

Trabalho Prático: O Quebra-cabeças das 8 Pastilhas

Este é um jogo de quebra cabeças que consiste em ordenar 8 peças (pastilhas) deslizantes em um tabuleiro de 3×3 quadrados, movendo sempre os números para o buraco no tabuleiro.



Figura 1: Exemplo do tabuleiro em madeira ordenado

Existem diversas versões do jogo com tamanho de tabuleiro distinto e o 8, 15, 24 peças ou mais. Na versão 3×3 , o jogo possui 9 posições, sendo 8 peças com números e um espaço vazio para movimentá-las. O objetivo do jogo é arrancar as peças em ordem, da esquerda para a direita, de cima a baixo, deixando a posição inferior direita do tabuleiro vazia, ou seja, as 8 peças devem ser deslizadas no tabuleiro, fazendo-se quantos movimentos fossem necessários, a fim de serem deixadas em uma sequência crescente, ficando apenas o nono quadra vazio.

Atenção: somente as peças adjacentes ao espaço vazio podem ser movidas.

O Trabalho Prático

Você deve implementar o jogo para um único jogador utilizando uma matriz para armazená-lo, e deve ser possível continuar um jogo previamente salvo.

Seu programa deve ser executado seguindo este fluxo:

1. O programa deve exibir um menu com as opções (0) para sair, (1) começar um novo jogo, (2) continuar o jogo atual, (3) continuar um jogo salvo, ou (4) ver instruções de como jogar.
 - (a) Se a opção for iniciar um novo jogo, o programa deve solicitar qual o nível de dificuldade desejado: fácil (1), médio (2) ou difícil (3).
 - (b) Se a opção for continuar o jogo atual, o programa volta para o jogo. Observe que essa opção só ser selecionada se o usuário já tiver começado.
 - (c) Se a opção for continuar um jogo salvo, o programa deve solicitar o nome do arquivo texto contendo o estado inicial e imprimir o jogo logo em seguida. Continue do passo 2.

- (d) Se a opção instruções for selecionado, o programa deve imprimir os comandos possíveis e instruções de como jogar.
2. O jogador deve digitar um dos comandos a seguir, até finalizar o jogo ou voltar para o menu inicial.

<num>: para mover uma peça. O usuário deve digitar o número da peça que quer mover.

salvar: para armazenar em um arquivo o jogo em seu estado atual. O usuário deve digitar o nome do arquivo logo após o comando. Exemplo: “**salvar jogo.txt**” salva o jogo no arquivo “jogo.txt”.

voltar: volta para o menu inicial.

Importante: seu programa deve proibir que o usuário execute comandos inválidos. O usuário deve ser alertado com uma mensagem de erro caso digite um valor inválido. O programa também deve detectar quando todas as peças estiverem na posição correta, ou seja, avisar ao jogador(a) que ele(a) venceu.

Criando um novo jogo

As peças iniciais do jogo **não poderão estar dispostas em qualquer sequência aleatória**, pois poderia resultar em um jogo sem solução possível.

Portanto, para se obter um jogo inicial válido será utilizada a mesma estratégia do jogo real: a partir das peças ordenadas, o programa fará n movimentos aleatórios com as peças no sentido de embaralhar o jogo. No entanto, você pode usar um tabuleiro aleatório para fins de validação as demais funcionalidades do programa.

Observe que os movimentos para embaralhar o jogo devem ser válidos, ou seja, somente peças adjacentes ao espaço vazio podem ser movidas. Atente-se ao fato do programa não ficar em laço embaralhando a mesma peça.

O nível de dificuldade do jogo indica quantos movimentos serão efetuados com as peças, afim de embaralhá-las.

fácil (1): Utilizar $n = 20$, ou seja, mover as peças 20 vezes aleatoriamente.

médio (2): Utilizar $n = 40$.

difícil (3): Utilizar $n = 80$.

Lendo e salvando um jogo

O arquivo de um jogo que será lido/escrito contém na linha 1 a dimensão da matriz ($n \times n$), ou seja, o número de linhas e colunas do tabuleiro. A seguir, a matriz é apresentada, contendo exatamente n linhas, cada uma com n números separados por espaços. Na linha após a matriz é informado o número jogadas que o jogador executou até o momento. Note que a posição vazia é representada pelo valor 0.

O quadro a seguir indica um exemplo de arquivo contendo um jogo.

```
1 3
2 8 3 2
3 1 5 4
4 7 6 0
5 12
```

Instruções

- O problema deve ser resolvido por meio de um programa em C.
- Inclua seu nome e número de matrícula como comentário em todos os arquivos .c e .h gerados.
- Não serão aceitos trabalhos que caracterizem cópia (mesma estrutura e algumas pequenas modificações) de outro.
- Após a entrega dos trabalhos serão marcadas entrevistas com cada um dos alunos para apresentação dos mesmos para os professores.

Entrega

- Você deverá entregar o código fonte (arquivo ZIP) e um link para um vídeo de cerca de 5 minutos feito por você explicando o código e mostrando seu programa. O vídeo pode ser disponibilizado em qualquer plataforma de sua escolha (Google Drive, Youtube, Vimeo, etc.).
- A entrega deve ser feita pelo Moodle até as 23h55 do dia XX de junho de 2022.
- As entrevistas serão feitas na semana do dia de entrega durante os horários das aulas.

Avaliação

- Funcionamento adequado do programa.
- Atendimento ao enunciado do trabalho.
- Clareza do código (que deve ser devidamente comentado e indentado).
- Utilização de funções.
- Adequação da estrutura do programa (variáveis e comandos utilizados).
- Apresentação do trabalho.
- Compilação (códigos que não compilam serão zerados, e *warnings* diminuirão a nota). Utilizaremos o compilador GCC.

Exemplo de Execução

Você pode (e deve) customizar e melhorar as saídas do programa. A seguir segue um exemplo simples apenas para entendimento (os dados digitados pelo usuário estão destacados em azul):

Exemplo 1:

./quebra-cabecas

Bem vindo ao Quebra-Cabeças

0. Sair do Jogo

1. Começar um novo jogo
2. Continuar o jogo atual
3. Continuar um jogo salvo
4. Instruções do jogo

Escolha a opção (digite “sair” em qualquer lugar para sair do jogo): **1**

1	2	3
4	5	6
7	8	

Tabuleiro ordenado!

Digite o nível de dificuldade para embaralhar, fácil (1), médio (2) ou difícil (3): **2**

Tabuleiro embaralhado. Vamos começar o jogo!

8	3	2
1		5
7	6	4

Digite um comando ou número da peça que quer mover: **3**

8		2
1	3	5
7	6	4

Quantidade de jogadas até agora: 1

Digite um comando ou número da peça que quer mover: **6**

Não é possível mover a peça **6**.

Digite um comando ou número da peça que quer mover: **8**

	8	2
1	3	5
7	6	4

Quantidade de jogadas até agora: 2

Digite um comando ou número da peça que quer mover: **1**

1	8	2
	3	5
7	6	4

Digite um comando ou número da peça que quer mover: **salvar meujogo.txt**

Arquivo “meujogo.txt” salvo com sucesso!

Digite um comando ou número da peça que quer mover: **sair**

Exemplo 2:

`./quebra-cabecas`

Bem vindo ao Quebra-Cabeças

- 0. Sair do Jogo
- 1. Começar um novo jogo
- 2. Continuar o jogo atual
- 3. Continuar um jogo salvo
- 4. Instruções do jogo

Escolha a opção (digite “sair” em qualquer lugar para sair do jogo): **3**

Digite o nome do arquivo: **meujogo3.txt**

Arquivo “meujogo3.txt” Carregado. Vamos começar o jogo!

1	2	3
4	6	8
7	5	

Quantidade de jogadas até agora: 63

Digite um comando ou número da peça que quer mover: **8**

1	2	3
4	6	
7	5	8

Quantidade de jogadas até agora: 64

Digite um comando ou número da peça que quer mover: **6**

1	2	3
4		6
7	5	8

Quantidade de jogadas até agora: 65

Digite um comando ou número da peça que quer mover: **5**

1	2	3
4	5	6
7		8

Quantidade de jogadas até agora: 66

Digite um comando ou número da peça que quer mover: **8**

1	2	3
4	5	6
7	8	

Parabéns! Você completou com quebra-cabeças com 67 movimentos.