



Fundação CECIERJ - Vice-Presidência de Educação Superior a Distância

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

AD1 – 1º semestre de 2020

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.
- 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.
- 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.

Boa Avaliação!

1ª Questão (2,0 pontos)

Faça um programa que leia strings da entrada padrão, até que a string vazia ("") seja digitada. Caso a primeira string lida seja vazia, escreva a mensagem "Nenhuma String Não Vazia Foi Lida!!!". Caso contrário escreva:

- (1) Qual a string que tem maior comprimento; caso haja empate escreva a primeira delas;
- (2) Qual a string possui mais dígitos, isto é, contém caracter(es) na string "0123456789". Caso haja empate escreva a última delas. Caso nenhuma possua dígitos escreva: "Nenhuma String Contém Dígito!!!"
- (3) Qual a quantidade de strings formadas apenas de vogais minúsculas e sem acento, isto é, contidas na string "aeiou".

Exemplo

Entrada	Saída
gol 47P a futebol laranja 8 x9 13 XL ou Juca 2 ué Aaaaá	Primeira de Maior Comprimento: futebol Última com Mais Dígitos: 13 XL Quantidade de Strings Apenas Com Vogais Minúsculas: 2

2ª Questão (2,0 pontos)

Utilizando sub-programação, faça um programa que leia da entrada padrão uma única linha podendo conter zero ou mais números inteiros. Caso exista(m), escreva qual(is) o(s) número(s) primo(s). Mostre-o(s) um por linha. Caso a linha lida seja uma string vazia, escreva a mensagem: "Nenhum Número Foi Lido!!!".

Exemplo

Entrada	Saída
100 15 13 8 4 2 1 0 -29 37 997 1024	Relação de Primo(s): 13 2 37 997 Fim da Relação.

Definição

Um número inteiro é primo se e somente se ele é maior que 1 e só é divisível por 1 e por ele mesmo.

3ª Questão (2,0 pontos)

Utilizando subprogramação, faça um programa que leia da entrada padrão, em uma única string, quatro números inteiros separados por espaços em branco. Os dois primeiros representam as duas dimensões de uma matriz bidimensional. Os dois últimos representam o valor mínimo e o valor máximo dos valores a serem gerados aleatoriamente para a matriz. Construa uma matriz bidimensional com as dimensões lidas e com valores gerados aleatoriamente, via função `randint` importada do módulo `random`, no intervalo mínimo e máximo lido. Suponha que todos os quatro números lidos sejam maiores ou iguais a um.

Escreva:

- (1) O conteúdo da matriz, onde cada linha a ser escrita possua apenas números inteiros, sem vírgulas nem colchetes;
- (2) O número de ponto flutuante, com uma casa de precisão, que representa a média de todos os números contidos na matriz;
- (3) Quais os números que se encontram nas bordas da matriz que possuam valor acima da média. A borda aqui é definida pelos valores que se encontram na primeira linha, na última linha, na primeira coluna e na última coluna da matriz, respectivamente.

Exemplo

Entrada	Saída
3 5 10 40	Conteúdo da Matriz: 33 10 23 37 38 10 11 30 11 23 32 16 28 19 21 Média dos Valores Sorteados: 22.8 Relação de Valores nas Bordas Acima da Média: 33 23 37 38 23 32 28

4ª Questão (2,0 pontos)

O máximo divisor comum, ou MDC, entre dois ou mais números inteiros é o maior número inteiro que é fator de tais números. Por exemplo, o MDC de 30 e 12 é 6, pois $30 = 2 \times 3 \times 5$ e $12 = 3 \times 2 \times 2$, nesse caso, note que tanto 30 quanto 12 possuem $6 = 3 \times 2$ em sua fatoração.

Uma observação interessante é que é possível encontrar o MDC de dois números A e B utilizando a seguinte relação recursiva para o caso em que A e B são maiores ou iguais a zero:

MDC de A e B, quando $B = 0$ ou quando $A = B$, é A

MDC de A e B, quando $A > B$, é o MDC de $A - B$ e B

MDC de A e B, quando $A < B$, é o MDC de A e $B - A$

Faça um programa que dada uma sequência de pares de valores inteiros A e B maiores ou iguais a zero escreva na saída padrão o MDC de cada par.

Seu programa deve conter uma função recursiva com o seguinte protótipo:

```
def mdc(a, b):
```

Entrada

A entrada é composta por várias linhas, cada uma contando um par de valores inteiros não negativos, A e B, separados por um espaço em branco. A última linha contém o par de valores "-1 -1", que não deve ser processado.

Saída

Uma linha deve ser emitida na saída padrão para cada par de valores dados como entrada. Essa linha deve conter o MDC do respectivo par de valores.

Exemplo

Entrada	Saída
30 12	6
5 68	1
5 5	5
25 5	5
9 31	1
14 80	2
54 24	6
-1 -1	

6ª Questão (2,0 pontos)

Faça um programa que peça ao usuário um valor inteiro N. Seu programa deverá gerar aleatoriamente uma lista contendo N valores inteiros sorteados no intervalo 0 e 10 (inclusive), em seguida gerar um valor inteiro K no mesmo intervalo e aplicar busca com sentinela para encontrar o valor K na lista. Seu programa deverá imprimir na saída padrão, conforme a formatação apresentada no exemplo:

- A lista contendo os N valores sorteados;
- O valor K sorteado;
- O índice da primeira ocorrência do valor K na lista, caso ele seja encontrado, ou a string "Valor não encontrado", caso contrário.

Exemplo

Entrada	Saída
15	Valores: 4 6 6 7 2 0 10 3 7 10 0 0 2 3 5 K: 3 Índice: 7