

**Lista 3 - Variáveis Compostas Heterogêneas**

1. Defina uma estrutura para armazenar uma data. Escreva uma função que possua dois parâmetros do tipo estrutura data (um parâmetro representando a data de hoje e o outro uma data de nascimento), e retorna a idade de uma pessoa. A leitura das datas devem serem feitas na `main`.
2. Suponha um cadastro de participantes como um registro contendo informações do nome, CPF, tipo de participação (A, B, C ou D) e sócio da SBC (S – Sim ou N – Não). Escreva uma função que receba um registro como parâmetro e calcule e retorne o valor que o participante pagará para participar do evento. Considere ainda que sócio da SBC possui 20% de desconto.

Tipo de Partitipação	Valor a Pagar
A - 1 Curso	R\$ 30,00
B - 2 Cursos	R\$ 60,00
C - 3 Cursos	R\$ 90,00
D - 4 Cursos	R\$ 100,00

3. Uma fração é um modo de expressar a quantidade a partir de uma razão de dois números inteiros. Declare um novo tipo estrutura chamado *fracao* que possua dois campos representando os dois inteiros. Crie as seguintes funções:
  - `simplifique`: recebe como argumento uma variável do tipo `fracao` e simplifica esta `fracao`;
  - `para_real`: recebe como argumento uma variável do tipo `fracao` e retorna o número real a representando;
  - `some`: recebe duas variáveis do tipo `fracao` e retorna uma nova variável do tipo `fracao` representando a soma;
4. Faça um programa que realize o cadastro de contas bancárias com as seguintes informações: número da conta, nome do cliente e saldo. O banco permitirá o cadastramento de apenas 15 contas e não poderá haver mais que uma conta com o mesmo número. Crie o menu de opções a seguir.

Menu de opções:

1. Cadastrar contas.
  2. Visualizar todas as contas de determinado cliente.
  3. Excluir a conta com menor saldo (supondo a não existência de saldos iguais.)
  4. Sair.
5. Declare uma enumeração chamada `estado_civil` representando os possíveis valores: solteiro(a), casado(a), divorciado(a), viúvo(a).

6. Implemente um programa em C que simule o sorteio de 3 cartas distintas de um baralho. Cada carta deve ser representada por uma estrutura contendo dois campos do tipo `enum`: um para o valor (face) e outro para o naipe.

- Crie os tipos `enum` para os valores e para os naipes.
- Crie a `struct carta` com os dois campos do tipo `enum`.
- Inicialize um vetor de 52 cartas, representando um baralho completo.
- Sorteie 3 cartas distintas do baralho.
- Imprima na tela as cartas sorteadas no formato: Rei de Espadas, Três de Copas, etc.

Use `rand()` para o sorteio. Não é necessário embaralhar todo o baralho.