**Códigos fontes em java ou c#.**

**Questão 1:**  q = [0.5, 1.0, 1.5, 2.0] , g = [0,1,0,1] . As listas q e g são respectivamente a nota (q) e o gabarito (g), sendo 0 para questão errada e 1 para certa. Faça uma função que receba estas listas como parâmetro e exiba: a nota total da prova e as questões certas e erradas.

**Questão 2 :**

Dado o arquivo q2, em anexo. Criar um método para ler o arquivo e gerar outro arquivo com um script para inserir (insert) cada linha na tabela Questao\_2, na coluna dsc\_obs, quando id\_questao\_2 for impar.

Ex: insert into Questao\_2 (dsc\_obs) values(‘teste’) where id\_questao\_2 = 1;

**Questão 3 :**

Algoritmo:

**Diferença Diagonal**

Dada uma matriz quadrada, calcule a diferença absoluta entre as somas de suas diagonais.

Por exemplo, a matriz quadrada é mostrada abaixo:

1 2 3

4 5 6

9 8 9

A diagonal da esquerda para a direita = 1 + 5 + 9 =15. A diagonal da direita para a esquerda = 3 + 5 + 9 = 17. Sua diferença absoluta é |15 – 17| = 2.

**Descrição da função**

Complete a função diagonalDifference abaixo. Deve retornar um inteiro representando a diferença diagonal absoluta.

diagonalDifference leva o seguinte parâmetro:

arr: uma matriz de inteiros.

**Formato de entrada**

A primeira linha contém um único inteiro, n, o número de linhas e colunas na matriz arr.

Cada uma das n linhas seguintes descreve uma linha, arr[i], e consiste em n inteiros separados por espaços arr[i][j].

**Restrições**

-100<= arr[i][j] <=100

**Formato de saída**

Imprima a diferença absoluta entre as somas das duas diagonais da matriz como um único inteiro.

**Exemplo de Entrada**

3

11 2 4

4 5 6

10 8 -12

**Exemplo de Saída**

15

**Explicação**

A diagonal primária é:

11

   5

     -12

Soma na diagonal primária: 11 + 5 - 12 = 4

A diagonal secundária é:

     4

   5

10

Soma na diagonal secundária: 4 + 5 + 10 = 19

Diferença: | 4 - 19 | = 15

Nota: | x | é o valor absoluto de x

**Questão 4 :**

Teste de SQL:

Dadas as tabelas CITY e COUNTRY, consulte os nomes de todos os continentes (COUNTRY.Continent) e suas respectivas populações de cidades médias (CITY.Population) arredondadas para o número inteiro mais próximo.

Nota: CITY.CountryCode e COUNTRY.Code estão combinando colunas-chave.

As tabelas CITY e COUNTRY são descritas da seguinte forma:



**Resposta questão 2:**

SELECT

COUNTRY.CONTINENT CONTINENT,

FLOOR (AVG(CITY.POPULATION) ) POPULATION

FROM COUNTRY INNER JOIN CITY

ON COUNTRY.CODE = CITY.COUNTRYCODE

GROUP BY COUNTRY.CONTINENT