



Pontifícia Universidade Católica de Minas Gerais  
Instituto de Ciências Exatas e Informática  
Departamento de Ciência da Computação  
Disciplina: Algoritmos e Estruturas de Dados II

## Trabalho Prático III

(Data de Entrega: 06/10)

A indústria de jogos digitais movimenta bilhões de dólares anualmente, sendo considerada um dos setores mais dinâmicos e lucrativos do entretenimento moderno. Plataformas como a **Steam** concentram milhares de títulos que atendem a públicos diversos, indo de clássicos consagrados, como *Max Payne* (lançado em 2011 pela Rockstar Games, com nota 89 no Metacritic), a sucessos contemporâneos, como *Oxygen Not Included* (2019, Klei Entertainment, nota 86 no Metacritic, com mais de 3,5 milhões de jogadores estimados).

Além disso, a variedade é ampla: jogos independentes como *Reigns: Game of Thrones* (2018), títulos de grande apelo competitivo como *Call of Duty® 4: Modern Warfare®* (2007, nota 92 no Metacritic) e experiências imersivas em realidade virtual como *Ragnarock* (2021). Esses dados mostram não apenas a diversidade de gêneros — **Ação, RPG, Simulação, Indie, Casual, Esportes**, entre outros — mas também a pluralidade de estúdios envolvidos, desde pequenos desenvolvedores independentes até grandes publishers como **Activision** e **Devolver Digital**.

Neste trabalho, propomos a **análise e manipulação de dados extraídos da Steam**, explorando dimensões como preço, popularidade (número estimado de jogadores), avaliações críticas (Metacritic score), conquistas desbloqueáveis e suporte multilíngue. A partir desse conjunto de dados, buscamos identificar padrões de consumo, tendências do mercado e fatores que podem influenciar o sucesso de um jogo digital.

O conjunto de dados contém informações de jogos disponíveis na plataforma Steam. A Tabela 1 apresenta os campos disponíveis e seus respectivos tipos de dados.

Cada linha do arquivo CSV representa um jogo, com os campos separados por “##”. Alguns campos, como **Supported languages**, **Categories**, **Genres** e **Tags**, contêm listas de valores entre colchetes e separados por vírgulas. Outros campos podem estar vazios ou conter valores especiais, como "Free to Play" para o preço ou "tbd" para a nota dos usuários. A seguir, apresentamos alguns exemplos de linhas do arquivo `games.csv`:

Campo	Tipo	Descrição
AppID	Numérico	Identificador único do jogo na Steam.
Name	Texto	Nome do jogo.
Release date	Data (texto)	Data de lançamento do jogo.
Estimated owners	Numérico	Estimativa do número de jogadores que possuem o jogo.
Price	Numérico (float)	Preço do jogo em dólares americanos.
Supported languages	Texto (lista)	Idiomas suportados pelo jogo.
Metacritic score	Numérico	Nota atribuída pela crítica especializada (Metacritic).
User score	Numérico	Nota atribuída pelos usuários da Steam.
Achievements	Numérico	Quantidade de conquistas disponíveis no jogo.
Publishers	Texto	Empresa(s) responsável(is) pela publicação do jogo.
Developers	Texto	Empresa(s) ou estúdio(s) responsável(is) pelo desenvolvimento.
Categories	Texto (lista)	Categorias associadas ao jogo (ex.: single-player, multi-player).
Genres	Texto (lista)	Gêneros do jogo (ex.: Ação, RPG, Simulação).
Tags	Texto (lista)	Conjunto de palavras-chave atribuídas pelos usuários.

Tabela 1: Campos do conjunto de dados da Steam.

```

AppID ## Name ## Release date ## Estimated owners ## Price ## Supported languages
## Metacritic score ## User score ## Achievements ## Publishers ## Developers ##
Categories ## Genres ## Tags

10 ## Counter-Strike ## 01/11/2000 ## 25000000 ## 9.99 ## [English, French,
German, Italian, Spanish] ## 96 ## 9.0 ## 167 ## Valve ## Valve ## [Multi-player,
Co-op] ## [Action, Shooter] ## [FPS, Classic, Competitive]

20 ## Team Fortress Classic ## 01/04/1999 ## 15000000 ## 4.99 ## [English, French,
German, Italian, Spanish] ## 83 ## 8.5 ## 100 ## Valve ## Valve ## [Multi-player] ##
[Action, Shooter] ## [FPS, Classic, Team-Based]

```

## Questões

1. Implemente a classe `Game` em Java tomando como referência o conjunto de dados `games.csv` (Steam). A classe deve modelar um jogo com os seguintes atributos (nomes sugeridos entre parênteses), tipos e regras de normalização:

- Identificador único do jogo (`id`): inteiro.

- Nome do jogo (**name**): string.
- Data de lançamento (**releaseDate**): string no formato "dd/mm/aaaa". Caso a data esteja incompleta, utilize "01" para dia ou mês faltantes. Exemplo: "01/05/2010" ou "15/01/2005".
- Estimativa do número de jogadores (**estimatedOwners**): inteiro. Remova quaisquer caracteres não numéricos (ex.: vírgulas).
- Preço do jogo (**price**): float. Caso o preço seja "Free to Play", atribua 0.0.
- Idiomas suportados (**supportedLanguages**): array de strings. Extraia os idiomas listados entre colchetes e separados por vírgulas.
- Nota da crítica especializada (**metacriticScore**): inteiro. Caso o campo esteja vazio, atribua -1.
- Nota dos usuários (**userScore**): float. Caso o campo esteja vazio ou contenha "tbd", atribua -1.0.
- Quantidade de conquistas disponíveis (**achievements**): inteiro. Caso o campo esteja vazio, atribua 0.
- Empresa(s) responsável(is) pela publicação (**publishers**): array de strings. Separe múltiplas empresas por vírgulas.
- Empresa(s) ou estúdio(s) responsável(is) pelo desenvolvimento (**developers**): array de strings. Separe múltiplas empresas por vírgulas.
- Categorias associadas ao jogo (**categories**): array de strings. Extraia as categorias listadas entre colchetes e separadas por vírgulas.
- Gêneros do jogo (**genres**): array de strings. Extraia os gêneros listados entre colchetes e separadas por vírgulas.
- Conjunto de palavras-chave atribuídas pelos usuários (**tags**): array de strings. Extraia as tags listadas entre colchetes e separadas por vírgulas.

2. Repetir a estrutura acima usando **struct** em C.