

Trabalho prático ALG-II

Ray Santiago Del Negro

Trabalho Prático

Disciplina: Algoritmos e Programação II

Professores: Ivone, Mauro e Ronaldo

Dono do Mercado: “seu Canindé”.

Nome do mercado: Capi Lagoa.

Estrutura dos produtos:

Codigo - **inteiro**, maior que 0, valor único.

Descricao - **vetor de char**, não possui espaço nem acento.

Quantidade - inteiro, maior ou igual a zero.

Preco - real, maior ou igual a zero.

Funções:

Verificador de palavras em ordem alfabética / não precisa fazer é só usar strcmp de string.h

Verificador de código na lista

- estrutura da função verifica_codigo(produto *lst, int codigo)
- retorna um ponteiro produto
- Null ou a posição que estiver o codigo inserido

Inserir e validar um novo produto no sistema.

- Forma geral estrutura inserir(int codigo, char *descricao, int quantidade, float valor)
- Verificar se o código é único na lista;
- Usar o verificador de palavra crescente
- Printf de Sucesso: “Produto <codigo> inserido com sucesso!”
- Printf de Falha: “Erro ao inserir o produto <codigo>.”

Excluir produtos a partir do código.

- Forma geral da estrutura excluir(int codigo)
- Printf de Sucesso: “Produto <codigo> excluído com sucesso!”
- Printf de Falha: “Produto <codigo> não cadastrado!”

Atualizar produto (estoque/quantidade = “-q” e preço = “-v”) por meio do código do produto

- Forma geral da estrutura atualizar(char *p, int codigo, float valor)
- retirar ou aumentar a quantidade de produto

- não pode passar de zero
- Printf de Sucesso: "Produto <codigo> atualizado!"
- Printf de Falha: "Produto <codigo> não cadastrado."

Consultar produtos, você vai inserir uma sequência de caracteres e vão aparecer todos os produtos que possuem essa cadeia de caracteres.

- usar strstr() do string.h para ver substring
- Printf de Sucesso: "Printar todos os <codigo> - <nome> - <qtd em estoque> - <valor>" de caracteres em comum"
- Printf de Falha: "Nenhum produto encontrado!"

Relatórios dos produtos, apresenta todos os produtos em ordem alfabética, por letras

- Forma geral da estrutura relatorio(produto *lst)
- É necessário armazenar essa mensagem em um "Relatorio.txt"
- Letras iniciais que não estão no lista do produtos não vão aparecer
- Printf de sucesso:

```
<letra inicial>
    <codigo> - <nome> - <qtd em estoque> - <valor>
    <codigo> - <nome> - <qtd em estoque> - <valor>
<letra inicial>
    <codigo> - <nome> - <qtd em estoque> - <valor>
<letra inicial>
    <codigo> - <nome> - <qtd em estoque> - <valor>
    <codigo> - <nome> - <qtd em estoque> - <valor>
    <codigo> - <nome> - <qtd em estoque> - <valor>
:
```

- Printf de Falha: Nenhuma mensagem é mostrada

Importar produtos a partir de um arquivo de texto com a formatação:

numero de produtos

codigo;descricao;quantidade;float;valor

codigo;descricao;quantidade;float;valor

codigo;descricao;quantidade;float;valor

- Printf de Sucesso: "<quantidade de produtos> produtos importados!"
- Printf de Falha: "Erro ao importar do arquivo <nome do arquivo>."

Registro de Vendas esse comando são passados um conjunto de códigos de produtos (um por linha após a linha do comando) terminado pelo código 0 (zero) para finalizar.

- É necessário verificar se o produto pode ser vendido ou não.
- Um produto não pode ser vendido quando o código informado não está cadastrado ou quando a quantidade do produto em estoque é igual a zero.
- Uma mensagem indicando o motivo de não-venda do produto deve ser mostrada.
- Printf de motivo de não ser possível realizar a venda: "<codigo> - produto em falta no estoque"
- Printf de motivo de não ser possível realizar a venda: "<codigo> - produto não cadastrado"

- Um produto só pode ser vendido se o código estiver cadastrado e a quantidade em estoque for maior que zero.
- No caso de sucesso de venda, deve-se decrementar a quantidade do produto em estoque e exibir uma mensagem que contenha o código, o nome e o valor de venda do produto.
- Printf de Venda: "<codigo> - <nome> - <valor>".
- Ao final (código zero), deve ser mostrado na tela a soma total dos valores dos produtos vendidos com sucesso.
- Printf de finalização: "Total <soma dos valores>".

Help comando para mostrar o motivo de usar o tipo de lista escolhida

Sair, o comando mostra em ordem não-decrescente pela quantidade em estoque, o código, o nome e a quantidade em estoque de todos os produtos que estão com o estoque baixo (menos de 15 unidades). E ser salvo em um arquivo texto de nome Comprar.txt.

- Printf dos produtos que precisam de estoque: "<codigo> - <nome> - <qtd em estoque>"
- Os produtos cadastrados no sistema são salvos no arquivo Produtos.dat, Depois disso mantendo o formato do arquivo e finalizando o sistema

Primeiro Help do exerc

Antes de explicar minha escolha de tipo de lista, é importante ressaltar o motivo de eu não escolher as seguintes opções:

Pilha: o sistema de LIFO (Last in First out) é interessante em problemas que envolvem a abertura e fechaduras de chaves, ou pilhas de produtos. Talvez se tivesse algum sistema de prateleira seria interessante usar pilha.

Fila: o sistema de FIFO (First in First out) é bom em sistema com filas em particular, como a lista que vou criar vai estar em ordem crescente pelo código, não será necessário usar uma fila. Se tivesse um sistema de caixa, poderia usar a fila para organizar os clientes.

Lista dupla encadeada: Pensei em usar esse tipo de lista mas todas as funcionalidades apresentadas e as que criei podem ser feitas por uma lista simples, então usar essa lista só ocuparia memória.

Lista circular: Pensei em usar essa para ajudar na operação de registro de venda, para facilitar a busca de produtos, porém acho que não é muito efetivo porque não me sinto confiante em usar um ponteiro que fica mudando de posição em diferentes funções.

Lista com cabeça: Por afinidade escolhi usar sem cabeça para não confundir e criar novas funções.

Help que está no sistema:

A escolhida foi a Lista Encadeada Ordenada Simples sem Cabeça:

Portanto, por esses motivos o uso da Lista Encadeada Ordenada Simples sem Cabeça é a pensada para resolver o problema de seu Canindé, algumas especificações mudariam totalmente a forma que seria implementada a lista, porém este não é o caso, já que uma lista ordenada já consegue suprir todas as exigências do mercado Capi Lagoa.

Resposta resumida no help:

A escolhida foi a Lista Encadeada Ordenada Simples sem Cabeça:\n

A escolha desta lista foi feita a partir de uma análise de todas as funções\n a fim de deixar o programa com a complexidade mais baixa possível.\n

A ordenação escolhida é crescente a partir do código do produto,\n é escolhido essa forma de ordenação a partir das funções que são mais utilizadas.\n

Poderiam existir uma ordenação por ordem alfabética, igual feito no comando relatório mas seria usual. Já que as funções sair do programa e consultar tiram\n melhor proveito da ordenação crescente pelo código. \n

Portanto, o uso da Lista Encadeada Ordenada Simples sem Cabeça é a \n pensada para resolver o problema de seu Canindé, algumas especificações mudariam \n totalmente a forma que seria implementada a lista, porém este não é o caso, já que \n uma ordenada simples já consegue suprir todas as exigências do mercado Capi Lagoa. \n