 <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Tipo de Prova Trabalho prático – Avaliação durante período letivo	Ano letivo 2023/2024	Data
	Curso Licenciatura em Engenharia Informática / Licenciatura em Segurança Informática em Redes de computadores	Hora	
	Unidade Curricular Paradigmas de Programação	Duração	

Objetivos

Com a realização do trabalho prático, pretende-se que os alunos ponham em prática todos os conhecimentos adquiridos na utilização do Paradigma de Programação Orientado a Objetos (POO) e a sua implementação na linguagem de programação Java, demonstrando as suas competências em:

- Conhecer e compreender os conceitos fundamentais associada à POO;
- Conceber e implementar, para problemas concretos, soluções que tenham por base a POO.
- Reconhecer e compreender a semântica e a sintaxe da linguagem Java.
- Reutilizar, alterar e desenvolver código recorrendo à linguagem Java tendo em vista um determinado problema com regras semânticas específicas.

Considere ainda que:

- Não é permitida a utilização de API's/conceitos Java que não tenham sido alvo de lecionação no ano letivo corrente na unidade curricular de Paradigmas de Programação (LEI e LSIRC). Os alunos que pretendam utilizar API's adicionais devem atempadamente pedir autorização a um dos docentes da unidade curricular.
- **Não é permitida** a utilização de coleções Java predefinidas (*Java Collections Framework*) nem qualquer API não lecionada ou não autorizada previamente por um dos docentes.
- Os recursos de suporte ao trabalho referenciados no enunciado, são de utilização **obrigatória**.

Introdução

Uma instituição de ajuda humanitária decidiu implementar um projeto para a recolha de bens essenciais. Desde alimentos e roupas até medicamentos, esses itens são fundamentais para ajudar a reconstruir vidas despedaçadas por desastres naturais, conflitos ou pobreza extrema.


Através de um conjunto de parceiros, um conjunto de caixas de suprimentos foram instaladas em diversas zonas de uma determinada região onde podem ser entregues bens, que posteriormente são recolhidos. Cada caixa de suprimentos pode possuir diferentes contentores associados a um tipo de item: Alimentos não perecíveis (como arroz ou enlatados), Alimentos perecíveis (como fruta ou congelados), Vestuário ou Medicamentos. Os contentores estão equipados com sensores que comunicam diariamente a sua lotação, e que está relacionada com a sua capacidade de armazenamento. Os alimentos perecíveis têm de ser recolhidos rapidamente independentemente da lotação, enquanto que os restantes podem aguardar até um os contentores fiquem cheios ou com uma lotação elevada.

Utilizando veículos equipados para o efeito, a instituição realiza frequentemente a recolha de contentores baseado nas leituras recebidas por cada uma das caixas de suprimentos. A ideia do projeto passa por disponibilizar esta informação publicamente de forma a que possa recolher os bens de forma mais rápida e eficiente.

Descrição técnica

As caixas de suprimentos (*AidBox*) são compostas por contentores (*Container*) de determinado tipo (*ContainerType*): Alimentos Perecível (*Perishable food*), Alimentos não Perecível (*Non-perishable food*), Vestuário (*Clothing*) e Medicamentos (*Medicine*). Considerando uma determinada rota (*Route*), devem ser realizadas recolhas aos contentores tendo por base as medições (*Measurement*) recebidas por contentor.

Existem ainda veículos (*Vehicle*) que apenas podem recolher um tipo de item (*ItemType*) de cada vez (um veículo não pode recolher, por exemplo, alimentos e vestuário), possuindo sempre uma carga máxima que conseguem transportar. Para a recolha de alimentos perecíveis, existem veículos específicos (*RefrigeratedVehicles*) que estão equipados com câmaras de refrigeração para garantir que os alimentos não perdem qualidades essenciais ao seu consumo. Este tipo de veículos não deve circular mais do que um determinado número de quilómetros com carga. Note que pode ser necessário realizar mais do que um caminho entre o ponto de partida e o destino para recolher todos os itens. A recolha deve ser despoletada de acordo com um critério definido por cada grupo. Em cada leitura/recolha (*Collection*) é necessário identificar o veículo envolvido e a respetiva rota (*Route*) realizadas.

 <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Tipo de Prova Trabalho prático – Avaliação durante período letivo	Ano letivo 2023/2024	Data
	Curso Licenciatura em Engenharia Informática / Licenciatura em Segurança Informática em Redes de computadores	Hora	
	Unidade Curricular Paradigmas de Programação	Duração	

Como suporte ao desenvolvimento da API, são disponibilizados um conjunto de recursos (*MA02_Resources*), de **utilização obrigatória** e que definem os contratos e definem o ponto de partida para o desenvolvimento da API. Os conteúdos fornecidos são um complemento ao presente enunciado, contendo informação adicional e específica sobre as particularidades de implementação. Deverá realizar a implementação do código necessário para suportar cada uma das operações definidas nos contratos. A existência dos contratos não deve ser impeditiva para a implementação de novos contratos, funcionalidades e/ou novos métodos ou classes. A utilização dos contratos constitui um ponto de partida, cujos ficheiros não podem ser alterados. **Caso não utilize os recursos disponibilizados, todo o trabalho é invalidado.**

Dados fornecidos

Como suporte ao trabalho prático, é disponibilizada uma *Web API* que disponibiliza:

- As caixas de suprimentos existentes (incluindo a sua localização) e respetivos contentores. O valor da capacidade máxima dos contentores é definido na unidade de medida: kg. Para obter esta informação é necessário realizar um pedido *web* a um dos seguintes endereços:
 - <https://data.mongodb-api.com/app/data-docuz/endpoint/aidboxes>: Devolve uma string em JSON com lista das caixas de suprimentos e respetivos contentores.
 - <https://data.mongodb-api.com/app/data-docuz/endpoint/aidboxesbyid?codigo={codigo}>: Devolve uma string em JSON com a caixa de suprimentos selecionada. **{codigo}** deverá ser substituído pelo código da caixa de suprimentos.
- As distâncias entre as caixas de suprimentos. Para a distância, é utilizada a unidade: metros e para a duração é utilizada a unidade: segundos. Na definição das distâncias existe a “Base” que representa o ponto de partida e de chegada dos veículos de recolha. Para obter esta informação é necessário realizar um pedido *web* para o endereço: <https://data.mongodb-api.com/app/data-docuz/endpoint/distances?from={codigoOrigem}&to={codigoDestino}>, onde **{codigoOrigem}** é o código de origem da caixa de suprimentos e **{codigoDestino}** representa a caixa de suprimentos de destino.
- Dados de leitura dos sensores num dado instante. Em cada hora a API irá atualizar os dados das leituras. Para o valor lido, é utilizada a unidade de medida: kg. Para obter esta informação é necessário realizar um pedido *web* para o endereço: <https://data.mongodb-api.com/app/data-docuz/endpoint/readings>. Os dados são atualizados a cada 12 horas de forma a que sejam realizadas duas recolhas diárias pelos veículos da empresa.


A lógica de interação com a API encontra-se encapsulada na classe: *HTTPProvider*. A classe utiliza a biblioteca *okhttp3*¹. A *Web API* devolve os dados sobre cada um dos itens descritos no formato JSON (cada pedido *web* gera um documento diferente). Como tal, cada grupo deverá realizar as ações necessárias para interpretar os dados recebidos e instanciar os respetivos objetos de acordo com os requisitos estabelecidos nos recursos de suporte ao trabalho prático. Aconselha-se a utilização da biblioteca Java *json-simple*², para facilitar a interpretação de documentos JSON.

Notas:

- Como suporte ao trabalho são disponibilizados ficheiros representativos de uma resposta típica de cada um dos pedidos realizados a cada um dos url mencionados. Deve primeiro codificar toda a lógica baseando-se nos ficheiros JSON e só depois testar com a API.
- Pode (e deve) testar a API antes de incorporar a informação disponibilizada em JSON. Pode também criar outros documentos JSON mais simples para testar a API.

¹ <https://mvnrepository.com/artifact/com.squareup.okhttp3/okhttp/4.12.0>

² <https://code.google.com/archive/p/json-simple/>

 <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Tipo de Prova Trabalho prático – Avaliação durante período letivo	Ano letivo 2023/2024	Data
	Curso Licenciatura em Engenharia Informática / Licenciatura em Segurança Informática em Redes de computadores	Hora	
	Unidade Curricular Paradigmas de Programação	Duração	

Requisitos adicionais

Para além dos requisitos apresentados, deverá incorporar a API desenvolvida numa aplicação funcional que permita ao utilizador aceder a todas as funcionalidades apresentados nos recursos de suporte ao trabalho e no presente enunciado.

Teste o mais exaustivamente possível o código que desenvolveu como resposta aos requisitos apresentados. Recorra a comentários JavaDoc e não só de modo a documentar, o mais exaustivamente possível, o código que desenvolveu.

Elaboração do trabalho

Este trabalho é realizado em grupo que deverá ser composto por, no máximo, 2 alunos da unidade curricular.

Datas e considerações

Os alunos devem comunicar o seu grupo de trabalho através da página da unidade curricular de Paradigmas de Programação na plataforma Moodle³, até à data constante nessa plataforma.

O trabalho deve ser entregue através da página da unidade curricular de Paradigmas de Programação na plataforma Moodle até à data constante nessa plataforma. **Não são aceites trabalhos entregues depois da hora definida.**

A data de defesa do trabalho será publicada na plataforma Moodle na página da unidade curricular de Paradigmas de Programação. A defesa será realizada individualmente por cada elemento do grupo. Considera-se por defesa satisfatória, quando o aluno demonstra que realizou o trabalho submetido e que domina todos os conceitos de programação orientada a objetos aplicados na resolução do trabalho. Tentativas de fraude (incluindo a utilização de Inteligência Artificial na realização do trabalho), resultarão na avaliação do trabalho como: **Fraude Académica**.

Formato da entrega

Os trabalhos entregues deverão evitar (se possível) utilizar caminhos absolutos ou endereços específicos, de modo que possam ser facilmente utilizados em qualquer máquina. Para além disso, e no sentido de facilitar a receção dos vários trabalhos recebidos, estes deverão observar as seguintes regras:

- Todos os elementos do grupo deverão submeter o trabalho no link respetivo (Entrega do Trabalho);
- O trabalho desenvolvido deverá ser entregue através do moodle, através da submissão de um ficheiro com o nome PP_AC_<nr_do_aluno>_<nr_do_aluno>.zip, contendo:
 - Os ficheiros criados incluindo o(s) projeto(s) do IDE Netbeans e uma pasta com a distribuição (jar) da solução proposta.
- Comentários JavaDoc, e não só, de modo a documentar, o mais exaustivamente possível, o código desenvolvido.
- Um comentário, no início de cada ficheiro de código entregue por cada grupo, com pelo menos a seguinte informação (com as adaptações óbvias para cada aluno/grupo):


```
/*
 * Nome: <Nome completo do aluno>
 * Número: <Número mecanográfico do aluno>
 * Turma: <Turma do aluno>
 *
 * Nome: <Nome completo do colega de grupo>
 * Número: <Número mecanográfico do colega de grupo>
 * Turma: <Turma do colega de grupo>
 */
```

Os alunos que não realizem a entrega do trabalho até à data/hora definida serão sujeitos a penalização ou a invalidação do trabalho.

³ <http://moodle.estg.ipp.pt>