

## LISTA DE EXERCÍCIOS 4

Resolva os exercícios a seguir utilizando a Linguagem C.

1. Crie uma função que receba como parâmetros dois vetores de inteiros: **v1** e **v2** e as suas respectivas quantidades de elementos: **n1** e **n2**.

A função deverá retornar um ponteiro para um terceiro vetor, **v3**, alocado dinamicamente, contendo a união de **v1** e **v2** e usar o ponteiro **qtd** para devolver o tamanho de **v3**.

Utilize a seguinte assinatura de função:

```
int* uniao(int *v1, int *v2, int n1, int n2, int* qtd)
```

2. Crie uma função que receba como parâmetros dois vetores de inteiros: **v1** e **v2** e as suas respectivas quantidades de elementos: **n1** e **n2**.

A função deverá retornar um ponteiro para um terceiro vetor, **v3**, alocado dinamicamente, contendo a interseção de **v1** e **v2** e usar o ponteiro **qtd** para devolver o tamanho de **v3**.

Utilize a seguinte assinatura de função:

```
int* intersecao(int *v1, int *v2, int n1, int n2, int* qtd)
```

3. Faça um programa que leia a quantidade de elementos **n**, aloque dinamicamente um vetor de **n** elementos inteiros e passe esse vetor para uma função que irá ler todos os elementos desse vetor. Depois, no programa principal, o vetor preenchido deve ser impresso.
4. Desenvolva uma função que receba um vetor original e retorne um novo vetor (alocado dinamicamente) somente com os números pares. Um parâmetro adicional é passado por referência, para ser atualizado com a quantidade de elementos pares. O protótipo da função será:

```
int* obtemPares(int *vetor, int tamanho, int *tamNovo)
```

Crie uma função principal para ler o vetor original e faça uso da sua função para imprimir um vetor apenas com os números pares.

5. Escreva uma função `conta_ocorrencias`, que recebe um vetor de inteiros, um `n` representando seu tamanho e um valor inteiro e retorna uma contagem de quantas vezes o valor especificado aparece no vetor. Cabeçalho da função:

```
int conta_ocorrencias(int *v, int n, int k)
```

Escreva uma função `remove_ocorrencias`, que recebe um vetor de inteiros, um inteiro `n` representando seu tamanho, um inteiro `k` e um inteiro `n2`, que representa o novo tamanho do vetor. Esta função deve remover todas as ocorrências de `k` no vetor e retornar um novo vetor sem as ocorrências de `k`. A função deve alocar dinamicamente um novo vetor `v2`, cujo tamanho é calculado, e retornar o endereço de memória onde o vetor foi alocado. Cabeçalho da função:

```
int* remove_ocorrencias(int *v, int *n, int k)
```

Na função `main`, leia o vetor de `n` elementos da entrada, alocando-o dinamicamente usando `malloc`, e atualize o endereço do vetor com o endereço resultante da função `remove_ocorrencias`. Ao final, imprima os elementos restantes do vetor.