

Lista de Exercícios - Estruturas de Dados (Ponteiros e Funções)

1. Escreva uma função em C que receba um vetor de inteiros preenchido aleatoriamente e seu tamanho n , e gere um novo vetor contendo apenas os valores únicos do vetor original, preservando a primeira ocorrência de cada valor.

A função deve ter a seguinte assinatura:

```
void remover_duplicatas(int *vetor, int n, int *compactado, int
*tam_compactado);
```

2. Considere a seguinte função em C:

```
void troca(int *a, int *b) {
    int temp = *a;
    *a = *b;
    *b = temp;
}
```

(a) Explique o que a função faz.

(b) Como ela deve ser chamada no `main` para trocar os valores de duas variáveis x e y ?

3. Dado um vetor v de n elementos inteiros, escreva uma função que receba um ponteiro para o vetor e um inteiro n , e retorne o maior valor encontrado.

```
int maior_elemento(int *v, int n);
```

4. Implemente uma função que receba uma matriz quadrada de ordem n e retorne a soma dos elementos da diagonal principal. Utilize ponteiros para percorrer a matriz.

```
int soma_diagonal(int **matriz, int n);
```

5. Escreva uma função que receba dois inteiros a e b e retorne um ponteiro para o maior valor.

```
int* maior(int *a, int *b);
```

6. Implemente uma função que recebe um vetor de inteiros e seu tamanho e retorna a média dos elementos.

```
float calcular_media(int *v, int n);
```

7. Escreva uma função em C que receba um vetor de inteiros preenchido aleatoriamente e seu tamanho n , e gere um novo vetor que contenha a contagem de ocorrências de cada valor presente no vetor original.

A função deve ter a seguinte assinatura:

```
void contar_ocorrencias(int *vetor, int n, int *ocorrencias, int  
*tam_ocorrencias);
```
