



Fundação CECIERJ - Vice-Presidência de Educação Superior a Distância

**Curso de Tecnologia em Sistemas de Computação
Disciplina Fundamentos de Programação**

AD2 – 2º semestre de 2018

ERRATA

- [15/10/2018] Esta versão da folha de prova inclui uma correção no exemplo da Questão 4.

IMPORTANTE

- As respostas (programas) deverão ser entregues pela plataforma em um arquivo ZIP contendo todos os arquivos de código fonte (extensão “.py”) necessários para que os programas sejam testados. Respostas entregues fora do formato especificado, por exemplo, em arquivos com extensão “.pdf”, “.doc” ou outras, não serão corrigidas.
- Serão aceitos apenas soluções escritas na linguagem Python 3. Programas com erro de interpretação não serão corrigidos. Evite problemas utilizando tanto a versão da linguagem de programação (Python 3.X) quanto a IDE (PyCharm) indicadas na Aula 1.
- Quando o enunciado de uma questão inclui especificação de formato de entrada e saída, tal especificação deve ser seguida à risca pelo programa entregue. Atender ao enunciado faz parte da avaliação e da composição da nota final.
- Faça uso de boas práticas de programação, em especial, na escolha de identificadores de variáveis, subprogramas e comentários no código.
- As respostas deverão ser entregues pela atividade "Entrega de AD2" antes da data final de entrega estabelecida no calendário de entrega de ADs. Não serão aceitas entregas tardias ou substituição de respostas após término do prazo.
- As ADs são um mecanismo de avaliação individual. As soluções podem ser buscadas por grupos de alunos, mas a redação final de cada prova tem que ser individual. Respostas plagiadas não serão corrigidas.
- Os exemplos fornecidos nos enunciados das questões correspondem a casos específicos apontados para fins de ilustração e não correspondem ao universo completo de entradas possíveis especificado no enunciado. Os programas entregues devem ser elaborados considerando qualquer caso que siga a especificação e não apenas os exemplos dados. Essa é a prática adotada tanto na elaboração das listas exercícios desta disciplina quanto no mercado de trabalho.

1ª Questão (2,0 pontos)

Faça um programa que leia da entrada padrão o nome de um arquivo texto, escolhido pelo usuário, contendo um número de ponto flutuante por linha. Sem manter o conteúdo do arquivo em memória principal, leia apenas uma vez o arquivo e escreva na saída padrão o menor número lido, a média dos números lidos e o maior número lido. Caso o arquivo esteja vazio, escreva `None` como o menor a média e o maior.

Entrada

O formato de entrada deve ser deduzido dos exemplos.

Saída

O formato de saída deve ser deduzido dos exemplos.

Exemplos

Entrada	
Entrada Padrão	Arquivo Ex1.txt
Ex1.txt	<i>Arquivo deixado em branco de propósito</i>
Saída Padrão	
Menor Valor = None Média dos Valores = None Maior Valor = None	

Entrada	
Entrada Padrão	Arquivo Ex1.txt
Ex2.txt	1 2 3 4.444 -2.222 13.13 80.77 5 7 12
Saída Padrão	
Menor Valor = -2.222 Média dos Valores = 12.6122 Maior Valor = 80.77	

2ª Questão (2,5 pontos)

Faça um programa que leia arquivo texto, escolhido pelo usuário, contendo dois números de ponto flutuante por linha, representando coordenadas x e y de pontos no espaço bidimensional. Suponha que todos os arquivos processados por este programa sempre possuam pelo menos um ponto. Sem copiar o conteúdo do arquivo para a memória principal, escreva quais são os dois pontos mais distantes entre si, e qual a distância que os separa.

Definição

A distância entre dois pontos (x_1, y_1) e (x_2, y_2) é dada pela raiz quadrada do quadrado da diferença entre os x's e os y's. Ou seja: o quadrado de $(x_1 - x_2)$ somado ao quadrado de $(y_1 - y_2)$.

Entrada

O formato de entrada deve ser deduzido dos exemplos.

Saída

O formato de saída deve ser deduzido dos exemplos.

Exemplos

Entrada	
Entrada Padrão	Arquivo Ex1.txt
Ex1.txt	1 2
Saída Padrão	
Ponto Inicial = (1.0, 2.0) Ponto Final = (1.0, 2.0) Distância entre eles = 0.0	

Entrada	
Entrada Padrão	Arquivo Ex2.txt
Ex2.txt	1 2 3 4
Saída Padrão	
Ponto Inicial = (1.0, 2.0) Ponto Final = (3.0, 4.0) Distância entre eles = 2.8284271247461903	

Entrada	
Entrada Padrão	Arquivo Ex3.txt
Ex3.txt	3.22 44.1 100 200 -37.22 80.4 1 10 10 1 13 26 26 130.99 65.2 65.1
Saída Padrão	
Ponto Inicial = (100.0, 200.0) Ponto Final = (10.0, 1.0) Distância entre eles = 218.4055860091495	

3ª Questão (2,5 pontos)

Pedro montou um site na Internet para vender figurinhas do álbum do Campeonato Brasileiro. Ele está organizando um catálogo com conjuntos distintos de figurinhas para vender. Cada conjunto de figurinhas é formado por 5 figurinhas, sendo uma de cada time que ele tem figurinhas à disposição (Botafogo, Flamengo, Fluminense, Vasco e São Paulo). Dois conjuntos de figurinhas são considerados distintos se existe pelo menos uma figurinha que está em um e não está no outro conjunto. Pedro quer expor apenas os conjuntos distintos mais caros.

O valor de um conjunto é a soma dos valores de cada figurinha que está nele. Sua tarefa é informar qual a soma dos valores dos K conjuntos distintos de figurinhas mais caras. Em caso de empate entre conjuntos mais caros, Pedro escolhe qualquer um dos conjuntos empatados.

Entrada

A entrada é dada por um arquivo texto chamado "entrada.txt". Este arquivo possui 6 linhas: A primeira linha contém um inteiro B ($5 \leq B \leq 10$), representando que Pedro tem B tipos diferentes de figurinhas do Botafogo, seguido por B inteiros X_i ($1 \leq X_i \leq 20$), representando os valores de cada figurinha desse time. A segunda linha contém um inteiro F ($5 \leq F \leq 10$), representando que Pedro tem F tipos diferentes de figurinhas do Flamengo, seguido por F inteiros X_i ($1 \leq X_i \leq 20$), representando os valores de cada figurinha do Flamengo. A terceira linha contém um inteiro L ($5 \leq L \leq 10$), representando que Pedro tem L tipos diferentes de figurinhas do Fluminense, seguido por L inteiros X_i ($1 \leq X_i \leq 20$), representando os valores de cada figurinha do Fluminense. A quarta linha contém um inteiro V ($5 \leq V \leq 10$), representando que Pedro tem V tipos diferentes de figurinhas desse time, seguido por V inteiros X_i ($1 \leq X_i \leq 20$), representando os valores de cada figurinha do Vasco. A quinta linha contém um inteiro S ($5 \leq S \leq 10$), representando que Pedro tem S tipos diferentes de figurinhas do São Paulo, seguido por S inteiros X_i ($1 \leq X_i \leq 20$), representando os valores de cada figurinha desse time. A sexta linha contém um inteiro K ($1 \leq K \leq B \times F \times L \times V \times S$), representando a quantidade de conjuntos distintos de figurinhas que o catálogo de figurinhas terá.

Saída

Seu programa deverá imprimir na saída padrão o valor da soma dos valores dos K conjuntos distintos de figurinhas mais caros.

Exemplo

Entrada (arquivo <code>entrada.txt</code>)	Saída Padrão
5 2 5 6 3 8 6 9 6 3 1 5 3 5 4 8 5 2 6 8 3 2 4 9 5 9 10 1 5 7 8 5 1 4 10	413

4ª Questão (3,0 pontos)

Errata: O exemplo apresentado foi atualizado, pois continha um erro de digitação. A porção alterada está marcada em vermelho.

Para aumentar a arrecadação anual, o governo criou um IPPE – Imposto por Palavra Escrita. Mensalmente é divulgada uma coleção de palavras e o valor em real cobrado por seu uso em mensagens de texto. Cada mensagem escrita é então taxada conforme a quantidade de ocorrências de cada palavra na coleção, multiplicado pelo valor do imposto da respectiva palavra. Mas uma medida provisória aprovada logo em seguida permite que a cobrança seja feita até, no máximo, o segundo uso de cada palavra. Para a alegria dos cidadãos, o terceiro uso em diante está liberado de cobrança.

Entrada

A entrada é dada por um arquivo binário e um arquivo texto.

O arquivo binário de nome “referencia.bin” é composto por uma sequência de registros formatos cada um por uma string que ocupa 256 bytes (uma palavra) e um número em ponto flutuante de precisão dupla (o valor do imposto sobre a respectiva palavra). O fim da sequência de registros é sinalizado pelo término do arquivo.

O arquivo texto contém a mensagem escrita que será taxada pelo novo imposto. O texto nesse arquivo é composto por múltiplas linhas e as palavras podem aparecer tanto escritas com letras minúsculas quanto com letras maiúsculas. O nome do arquivo é dado pelo usuário através da entrada padrão.

As palavras poderão aparecer tanto nos registros do arquivo binário quanto no arquivo texto com letras maiúsculas quanto com letras minúsculas. Seu programa deverá ser capaz de identifica-las como iguais, independentes do caso. Ou seja, “Abacaxi”, “abacaxi” e “ABACAXI” correspondem à mesma palavra. Além disso, o arquivo texto pode incluir pontuação (ponto, vírgula, etc.), que deverá ser ignorada pelo seu programa. Nenhuma palavra nos arquivos será acentuada.

Saída

Seu programa deverá emitir um arquivo texto de nome “imposto.txt” com a mensagem “O imposto cobrado é de R\$<VALOR>”, onde <VALOR> deve ser substituído pelo valor total do imposto cobrado sobre o arquivo texto informado na entrada, com duas casas decimais.

Exemplo

O exemplo que segue mostra o conteúdo do arquivo binário na entrada de forma textual apenas para facilitar a interpretação dos dados. Você deve estar ciente que o arquivo “referencia.bin” é binário, tendo cada registro armazenado conforme a especificação.

Entrada	
Arquivo referencia.bin	Entrada Padrão
Voce 0.05 Dica 0.75 Obrigado 0.30 Tempo 1.40 Tudo 0.01 Abacaxi 0.89 EU 0.34 As 0.01 De 0.02 Novamente 1.34 Caso 0.21	uma_carta.txt
Arquivo uma_carta.txt	
Ola amigo, Quanto tempo! Tudo bem com voce? Esses dias lembrei daquela vez que eu, voce e Amanda estávamos conversando sobre as ADs do CEDERJ. Voce nos falou para fazer todas as questoes de Fundamentos de Programacao e procurar os tutores em caso de duvida. A dica que voce deu foi muito boa. Obrigado! Um abraço, nos veremos novamente no Polo.	
Saída (arquivo imposto.txt)	
O imposto cobrado é de R\$4.51	

Observações

Se a questão for resolvida considerando arquivos “referencia.bin” textuais então a nota atribuída para a mesma será 0 (zero), mesmo que a solução esteja correta no contexto de arquivos texto.

Dicas

Para fins de depuração de código, escreva um programa auxiliar que crie arquivos binários contendo nada mais do que valores de entrada armazenados conforme a especificação do enunciado. O programa auxiliar não deve ser entregue junto com a solução da questão. Caso seja entregue, o mesmo não será considerado na correção, independentemente de estar correto ou errado.

Os tamanhos (quantidade de bytes) assumidos para formatos nativos de valores inteiros e de valores em ponto flutuante lidos ou escritos de arquivos binários podem variar de plataforma para plataforma. Ou seja, podem ocorrer problemas de compatibilidade entre programas que rodam perfeitamente em computadores que assumem determinados tamanhos para tipos primitivos, mas que não rodam corretamente em computadores que assumem outros tamanhos para o mesmo tipo. Para forçar a leitura e escrita assumindo os tamanhos padrão (standard) que são indicados na Aula 12 e ficar livre de problemas de compatibilidade, inclua o símbolo “=” na frente do formato indicado nas funções `.pack` e `.unpack` de `struct`. Por exemplo, `struct.unpack("i", bloco)` converte o bloco de bytes em um valor inteiro, mas o tamanho do bloco é dependente da plataforma (não é necessariamente de 4 bytes), enquanto que `struct.unpack(“=i”, bloco)` converte blocos de 4 bytes em valores inteiros, independentemente da plataforma.

Boa Avaliação!