

Pontifícia Universidade Católica do Rio Grande do Sul
Programação Orientada a Objetos
Prof. Marcelo H. Yamaguti
2023/1

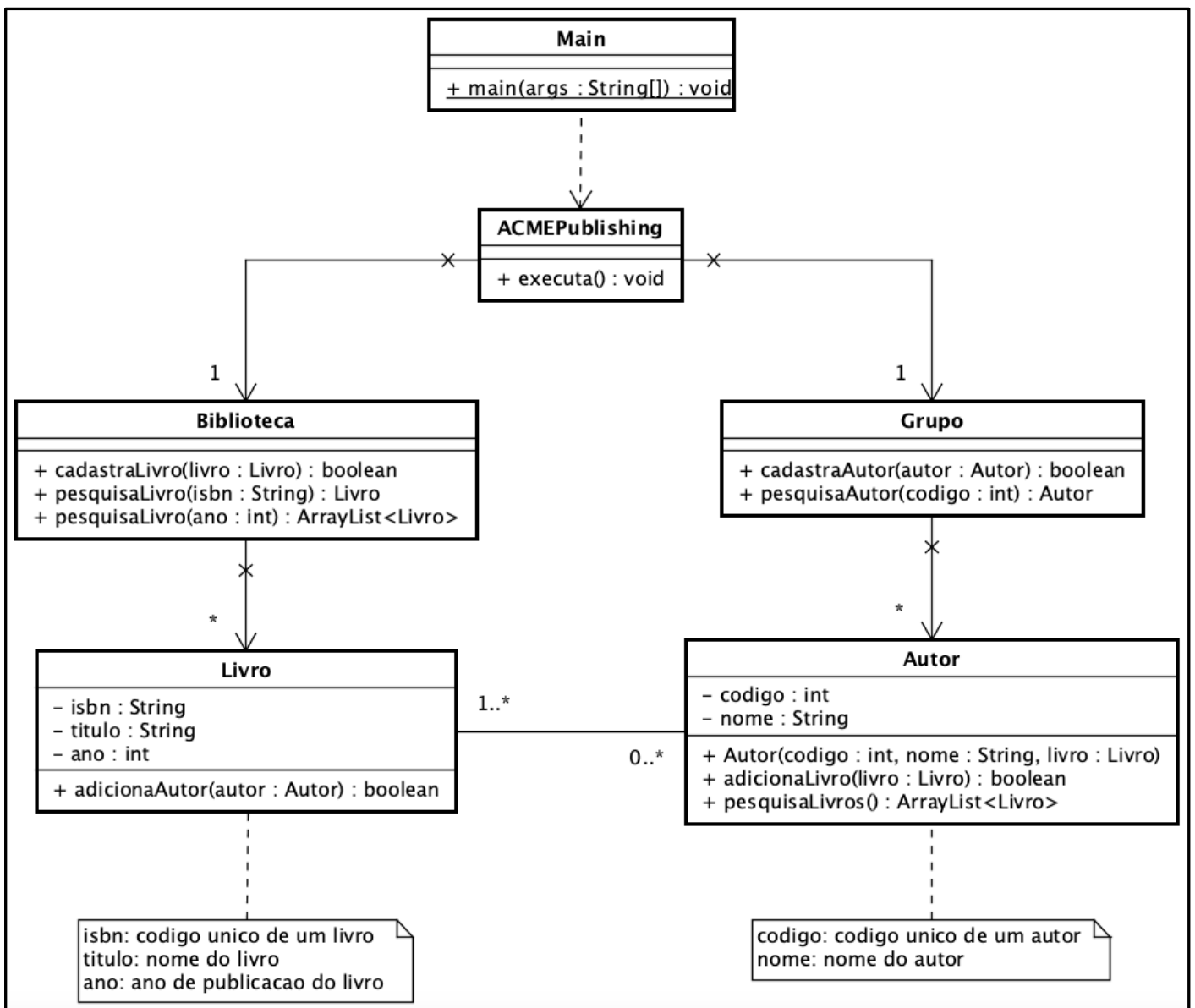
Exercício de Avaliação 1

1. Enunciado geral:

A ACMEPublishing é uma editora que gerencia livros e autores e que deseja automatizar o seu catálogo.

Você será responsável pelo desenvolvimento do aplicativo que irá catalogar livros e autores e gerar informações.

O analista de sistemas identificou as seguintes classes iniciais, com alguns atributos e relacionamentos apresentados a seguir.



O analista identificou operações básicas iniciais das classes:

- **Biblioteca**: classe catálogo que gerencia o cadastro de livros:

- **cadastraLivro(Livro)**: recebe como parâmetro um novo Livro e cadastra-o no sistema. Não pode haver Livros com o mesmo código 'isbn'. Retorna *true* se o cadastro teve sucesso; ou *false* em caso contrário.
- **pesquisaLivro(String)**: retorna o livro com o código 'isbn' indicado. Se não houver livro com este código retorna *null*.
- **pesquisaLivro(int)**: retorna uma lista de livros com o ano indicado. Se não houver nenhum livro com este ano retorna *null*.
- **Grupo**: classe catálogo que gerencia o cadastro de autores:
 - **cadastraAutor(Autor)**: recebe como parâmetro um novo Autor e cadastra-o no sistema. Não pode haver autores com o mesmo código. Retorna *true* se o cadastro teve sucesso; ou *false* em caso contrário.
 - **pesquisaAutor(int)**: retorna o autor com o código indicado. Se não houver nenhum autor com este código retorna *null*.
- **Livro**: classe que representa um livro:
 - **adicionaAutor(Autor)**: adiciona um novo autor ao livro.
- **Autor**: classe que representa um autor:
 - **Autor(int,String,Livro)**: construtor.
 - **adicionaLivro(Livro)**: adiciona um novo livro ao autor.
 - **pesquisaLivros()**: retorna uma lista de livros do autor.
- **ACMEPublishing**: classe do aplicativo:
 - **executa()**: executa o funcionamento do aplicativo.
- **Main**: classe principal (inicial) do sistema:
 - **main(String[])**: cria um objeto ACMEPublishing e depois chama o método **executa()**.

O método **executa()** da classe ACMEPublishing deve realizar a sequência de passos:

1. **Cadastrar livros**: lê todos os dados de cada livro e, se o isbn não for repetido, cadastra-o no sistema. Para cada livro cadastrado com sucesso no sistema, mostra os dados do livro no formato: **1;isbn;titulo;ano**
2. **Mostrar quantos livros foram cadastrados**: mostra a quantidade de livros que foram cadastrados no sistema no formato: **2;quantidade de livros cadastrados**
3. **Cadastrar autores**: lê todos os dados de cada autor, inclusive o isbn de um livro, e, se o código não for repetido, cadastra-o no sistema. Para cada autor cadastrado com sucesso no sistema, mostra os dados do autor no formato: **3;codigo;nome;isbn**
4. **Mostrar quantos autores foram cadastrados**: mostra a quantidade de autores que foram cadastrados no sistema no formato: **4;quantidade de autores cadastrados**
5. **Adicionar livro a um autor**: lê o código de um autor e um isbn de livro, e cadastra no sistema. Para cada livro adicionado com sucesso no autor, mostra os dados no formato: **5;codigo;nome;isbn;titulo;ano**
6. **Mostrar os livros de um determinado autor**: lê um código de autor e mostra os dados de cada livro deste autor no formato: **6;codigo;nome;isbn;titulo;ano**
7. **Mostrar os nomes dos autores de um determinado livro**: lê um isbn e mostra os nomes dos autores do livro no formato: **7;isbn;nome1;nome2;...**
8. **Mostrar os títulos dos livros que possuem mais de um autor**: para cada livro que possui mais de um autor, mostra no formato: **8;isbn;titulo**
9. **Mostrar os nomes dos autores com mais de um livro**: para cada autor com mais de um livro, mostra o nome seguido dos isbn de cada livro no formato: **9;nome;isbn1;isbn2,...**
10. **Mostrar os livros de um determinado ano**: lê um ano e mostra os dados dos livros do ano indicado, no formato: **10;isbn;titulo;ano**

2. Definição do exercício:

O objetivo do exercício é implementar um sistema que capaz de atender as necessidades da empresa descrita no enunciado geral, e que atenda as restrições a seguir:

- A entrada de dados ocorrerá por leitura de arquivo de texto. Inclua na classe ACMEPublishing o seguinte trecho de código:

```
private Scanner entrada = null; // Atributo para entrada de dados

// Construtor
public ACMEPublishing() {
    try {
        BufferedReader streamEntrada = new BufferedReader(new FileReader("dados.txt"));
        entrada = new Scanner(streamEntrada); // Usa como entrada um arquivo
        PrintStream streamSaida =
            new PrintStream(new File("saida.txt"), Charset.forName("UTF-8"));
        System.setOut(streamSaida); // Usa como saida um arquivo
    } catch (Exception e) {
        System.out.println(e);
    }
    entrada.useLocale(Locale.ENGLISH);
    // Implemente aqui o seu código adicional do construtor
}
```

- Os dados de entrada estarão no arquivo 'dados.txt' e a saída será gravada no arquivo 'saida.txt'
 - No passo **1. Cadastrar livros**: cada linha corresponde ao isbn, título e ano de um livro. Quando o isbn lido for -1, não há mais livros a serem cadastrados.
 - No passo **3. Cadastrar autores**: cada linha corresponde ao código, nome e isbn do livro de um autor. Quando o código lido for -1, não há mais autores a serem cadastrados.
 - No passo **5. Adicionar livro a um autor**: cada linha corresponde ao código do autor e o isbn do livro. Quando o código lido for -1, não há mais adições de livros a autores.
 - As últimas linhas do arquivo 'dados.txt' correspondem a:
 - Código do autor para o passo 6.
 - Isbn do livro para o passo 7.
 - Ano do livro para o passo 10.
- Toda entrada e saída de dados com o usuário deve ocorrer apenas na classe ACMEPublishing.
- É permitida a criação de novos métodos, atributos e relacionamentos, mas as informações definidas no diagrama de classes original não podem ser alteradas.
- O diagrama de classes deve ser atualizado conforme as alterações realizadas e deve ser entregue em arquivo Astah ou PDF.

3. Critérios de avaliação

O exercício será avaliado conforme os seguintes critérios:

- Diagrama de classes atualizado e coerente com a solução: 1 ponto.
- Implementação correta conforme especificação e diagrama de classes: 4 pontos.
- Execução correta dos passos solicitados: 5 pontos.
- *Ponto extra (opcional) - 1 ponto (máximo de 10 pontos): após a execução dos passos, apresentar um menu de opções cíclico, para o usuário, com as opções:*
 - [1] Cadastrar um novo autor e livro correspondente.

- [2] *Mostrar todos os autores cadastrados e livros correspondentes.*
- [0] *Sair do sistema.*

4. Entrega:

- A entrega do exercício envolverá:
 - arquivos dos códigos-fonte do sistema (e demais arquivos necessários para a compilação do sistema).
 - arquivo com o diagrama de classes atualizado.
- Deverá ser gerado um arquivo compactado (.zip ou .rar), com os itens acima, e entregue na tarefa da área Moodle da disciplina.
- **Data de entrega:** 10 / 4 / 2023.

5. Considerações finais:

- O exercício deve ser desenvolvido individualmente.
- A implementação deve seguir o Java Code Conventions para nomes de identificadores e estruturas das classes.
- Não será aceito exercício com erros de compilação. Programas que não compilarem corretamente terão nota zerada.
- A cópia parcial ou completa do exercício terá como consequência a atribuição de nota 0 (zero) aos exercícios dos alunos envolvidos. Para análise de similaridade será utilizado o MOSS (<https://theory.stanford.edu/~aiken/moss/>).