



Lista de Exercícios 1 – Ponteiros

1. Defina o que é um ponteiro.
2. Defina a diferença do uso de * na declaração de um ponteiro e na atribuição de valor em um ponteiro.
3. Explique por que o seguinte trecho de código tem saída “Erro” e altere apenas o método |”quadrado” para que saída seja “Sucesso”.

```
void quadrado(int *x) {  
    x = (*x) * (*x);  
}  
  
int main() {  
    int x = 5;  
    quadrado(&x);  
    if (x == 25) {  
        printf("Sucesso");  
    } else {  
        printf("Erro");  
    }  
}
```

4. Dada a variável a seguir, construa um laço de repetição que imprima um número por vez utilizando aritmética de ponteiros.

```
int vetor[10] = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
```

5. Escreva uma função que troque o valor de duas variáveis utilizando ponteiros.

Considere a seguinte struct para os exercícios seguintes:

```
typedef struct TCoordenadas {  
    float x;  
    float y;  
} TCoordenadas;
```

6. Crie um método que retorne um ponteiro para a struct e imprima os valores de x e y.
7. Crie um método que receba um vetor de coordenadas. Utilizando aritmética de ponteiros imprima o maior x e sua posição no vetor e o menor y e sua posição no vetor.
8. Explique por que o ponteiro retornado **não** pode ser utilizado.

```
TCoordenadas *mapa(float x, float y) {  
    TCoordenadas temp = {x, y};  
    return &temp;  
}
```