

Universidade Federal da Fronteira Sul - UFFS
Campus Chapecó
Ciência da Computação
Banco de Dados I
Prof.: Denio Duarte

Instruções

1. Coloque sua MATRÍCULA na folha resposta.
2. A compreensão das questões faz parte da prova.
3. Responda as questões na folha resposta, enumerando as mesmas. Consulta permitida apenas a cola oficial.
4. As repostas podem ser ridigidas à lápis porém o professor se reserva a não aceitar reclamações oriundas da correção das questões.

Segunda Avaliação (A₃)

1. Dado o seguinte esquema relacional:

pessoa (cpf, nome, dtnasc, cpfpai(*pessoa*))

IRPF(cpf (*pessoa*), ano, *vlpg*, *vlrest*)

tipoitens(cod, *descr*)

itensIRPF(cpf(*IRPF*), ano(*IRPF*) *tpitem*(*tipoitens*), *vldecl*, *tribut*).

Resolva as consultas abaixo em álgebra (AR) ou Cálculo (CR):

- (a) Retorne os nomes das pessoas e nomes dos itens que foram declarados em 2017 (AR e CR). 1.0
 - (b) Retorne o nome da pessoa que teve o maior valor restituído em 2012 (AR). 1.5
 - (c) Retorne o nome do contribuinte que declarou imposto de renda em todos os anos cadastrados em IRPF. (AR) 2.0
2. Dadas a relação $R(A, B, C, D)$ e as dependências funcionais $DF=\{A \rightarrow B, C \rightarrow B, D \rightarrow A\}$, encontre todas as chaves de R utilizando o algoritmo de fechamento (limite as chaves em 2 atributos). 2.5
 3. A Empresa *My Beauty Car* possui um banco de dados com a seguinte tabela: 3.0

CPF	Nome	Modelo	Ano	Placa	Itens	DtLoc	DtDev
12345	John	Gordini	1971	AK1313	<ar,abs,airbag>	03/07/2012	10/07/2012
56789	Mary	Gordini	1971	AK1313	<ar,abs,airbag>	07/08/2012	11/08/2012
12345	John	DKV	1969	AX5555	<trava,ar>	04/09/2012	15/09/2012
13579	Clark	Fusca	1975	JJ5151	<computador,abs>	21/03/2013	

Você é responsável por normalizar as mesmas para evitar as anomalias de atualização que andam ocorrendo no sistema. A documentação do sistema apresenta as seguintes dependências funcionais válidas:

$CPF \ Placa \ DtLoc \rightarrow Nome \ Modelo \ Ano \ Itens \ DtDev$

$CPF \rightarrow Nome \quad Placa \rightarrow Nome \ Modelo \ Ano \ Itens$

Aplique a 1FN e apresente o resultado, aplique a 2FN e apresente o resultado, finalmente aplique a 3FN e apresento resultado.

Questão 1:

a) Retorne os nomes das pessoas e nomes dos itens que foram declarados em 2017 (AR e CR)

Raciocínio: a tabela *pessoa* armazena o nome das pessoas e a tabela *tipoitens* armazena o nome dos itens passíveis de declaração. Não é possível fazer um *join* entre *pessoa* e *tipoitens* pois não existe uma chave que une ambas. Estudando o esquema do banco, percebe-se que *IRPF* armazena o resumo da declaração anual de uma pessoa, existindo uma chave que liga ambas *cpf*. Já a tabela *itensIRPF* armazena quais itens uma pessoa declarou em um determinado ano. Essa tabela possui um atributo *tpitem* que é uma chave estrangeira que aponta para *tipoitens*. Feito, o caminho está pronto: *pessoa* → (*cpf*) *IRPF* → (*cpf, ano*) *itensIRPF* → (*tpitem:cod*) *tipoitens*.

Álgebra: $\pi_{nome, descr}(\sigma_{ano=2017}(pessoa \bowtie IRPF \bowtie itensIRPF \bowtie_{tpitem=cod} tipoitens))$

Obs.: o operador sigma poderia ser utilizado diretamente em *IRPF*

Cálculo:

$\{t \mid \exists p \in pessoa (\exists i \in IRPF (\exists ii \in itensIRPF (\exists tp \in tipoitens (p.cpf = i.cpf \wedge i.cpf = ii.cpf \wedge ii.ano = tp.cod \wedge i.ano = 2017 \wedge t.nome = p.nome \wedge t.tipo = tp.descr))))\}$

b) Retorne o nome da pessoa que teve o maior valor restituído em 2012 (AR)

Raciocínio: o atributo *vlrest* armazena o valor restituído da declaração anual de cada pessoa (contribuinte). O enunciado pede que a consulta retorne o nome (ou nomes) da pessoa que teve o maior valor restituído em 2012.

Pode-se fazer em 2 etapas. Na primeira cria-se uma tabela que armazenará apenas o maior valor encontrado no atributo *vlrest* no ano 2012.

$MRest \leftarrow \rho_{mvl}(G_{max(vlrest)}(\sigma_{ano=2012}(IRPF)))$ o operador ρ renomeia o nome do atributo gerado pelo $max(vlrest)$.

Agora, podemos fazer um *join* com *IRPF* e *MRest* utilizando a condição $vlrest=mvl$ que retornará apenas as declarações cujo os valores serão iguais a *mvl*: $IRPF \bowtie_{vlrest=mvl} MRest$

Porém, queremos o nome do contribuinte, assim a consulta deveria ficar:

$\pi_{nome}(pessoa \bowtie IRPF \bowtie_{vlrest=mvl} MRest)$

Se optarmos por fazer uma consulta igual ao SQL:

```
select nome from pessoa natural join IRPF
where vlrest=(select max(vlrest) from IRPF where ano=2012)
```

A opção seria:

$\pi_{nome}(\sigma_{vlrest=((G_{max(vlrest)}(\sigma_{ano=2012}(IRPF))))}(pessoa \bowtie IRPF))$

c) Retorne o nome do contribuinte que declarou imposto de renda em todos os anos cadastrados em *IRPF*. (AR)

Raciocínio: para este caso, temos que utilizar o operador de divisão pois o enunciado pede o nome do contribuinte que declarou em todos os anos presentes na tabela *IRPF*.

Divisão:

$R \div S$ então $att(S) \subset att(R)$ e a projeção será os atributos de *R* que não aparecem em *S*. Assim, tem-se que criar duas tabelas para obter o que é solicitado respeitando a regra da divisão. A primeira tabela deverá ter os nomes e os anos de declaração de imposto para todas as pessoas:

$\pi_{nome, ano}(pessoa \bowtie IRPF)$, a outra, apenas os anos existentes em *IRPF* $\pi_{ano}(IRPF)$, agora podemos dividir ambas e obter o resultado esperado:

$\pi_{nome, ano}(pessoa \bowtie IRPF) \div \pi_{ano}(IRPF)$. O resultado contém apenas o atributo *nome*.

Questão 2: encontrar todas as chaves de R com no máximo 2 atributos

Relação R(A,B,C,D) e

Dfs:

1) $A \rightarrow B$

2) $C \rightarrow B$

3) $D \rightarrow A$

Para encontrar todas as chaves com no máximo 2 atributos, deve-se testar o fechamento de todas as combinações de atributos de R. Se o resultado do fechamento resultar em todos os atributos de R, o candidato é chave. Combinações: A, B, C, D, AB, AC, AD, BC, BD e CD.

$\{A\}^+ \Rightarrow$ verifica as df que tem A na esquerda, sim 1, acrescenta o símbolo da direita $B \Rightarrow \{A,B\} \Rightarrow$ verifica se existe alguma outra df que tenha A ou B à esquerda para inserir um novo símbolo no conjunto, não existe. Para. Resultado $\{A,B\}$ faltam os atributos C e D. NÃO É CHAVE

$\{B\}^+ \Rightarrow$ sem df com B à esquerda, resultado $\{B\}$ NÃO É CHAVE

$\{C\}^+ \Rightarrow$ a df 2 atende, assim $\{C,B\}$, procura outras dfs, não tem, resultado $\{C,B\}$ NÃO É CHAVE

$\{D\}^+ \Rightarrow$ a df 3 atende, assim $\{D,A\}$, agora a df 1 atende também, $\{D,A,B\}$. Não existem mais dfs que atendem, resultado $\{D,A,B\}$. NÃO É CHAVE

$\{AB\}^+ \Rightarrow$ df 1 atende mas B já está no conjunto, nenhuma outra atende, resultado $\{A,B\}$ NÃO É CHAVE

$\{AC\}^+ \Rightarrow$ df 1 atende, assim B é inserido no conjunto $\{A,C,B\}$. df 2 atende mas B já está no conjunto. Nenhuma outra atende. Resultado $\{A,C,B\}$ NÃO É CHAVE

$\{AD\}^+ \Rightarrow$ df 1 atende, assim B é inserido no conjunto $\{A,D,B\}$. df 3 atende mas A já está no conjunto. Resultado $\{A,D,B\}$ NÃO É CHAVE

$\{BC\}^+ \Rightarrow$ df 2 atende mas B já está no conjunto, nenhuma outra atende. Resultado $\{B,C\}$ NÃO É CHAVE

$\{BD\}^+ \Rightarrow$ df 3 atende e A é inserido no conjunto $\{B,D,A\}$, nenhuma outra atende. Resultado $\{B,D,A\}$ NÃO É CHAVE

$\{CD\}^+ \Rightarrow$ df 1 não atende, df 2 sim, resultado $\{C,D,B\}$, df 3 atende, resultado $\{C,D,B,A\}$, df 1 atende mas A já está no conjunto. Resultado final $\{C,D,B,A\}$. **CHAVE**

Questão 3:

Normalizar tabela com as seguintes DF:

CPF Placa DtLoc \rightarrow Nome Modelo Ano Itens DtDev

CPF \rightarrow Nome

Placa \rightarrow Modelo Ano Itens

Sendo o atributo Itens multivalorado.

1FN: eliminar os atributos multivalorados.

Como Itens faz parte da DF com Placa, cria-se uma tabela (itenscarro) com ambos os atributos:

Placa Item com a DF Placa Item \rightarrow Placa Item (trivial), note que a DF não pode ser Placa \rightarrow Item pois uma placa poderia ter apenas um item.

O resultado seria:

Placa	Item
AK1313	ar
AK1313	abs
AK1313	airbag
:	:

2FN: atributos dependendo parcialmente da chave. Temos Nome que depende apenas de CPF, e Modelo e Ano que dependem apenas de Placa (perceba que Itens foi removido da tabela original)

2 tabelas são criadas, uma para as pessoas composta por CPF Nome e DF CPF \rightarrow Nome e outra para carros composta por Placa Modelo Ano e DF Placa \rightarrow Modelo Ano

A tabela original se torna MyBeautyCar(CPF, Placa, DtLoc, DtDev), e três novas tabelas são criadas: itenscarro(Placa, item), cliente(CPF,nome) e carro (Placa, Modelo, Ano).

BCFN: atende pois todas as DF possuem na esquerda chaves.