

Trabalho AEDA - Parte 1

Neste trabalho deve modelar um problema recorrendo ao paradigma da orientação por objetos e usar a linguagem C++ para implementar a solução correspondente. A aplicação a desenvolver deve permitir registar e gerir entidades, conjuntos de entidades e relações entre elas, isto é, deve permitir as operações básicas CRUD (*Create, Read, Update, Delete*).

A aplicação a desenvolver deve:

- Utilizar **classes adequadas** para representação das entidades envolvidas;
- Conter **atributos** adequados para cada classe, escolhidos de modo a caracterizar os conceitos mais importantes das entidades a representar;
- Considerar relações adequadas entre classes;
- Permitir as operações básicas **CRUD** (*Create, Read, Update, Delete*);
- Utilizar os conceitos de **herança e polimorfismo**;
- Utilizar estruturas lineares (por exemplo, vetores) para gerir conjuntos (elementos de uma mesma entidade/classe);
- Tratar convenientemente possíveis **exceções**;
- Guardar informação em **ficheiros** para uso futuro.

A aplicação deve também permitir listagens várias:

- As listagens podem ser **totais** ou **parciais** com critérios a definir pelo utilizador;
- Podem também ter **ordenações** distintas definidas pelo utilizador;
- Devem ser usados algoritmos de **pesquisa e ordenação** para este efeito.

As classes implementadas devem ser devidamente **documentadas** (usando Doxygen).

Enunciado

Desenvolva uma aplicação em C++ para gestão de uma biblioteca. O sistema deve conter informação sobre livros, leitores, funcionários e empréstimos de livros.

Um empréstimo está associado a um leitor e diz respeito a um livro. Um livro pode ser emprestado por um período máximo de 1 semana ou não pode ser emprestado.

A informação sobre livros deve incluir, pelo menos, título, autores, ISBN, cota e nº de páginas.

Cada leitor não pode ter mais de 3 livros emprestados em simultâneo. Deve ser possível notificar o leitor por telefone ou email, caso os livros não sejam devolvidos dentro do prazo. Por cada dia de atraso na devolução de um livro, o leitor incorre numa multa de 0,25Eur por dia.

Cada empréstimo é efetuado por um funcionário. Os funcionários podem ou não ser supervisores. Um supervisor é responsável por um ou mais funcionários. Um supervisor não pode ser responsável por outro supervisor. O número de funcionários sob a responsabilidade de cada um dos supervisores deve ser equilibrado.

A aplicação deve permitir registar e gerir os empréstimos efetuados na biblioteca, assim como toda a informação relativa aos livros, leitores e funcionários. Deve ser também possível consultar esta informação.

Implemente outras funcionalidades que considere relevantes, para além dos requisitos globais enunciados.

Trabalho AEDA - Parte 2

Este enunciado refere-se à segunda parte do trabalho, que complementa o implementado na primeira parte do mesmo.

A aplicação a desenvolver faz uso das estruturas de dados não lineares: árvore binária de pesquisa, fila de prioridade e tabela de dispersão. O grupo de trabalho deve efetuar as considerações que considerar necessárias para, sobre estas estruturas de dados, ilustrar:

- operações básicas **CRUD** (Create, Read, Update, Delete)
- listagens várias: **totais** ou **parciais com critérios a definir** pelo utilizador (não aplicável a filas de prioridade)

As classes implementadas devem ser devidamente **documentadas** (usando Doxygen)

Enunciado

Complemente o sistema já implementado com as seguintes funcionalidades:

- Considere que os livros não sujeitos a empréstimo são agora guardados numa árvore binária de pesquisa. Complete a informação do livro com o ano de edição do livro e implemente a ordenação da árvore por ano de edição (os mais antigos em primeiro lugar), pelo título e pelos nomes dos autores. Deve ser permitido: adicionar novos livros à biblioteca, remover livros (porque se extraviaram ou foram doados a outras bibliotecas, por exemplo), alterar informações do livro. Devem ainda ser permitidas listagens várias tirando partido da ordenação da árvore.
- Para efeitos de promoção da leitura, a biblioteca passou a guardar numa tabela de dispersão todos os seus leitores inativos, isto é, que já não realizam empréstimos de livros há mais de 12 meses. Para efeitos de envio de newsletters e convites para eventos a realizar na biblioteca, interessa manter os contatos dos leitores atualizados (email, morada e telefone). Deve ser permitido a adição de leitores inativos, remover registos dos leitores (assim que realizem um empréstimo), ou alterar os seus dados, como morada, telefone, etc. Deve ainda ser possível listar os leitores inativos.
- Para melhorar a gestão dos empréstimos de livros, a biblioteca decidiu implementar uma fila de espera quando não há exemplares disponíveis de um determinado livro. Assim os pedidos de empréstimo em espera são guardados numa fila de prioridade, em que os principais critérios de prioridade são a data do pedido e o tipo de leitor (portador de deficiência, criança com idade inferior ou igual a 12 anos e adulto). Deve ser permitido: adicionar novos pedidos de empréstimo à fila de espera, cancelar empréstimos quando o leitor deixa de estar interessado no livro ou assim que haja um exemplar disponível para empréstimo, ou alterar informações do pedido.

Implemente também outras funcionalidades que considere relevantes, para além dos requisitos globais enunciados.