

Lista de exercícios 10 - Registros

1. Crie uma estrutura denominada **Tempo**, composta pelos campos **horas**, **minutos** e **segundos**. Leia do teclado um tempo expresso em segundos divida-o em horas, minutos e segundos. Armazene o resultado em uma variável do tipo **Tempo** (não use variáveis desnecessárias).
2. Adapte o programa anterior de tal forma a criar uma **função** que recebe como parâmetro de entrada um tempo em segundos e retorna uma estrutura **Tempo**, contendo tais segundos divididos em horas, minutos e segundos.
3. Crie um tipo registro com os campos nome, dia de aniversário e mês de aniversário. Desenvolva um programa que leia do teclado 5 registros e os armazene em um vetor. Depois, mostre para cada um dos meses do ano o nome dos aniversariantes.
4. Preencha **com dados aleatórios** um conjunto de 15 registros contendo código, telefone da loja e preço de um eletrodoméstico. Desenvolva um programa que permita exibir qual foi a média dos preços cadastrados e uma relação contendo o telefone das lojas cujo preço estava abaixo da média.
5. Um sistema de administração de pessoal mantém as informações referentes aos funcionários de uma empresa de duas formas: em um vetor de ponteiros para dados estruturados do tipo **Funcionario** ou em um vetor contendo dados estruturados do tipo **Funcionario**. Este tipo é descrito a seguir:

```
struct funcionario {  
    int mat; /* matricula do funcionario */  
    char nome[51]; /* nome do funcionario */  
    int sup; /* matricula do superior imediato */  
};  
typedef struct funcionario Funcionario;
```

- (a) Escreva uma função em C que recebe como parâmetros um vetor **contendo estruturas do tipo Funcionario**, através do ponteiro **vet**, para seu primeiro elemento, e do inteiro **n**, indicando seu número de elementos, e um inteiro **mat**, representando a matrícula de um funcionário. A função retorna o índice no vetor do funcionário correspondente, ou -1, se este não for encontrado. Protótipo:

```
int Indice_funcionario(Funcionario* vet, int n, int mat);
```

- (b) Escreva uma função em C que recebe como parâmetros **um vetor de ponteiros** para **Funcionario** (através do ponteiro **vet**, para seu primeiro elemento), um inteiro **n**, indicando seu número de elementos e um inteiro **mat**, representando a matrícula de um funcionário. A função deve retornar um novo vetor de estruturas **Funcionario** alocado dinamicamente contendo os funcionários que são subordinados diretos do funcionário cuja matrícula foi recebida por parâmetro. Um funcionário no topo da hierarquia, isto é, que não tem um superior imediato, tem o valor -1 no campo **sup**. Por exemplo, suponha que o vetor aponta para as funcionárias Luiza (**mat** = 123 e **sup** = 125), Diana (**mat** = 124 e **sup** = 129), Gina (**mat** = 125 e **sup** = 126), Dolores (**mat** = 126 e **sup** = -1), Beatriz (**mat** = 128 e **sup** = 126) e Ana (**mat** = 129 e **sup** = 128). Neste cenário, se a matrícula 126 for recebida por parâmetro, a função deve retornar um vetor de estruturas contendo os dados das funcionárias Gina e Beatriz. Protótipo:

```
Funcionario* determinaSubordinado(Funcionario** vet, int n, int mat);
```

6. Para o controle dos veículos de uma cidade, a secretaria dos transportes criou o seguinte registro:

Proprietário:_____ Combustível:_____
Modelo:_____ Cor:_____
No.Chassi:_____ Ano:____ Placa:_____

Sendo que o combustível pode ser álcool, gasolina ou diesel e a placa é formada por 3 letras e 4 dígitos. Sabendo que o número máximo de veículos da cidade é 5000, faça um programa que:

- (a) Contenha uma função que leia os dados dos veículos até que seja fornecido um nome de proprietário em branco;
- (b) Contenha uma função que imprima uma relação dos carros que sejam de um determinado ano em diante e sejam movidos com um determinado tipo de combustível;
- (c) Contenha uma função que imprima uma relação com todos os dados da ficha, para os veículos cuja placa começa com uma determinada letra e termina com um determinado dígito;
- (d) Contenha uma função que permita a troca de proprietário de um veículo, com o fornecimento do número do chassi. Para isso o usuário deve fornecer o nome do novo proprietário e o programa deve invocar uma função que busca qual registro contém o chassi a ser localizado e outra função para realizar a troca.

7. Considere o seguinte esqueleto de um código fonte C:

```
typedef float prova;  
typedef float media;  
  
struct provas{  
    prova p[3];  
    media M;  
};  
typedef struct provas Provas;  
  
struct aluno{  
    char ra[7];  
    int frequencia;  
    Provas Ps;  
};  
  
int main(void) {  
    int n,i,j;  
    float desvios_Ps[3]={0.0, 0.0, 0.0}, medias_Ps[3]={0.0, 0.0, 0.0};  
    struct aluno alunos[MAX_ALUNOS];  
    scanf("%d", &n);  
    /* COMPLETE AQUI */  
}
```

Complete o esqueleto acima. O programa deve ler um número inteiro n que indicará a quantidade de alunos na turma. Após isso, deve ler as notas das provas de cada aluno, calcular e mostrar na tela as notas e a média de cada aluno, a média geral de cada prova de todos os alunos e o desvio padrão de cada prova de todos os alunos. Note que não é necessária a declaração de mais nenhuma variável. Exemplo:

ENTRADA:	SAIDA:	
3	RA: 030034	RA: 123456
030034	Prova 0: 4.000000	Prova 0: 0.000000
4 5 6	Prova 1: 5.000000	Prova 1: 5.000000
	Prova 2: 6.000000	Prova 2: 10.000000
123456	Média: 5.000000	Média: 5.000000
0 5 10	-----	-----
	RA: 987654	
987654	Prova 0: 6.000000	
6 9 3	Prova 1: 9.000000	
	Prova 2: 3.000000	
	Média: 6.000000	

	Média geral P0: 3.333333	
	Desvio padrão P0: 2.494438	
	Média geral P1: 6.333333	
	Desvio padrão P1: 1.885618	
	Média geral P2: 6.333333	
	Desvio padrão P2: 2.867442	