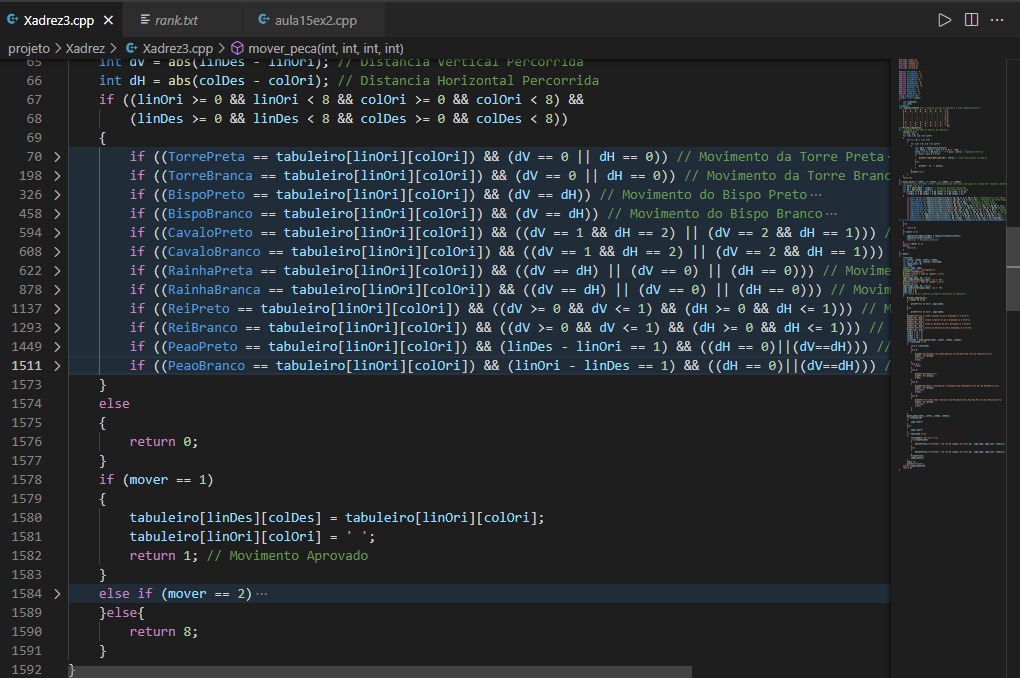
Dando continuidade ao código, em seguida foi adicionada uma função do tipo “int” pintar o tabuleiro, trazendo a identificação e semelhança das cores preto e branco.

Usando os laços de repetição “for”, do tipo aninhado, para pintar cada espaço do tabuleiro.



Aqui foi onde surgiu a primeira dificuldade, pois era necessário desenvolver uma forma de que, seria pintado uma cor branca e uma preta até completar todo o tabuleiro. O código \xB2 serve para formar uma casinha no tabuleiro. Para distinguir as cores, foi colocado um if e else que se a soma das linhas com as colunas desse par, seria preenchido (pintado), usando o código \xB2, e se fosse impar, seria um espaço em branco. Porém, toda vez que se colocava uma peça, as cores desapareciam e o espaço era muito pequeno. Para corrigir isso, foi adicionado o “printf("\xB2\xB2%c\xB2\xB2",letra);” para deixar a letra na meio da bloco preenchido. E para o tamanho, foi acrescentado uma variável r que, a cada três colunas pularia uma linhas, como mostra o forma imagem acima.

Depois, foi introduzida a função int mover\_peca(int linOri, int lindes, int colOri, int colDes), necessitando de variáveis de **xadrez.cpp** para que sirva para mover cada peça do jogo para um determinado lugar. Declarando as variáveis do tipo inteiro com nomes de colunas e linhas, linha e coluna para deslocamento e xeque. Para movimentação foram adicionadas” int mover = 0, i, j, k, l, check\_move\_bispo, check = 0;”,(foram retirados “k, l, ckeck\_move\_bispo, porquê estavam afetando negativamente o código”) no qual i corresponde as linhas que as peças podem se locomover, j é a coluna em que a peça pode se locomover. As variáveis dV e dH servem para medir as distancias entre seus movimentos. dV é a distancia vertical e dH é a distancia horizontal, usando a função abs(Serve para dar um valor absoluto, sempre positivo). Logo abaixo, entre a linha 70 e a 1511, temos uma sequencia de if e else para definir as regras do movimento de cada peça, para que o código saiba identificar se as peças podem se locomover em determinados parâmetros descritos.



Logo quando fizemos isso, notamos o problema de como capturar as peças depois, quando apresentou os erros. Para dar inicio a solução, foi usado (j = colOri) e o “if (i==linDes)” e em seguida incrementado mais códigos para solucionar o problema. E, para evitar erros, foi incrementado for “(i = linOri; i <= linDes; i++)/“(i = linOri; i >= linDes; i--)”” para verificar se não está pulando peças dentro do for, no final, foi colocado o j++ e o J-- para acompanhar o movimento das linhas e colunas.

Mais em frente, foi encontrado o problema da movimentação diagonal da rainha. Para a solução foi adicionado “check\_move\_bispo = min(i - 1, j - 1) + min(i - 1, 8 - j) + min(8 - i, j - 1) + min(8 - i, 8 - j);” e (adicionando a biblioteca “iostream” para calcular o mínimo de dois números para a movimentação. Foram retirados).

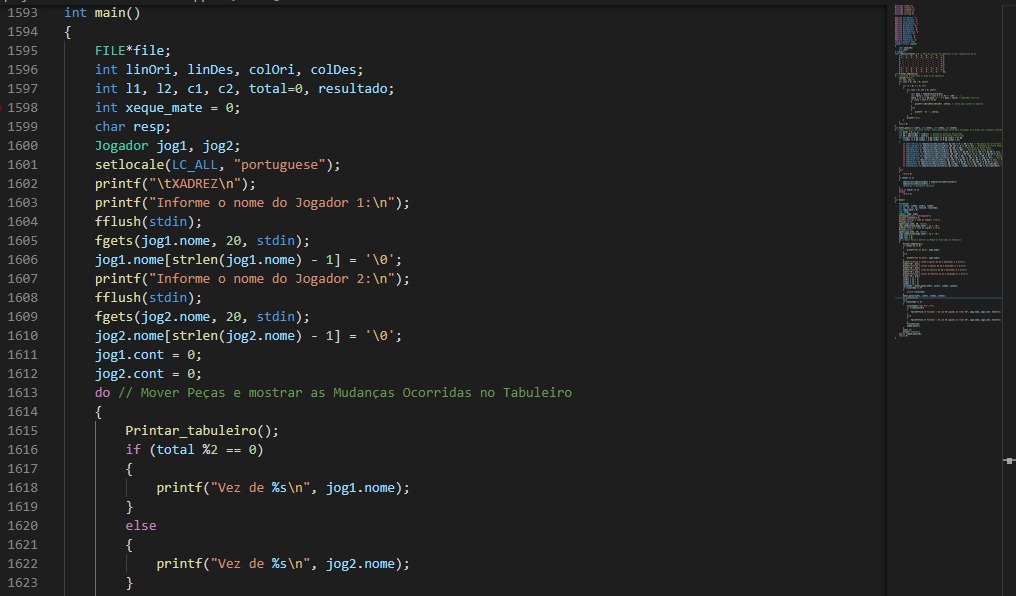
Existem outros, como o check e sair do loop que ainda não foram solucionados.( Foram solucionados).

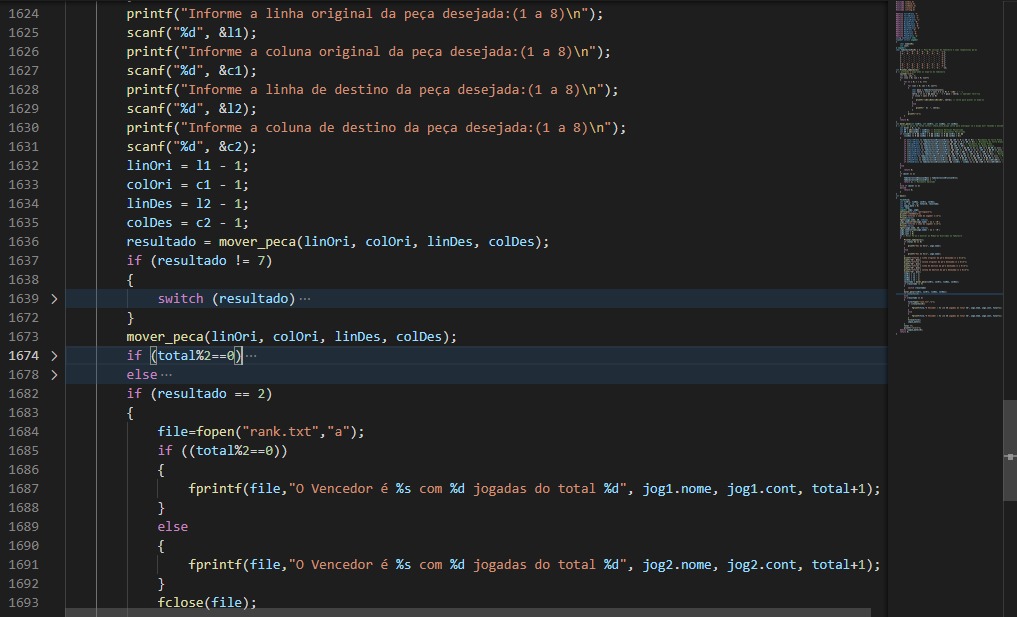
A função **int main** em **xadrez.cpp** tem como função de apresentar na tela o começo, meio e fim do código, foram criadas as variáveis: linOri, lindes, colOri, colDes, l1, l2, c1, c2, total, resultado, xeque\_mate, resp;

A **struct Jogardor** é utilizada para receber os nomes dos dois Jogadores e contar as jogadas que ambos fizeram.

As variáveis l1, l2, c1, c2; Correspondem respectivamente a: linOri, lindes, colOri, colDes; Estão no código para auxiliar na recepção dos valores de linha de origem e destino, coluna de origem e destino, por membros do grupo acharem estranho e confuso informar de 0 á 7 tanto nas linhas como nas colunas, então l1, l2, c1, c2 recebem valores entre 1 á 8 e linOri, lindes, colOri, colDes recebem por subtração 0 a 7, baseado em suas variáveis respectivas;

A variável **resultado** serve para receber o return da biblioteca “pecas.h” especificamente da função **mover\_pecas**(int linOri, int lindes, int colOri, int colDes), fazendo a prevenção de jogadas ilegais;





Conseguimos mostrar o tabuleiro e mexer as peças usando o **Do While** enquanto a variável xeque\_mate=0 vai continuar em loop, só saindo quando um dos reis for comido pois a variável **resultado** vai desencadear no código xeque\_mate=1 terminado o loop.

Código Fonte Biblioteca(“pecas.h”)

#define TorrePreta 'T'

#define TorreBranca 't'

#define CavaloPreto 'C'

#define CavaloBranco 'c'

#define BispoPreto 'B'

#define BispoBranco 'b'

#define RainhaPreta 'Q'

#define RainhaBranca 'q'

#define ReiPreto 'K'

#define ReiBranco 'k'

#define PeaoPreto 'P'

#define PeaoBranco 'p'

using namespace std;

char tabuleiro[9][9] = { // Posição inicial do tabuleiro e suas respectivas peças

{'T', 'C', 'B', 'Q', 'K', 'B', 'C', 'T', '1'},

{'P', 'P', 'P', 'P', 'P', 'P', 'P', 'P', '2'},

{' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', '3'},

{' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', '4'},

{' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', '5'},

{' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', '6'},

{'p', 'p', 'p', 'p', 'p', 'p', 'p', 'p', '7'},

{'t', 'c', 'b', 'q', 'k', 'b', 'c', 't', '8'},

{'1', '2', '3', '4', '5', '6', '7', '8', ' '}};

int Printar\_tabuleiro()

{ // pintando e espaçando os espaços do tabuleiro

system("CLS");

int lin, col, r;

for (lin = 0; lin < 9; lin++)

{

for (r = 0; r < 3; r++)

{

for (col = 0; col < 9; col++)

{

char peca = tabuleiro[lin][col];

char letra = ((lin + col) % 2 == 0) ? '\xB2' : ' ';

letra = (r == 1 && peca != ' ') ? peca : letra; // operador ternário

if ((lin + col) % 2 == 0)

{

printf("\xB2\xB2%c\xB2\xB2", letra); // serve para pintar os espaços

}

else

{

printf(" %c ", letra);

}

}

printf("\n");

}

}

return 0;

}

int mover\_peca(int linOri, int colOri, int linDes, int colDes)

{ // mover as peças de forma certa// check\_move\_bispo serve para averiguar se o bispo está fazendo o movimento correto em seu percurso,sempre sendo 13 a soma;

int mover = 0, i, j;

int dV = abs(linDes - linOri); // Distancia Vertical Percorrida

int dH = abs(colDes - colOri); // Distancia Horizontal Percorrida

if ((linOri >= 0 && linOri < 8 && colOri >= 0 && colOri < 8) &&

(linDes >= 0 && linDes < 8 && colDes >= 0 && colDes < 8))

{

if ((TorrePreta == tabuleiro[linOri][colOri]) && (dV == 0 || dH == 0)) // Movimento da Torre Preta

{

if (linOri < linDes)//Checando se linha de origem é menor que linha de destino

{

for (i = linOri; i <= linDes; i++) // Checagem de linha

{

if (i==linDes)//Permite que coma peça do oponente

{

if ((tabuleiro[i][colDes] == BispoBranco) || (tabuleiro[i][colDes] == CavaloBranco) || (tabuleiro[i][colDes] == RainhaBranca) || (tabuleiro[i][colDes] == PeaoBranco) || (tabuleiro[i][colDes] == TorreBranca)||(tabuleiro[i][colDes] == ' '))

{

mover=1;

}else if (tabuleiro[i][colDes] == ReiBranco)

{

mover=2;

}

else

{

return 8;

}

}

else{

if (((tabuleiro[i][colDes] == BispoPreto) || (tabuleiro[i][colDes] == CavaloPreto) || (tabuleiro[i][colDes] == RainhaPreta) || (tabuleiro[i][colDes] == ReiPreto) || (tabuleiro[i][colDes] == PeaoPreto) || (tabuleiro[i][colDes] == TorrePreta)

||(tabuleiro[i][colDes] == BispoBranco) || (tabuleiro[i][colDes] == CavaloBranco) || (tabuleiro[i][colDes] == RainhaBranca) ||(tabuleiro[i][j]==ReiBranco)||(tabuleiro[i][colDes] == PeaoBranco) || (tabuleiro[i][colDes] == TorreBranca)) && (tabuleiro[i][colDes] != tabuleiro[linOri][colOri]))

{//garante que não pule peças

return 9;

}

else

{

mover = 1;

}

}

}

}

else if (linOri > linDes)//Checando se linha de origem é maior que linha de destino

{

for (i = linOri; i >= linDes; i--)

{

if (i==linDes)

{

if ((tabuleiro[i][colDes] == BispoBranco) || (tabuleiro[i][colDes] == CavaloBranco) || (tabuleiro[i][colDes] == RainhaBranca) || (tabuleiro[i][colDes] == PeaoBranco) || (tabuleiro[i][colDes] == TorreBranca)||(tabuleiro[i][colDes] == ' '))

{

mover=1;

}else if (tabuleiro[i][colDes] == ReiBranco)

{

mover=2;

}

else

{

return 8;

}

}

else

{

if (((tabuleiro[i][colDes] == BispoPreto) || (tabuleiro[i][colDes] == CavaloPreto) || (tabuleiro[i][colDes] == RainhaPreta) || (tabuleiro[i][colDes] == ReiPreto) || (tabuleiro[i][colDes] == PeaoPreto) || (tabuleiro[i][colDes] == TorrePreta)||

(tabuleiro[i][colDes] == BispoBranco) || (tabuleiro[i][colDes] == CavaloBranco) || (tabuleiro[i][colDes] == RainhaBranca) || (tabuleiro[i][colDes] == PeaoBranco) || (tabuleiro[i][colDes] == TorreBranca)) && (tabuleiro[i][colDes] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

}

}

if (colOri < colDes)//Checando se coluna de origem é menor que coluna de destino

{

for (j = colOri; j <= colDes; j++)

{

if(j == colDes)

{

if ((tabuleiro[linDes][j] == BispoBranco) || (tabuleiro[linDes][j] == CavaloBranco) || (tabuleiro[linDes][j] == RainhaBranca) || (tabuleiro[linDes][j] == PeaoBranco) || (tabuleiro[linDes][j] == TorreBranca)||(tabuleiro[linDes][j] ==' '))

{

mover=1;

}else if (tabuleiro[linDes][j] == ReiBranco)

{

mover=2;

}

else

{

return 8;

}

}

else{

if (((tabuleiro[linDes][j] == BispoPreto) || (tabuleiro[linDes][j] == CavaloPreto) || (tabuleiro[linDes][j] == RainhaPreta) || (tabuleiro[linDes][j] == ReiPreto) || (tabuleiro[linDes][j] == PeaoPreto) || (tabuleiro[linDes][j] == TorrePreta)

||(tabuleiro[linDes][j] == BispoBranco) || (tabuleiro[linDes][j] == CavaloBranco) || (tabuleiro[linDes][j] == RainhaBranca) || (tabuleiro[linDes][j] ==ReiBranco) || (tabuleiro[linDes][j] == PeaoBranco) || (tabuleiro[linDes][j] == TorreBranca)) && (tabuleiro[linDes][j] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

}

}

else if (colOri > colDes)//Checando se coluna de origem é menor que coluna de destino

{

for (j = colOri; j >= colDes; j--)

{

if(j == colDes)

{

if ((tabuleiro[linDes][j] == BispoBranco) || (tabuleiro[linDes][j] == CavaloBranco) || (tabuleiro[linDes][j] == RainhaBranca) || (tabuleiro[linDes][j] == PeaoBranco) || (tabuleiro[linDes][j] == TorreBranca)||(tabuleiro[linDes][j] ==' '))

{

mover=1;

}else if (tabuleiro[linDes][j] == ReiBranco)

{

mover=2;

}

else

{

return 8;

}

}

else{

if (((tabuleiro[linDes][j] == BispoPreto) || (tabuleiro[linDes][j] == CavaloPreto) || (tabuleiro[linDes][j] == RainhaPreta) || (tabuleiro[linDes][j] == ReiPreto) || (tabuleiro[linDes][j] == PeaoPreto) || (tabuleiro[linDes][j] == TorrePreta)

||(tabuleiro[linDes][j] == BispoBranco) || (tabuleiro[linDes][j] == CavaloBranco) || (tabuleiro[linDes][j] == RainhaBranca) || (tabuleiro[linDes][j] == PeaoBranco) || (tabuleiro[linDes][j] == TorreBranca)) && (tabuleiro[linDes][j] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

}

}

}

if ((TorreBranca == tabuleiro[linOri][colOri]) && (dV == 0 || dH == 0)) // Movimento da Torre Branca

{

if (linOri < linDes)

{

for (i = linOri; i <= linDes; i++) // Checagem de linha

{

if (i==linDes)

{

if ((tabuleiro[i][colDes] == BispoPreto) || (tabuleiro[i][colDes] == CavaloPreto) || (tabuleiro[i][colDes] == RainhaPreta) || (tabuleiro[i][colDes] == PeaoPreto) || (tabuleiro[i][colDes] == TorrePreta)|| (tabuleiro[i][colDes] == ' '))

{

mover=1;

}else if (tabuleiro[i][colDes] == ReiPreto)

{

mover=2;

}

else

{

return 8;

}

}else

{

if (((tabuleiro[i][colDes] == BispoPreto) || (tabuleiro[i][colDes] == CavaloPreto) || (tabuleiro[i][colDes] == RainhaPreta) || (tabuleiro[i][colDes] == ReiPreto) || (tabuleiro[i][colDes] == PeaoPreto) || (tabuleiro[i][colDes] == TorrePreta)

||(tabuleiro[i][colDes] == BispoBranco) || (tabuleiro[i][colDes] == CavaloBranco) || (tabuleiro[i][colDes] == RainhaBranca) || (tabuleiro[i][colDes] == PeaoBranco) || (tabuleiro[i][colDes] == TorreBranca)) && (tabuleiro[i][colDes] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

}

}

else if (linOri > linDes)

{

for (i = linOri; i >= linDes; i--)

{

if (i==linDes)

{

if ((tabuleiro[i][colDes] == BispoPreto) || (tabuleiro[i][colDes] == CavaloPreto) || (tabuleiro[i][colDes] == RainhaPreta) || (tabuleiro[i][colDes] == PeaoPreto) || (tabuleiro[i][colDes] == TorrePreta)|| (tabuleiro[i][colDes] == ' '))

{

mover=1;

}else if (tabuleiro[i][colDes] == ReiPreto)

{

mover=2;

}

else

{

return 8;

}

}

else

{

if (((tabuleiro[i][colDes] == BispoPreto) || (tabuleiro[i][colDes] == CavaloPreto) || (tabuleiro[i][colDes] == RainhaPreta) || (tabuleiro[i][colDes] == ReiPreto) || (tabuleiro[i][colDes] == PeaoPreto) || (tabuleiro[i][colDes] == TorrePreta)||

(tabuleiro[i][colDes] == BispoBranco) || (tabuleiro[i][colDes] == CavaloBranco) || (tabuleiro[i][colDes] == RainhaBranca) || (tabuleiro[i][colDes] == PeaoBranco) || (tabuleiro[i][colDes] == TorreBranca)) && (tabuleiro[i][colDes] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

}

}

if (colOri < colDes)

{

for (j = colOri; j <= colDes; j++)

{

if(j == colDes)

{

if ((tabuleiro[linDes][j] == BispoPreto) || (tabuleiro[linDes][j] == CavaloPreto) || (tabuleiro[linDes][j] == RainhaPreta) || (tabuleiro[linDes][j] == PeaoPreto) || (tabuleiro[linDes][j] == TorrePreta)|| (tabuleiro[linDes][j] == ' '))

{

mover=1;

}else if (tabuleiro[linDes][j] == ReiPreto)

{

mover=2;

}

else

{

return 8;

}

}

else{

if (((tabuleiro[linDes][j] == BispoPreto) || (tabuleiro[linDes][j] == CavaloPreto) || (tabuleiro[linDes][j] == RainhaPreta) || (tabuleiro[linDes][j] == ReiPreto) || (tabuleiro[linDes][j] == PeaoPreto) || (tabuleiro[linDes][j] == TorrePreta)

||(tabuleiro[linDes][j] == BispoBranco) || (tabuleiro[linDes][j] == CavaloBranco) || (tabuleiro[linDes][j] == RainhaBranca) || (tabuleiro[linDes][j] == PeaoBranco) || (tabuleiro[linDes][j] == TorreBranca)) && (tabuleiro[linDes][j] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

}

}

else if (colOri > colDes)

{

for (j = colOri; j >= colDes; j--)

{

if(j == colDes)

{

if ((tabuleiro[linDes][j] == BispoPreto) || (tabuleiro[linDes][j] == CavaloPreto) || (tabuleiro[linDes][j] == RainhaPreta) || (tabuleiro[linDes][j] == PeaoPreto) || (tabuleiro[linDes][j] == TorrePreta)|| (tabuleiro[linDes][j] == ' '))

{

mover=1;

}else if (tabuleiro[linDes][j] == ReiPreto)

{

mover=2;

}

else

{

return 8;

}

}

else{

if (((tabuleiro[linDes][j] == BispoPreto) || (tabuleiro[linDes][j] == CavaloPreto) || (tabuleiro[linDes][j] == RainhaPreta) || (tabuleiro[linDes][j] == ReiPreto) || (tabuleiro[linDes][j] == PeaoPreto) || (tabuleiro[linDes][j] == TorrePreta)

||(tabuleiro[linDes][j] == BispoBranco) || (tabuleiro[linDes][j] == CavaloBranco) || (tabuleiro[linDes][j] == RainhaBranca) || (tabuleiro[linDes][j] == PeaoBranco) || (tabuleiro[linDes][j] == TorreBranca)) && (tabuleiro[linDes][j] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

}

}

}

if ((BispoPreto == tabuleiro[linOri][colOri]) && (dV == dH)) // Movimento do Bispo Preto

{

j = colOri;

if ((linOri < linDes) && (colOri < colDes))

{

for (i = linOri; i <= linDes; i++)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)|| (tabuleiro[i][j] == ' '))

{

mover=1;

}else if (tabuleiro[i][j] == ReiBranco)

{

mover=2;

}

else

{

return 8;

}

}else

{

if (((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == ReiBranco) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)

||(tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)) && (tabuleiro[i][j] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

j++;

}

}

else if ((linOri < linDes) && (colOri > colDes))

{

for (i = linOri; i <= linDes; i++)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)|| (tabuleiro[i][j] == ' '))

{

mover=1;

}else if (tabuleiro[i][j] == ReiBranco)

{

mover=2;

}

else

{

return 8;

}

}else

{

if (((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == ReiBranco) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)

||(tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)) && (tabuleiro[i][j] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

j--;

}

}

else if ((linOri > linDes) && (colOri < colDes))

{

for (i = linOri; i >= linDes; i--)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)|| (tabuleiro[i][j] == ' '))

{

mover=1;

}else if (tabuleiro[i][j] == ReiBranco)

{

mover=2;

}

else

{

return 8;

}

}else

{

if (((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == ReiBranco) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)

||(tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)) && (tabuleiro[i][j] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

j++;

}

}

else if ((linOri > linDes) && (colOri > colDes))

{

for (i = linOri; i >= linDes; i--)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)|| (tabuleiro[i][j] == ' '))

{

mover=1;

}else if (tabuleiro[i][j] == ReiBranco)

{

mover=2;

}

else

{

return 8;

}

}else

{

if (((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == ReiBranco) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)

||(tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)) && (tabuleiro[i][j] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

j--;

}

}

}

if ((BispoBranco == tabuleiro[linOri][colOri]) && (dV == dH)) // Movimento do Bispo Branco

{

j = colOri;

if ((linOri < linDes) && (colOri < colDes))

{

j = colOri;

for (i = linOri; i <= linDes; i++)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)|| (tabuleiro[i][j] == ' '))

{

mover=1;

}else if (tabuleiro[i][j] == ReiPreto)

{

mover=2;

}

else

{

return 8;

}

}else

{

if (((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == ReiBranco) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)

||(tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)) && (tabuleiro[i][j] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

j++;

}

}

else if ((linOri < linDes) && (colOri > colDes))

{

j = colOri;

for (i = linOri; i <= linDes; i++)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)|| (tabuleiro[i][j] == ' '))

{

mover=1;

}else if (tabuleiro[i][j] == ReiPreto)

{

mover=2;

}

else

{

return 8;

}

}else

{

if (((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == ReiBranco) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)

||(tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)) && (tabuleiro[i][j] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

j++;

}

}

else if ((linOri > linDes) && (colOri < colDes))

{

j = colOri;

for (i = linOri; i >= linDes; i--)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)|| (tabuleiro[i][j] == ' '))

{

mover=1;

}else if (tabuleiro[i][j] == ReiPreto)

{

mover=2;

}

else

{

return 8;

}

}else

{

if (((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == ReiBranco) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)

||(tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)) && (tabuleiro[i][j] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

j++;

}

}

else if ((linOri > linDes) && (colOri > colDes))

{

j = colOri;

for (i = linOri; i >= linDes; i--)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)|| (tabuleiro[i][j] == ' '))

{

mover=1;

}else if (tabuleiro[i][j] == ReiPreto)

{

mover=2;

}

else

{

return 8;

}

}else

{

if (((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == ReiBranco) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)

||(tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)) && (tabuleiro[i][j] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

j--;

}

}

}

if ((CavaloPreto == tabuleiro[linOri][colOri]) && ((dV == 1 && dH == 2) || (dV == 2 && dH == 1))) // Movimento do Cavalo Preto

{

if ((tabuleiro[linDes][colDes] == TorrePreta) || (tabuleiro[linDes][colDes] == BispoPreto) || (tabuleiro[linDes][colDes] == RainhaPreta) || (tabuleiro[linDes][colDes] == ReiPreto) || (tabuleiro[linDes][colDes] == PeaoPreto) || (tabuleiro[linDes][colDes] == CavaloPreto))

{

return 8;

}else if (tabuleiro[linDes][colDes] == ReiBranco)

{

mover=2;

}

else

{

mover = 1;

}

}

if ((CavaloBranco == tabuleiro[linOri][colOri]) && ((dV == 1 && dH == 2) || (dV == 2 && dH == 1))) // Movimento do Cavalo Branco

{

if ((tabuleiro[linDes][colDes] == TorreBranca) || (tabuleiro[linDes][colDes] == BispoBranco) || (tabuleiro[linDes][colDes] == RainhaBranca) || (tabuleiro[linDes][colDes] == ReiBranco) || (tabuleiro[linDes][colDes] == PeaoBranco) || (tabuleiro[linDes][colDes] == CavaloBranco))

{

return 9;

}else if (tabuleiro[linDes][colDes] == ReiPreto)

{

mover=2;

}

else

{

mover = 1;

}

}

if ((RainhaPreta == tabuleiro[linOri][colOri]) && ((dV == dH) || (dV == 0) || (dH == 0))) // Movimento da Rainha Preta

{

j = colOri;

if ((linOri < linDes) && (colOri < colDes))

{

for (i = linOri; i <= linDes; i++)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)|| (tabuleiro[i][j] == ' '))

{

mover=1;

}else if (tabuleiro[i][j] == ReiBranco)

{

mover=2;

}

else

{

return 8;

}

}else

{

if (((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == ReiBranco) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)

||(tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)) && (tabuleiro[i][j] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

j++;

}

}

else if ((linOri < linDes) && (colOri > colDes))

{

for (i = linOri; i <= linDes; i++)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)|| (tabuleiro[i][j] == ' '))

{

mover=1;

}else if (tabuleiro[i][j] == ReiBranco)

{

mover=2;

}

else

{

return 8;

}

}else

{

if (((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == ReiBranco) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)

||(tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)) && (tabuleiro[i][j] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

j--;

}

}

else if ((linOri > linDes) && (colOri < colDes))

{

for (i = linOri; i >= linDes; i--)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)|| (tabuleiro[i][j] == ' '))

{

mover=1;

}else if (tabuleiro[i][j] == ReiBranco)

{

mover=2;

}

else

{

return 8;

}

}else

{

if (((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == ReiBranco) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)

||(tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)) && (tabuleiro[i][j] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

j++;

}

}

else if ((linOri > linDes) && (colOri > colDes))

{

for (i = linOri; i >= linDes; i--)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)|| (tabuleiro[i][j] == ' '))

{

mover=1;

}else if (tabuleiro[i][j] == ReiBranco)

{

mover=2;

}

else

{

return 8;

}

}else

{

if (((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == ReiBranco) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)

||(tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)) && (tabuleiro[i][j] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

j--;

}

}

else if ((linOri == linDes) && (colOri > colDes))

{

for (j = colOri; j >= colDes; j--)

{

if(j == colDes)

{

if ((tabuleiro[linDes][j] == BispoBranco) || (tabuleiro[linDes][j] == CavaloBranco) || (tabuleiro[linDes][j] == RainhaBranca) || (tabuleiro[linDes][j] == PeaoBranco) || (tabuleiro[linDes][j] == TorreBranca)|| (tabuleiro[linDes][j] == ' '))

{

mover=1;

}else if (tabuleiro[linDes][j] == ReiBranco)

{

mover=2;

}

else

{

return 8;

}

}

else{

if (((tabuleiro[linDes][j] == BispoPreto) || (tabuleiro[linDes][j] == CavaloPreto) || (tabuleiro[linDes][j] == RainhaPreta) || (tabuleiro[linDes][j] == ReiPreto) || (tabuleiro[linDes][j] == PeaoPreto) || (tabuleiro[linDes][j] == TorrePreta)

||(tabuleiro[linDes][j] == BispoBranco) || (tabuleiro[linDes][j] == CavaloBranco) || (tabuleiro[linDes][j] == RainhaBranca) || (tabuleiro[linDes][j] ==ReiBranco) || (tabuleiro[linDes][j] == PeaoBranco) || (tabuleiro[linDes][j] == TorreBranca)) && (tabuleiro[linDes][j] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

}

}

else if ((linOri == linDes) && (colOri < colDes))

{

for (j = colOri; j <= colDes; j++)

{

if(j == colDes)

{

if ((tabuleiro[linDes][j] == BispoBranco) || (tabuleiro[linDes][j] == CavaloBranco) || (tabuleiro[linDes][j] == RainhaBranca) || (tabuleiro[linDes][j] == PeaoBranco) || (tabuleiro[linDes][j] == TorreBranca)|| (tabuleiro[linDes][j] == ' '))

{

mover=1;

}else if (tabuleiro[linDes][j] == ReiBranco)

{

mover=2;

}

else

{

return 8;

}

}

else{

if (((tabuleiro[linDes][j] == BispoPreto) || (tabuleiro[linDes][j] == CavaloPreto) || (tabuleiro[linDes][j] == RainhaPreta) || (tabuleiro[linDes][j] == ReiPreto) || (tabuleiro[linDes][j] == PeaoPreto) || (tabuleiro[linDes][j] == TorrePreta)

||(tabuleiro[linDes][j] == BispoBranco) || (tabuleiro[linDes][j] == CavaloBranco) || (tabuleiro[linDes][j] == RainhaBranca) || (tabuleiro[linDes][j] ==ReiBranco) || (tabuleiro[linDes][j] == PeaoBranco) || (tabuleiro[linDes][j] == TorreBranca)) && (tabuleiro[linDes][j] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

}

}

else if ((linOri > linDes) && (colOri == colDes))

{

for (i = linOri; i >= linDes; i--)

{

if (i==linDes)//Permite que coma peça do oponente

{

if ((tabuleiro[i][colDes] == BispoBranco) || (tabuleiro[i][colDes] == CavaloBranco) || (tabuleiro[i][colDes] == RainhaBranca) || (tabuleiro[i][colDes] == PeaoBranco) || (tabuleiro[i][colDes] == TorreBranca)|| (tabuleiro[i][colDes] == ' '))

{

mover=1;

}else if (tabuleiro[i][colDes] == ReiBranco)

{

mover=2;

}

else

{

return 8;

}

}else

{

if (((tabuleiro[i][colDes] == BispoPreto) || (tabuleiro[i][colDes] == CavaloPreto) || (tabuleiro[i][colDes] == RainhaPreta) || (tabuleiro[i][colDes] == ReiPreto) || (tabuleiro[i][colDes] == PeaoPreto) || (tabuleiro[i][colDes] == TorrePreta)

||(tabuleiro[i][colDes] == BispoBranco) || (tabuleiro[i][colDes] == CavaloBranco) || (tabuleiro[i][colDes] == RainhaBranca) ||(tabuleiro[i][j]==ReiBranco)||(tabuleiro[i][colDes] == PeaoBranco) || (tabuleiro[i][colDes] == TorreBranca)) && (tabuleiro[i][colDes] != tabuleiro[linOri][colOri]))

{//garante que não pule peças

return 9;

}

else

{

mover = 1;

}

}

}

}

else if ((linOri < linDes) && (colOri == colDes))

{

for (i = linOri; i <= linDes; i++)

{

if (i==linDes)//Permite que coma peça do oponente

{

if ((tabuleiro[i][colDes] == BispoBranco) || (tabuleiro[i][colDes] == CavaloBranco) || (tabuleiro[i][colDes] == RainhaBranca) || (tabuleiro[i][colDes] == PeaoBranco) || (tabuleiro[i][colDes] == TorreBranca)|| (tabuleiro[i][colDes] == ' '))

{

mover=1;

}else if (tabuleiro[i][colDes] == ReiBranco)

{

mover=2;

}

else

{

return 8;

}

}else

{

if (((tabuleiro[i][colDes] == BispoPreto) || (tabuleiro[i][colDes] == CavaloPreto) || (tabuleiro[i][colDes] == RainhaPreta) || (tabuleiro[i][colDes] == ReiPreto) || (tabuleiro[i][colDes] == PeaoPreto) || (tabuleiro[i][colDes] == TorrePreta)

||(tabuleiro[i][colDes] == BispoBranco) || (tabuleiro[i][colDes] == CavaloBranco) || (tabuleiro[i][colDes] == RainhaBranca) ||(tabuleiro[i][j]==ReiBranco)||(tabuleiro[i][colDes] == PeaoBranco) || (tabuleiro[i][colDes] == TorreBranca)) && (tabuleiro[i][colDes] != tabuleiro[linOri][colOri]))

{//garante que não pule peças

return 9;

}

else

{

mover = 1;

}

}

}

}

}

if ((RainhaBranca == tabuleiro[linOri][colOri]) && ((dV == dH) || (dV == 0) || (dH == 0))) // Movimento da Rainha Branca

{

j = colOri;

if ((linOri < linDes) && (colOri < colDes))

{

for (i = linOri; i <= linDes; i++)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)|| (tabuleiro[i][j] == ' '))

{

mover=1;

}else if (tabuleiro[i][j] == ReiPreto)

{

mover=2;

}

else

{

return 8;

}

}else

{

if (((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == ReiBranco) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)

||(tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)) && (tabuleiro[i][j] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

j++;

}

}

else if ((linOri < linDes) && (colOri > colDes))

{

j = colOri;

for (i = linOri; i <= linDes; i++)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)|| (tabuleiro[i][j] == ' '))

{

mover=1;

}else if (tabuleiro[i][j] == ReiPreto)

{

mover=2;

}

else

{

return 8;

}

}else

{

if (((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == ReiBranco) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)

||(tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)) && (tabuleiro[i][j] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

j--;

}

}

else if ((linOri > linDes) && (colOri < colDes))

{

j = colOri;

for (i = linOri; i >= linDes; i--)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)|| (tabuleiro[i][j] == ' '))

{

mover=1;

}else if (tabuleiro[i][j] == ReiPreto)

{

mover=2;

}

else

{

return 8;

}

}else

{

if (((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == ReiBranco) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)

||(tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)) && (tabuleiro[i][j] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

j++;

}

}

else if ((linOri > linDes) && (colOri > colDes))

{

j = colOri;

for (i = linOri; i >= linDes; i--)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)|| (tabuleiro[i][j] == ' '))

{

mover=1;

}else if (tabuleiro[i][j] == ReiPreto)

{

mover=2;

}

else

{

return 8;

}

}else

{

if (((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == ReiBranco) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)

||(tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)) && (tabuleiro[i][j] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

j--;

}

}

else if ((linOri > linDes) && (colOri == colDes))

{

for (i = linOri; i >= linDes; i--)

{

if (i==linDes)//Permite que coma peça do oponente

{

if ((tabuleiro[i][colDes] == BispoPreto) || (tabuleiro[i][colDes] == CavaloPreto) || (tabuleiro[i][colDes] == RainhaPreta) || (tabuleiro[i][colDes] == PeaoPreto) || (tabuleiro[i][colDes] == TorrePreta)|| (tabuleiro[i][colDes] == ' '))

{

mover=1;

}else if (tabuleiro[i][colDes] == ReiPreto)

{

mover=2;

}

else

{

return 8;

}

}else

{

if (((tabuleiro[i][colDes] == BispoPreto) || (tabuleiro[i][colDes] == CavaloPreto) || (tabuleiro[i][colDes] == RainhaPreta) || (tabuleiro[i][colDes] == ReiPreto) || (tabuleiro[i][colDes] == PeaoPreto) || (tabuleiro[i][colDes] == TorrePreta)

||(tabuleiro[i][colDes] == BispoBranco) || (tabuleiro[i][colDes] == CavaloBranco) || (tabuleiro[i][colDes] == RainhaBranca) ||(tabuleiro[i][j]==ReiBranco)||(tabuleiro[i][colDes] == PeaoBranco) || (tabuleiro[i][colDes] == TorreBranca)) && (tabuleiro[i][colDes] != tabuleiro[linOri][colOri]))

{//garante que não pule peças

return 9;

}

else

{

mover = 1;

}

}

}

}

else if ((linOri < linDes) && (colOri == colDes))

{

for (i = linOri; i <= linDes; i++)

{

if (i==linDes)//Permite que coma peça do oponente

{

if ((tabuleiro[i][colDes] == BispoPreto) || (tabuleiro[i][colDes] == CavaloPreto) || (tabuleiro[i][colDes] == RainhaPreta) || (tabuleiro[i][colDes] == PeaoPreto) || (tabuleiro[i][colDes] == TorrePreta)|| (tabuleiro[i][colDes] == ' '))

{

mover=1;

}else if (tabuleiro[i][colDes] == ReiPreto)

{

mover=2;

}

else

{

return 8;

}

}else

{

if (((tabuleiro[i][colDes] == BispoPreto) || (tabuleiro[i][colDes] == CavaloPreto) || (tabuleiro[i][colDes] == RainhaPreta) || (tabuleiro[i][colDes] == ReiPreto) || (tabuleiro[i][colDes] == PeaoPreto) || (tabuleiro[i][colDes] == TorrePreta)

||(tabuleiro[i][colDes] == BispoBranco) || (tabuleiro[i][colDes] == CavaloBranco) || (tabuleiro[i][colDes] == RainhaBranca) ||(tabuleiro[i][j]==ReiBranco)||(tabuleiro[i][colDes] == PeaoBranco) || (tabuleiro[i][colDes] == TorreBranca)) && (tabuleiro[i][colDes] != tabuleiro[linOri][colOri]))

{//garante que não pule peças

return 9;

}

else

{

mover = 1;

}

}

}

}

else if ((linOri == linDes) && (colOri > colDes))

{

for (j = colOri; j >= colDes; j--)

{

if(j == colDes)

{

if ((tabuleiro[linDes][j] == BispoPreto) || (tabuleiro[linDes][j] == CavaloPreto) || (tabuleiro[linDes][j] == RainhaPreta) || (tabuleiro[linDes][j] == PeaoPreto) || (tabuleiro[linDes][j] == TorrePreta)|| (tabuleiro[linDes][j] == ' '))

{

mover=1;

}else if (tabuleiro[linDes][j] == ReiPreto)

{

mover=2;

}

else

{

return 8;

}

}

else{

if (((tabuleiro[linDes][j] == BispoPreto) || (tabuleiro[linDes][j] == CavaloPreto) || (tabuleiro[linDes][j] == RainhaPreta) || (tabuleiro[linDes][j] == ReiPreto) || (tabuleiro[linDes][j] == PeaoPreto) || (tabuleiro[linDes][j] == TorrePreta)

||(tabuleiro[linDes][j] == BispoBranco) || (tabuleiro[linDes][j] == CavaloBranco) || (tabuleiro[linDes][j] == RainhaBranca) || (tabuleiro[linDes][j] ==ReiBranco) || (tabuleiro[linDes][j] == PeaoBranco) || (tabuleiro[linDes][j] == TorreBranca)) && (tabuleiro[linDes][j] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

}

}

else if ((linOri == linDes) && (colOri < colDes))

{

for (j = colOri; j <= colDes; j++)

{

if(j == colDes)

{

if ((tabuleiro[linDes][j] == BispoPreto) || (tabuleiro[linDes][j] == CavaloPreto) || (tabuleiro[linDes][j] == RainhaPreta) || (tabuleiro[linDes][j] == PeaoPreto) || (tabuleiro[linDes][j] == TorrePreta)|| (tabuleiro[linDes][j] == ' '))

{

mover=1;

}else if (tabuleiro[linDes][j] == ReiPreto)

{

mover=2;

}

else

{

return 8;

}

}

else{

if (((tabuleiro[linDes][j] == BispoPreto) || (tabuleiro[linDes][j] == CavaloPreto) || (tabuleiro[linDes][j] == RainhaPreta) || (tabuleiro[linDes][j] == ReiPreto) || (tabuleiro[linDes][j] == PeaoPreto) || (tabuleiro[linDes][j] == TorrePreta)

||(tabuleiro[linDes][j] == BispoBranco) || (tabuleiro[linDes][j] == CavaloBranco) || (tabuleiro[linDes][j] == RainhaBranca) || (tabuleiro[linDes][j] ==ReiBranco) || (tabuleiro[linDes][j] == PeaoBranco) || (tabuleiro[linDes][j] == TorreBranca)) && (tabuleiro[linDes][j] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

}

}

}

if ((ReiPreto == tabuleiro[linOri][colOri]) && ((dV >= 0 && dV <= 1) && (dH >= 0 && dH <= 1))) // Movimento do Rei Preto

{

j=colOri;

if ((linOri < linDes) && (colOri < colDes))

{

for (i = linOri; i <= linDes; i++)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)||(tabuleiro[i][j]==' '))

{

mover=1;

}else if (tabuleiro[i][j] == ReiBranco)

{

mover=2;

}

else

{

return 8;

}

}

j++;

}

}

else if ((linOri < linDes) && (colOri > colDes))

{

for (i = linOri; i <= linDes; i++)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)||(tabuleiro[i][j]==' '))

{

mover=1;

}else if (tabuleiro[i][j] == ReiBranco)

{

mover=2;

}

else

{

return 8;

}

}

j--;

}

}

else if ((linOri > linDes) && (colOri < colDes))

{

for (i = linOri; i >= linDes; i--)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)||(tabuleiro[i][j]==' '))

{

mover=1;

}else if (tabuleiro[i][j] == ReiBranco)

{

mover=2;

}

else

{

return 8;

}

}

j++;

}

}

else if ((linOri > linDes) && (colOri > colDes))

{

for (i = linOri; i >= linDes; i--)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco)||(tabuleiro[i][j]==' '))

{

mover=1;

}else if (tabuleiro[i][j] == ReiBranco)

{

mover=2;

}

else

{

return 8;

}

}

j--;

}

}

else if ((linOri < linDes) && (colOri == colDes))

{

for (i = linOri; i <= linDes; i++)

{

if ((tabuleiro[i][colDes] == TorrePreta) || (tabuleiro[i][colDes] == BispoPreto) || (tabuleiro[i][colDes] == CavaloPreto) || (tabuleiro[i][colDes] == RainhaPreta) || (tabuleiro[i][colDes] == PeaoPreto))

{

return 9;

}else if (tabuleiro[i][colDes] == ReiBranco)

{

mover=2;

}

else

{

mover = 1;

}

}

}

else if ((linOri > linDes) && (colOri == colDes))

{

for (i = linOri; i >= linDes; i--)

{

if ((tabuleiro[i][colDes] == TorrePreta) || (tabuleiro[i][colDes] == BispoPreto) || (tabuleiro[i][colDes] == CavaloPreto) || (tabuleiro[i][colDes] == RainhaPreta) || (tabuleiro[i][colDes] == PeaoPreto))

{

return 9;

}else if (tabuleiro[i][colDes] == ReiBranco)

{

mover=2;

}

else

{

mover= 1;

}

}

}

else if ((linOri == linDes) && (colOri < colDes))

{

for (j = colOri; j <= colDes; j++)

{

if ((tabuleiro[linDes][j] == TorrePreta) || (tabuleiro[linDes][j] == BispoPreto) || (tabuleiro[linDes][j] == CavaloPreto) || (tabuleiro[linDes][j] == RainhaPreta) || (tabuleiro[linDes][j] == PeaoPreto))

{

return 9;

}else if (tabuleiro[linDes][j] == ReiBranco)

{

mover=2;

}

else

{

mover = 1;

}

}

}

else if ((linOri == linDes) && (colOri > colDes))

{

for (j = colOri; j >= colDes; j--)

{

if ((tabuleiro[linDes][j] == TorrePreta) || (tabuleiro[linDes][j] == BispoPreto) || (tabuleiro[linDes][j] == CavaloPreto) || (tabuleiro[linDes][j] == RainhaPreta) || (tabuleiro[linDes][j] == PeaoPreto))

{

return 9;

}else if (tabuleiro[linDes][j] == ReiBranco)

{

mover=2;

}

else

{

mover = 1;

}

}

}

}

if ((ReiBranco == tabuleiro[linOri][colOri]) && ((dV >= 0 && dV <= 1) && (dH >= 0 && dH <= 1))) // Movimento do Rei Branco

{

j=colOri;

if ((linOri < linDes) && (colOri < colDes))

{

for (i = linOri; i <= linDes; i++)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)||(tabuleiro[i][j] == ' '))

{

mover=1;

}else if (tabuleiro[i][j] == ReiPreto)

{

mover=2;

}

else

{

return 8;

}

}

j++;

}

}

else if ((linOri > linDes) && (colOri < colDes))

{

for (i = linOri; i >= linDes; i--)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)||(tabuleiro[i][j] == ' '))

{

mover=1;

}else if (tabuleiro[i][j] == ReiPreto)

{

mover=2;

}

else

{

return 8;

}

}

j++;

}

}

else if ((linOri < linDes) && (colOri > colDes))

{

for (i = linOri; i <= linDes; i++)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)||(tabuleiro[i][j] == ' '))

{

mover=1;

}else if (tabuleiro[i][j] == ReiPreto)

{

mover=2;

}

else

{

return 8;

}

}

j--;

}

}

else if ((linOri > linDes) && (colOri > colDes))

{

for (i = linOri; i >= linDes; i--)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto)||(tabuleiro[i][j] == ' '))

{

mover=1;

}else if (tabuleiro[i][j] == ReiPreto)

{

mover=2;

}

else

{

return 8;

}

}

j--;

}

}

else if ((linOri == linDes) && (colOri < colDes))

{

for (j = colOri; j <= colDes; j++)

{

if ((tabuleiro[linDes][j] == TorreBranca) || (tabuleiro[linDes][j] == BispoBranco) || (tabuleiro[linDes][j] == CavaloBranco) || (tabuleiro[linDes][j] == RainhaBranca) || (tabuleiro[linDes][j] == PeaoBranco))

{

return 9;

}else if (tabuleiro[linDes][j] == ReiPreto)

{

mover=2;

}

else

{

mover = 1;

}

}

}

else if ((linOri == linDes) && (colOri > colDes))

{

for (j = colOri; j >= colDes; j--)

{

if ((tabuleiro[linDes][j] == TorreBranca) || (tabuleiro[linDes][j] == BispoBranco) || (tabuleiro[linDes][j] == CavaloBranco) || (tabuleiro[linDes][j] == RainhaBranca) || (tabuleiro[linDes][j] == PeaoBranco))

{

return 9;

}else if (tabuleiro[linDes][j] == ReiPreto)

{

mover=2;

}

else

{

mover = 1;

}

}

}

else if ((linOri < linDes) && (colOri == colDes))

{

for (i = linOri; i <= linDes; i++)

{

if ((tabuleiro[i][colDes] == TorreBranca) || (tabuleiro[i][colDes] == BispoBranco) || (tabuleiro[i][colDes] == CavaloBranco) || (tabuleiro[i][colDes] == RainhaBranca) || (tabuleiro[i][colDes] == PeaoBranco))

{

return 9;

}else if (tabuleiro[i][colDes] == ReiPreto)

{

mover=2;

}

else

{

mover = 1;

}

}

}

else if ((linOri > linDes) && (colOri == colDes))

{

for (i = linOri; i >= linDes; i--)

{

if ((tabuleiro[i][colDes] == TorreBranca) || (tabuleiro[i][colDes] == BispoBranco) || (tabuleiro[i][colDes] == CavaloBranco) || (tabuleiro[i][colDes] == RainhaBranca) || (tabuleiro[i][colDes] == PeaoBranco))

{

return 9;

}else if (tabuleiro[i][colDes] == ReiPreto)

{

mover=2;

}

else

{

mover = 1;

}

}

}

}

if ((PeaoPreto == tabuleiro[linOri][colOri]) && (linDes - linOri == 1) && ((dH == 0)||(dV==dH))) // Movimento do Peão Preto

{

if (colOri < colDes)

{

j = colOri;

for (i = linOri; i <= linDes; i++)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco))

{

mover=1;

}else if (tabuleiro[i][j] == ReiBranco)

{

mover=2;

}

else

{

return 8;

}

}

j++;

}

}

else if(colOri > colDes)

{

j = colOri;

for (i = linOri; i <= linDes; i++)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorreBranca) || (tabuleiro[i][j] == CavaloBranco) || (tabuleiro[i][j] == RainhaBranca) || (tabuleiro[i][j] == PeaoBranco) || (tabuleiro[i][j] == BispoBranco))

{

mover=1;

}else if ((tabuleiro[i][j] == ReiBranco))

{

mover=2;

}

else

{

return 8;

}

}

j++;

}

}

else

{

for ( i = linOri; i <= linDes; i++)

{

if (((tabuleiro[i][colDes] == BispoPreto) || (tabuleiro[i][colDes] == CavaloPreto) || (tabuleiro[i][colDes] == RainhaPreta) || (tabuleiro[i][colDes] == ReiPreto) || (tabuleiro[i][colDes] == PeaoPreto) || (tabuleiro[i][colDes] == TorrePreta)

||(tabuleiro[i][colDes] == BispoBranco) || (tabuleiro[i][colDes] == CavaloBranco) || (tabuleiro[i][colDes] == RainhaBranca) || (tabuleiro[i][colDes] ==ReiBranco) || (tabuleiro[i][colDes] == PeaoBranco) || (tabuleiro[i][colDes] == TorreBranca)) && (tabuleiro[i][colDes] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

}

}

if ((PeaoBranco == tabuleiro[linOri][colOri]) && (linOri - linDes == 1) && ((dH == 0)||(dV==dH))) // Movimento do Peão Branco

{

if (colOri < colDes)

{

j = colOri;

for (i = linOri; i >= linDes; i--)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto))

{

mover=1;

}else if (tabuleiro[i][j] == ReiPreto)

{

mover=2;

}

else

{

return 8;

}

}

j++;

}

}

else if (colOri > colDes)

{

j = colOri;

for (i = linOri; i >= linDes; i--)

{

if ((i==linDes)&&(j==colDes))

{

if ((tabuleiro[i][j] == TorrePreta) || (tabuleiro[i][j] == CavaloPreto) || (tabuleiro[i][j] == RainhaPreta) || (tabuleiro[i][j] == PeaoPreto) || (tabuleiro[i][j] == BispoPreto))

{

mover=1;

}else if (tabuleiro[i][j] == ReiPreto)

{

mover=2;

}

else

{

return 8;

}

}

j--;

}

}

else if(colOri==colDes)

{

for ( i = linOri; i >=linDes; i--)

{

if (((tabuleiro[i][colDes] == BispoPreto) || (tabuleiro[i][colDes] == CavaloPreto) || (tabuleiro[i][colDes] == RainhaPreta) || (tabuleiro[i][colDes] == ReiPreto) || (tabuleiro[i][colDes] == PeaoPreto) || (tabuleiro[i][colDes] == TorrePreta)

||(tabuleiro[i][colDes] == BispoBranco) || (tabuleiro[i][colDes] == CavaloBranco) || (tabuleiro[i][colDes] == RainhaBranca) || (tabuleiro[i][colDes] ==ReiBranco) || (tabuleiro[i][colDes] == PeaoBranco) || (tabuleiro[i][colDes] == TorreBranca)) && (tabuleiro[i][colDes] != tabuleiro[linOri][colOri]))

{

return 9;

}

else

{

mover = 1;

}

}

}

}

}

else

{

return 0;

}

if (mover == 1)

{

tabuleiro[linDes][colDes] = tabuleiro[linOri][colOri];

tabuleiro[linOri][colOri] = ' ';

return 1; // Movimento Aprovado

}

else if (mover == 2)

{

tabuleiro[linDes][colDes] = tabuleiro[linOri][colOri];

tabuleiro[linOri][colOri] = ' ';

return 2;//Checkmate

}else{

return 8;

}

}

Código Fonte Xadrez.cpp

#include <stdio.h>

#include <stdlib.h>

#include <stdbool.h>

#include <locale.h>

#include <string.h>

#include “pecas.h”

typedef struct jogador

{

char nome[20];

int cont;

} Jogador;

int main()

{

FILE\*file;

int linOri, linDes, colOri, colDes;

int l1, l2, c1, c2, total=0, resultado;

int xeque\_mate = 0;

char resp;

Jogador jog1, jog2;

setlocale(LC\_ALL, "portuguese");

printf("\tXADREZ\n");

printf("Informe o nome do Jogador 1:\n");

fflush(stdin);

fgets(jog1.nome, 20, stdin);

jog1.nome[strlen(jog1.nome) - 1] = '\0';

printf("Informe o nome do Jogador 2:\n");

fflush(stdin);

fgets(jog2.nome, 20, stdin);

jog2.nome[strlen(jog2.nome) - 1] = '\0';

jog1.cont = 0;

jog2.cont = 0;

do // Mover Peças e mostrar as Mudanças Ocorridas no Tabuleiro

{

Printar\_tabuleiro();

if (total %2 == 0)

{

printf("Vez de %s\n", jog1.nome);

}

else

{

printf("Vez de %s\n", jog2.nome);

}

printf("Informe a linha original da peça desejada:(1 a 8)\n");

scanf("%d", &l1);

printf("Informe a coluna original da peça desejada:(1 a 8)\n");

scanf("%d", &c1);

printf("Informe a linha de destino da peça desejada:(1 a 8)\n");

scanf("%d", &l2);

printf("Informe a coluna de destino da peça desejada:(1 a 8)\n");

scanf("%d", &c2);

linOri = l1 - 1;

colOri = c1 - 1;

linDes = l2 - 1;

colDes = c2 - 1;

resultado = mover\_peca(linOri, colOri, linDes, colDes);

if (resultado != 7)

{

switch (resultado)

{

case 0:

{

printf("Coordenadas inválidas,Destino ou Entrada Para fora do Tabuleiro:\n");

scanf(" %c",&resp);

continue;

break;

}

case 1:{

break;

}

case 2:

{

printf("Checkmate\n");

scanf(" %c",&resp);

break;

}

case 8:

{

printf("Movimento Invalido,Peça Tentando Fazer Movimento Fora de seu Parametro:\n");

scanf(" %c",&resp);

continue;

break;

}

case 9:

{

printf("A Peça Não Pode realizar esse Movimento,Pois Tem Uma Peça em Seu Percurso:\n");

scanf(" %c",&resp);

continue;

break;

}

}

}

mover\_peca(linOri, colOri, linDes, colDes);

if (total%2==0)

{

jog1.cont++;

}

else

{

jog2.cont++;

}

total ++;

if (resultado == 2)

{

file=fopen("rank.txt","a");

if ((total%2==0))

{

fprintf(file,"O Vencedor é %s com %d jogadas do total %d", jog2.nome, jog2.cont, total);

}

else

{

fprintf(file,"O Vencedor é %s com %d jogadas do total %d", jog1.nome, jog1.cont, total);

}

fclose(file);

xeque\_mate=1;

}

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

}while (xeque\_mate==0);

return 0;

}