

INSTRUÇÕES GERAIS

1. A avaliação é individual e sem consulta.
 2. A única linguagem de programação aceita é Python 3.
 3. Só é permitido o uso de folhas fornecidas pelo professor.
 4. Questões sem desenvolvimento são consideradas incorretas.
 5. As questões podem apresentar restrições em seu enunciado. Será atribuída nota ZERO para a questão caso alguma de suas restrições não seja cumprida.
 6. Os programas escritos devem funcionar para qualquer caso de entrada possível definido pelo enunciado, não apenas para o(s) exemplo(s) dado(s).
 7. O(A) estudante não pode sair da sala durante a avaliação. Caso o faça, então deve encerrar a prova e entregar as soluções produzidas até o momento.
 8. Conversas e trocas de qualquer tipo de mensagem verbal ou textual são proibidas durante a avaliação e são consideradas tentativa de “cola”.
 9. É proibido o uso de smartphone, tablet, notebook ou qualquer outros dispositivos eletrônicos durante a prova, sendo o acesso aos mesmos considerado tentativa de “cola”.
 10. Será atribuída nota ZERO aos estudantes que: (i) forem pegos tentando “colar”; (ii) forem pegos tentando “passar cola”; (iii) que tenham “colado”, ou (iv) que tenham “passado cola”. A coordenação do curso será notificada do ocorrido para tomar as providências cabíveis.
-

QUESTÃO 1 - Inventário de Supermercados

Você foi contratado(a) para escrever um programa para uma rede de supermercados. O problema que tal rede quer que você resolva utilizando Python básico e os conhecimentos adquiridos na disciplina de Programação de Computadores I é a consulta ao inventário de produtos existentes nas lojas.

Mais especificamente, dados arquivos texto contendo as listagens de produtos e quantidades existentes em cada loja, a rede de supermercados está interessada em informar o nome parcial de algum produto, obter a quantidade total de itens com esse nome em cada loja e o somatório dessas quantidades em todas as lojas.

Exemplo de Entrada e Saída

A interface do seu programa deverá corresponder à seguinte interação:

```
Informe um arquivo de inventário, ou "fim" para encerrar: arquivo1.txt
Informe um arquivo de inventário, ou "fim" para encerrar: arquivo2.txt
Informe um arquivo de inventário, ou "fim" para encerrar: arquivo3.txt
Informe um arquivo de inventário, ou "fim" para encerrar: fim
Informe um produto, ou "0" para encerrar: leite
59 unidades em "Loja Niterói"
91 unidades em "Loja São Gonçalo"
0 unidades em "Loja Maricá"
150 unidades no total
Informe um produto, ou "0" para encerrar: vassoura
78 unidades em "Loja Niterói"
0 unidades em "Loja São Gonçalo"
20 unidades em "Loja Maricá"
98 unidades no total
Informe um produto, ou "0" para encerrar: 0
```

Os textos digitados pelo(a) usuário(a) na entrada padrão estão sublinhados para facilitar a identificação. O restante é impresso pelo seu programa na saída padrão. Observe que são solicitados caminhos de arquivos até que a palavra “fim” (letras minúsculas e sem aspas) seja informada pelo(a) usuário(a). Depois, são solicitados nomes parciais de produtos até que “0” (sem aspas) seja informado.

Nos arquivos informados pelo usuário, a primeira linha identifica a loja e as linhas seguintes identificam o nome do produto e a quantidade (valor inteiro), separados por ponto e vírgula. Os arquivos foram gerados utilizando UTF-8 e quebra de linha no padrão Linux. O nome dos produtos não possuem acentos e pontuação, mas podem conter espaços em branco, caracteres numéricos, letras maiúsculas ou minúsculas.

Abaixo está o conteúdo dos arquivos utilizados no exemplo:

arquivo1.txt	arquivo2.txt	arquivo3.txt
Loja Niterói Creme de leite;50 Margarina;60 Vassoura;78 Leite;9	Loja São Gonçalo Manga;42 Leite;91 Arroz branco;65	Loja Maricá Frango congelado;71 Vassoura;20 Margarina;4

Restrições

- 1) Seu programa deve funcionar com qualquer entrada que siga o formato especificado, não só para o exemplo dado no enunciado.
- 2) Os arquivos só podem ser percorridos uma vez cada um e novos arquivos não podem ser criados.
- 3) Não é permitido alterar o conteúdo das variáveis `sys.stdin` e `sys.stdout`.
- 4) Não é permitido utilizar bibliotecas como `pandas`, `numpy` ou similares, nem banco de dados.

Dica

Convém criar uma função que recebe como argumento o caminho para um arquivo de inventário e retorna o nome da loja e uma lista contendo o nome e a quantidade de cada produto, pois essa função poderá ser reutilizada na próxima questão.

Distribuição de Pontos

Esta questão vale 6,0 dos 10,0 pontos da avaliação. Os pontos são distribuídos em:

- Q1.1 - Leitura e escrita na entrada e saída padrão, e fluxo principal do programa: 1,0
- Q1.2 - Abertura e leitura dos arquivos de entrada: 1,0
- Q1.3 - Organização dos dados lidos de arquivos: 2,0
- Q1.4 - Contagem de produtos por loja: 1,0
- Q1.5 - Contagem total de produtos: 1,0

QUESTÃO 2 - Organização de Supermercados

A rede de supermercados que contratou você para resolver o problema descrito na Questão 1 ficou tão satisfeita com seu trabalho que resolveu contratá-lo(a) novamente, mas agora para organizar as lojas.

Dado um arquivo de inventário que segue o mesmo formato adotado na Questão 1, seu programa deverá gerar outro arquivo que organiza os produtos primeiro em ordem decrescente de quantidade e, em caso de empate, em ordem crescente de nome (ordem alfabética).

Entrada

Pela entrada padrão é informado o caminho para um arquivo de inventário que segue o mesmo formato especificado na questão anterior.

Saída

Seu programa deverá escrever um arquivo texto que segue o mesmo formato especificado para o arquivo de entrada. O nome do arquivo de saída deverá ser “arquivo_organizado.txt”.

Exemplos

Para os arquivos de entrada exemplificados anteriormente, os arquivos de saída esperados são:

arquivo_organizado.txt	arquivo_organizado.txt	arquivo_organizado.txt
Loja Niterói Vassoura;78 Margarina;60 Creme de leite;50 Leite;9	Loja São Gonçalo Leite;91 Arroz branco;65 Manga;42	Loja Maricá Frango congelado;71 Vassoura;20 Margarina;4

Restrições

- 1) Seu programa deve funcionar com qualquer arquivo de entrada que siga o formato especificado, não só para o exemplo dado no enunciado.
- 2) Deve ser utilizada subprogramação na ordenação, leitura e escrita de arquivos. A função de leitura recebe como argumento o caminho para um arquivo de inventário e retorna o nome da loja e uma lista contendo o nome e a quantidade de cada produto. O procedimento de escrita recebe como argumentos o nome da loja e a lista de produtos já ordenada. Já o procedimento de ordenação recebe a lista de produtos e realiza a ordenação sobre a própria lista, sem a criação de listas auxiliares.
- 3) A ordenação deve ser feita pela implementação de um dos três algoritmos de ordenação vistos em sala de aula. Logo, não é permitido o uso do método `.sort`, da função `sorted` ou similares.
- 4) Não é permitido alterar o conteúdo das variáveis `sys.stdin` e `sys.stdout`.
- 5) Não é permitido utilizar bibliotecas como `pandas`, `numpy` ou similares, nem banco de dados.

Distribuição de Pontos

Esta questão vale 4,0 dos 10,0 pontos da avaliação. Os pontos são distribuídos em:

- Q2.1 - Leitura da entrada padrão e uso de subprogramação na leitura do arquivo: 0,5
- Q2.2 - Escrita de arquivo conforme especificado e usando subprogramação: 1,5
- Q2.3 - Ordenação conforme especificado e usando subprogramação: 2,0

A implementação da leitura do arquivo já foi pontuada na questão anterior. Caso a dica dada tenha sido seguida, a função criada anteriormente poderá ser utilizada nesta questão sem a necessidade de implementação redundante. Caso a dica não tenha sido seguida, então é necessário implementar a função, pois a subprogramação faz parte da avaliação do quesito Q2.1.