Ficha Prática nº 1

Objetivo: *Pretende-se adquirir competências de programação em* MATLAB | Python.

Linguagem de Programação: MATLAB | Python.

Exercícios:

1. Considere a matriz A:

$$A = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \end{bmatrix}$$

1.1. Defina a matriz A no *Workspace* do MATLAB (ou em Python).

1.2. Defina uma matriz B com as mesmas dimensões de A, com valores inteiros aleatórios entre 2 e 9, seguindo uma distribuição uniforme.

1.3. Guarde as duas matrizes num ficheiro *.mat* de nome *abfile.mat* (apenas em MATLAB).

1.4. Limpe o *Workspace* do MATLAB (apenas em MATLAB).

1.5. Carregue as matrizes armazenadas no ficheiro *abfile.mat* (apenas em MATLAB).

1.6. Elimine a segunda coluna de A e a terceira coluna de B.

1.7. Concatene a matriz coluna $C_A = \begin{bmatrix} 10 \\ 30 \end{bmatrix}$ no início de A e a matriz coluna $C_B = \begin{bmatrix} 20 \\ 50 \end{bmatrix}$ no final da matriz B.

1.8. Obtenha a matriz Cp que é formada pela primeira linha de A e pela última linha de B.

1.9. Crie uma matriz C a partir da:

1.9.1. Soma de A e B;

1.9.2. Subtração de B a A;

1.9.3. Multiplicação de A pela transposta de B;

1.9.4. Multiplicação elemento a elemento de A e B;

1.9.5. Divisão à direita de A e B;

1.9.6. Divisão à direita elemento a elemento de A e B.

2. Considere a seguinte função dependente do tempo $f(t) = sin(2\pi t) + sin(\pi t)$.

2.1. Crie um vetor tempo t que permita obter valores para $f(t)$ entre -10 e 10 segundos. Considere uma diferença de 0.01s entre valores do vetor tempo. Indique a dimensão do vetor tempo.

2.2. Represente graficamente a evolução de $f(t)$ em função do tempo.

3. Realize o exercício 2, mas recorrendo a cálculo matemático simbólico.

4. Crie um *script* para representar graficamente a função $f(x,y) = sin(xy) + cos(x)$ recorrendo a cálculo numérico e a cálculo simbólico. Considere que x e y assumem valores entre -4 e 4 com um passo adequado à visualização gráfica da função.

5. Crie um *script* para encontrar os coeficientes de um polinómio de grau 2 que se ajusta a uma dada série temporal, pelo método dos mínimos quadrados, e para representar graficamente, no mesmo gráfico, os dados originais e o resultado do ajuste. Considere que a série temporal é definida para um intervalo de tempo de 0 a 10 segundos (passo de 1s) com os seguintes valores:
   $y = [0 \;\; 0.7 \;\; 2.4 \;\; 3.1 \;\; 4.2 \;\; 4.8 \;\; 5.7 \;\; 5.9 \;\; 6.2 \;\; 6.4 \;\; 6.3]$.

6. Desenvolva uma função não recursiva que calcule o fatorial de um número.

7. Desenvolva uma função não recursiva que calcule os primeiros N números da série de *Fibonacci*.

8. O modelo de *Nicholson–Bailey* foi criado para simular a coevolução de duas populações, uma de hospedeiros e outra de parasitas. O modelo é baseado em equações de diferença que descrevem o crescimento de ambas as populações. Mais concretamente, o modelo assume que os parasitas procuram hospedeiros de forma aleatória e que ambos (parasitas e hospedeiros) se encontram distribuídos de forma não contígua no espaço. Matematicamente o modelo é descrito por:

$$H_{g+1} = kH_g e^{-aP_g}$$
$$P_{g+1} = cH_g\left(1 - e^{-aP_g}\right)$$

   Nas equações anteriores *Hg* e *Pg* representam o tamanho da população de hospedeiros e da população de parasitas na geração *g*, respetivamente, sendo *g* definido de 0 a *N*. *k* representa a taxa de reprodução dos hospedeiros, *c* representa o número médio de ovos viáveis depositados pelos parasitas num único hospedeiro, e *a* representa a probabilidade de um determinado parasita encontrar um hospedeiro durante a sua vida.
   8.1. Desenvolva uma função que permita simular a evolução de ambas as populações.
   8.2. Desenvolva também um *script* que chame a função desenvolvida e represente graficamente a evolução de ambas as populações, considerando, por exemplo, os seguintes casos:
      8.2.1. a=0.025; k=1.0; c=2; N=30; H_0=20; P_0=10.
      8.2.2. a=0.025; k=1.5; c=2; N=30; H_0=20; P_0=10.
      8.2.3. a=0.025; k=1.5; c=2; N=30; H_0=15; P_0=10.