# **Estruturas de dados CONTABILIDADE e COMPRAS**

Os nomes das estruturas mostrados aqui não têm que ser e nem devem ser os finais. O professor liga aos nomes que damos às coisas, portanto temos que escolher com algum cuidado.

## Contabilidade

O módulo da contabilidade terá duas árvores, uma para guardar os produtos não comprados, e outra para os produtos com pelo menos uma compra.

/\* Estrutura principal do modulo\*/

struct contabilidade{

ARVORE prods\_nao\_comprados;

ARVORE prods\_comprados;

/\*Array para guardar dados globais de nº de compras

e total facturado em cada mês (ver querie 7)\*/

int contabilidade\_global[12][2];

};

/\*Estrutura de um nodo da arvore de produtos comprados\*/

struct nodo\_prod\_comprado{

char \*codigo;

int compras[12][4];

};

Como o nodo da árvore dos produtos não comprados é apenas o codigo (char \*) não se justifica criar struct.

Queries associadas e forma como o módulo as resolve:

**Querie 3** – Procurar o produto em ambas as árvores. Se for encontrado na arvore dos produtos nao comprados devolver 0. Se for encontrado na arvore dos produtos comprados, ler os valores da matriz.

**Querie 4** – Nº de produtos sem vendas é ler o valor de uma variavel (da libavl), listar os produtos é apenas iterar a árvore dos produtos não comprados.

**Querie 7** – Ler valores da matriz contabilidade\_global.

**Querie 11** – Responde ao nº de compras (contabilidade\_global), a parte dos clientes é respondida pelo módulo de compras.

**Querie 12** – Percorrer árvore, guardar dados numa estrutura (array?), ordenar e listar.

**Querie 14** – Indica os produtos (mesmo que querie 4), os clientes diz o modulo de compras.

## Compras

/\* Estrutura principal do modulo\*/

struct modulo\_compras{

ARVORE avl\_clientes;

};

struct nodo\_avl\_clientes{

char \*cod\_clientes;

ARVORE avl\_produtos;

Int num\_prods\_comprados[12];

char top3[3];

int num\_vendas\_top3[3];

};

struct nodo\_avl\_produtos{

char \*produtos;

/\*Numero de compras para cada mes, distinguindo entre normal e promocao.\*/

int num\_compras[12][2];

};

Queries associadas e forma como o módulo as resolve:

**Querie 5** – Procurar cliente e ler valores de num\_compras.

**Querie 8** – Percorrer a arvore dos clientes e ver se o produto existe para cada cliente, adicionar a estrutura auxiliar e ordenar.

**Querie 9** – Procurar o cliente e percorrer a sua árvore de produtos, adicionar a estrutura auxiliar e ordenar.

**Querie 11** – Ir percorrendo a arvore de clientes, ver valor de num\_prods\_comprados e preencher matriz auxiliar.

**Querie 13** – Ver array top3.

**Querie 14** – Ver num\_prods\_comprados e ir incrementando uma variavel, os produtos diz o modulo da contabilidade.