

## Algoritmos e Estruturas de Dados I

### Lista 7: arranjos

Nome: \_\_\_\_\_

Matrícula: \_\_\_\_\_

1. Escreva um programa em C++ que lê uma sequência de números de um **arquivo** e armazena em um arranjo. Assuma que a quantidade de números no arquivo não ultrapassa o limite máximo de elementos do arranjo (e.g. 1000).

t

2. Escreva uma função em C++ que recebe um arranjo de números reais **v** e número de elementos **n** armazenados em **v** e que retorna a média dos **n** elementos armazenados em **v**. Assuma que **n** é menor que o número máximo de elementos do arranjo (e.g. 1000).

3. Escreva uma função em C++ que recebe um arranjo de números reais **v** e número de elementos **n** armazenados em **v** e que retorna a variância dos **n** elementos armazenados em **v**. Assuma que **n** é menor que o número máximo de elementos do arranjo (e.g. 1000).

4. Escreva uma função em C++ que recebe um arranjo de números reais **v** e número de elementos **n** armazenados em **v** e que retorna o **maior** dos **n** elementos armazenados em **v**. Assuma que **n** é menor que o número máximo de elementos do arranjo (e.g. 1000).

5. Escreva uma função em C++ que recebe um arranjo de números reais **v** e número de elementos **n** armazenados em **v** e que retorna o **menor** dos **n** elementos armazenados em **v**. Assuma que **n** é menor que o número máximo de elementos do arranjo (e.g. 1000).

6. Escreva uma função em C++ que recebe dois arranjos de números reais **u** e **v** e a dimensão **n** dos dois arranjos e que retorna o produto escalar de **u** e **v**. O produto escalar de dois arranjos é dado pela seguinte expressão:  $u \cdot v = u_0 \cdot y_0 + u_1 \cdot y_1 + u_2 \cdot y_2 + \dots + u_{n-1} \cdot y_{n-1}$ . Assuma que **n** é menor que o número máximo de elementos do arranjo (e.g. 1000).