



EXERCÍCIO 13 – CAVALOS REBELDES (PESO 2)

- **Prazo para entrega: 21/05/2017 – 23:55:00**
- **Atenção:**
 1. **Arquivo:** o nome do arquivo referente ao código-fonte do exercício deverá respeitar o seguinte padrão: <número do RA>_EX<número do exercício>.c. Exemplo: 123456_EX13.c;
 2. **E/S:** tanto a entrada quanto a saída de dados devem ser “secas”, ou seja, não devem apresentar frases explicativas. Siga o modelo fornecido e apenas complete as partes informadas.
 3. **Identificadores de variáveis:** escolha nomes apropriados;
 4. **Documentação:** inclua comentários e indentação no programa.

- **Descrição**

A Suprema Corte dos Bruxos finalmente cedeu aos muitos pedidos das escolas de magia da Europa e autorizou a realização de um novo Torneio Tribruxo, entretanto com tarefas bem menos perigosas. Entre estas tarefas haverá uma competição de Xadrez de Bruxo.

Por um motivo que todos desconhecem Neville Longbottom foi escolhido pelo Cálice de Fogo para ser o campeão de Hogwarts na nova edição do torneio, e está tendo grandes dificuldades de se preparar para a competição de xadrez. Neville se atrapalha com as ordens de movimentação dos cavalos e acaba os sacrificando desnecessariamente, assim os cavalos não confiam mais nele e pararam de seguir suas ordens.

Hermione Granger (que parece ser a única pessoa que estuda em Hogwarts) resolveu então ajudá-lo a treinar e criou uma série de tabuleiros com algumas posições encantadas, sobre as quais se uma peça for colocada será engolida, e os deu à Neville junto com diversas peças de cavalo de xadrez comum, que não se movem sozinhas. Hermione então diz a ele duas posições de um certo tabuleiro, e Neville tem que levar um cavalo de uma posição à outra sem que ele seja engolido ou jogado para fora do tabuleiro.

Neville achou muito chato mover as peças com a mão, então pediu ajuda para você: um não-bruxo (já que não há aulas de programação em Hogwarts). Sua tarefa é implementar um código encantado (ou um programa) que armazene os tabuleiros que Hermione dará a ele e calcule qual é o número mínimo de movimentos necessários para levar o cavalo da origem ao destino que ela escolherá, ou dizer se é impossível. O programa deve ter a seguinte interface:

1. **Inserir tabuleiro:** Recebe um título de identificação do tabuleiro, com no máximo 50 caracteres, em uma linha (caso um tabuleiro de mesmo título já exista em memória a mensagem pré-definida deve ser impressa, encerrando a opção sem continuar a realização das leituras). Na próxima linha há dois inteiros $1 \leq n, m \leq 100$, representando o número de linhas e colunas do tabuleiro, respectivamente. Seguem então n linhas com m colunas cada, representando as posições do tabuleiro, onde “.” representa uma posição comum e “#” representa uma posição encantada. O tabuleiro é então armazenado em memória.
2. **Calcular movimentos:** Recebe o título do tabuleiro a ser utilizado (caso não exista em memória deve ser exibida a mensagem no código fonte e a opção deve ser encerrada sem mais leituras) em uma linha, e então seguem mais duas linhas, cada uma contendo dois valores inteiros i (linha) e j (coluna), $1 \leq i \leq n$ e $1 \leq j \leq m$, representando a posição de origem (primeira linha) e a posição destino (segunda linha) do cavalo. O programa deve então calcular e imprimir na tela o número mínimo de movimentos para mover o cavalo da origem ao destino ou -1, caso seja impossível.
3. **Sair:** Libera a memória e finaliza o programa.

A movimentação de um cavalo em um tabuleiro de bruxos ocorre da mesma maneira que em um de não-bruxos, sempre em **L**. O cavalo se movimenta 2 casas para frente ou para trás e em seguida 1 casa para a direita ou para a esquerda, ou então 2 casas para a direita ou para a esquerda e em seguida 1 casa para frente ou para trás. Pode ainda pular outras peças (ou obstáculos). Observe a Figura 1 para um melhor entendimento, na qual os círculos pretos representam as posições possíveis de movimentação e “#” são posições encantadas.

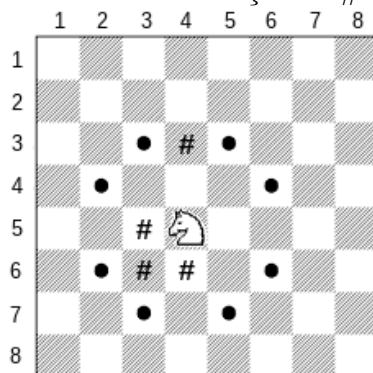


Figura 1 - Exemplo de movimentação de um cavalo

O número máximo de cálculos de movimento realizados nos testes é 2000. A dedução de como minimizar o número de movimentos do cavalo e/ou detectar impossibilidade faz parte do exercício. Não são necessárias quaisquer validações sobre as entradas (a não ser sobre a opção), visto que todas respeitam os padrões e limites estabelecidos acima.

Complete o arquivo `ex13.c`

Exemplo de E/S (os comentários entre parênteses não deverão ser exibidos):

Entrada	Saída
1 (inserir tabuleiro)	
Facile Tabula (título)	
3 3 (n e m)	
...	
...	
..# (tabuleiro)	
1 (inserir tabuleiro)	
Durum Tabula (título)	
8 8 (n e m)	
.#.#.#..	
#...#.#.	
.#.#....	
..#...#.	
#.#..#..	
..##...##	
.#....#.	
..#.#... (tabuleiro)	
2 (calcular movimentos)	
Facile Tabula (tabuleiro)	
1 1 (posição de origem)	
3 1 (posição de destino)	
	2 (mínimo possível de movimentos)
2 (calcular movimentos)	
Facile Tabula (tabuleiro)	
1 1 (posição de origem)	
2 2 (posição de destino)	
	-1 (impossível chegar ao destino)
2 (calcular movimentos)	
Durum Tabula (tabuleiro)	
7 3 (posição de origem)	
1 5 (posição de destino)	
	4 (mínimo possível de movimentos)
3 (sair)	

- **Cuidados:**

1. **Erros de compilação:** nota **zero** no exercício
2. **Tentativa de fraude:** nota **zero na média** para todos os envolvidos.