

Aula Prática 9

Adriano César Machado Pereira
João Guilherme Maia de Menezes

Instruções para Submissão

Na aula prática de hoje, você terá que elaborar programas para resolver problemas diversos, conforme descrito abaixo. Cada uma das soluções deverá ser implementada em seu próprio arquivo com extensão `.py`. Por exemplo, a solução para o problema 1 deverá ser implementada em um arquivo chamado `problema1.py`, a solução para o problema 2 deverá ser implementada no arquivo `problema2.py` e assim por diante. Finalmente, submeta cada um dos arquivos pelo Moodle.

Dica: se você tiver problemas com caracteres especiais (caracteres com acentos, por exemplo), adicione a linha abaixo na primeira linha de todos os arquivos `*.py`

```
# -*- coding: utf-8 -*-
```

Problema 1

Escreva um programa que lê uma palavra e imprima na tela a letra com o maior número de ocorrências na palavra digitada.

Observação: as mensagens exibidas para o usuário deverão ser exatamente como apresentado abaixo (mensagens exibidas com os comandos `input()` e `print()`).

Exemplo 1 de execução do programa:

```
banana  
a
```

Exemplo 2 de execução do programa:

```
assassino  
s
```

Problema 2

Escreva um programa que lê uma palavra e imprima na tela a vogal com o maior número de ocorrências na palavra digitada.

Observação: as mensagens exibidas para o usuário deverão ser exatamente como apresentado abaixo (mensagens exibidas com os comandos `input()` e `print()`).

Exemplo 1 de execução do programa:

```
banana  
a
```

Exemplo 2 de execução do programa:

```
assassino  
a
```

Problema 3

Escreva um programa para gerenciar as notas de uma turma de alunos. Para isso, a entrada de dados deve ser da seguinte forma: o programa deve ler o nome de um aluno, duas notas e em seguida passar para o próximo aluno. Esse processo de entrada de dados termina quando uma string vazia for digitada para o nome de um aluno. Em seguida, o programa deve exibir o nome e a média de todos os alunos, ordenada da maior para a menor média, conforme exemplo de execução abaixo.

Observação: as mensagens exibidas para o usuário deverão ser exatamente como apresentado abaixo (mensagens exibidas com os comandos `input()` e `print()`).

Exemplo de execução do programa:

```
Tunico  
3.5  
5  
Tinoco  
10  
8.5  
  
Tinoco - 9.25  
Tunico - 4.25
```

Problema 4

Escreva um programa que lê uma sequência de números inteiros até que o número -1 for digitado. Em seguida, o programa deve exibir na tela o número que mais apareceu na sequência de números digitados, conforme exemplo de execução abaixo

Observação: as mensagens exibidas para o usuário deverão ser exatamente como apresentado abaixo (mensagens exibidas com os comandos `input()` e `print()`).

Exemplo de execução do programa:

```
1
2
3
2
5
1
2
-1
2
```

Problema 5

Sabe-se que uma molécula de RNA mensageiro é utilizada como base para sintetizar proteínas, no processo denominado de tradução. Cada trinca de bases de RNA mensageiro está relacionado com um aminoácido. Combinando vários aminoácidos, temos uma proteína. Com base na tabela (simplificada) de trincas de RNA abaixo, escreva um programa que receba uma string representando uma molécula de RNA mensageiro válida, segundo essa tabela, e imprime na tela a cadeia de aminoácidos que representam a proteína correspondente:

Trinca de RNA	Nome do Aminoácido
UUU	Phe
CUU	Leu
UUA	Leu
AAG	Lisina
UCU	Ser
UAU	Tyr
CAA	Gln

Observação: as mensagens exibidas para o usuário deverão ser exatamente como apresentado abaixo (mensagens exibidas com os comandos `input()` e `print()`).

Exemplo 1 de execução do programa:

Digite o RNA: UUUUUAUCU

Cadeia de Aminoácidos: Phe-Leu-Ser

Exemplo 2 de execução do programa:

Digite o RNA: UUUCAA

Cadeia de Aminoácidos: Phe-Gln

Problema 6

Escreva um programa que converte um número inteiro (entre 1 e 999) para algarismos romanos. Use os três dicionários abaixo:

- UNIDADES = 0: "", 1: 'I', 2: 'II', 3: 'III', 4: 'IV', 5: 'V', 6: 'VI', 7: 'VII', 8: 'VIII', 9: 'IX'
- DEZENAS = 0: "", 1: 'X', 2: 'XX', 3: 'XXX', 4: 'XL', 5: 'L', 6: 'LX', 7: 'LXX', 8: 'LXXX', 9: 'XC'
- CENTENAS = 0: "", 1: 'C', 2: 'CC', 3: 'CCC', 4: 'CD', 5: 'D', 6: 'DC', 7: 'DCC', 8: 'DCCC', 9: 'CM'

Observação: as mensagens exibidas para o usuário deverão ser exatamente como apresentado abaixo (mensagens exibidas com os comandos `input()` e `print()`).

Exemplo 1 de execução do programa:

Digite um número: 819

Número Romano: DCCCXIX

Exemplo 2 de execução do programa:

Digite um número: 51

Número Romano: LI

Problema 7

Em uma corrida de Kart, cada competidor tem direito a dar 5 voltas. O arquivo `tempos.txt` armazena o nome e os tempos em segundos de todos os competidores durante a corrida. Escreva um programa que lê o arquivo `tempos.txt` e exibe na tela o nome do competidor que realizou a melhor volta e o tempo da melhor volta, além da classificação final dos competidores na corrida (começando do campeão). O campeão é o competidor com a menor média de tempos.

Observação: as mensagens exibidas para o usuário deverão ser exatamente como apresentado abaixo (mensagens exibidas com os comandos `input()` e `print()`).

Exemplo de execução do programa:

Assumindo que o arquivo `tempos.txt` possui o conteúdo abaixo:

```
Burnquist 151 50 130 140 85
Marley 112 51 55 60 137
Bob 98 145 95 89 87
Ariel 112 53 80 98 69
```

Seu programa deve imprimir:

Melhor volta: Burnquist - 50 segundos

Classificação final:

1 - Ariel - 82.40

2 - Marley - 83.00

3 - Bob - 102.80

4 - Burnquist - 111.20