 <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Tipo de Prova Trabalho prático – Época de Normal	Ano letivo 2021/2022	Data
	Curso Licenciatura em Engenharia Informática Licenciatura em Segurança Informática em Redes de Computadores	Data entrega	
	Unidade Curricular Estruturas de dados		

Observações

Este trabalho destina-se a todos os estudantes inscritos na unidade curricular de **Estruturas de Dados (ED)** e irá servir para avaliar a respetiva componente prática. Os estudantes deverão juntar-se em grupos de **2 elementos** de modo a dividir, da melhor forma, as tarefas definidas neste trabalho. Excecionalmente, e quando se justifique, poderão ser considerados grupos com outro número de elementos.

Objetivos

- Utilizar os conhecimentos sobre estruturas de dados para escolher as estruturas de dados que melhor se aplicam à resolução do problema proposto;
- Conceber e implementar, eficaz e eficientemente, o algoritmo de resolução do problema proposto.

Implementação


- Deverá ser usada a linguagem Java;
- O código deverá estar comentado através do JavaDoc;
- Não pode ser usada nenhuma coleção da plataforma de coleções do Java, sempre que for necessário terá de seleccionar a estrutura de dados com o comportamento desejado desenvolvida durante as aulas (cada grupo deverá usar as **suas** versões).

Resumo

Pretende-se que seja desenvolvida uma *Application Programming Interface* (API) para a gestão e planeamento de entrega de mercadorias de uma empresa. Considere o seguinte problema: Uma empresa tem vários vendedores tendo cada um deles de visitar um conjunto pré-definido de mercados para satisfazer a procura da mercadoria que transporta. Por sua vez, cada mercado possui um conjunto de clientes que são servidos sequencialmente. Cada vendedor possui uma capacidade máxima de mercadoria que pode transportar, quando a mercadoria acaba ou é insuficiente para satisfazer um dos clientes, o vendedor tem que se deslocar para um armazém de forma a restabelecer. Este processo repete-se até que seja satisfeita a procura de todos os clientes dos mercados que lhe foram atribuídos no início da volta.

A API deverá suportar:

- Gestão da empresa
 - Deve ser possível criar e alterar a informação relativa aos vendedores.
 - Deve ser possível criar e alterar a informação relativa aos mercados.
 - Deve ser possível criar e alterar a informação relativa aos armazéns.
 - Deve ser possível criar e alterar a informação relativa aos caminhos existentes entre locais.
 - Os locais podem conter mercados e armazéns. Deverá existir um local que identifique a empresa, ou seja, o ponto de partida para todos os vendedores.
 - Listar vendedores, mercados, armazéns e respetivos caminhos segundo vários critérios (por exemplo, ordenados por um determinado parâmetro).
 - Deve ser possível exportar um ficheiro JSON com toda informação relativa a uma empresa.
- Gestão de vendedores
 - Deve ser possível atribuir uma lista de mercados a visitar.
 - Definir a capacidade máxima que um vendedor pode transportar (em kg).

 ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO	Tipo de Prova Trabalho prático – Época de Normal	Ano letivo 2021/2022	Data
	Curso Licenciatura em Engenharia Informática Licenciatura em Segurança Informática em Redes de Computadores	Data entrega	
	Unidade Curricular Estruturas de dados		


- Deve ser possível listar toda informação relativa a um vendedor, incluindo a lista de mercados a visitar.
- Deve ser possível exportar para ficheiro JSON toda a informação relativa a um vendedor.
- Gestão de mercados.
 - Definir uma lista de clientes, caracterizados essencialmente por um valor que corresponde à procura.
 - Os clientes são servidos pela ordem com que foram inseridos.
 - Depois de satisfeita a procura, o cliente sai do mercado.
 - Deve ser possível listar toda informação relativa aos clientes do mercado.
 - Deve ser possível exportar para ficheiro JSON toda informação relativa a um mercado.
- Gestão de armazéns
 - Deverá ser possível atribuir a capacidade máxima e quantidade atual de mercadoria disponível.
 - Deverá ser possível listar todos os armazéns.
 - Deve ser possível exportar para ficheiro JSON toda informação relativa a um armazém.
- Gestão de rotas
 - Deverá ser possível gerar uma rota para cada vendedor (considerando o caminho mais curto)
 - Quando um vendedor tem de reabastecer, deverá ser gerada uma rota que identifica o caminho mais curto entre a posição do vendedor e do armazém com capacidade atual para recarregar a mercadoria. Quando não for possível satisfazer os clientes, a rota é dada como concluída.

Implementação

Deverá implementar um programa que permita testar a API.

Deverá ser possível importar os dados do problema utilizando o seguinte formato JSON:

```
{
  "vendedores": [
    {
      "id": 1,
      "nome": "Vendedor 1",
      "capacidade": 250,
      "mercados_a_visitar": [
        "Mercado 1",
        "..."
      ]
    },
    {
      "id": 2,
      "nome": "Vendedor 2",
      "capacidade": 200,
      "mercados_a_visitar": [
        "..."
      ]
    }
  ],
  "locais": [
    {
      "nome": "Empresa",
      "tipo": "Sede"
    },
    {
      "nome": "Mercado 1",
      "tipo": "Mercado",
      "clientes": [
        40, 50, 25, 20, 80
      ]
    }
  ]
}
```

 <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Tipo de Prova Trabalho prático – Época de Normal	Ano letivo 2021/2022	Data
	Curso Licenciatura em Engenharia Informática Licenciatura em Segurança Informática em Redes de Computadores	Data entrega	
	Unidade Curricular Estruturas de dados		

```

    },
    {
        "nome": "Armazém 1",
        "tipo": "Armazém",
        "capacidade": 500,
        "stock": 420
    }
],
"caminhos": [
    {
        "de": "Empresa",
        "para": "Mercado 1",
        "distancia": 30
    },
    {
        "de": "Mercado 1",
        "para": "Mercado 2",
        "distancia": 30
    },
    {
        "de": "Empresa",
        "para": "Armazém 1",
        "distancia": 30
    },
    {
        "de": "Mercado 1",
        "para": "Armazém 1",
        "distancia": 30
    }
]
}

```

Adicionalmente, também deverá ser possível introduzir os dados manualmente, podendo as quantidades (número de clientes, valor da procura de cada cliente e quantidade existente em cada armazém) serem geradas aleatoriamente.

Avaliação

- Apenas serão considerados para avaliação os trabalhos entregues antes da data-limite definida pelos docentes da UC e disponibilizada no Moodle. A não submissão do trabalho até esta data invalida a sua avaliação;
- A defesa é obrigatória e será realizada no dia do exame da época correspondente (ver calendário de exames). **A não comparência de um membro do grupo não invalida a defesa dos restantes;**
- Critérios de avaliação:
 - A escolha apropriada das estruturas de dados e o uso destas será o fator de avaliação preponderante em todas as funcionalidades implementadas.
 - Boas práticas:
 - Comentários e Javadoc.
 - Uso de controlo de versões (desde o início do projeto).
 - Teste unitários.
 - Uso das convenções do Java (ex.: <https://www.geeksforgeeks.org/java-naming-conventions/>).
 - Usabilidade.