

Disciplina: Programação Estruturada e Modular

Prof. Carlos Veríssimo

Relatorio Atividade N2-7

Data - 29/11/2024

Autores: Caio Yugo Saeda Uemura

1. Modularização

O código apresenta uma boa estrutura com módulos, com cada operação CRUD e outras funcionalidades (como listar e comprar produtos) implementadas em funções separadas. Mas há algumas melhorias que poderiam ser feitas:

- As funções possuem lógica redundante (como a busca por ID em várias delas), o que poderia ser extraído para uma função auxiliar (`buscarProdutoPorID`) para evitar duplicação.

2. Elementos Conceituais

- Uso de structs para representar os produtos, encapsulando seus atributos.
- Utilização de ponteiros para manipular diretamente os dados do array de produtos.
- Uso do `#define` limites como o número máximo de produtos e tamanhos de strings.

Melhorias ao código:

- Adotar o tipo `double` para valores monetários em vez de `float`, como indicado nos requisitos, já que o `double` tem maior precisão.
- Melhor tratamento de erros, como validar entradas do usuário para evitar valores inválidos. Ex: Colocar letras ao invés de números no preço.

3. Elementos de Negócio (Requisitos)

O código os requisitos funcionais principais:

- Cadastro, consulta, alteração, exclusão e listagem de produtos estão implementados.
- A função de compra reduz o estoque e calcula o total baseado no preço e quantidade.

O código inclui um atributo `descricao` que não estava nos requisitos.

4. Regras Explícitas

- **CRUD:** Todas as operações estão implementadas de maneira funcional.
- **Manipulação com ponteiros:** As funções recebem e manipulam diretamente a estrutura de produtos por ponteiro.
- **Uso de struct**

5. Regras Implícitas

Algumas regras implícitas foram abordadas:

- **Evitar IDs duplicados:** Embora não haja verificação explícita de duplicidade, isso deveria ser adicionado ao código.
- **Estoque não negativo:** A lógica para compras verifica se há quantidade suficiente em estoque antes de realizar a operação.
- **Validação de entradas:** Algumas entradas são verificadas (como o limite de produtos), mas outras, como o tipo de dado informado pelo usuário, não são.

Sugestões de Melhoria

1. **Validação de Entradas:** Adicionar verificações para validar o tipo de dado digitado pelo usuário e tratar erros, como caracteres inválidos.
2. **Trocar float por double:** Atender à especificação que pede o uso de double para valores monetários.
3. **Prevenção de IDs Duplicados:** Antes de inserir um produto, verificar se o ID já existe na lista.
4. **Mensagens ao Usuário:** Melhorar a clareza das mensagens de erro e sucesso, além de padronizá-las.