

22. Exercícios

DIM0321

2015.1

1. (*Soma dos elementos de dois arranjos*) Escrever uma sub-rotina `soma_arranjo` que:

- recebe como parâmetros três variáveis arranjos `u`, `v` e `w` de tipo de base `float`, e de mesmo tamanho, e uma variável `tamanho` de tipo `int` referente ao tamanho desses arranjos;
- soma um a um cada um dos elementos dos arranjos `u` e `v` e armazena os resultados nos elementos respectivos do arranjo `w`

A interface a ser utilizada é a seguinte:

```
1| void soma_arranjo(float *u, float *v, float *w, int tamanho);
```

Exemplo

u	v	tamanho	w
{1, 2, 3}	{2, 3, 4}	3	{3, 5, 7}

2. (*Busca em um arranjo*). Escrever uma sub-rotina `busca_arranjo` que:

- recebe como parâmetros um arranjo de 1000 números inteiros e um valor do tipo inteiro a ser buscado no arranjo
- retorna 1 (ou true) se o valor buscado for um elemento do arranjo, 0 (ou false) caso contrário

A interface a ser utilizada é a seguinte:

```
1| int busca_arranjo(int vetor[1000], int i);
```

3. (*Posição em um arranjo*). Escrever uma sub-rotina `posicao_arranjo` que:

- recebe como parâmetros um arranjo de 1000 números inteiros e um valor do tipo inteiro a ser buscado no arranjo
- retorna a menor posição (ou índice) do arranjo que contém o valor buscado, ou -1 caso o valor não esteja presente.

A interface a ser utilizada é a seguinte:

```
1| int posicao_arranjo(int vetor[1000], int i);
```

4. (*Contagem de ocorrências*). Escrever uma sub-rotina `numero_arranjo` que:

- recebe como parâmetros um arranjo de 1000 números inteiros e um valor do tipo inteiro a ser buscado no arranjo

- retorna o número de elementos do arranjo que contém o valor buscado

A interface a ser utilizada é a seguinte:

```
1| int numero_arranjo(int vetor[1000], int i);
```