



**FACULDADE DE EDUCAÇÃO TECNOLÓGICA DO ESTADO DO RIO DE JANEIRO**

**Exercícios de Fundamentos de Programação - Professor Leonardo**

**Vianna Structs**

**[2020/1]**

**Questão 01:**

Determinada zona eleitoral possui  $N$  eleitores inscritos que poderão votar, uma única vez, em um dos  $C$  candidatos para o único cargo a ser definido. Pede-se:

- A declaração de todo(s) o(s) tipo(s) de dados necessário(s) para o armazenamento dos números dos candidatos, assim como a quantidade de votos brancos, nulos e votos para cada candidato;
- Uma função que leia o voto de cada eleitor, considerando que o processo terminará quando os  $N$  eleitores votarem ou quando o número -1 for digitado como voto;
- Uma função que exiba:
  - O total de votos para cada candidato, assim com a quantidade de votos brancos (código 0) e de nulos (código sem candidato associado);
  - A quantidade de abstenções (eleitores que não votaram);
  - O número do candidato vencedor (caso tenha mais de 50% dos votos válidos) ou dos que irão para o segundo turno (os dois mais votados).

**Notas:**

- a) o número do candidato deve conter dois algarismos apenas;*
- b) podem considerar que não haverá mais de um candidato com a mesma quantidade de votos.*

**Questão 02:**

Uma loja de automóveis mantém os carros à venda sob a forma de um vetor de *structs* contendo as seguintes informações, para cada veículo: *placa, modelo, marca, cor, quilometragem, ano modelo/fabricação (deve ser um struct), valor e tipo (usado ou 0 km, conforme o valor do campo quilometragem)*.

Pede-se a declaração de uma variável que represente o estoque de veículos da loja, incluindo todas as declarações de tipos que possam ser necessárias.

Além disso, implementar as seguintes funções:

- Exibir todos os carros do modelo  $m$ , ano de fabricação entre  $a1$  e  $a2$  (inclusive), com valor não superior a  $x$  reais;
- Reajustar os valores de todos os carros 0 km, considerando um aumento de 2.5%;
- Retirar do estoque um determinado veículo, dada a sua placa.

**Questão 03:**

Determinada instituição de ensino armazena, sob a forma de vetores (de *structs*), as seguintes informações:

- Alunos: nome, matrícula;
- Disciplinas do curso: nome, código;
- Inscrições realizadas: matrícula do aluno, código da disciplina, semestre da inscrição (semestre/ano).

Pede-se o desenvolvimento de funções que realizem as seguintes operações:

- Exibir, para cada aluno, o total de inscrições já realizadas;
- Dado um semestre, informar para cada disciplina o número de alunos inscritos.

**Questão 04:**

Suponha a existência de um vetor de tamanho *TAM*, cada posição armazenando o *nome da pessoa* e a sua *data de aniversário* (representada por um *struct* do tipo *TData*, contendo os campos *dia* e *mes*). Pede-se o desenvolvimento das seguintes funções:

- Determinar a quantidade de pessoas que fazem aniversário no mês  $M$ ;
- Exibir os nomes de todas as pessoas que fazem aniversário entre as datas  $d1$  e  $d2$ , ambas do tipo *Tdata*.

**Questão 05:**

O histórico de um aluno é representado por um vetor de *structs* onde cada posição armazena o código da disciplina cursada, semestre e ano que a cumpriu e a média final na disciplina. Implementar uma função que exiba o histórico do aluno com o seguinte formato:

**NomeDisciplina1 (código1) MédiaDisciplina1**  
**NomeDisciplina2 (código2) MédiaDisciplina2**

..

**NomeDisciplinaN (códigoN) MédiaDisciplinaN**  
Coeficiente de rendimento: **CR**

Observações:

1. Para obter os dados da disciplina, um outro vetor de *structs* deve ser consultado. Este, por sua vez, armazena para cada disciplina do curso as seguintes informações: *código*, *nome* e *número de créditos*;
2. O coeficiente de rendimento consiste em uma média ponderada de todos os graus atribuídos às disciplinas cursadas, onde os pesos são representados pelo *número de créditos* da disciplina;
3. Todos os dados em negrito apresentados no formato do histórico devem ser obtidos a partir dos vetores.