

Resolução Exame de Recurso 2018/19

Grupo I.1

I.1.a) ver página a seguir

I.1.b)

Tabelas(cod_tabela, d_início, d_fim)

Combustíveis(cod_comb, designação)

Preço_combustíveis(cod_tabela, cod_comb, preço)

CEs: cod_tabela referindo Tabelas(cod_tabela) e cod_comb referindo Combustíveis(cod_comb)

Notação alternativa: cod_tabela -> Tabelas(cod_tabela) e cod_comb -> Combustíveis(cod_comb)

Estações(cod_estação, nome, morada, telefone, latitude, longitude, h_abertura, h_fecho,cod_tabela)

CE: cod_tabela -> Tabelas(cod_tabela)

Localidades(cod_local, nome, distrito)

Localizadas(cod_estação, cod_local)

CEs: cod_estação -> Estações(cod_estação) e cod_local -> Localidades(cod_local)

Bombas(cod_estação, n_bomba, pré_pagamento)

CE: cod_estação -> Estações(cod_estação)

Caixas(IP, data_instalação, cod_estação)

CE: cod_estação -> Estações(cod_estação)

Self_service(IP, cod_estação, n_bomba)

CEs: IP -> Caixas(IP) e (cod_estação, n_bomba) -> Bombas(cod_estação, n_bomba)

Disponibiliza(cod_estação, n_bomba,cod_comb)

CEs: (cod_estação, n_bomba) -> Bombas(cod_estação, n_bomba) e cod_comb ->

Combustíveis(cod_comb)

I.2.a) ver diagrama a seguir

I.2.b)

Manuais(IP)

CE: IP -> Caixas(IP)

Abertura(IP,dh_abertura,NIF)

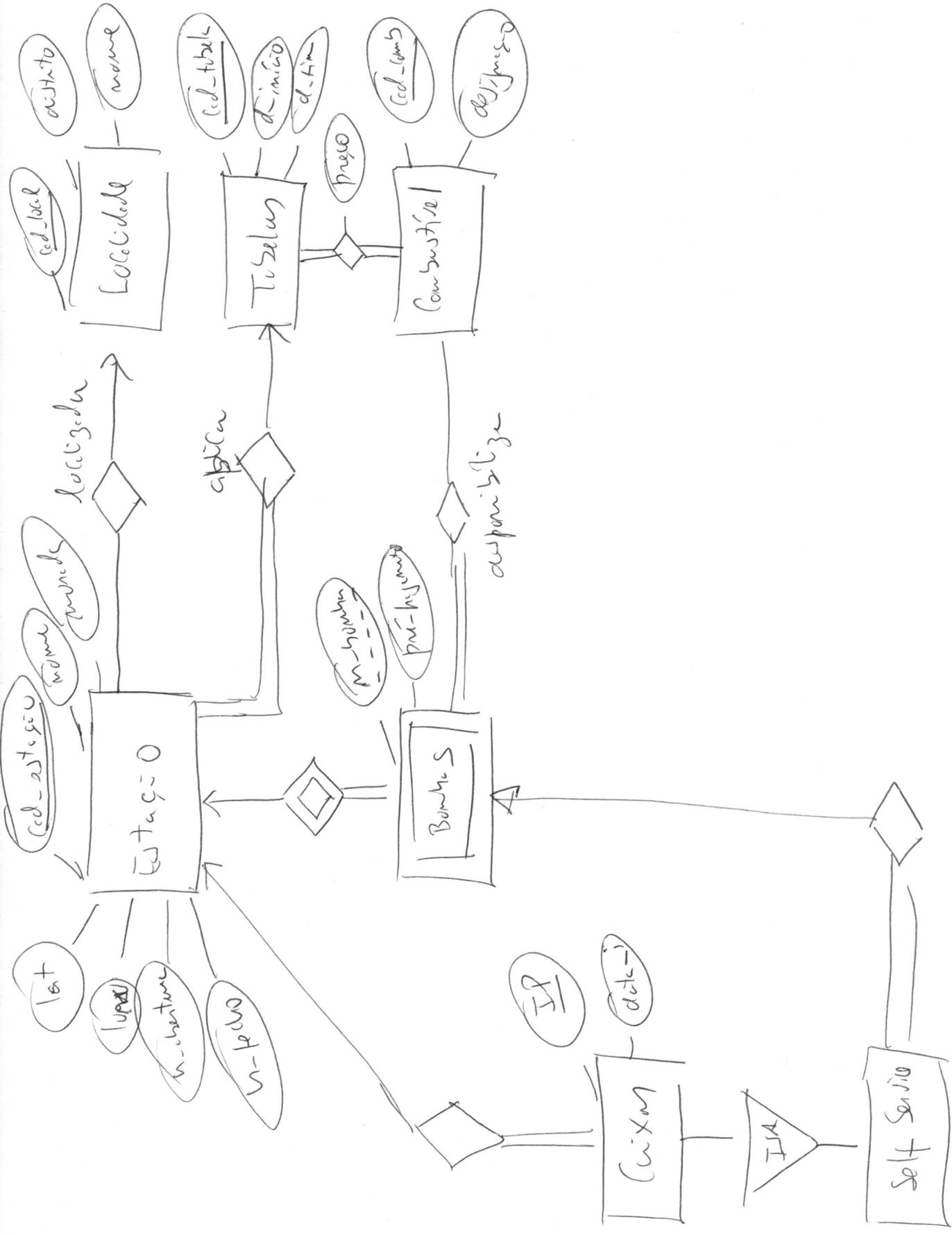
CEs: IP -> Manuais(IP) e NIF -> Operadores(NIF)

Fechos(IP,dh_abertura,dh_fecho, NIF)

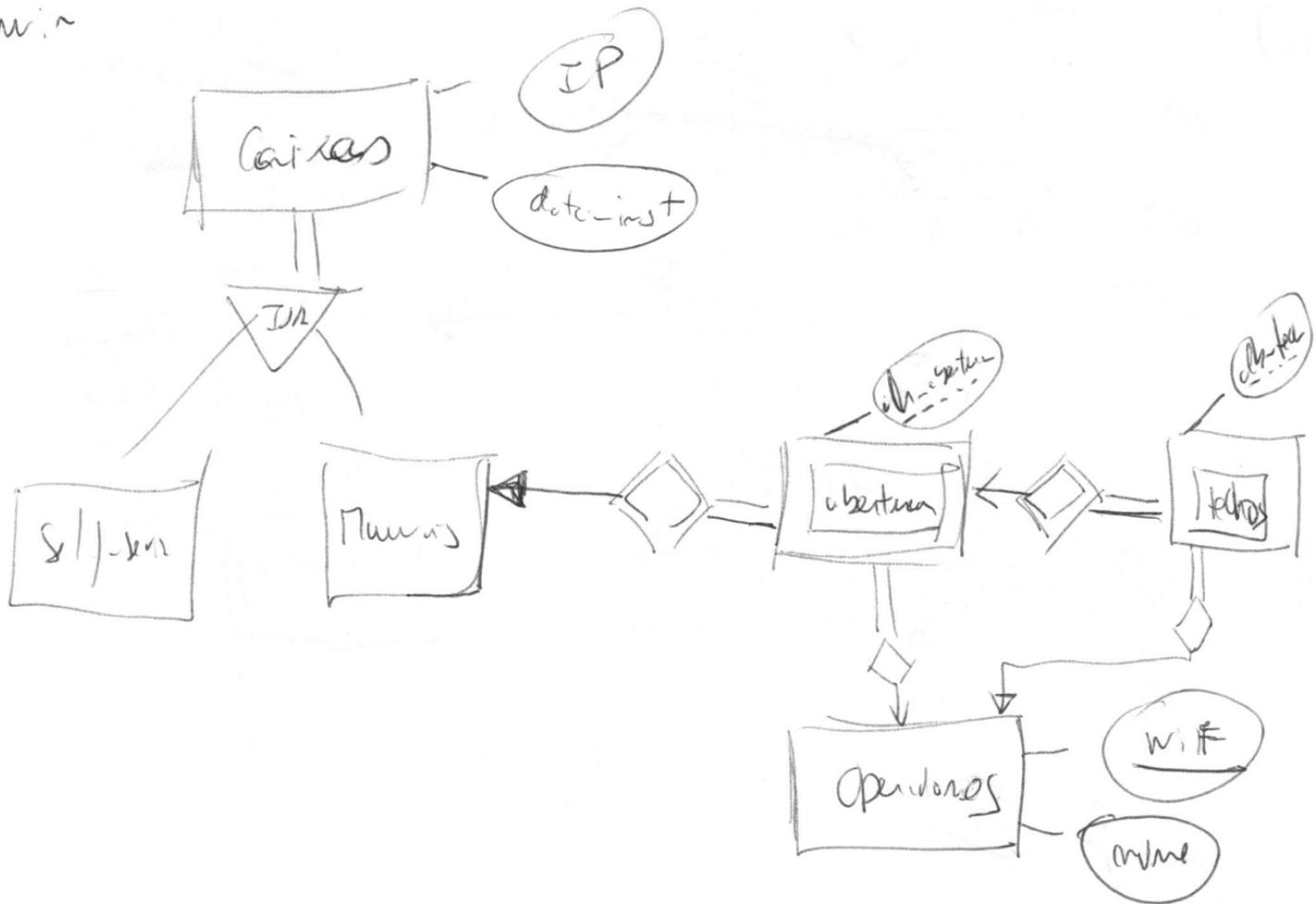
CEs: (IP, dh_abertura) -> Aberturas(IP, dh_abertura) e NIF -> Operadores(NIF)

Operadores(NIF,Nome)

I. l. l. a)



2.) IS Min



Grupo I.2

I.2.a) C e D têm obrigatoriamente de pertencer a qualquer chave candidata, pois nunca ocorrem em qualquer lado direito de uma dependência funcional. Mas $CD^+ = CD$ logo há mais que uma chave candidata.

As chaves candidatas são: CDI e CDEB.

$CDI^+ = R$ (e é minimal pois CD tem de pertencer a todas e não é super-chave)

$CDEB^+ = R$ (basta testar $CDE^+ = CDEPU$ e $CDB^+ = CDB$)

I.2.b) F não tem atributos dispensáveis logo está na forma canónica

$EBC \rightarrow I$

$EC \rightarrow P$

$ID \rightarrow EB$

$ED \rightarrow U$

Para ver que por exemplo E não é dispensável à esquerda em $EBC \rightarrow I$, calcula-se BC^+ em F, obtendo $BC^+ = BC$ não contendo I (é preciso verificar tudo...). Os atributos do lado direito de todas as dependências não são dispensáveis pois cada um deles só ocorre uma vez. Se os retirarmos vai-se obviamente obter um conjunto de DFs que não é cobertura de F.

I.2.c) Não está na 3FN. As causas de violação de R estar na 3FN são as dependências funcionais não triviais $EC \rightarrow P$ e $ED \rightarrow U$ pois os lados esquerdos não são super-chave, e nem P nem U pertencem a uma chave candidata.

Aplicando o algoritmo de síntese para a 3FN ficamos com $R1(EBCI)$, $R2(EBDI)$, $R3(ECP)$ e $R4(EDU)$. Temos ainda de juntar uma chave candidata aos esquemas anteriores, ou seja $R5=(CDI)$ ou $R5=(CDEB)$ pois nenhum de $R1$ a $R4$ contém uma chave candidata.

I.2.d) (segue árvore sem justificação)

$R(CDIEBPU)$: não está na FNBC sendo $ED \rightarrow U$ causa, decompondo com $ED \rightarrow U$

$R1(EDU)$: está na FNBC

$R2(CDIEBP)$: não está na FNBC sendo $EC \rightarrow P$ causa, decompondo

$R3(ECP)$

$R4(CDIEB)$: não está na FNBC sendo $EB \rightarrow CI$ causa, decompondo

$R5(EBCI)$: está na FNBC

$R6(EBCD)$: está na FNBC

Vamos verificar que $R6$ está na FNBC (fazer para o resto é obrigatório)

$E^+ = E$ (só E)	$EB^+ = EB$ (só EB)
$B^+ = B$ (só B)	$EC^+ = ECP$ (só EC em $R4$)
$C^+ = C$ (só C)	$ED^+ = EDU$ (só ED em $R4$)
$D^+ = D$ (só D)	$BC^+ = BC$ (só BC em $R4$)
	$CD^+ = R$ (super-chave)
	$PU^+ = PNU$ (só PU)

Logo a decomposição para a FNBC é $R1(EDU)$, $R3(ECP)$, $R5(EBCI)$ e $R6(EBCD)$.

Grupo I.3

I.3.1.a)

$\Pi_{tipo} [\Pi_{IdB} (\sigma_{estacao='Aveiras'}(bombas)) \bowtie mangueiras \bowtie depositos \bowtie combustiveis]$

I.3.1.b)

$\Pi_{IdB} (mangueiras) - \Pi_{IdB} (mangueiras \bowtie \sigma_{IdComb='S95' \vee IdComb='D95'}(depositos))$

I.3.1.c)

$v \leftarrow \Pi_{IdB} (mangueiras \bowtie depositos \bowtie combustiveis)$
 $\Pi_{IdB} [\sigma_{a1.t1 <> a2.t2} (\rho_{a1}(idB,t1)(v) \bowtie \rho_{a2}(idB,t2)(v))]$

Grupo II.1

II.1.1.a)

```
SELECT Tipo
FROM bombas INNER JOIN mangueiras USING (IdB)
             INNER JOIN depósitos USING (IdDep)
             INNER JOIN combustíveis USING (IdComb)
WHERE estação = 'Aveiras'
```

II.1.1.b)

$\Pi_{IdB} (mangueiras) - \Pi_{IdB} (mangueiras \bowtie \sigma_{IdComb='S95' \vee IdComb='D95'}(depositos))$

```
SELECT DISTINCT IdB
FROM mangueiras
WHERE IdB NOT IN (
    SELECT IdB
    FROM mangueiras NATURAL INNER JOIN depósitos
    WHERE IdComb='D95' OR IdComb='S95'
)
```

II.1.1.c)

```
WITH tipos_combustível AS
(SELECT DISTINCT IdB, Tipo
 FROM mangueiras NATURAL INNER JOIN depósitos NATURAL INNER JOIN combustíveis
)
SELECT DISTINCT t1.IdB
FROM tipos_combustível t1, tipos_combustível t2
WHERE t1.IdB = t2.IdB AND t1.tipo <> t2.tipo;
```

II.1.2.a)

bombas_sem_95(IDB) :-
mangueiras(IDB,_,_),
not bombas_com_95(IDB).

bombas_com_95(IDB) :-
mangueiras(IDB,_,DEP),
depósitos(DEP,_,D95').

bombas_com_95(IDB) :-
mangueiras(IDB,_,DEP),
depósitos(DEP,_,S95').

II.1.3.a)

```
SELECT DISTINCT b
FROM vendas INNER JOIN mangueiras ON (IdB=b AND número = n)
      NATURAL INNER JOIN depósitos
WHERE data='2019-07-03'
GROUP BY b,IdComb
HAVING SUM(litros) >= 100;
```

II.1.3.b)

```
WITH reabastecimentos_diários AS
( SELECT data, SUM(litros) total_diário
  FROM entregas NATURAL INNER JOIN reabastecimentos
  GROUP BY data
)
SELECT AVG(total_diário)
FROM reabastecimentos_diários;
```

Grupo II.2

II.2.1.a)

```
ALTER TABLE bombas ADD CONSTRAINT num_único UNIQUE (estação,número)
```

II.2.1.b)

```
CREATE ASSERTION só_uma_mangueira CHECK
NOT EXISTS (
    SELECT * FROM mangueiras m1 NATURAL INNER JOIN depósitos
    WHERE IdComb='GPL' AND EXISTS (
        SELECT COUNT(*) FROM mangueiras m2 NATURAL INNER JOIN depósitos
        WHERE m1.IdB = m2.Idb
        HAVING COUNT(*) > 1
    )
)
```

II.2.1.c)

```
ALTER TABLE reabastecimentos ADD CONSTRAINT reab1000 CHECK ( MODE(litros,1000) = 0 )
```

II.2.2.a)

```
CREATE TRIGGER atualiza_litros
AFTER INSERT ON reabastecimentos
REFERENCING NEW ROW AS nrow
FOR EACH ROW
BEGIN
    UPDATE depósitos
    SET total_litros = total_litros + nrow.litros
    WHERE idDep = nrow.idDep;
END;
```

```
CREATE TRIGGER capacidade_ultrapassada
AFTER UPDATE OF total_litros ON depósitos
REFERENCING NEW ROW AS nrow
FOR EACH ROW WHEN (nrow.total_litros > nrow.capacidade OR nrow.total_litros < 0)
BEGIN
    UPDATE medições
    SET litros_medidos = NULL, data = NULL
    WHERE idDep = nrow.idDep;
END;
```

II.2.2.b)

```
CREATE TRIGGER faz_medição
BEFORE UPDATE OF data, litros_medidos ON medições
REFERENCING OLD ROW AS orow
REFERENCING NEW ROW AS nrow
FOR EACH ROW WHEN (nrow.data IS NOT NULL AND nrow.litros_medidos IS NOT NULL)
BEGIN
    IF nrow.data >= orow.data THEN
        UPDATE depósitos
        SET total_litros = nrow.litros_medidos
        WHERE idDep = nrow.idDep;
    ELSE
        raise_application_error(...)
    END IF;
END;
```

Grupo II.3

II.3.1) Sim, é serializável. A sequência de execução T2, T3, T1 resulta em total_litros[1]=100, total_litros[2]=800 e total_litros[3]=500.

II.3.2) Não é serializável de conflito. Construindo o grafo de conflitos, verifica-se que existe um conflito entre a 1ª e 2ª instrução (T1->T2) e ente a 5ª e a 9ª instrução (T2->T1), introduzindo um ciclo no grafo. Logo, não é serializável de conflito.

Grupo II.4

II.4.1)

```
<estações>
  <estação nome='Aveiras'>
    <estrada km='44' cod='AE1' />
    <combustível>Gasóleo</combustível >
    <combustível>Gasolina simples</combustível >
    <combustível>Gasolina aditivada</combustível >
    <rápido />
  </estação>
  <estação nome='Aeroporto' cod='LIS-1'>
    <localidade>Lisboa</localidade>
    <combustível>Gasóleo</combustível >
  </estação>
</estações>
```

II.4.2.a) /estações/estação[rápido]/@nome

II.4.2.b) /estações/estação[estrada/@cod='AE1']/combustível

II.4.2.c) /estações/estação[count(combustível) >= 2 and not(rápido)]

II.4.3) Produz uma lista indicando para cada tipo de combustível as localidades ou estradas onde é comercializado. O nome do combustível é colocado no atributo c do elemento combustível.

```
<lista>
  <combustível c='Gasóleo'>
    <estação><estrada km='44' cod='AE1' /></estação>
    <estação<localidade>Lisboa</localidade></estação>
  </combustível >
  <combustível c='Gasolina simples'>
    <estação><estrada km='44' cod='AE1' /></estação>
  </combustível >
  <combustível c='Gasolina aditivada'>
    <estação><estrada km='44' cod='AE1' /></estação>
  </combustível >
</lista>
```