
Laboratório #3 - lab3.c

Mercearia

Prof. Valério Rosset

1 DEFINIÇÃO DO PROBLEMA

Você deve desenvolver um sistema de gerenciamento de estoque de uma mercearia. Através do seu sistema, o dono da mercearia deve ser capaz de registrar novos produtos, registrar as vendas realizadas e remover produtos cadastrados.

TAREFA A

Desenvolva um algoritmo em linguagem C - **utilizando lista dinâmica encadeada** - que forneça uma lista com os produtos cadastrados na mercearia **ordenados** pelo preço dos produtos. Ao final da execução, seu algoritmo deve informar o lucro obtido pela venda de cada um dos produtos cadastrados.

Seu algoritmo deve ser capaz de realizar as seguintes operações:

- **Inserção:** Seu algoritmo deve permitir a inserção **ordenada de forma decrescente** de produtos de acordo com o preço informado de cada produto.
- **Alteração:** Seu algoritmo deve permitir a alteração da quantidade disponível de qualquer produto cadastrado na lista.
- **Remoção:** Seu algoritmo deve permitir a remoção de qualquer produto cadastrado na lista.
- **Contabilizar:** Seu algoritmo deve contabilizar o lucro obtido através da venda de cada produto cadastrado na lista. Assim, sempre que um item for vendido, seu algoritmo deve incrementar o lucro obtido pela venda daquele produto.

ENTRADA

A entrada contém um único conjunto de teste, que deve ser lido do dispositivo de entrada padrão (normalmente o teclado). O conjunto de teste é composto por $L + 1$ linhas, $1 \leq L \leq 100$.

1. A primeira linha contém um valor inteiro L , $1 \leq L \leq 100$, que representa a quantidade de movimentações (reservas e retiradas) que serão realizadas.
2. Em cada uma das L linhas seguintes são dadas as informações. Cada linha contém um valor inteiro P , $1 \leq P \leq 4$, que representa a operação desejada, seguida de um espaço e o restante das informações necessárias.
 - a) **1:** A opção 1 permite ao usuário inserir um novo produto no cadastro. As informações que vem em seguida são:
 - ID do produto (int)
 - Quantidade do produto (int)
 - preço do produto (float)
 - b) **2:** A opção 2 permite aumentar a quantidade disponível de um determinado produto. As informações que vem em seguida são:
 - ID do produto (int)
 - Quantidade a ser adicionada (int)
 - c) **3:** A opção 3 significa que houve uma compra do produto. As informações que vem em seguida são:
 - ID do produto (int)
 - d) **4:** A opção 4 permite excluir um produto do cadastro. As informações que vem em seguida são:
 - ID do produto (int)

Para as operações de entrada, considere a sintaxe:

```
scanf("%d",&variavel);
```

```
scanf("%f",&preco);
```

```
scanf("%d %d",&variavel1, &variavel2);
```

SAÍDA

Como saída padrão, seu programa deve imprimir uma lista com o valor total arrecadado por cada produto cadastrado na lista. Note que quando um produto é removido do cadastro, nenhuma informação sobre ele deve ser impressa no final. A saída deve estar ordenada pelo preço **individual** de cada produto, e não pelo lucro total.

Além disso, as seguintes mensagens também devem aparecer:

1. Tentativa de cadastro de um produto já existente na lista: "ja existe".
2. Tentativa de remoção de um produto que não existe na lista: "nao existe".
3. Tentativa de compra de um produto que não existe na lista: "nao existe".
4. Tentativa de compra de um produto com quantidade 0: "nao existe".
5. Tentativa de alteração da quantidade de um produto que não existe na lista 0: "nao existe".

Ao final da execução, você deverá o lucro obtido pela venda de cada um dos produtos. Para as operações de saída, considere a sintaxe:

```
printf("Instrucao de saida\n");
```

```
printf("%d %f\n", id, lucrototal);
```

Entrada	Saída
6	nao existe
1 1 2 15	1 45.000000
2 1 1	
3 1	
3 1	
3 1	
3 1	

Entrada	Saída
8	ja existe
1 1 2 10	2 20.000000
1 1 1 10	3 30.000000
1 2 2 20	1 10.000000
1 3 2 15	
3 1	
3 2	
3 3	
3 3	

OBSERVAÇÕES

- É fundamental que o programa imprima apenas o valor de saída.
- Os exercícios devem ser postados no site <http://judge.sjc.unifesp.br>.
- Cada aluno deverá postar apenas um exercício dentro da data limite.
- A resolução submetida deve utilizar a estrutura de dados definida no enunciado do exercício.

- Os plágios serão penalizados. Como não será possível saber quem copiou, todos os casos envolvidos receberão punições iguais, portanto, não divulgue seu código.