

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 - “Bora” Organizar as Festas

A turma que ingressou em 2023/2 no curso de Bacharelado em Ciência da Computação já está se organizando para festejar todos os aniversários do ano que vem. Para isso, fizeram uma votação e coube a você usar os conhecimentos adquiridos em Programação de Computadores I para resolver o problema.

Dado um arquivo texto contendo a data de nascimento e o nome de cada estudante da turma, escreva um programa que produza outro arquivo texto colocando os(as) estudantes em ordem crescente de data de aniversário. Em caso de empate, o critério de desempate é o nome em ordem alfabética.

Entrada

O nome do arquivo texto de entrada é “turma_2023_2.txt” e foi escrito com UTF-8 e quebra de linha no padrão adotado pelo Linux. Cada linha do arquivo contém a data de nascimento de um(a) estudante no formato dia/mês/ano, seguido por um espaço em branco e o nome completo do(a) estudante. Os nomes não incluem acentos nem cedilhas, mas podem conter letras maiúsculas, minúsculas e espaços em branco.

Saída

O programa deve produzir como saída o arquivo “aniversarios.txt”, utilizando UTF-8 e quebra de linha no padrão adotado pelo Linux. Cada linha do arquivo de saída deve indicar a data de aniversário do(a) estudante, seguido por um espaço em branco e o nome completo conforme informado na entrada. A saída deve estar organizada em ordem crescente de data de aniversário no formato dia/mês e, caso dois ou mais estudantes façam aniversário na mesma data, o empate é resolvido por ordem alfabética de nome.

Exemplo

Entrada (arquivo <code>turma_2023_2.txt</code>)	Saída (arquivo <code>aniversarios.txt</code>)
5/5/2004 Fulano de Carvalho 10/11/2005 Deltrano das Flores 31/3/2005 Beltrana Silva 10/11/2005 Cicrano de Tal	31/3 Beltrana Silva 5/5 Fulano de Carvalho 10/11 Cicrano de Tal 10/11 Deltrano das Flores

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) Não é permitido alterar o conteúdo das variáveis `sys.stdin` e `sys.stdout`.
- 3) Não é permitido o uso do módulo `datetime` ou de módulos similares.
- 4) 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.

Distribuição de Pontos

Esta questão vale 5,0 dos 10,0 pontos da avaliação. Os pontos são distribuídos em:

- Q1.1 - Abertura e leitura do arquivo de entrada: 1,5
- Q1.2 - Ordenação dos aniversários: 2,0
- Q1.3 - Abertura e escrita do arquivo de saída: 1,5

QUESTÃO 2 - Estatística em Documentos

A Biblioteca Central da Universidade Federal Fluminense chamou você para um projeto que tem por objetivo gerar estatísticas sobre o conteúdo dos livros contidos nas bibliotecas dos campi da Universidade. Para resolver esse problema, você deverá escrever um programa que recebe como entrada uma lista de arquivos textos que representam livros escaneados na biblioteca e, para cada um dos arquivos, seu programa deve contar:

- a) A quantidade de linhas.
- b) A quantidade de palavras.
- c) A quantidade de caracteres, desconsiderando espaços em branco, tabulações e quebras de linha.

Entrada

O usuário informará pela entrada padrão o caminho dos arquivos que devem ser processados, um por linha. O término da lista de caminhos é identificado com a palavra "FIM" (sem aspas, em letras maiúsculas).

Cada arquivo de entrada é composto por uma ou mais linhas que, por sua vez, são compostas por uma ou mais palavras separadas por um ou mais espaços em branco, tabulações e quebras de linha.

É assumido o uso de UTF-8 e de quebra de linha conforme o padrão adotado pelo Linux.

Saída

Para cada arquivo de entrada o programa deve imprimir, na saída padrão, uma linha com o caminho do arquivo, seguido por três linhas contendo, respectivamente, "Linhas: A", "Palavras: B", "Caracteres: C" e uma linha em branco, onde A, B e C são valores inteiros resultantes das contagens nos itens (a), (b) e (c).

Exemplo

Arquivo calculo_avancado.txt	Arquivo programacao_basica.txt	Arquivo introducao_a_algebra.txt
Este livro é destinado a estudantes que buscam aprender mais sobre Cálculo Diferencial e Integral.	Uma boa maneira de organizar o fluxo de desenvolvimento de um programa é a aplicação do método de Pólya.	Dedico este livro aos apaixonados por manipulação algébrica.

Entrada Padrão	Saída Padrão
calculo_avancado.txt programacao_basica.txt introducao_a_algebra.txt FIM	calculo_avancado.txt Linhas: 4 Palavras: 15 Caracteres: 84 programacao_basica.txt Linhas: 5 Palavras: 19 Caracteres: 86 introducao_a_algebra.txt Linhas: 3 Palavras: 8 Caracteres: 53

Restrições

- 1) Seu programa deve funcionar com qualquer entrada que siga o formato especificado, não só para o exemplo dado no enunciado.
- 2) Não é permitido alterar o conteúdo das variáveis `sys.stdin` e `sys.stdout`.

Distribuição de Pontos

Esta questão vale 3,0 dos 10,0 pontos da avaliação. Os pontos são distribuídos em:

- Q2.1 - Leitura e escrita na entrada e saída padrão: 0,2
- Q2.2 - Abertura e leitura dos arquivos de entrada: 0,5
- Q2.3 - Contagem de linhas: 0,7
- Q2.4 - Contagem de palavras: 0,8
- Q2.5 - Contagem de caracteres: 0,8

QUESTÃO 3 - Troca de Padrão

A escrita de números em inglês adota a vírgula como separador de milhar e o ponto como separador decimal. Já a escrita de números em português adota a vírgula como separador decimal e o ponto como separador de milhar. Ou seja, o mesmo número pode ser escrito como 5,432.10 ou 5.432,10, dependendo do padrão adotado.

Escreva um programa que recebe como entrada uma sequência de strings que representam números no formato adotado na língua portuguesa e converte a representação para o formato da língua inglesa.

Entrada

Os valores no formato adotado na língua portuguesa são fornecidos via entrada padrão, um por linha, sendo o final da sequência de valores identificada pela palavra "FIM" (sem aspas, em letras maiúsculas). Os valores são compostos apenas por dígitos numéricos, pontos, vírgulas e sinal.

Saída

Para cada valor fornecido na entrada, seu programa deverá escrever na saída padrão o mesmo valor, mas no formato adotado na língua inglesa. Caso o número dado como entrada apresente o sinal, então o mesmo sinal deve ser colocado na saída. O número de casas decimais também deve ser preservado.

Exemplo

Entrada	Saída
5.432,10	5,432.10
-60	-60
10.000	10,000
2.145.648,23564	2,145,648.23564
+3,14	+3.14
2	2
0	0
FIM	

Restrição

- 1) Deve ser criada e utilizada uma função que recebe como argumento a string com o número no formato adotado na língua portuguesa e retorna uma string com o número no formato adotado na língua inglesa.
- 2) Não é permitido o uso de rotinas prontas para conversão entre padrões de escrita de números, tais como as rotinas presentes no módulo `locale` ou similares.

Distribuição de Pontos

Esta questão vale 2,0 dos 10,0 pontos da avaliação. Os pontos são distribuídos em:

Q3.1 - Leitura e escrita conforme especificado: 0,5

Q3.2 - Conversão entre padrões: 1,5