

## Laboratório 12

### Ponteiros (parte 2)

#### Introdução à Programação C (CMT012)

Prof. Ronald Souza

IC/UFRJ — 08/11/2023

#### Objetivo

Praticar manipulações básicas de variáveis do tipo ponteiro.

**Todos os seus programas devem preencher a estrutura abaixo.**

```
/*
Autor: <nome do aluno>
Data: <data de hoje>
Descrição: <o que o programa faz>
Entrada: <o que o programa espera receber como entrada>
Saída: <o que o programa retorna para o usuário>
Defesa: <restrições sobre os dados de entrada, se existirem>
*/

#include <stdio.h>

int main() {
    //dicionário de dados

    //corpo do programa

    return 0;
}
```

**Atividade 1:** Copie o código abaixo e **NÃO** o execute. Ele possui um objetivo, mas **contém erros**.

Proceda da seguinte forma:

- Analise o código e **procure entender** o seu propósito;
- Tente detectar seus erros já nessa etapa de avaliação;
- Compile o código;
- Se ainda há erros informados pelo compilador, corrija-os e **execute o código**.

```
#include<stdio.h>
#include<stdlib.h>
#include<math.h>

typedef struct {
    double x, y;
} t_Ponto;

double comp(t_Ponto*, t_Ponto*); //Assinatura

int main() { //Lógica principal
    t_Ponto *p1, p2;
    puts("Coordenadas do ponto 1:");
    scanf("%lf %lf", &(p1->x), &(p1->y));
    puts("Coordenadas do ponto 2:");
    scanf("%lf %lf", &p2.x, &p2.y);
    printf("\nComprimento da reta= %.1f\n", comp(p1, &p2));
    return 0;
}

double comp(t_Ponto* p1, t_Ponto* p2) { //Implementação
    return sqrt(pow(p2.x - *(p1->x), 2) + pow(p2.y - p1.y, 2));
}
```

**Atividade 2:** Nessa tarefa você irá **refazer a Questão 5 da lista 3** conforme o enunciado abaixo:

Crie um **tipo de dados** para representar uma **Pessoa**, mas agora, além dos campos **idade** (inteiro) e **peso** (float), o campo **nome** (string) deverá ser **um ponteiro de caractere**.

Implemente a função **main**, onde um vetor de Pessoas com o total de posições que o usuário informar (máximo de 50) **será alocado dinamicamente via ponteiro**. O usuário deverá então preencher o vetor.

Crie uma **função** para ordenar esse vetor em ordem **decrescente** de **idade** e, de volta na **main()**, imprima na tela o vetor **já ordenado**.

Agora, após a ordenação, permita que o usuário insira uma idade na **main()**. Crie uma **função** que receba de entrada o vetor já ordenado e a idade informada, e então realize uma **busca binária** no vetor de Pessoas, retornando o índice no vetor onde a pessoa possui a referida idade, ou -1 se ninguém for encontrado. No caso de duas ou mais pessoas com idades iguais, basta retornar um único índice. **Na main, imprima o nome dessa pessoa**.

**CUIDADO:** o vetor está em ordem **decrescente**. Sua busca binária leva isso em conta!