



MINISTÉRIO DA EDUCAÇÃO
CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA CELSO SUCKOW DA FONSECA
COORDENAÇÃO DE CONCURSOS - CCONC
Edital 04/2023 - Professor Efetivo
Área de Conhecimento: COMPUTAÇÃO Programação / Algoritmos
Campus Petrópolis



Questão 1) [2,0 pontos] Responda os itens abaixo sobre o algoritmo de ordenação Quicksort:

a) [1,0 ponto] Sabe-se que o Quicksort original não é um algoritmo estável. Especifique em pseudo-código uma versão recursiva estável do Quicksort

b) [1,0 ponto] Após executar o procedimento de Partição, o Quicksort original ordena recursivamente o subarranjo da esquerda e depois ordena recursivamente o subarranjo da direita. Esta segunda chamada recursiva não é realmente necessária. Especifique em pseudo-código uma versão recursiva do Quicksort que mantenha o tamanho da pilha de recursão o menor possível, sem comprometer o tempo de execução $O(n \log n)$ do Quicksort.

Questão 2) [2,0 pontos] Responda os itens abaixo sobre as árvores AVL:

a) [0,5 ponto] Implemente na linguagem C uma função que retorne a altura de uma árvore AVL.

b) [1,5 ponto] Implemente na linguagem C uma função de inserção de um elemento em uma árvore AVL, que mantenha a árvore balanceada ao término da mesma.

Questão 3) [2,0 pontos] Implemente três classes em Python para a criação de arrays:

a) [0,3 ponto] A primeira classe é a classe de um array abstrato. Ela é a classe pai das outras duas classes de array. A classe de array abstrato deve definir os métodos que as classes filhas devem possuir, mas sem implementá-los. Os métodos definidos devem ser: um método de soma de arrays, considerando o próprio objeto e outro objeto da mesma classe, que realiza uma soma elemento a elemento retorna um novo array; um método de cálculo da distância euclidiana do objeto em relação a outro da mesma classe; e um método para encontrar o array com menor distância em relação ao objeto, dados vários arrays da mesma classe.

b) [0,7 ponto] A segunda classe, herdeira da primeira, define um array denso. Ela deve permitir que os elementos do array sejam guardados em um atributo. Os três métodos da classe pai devem ser implementados de acordo com a estrutura definida.

c) [1,0 ponto] A terceira classe, também herdeira da primeira, define um array esparsos. Ela deve permitir que os elementos do array sejam guardados de forma a aproveitar esparsividade dos dados. Assim como na outra classe, os três métodos da classe pai devem ser implementados de acordo com a estrutura definida.

Em ambas implementações, o método de encontrar o array com menor distância deve usar o método de cálculo da distância para calcular a distância do próprio objeto com os demais arrays recebidos. Os métodos devem realizar os cálculos somente para arrays de mesmo tamanho. Possíveis erros gerados por arrays de tamanhos diferentes devem ser tratados.

A distância euclidiana entre dois arrays $x = [x_1, x_2, x_3, \dots, x_n]$ e $y = [y_1, y_2, y_3, \dots, y_n]$, que representam pontos em um espaço de n dimensões, é dada por:

$$d = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (1)$$

Questão 4) [2,0 pontos] No CEFET-RJ campus Petrópolis, os alunos da graduação precisam registrar atividades complementares, como, por exemplo, cursos de língua estrangeira. Estas atividades precisam ser revisadas por professores avaliadores que poderão, ou não, deferir uma quantidade de horas, independentemente da quantidade solicitada pelo aluno para uma determinada atividade. Considere a seguir os comandos SQL para criar as tabelas deste banco de dados:

```
CREATE TABLE "aluno" (  
    "id"    INTEGER NOT NULL,  
    "nome"  TEXT,  
    "matricula"  INTEGER,  
    PRIMARY KEY("id")  
);
```

```

CREATE TABLE "atividade" (
  "id"    INTEGER NOT NULL,
  "descricao" TEXT,
  PRIMARY KEY("id")
);

CREATE TABLE "avaliador" (
  "id"    INTEGER,
  "nome"   TEXT,
  PRIMARY KEY("id")
);

CREATE TABLE "atividade_submetida" (
  "id"    INTEGER NOT NULL,
  "id_aluno"    INTEGER,
  "id_atividade"    INTEGER,
  "id_avaliador"    INTEGER,
  "qtd_horas_solicitadas" REAL,
  "qtd_horas_deferidas" REAL,
  "comprovante"    BLOB,
  "data_submissao"    DATE,
  "data_avaliacao"    DATE DEFAULT NULL,
  PRIMARY KEY("id"),
  FOREIGN KEY("id_aluno") REFERENCES "aluno"("id"),
  FOREIGN KEY("id_avaliador") REFERENCES "avaliador"("id"),
  FOREIGN KEY("id_atividade") REFERENCES "atividade"("id")
);

```

Seja $H_{s_{i,k}}$ a quantidade de horas solicitadas e $H_{d_{i,k}}$ a quantidade de horas deferidas para uma única atividade submetida por um aluno. O índice i faz referência ao id na tabela aluno e o índice k faz referência ao id na tabela atividade_submetida. Dado um aluno i podemos calcular o somatório da diferença entre as horas solicitadas e as horas deferidas. Ou seja,

$$D_i = \sum_k |H_{s_{i,k}} - H_{d_{i,k}}|.$$

O cálculo de D_i só deve considerar as atividades submetidas que foram avaliadas, ou seja as entradas cujo campo data_avaliacao seja diferente de NULL.

a) [1,3 pontos] Faça uma consulta em SQL que retorne, para cada aluno, um resultado com seu nome, a quantidade de horas deferidas, a quantidade avaliadores diferentes e o valor do D_i ordenado (de forma decrescente) pelo D_i .

b)[0,7 ponto] Considere as tabelas populadas conforme os valores abaixo:

Tabela aluno

id	nome	matricula
1	Ana Carolina Oliveira	278395
2	Patrick Ferreira	278893
3	Helena Vieira	289049
4	Lucas Silva	287171

Tabela atividade

id	descricao
1	Curso de Língua Estrangeira
2	Iniciação Científica
3	Monitoria
4	Organização de Evento

id	nome
1	José Carlos
2	Claudio Maia
3	Paloma Fernandes

id	id_aluno	id_atividade	id_avaliador	qtd_horas_solicitadas	qtd_horas_deferidas	data_submissao	data_avaliacao
1	1	3	3	25.0	20.0	2022-09-04	2022-09-10
2	1	2	2	60.0	60.0	2022-09-04	2022-09-09
3	1	1	2	80.0	65.0	2022-09-02	2022-09-12
4	3	1	2	120.0	120.0	2022-08-02	2022-09-10
5	3	4	1	20.0	15.0	2022-08-02	2022-08-03
6	2	1	1	60.0	60.0	2022-09-05	2022-09-10
7	2	4	3	10.0	10.0	2022-09-05	2022-09-08

Qual é o resultado após a execução da consulta abaixo?

```
SELECT SUM(s.qtd_horas_deferidas), COUNT(*), MAX(s.data_avaliacao) FROM
aluno a inner join atividade_submetida s, atividade t on
a.id=s.id_aluno AND t.id=s.id_atividade
GROUP BY a.id ORDER BY a.nome
```

Questão 5) [2,0 pontos] Considere o Diagrama Entidade-Relacionamento (DER) mostrado na figura abaixo para parte de um banco de dados BANCO. Cada banco pode ter várias filiais, e cada filial pode ter várias contas e empréstimos.

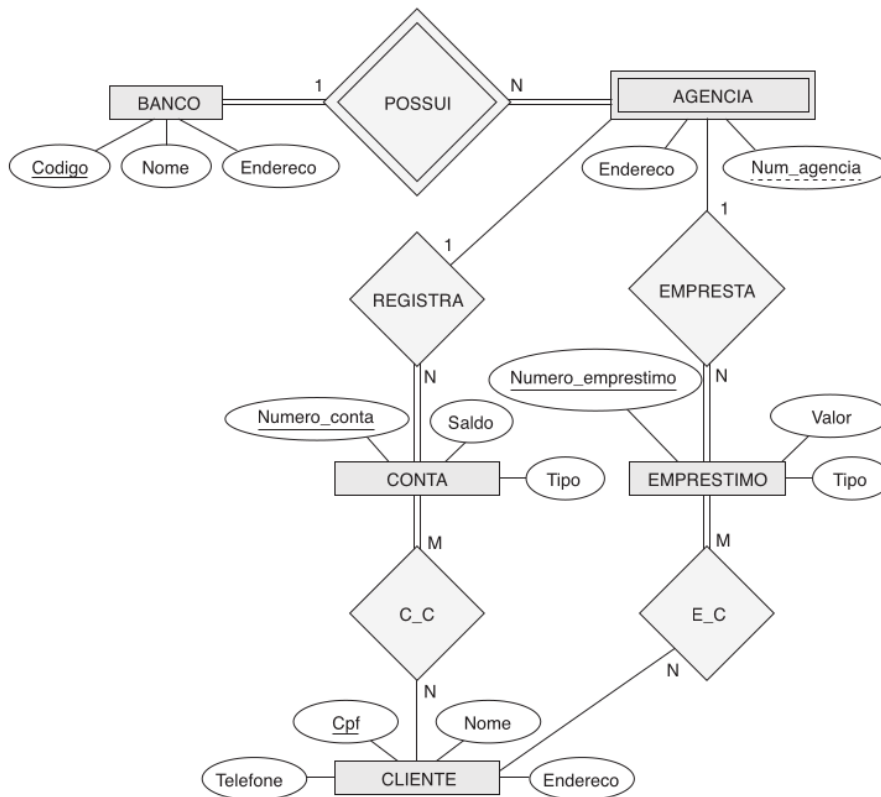


Figura: Diagrama Entidade-Relacionamento para o banco de dados BANCO.

a) [0,3 ponto] Existe um tipo de entidade fraca? Se houver, diga seu nome, chave parcial e relacionamento de identificação.

b) [1,7 pontos] Mapeie o esquema DER BANCO em um esquema relacional. Especifique todas as chaves primárias e estrangeiras.