

Tabela de Vendas

Você foi contratado para trabalhar em uma empresa do ramo alimentício. Nessa empresa existem registros sobre a quantidade de produtos vendidos mensalmente, de Setembro até Novembro de 2020, em um arquivo no formato [CSV](#).

Arquivos do tipo [CSV](#) armazenam dados em formato tabular. A primeira linha é o cabeçalho, indicando o nome de cada coluna da tabela. Nas linhas seguintes, existem valores atribuídos a cada uma das colunas. Os valores de cada linha são separados por vírgulas.

O seu chefe quer que você faça um programa que, dada a tabela de produtos vendidos por mês, fará a ordenação pelas regras que ele indicará. Ele sempre indicará as colunas que deverão ser utilizadas para a ordenação, da coluna mais importante para a menos importante. Não necessariamente todas as colunas serão utilizadas para a ordenação. Em caso de empate entre os itens na ordenação (considerando todas as colunas indicadas para ordenação), deverá ser mantido a ordem inicial dos itens.

Para a ordenação, você poderá usar [métodos de ordenação estáveis](#), que são métodos que preservam a ordem de itens de valores iguais, como [Bubble Sort](#) e [Insertion Sort](#).

O exemplo a seguir mostra dados desordenados em relação à coluna Quantidade:

Produto	Quantidade
Margarina	20
Manteiga	5
Farinha	20
Fermento	15
Pao	40

Algoritmos de ordenação estáveis mantêm a ordem inicial dos dados em caso de empate. Na ordenação a seguir, Margarina aparece primeiro que Farinha após a ordenação, seguindo a propriedade de ordenação estável:

Produto	Quantidade
Manteiga	5
Fermento	15
Margarina	20

Farinha	20
Pao	40

Utilizando algoritmos não estáveis, essa propriedade não é garantida, conforme a ordenção a seguir, onde Farinha aparece antes de Margarina:

Produto	Quantidade
Manteiga	5
Fermento	15
Farinha	20
Margarina	20
Pao	40

Quando existe uma lista de prioridades para a ordenação, basta executar uma ordenação estável para cada um dos itens da lista, começando pelo critério de menor prioridade até o critério com maior prioridade.

O seu programa irá receber um valor **N**, indicando a quantidade de linhas que existem na tabela, excluindo o cabeçalho. Em seguida, seu programa irá receber dados no formato **CSV** (não será feita a leitura de um arquivo CSV, mas sim dados no formato CSV). A primeira linha dos dados consiste no cabeçalho da tabela, que por padrão irá conter "Produto" e o nome dos meses do ano, separados por vírgulas. Na sequência, seu programa deverá ler **N** linhas, contendo os nomes dos produtos e as quantidades vendidas em cada mês, separados por vírgulas. Ao final, o seu programa irá receber a ordem de prioridade das colunas que o seu chefe indicou para a ordenação das linhas da tabela, começando da coluna com maior prioridade até a coluna com menor prioridade. A lista de colunas será dada numa única linha, com os nomes das colunas separados por um espaço.

A saída deverá ser a tabela ordenada seguindo os critérios indicados pelo seu chefe. Na saída, a primeira linha deverá ser o cabeçalho da tabela. A ordenação final deverá seguir as propriedades de ordenação estável.

Exemplos de entradas e saídas esperadas pelo seu programa:

Teste 01

Entrada

```
5
Produto,Setembro,Outubro,Novembro
Manteiga,5,4,5
Margarina,20,90,8
Farinha,20,22,25
Fermento,15,17,13
```

Pao,40,50,48
Setembro Produto

Saída

Produto	Setembro	Outubro	Novembro
Manteiga	5	4	5
Fermento	15	17	13
Farinha	20	22	25
Margarina	20	90	8
Pao	40	50	48

Teste 02

Entrada

5
Produto,Setembro,Outubro,Novembro
Manteiga,50,45,55
Margarina,20,9,87
Farinha,20,22,25
Fermento,15,17,13
Pao,40,50,48
Produto Setembro

Saída

Produto	Setembro	Outubro	Novembro
Farinha	20	22	25
Fermento	15	17	13
Manteiga	50	45	55
Margarina	20	9	87
Pao	40	50	48

Teste 03

Entrada

5
Produto,Setembro,Outubro,Novembro
Manteiga,5,4,5
Margarina,20,9,87
Farinha,20,22,25
Fermento,15,17,13
Pao,40,50,48
Novembro Setembro Produto

Saída

Produto	Setembro	Outubro	Novembro
Manteiga	5	4	5
Fermento	15	17	13
Farinha	20	22	25
Pao	40	50	48
Margarina	20	9	87

Código Base

No arquivo auxiliar lab13.py você irá encontrar um código base para dar início ao processo de elaboração desta tarefa. Para facilitar a implementação do seu programa, o código base contém a saída de dados, conforme mostrado a seguir:

```
for linha in dados:
    print('{:15s}'.format(linha[0]), ''.join('{:>10}'.format(item) for item in lir
```

Orientações

- Veja [aqui](#) a página de submissão da tarefa.
- O arquivo a ser submetido deve se chamar lab13.py.
- No link "Arquivos auxiliares" há um arquivo compactado (aux13.zip) que contém todos os arquivos de testes abertos (entradas e saídas esperadas).
- O laboratório é composto de 10 testes abertos e 10 testes fechados.
- O limite máximo será de 20 submissões.
- Acesse o sistema SuSy com seu RA (apenas números) e a senha que você utiliza para fazer acesso ao sistema da DAC.
- Você deve seguir as instruções de submissão descritas no enunciado.
- Serão considerados apenas os resultados da última submissão.
- Esta tarefa tem peso 3.
- O prazo final para submissão é dia 17/01/2021 (domingo).