

Enunciados dos Trabalhos para Casa (TPC)

da UC de **Processamento de Linguagens**, ano Letivo 2022/23 , 3º ano de LEI

Relativos à 1º parte do programa: Filtros de Texto baseados em ER **texto em negrito**

TPC1: Análise de dados: doença cardíaca

(publicado em **2023.02.14**)

Descarregue o ficheiro de dados: `myheart.csv` Crie um programa em Python, conjunto de funções, que responda às seguintes questões:

- Crie uma função que lê a informação do ficheiro para um modelo, previamente pensado em memória;
- Pense num modelo para guardar uma distribuição;
- Crie uma função que calcula a distribuição da doença por sexo;
- Crie uma função que calcula a distribuição da doença por escalões etários. Considere os seguintes escalões: [30-34], [35-39], [40-44], ...
- Crie uma função que calcula a distribuição da doença por níveis de colesterol. Considere um nível igual a um intervalo de 10 unidades, comece no limite inferior e crie os níveis necessários até abranger o limite superior;
- Crie uma função que imprime na forma de uma tabela uma distribuição;
- Especifique um programa que ao executar apresenta as tabelas correspondentes às distribuições pedidas;
- Extra: explore o módulo `matplotlib` e crie gráficos para as suas distribuições.

TPC2: Somador on/off

(publicado em **2023.02.21**)

Crie um programa em Python que tenha o seguinte comportamento:

- Pretende-se um programa que some todas as sequências de dígitos que encontre num texto;
- Prepare o programa para ler o texto do canal de entrada: `stdin`;
- Sempre que encontrar a string "Off" em qualquer combinação de maiúsculas e minúsculas, esse comportamento é desligado;
- Sempre que encontrar a string "On" em qualquer combinação de maiúsculas e minúsculas, esse comportamento é novamente ligado;
- Sempre que encontrar o carácter "=", o resultado da soma é colocado na saída.

TPC3: Processador de Pessoas listadas nos Róis de Confessados

(publicado em **2023.02.28**)

Construa agora um ou vários programas Python para processar o texto 'processos.txt' (procurar o ficheiro no Bb) com o intuito de calcular frequências de alguns elementos (a ideia é utilizar arrays associativos, dicionários em Python, para o efeito) conforme solicitado a seguir:

- a) Calcula a frequência de processos por ano (primeiro elemento da data);
- b) Calcula a frequência de nomes próprios (o primeiro em cada nome) e apelidos (o ultimo em cada nome) por séculos e apresenta os 5 mais usados;
- c) Calcula a frequência dos vários tipos de relação: irmão, sobrinho, etc.;
- d) Converta os 20 primeiros registos num novo ficheiro de output mas em formato **Json**.