

CK0117 - Sistemas de Bancos de Dados - 2025-1

Javam Machado, Antonio Marreiras, Gabriel Magalhães

TRABALHO II - Operadores Algébricos Junção Sort-Merge com Ordenação Externa

1 Aspectos Gerais

O trabalho consiste em implementar a operação Junção Sort/Merge (*Sort Merge Join*) da álgebra relacional. A implementação faz uso da ordenação externa das tabelas na sua versão *External merge sort*. Grave as tabelas ordenadas em arquivos do disco antes de fazer a etapa de comparação. Na implementação, desconsidere o uso de SGBDs, você deve usar apenas o sistema de arquivos do sistema operacional.

Para executar os operadores implementados, vamos utilizar o esquema apresentado na Figura 1. Isto é, o operador receberá como entrada duas tabelas presentes nesse esquema. Mais detalhes de implementação serão descritos na seção 2.

Resumidamente, observa-se, a partir do esquema apresentado, que existe uma tabela *Vinho*, que possui como chave primária (PK) *vinho_id* e duas chaves estrangeiras (FK): *uva_id* que referencia o atributo de mesmo nome da tabela *Uva* e *pais_producao_id* que referencia *pais_id* da tabela *País*. Ademais, essa mesma situação ocorre entre as tabelas *Uva* e *País*.

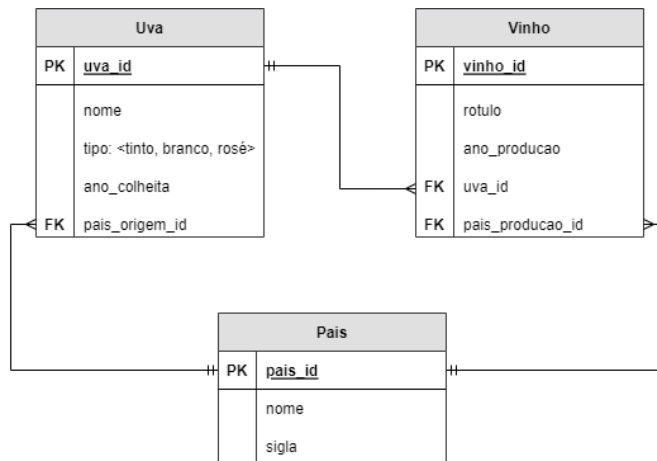


Figure 1: Modelagem E-R do esquema a ser utilizado.

2 Implementação

Cada equipe de, no máximo, dois alunos, conforme definido na tabela de equipes do Trabalho I, implementará, em linguagem **C**, **C++** ou **C#**, conforme a especificação abaixo. O resultado do operador deve conter as tuplas retornadas da junção Sort/Merge e deve ser salvo em disco. Além disso, também deve ser mostrado: (1) Quantidade de IO's (vezes que leu uma página); (2) Quantidade de páginas gravadas em disco pela operação;

(3) Quantidade de tuplas geradas na junção. **Importante:** As tabelas originais não devem ser modificadas!

Nesse trabalho, um banco de dados será mapeado em disco de modo que cada tabela terá uma quantidade não definida de páginas, de forma que cada página irá armazenar **10 tuplas, no máximo**. A estrutura da tupla é a definida no esquema da figura 1. A implementação dessas abstrações deverá utilizar as classes propostas na figura abaixo. Todas essas estruturas devem estar armazenadas em arquivos do disco em formato de texto. Portanto, uma maneira de como realizar o mapeamento dessas estruturas em arquivos deve ser definida.

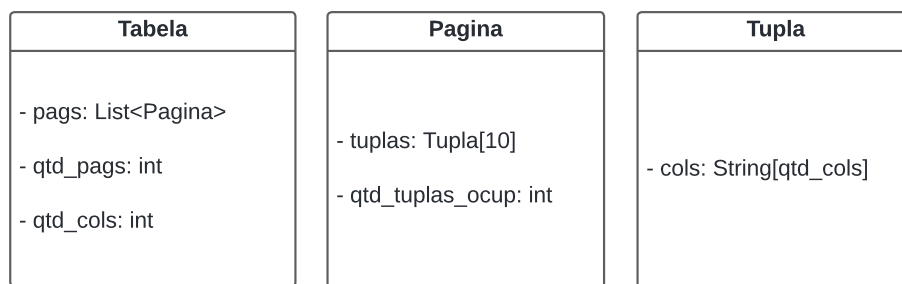


Figure 2: Sugestão de implementação por meio do diagrama de classes.

Para fazer a ordenação externa, será necessário manter **ATÉ 4 páginas, no total, em memória** e utilizar o arquivo Main, que será fornecido, na sua implementação. Esse arquivo possui as operações que devem ser implementadas, juntamente com alguns casos de teste para validar a implementação. A implementação a ser feita deve estar preparada para receber diferentes tabelas na junção em qualquer ordem. Exemplos de junções que podem ser pedidas:

- $Uva \bowtie_{uva_id=uva_id} Vinho$;
- $Vinho \bowtie_{uva_id=uva_id} Uva$;
- $Uva \bowtie_{pais_origem_id=pais_id} Pais$;

3 Entrega

Data da entrega: Sexta-Feira - 13 de junho de 2025 até as 10h00 com apresentação e arguição no LEC/DC, no horário da aula. O código do programa e os arquivos de resultado devem ser enviados pelo **classroom** até o final do horário da entrega. Envios posteriores serão penalizados. Quaisquer dúvidas podem ser enviadas aos monitores: Gabriel Magalhães (gabriel.alves@lsbd.ufc.br) ou Antonio Alves (antonio.marreiras@lsbd.ufc.br).