

ATIVIDADE 2 – INTRODUÇÃO A PYTHON: Parte 2

Obs: Todas as funções devem ser desenvolvidas em Python 3 e devem ser devidamente documentadas (docstring) e testadas. Para testá-las, crie um script de testes com o número de exemplos de testes por função que julgar adequado. O arquivo de solução deve ser devolvido até a data de entrega da atividade. Salve as funções em uma pasta, pois elas poderão ser úteis durante o curso de Processamento Digital de Sinais.

- 1) **Geração de vetores e/ou matrizes a partir de métodos prontos do numpy:** Consulte a documentação do numpy para implementar o que se pede.

Função para gerar:

- a) Matriz identidade de ordem N;
- b) Matriz ou vetor de zeros de ordem N;
- c) Matriz ou vetor de uns de ordem N;
- d) Vetor com valores regularmente espaçados (linspace);
- e) Vetor com os elementos da diagonal principal de uma matriz;
- f) Matriz diagonal a partir de elementos de um vetor.

- 2) **Operação matemática com vetores e/ou matrizes do numpy:** Consulte a documentação do numpy para implementar o que se pede:

Função para:

- a) Somar dois vetores ou matrizes (usar mecanismo de exceção caso dimensões ou tipos sejam incompatíveis);
- b) Subtrair dois vetores ou matrizes (usar mecanismo de exceção caso dimensões ou tipos sejam incompatíveis);
- c) Calcular o produto escalar de um vetor ou matriz (usar mecanismo de exceção caso dimensões ou tipos sejam incompatíveis);
- d) Calcular a exponencial de um vetor ou matriz (usar mecanismo de exceção caso dimensões ou tipos sejam incompatíveis).

- 3) **Geração e visualização de sinais senoidais (cossenoidais):**

- a) Gere o sinal $x_1 = 2 \cdot \cos(2 \cdot \pi \cdot f \cdot t)$ usando a biblioteca math (π e \cos) e numpy ($t = \text{linspace} \dots$). Faça a frequência f igual a 15Hz. Plote a função usando a biblioteca matplotlib (comando plot). Ajuste o incremento de t pequeno o suficiente para que a função pareça suave na plotagem. Especifique um intervalo de tempo t grande o bastante para mostrar 2 períodos do sinal x_1 ;
- b) Use os métodos do numpy para determinar o valor máximo e mínimo da função;
- c) Use as funções **Maximum** e **Minimum** que você implementou para obter os máximos e mínimos, bem como as posições onde eles ocorrem. Faça as adaptações necessárias naquelas funções para que funcionem com o tipo array do numpy.