

Questão 1)

Como engenheiro de dados, você foi requisitado a fazer uma ETL em um novo projeto. Esta ETL consiste na leitura de um arquivo excel, extração do valor de algumas colunas do excel e transformação destes dados. Por fim, os dados serão armazenados em uma instância do SQL Server.

O arquivo .xls consiste em dados coletados da ANP, com o valor das vendas de combustíveis para cada mês dos anos entre 2000 e o ano/mês atual. Estes valores de venda são números inteiros positivos. Por motivos técnicos, você não poderá fazer a leitura desse arquivo como .xls, mas sim como um arquivo da biblioteca libreoffice do Linux : .ods. Ao fazer a conversão de .xls para .ods, você verificou que para alguns arquivos os valores no .ods encontram-se fora da ordem correta.

As colunas presentes no arquivo .ods são: Nome do Combustível, Unidade de Medida, Refinaria, Estado, Ano, Janeiro, Fevereiro, Março, Abril, Maio, Junho, Julho, Agosto, Setembro, Outubro, Novembro, Dezembro, Total. As colunas dos meses contém os valores das vendas correspondentes para aquele mês, e a coluna Total contém a soma dos valores de cada mês da linha correspondente. Entretanto, em algumas conversões de arquivos, os valores dos meses e do total sofrem um shuffle de $n + k$ casas para a direita, a partir da primeira linha, com k sendo incrementado de 1 a cada linha seguinte. Mais especificamente, a primeira linha sofre um shuffle de n casas, a segunda linha sofre um shuffle de $n + 1$ casas, a terceira linha sofre um shuffle de $n + 2$ casas... a décima terceira linha sofre um shuffle de 13 casas (retornando à configuração original), e assim até o fim do arquivo.

Internamente, você e seus colegas de equipe apelidaram esse problema de "Escadinha".

Exemplo de primeira linha:

Correto:

Jan	Fev	Mar	Abril	Mai	Jun	Jul	Ago	Set	Out	Nov	Dez	Total
1000	2500	1200	3000	1234	700	300	1000	0	800	2400	3500	17634

Com shuffle de $n = 4$:

Jan	Fev	Mar	Abril	Mai	Jun	Jul	Ago	Set	Out	Nov	Dez	Total
800	2400	3500	17634	1000	2500	1200	3000	1234	700	300	1000	0

Com shuffle de $n = 1$:

Jan	Fev	Mar	Abril	Mai	Jun	Jul	Ago	Set	Out	Nov	Dez	Total
17634	1000	2500	1200	3000	1234	700	300	1000	0	800	2400	3500

Tranformando os arquivos em matrizes, somente com as colunas acima, teremos os seguintes exemplos:

Exemplos de padrões desejados (matriz correta ou com escadinha):

```
1 matrizCorreta = [  
#      [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78],  
#      [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78],  
#      [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78],  
#      [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78],  
#      [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78],  
#      [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78],  
#      [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78],  
#      [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78],  
#      [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78],  
#      [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78],  
#      [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78],  
#      [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78],  
#      ]
```

```
2 matrizEscadinha = [  
#      [11, 12, 78, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10],  
#      [10, 11, 12, 78, 1, 2, 3, 4, 5, 6, 7, 8, 9],  
#      [9, 10, 11, 12, 78, 1, 2, 3, 4, 5, 6, 7, 8],  
#      [8, 9, 10, 11, 12, 78, 1, 2, 3, 4, 5, 6, 7],  
#      [7, 8, 9, 10, 11, 12, 78, 1, 2, 3, 4, 5, 6],  
#      [6, 7, 8, 9, 10, 11, 12, 78, 1, 2, 3, 4, 5],  
#      [5, 6, 7, 8, 9, 10, 11, 12, 78, 1, 2, 3, 4],  
#      [4, 5, 6, 7, 8, 9, 10, 11, 12, 78, 1, 2, 3],  
#      [3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78, 1, 2],  
#      [2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78, 1],  
#      [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78],  
#      [78, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]  
#      ]
```

```

3 matrizEscadinhaEstendida = [
#      [11, 12, 78, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10],
#      [10, 11, 12, 78, 1, 2, 3, 4, 5, 6, 7, 8, 9],
#      [9, 10, 11, 12, 78, 1, 2, 3, 4, 5, 6, 7, 8],
#      [8, 9, 10, 11, 12, 78, 1, 2, 3, 4, 5, 6, 7],
#      [7, 8, 9, 10, 11, 12, 78, 1, 2, 3, 4, 5, 6],
#      [6, 7, 8, 9, 10, 11, 12, 78, 1, 2, 3, 4, 5],
#      [5, 6, 7, 8, 9, 10, 11, 12, 78, 1, 2, 3, 4],
#      [4, 5, 6, 7, 8, 9, 10, 11, 12, 78, 1, 2, 3],
#      [3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78, 1, 2],
#      [2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78, 1],
#      [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78],
#      [78, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12],
#      [12, 78, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11],
#      [11, 12, 78, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10],
#      [10, 11, 12, 78, 1, 2, 3, 4, 5, 6, 7, 8, 9],
#      [9, 10, 11, 12, 78, 1, 2, 3, 4, 5, 6, 7, 8],
#      [8, 9, 10, 11, 12, 78, 1, 2, 3, 4, 5, 6, 7],
#      [7, 8, 9, 10, 11, 12, 78, 1, 2, 3, 4, 5, 6],
#      [6, 7, 8, 9, 10, 11, 12, 78, 1, 2, 3, 4, 5],
#      [5, 6, 7, 8, 9, 10, 11, 12, 78, 1, 2, 3, 4],
#      ]

```

Exemplos de arquivos que não correspondem a nenhum padrão desejado:

```

1 matrizEscadinhaReversa = [
#      [5, 6, 7, 8, 9, 10, 11, 12, 78, 1, 2, 3, 4],
#      [6, 7, 8, 9, 10, 11, 12, 78, 1, 2, 3, 4, 5],
#      [7, 8, 9, 10, 11, 12, 78, 1, 2, 3, 4, 5, 6],
#      [8, 9, 10, 11, 12, 78, 1, 2, 3, 4, 5, 6, 7],
#      [9, 10, 11, 12, 78, 1, 2, 3, 4, 5, 6, 7, 8],
#      [10, 11, 12, 78, 1, 2, 3, 4, 5, 6, 7, 8, 9],
#      [11, 12, 78, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10],
#      [12, 78, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11],
#      [78, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12],
#      [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78],
#      [2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78, 1],
#      [3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78, 1, 2]
#      ]

```

```

2 matrizInutilizavel_1 =[
#      [11, 12, 78, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10],
#      [10, 11, 12, 78, 1, 2, 3, 4, 5, 6, 7, 8, 9],
#      [9, 10, 11, 12, 78, 1, 2, 3, 4, 5, 6, 7, 8],
#      [8, 9, 10, 11, 12, 2, 1, 78, 3, 4, 5, 6, 7],
#      [7, 8, 9, 10, 11, 12, 78, 1, 2, 3, 4, 5, 6],
#      [6, 7, 8, 9, 10, 11, 12, 78, 1, 2, 3, 4, 5],
#      [5, 6, 7, 8, 9, 10, 11, 12, 78, 1, 2, 3, 4],
#      [4, 5, 6, 7, 8, 9, 10, 11, 12, 78, 1, 2, 3],
#      [3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78, 1, 2],
#      [2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78, 1],
#      [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78],
#      [78, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]
#      ]

```

```

3 matrizInutilizavel_2 =[
#      [11, 12, 1, 2, 78, 3, 4, 5, 6, 7, 8, 9, 10],
#      [10, 11, 12, 78, 1, 2, 3, 4, 5, 6, 7, 8, 9],
#      [9, 10, 11, 12, 1, 2, 3, 4, 5, 6, 7, 8, 78],
#      [8, 9, 10, 11, 12, 2, 1, 78, 3, 4, 5, 6, 7],
#      [7, 8, 9, 10, 11, 12, 78, 1, 2, 3, 4, 5, 6],
#      [6, 7, 8, 9, 10, 11, 12, 78, 1, 2, 3, 4, 5],
#      [5, 6, 7, 8, 9, 10, 11, 12, 78, 1, 2, 3, 4],
#      [4, 5, 6, 7, 8, 9, 10, 11, 12, 78, 1, 2, 3],
#      [3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78, 1, 2],
#      [2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78, 1],
#      [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78],
#      [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 78]
#      ]

```

OBS: Os exemplos acima são meramente ilustrativos. Lembre-se que cada linha e coluna terá valores diferentes, e não repetidos como nas matrizes acima. Os exemplos foram descritos assim para melhor entendimento. Sempre existirão valores inteiros para cada mês, e pelo menos uma coluna com o valor total correspondente à soma dos valores dos meses.

Para solucionar o problema, entregue um código que:

1) Verifica se há escadinha, dada uma matriz como os exemplos acima.

- 2) Se houver escadinha, mapear os dados corretamente, retornando a matriz correta.
- 3) Se não houver escadinha, retorne se a matriz está com uma formatação que pode ser utilizada (padrão correto) ou se corresponde a um padrão inutilizável.
- 4) Caso fossem possíveis números negativos nos valores dos meses (não no total), a sua solução ainda seria válida? Caso não, como seria a nova solução?

Questão 2) Você tem duas tabelas em um banco de dados relacional.

Tabela Vendedor:

vendedor_id (int)

vendedor_nome (string)

Tabela Venda:

venda_id (int)

vendedor_id (int)

venda_data (data dd/mm/aaaa)

venda_valor (double)

Retorne a lista com a venda de valor mais alto de cada vendedor para o ano de 2016.
Não utilize as funções Min/Max.

Escreva seu código em python(questão 1) e SQL(questão 2), coloque sua solução em um repositório github e compartilhe conosco!

Ah, não se preocupe se não conseguir implementar todo o código! Você pode escrever qual a sua idéia de solução também!

Boa Sorte!