

Rebeca da Costa  
Matrícula: 202510703613

## Relatório Projeto Programação

Para compilar:

make

./mercearia

estoque\_funcoes.c

validar\_data

A função valida se a data inserida pelo usuário está no formato correto (dd/mm/aaaa) também valida se a data existe incluindo os anos bissextos, por exemplo, se o usuário digitar a data 31/02/2025 vai informar que a data é inválida pois o código faz a verificação do ano bissexto usando if, voltando para o 1 se o formato for inválido, imprimindo no terminal a mensagem "Entrada inválida" e voltando para a opção de escrever a data novamente, assim fazendo usuário ficar em um loop até escrever uma data válida.

adicionar\_produto

A função usa if e for para verificar se o estoque está cheio, se estiver ele vai imprimir no terminal a mensagem "Estoque cheio!"

se não estiver ela mostra as opções para o usuário inserir para adicionar o produto no estoque, ela verifica também se todos os dados escritos pelo usuário são válidos, se o código já existe, se a quantidade e preço são válidos e se a data é válida. Usando ponteiros para apontar para outras funções e atualizar os produtos.

verificar\_validade

A função verifica a validade dos produtos em estoque, se tiver algum vencido irá gerar um relatório das informações do produto no terminal, usa laços for para percorrer os produtos e ponteiros para preencher os arranjos de resultados que serão usados no main também usado a função malloc que aloca um bloco de memória durante a execução do programa, que está inclusa na biblioteca <stdlib.h>

buscar\_produto

A função utiliza if, while e else para buscar produtos pelo código ou nome depois de adicionados no estoque usando a função malloc para encontrar os produtos no bloco de memória e imprimir o resultado da busca no terminal, se o nome ou código escrito para busca estiver errado ela irá informar a voltar para o menu.

carregar\_estoque

A função utiliza if, while, else e as funções fopen para abrir um arquivo e retornar para o ponteiro do tipo

FILE e fclose que fecha o arquivo apontado pelo ponteiro assim lendo os dados salvos no arquivo e abrindo o arquivo estoque.txt para leitura além de ele zerar o contador total\_produtos para começar a escrever do início.

#### gerar\_relatorio

Define uma função que recebe um ponteiro para uma string (char \*relatorio) e é onde o relatório vai ser armazenado.

Monta uma string com o relatório completo do estoque, incluindo todos os produtos e o valor total usando o sprintf para formatar.

#### mercearia.c

A função main é a única que interage com o usuario e mostra os menus e responde ao usuário, utiliza o do-while para manter o menu aberto fazendo um loop até o usuário escolher a opção para sair.

Usa switch para selecionar a ação de acordo com a opção do que o usuário escolheu como: cadastrar, buscar, atualizar e entre outros.

Chama funções de estoque\_funcoes.c e trata os retornos usando if/else para imprimir as mensagens no terminal, também utiliza ponteiros para receber o resultado das funções de busca e de validade.

#### util.c

##### limpar\_buffer

Limpa o buffer do teclado para não ter problemas com o scanf ou getchar.

##### copiar\_string

Copia a informação de uma string para outra, para que assim não passe do tamanho tamanho limite e preenche o resto com caracteres nulos.

##### tamanho\_string

Ela determina quantos caracteres uma string vai ter.

##### concatenar\_string:

Ela junta o conteúdo de uma string ao final de outra para ficar com a formatação certa.

#### estoque.h

AS funções do arquivo usam ponteiros e o arquivo define a estrutura e as funções do sistema de gerenciamento de estoque como:

```
#ifndef ESTOQUE_H
#define ESTOQUE_H
#define MAX_PRODUTOS 500
#define DATA_TAMANHO 11
```

util.h

O arquivo define as funções para manipular com as strings, ele tem funções para limpar o buffer de entrada, copiar strings com controlando o tamanho, comparando strings, calculando o tamanho de strings e concatenando as strings.