

# {introcomp}

## Working 08 : Tipos Abstratos de Dados

### Objetivos:

- Compreender a definição e dominar a implementação estruturas e TAD's em C.

**Prazo de Envio:** sabado, 03/11, 04:00.

# 1 STRUCTS

Structs(abreviação de structures, estruturas , em inglês) são conjuntos de variáveis que podem ou não ter tipos diferentes.

## 1.1 Declarando uma Struct

O exemplo aado mostra a declaração de uma struct “x” com 3 variáveis que podem ser utilizadas para armazenar nome, idade e nota.

```
1 struct{
2     char nome;
3     int idade;
4     float nota;
5 };
```

Você pode também dar um nome a struct da seguinte forma.

```
1 struct aluno{
2     char nome;
3     int idade;
4     float nota;
5 };
```

Para declaração de uma struct:

```
1 struct aluno aluno1;
2 struct aluno abc;
```

Para acessar cada variavel da estrutura basta utilizar:

```
1 aluno1.nome = "Joao";
2 aluno1.idade = 16;
3 aluno1.nota = 8.5;
```

## 1.2 Typedef

Structs podem ser tratadas como um novo tipo de dados da seguinte forma:

```
1 typedef struct aluno{
2     char nome;
3     int idade;
4     float nota;
5 }Aluno;
```

OU

```
1 struct aluno{  
    char nome;  
3    int idade;  
    float nota;  
5 };  
typedef struct aluno Aluno;  
7
```

Após declarado o novo tipo de dados é possível criar uma variável:

```
Aluno aluno1;  
2 Aluno abc;
```

# Praticando

Agora vamos praticar! Para todos os praticando em que é pedido que se escreva um programa, escreva o código do seu programa e nos envie o .c (código fonte) correspondente.

1. Faça um programa que armazene uma data em um struct. A entrada será ser escrita por extenso, ou seja, dia (inteiro), mês (escrito por extenso com a primeira letra maiúscula) e ano (inteiro). A saída deverá ser mostrada no formato DD/MM/AAAA conforme o exemplo abaixo.

**Obs: Todas as saídas dos seus códigos devem ser finalizadas sem quebra de linha.**

**Entrada:**

```
01
2 Setembro
1993
```

**Saída:**

```
01/09/1993
```

2. Suzana é professora e quer um programa que a auxilie com seus alunos. Tendo ela N alunos, ela quer colocar as notas da P1, da P2 e da P3, e saber se eles estarão aprovados ou reprovados. Considerando que a média é 7, faça um programa que receba o valor N, os nomes dos N alunos (separados por quebra e linha) e em seguida receba os valores da P1 dos N alunos, depois da P2 dos N alunos e da P3 dos N alunos e imprima se cada aluno está aprovado ou não, junto com sua média. A saída deverá conter os dados dos N alunos, sendo eles: nome, a média e sua situação (Aprovado ou Reprovado). Siga exatamente a saída. Faça a estrutura necessária para receber os nomes e suas devidas notas.

**Obs:** O tamanho máximo de caracteres para o nome de cada aluno é igual a 25.

Todas as saídas dos seus códigos devem ser finalizadas com uma quebra de linha.

**Entrada:**

```
3
2 Aluno1
Aluno2
4 Aluno3
8 7 7 (notas P1)
6 10 7 (notas P2)
5.5 7 7 (notas P3)
```

**Saída:**

```
Aluno1: media 6.5, Reprovado.
2 Aluno2: media 8, Aprovado.
Aluno3: media 7, Aprovado.
```

3. Você foi contratado por um banco para fazer um programa de gerenciamento de clientes. O cadastro dos clientes contém nome, cidade, número da conta e o saldo. O seu programa deve conter uma função para cadastrar clientes, uma para checar o saldo de um cliente e outra para imprimir o saldo de um cliente na tela.

Seu código deve ser dividido em 2 arquivos (banco.c, banco.h) e funcionar com o código main abaixo. **NÃO ENVIE O CÓDIGO MAIN PARA O RUN.CODES**

```

1 #include <stdio.h>
2 #include "banco.h"
3
4 int main(){
5     int n;
6     scanf("%d", &n);
7
8     Cliente clientes[n];
9
10    for(int i = 0; i < n; i++){
11        cadastraCliente(clientes, i);
12    }
13
14    float media = 0;
15    for(int i = 0; i < n; i++){
16        media+=saldoCliente(clientes, i);
17        imprimeSaldo(clientes, i);
18    }
19
20    printf("Media dos Salarios: %.2lf\n", media/(float)n);
21    return 0;
22 }

```

introcomp\_main.c

### Entrada:

A primeira linha da entrada contém um inteiro N indicando quantos clientes serão cadastrados. Cada uma das N linhas seguintes contem um inteiro, duas strings e um número decimal indicando o número da conta, nome, cidade e saldo do cliente, respectivamente.

**Obs:** Considere também que as strings possuem tamanho máximo igual a 50 e que não possuem espaços em branco.

```

1 3
2 333 Clara Vitoria 300.00
3 444 Brisa VilaVelha 0.02
4 555 Isabel Cariacica 2000.00

```

### Saída:

```

1 Cliente Clara mora em Vitoria e tem R$300.00 na conta 333
2 Cliente Brisa mora em VilaVelha e tem R$0.02 na conta 444
3 Cliente Isabel mora em Cariacica e tem R$2000.00 na conta 555
4 Media dos Salarios: 766.67

```

## Empreendendo

No mundo da tecnologia o que você não pode é ficar parado, ocioso no tempo. Você deve correr atrás, buscar a informação, o conhecimento. Empreender é isso, é correr atrás. Não é fundar uma empresa, ser um grande empresário e blablabla. É simplesmente correr atrás dos seus planos e sonhos e não deixar o tempo te pegar. Às vezes, ao nosso redor talvez não tenhamos exemplos de pessoas que são assim, querendo e buscando sempre mais do que é possível. Mas o que importa mesmo é explorarmos nossa capacidade e fazer aquilo que gostamos. Se você está gostando de programar, de pensar algoritmicamente, se você não vê a hora de produzir tecnologia, sei lá, um aplicativo de celular, uma solução na web, você está no caminho certo. Mas, entenda, em nada você terá sucesso sem esforço, sem ultrapassar barreiras insuportáveis, desde que essas barreiras façam parte dos seus objetivos. O Introcomp está aí pra isso, testar sua capacidade, sua vontade de fazer a diferença. Faça os praticando, curta os Workings, pergunte suas dúvidas, faça mais do que é pedido, busque mais na internet sobre programação de computadores, aplicações etc. Não é proibido que você saiba o que é uma struct ou até mesmo uma função recursiva antes de abordarmos estes tópicos com você (;D). Vá além! O desafio a seguir é proposto para que você possa ir além de seus limites, e desenvolver ainda mais sua capacidade de programação em C! Utilize todos os conceitos aprendidos em sala de aula para a compreensão do enunciado destes desafios, e preste muita atenção nos dados de entrada e saída. Explique o raciocínio utilizado para resolver o problema, e comente o código. Não esqueça de utilizar as boas práticas de programação, como indentação do código fonte, e atribuição de nomes de variáveis coerentes com o programa a ser desenvolvido.

## Desafios

1. Crie uma biblioteca(data.c e data.h) que contenha uma struct data(dia, mes e ano) e com as funções: InicializaData e AvançaData. A função InicializaData deve receber uma data e criar uma variável do tipo da struct. A função AvançaData recebe a data e retorna o próximo dia).

**Obs: A saída deverá conter a data no formato: DIA MES ANO, todas em número, e sem quebra de linha. Cabeçalho das funções:**

```
1 Data InicializaData();  
2 Data AvançaData(Data x);
```

### Exemplo de entradas e saídas

#### Entrada:

```
1 28 2 2012
```

#### Saída:

```
29 2 2012
```

**Entrada:**

31 8 2014

**Saída:**

1 9 2014