

Programação Orientada por Objetos

Centro de Distribuição - Primeira fase

Ano Letivo: 2022/2023

Época Normal

1. Introdução

O objetivo deste projeto é o desenvolvimento de uma simulação do funcionamento de um centro de distribuição de produtos utilizando a linguagem Java e a Programação Orientada por Objetos (POO).

Neste centro de distribuição, os produtos são recebidos num local específico do armazém, são embalados e depois são transportados para os locais onde ficam armazenados. O transporte é efetuado por veículos guiados automaticamente (Automated Guided Vehicle - AGV). A distribuição dos produtos para os locais de venda segue um processo idêntico onde os produtos armazenados são retirados dos locais de armazenamento e levados para um local de recolha dentro armazém.

O projeto consiste na modelação e implementação de um conjunto de classes que permitam representar a lógica da aplicação e que usam a **consola** como forma de visualização.

Tenha particular atenção ao uso do paradigma de POO na modelação das classes, dos seus componentes e das relações estabelecidas entre as mesmas, i.e., ao correto uso dos conceitos de encapsulamento, herança, classes abstratas, polimorfismo, interfaces, maximização da coesão (responsabilidade única), minimização do acoplamento, desenho orientado por responsabilidades, etc. Uma modelação bem pensada facilitará todo o processo de desenvolvimento e manutenção.

O presente documento versa apenas a primeira fase do projeto. Em data oportuna, após entrega da primeira fase, será apresentado o enunciado da segunda fase.

Nota: A interpretação do problema faz parte integrante da avaliação. Qualquer dúvida em relação a este enunciado que não foi vista com o professor de laboratório deverá ser resolvida usando as boas práticas abordadas nas aulas.

2. Centro de Distribuição

O centro de distribuição é um espaço retangular que vai conter todos os elementos da simulação. Este espaço está dividido em posições, onde cada uma delas representa uma área quadrada de 2x2m. Os elementos existentes podem ser locais de armazenamento com prateleiras/estantes para guardar os produtos, veículos que circulam no armazem ou as paredes que limitam o armazém. Além disso, existem espaços próprios para a carga e descargas dos produtos que o armazém contém e zonas em frentes aos locais de armazenamento internos para depositar e recolher os produtos.

2.1 Produtos

Para esta simulação, os produtos são muito simples caracterizando-se apenas por um nome, um identificador único, o seu peso e o seu tipo. Neste caso, vão-se considerar os seguintes tipos de produtos: roupa, acessórios, brinquedos pequenos e grandes, equipamentos eletrónicos de pequena e grande dimensão e livros.

Os produtos recebidos são normalmente embalados antes de serem armazenados. Sendo assim, foram definidos os seguintes tipos de embalagens:

- **Saco** – usado para guardar um produto leve de pequena dimensão, nomeadamente acessórios, brinquedos ou equipamentos pequenos e roupa;
- **Caixa** - usada para guardar um produto tal como os sacos com a diferença que não é utilizada para guardar roupa, mas pode ser utilizada para guardar um livro;
- **Caixa de cartão** – permite guardar até 10 produtos que não sejam de grande dimensão. Em alternativa, pode ser usada para guardar um único produto de grande dimensão como sejam um brinquedo ou equipamento eletrónico grande.
- **Palete** - pode acondicionar várias caixas de cartão.

Todos os tipos de embalagem têm um código inteiro que os identifica univocamente. A identificação das embalagens é feito por uma abreviatura para o tipo da embalagem e pelo seu código.

2.2 Locais de Recolha e Entrega

Cada armazém tem dois locais separados para a recolha e entrega de produtos a partir do exterior. Estes locais estão situados em paredes opostas do armazém ocupando 50% da espaço dessa parede. Por exemplo, pode existir na parede da esquerda um local de entrega dos produtos para armazenamento e na parede da direita um local de recolha de produtos do armazém. Este locais podem ser definidos substituindo a parede por um espaço aberto de recolha com um ponto de largura (uma linha com a espessura de uma posição como referido antes).

2.2 Locais de Armazenamento

Os locais de armazenamento são prateleiras constituídas por áreas retangulares dentro do armazém que podem guardar qualquer tipo de produto sem limite. A única restrição é que os produtos armazenados devem estar dentro de uma embalagem. Por sua vez, podem existir paletes com várias caixas de cartão com produtos. Estes locais possuem um espaço em frente a um dos lados mais longos com uma espessura de uma posição que ocupa todo o lado. Este espaço é usado pelos veículos para depositar e recolher as embalagens do respetivo local.

2.3 Veículos Guiados Automaticamente (AGV)

Os veículos são usados no interior do armazém para transportar os produtos no seu interior. Estes veículos podem seguir caminhos pré-determinados ou definirem as suas próprias rotas baseando-se no local de recolha e no local de destino do produto. Para os ajudar no trajeto, podem valer-se de diferentes tipos de sensores.

2.3.1 Tipos de Veículos

Dependendo do tipo de produto a ser transportado podem ser utilizados diferentes tipos de veículos. Sendo assim, vão-se considerar os seguintes tipos de veículos:

- **Transportador de Carga Unitária** (Unit Load Carrier - ULC) - É um AGV usado para transportar uma única paleta. Não tem limites de peso para a carga transportada;
- **Carrinho Guiado Automaticamente** (Automatic Guided Card - AGC) – É um AGV usado para transportar produtos leves, normalmente em sacos ou caixas; Tem um limite de peso total de 100 Kg.
- **Veículo Rebocador** – É um AGV que não transporta diretamente os produtos. Neste caso, é usado na simulação para rebocar um carrinho de transporte.
- **Carrinho de Transporte** - Não é um AGV. Destina-se a transportar caixas ou sacos até um ao limite de peso de 200 Kg e é puxado por um veículo rebocador.

Cada um destes veículos ocupa uma posição no armazém.

2.3.2 Movimentação dos Veículos

O movimento dos veículos dentro do armazém faz-se exclusivamente na direção horizontal ou na direção vertical. Qualquer deles se movimenta uma posição por cada passo da simulação, podendo virar ou parar. No caso específico do ULC, é possível aumentar a velocidade até ao máximo de 3 posições por passo da simulação. O aumento da velocidade deste veículo é, no máximo, de uma posição por passo da simulação. A paragem é feita imediatamente sem necessidade de desaceleração.

O ULC é o único veículo AGV que não consegue determinar automaticamente os seus trajetos. Neste caso, segue um caminho pré-determinado. Os outros veículos AGV conseguem estabelecer o seu trajeto entre duas posições de uma forma automática.

2.3.3 Tipos de Sensores

Todos os AGV descritos possuem 3 tipos de sensores com diferentes características. Sendo assim foram definidos os seguintes tipos de sensores:

- **Lidar** (Light Detection and Ranging) - É um sensor que permite determinar quais as posições do armazém que estão ocupadas ou vazias. Apenas obtém a informação até um limite de 20 metros na direção de movimento do veículo onde está montado e num ângulo de visão de 30°.
- **Sensor Ultrasónico** – É idêntico ao Lidar, mas tem um ângulo de visão de 180° e um alcance de 8 metros;
- **Câmara** – A câmara é usada para reconhecer os elementos do armazém. Consegue identificar os elementos até uma distância de 30 metros com um ângulo de visão de 90°. Não tem informação da distância dos elementos identificados e não consegue ver para além da primeira posição que está ocupada numa determinada direção do seu campo de visão. Assim, é normalmente usada com a informação do Lidar ou do sensor ultrasónico para identificar o primeiro ponto não vazio numa determinada direção. Neste caso, permite saber se o ponto corresponde a um veículo, um local de armazenamento ou uma parede. ###

3. Simulação

Para esta simulação deve ser definido um armazém. Podem ser depois adicionados veículos e locais de armazenamento ao armazém criado. Durante a simulação, serão entregues produtos que devem ser embalados e armazenados. Deve ser feita uma seleção das embalagens e dos veículos para transportar os produtos até ao seu local de armazenamento. Da mesma forma, podem ser solicitados vários produtos

entre os existentes no armazém. Neste caso, os mesmos veículos podem ser usados para recolher os produtos e deixá-los nos pontos de entrega.

Os dados da simulação deverão ser fornecidos em ficheiros no formato "CSV" que devem ser lidos pelo simulador para serem usados na preparação da simulação e durante a execução da mesma. Este ficheiros tem a informação da dimensão do armazém, a lista dos locais de armazenamento com as posições e dimensões respetivas e a lista dos veículos disponíveis. Para a simulação deve existir um ficheiro com a lista de produtos entregues para armazenamento e o passo da simulação em que isso acontece. Deverá incluir ainda a lista de produtos a recolher do armazém com a altura em que isso acontece.

Durante a simulação devem ser apresentadas na consola mensagens de registro (log) do que está a acontecer.

5. Fases de desenvolvimento e entrega

O projeto está dividido em 2 fases, com a cotação distribuída da seguinte forma:

- Fase I – 70% da avaliação final
- Fase II – 30% da avaliação final

Conforme referido na ficha da disciplina, poderá alternativamente entregar o projeto numa só data, na época de recurso (sem possibilidade de incorporar componente de avaliação contínua), contemplando as duas fases do projeto.

6. Implementação e codificação

O programa deve ser desenvolvido utilizando a linguagem Java, colocando em prática os conceitos fundamentais do paradigma de Programação Orientada por Objetos.

Em relação às regras de codificação, siga as convenções adotadas normalmente para a linguagem Java:

- A notação *camelCase* para o nome das variáveis locais e identificadores de atributos e métodos;
- A notação *PascalCase* para os nomes das classes e interfaces;
- Utilização de maiúsculas para os nomes das constantes e dos valores enumerados;
- Não utilize o símbolo '_' nos identificadores (exceto nas constantes), nem abreviaturas.

É necessário que o projeto cumpra o que é pedido no seu enunciado, sendo deixado ao critério do programador qualquer pormenor de implementação que não seja referido, o qual deverá ser devidamente documentado.

Deverá obrigatoriamente implementar: Coleções, Herança, Polimorfismo, Interfaces, Exceções e Entrada e Saída.

7. Constituição de grupos

Cada projeto deverá ser elaborado em grupos de dois alunos, podendo eventualmente ser elaborado individualmente. Não serão permitidos, em nenhum caso, grupos com mais do que dois alunos.

Os grupos dos alunos já se encontram determinados através da metodologia de *pair programming* que está a ser utilizada nos laboratórios. Caso existam alunos que não têm o grupo escolhido, deverão contactar o respetivo docente de laboratório para regularizar a situação.

8. Entrega do projeto

O projeto será entregue em duas fases:

- Uma primeira fase (**até às 08:00:00 do dia 30 de maio de 2023**) com a implementação da lógica da aplicação (descrita no presente documento);
- A segunda fase (**até às 08:00:00 do dia 26 de junho de 2023**) com a parte gráfica (descrita em documento entregue futuramente).

O projeto deverá ser entregue até à data limite especificada **por via exclusivamente eletrónica utilizando os grupos criados no Github**.

Não serão aceites quaisquer projetos entregues fora do prazo!

Todos os materiais do projeto devem ser devidamente identificados com nome, número e endereço de correio eletrónico dos alunos.

Os materiais do projeto deverão incluir:

- Um **Manual Técnico** onde conste uma breve descrição do programa, incluindo a explicação das classes/interfaces implementadas, principais atributos e métodos e suas relações.
- A documentação do programa em **JavaDoc** (não converta o documento gerado automaticamente em HTML para DOC!).
- O código fonte do programa na forma de projeto em *NetBeans* ou *IntelliJ*, com um *main* de testes a funcionar e com todas as funcionalidades implementadas.
- Todos os ficheiros que compõem o projeto deverão estar guardados num único ficheiro compactado em formato ZIP cujo nome deverá ter a seguinte nomenclatura: .zip.

9. Regras e Critérios de Avaliação do Projeto

9.1 Regras de Avaliação

A avaliação do projeto está sujeita às seguintes regras:

- Caso o aluno falte ao momento de supervisão, terá essa componente avaliada com zero valores.
- **Não serão aceites quaisquer projetos entregues fora do prazo!**
- A classificação do programa terá em conta a qualidade da programação (fatores de qualidade do software), a estrutura do código criado segundo os princípios da Programação Orientada por Objetos, tendo em conta conceitos como a coesão de classes e métodos, o grau de acoplamento entre classes e o desenho de classes orientado pela responsabilidade, e a utilização/conhecimento da linguagem Java.
- Serão premiadas a facilidade de utilização, a apresentação, a imaginação e a criatividade.
- O projeto terá uma componente de avaliação oral obrigatória com classificação individual dos elementos do grupo.

- Os alunos que não comparecerem à discussão serão classificados com zero na fase respetiva. Nesta discussão será apurada a capacidade do aluno de produzir o código apresentado. Nos casos em que essa capacidade não for demonstrada, a nota atribuída será zero.
- A avaliação oral é realizada pelo respetivo professor de laboratório e irá ser feita uma marcação prévia para cada grupo de trabalho.
- Todos os projetos serão submetidos a um sistema automático de deteção de cópias. Os projetos que forem identificados como possíveis cópias, e verificando-se serem realmente cópias, serão anulados.
- As avaliações da primeira fase do projeto serão realizadas na primeira semana de junho de 2023.

9.2 Critérios de Avaliação

A primeira fase do projeto será avaliada segundo os seguintes critérios:

Funcionalidades 45%	
<hr/>	
Implementação	35%
<hr/>	
Estrutura de classes	20%
<hr/>	
Conhecimento e boa utilização da linguagem	5%
<hr/>	
Bom estilo (nomes, comentários, indentação)	5%
<hr/>	
Definição de testes unitários	5%
<hr/>	
Documentação 10%	
<hr/>	
JavaDOC	5%
<hr/>	
Manual técnico	5%
<hr/>	
Avaliação qualitativa 10%	
<hr/>	

10. Resumo das Datas Importantes

10.1 Entrega da 1ª fase

A entrega da 1ª fase do projeto será até **às 08:00:00 de segunda-feira, dia 30 de maio de 2023.**

10.2 Avaliações da 1ª fase

As avaliações da primeira fase do projeto serão realizadas na semana de **30 de maio a 03 de junho de 2023.**