



**UNIVERSIDADE FEDERAL DO CEARÁ – UFC SOBRAL**  
**TÉCNICAS DE PROGRAMAÇÃO – PROF. FISCHER**

**Lista 3 – Funções**

Faça uma função para cada um dos itens abaixo. Você deverá testar cada função com dados que exercitem a lógica de cada uma das funções abaixo.

- 1) **Somatório:** Sua função deverá receber um vetor como parâmetro e retornar a soma de todos elementos do vetor, sendo que as primeiras duas e as últimas posições do vetor de entrada não devem ser consideradas;
- 2) **Média:** Sua função deverá receber dois vetores como entrada e retornar a média dos valores presentes nos vetores.
- 3) **Média ponderada:** Sua função deverá receber um vetor de notas e um vetor de pesos correspondentes. A função deve retornar a média ponderada das notas. Exemplo: [7, 8, 6], [2, 3, 1] -> 7.33
- 4) **Contagem de elementos:** Sua função deverá receber dois vetores e um elemento a ser contado. A função deve retornar a quantidade de vezes que esse elemento aparece nos vetores. Exemplo: [1,3, 3, 2, 5],[1, 2, 3, 2, 4], 2 -> 3
- 5) **Cópia de vetor:** Sua função deverá receber um vetor e retornar uma cópia exata desse vetor.
- 6) **Ordenação crescente:** Sua função deverá receber um vetor e retornar o vetor ordenado em ordem crescente. Exemplo: [5, 2, 9, 1, 7] -> [1, 2, 5, 7, 9]
- 7) **Verificação de ordenação:** Sua função deverá receber um vetor e verificar se ele está ordenado em ordem crescente. Se sim retornar true.
- 8) **K-ésimo maior valor:** Sua função deverá receber um vetor e um K e retornar o K-ésimo maior valor presente no vetor.
- 9) **K-ésimo menor valor:** Sua função deverá receber um vetor e um K e retornar o K-ésimo menor valor presente no vetor.
- 10) **Verificação de igualdade:** Sua função deverá receber dois vetores e verificar se eles são iguais, ou seja, se possuem os mesmos elementos na mesma ordem.

- 11) **Potenciação:** Sua função deverá receber dois números, a base e o expoente, e retornar a potência da base elevada ao expoente.
- 12) **Primo:** Sua função deverá receber um número inteiro e verificar se ele é primo, ou seja, se é divisível apenas por 1 e por ele mesmo.
- 13) **Busca por ocorrências:** Sua função deverá receber dois vetores e deverá retornar um novo vetor com todos os elementos que aparecem em ambos os vetores.
- 14) **Remoção de elemento do vetor:** Sua função deverá receber um vetor e um elemento a ser removido. A função deve remover todas as ocorrências desse elemento no vetor. Um novo vetor, apenas com elementos mantidos, deve ser retornado: Exemplo: [1, 2, 3, 2, 4], 2 -> [1, 3, 4]
- 15) **Verificação de palíndromo:** Sua função deverá receber uma string e verificar se ela é um palíndromo, ou seja, se pode ser lida da mesma forma de trás para frente. Exemplo: "ana" -> True.
- 16) **Verificação de número primo:** Sua função deverá receber um número inteiro e verificar se ele é um número primo. Exemplo: 7 -> True.
- 17) **Inversão de string:** Sua função deverá receber uma string e retornar a mesma string invertida. Exemplo: "hello" -> "olleh".
- 18) **Cálculo de juros compostos:** Sua função deverá receber um valor inicial, uma taxa de juros e um número de períodos. A função deve retornar o valor final após a aplicação dos juros compostos. Exemplo: 1000, 0.05, 3 -> 1157.63
- 19) **Verificação de número perfeito:** Sua função deverá receber um número inteiro e verificar se ele é um número perfeito, ou seja, se a soma dos seus divisores próprios é igual a ele mesmo. Exemplo: 6 -> True
- 20) **Diminuição do vetor:** sua função deverá receber um número qualquer que represente e criar um vetor com o tamanho informado. Este vetor deverá ter sucessivas diminuições do seu tamanho em um unidade até que seu tamanho seja igual a 1.