



Fundação CECIERJ - Vice-Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação
Disciplina Fundamentos de Programação

AD1 – 2º semestre de 2018

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 AD1" 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.

1ª Questão (1,5 pontos)

Faça um programa que leia da entrada padrão a quantidade de números de ponto flutuante a serem digitados. Em seguida, leia todos os números, um por linha. Identifique e escreva o menor número lido, a média dos números lidos e o maior número lido.

Exemplos

Entrada	Saída
0	Menor Lido: Nenhum Média dos Lidos: Nenhuma Maior Lido: Nenhum

Entrada	Saída
5 8.4 9.8 4.9 1.2 2.5	Menor Lido: 1.2 Média dos Lidos: 5.36 Maior Lido: 9.8

2ª Questão (1,5 pontos)

Faça um programa que leia frases da entrada padrão até que uma string vazia seja digitada. Para cada frase lida escreva a quantidade de palavras contidas na frase, a quantidade de vogais, a quantidade de dígitos, e todas as palavras que contenham número par de vogais.

Exemplo

Entrada
Foi bonita a festa junina Antes de viajar ligue pra universidade Telefone da UFF 2629-5000
Saída
Palavras contidas: 5 Total de Vogais: 11 Total de Dígitos: 0 Palavras com total par de vogais: Foi bonita festa
Palavras contidas: 6 Total de Vogais: 16 Total de Dígitos: 0 Palavras com total par de vogais: Antes universidade
Palavras contidas: 4 Total de Vogais: 6 Total de Dígitos: 8 Palavras com total par de vogais: Telefone

3ª Questão (2,0 pontos)

Faça um programa que leia as dimensões de uma matriz, isto é: quantidade de linhas e a quantidade de colunas, considere que as dimensões sempre serão maiores ou iguais a um. Em seguida leia linha(s) de valor(es) a ser(em) preenchido(s) na matriz. Mostre a matriz produzida, com uma formatação considerando um espaço em branco entre os números e todos os valores tenham no máximo dois dígitos antes do ponto flutuante e uma casa decimal de precisão.

Mostre a posição na matriz, isto é: sua linha e coluna, do menor valor lido. Escreva a média de todos os valores lidos, com uma casa decimal de precisão. Escreva as posições das células da matriz que possuam valor acima da média calculada.

Exemplo

Entrada	Saída
3 7 8 4 2 9 7 10 6 3.4 9.7 2 3 4 5 8 9.1 8.4 0 8 7 1 5.2	Chico com 1 voto(s) Matriz Lida: <div> <div>8.0</div> <div>4.0</div> <div>2.0</div> <div>9.0</div> <div>7.0</div> <div>10.0</div> <div>6.0</div> </div> <div> <div>3.4</div> <div>9.7</div> <div>2.0</div> <div>3.0</div> <div>4.0</div> <div>5.0</div> <div>8.0</div> </div> <div> <div>9.1</div> <div>8.4</div> <div>0.0</div> <div>8.0</div> <div>7.0</div> <div>1.0</div> <div>5.2</div> </div> Posição do Menor: Linha = 3, Coluna = 3 Média dos Valores: 5.7 Posições das Células Acima da Média: Linha = 1, Coluna = 1 Linha = 1, Coluna = 4 Linha = 1, Coluna = 5 Linha = 1, Coluna = 6 Linha = 1, Coluna = 7 Linha = 2, Coluna = 2 Linha = 2, Coluna = 7 Linha = 3, Coluna = 1 Linha = 3, Coluna = 2 Linha = 3, Coluna = 4 Linha = 3, Coluna = 5

4ª Questão (2,0 pontos)

Faça um programa composto por subprogramas que processam uma coleção de tuplas conforme indicado nas instruções abaixo:

(a) A entrada é composta por três grupos de valores. Você só precisa se preocupar com o primeiro grupo, pois os outros dois serão lidos pelo código disponibilizado no item (f) desta questão. O primeiro grupo de valores é composto por um valor inteiro N indicando o número de tuplas a serem processadas, seguido por N linhas contendo uma palavra e um valor inteiro não negativo cada. A palavra representa o nome de uma cidade e o valor inteiro é sua população.

(b) O programa deve implementar a função `lerCidades()`, que lê as tuplas a partir da entrada padrão e retorna um vetor contendo as N tuplas lidas;

(c) O programa deve implementar a função `obterPopulacao(cidades, nome)`, que recebe como parâmetros o vetor `cidades` contendo as tuplas e a string `nome` contendo o nome da cidade que se deseja consultar a população. A função retorna um valor inteiro que corresponde à população da cidade consultada ou -1 caso a cidade não esteja contida no vetor.

(d) O programa deve implementar a função `obterCidadesGrandes(cidades, populacaoMinima)`, que recebe como parâmetros o vetor `cidades` contendo as tuplas e um valor inteiro `populacaoMinima`. A função retorna um vetor de tamanho N composto por valores lógicos (`True` ou `False`) indicando se a i-ésima cidade no vetor de tuplas possui população maior ou igual ao valor `populacaoMinima` (valor `True`) ou menor que o valor `populacaoMinima` (valor `False`).

(e) O programa deve implementar o procedimento `imprimirNomes(cidades, quais)`, que recebe como argumentos o vetor `cidades` contendo as tuplas e o vetor de valores lógicos `quais` que indica as cidades cujos nomes devem ser impressos na saída padrão.

(f) A estrutura do programa principal deve ser a seguinte:

```
print("Informe os dados de entrada:")
tuplas = lerCidades()

print("Indique o nome de uma cidade a ser consultada: ", end="")
consulta = input()
print("A população é de", obterPopulacao(tuplas, consulta),
      "pessoa(s).")

print("Indique a população mínima requerida: ", end="")
consulta = int(input())
imprimirNomes(tuplas, obterCidadesGrandes(tuplas, consulta))
```

Restrições

Na correção serão consideradas apenas a implementação do que é pedido nos itens (b) a (e), se e somente se eles forem apresentados como subprogramas que atendam às especificações dadas, pois essa questão visa avaliar a capacidade do aluno de aplicar subprogramação. Não é permitido alterar a quantidade nem o significado de parâmetros dos subprogramas exigidos.

O programa entregue deve ser completo, composto pelo programa principal e os subprogramas especificados. Entretanto, não será atribuído pontos ao programa principal uma vez que o mesmo é dado na folha de prova.

Exemplo (a saída padrão está em preto e a entrada padrão está em azul)

Execução
Informe os dados de entrada: 3 Rio 6520266 Niterói 496696 Petrópolis 298235 Indique o nome de uma cidade a ser consultada: Niterói A população é de 496696 pessoa(s). Indique a população mínima requerida: 300000 Rio Niterói

5ª Questão (1,5 pontos)

Faça um programa que leia uma única linha da entrada padrão e armazene seu conteúdo como elementos em um vetor. Essa linha contém uma coleção de valores inteiros separados por espaços em branco. Após a leitura, o programa deve chamar os subprogramas especificados nos itens (a), (c), (b) e (c), novamente, para processar o vetor de entrada (subprogramas (a) e (b)) e exibir o resultado do processamento (subprograma (c)):

(a) O procedimento `inverter(colecao)` recebe como parâmetro um vetor contendo os valores inteiros e inverte a posição dos elementos nesse mesmo vetor.

(b) A função `menorEMaior(colecao)` recebe como parâmetros um vetor contendo valores inteiros e retorna um vetor com duas posições. A primeira posição contém o menor valor encontrado na coleção e a segunda posição contém o maior valor encontrado na coleção.

(c) O procedimento `imprimir(colecao)` recebe como parâmetro um vetor contendo valores inteiros e imprime na saída padrão seu conteúdo em uma única linha, separando os valores por espaços em branco.

Os subprogramas especificados nos itens (a), (b) e (c) devem ser resolvidos com recursão. É permitida a escrita de subprogramas auxiliares para ajudar nas tarefas, desde que a solução para o problema em si seja recursiva.

Restrições

Na correção dos itens (a), (b) e (c) serão consideradas apenas soluções que fazem uso de recursão para resolver os problemas propostos. O objetivo dessa questão é observar a capacidade do aluno de aplicar recursão. Não é permitido alterar a quantidade nem o significado de parâmetros dos subprogramas exigidos.

Exemplos

Entrada	Saída
-4 65 32 -20 56 42 44	44 42 56 -20 32 65 -4 -20 65

Entrada	Saída
0 0 0 0	0 0 0 0 0 0

6ª Questão (1,5 pontos)

A busca binária foi apresentada na Aula 6. O algoritmo apresentado em aula retorna a posição do elemento procurado caso esse esteja presente no vetor informado ou -1 caso contrário. Nessa questão você deverá escrever um programa que implementa uma variação da busca binária. Seu programa deverá utilizar o mesmo mecanismo de divisão do espaço de busca em duas partições até que o elemento procurado seja encontrado. Entretanto, aqui o vetor de entrada poderá conter várias cópias de um mesmo elemento, de modo que ao fim da busca seu procedimento deverá retornar um par de índices definindo a faixa de posições que contém o elemento procurado. Essa parte final do algoritmo pode ser resolvida sequencialmente.

A entrada consiste em uma coleção de valores inteiros dados em ordem crescente, todos dispostos na mesma linha da entrada padrão e, na linha seguinte, um valor a ser procurado.

A saída impressa esperada é a posição da primeira e da última cópia do elemento procurado, ou -1 -1 caso o valor não esteja contido na coleção de entrada.

Exemplo

Entrada	Saída
-25 -13 -13 2 2 2 2 4 65 159 2	3 6

Boa Avaliação!