

Exame Época Recurso



3 de Março 2021

Duração 2 horas

## Departamento de Engenharia Informática

## Resolva cada exercício em folhas separadas

4 p<sup>tos</sup> 1. Escreva um método genérico para manter uma lista de elementos favoritos. Os elementos dessa lista devem estar ordenados por ordem descendente da sua pesquisa. Por exemplo, dada a lista de inteiros l={4, 5, 8, 9}, e a pesquisa dos elementos pela seguinte ordem p={8,9,9,3,9,5,8,8,3}, produz a lista de favoritos l={9, 8, 5, 4}. Assuma que que a lista l não contém repetidos e dois ou mais elementos com a mesma frequência de pesquisa aparecem por ordem arbitrária (a lista de favoritos da pesquisa indicada também pode ser l={8, 9, 5, 4}).

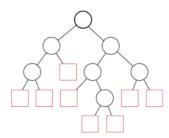
```
public static<E> void favouriteList(List<E> 1, List<E> p)
```

3 p<sup>tos</sup> 2. Considere a seguinte função que verifica os fatores de balanceamento numa classe AVL.

```
public boolean balFactor(Node<E> node) {
    if (node == null)
        return true;
    int bf = height(node.getRight())-height(node.getLeft());
    return bf >= -1 && bf <= 1 && balFactor(node.getLeft()) && balFactor(node.getRight());
}</pre>
```

Analise o método quanto à sua complexidade temporal, utilizando a notação Big-Oh. Justifique adequadamente.

5 p<sup>tos</sup> 3. O comprimento do **caminho interno** numa árvore binária é a soma das profundidades dos seus nós internos (nós não folha), sendo a profundidade de um nó o número de ancestrais desse mesmo nó. Por exemplo na árvore apresentada, o comprimento do caminho interno é 11 = (0+1+2+1+2+3+2).



Desenvolva um método na classe genérica TREE<E> que devolva o comprimento do caminho interno de uma árvore binária.

int lengthInternalPath()



Exame Época Recurso



Departamento de Engenharia Informática

3 de Março 2021 Duração 2 horas

## Resolva cada exercício em folhas separadas

5 p<sup>tos</sup> 4. Considere um grafo que representa um mapa para distribuição de um medicamento. Este mapa tem um conjunto de pontos geográficos (Ponto), interligados por estradas com um determinado comprimento. Um ponto geográfico pode ser uma farmácia ou um cliente. O método isFarmacia() indica os pontos farmácia.

```
public class Ponto {
    public boolean isFarmacia() { ... }
}

public class GrafoMedicamento {
    private Graph<Ponto, Double> g = new Graph<>(true);
    public Ponto farmaciaMaisProxima(Ponto pIni, List<Ponto> pointsVisited)
}
```

Implemente o método farmaciaMaisProxima, que recebe um Ponto inicial (cliente) e encontra no GrafoMedicamento, o ponto **mais próximo** ao Ponto inicial que seja farmácia e não tenha sido já visitado (não se encontre na lista pointsVisited).

5. Elabore um método que divide os elementos de uma fila de prioridades em três níveis: vermelho (1), amarelo (2) e verde (3), de tal modo que os n primeiros elementos ficam com prioridade vermelha, os n últimos elementos com prioridade verde e os restantes com prioridade amarela. As filas vermelha e amarela têm maior prioridade pelo que só devem ser alocados elementos à fila verde depois de distribuídos na vermelha e amarela. O método só devolve a lista relativa à prioridade solicitada. Para as heaps abaixo representadas e n=2:

```
vermelha (1): [A,B]
amarela (2): [C,D,E,F,K]
verde (3): [P,S]

Vermelha (1): [A,B]
amarela (2): [C,K]
verde (3): []
```