



Fundação CECIERJ - Vice-Presidência de Educação Superior a Distância

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

AD1 – 1º semestre de 2021

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 via atividade específica na Plataforma 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 (1,5 pontos)

No painel de um carro, está indicado no hodômetro a distância, em km , que o carro “rodou”. Faça um programa que escreva a ordem de grandeza do número de voltas efetuadas pela roda desse carro, sabendo o valor do diâmetro da roda, em cm :

Obs 1: Despreze possíveis derrapagens e frenagens do carro.

Obs 2: O comprimento da roda é igual a $2\pi R$, onde R é o valor do raio da roda do carro.

Obs 3: Utilize π importado do módulo *math* e adote π com apenas três casas decimais.

Obs 3: A ordem de grandeza de um número é a potência de 10 mais próxima deste número. Assim, dado X um número escrito na notação científica, $X = m \cdot 10^n$, onde $1 \leq m < 10$ e $n \in \mathbb{Z}$, a ordem de grandeza de X será: 10^n , se $m < 3,16$, ou 10^{n+1} , caso contrário.

Entrada: Dois inteiros. O primeiro indica a distância percorrida pelo carro, em km , e o segundo indica o diâmetro da roda do carro, em cm .

Saída: Ordem de grandeza do número de voltas efetuadas pela roda deste carro.

Exemplo

Entrada	Saída
120.000 50	Distância percorrida: 120.000 km Diâmetro da roda: 50 cm Ordem de grandeza da quantidade de voltas efetuadas: 10 elevado a 8

2ª Questão (1,5 pontos)

Considere um DNA por uma sequência do alfabeto A, C, G, T, associando as bases nitrogenadas Adenina, Citosina, Guanina ou Timina, respectivamente. Considere também um *motif* por um intervalo de nucleotídeos (ou de aminoácidos, em proteínas) que possui alguma importância biológica. Queremos obter todas as substrings “mais próximas” do motif. Neste problema, “mais próxima” é definida pela distância de Hamming de duas substrings.

Definição: Distância de Hamming entre duas strings de mesmo tamanho é o número de posições nas quais os caracteres das strings se diferem entre si. Exemplo: AGCT e CGCA possuem distância de Hamming igual a 2.

Entrada: Um inteiro não negativo k , onde $k \leq 50$, uma string s de motif, com no máximo 50 caracteres, e uma string t de DNA, com no máximo 500 caracteres.

Saída: Todas substrings t' de t tal que a distância de Hamming entre t' e s seja no máximo k . Cada substring deve ser codificada na sua saída pela posição inicial em t seguida pelas posições onde t' e s diferem. Se as entradas estiverem fora dos valores delimitados, então escreva "Valores não estão de acordo".

Exemplo

Entrada	Saída
3 ACGTAG ACGGATCGGCATCGT	As substrings com distância no máximo 3 do mofit e as posições são: 1 4 6 9 1 3 5

3ª Questão (2,0 pontos)

Sr Cederjeano notou que estava engordando muito durante a pandemia e decidiu entrar para a academia. Assim, Sr Cederjeano precisa comprar equipamentos para sua malhação e vai, primeiramente, em algumas lojas pesquisar os preços desses equipamentos.

Faça um programa que receba como entrada um inteiro L , um inteiro P , ambos não nulos, L nomes associando às lojas pesquisadas, P nomes associando aos produtos pesquisados nas lojas, e para cada produto, receba dois inteiros não negativos a e b , onde $a \leq b$, que delimitarão os valores mínimo e máximo de cada produto nas lojas.

Construa uma matriz bidimensional $L \times P$ com as dimensões lidas e com valores gerados aleatoriamente, com duas casas decimais, importada do módulo *random*.

Escreva:

- (1) A matriz com os valores, incluindo, também, os nomes das lojas e produtos nos índices das linhas e colunas, respectivamente;
- (2) Exiba por linha o par produto e loja com menores preço;
- (3) Exiba o total a ser gasto ao considerar os menores preços.

Exemplo

Entrada	Saída
2 3 Loja1 Loja2 Prod1 10 20 Prod2 30 60 Prod3 15 30	Resultado da pesquisa: Prod1 Prod2 Prod3 Loja1 11.50 45.20 27.30 Loja2 14.80 50.05 22.57 Menores preços: Prod1 Loja1 Prod2 Loja1 Prod3 Loja2 Valor total: R\$ 79.27

4ª Questão (2,0 pontos)

Faça um programa, contendo subprogramas, que leia linhas da entrada padrão até que uma linha vazia seja digitada. Com exceção da linha vazia, todas as anteriores contém um par de números inteiros, representando a coordenada x e y de um ponto no espaço bidimensional.

Calcule e escreva o centro geométrico, ou ponto centróide, de todos os pontos lidos. A coordenada x do centróide é dada pela média de todos os x's lidos. Assim como a sua coordenada y é dada pela média de todos os y's lidos.

Em seguida, escreva na saída padrão o ponto mais próximo e o ponto mais distante do centróide.

Caso nenhum ponto seja digitado, escreva a mensagem: "Nenhum ponto lido. Portanto, não há centróide!!!".

Definição: a **distância** entre dois pontos (x_A, y_A) e (x_B, y_B) é dada pela raiz quadrada da soma do quadrado das diferenças $(x_B - x_A)$ e $(y_B - y_A)$.

Exemplo

Entrada	Saída
2 3 66 25 4 2 8 70	Centróide: (20.0, 25.0) Ponto mais próximo do Centróide: (4, 2) Ponto mais distante do Centróide: (8, 70)

5ª Questão (2,0 pontos)

Faça um programa, contendo subprogramas, que leia inicialmente a contagem de candidatos em uma eleição. Em seguida, em cada linha, leia o nome de um dos candidatos e seu respectivo número positivo, separados por um #. Utilize uma estrutura de dados para manter estas informações. Em seguida, leia números representando votos da eleição. Um voto por linha lida, até que um número negativo seja digitado. Totalize e escreva na saída padrão os votos de cada candidato, suponha que o número zero represente um voto em branco e um número positivo que não pertença a nenhum dos candidatos seja um voto nulo. Ao final, também escreva quantos foram os votos em branco e quantos foram os votos nulos.

Exemplo

Entrada	Saída
3 Juca da Venda#171 Pastor das Ovelhas#1234 Doido da Esquina#22	Juca da Venda - 171 - com 2 voto(s) Pastor das Ovelhas - 1234 - com 2 voto(s) Doido da Esquina - com 3 voto(s)

1234	Branços - com 1 voto(s) Nulos - com 2 voto(s)
22	
0	
666	
171	
171	
22	
22	
1234	
55	
-1	

6ª Questão (1,0 pontos)

Faça um programa, contendo subprograma, que leia strings da entrada padrão até que uma string vazia seja digitada. Faça e utilize uma **função recursiva** que analise se a string lida é uma palíndrome. Toda string palíndrome deve ser escrita na saída padrão.

Definição: uma string é **palíndrome** se e somente se o primeiro caractere é igual ao último, o segundo caractere é igual ao penúltimo, e assim sucessivamente.

Exemplo

Entrada	Saída
covid	banana ananab
banana ananab	ala
ala	12344321
vacina	
12344321	