

Lista de Exercícios VI – Otimização de Consultas

Considere o seguinte esquema relacional "Empresa", que armazena dados referentes a empregados, departamentos e projetos de uma determinada empresa.

Lista de relações:

- Empregado (Codigo, Nome, Salario, CodSupervisor, CodDepto)
- Departamento (Codigo, Nome, CodGerente)
- Projeto (Codigo, Nome, CodDepto)
- Trabalha (CodEmpregado, CodProj, Horas)

Lista de restrições de integridade referencial:

- Empregado [CodSupervisor] → {n} Empregado [Codigo]
- Empregado [CodDepto] → {b} Departamento [Codigo]
- Departamento [CodGerente] → {b} Empregado [Codigo]
- Projeto [CodDepto] → {b} Departamento [Codigo]
- Trabalha [CodEmpregado] → {p} Empregado [Codigo]
- Trabalha [CodProj] → {p} Projeto [Codigo]

Para cada uma das seguintes consultas SQL, apresente a árvore algébrica inicial e a correspondente árvore otimizada, gerada pelas etapas "tradução" e "otimização de consulta", respectivamente, do processamento de consultas. Para a geração da árvore otimizada, devem ser considerados os passos de otimização e as regras de equivalência algébrica.

a) Recuperar o nome e o salário dos empregados que pertencem ao departamento 'Pesquisa'.

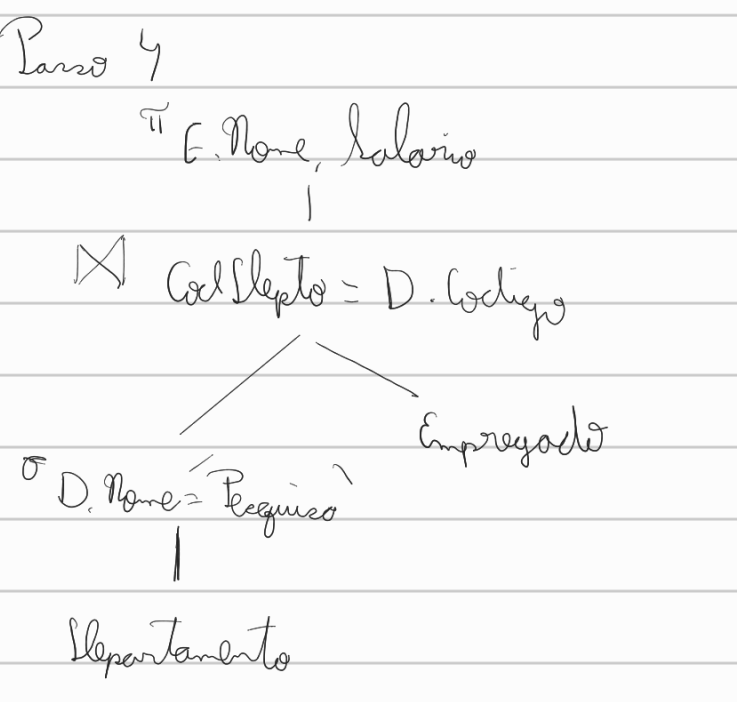
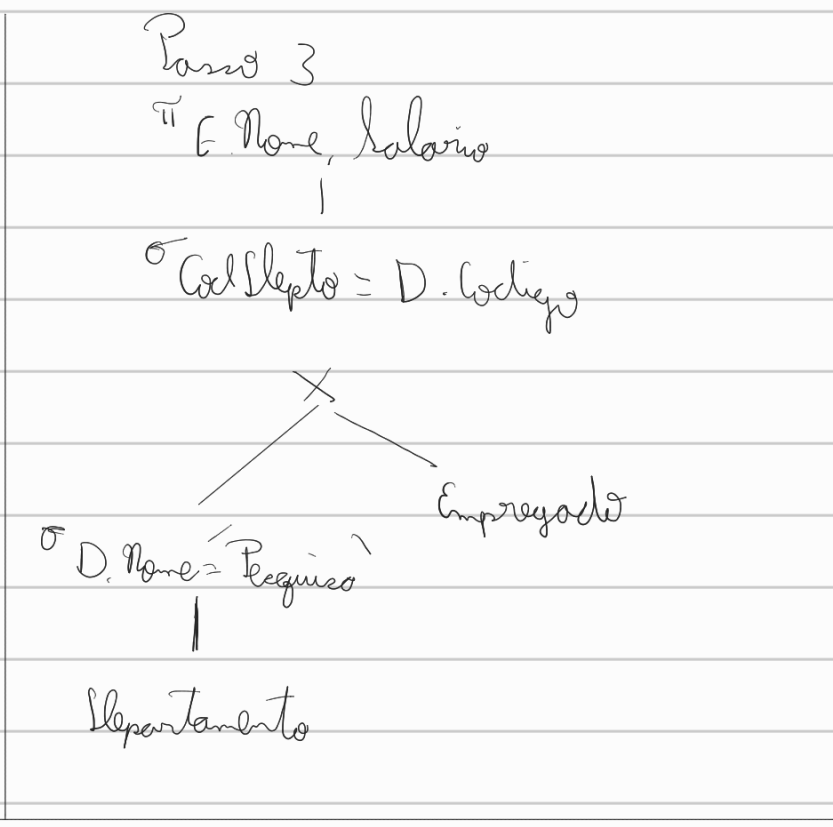
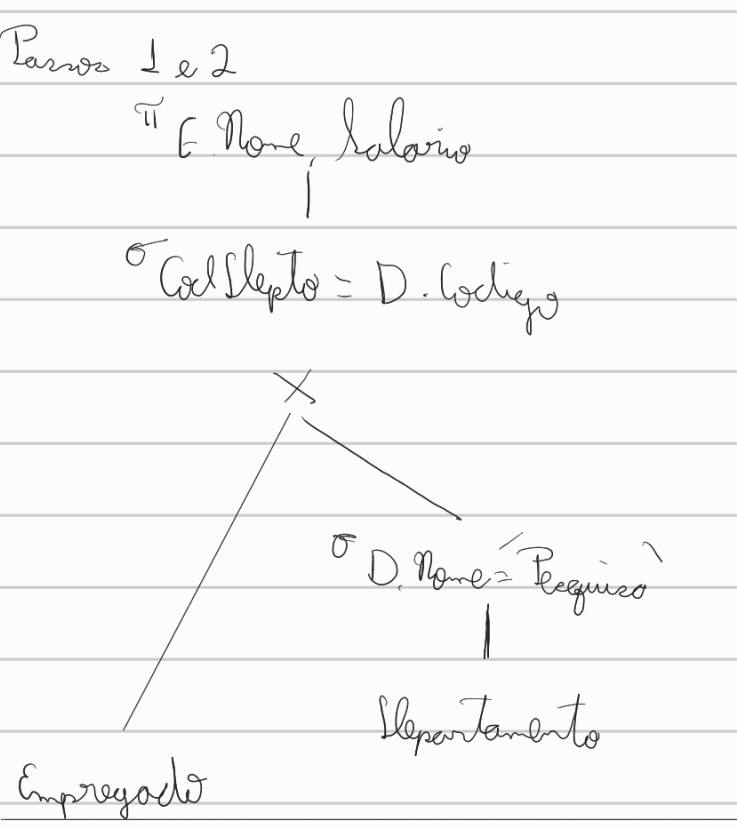
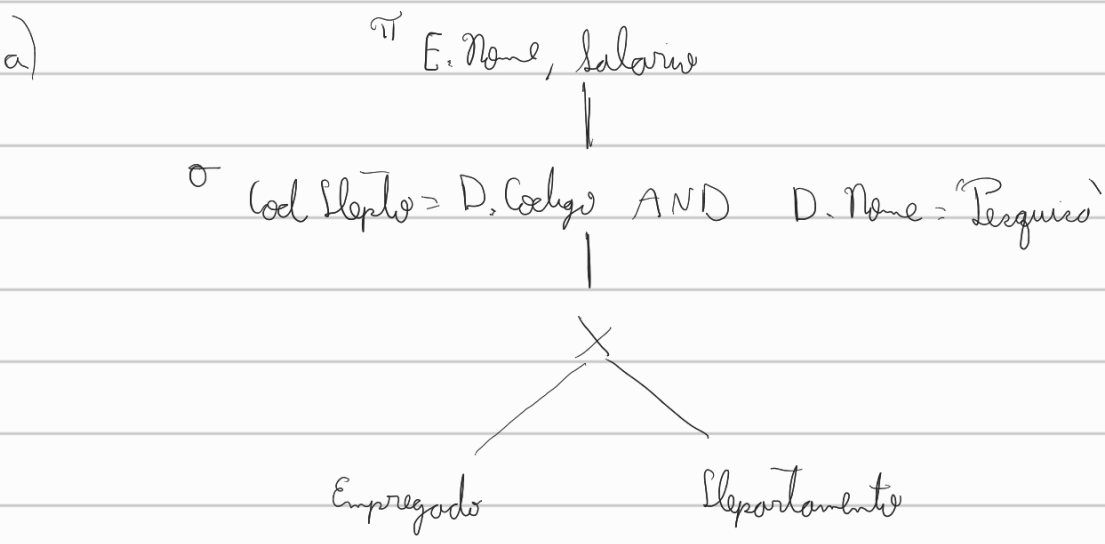
```
SELECT  E.Nome, Salario
FROM    Empregado AS E, Departamento AS D
WHERE    (CodDepto = D.Codigo) AND (D.Nome = 'Pesquisa');
```

b) Recuperar o nome dos empregados que pertencem ao departamento 'Pesquisa', ganham mais de R\$4000,00 e trabalham no projeto 'ProdutoX'.

```
SELECT  E.Nome
FROM    Empregado AS E, Departamento AS D, Trabalha AS T, Projeto AS P
WHERE    (E.CodDepto = D.Codigo) AND (E.Codigo = T.CodEmpregado) AND (P.Codigo = T.CodProj)
AND (D.Nome = 'Pesquisa') AND (Salario > 4000) AND (P.Nome = 'ProdutoX');
```

c) Recuperar os códigos dos projetos que envolvem um empregado cujo salário seja superior a R\$4000,00, como trabalhador ou como gerente de um departamento que controla o projeto.

```
SELECT  P.Codigo
FROM    Projeto AS P, Departamento AS D, Empregado AS E
WHERE    (P.CodDepto = D.Codigo) AND (D.CodGerente = E.Codigo) AND (Salario > 4000)
UNION
SELECT  T.CodProj
FROM    Trabalha AS T, Empregado AS E
WHERE    (T.CodEmpregado = E.Codigo) AND (Salario > 4000);
```



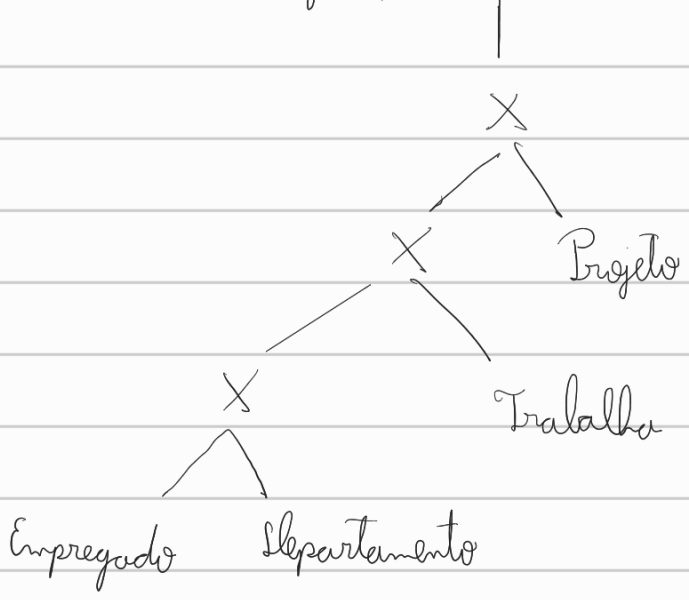
Passos 5 e 6

mantém a árvore uma vez que as projeções não podem ser quebradas

b)

π E.Nome

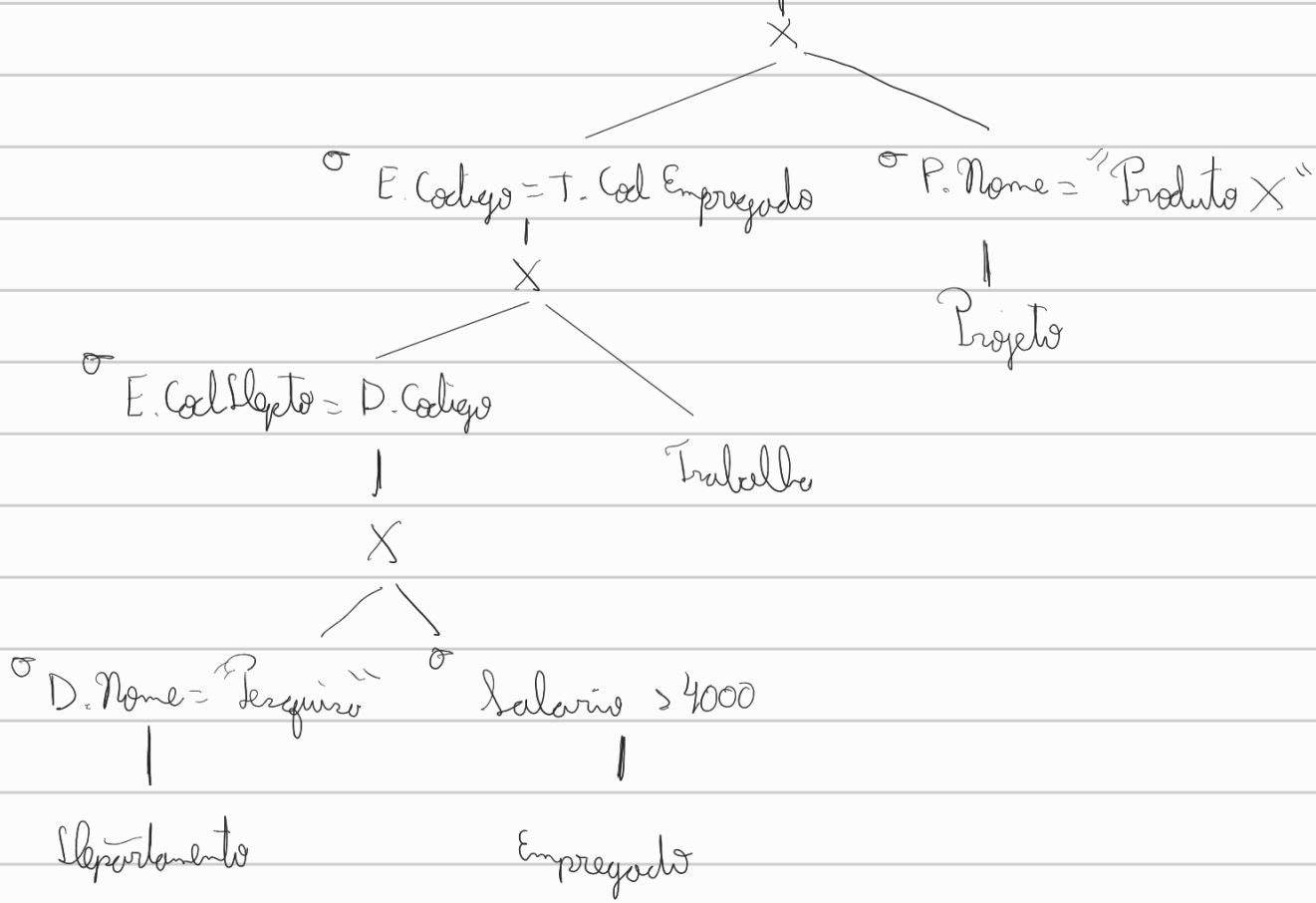
σ $(E.Cod Depto = D.Codigo) \text{ AND } (E.Codigo = T.Cod Empregado) \text{ AND } (P.Codigo = T.Cod Proj) \text{ AND } (D.Nome = \text{"Perguizo"}) \text{ AND } (Salario > 4000) \text{ AND } (P.Nome = \text{"Produto X"})$;



Passos 1 e 2

π E.Nome

σ P.Codigo = T.Cod Proj



Passo 3 - já está em ordem de mais recente

Passo 4

