

Prova de Fundamentos de Bancos de Dados

1ª Prova

Prof. Carlos A. Heuser

Abril de 2004

1. Uma agência de viagens resolveu montar uma base de dados com informações sobre vôos das companhias de aviação das quais vende bilhetes.

A agência atende várias companhias de aviação, cada qual identificada por uma sigla (por exemplo, “RG”) e tendo um nome (por exemplo, “Varig”).

Cada companhia possui vários vôos. Um vôo é identificado por uma sigla (por exemplo, “RG2324”) que inclui a sigla da companhia responsável.

Cada vôo sai de um aeroporto e tem como destino outro aeroporto. Os aeroportos são igualmente identificados por uma sigla (por exemplo “POA”) e têm um nome (por exemplo “Porto Alegre”). Cada vôo tem um horário de saída e um horário de chegada.

Há alguns vôos compartilhados. Um vôo compartilhado é um vôo de uma companhia, mas que usa, na realidade, um vôo de outra companhia. Por exemplo, o vôo “JJ4598” da TAM não existe fisicamente, mas utiliza o vôo “RG2324” da Varig. Para vôos compartilhados é necessário conhecer o vôo que eles usam.

Projete um esquema textual para a base de dados em questão. Neste esquema devem aparecer ao menos os nomes das tabelas, os nomes das colunas, as chaves primárias e as chaves estrangeiras.

Resposta:

```
Companhia (SiglaCia, NomeCia)
Voo (SiglaCia, NoVoo, SiglaAeroSaida, SiglaAeroDestino
     HoraSaida, HoraChegada)
     SiglaAeroSaida referencia Aeroporto
     SiglaAeroDestino referencia Aeroporto
     SiglaCia referencia Companhia
Aeroporto (SiglaAero, NomeAero)
```

```
VooCompart(SiglaCiaCompart,NoVooCompart,SiglaCia,NoVoo)
(SiglaCia,NoVoo) referencia Voo
```

2. Para os exercícios de álgebra e cálculo relacional, considere o seguinte esquema de um banco de dados sobre computadores e configurações:

```
Modelo(CodModelo,NomeModelo,CodCPU)
CodCPU referencia CPU
/* Tabela com os modelos de computadores que são
fabricadas */
```

```
CPU(CodCPU,NomeCPU,CodFabricante)
CodFabricante referencia Fabricante
/* Tabela com os modelos de CPUs que são usadas */
```

```
FabricanteCPU(CodFabricante,NomeFabricante)
/* Tabela com os fabricantes de CPUs */
```

```
Configuracao(CodModelo,NoConfig,NomeConfig)
CodModelo referencia Modelo
/* Tabela com as várias configurações nas quais um
modelo pode ser fabricado. Cada configuração tem
um nome */
```

```
PerifComp(CodPerifComp,NomePerifComp)
/* Tabela com os periféricos e componentes usados
na fabricação de computadores */
```

```
PerifCompConfig(CodPerifComp,CodModelo,NoConfig,Quantidade)
CodPerifComp referencia PerifComp
(CodModelo,NoConfig) referencia Configuração
/* Tabela informando que componentes e em que
quantidade são usados dentro de cada configuração*/
```

- a) Expresse as seguintes consultas em álgebra relacional:

- i. Obter o nome dos modelos cujas configurações contém ao menos um periférico chamado “Impressora jato de tinta fotográfica XYZ”.

RESPOSTA:

π NomeModelo

```

(σ NomePerifComp =
  'Impressora jato de tinta fotográfica XYZ' AND
  Modelo.CodModelo = PerifCompConfig.CodModelo AND
  PerifCompConfig.CodPerifComp =
    PerifComp.CodPerifComp
  (Modelo X PerifCompConfig X PerifComp)
)

```

ou, com junções explícitas

```

π NomeModelo
  (σ NomePerifComp =
    'Impressora jato de tinta fotográfica XYZ'
    (Modelo ⋈ (PerifCompConfig ⋈ PerifComp))
  )

```

- ii. Obter os nomes dos fabricantes cujas CPUs não são usadas em nenhum modelo.

Resposta:

```

π NomeFabricante (Fabricante)
-
π NomeFabricante
  (Fabricante ⋈ CPU ⋈ Modelo)

```

- iii. Obter uma tabela com duas colunas, nome de periférico/componente e nome de modelo. Uma linha aparece nesta tabela quando o periférico/componente for utilizado no modelo. Caso o periférico/componente não for utilizado em nenhum modelo, deve aparecer no resultado seguido de vazio.

Resposta:

```

π NomePerifComp, NomeModelo
  (PerifComp ⋈
    (PerifCompConfig ⋈ Modelo)
  )

```

ou

```

π NomePerifComp, NomeModelo
  ((PerifComp ⋈ PerifCompConfig)
   ⋈ Modelo)
  )

```

- iv. Obter os códigos dos periféricos/componentes que são usados em todos modelos.

Resposta:

```

π CodPerifCom, CodModelo (PerifCompConfig)

```

\div
 π CodModelo (Modelo)

b) Expresse as seguintes consultas em cálculo relacional:

i. Idem questão 2(a)i.

Resposta:

```
Modelo.NomeModelo WHERE
  ∃ PerifCompConfig (
    PerifCompConfig.CodModelo=Modelo.CodModelo
    AND ∃ PerifComp (
      PerifComp.CodPerifComp=
        PerifCompConfig.CodPerifComp
      AND PerifComp.NomePerifComp =
        'Impressora jato de tinta fotográfica XYZ'
    )
  )
```

ii. Idem questão 2(a)ii.

Resposta:

```
FabricanteCPU.NomeFabricante WHERE
NOT ∃ Modelo(
  ∃ CPU(
    CPU.CodCPU=Modelo.CodCPU AND
    CPU.CodFabricante=FabricanteCPU.CodFabricante
  )
)
```

iii. Idem questão 2(a)iv.

```
PerifComp.CodPerifComp WHERE
∀ Modelo (
  ∃ PerifCompConfig (
    PerifCompConfig.CodPerifComp = PerifComp.CodPerifComp
    PerifCompConfig.CodModelo = Modelo.CodModelo
  )
)
```

3. Represente a expressão de álgebra relacional abaixo na forma de uma árvore. Mostre as árvores resultantes de cada um dos passos da otimização algébrica, até chegar na árvore otimizada.

```
 $\pi$  NomeCPU
(  $\sigma$  NomeFabricante = 'Intel' AND
```

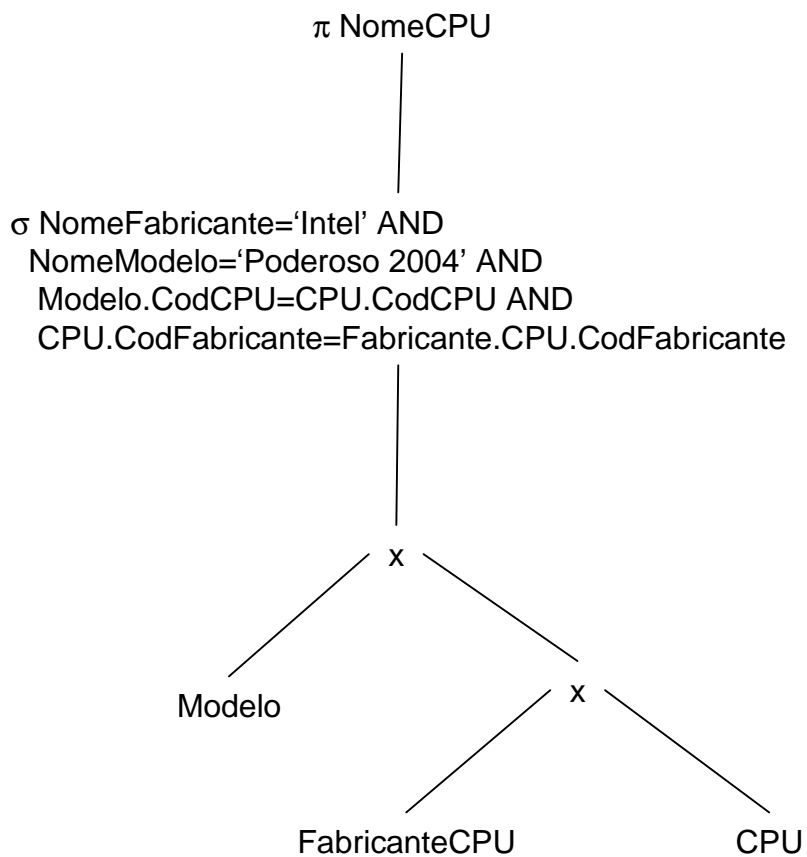


Figura 1: Passo 0: representação em árvore

```

NomeModelo = 'Poderoso 2004' AND
Modelo.CodCPU = CPU.CodCPU AND
CPU.CodFabricante = FabricanteCPU.CodFabricante
(Modelo X FabricanteCPU X CPU)
)
  
```

A resposta está apresentada nas figuras 1, 2, 3 e 4.

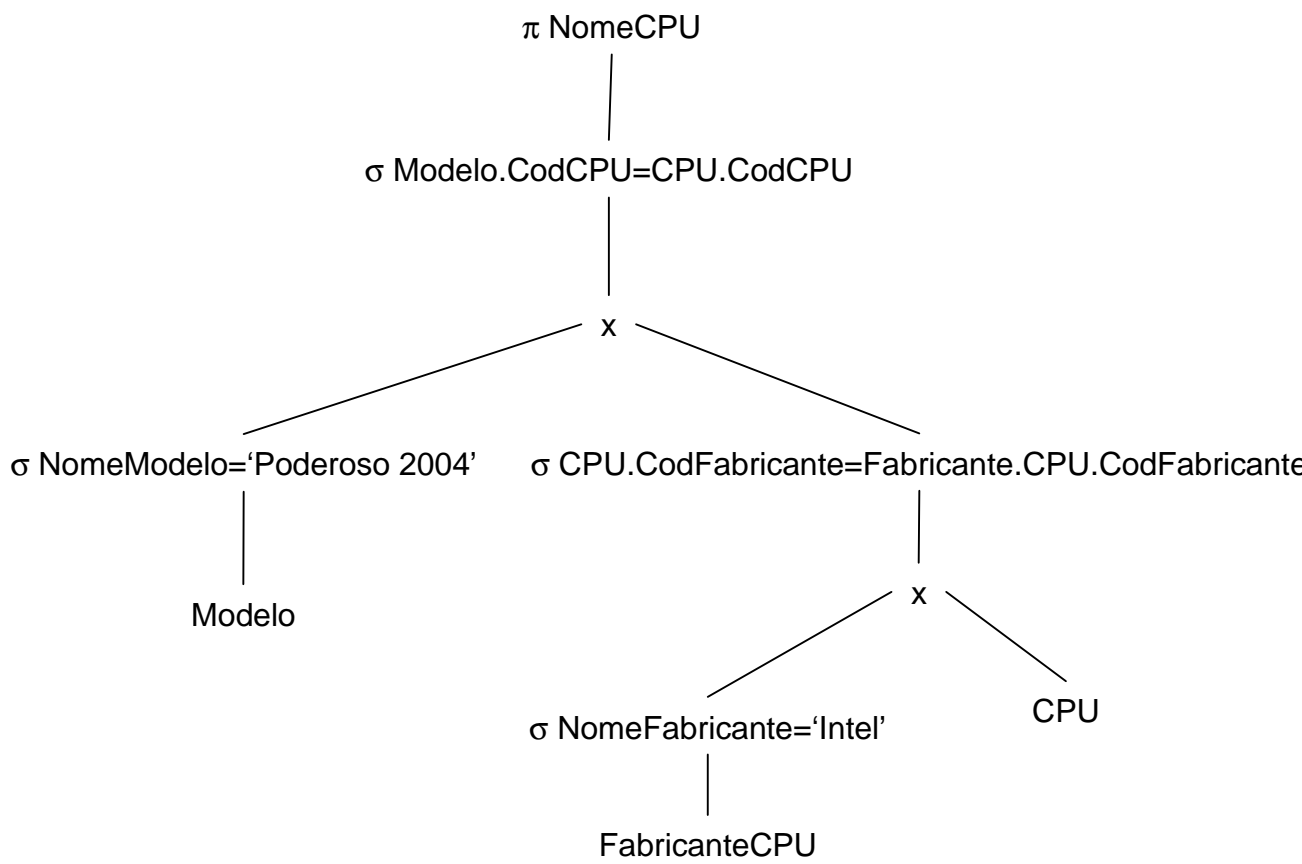


Figura 2: Passo 1: seleções são executadas mais cedo

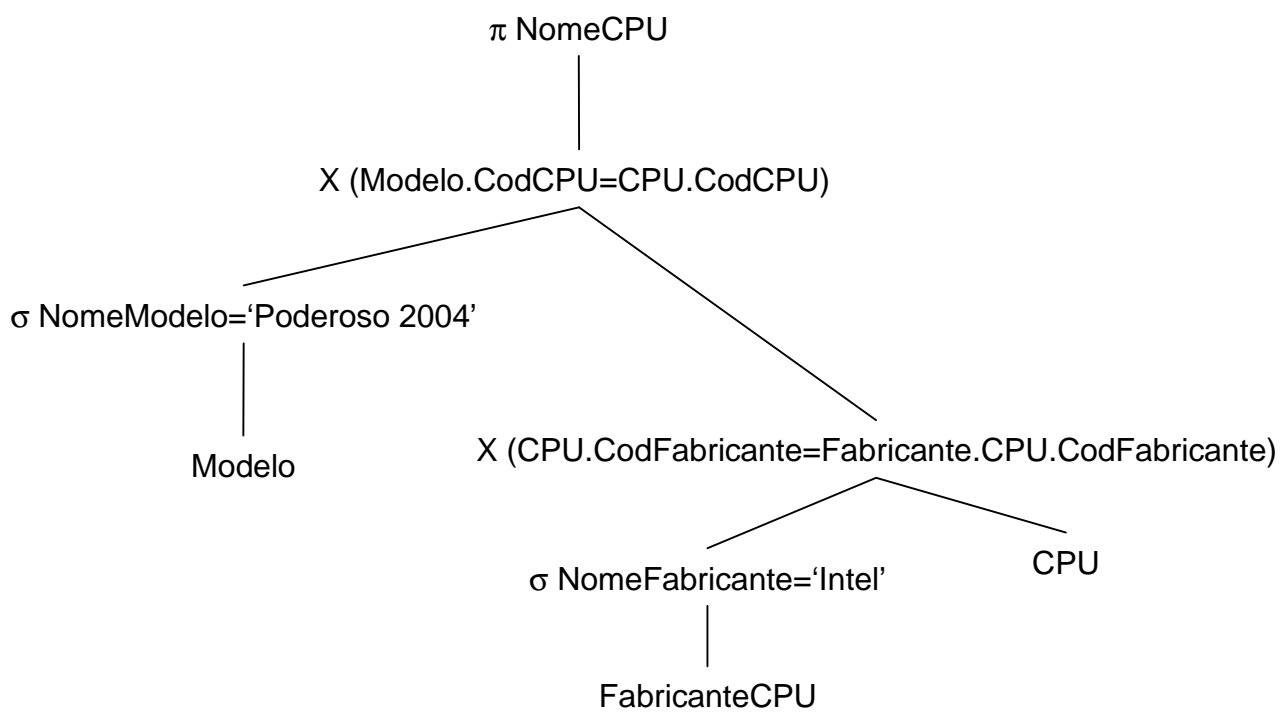


Figura 3: Passo 1: produto cartesiano seguido de seleção é convertido em junção

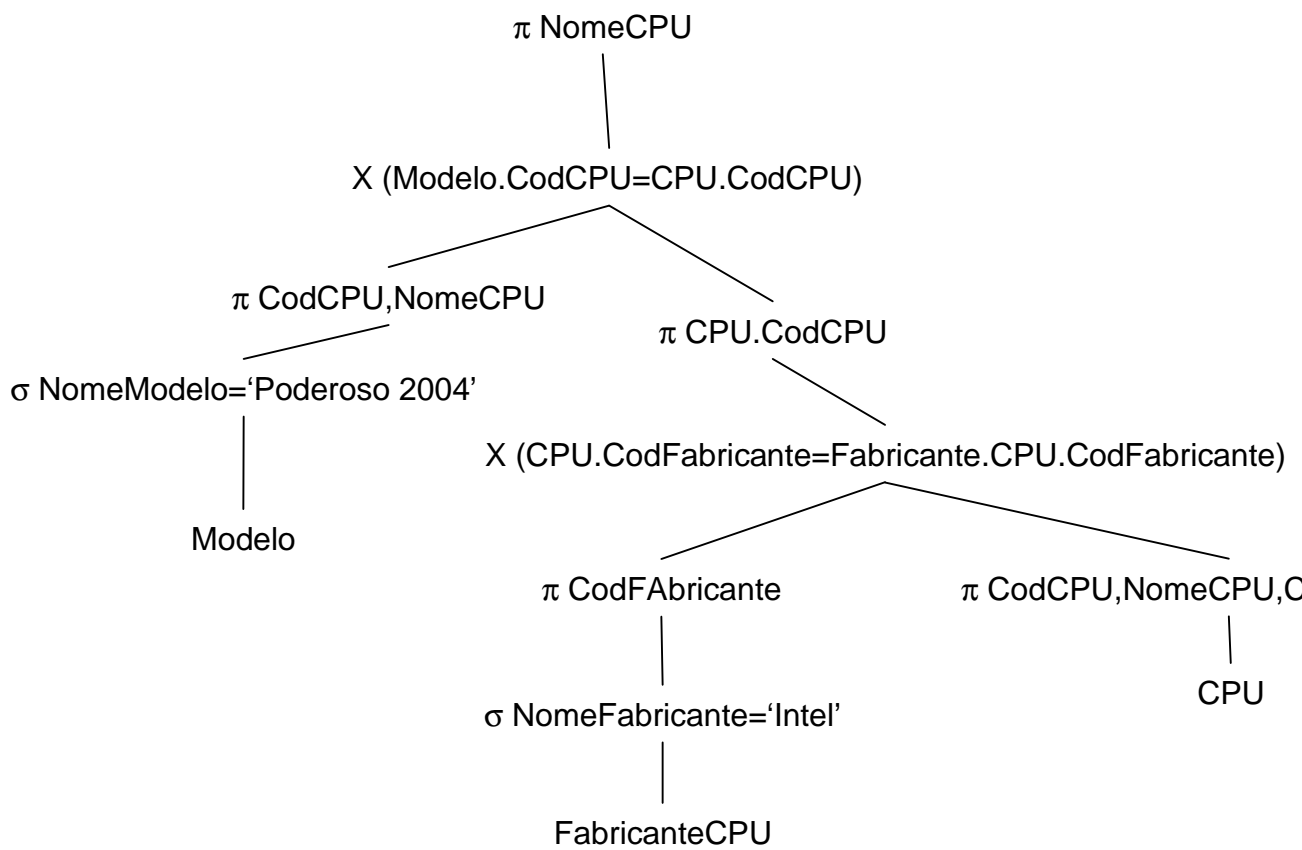


Figura 4: Passo 1: projeções são executadas mais cedo