isep Instituto Superior de Engenharia do Porto

Estruturas de Informação

Época Normal

23 de Janeiro 2018

Departamento de Engenharia Informática

Duração 2 horas

Resolva cada exercício em folhas separadas

3.5 p^{tos}

1. Elabore um método que recebe uma lista de strings e altera as strings que são repetidas adicionando-lhe a ordem de repetição dessa string na lista. Por exemplo a lista de strings:

   ["carro", "auto", "auto", "autocarro", "carro", "carro", "autocarro", "auto"]

   é transformada em:

   ["carro", "auto", "auto 1", "autocarro", "carro 1", "carro 2", "autocarro 1", "auto 2"]

   Considere o método com a seguinte assinatura:

   ```
   List<String> renomear(List<String> lst)
   ```

   Nota: Será valorizada uma resolução o mais eficiente possível

3.5 p^{tos}

2. Considere o seguinte código:

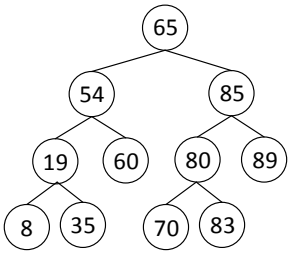
```
int mistery(int n){
   int i, j, res = 0;

   for (i = n / 2; i <= n; i++)
         for (j = 2; j <= n; j = j * 2)
             res = res + n / 2;
   return res;
}
```

   a) Explique o que faz o método mistery acima apresentado.

   b) Analise a função quanto à sua complexidade temporal. Justifique.

5 p^{tos}

3. Adicione à classe TREE<E> um método que calcula a distância (**número mínimo de ramos**) entre dois nodos numa árvore binária de pesquisa. Por exemplo na árvore abaixo a distância entre os nodos 19 e 80 é 4 ramos, a distância entre os nodos 70 e 83 é 2 ramos.



Resolva cada exercício em folhas separadas

5 p^tos

4. Partindo de um grafo é possível generalizar um outro - **Grafo de Vizinhança Relativa** (*Relative Neighbourhood Graph*), em que cada par de vértices X e Y tem um ramo E se não existir nenhum vértice Z que esteja a uma distância de X e Y inferior à distância mínima entre X e Y. Ou seja, formalmente, existe um ramo entre X e Y se:

$$dist\_\min(X,Y) \leq \max(dist\_\min(X,Z), dist\_\min(Y,Z)), \forall Z$$

Usando a representação **map de adjacência** implemente um método que, dado um grafo, devolve outro grafo que constitui o seu Grafo de Vizinhança Relativa. O método a desenvolver deve obedecer à interface:

```
Graph<V,E> RNG(Graph<V,E> g)
```

3 p^tos

5. Implemente na classe HeapPriorityQueue<K,V> uma função que determina e retorna a ordem em que a primeira ocorrência de um dado valor é processado. Considere a seguinte assinatura:

```
public<K,V extends Comparable<V>> int orderofService(HeapPriorityQueue<K,V> heap, V value)
```