
Primeira prova

Orientações gerais

- A prova é individual e provas semelhantes receberão nota zero;
- Colocar nome e matrícula em todos os arquivos pedidos, a não observação desse requisito pode acarretar nota zero;
- Leia atentamente a prova!
- Boa prova!

Questão 1 (20 pontos)

Enunciado

Um aluno de orientação a objetos decidiu criar um jogo de tabuleiro que consiste em movimentar um jogador com uma quantidade pré-definida de movimentos n pelo tabuleiro de dimensão d . Durante este deslocamento o programa deve fazer a simulação dos movimentos do personagem até que ele seja movimentado para fora do tabuleiro ou que os movimentos n movimentos sejam concluídos. Baseado nesta descrição, faça o que se pede:

Crie uma classe *Movimento* **(1 ponto)** que contenha:

- 1) Um atributo *direcao* do tipo *char* que vai indicar a direção do movimento: *d* – move o personagem para a direita, *e* – move o personagem para a esquerda, *c* – move o personagem para cima, e *b* – move o personagem para baixo **(1 ponto)**;
- 2) Um atributo *deslocamento* do tipo *int* que indica a quantidade de casas deslocadas pelo personagem **(1 ponto)**.

Crie uma classe *Tabuleiro* **(1 ponto)** que vai controlar os movimentos do *Jogador* contendo:

- 1) O atributo inteiro *ordem* que vai definir um tabuleiro quadrado com a quantidade de linhas e colunas igual a ordem. Vale ressaltar que esse valor não deve ser alterado após a inicialização. Por exemplo, um tabuleiro de ordem 5 terá 5 linhas e 5 colunas **(1 ponto)**;
- 2) Os atributos inteiros *x* e *y* que vão representar a posição atual do Jogador **(1 ponto)**;
- 3) Um método *movimentar* que recebe um parâmetro do tipo *Movimento*, altera os valores de *x* e *y* de acordo com o

movimento recebido como parâmetro e retorna true, caso o personagem permaneça dentro do tabuleiro, ou false, caso contrário **(3 pontos)**;

4) Um método deslocamentoRelativo que não recebe parâmetros e retorna um double a distância entre o ponto (0, 0) e o ponto (x, y) **(2 pontos)**;

5) Um método imprime que vai mostrar a representação atual do tabuleiro onde os pontos vazios são representados pelo caractere "*" e o ponto do personagem é representado pelo caractere "J" **(2 pontos)**;

Crie uma classe principal chamada *Jogo* que é responsável por viabilizar o funcionamento do programa. Nesta classe realize as seguintes atividades:

1) leia a ordem do tabuleiro e a quantidade de movimentos **(1 ponto)**;

2) crie o tabuleiro com a ordem definida **(1 ponto)**;

3) leia os movimentos e armazene em uma lista **(2 pontos)**;

4) faça a simulação de cada movimento **(2 pontos)**;

a. Para cada movimento, imprima uma mensagem que descreva o movimento. Por exemplo, "Movimento (b, 3)";

b. Se o movimento mantiver o personagem dentro do tabuleiro, imprima a simulação do tabuleiro, a posição atual e o deslocamento relativo;

c. Se o deslocamento levar o personagem para fora do tabuleiro, imprima a posição atual, a mensagem "Personagem saiu do tabuleiro!" e a mensagem "Fim de jogo!", e encerre a simulação;

Encapsule os campos das classes e criem os construtores adequados **(1 pontos)**.

OBS1: Durante a resolução é permitido criar métodos auxiliares.

OBS2: Utilize teclado.next().charAt(0) para ler os caracteres.

Exemplo de execução para fora do tabuleiro

```
Digite a dimensao do tabuleiro:
5
Digite a quantidade de movimentos:
3
Movimento 1:
Digite a direção do movimento (d,e,c ou b):
d
Digite o deslocamento do movimento:
3
Movimento 2:
Digite a direção do movimento (d,e,c ou b):
e
Digite o deslocamento do movimento:
2
Movimento 3:
Digite a direção do movimento (d,e,c ou b):
b
Digite o deslocamento do movimento:
6
Simulacao:
Movimento (d, 3)
* * * J *
* * * * *
* * * * *
* * * * *
* * * * *
Posicao atual: (3,0)
Deslocamento relativo:3.0
Movimento (e, 2)
* J * * *
* * * * *
* * * * *
* * * * *
* * * * *
Posicao atual: (1,0)
Deslocamento relativo:1.0
Movimento (b, 6)
Posicao atual: (1,6)
Personagem saiu do tabuleiro!
Fim de jogo!
```

Exemplo de execução para fora do tabuleiro

Digite a dimensao do tabuleiro:

5

Digite a quantidade de movimentos:

3

Movimento 1:

Digite a direção do movimento (d,e,c ou b):

b

Digite o deslocamento do movimento:

2

Movimento 2:

Digite a direção do movimento (d,e,c ou b):

b

Digite o deslocamento do movimento:

1

Movimento 3:

Digite a direção do movimento (d,e,c ou b):

d

Digite o deslocamento do movimento:

4

Simulacao:

Movimento (b, 2)

* * * * *

* * * * *

J * * * *

* * * * *

* * * * *

Posicao atual: (0,2)

Deslocamento relativo:2.0

Movimento (b, 1)

* * * * *

* * * * *

* * * * *

J * * * *

* * * * *

Posicao atual: (0,3)

Deslocamento relativo:3.0

Movimento (d, 4)

* * * * *

* * * * *

* * * * *

* * * * J

* * * * *

Posicao atual: (4,3)

Deslocamento relativo:5.0

Questão 2 (10 pontos)

Enunciado

Um escritor deseja utilizar um programa que substitua as abreviações de um texto para forma completa da palavra. Por coincidência, ele estudava Orientação a Objetos e resolveu utilizar Map para solucionar o seu problema.

Crie um programa que leia um conjunto de abreviações e seus valores até o usuário digite a palavra "Fim" na abreviação. Em seguida leia uma frase que contém abreviações e imprima a frase traduzida.

OBS1: É obrigatório o uso de Map e HashMap.

Exemplo de execução

```
Digite a abreviacao ou Fim para sair
OO
Digite o valor de OO:
Orientacao a Objetos
Digite a abreviacao ou Fim para sair
hj
Digite o valor de hj:
hoje
Digite a abreviacao ou Fim para sair
Fim
Digite uma frase para ser traduzida:
A prova de OO eh hj
Frase traduzida:
A prova de Orientacao a Objetos eh hoje
```