

## Parte Teórica

### Grupo I

1. nenhuma
2. (b)
3. (a)
4. (c)

### Grupo II

5.1 –

5.2 – Liste o código e o nome dos ingredientes que possuem TODOS os nutrientes;

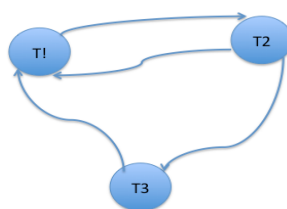
uma solução possível

$$X = \pi_{cod\_Ing} (\sigma_{categoria <> "hidratos de carbono"} (Nutrientes \bowtie Ingredientes \bowtie Ing\_Nut))$$

$$Y = \pi_{cod\_Ing} (\sigma_{categoria = "hidratos de carbono"} (Nutrientes \bowtie Ingredientes \bowtie Ing\_Nut))$$

Resultado final  $\rightarrow \pi_{nome\_Ing} (Ingredientes \bowtie (X - Y))$

6. É não serializável porque o grafo do escalonamento apresenta ciclos



## Parte Prática

**DURAÇÃO 1 HORA e 30 MINUTOS**

Leia a prova com atenção. Seja claro(a) e conciso(a). Responda só ao que se pergunta.

### Grupo I

(6 valores)

Responda numa folha separada

1. Uma companhia de Produtos Electrónicos e Instrumentais projeta e fabrica vários produtos eletrónicos de diferentes funções e características. Um produto pode ser constituído por vários produtos (componentes). Cada produto componente pode entrar na composição de vários produtos e também ser composto por vários produtos componentes numa estrutura em árvore.

Para uma melhor planificação e gestão das tarefas do departamento técnico pretende-se armazenar na Base de Dados todos os dados relacionados com a atribuição e realização das tarefas de design de cada produto.

Cada tarefa, correspondente a um produto, é identificada por um código e atribuída a um técnico definindo-se a datas previstas de início e fim. Caso se justifique, deverão ser definidas quais as tarefas já planeadas de que dependem, isto é devem estar concluídas antes daquela se iniciar. Aquando da realização da tarefa deverão ser registadas na Base de Dados as datas efetivas de início e fim por forma a analisar os desvios em relação à estimativa inicial.

Cada técnico é caracterizado por um código identificativo, nome, apelido e cargo.

Pretende-se também registar na base de dados todos os defeitos que venham a ser detetados. Para cada defeito detetado, identificado por um numero sequencial, deverá registar-se na Base de dados o produto onde foi detetado, o técnico que o detetou, a data de detecção, o tipo de defeito( constante de uma tabela de tipos de defeitos) e uma descrição.

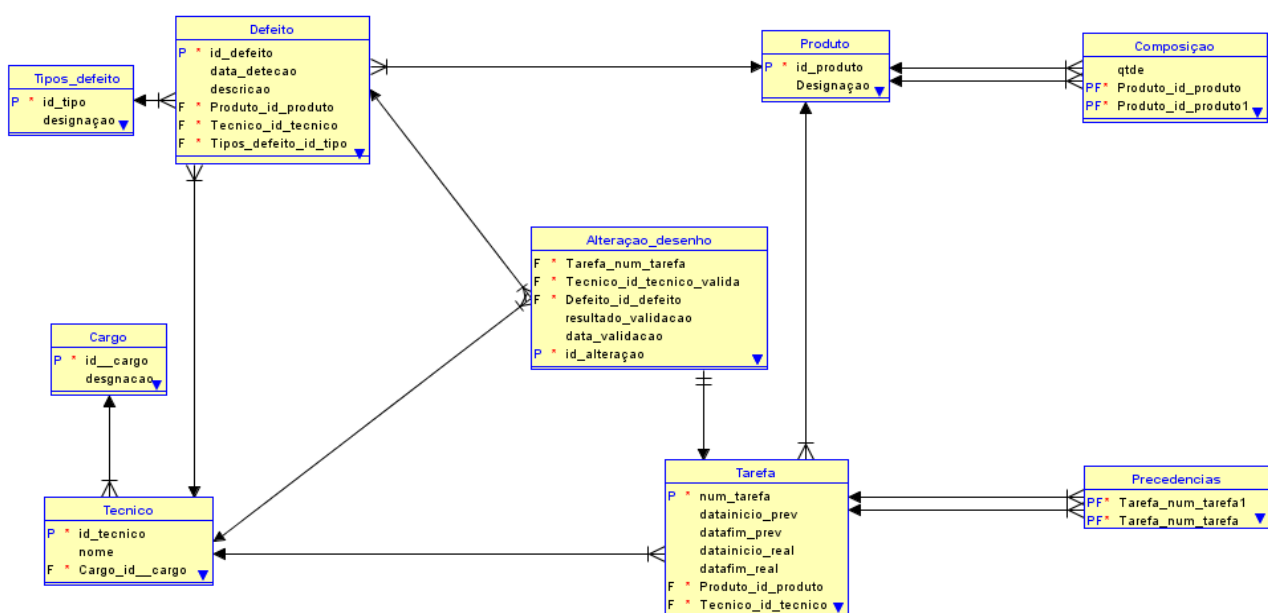
O defeito detetado num produto pode implicar a alteração do desenho deste produto e eventualmente de alguns produtos em que é componente.

Cada alteração ao desenho de um produto motivada pelo defeito detetado deve implicar o registo de uma tarefa, atribuída a um técnico responsável pela sua realização e designando-se outro técnico para validação da mesma .Após a validação devem ser registados o resultado da validação(1-aprovado; 2-rejeitado) e a data da validação.

Com base nos requisitos descritos,

a) [4,5 valores]: Esboce um modelo de dados normalizado explicitando as principais entidades e os atributos mais significativos. Deve ser claro no modelo quais são os atributos que constituem as chaves primárias e as chaves estrangeiras assim como as relações entre as entidades e respectivas cardinalidades.

**Nota:** não se esqueça de mencionar as restrições de integridade, se houver alguma, que seja impossível de representar no desenho do modelo.



Restrições de integridade:

- Os produtos objeto de alterações de desenho motivadas por defeitos num produto têm de ter este produto na sua composição.

- O técnico alocado à realização de uma tarefa de alteração do desenho num produto tem que ser distinto do técnico que faz a validação desta alteração.

b) [1,5 valores]: Escreva um comando SQL que permita listar as designações dos produtos que já tiveram de ser redesenhados devido a defeitos, mas nunca tiveram como técnico responsável o empregado “João Santos”.

```
Select P.designação
from Produto P , Tarefa T, Alteração_Desenho AD
Where AD.Tarefa_num_tarefa=P.num_tarefa and P.id_produto=T.Produto.id_Produto
and T.Tecnico_id_tecnico not in(select id_tecnico from Tecnico where nome="João Santos");
```

## Grupo II (9 valores)

### Responda a todas as questões numa folha separada

Considere o seguinte esquema relacional

Livros = {**ISBN**, título, *AreaEspecialidade*}

Autores = {**nr\_autor**, nomeautor, telefone, data\_nasc}

Livros\_Escritos = {**nr\_autor**, **ISBN**, data\_publicação}

Dominios = {**AreaEspecialidade**, nomeEspecialidade}

com as chaves primárias a bold e sublinhadas e as chaves estrangeiras em itálico

1. [1 valores]: Crie as tabelas Livros, Autores e Livros\_Escritos atendendo às seguintes restrições:

- o atributo telefone inicia-se por 2 ou 9 e tem na 2ª posição o carácter “2” ou “6” tendo o tamanho variável de 9 carateres;
- o atributo data\_nascimento deve ser superior a 1/1/1920 e inferior a 31/12/1990;
- o atributo AreaEspecialização inicia-se por “AE” e na 3ª posição possui o carácter “-”, tendo o tamanho obrigatório de 6 caracteres;

```
CREATE TABLE LIVROS(
ISBN CHAR(10),
título VARCHAR2(20),
AreaEspecialidade CHAR(6)
CONSTRAINT fk_livros_dominio REFERENCES DOMINIO(AreaEspecialidade),
CONSTRAINT pk_livros PRIMARY KEY (ISBN));
```

```
CREATE TABLE AUTORES(
nr_autor number,
nomeautor VARCHAR2(20),
telefone CHAR(9)
CONSTRAINT ck_aut_telef CHECK(telefone LIKE ('22_____' ) OR
telefone LIKE ('26_____' )OR telefone LIKE ('92_____' )OR
telefone LIKE ('96_____' )),
data_nasc date,
CONSTRAINT ck_aut_dt_nasc CHECK (data_nasc between
```

```
to_date('01/01/1920','DD/MM/YYYY') AND to_date('31/12/1990','DD/MM/YYYY')),  
CONSTRAINT pk_autores PRIMARY KEY (nr_autor));
```

ou

```
CREATE TABLE AUTORES(  
nr_autor number,  
nomeautor VARCHAR2(20),  
telefone CHAR(9)  
CONSTRAINT ck_aut_telef  
CHECK(REGEXP_LIKE(telefone,'^[2|9][2|6]......')),  
data_nasc date,  
CONSTRAINT ck_aut_dt_nasc CHECK (data_nasc between  
to_date('01/01/1920','DD/MM/YYYY') AND to_date('31/12/1990','DD/MM/YYYY')),  
CONSTRAINT pk_autores PRIMARY KEY (nr_autor));
```

```
CREATE TABLE LIVROS_ESCRITOS(  
nr_autor number  
CONSTRAINT fk_livros_esc_autores REFERENCES AUTORES(nr_autor),  
ISBN CHAR(10)  
CONSTRAINT fk_livros_esc_livros REFERENCES LIVROS(ISBN),  
data_publicação date,  
CONSTRAINT pk_livros_escritos PRIMARY KEY (nr_autor, ISBN));
```

2. [2 valores]: Escreva uma view que permita listar o nome dos autores que escreveram mais livros do que qualquer autor com idade entre os 25 e 50 anos.

```
CREATE VIEW AutoresMaisProdutivos AS  
SELECT nomeautor  
FROM Autores a, Livros_Escritos l  
WHERE a.nr_autor = l.nr_autor  
GROUP BY nomeautor  
HAVING COUNT(*) > ALL(  
SELECT COUNT(*)  
FROM Autores as, Livros_Escritos ls  
WHERE as.nr_autor = ls.nr_autor  
AND MONTHS_BETWEEN(data_nasc, SYSDATE) BETWEEN 25*12 AND  
50*12  
GROUP BY ls.nr_autor)
```

3. [2 valores]: Escreva uma função denominada “NrAreasSemEspecialidade” **que receba como parâmetro um ano e retorne quantas Áreas de Especialidade** não têm livros publicados nesse ano, mas têm livros publicados noutros anos.

```
Create FUNCTION NrAreasSemEspecialidade (auxAno Number) RETURN Number IS  
auxTot Number;
```

```
Select Count(*) Into auxTot  
From Dominios  
Where AreaEspecialidade not in (  
Select AreaEspecialidade  
From Livros, Livros_escritos  
Where Livros.ISBN = Livros_escritos.ISBN  
And extract(year from data_publicação)= auxAno  
)  
And AreaEspecialidade in (  
Select AreaEspecialidade
```

```

From Livros, Livros_escritos
Where Livros.ISBN = Livros_escritos.ISBN
And extract(year from data_publicação)<> auxAno
)

RETURN auxTot;
END NrAreasSemEspecialidade;

```

4. [2 valores]: Escreva um procedimento que permita listar para cada autor o seu nome e o título do último livro escrito pelo autor. Caso o autor não tenha escrito nenhum livro, deverá ser levantada uma exceção.

```

create or replace procedure ex4 as
cursor c is
select a.nomeautor, l.titulo
from autores a, livros l, livros_escritos le
where a.nr_autor = le.nr_autor and le.isbn = l.isbn
and le.data_publicacao = (select max(data_publicacao)
from livros_escritos
where nr_autor = a.nr_autor);
order by a.nomeautor;
r c%rowtype;
begin
open c;
fetch c into r;
while c%found loop
dbms_output.put_line('Autor: ' || r.nomeautor);
dbms_output.put_line('Título: ' || r.titulo);
fetch c into r;
end loop;
close c;
end ex4;

```

5. [2 valores]: Considere que a tabela autores foi atualizada, passando a ter mais um atributo, designado de nrlivros:

Autores (**nr\_autor**, nomeautor, telefone, data\_nasc, nrlivros)

- O atributo nrlivros representa o número de livros escritos pelo autor no ano corrente.

Escreva um trigger que permita atualizar o atributo nrlivros na tabela autores, quando se insere ou elimina um registro na tabela livros\_escritos.

```

CREATE OR REPLACE TRIGGER atualizar_nrlivros
AFTER INSERT OR DELETE ON livros_escritos
FOR EACH ROW
WHEN ( EXTRACT(YEAR FROM NEW.data_publicacao) = EXTRACT(YEAR FROM SYSDATE) OR
EXTRACT(YEAR FROM OLD.data_publicacao) = EXTRACT(YEAR FROM SYSDATE) )
DECLARE
l_atualizacao NUMBER(1);
l_nr_autor livros_escritos.nr_autor%TYPE;
BEGIN
IF INSERTING THEN
l_atualizacao := 1;
l_nr_autor := :NEW.nr_autor;
ELSE
l_atualizacao := -1;
l_nr_autor := :OLD.nr_autor;
END IF;
UPDATE Autores
SET nrlivros = nrlivros + l_atualizacao
WHERE nr_autor = l_nr_autor;

```

END atualizar\_nrlivros;