

Teste Escrito de Programação Orientada aos Objetos Licenciatura em Engenharia Informática

Ano Letivo: 2023/2024

Data: 05/01/2024 **Duração:** 90 minutos **Cotação:** 10 valores **Consulta:** 2 páginas

Pretende-se desenvolver uma aplicação para uma empresa que organiza caminhadas. Uma caminhada é constituída por etapas. Cada etapa tem um local de origem, um local de destino e a distância entre eles. Os locais são caracterizados pelo nome e densidade populacional, que pode ser baixa, média ou alta. Existem etapas culturais e etapas de montanha. As etapas culturais são caracterizadas pelo número de pontos de interesse a visitar e as etapas de montanha são caracterizadas pelo declive acumulado.

- a) (2 valores) Utilizando UML, desenhe o diagrama de classes representativo do problema descrito, incluindo uma classe para gerir o sistema. Deve identificar as classes e as relações entre classes, incluindo também a cardinalidade. Em cada classe deve indicar apenas o nome da classe e os atributos. Para cada atributo deve indicar o tipo e o nível de proteção adequado (por exemplo: `-nomeLocal: String`).
- b) (2 valores) Pretende-se encontrar a etapa mais longa de uma caminhada. Desenvolva todos os métodos necessários para realizar esta operação. Indique a classe em que cada um dos métodos deve ser colocado.
- c) (2 valores) Uma caminhada é considerada como circular se começar e terminar no mesmo local. Desenvolva todos os métodos necessários para verificar se uma caminhada é circular. Indique a classe em que cada um dos métodos deve ser colocado.
- d) (4 valores) Pretende-se verificar se duas etapas são iguais. Duas etapas apenas são consideradas iguais quando todas as suas características forem iguais. Desenvolva todos os métodos necessários para realizar esta operação. Indique a classe em que cada um dos métodos deve ser colocado.

Importante:

- Não é necessário pedir dados ao utilizador.
- Não é necessário desenvolver construtores.
- Deve apenas desenvolver os métodos de acesso necessários à resolução dos problemas.
- Serão penalizadas soluções que não sigam os princípios da programação orientada aos objetos e os conceitos de herança e polimorfismo.
- Não é permitido utilizar operadores e métodos que permitam determinar a classe de um objeto (por exemplo, `instanceOf` ou `getClass()`).