

MC322 – Programação Orientada a Objetos

Laboratório 13 (Avaliação 4) – 1s2020

Leonardo Montecchi (Professor)
leonardo@ic.unicamp.br

Iury Cleveston (PED)
iurycl@gmail.com

Matheus Rotta (PAD)
matheusrotta7@gmail.com

Matheus Mazon (PAD)
matheusmazon@outlook.com

17/06/2020

1 Introdução

Este laboratório tem por objetivo avaliar os conceitos aprendidos na Unidade 4, como o uso de interfaces e tratamento de exceções, além de herança e polimorfismo.

2 Avaliação

Desenvolva o minigame **Jewel Collector 2.0**, implementado previamente no Laboratório 04. O objetivo dessa nova versão é melhorar o código anterior através da implementação dos novos conceitos de POO aprendidos até o momento.

Para isso, implemente o mapa como uma matriz de items (jewels, obstacles, demais elementos mostrados no mapa). Seu código **deverá** imprimir o mapa de forma simples, como o exemplo abaixo:

```
public void print() {  
    for (int i = 0; i < map.length; i++) {  
        for (int j = 0; j < map[i].length; j++) {  
            System.out.print(map[i][j]);  
        }  
        System.out.print("\n");  
    }  
}
```

Lembrando que as joias serão do tipo **Red**, valor de 100 pontos, símbolo JR; **Green**, no valor de 50 pontos, símbolo JG; e **Blue**, no valor de 10 pontos,

simbolo JB. Obstáculos serão do tipo **Water** com simbolo ##, ou **Tree** com simbolo \$\$\$. Espaços vazios com o simbolo - -. Robô com o simbolo ME.

Nesta versão do jogo, o robô inicia com 5 pontos de energia e poderá se deslocar nas quatro direções. A cada deslocamento, ele perde 1 ponto de energia. Quando chegar a zero, o robô não poderá se mover mais.

O robô interage com o ambiente podendo *usar* os itens no mapa quando ele estiver em posições adjacentes a estes itens. O efeito do uso depende das características do item. Alguns poderão ser coletados (*collect*), sendo assim removidos do mapa e guardados na sacola do robô. Outros poderão ser usados pelo robô para recarregar (*recharge*) sua energia.

Os itens *Tree* e *Blue Jewel* fornecerão energia para o robô. *Tree* fornecerá 3 pontos de energia, enquanto que *Blue Jewel* fornecerá 5 pontos. Todas as joias serão coletadas após o uso. Utilize o conceito de interface para realizar essas ações.

Implemente também exceções para tratar os seguintes casos:

1. robô tenta se deslocar para uma posição fora dos limites do mapa;
2. robô tenta se deslocar para uma posição ocupada por outro item;
3. outras situações que achar pertinente o uso de exceções.

O usuário poderá controlar o robô através dos seguintes comandos: tecla **w** desloca o robô para o norte, a tecla **s** desloca para o sul, a tecla **a** desloca para oeste e a tecla **d** para leste. Para usar (coletar/recarregar) um item, use a tecla **u**.

Todos os itens são intransponíveis. Para cada comando executado pelo usuário, imprima o estado atual do mapa, a energia do robô, bem como o estado da sacola do robô.

Observação: Caso julgue necessário para uma melhor estruturação do código, é permitido criar classes adicionais.

2.1 Fluxo de Execução

1. Crie um mapa com dimensão 10x10
2. Crie e insira as joias de acordo com o tipo e posição (x, y) abaixo:
 - (a) Red - (1, 9)
 - (b) Red - (8, 8)
 - (c) Green - (9, 1)
 - (d) Green - (7, 6)
 - (e) Blue - (3, 4)
 - (f) Blue - (2, 1)
3. Crie e insira os obstáculos de acordo com o tipo e posição (x, y) abaixo:

- (a) Water - (5, 0)
 - (b) Water - (5, 1)
 - (c) Water - (5, 2)
 - (d) Water - (5, 3)
 - (e) Water - (5, 4)
 - (f) Water - (5, 5)
 - (g) Water - (5, 6)
 - (h) Tree - (5, 9)
 - (i) Tree - (3, 9)
 - (j) Tree - (8, 3)
 - (k) Tree - (2, 5)
 - (l) Tree - (1, 4)
4. Crie o robô na posição $(x, y) = (0, 0)$.
 5. Inicie o jogo, isto é, leia o teclado e colete todas as joias, explore os itens para conseguir energia, e desvie dos obstáculos interativamente.

3 Critérios de Avaliação

Em particular, o código será avaliado de acordo com os seguintes aspectos:

1. Aplicação de princípios de POO (70%).
 - Ex: encapsulamento de atributos, responsabilidades de classes, uso de herança e polimorfismo, uso de interfaces, tratamento de exceções.
2. Funcionalidades implementadas (30%).

4 Submissão

Submeta o trabalho no link de entrega na página do Classroom da disciplina, em formato de arquivo compactado (zip). Envie o arquivo com o nome **{ra}_Lab13.zip**. Arquivos a serem submetidos:

- A pasta compactada com o projeto criado no Eclipse com todos os códigos fonte. Caso não use Eclipse, a pasta compactada com todos os códigos fontes do trabalho realizado.

Este laboratório **vale** nota. Entrega para o dia **29/06/2020 às 23h59**.