**README - Sistema de Mercado de Esquina do Weldes**

**Descrição do Projeto**

**Este é um sistema simples de gerenciamento de produtos e carrinho de compras desenvolvido em linguagem C. O sistema simula o funcionamento de um mercado, permitindo ao usuário cadastrar produtos, listar os produtos cadastrados, realizar compras e finalizar um pedido. Os dados de produtos e carrinho são armazenados em arquivos de texto, garantindo persistência entre execuções do programa.**

**Funcionalidades**

* **Cadastro de Produtos: Adicionar novos produtos com código, nome, valor e quantidade. Se o produto já existir, a quantidade será atualizada.**
* **Listagem de Produtos: Exibir uma lista de todos os produtos cadastrados, mostrando código, nome, quantidade e valor unitário.**
* **Compra de Produtos: Adicionar produtos ao carrinho, verificando a disponibilidade em estoque.**
* **Listar Carrinho: Visualizar os produtos adicionados ao carrinho de compras com seus detalhes.**
* **Finalizar Pedido: Exibir o valor total da compra e, após confirmação do usuário, remover os arquivos relacionados ao carrinho.**
* **Persistência de Dados: Produtos e carrinho são armazenados em arquivos para garantir a persistência das informações entre execuções.**

**Estrutura do Projeto**

**O projeto é composto por quatro arquivos principais:**

1. **main.c: Arquivo principal que inicia o sistema chamando o menu principal.**
2. **telas.c: Responsável por exibir o menu principal e navegar pelas diferentes opções do sistema.**
3. **procedures.h: Contém as funções que manipulam arquivos, realizam operações no carrinho e cadastram produtos.**
4. **bibli.c: Define a estrutura de dados do produto e declara as funções usadas no projeto.**

**Comentários do Código**

**Arquivo main.c**

* **int main():** Função principal do sistema, responsável por iniciar o programa.
* **menuPrincipal():** Chama a função que exibe o menu principal do sistema e suas opções.

**Arquivo telas.c**

* **void menuPrincipal():** Função responsável por exibir o menu principal e gerenciar as opções do usuário.
  + O sistema utiliza system("cls") para limpar a tela e Sleep(1500) para adicionar uma breve pausa antes de exibir o menu.
  + Exibe as opções do sistema, como cadastrar produtos, listar produtos, comprar produtos, visualizar o carrinho, fechar o pedido, e sair do sistema.
  + A entrada do usuário é capturada usando scanf("%d", &x) e processada em um switch.
  + Cada caso no switch chama a função correspondente, como cadastrarProduto() para adicionar um produto, ou listarProdutos() para exibir os produtos cadastrados.

**Arquivo procedures.h**

* **FILE\* abrirArquivo(char \*arquivo, char \*modo):** Função genérica para abrir arquivos. Se o arquivo não puder ser aberto, retorna ao menu principal.
* **void lerArquivo(char \*arquivo, char \*modo):** Carrega os dados de produtos a partir de um arquivo. A função utiliza fscanf para ler os atributos do produto e armazená-los em um array de produtos.
* **void gravarArquivo(char \*arquivo, char \*modo):** Grava os dados de produtos em um arquivo. Itera sobre o array de produtos e grava suas informações se o código do produto for diferente de zero.
* **void cadastrarProduto():** Permite cadastrar um novo produto ou atualizar a quantidade de um produto existente.
  + Lê os produtos do arquivo com lerArquivo().
  + Verifica se o código do produto já existe, caso sim, atualiza a quantidade.
  + Caso o produto não exista, captura os detalhes do produto (nome, valor, quantidade) e grava no arquivo.
* **void listarProdutos():** Exibe todos os produtos cadastrados no sistema.
  + Utiliza lerArquivo() para carregar os produtos e lerIndice() para saber quantos produtos estão cadastrados.
  + Exibe o código, nome, quantidade e valor de cada produto.
* **void comprarProduto():** Permite ao usuário comprar um produto.
  + Carrega os produtos usando lerArquivo().
  + Solicita o código do produto desejado e a quantidade.
  + Verifica a quantidade disponível em estoque e atualiza o arquivo de produtos se a compra for válida.
  + Grava os produtos no carrinho usando gravarArquivoCarrinho().
* **void listarCarrinho():** Exibe os produtos que foram adicionados ao carrinho.
  + Carrega os itens do carrinho com lerArquivoCarrinho() e exibe suas informações.
* **void receberPagamento():** Finaliza a compra e exibe o valor total dos itens no carrinho.
  + Calcula o valor total da compra somando os valores de cada item no carrinho.
  + Após a confirmação, remove os arquivos relacionados ao carrinho para simular a finalização da compra.

**Arquivo bibli.c**

* **Estrutura struct produto:** Define a estrutura de um produto, que inclui código, nome, valor e quantidade.
  + Um array de produto[50] é utilizado para armazenar até 50 produtos.
* **Variável quantCompra[50]:** Armazena a quantidade de cada produto que foi adicionado ao carrinho de compras.
* **Declaração das funções:** Contém as declarações das funções principais usadas para manipulação de arquivos, cadastro e compra de produtos.