

# Relatório de Programação

## Gestão de Vendas

Este relatório tem como objetivo apresentar a implementação do programa de gestão de vendas, desenvolvido como atividade formativa 2, do curso de Laboratório de Programação. O enunciado escolhido foi o enunciado A, que solicita o desenvolvimento de um programa capaz de gerir as vendas de uma loja. O programa foi organizado em cinco módulos: `common.h`, `file_io.h`, `product.h`, `sale.h` e `search`.

Na função `main` designado por “`VendasModulo.c`” o programa começa incluindo as bibliotecas e os arquivos de cabeçalho necessários para o funcionamento do programa. Em seguida, chama as funções **`read_products`** e **`read_sales`** para ler as informações dos produtos e das vendas de arquivos CSV. Em seguida, a função **`update_sales`** é chamada para atualizar as quantidades vendidas de cada produto com base nas informações de venda. Depois disso, ele entra em um loop de menu que exibe as opções disponíveis e aguarda a entrada do usuário. Dependendo da opção selecionada, ele chama as funções correspondentes: **`add_product`**, **`remove_product`**, **`edit_product`**, **`search_product`** ou **`generate_report`**.

Se a opção 6 for selecionada, o programa termina.

Abaixo encontram-se as descrições das funções disponíveis no programa:

- **`read_products`**: lê as informações dos produtos de um arquivo CSV e armazena-os em uma matriz de estruturas **`Product`**.
- **`read_sales`**: lê as informações de vendas de um arquivo CSV e armazena-os em uma matriz de estruturas **`Sale`**.
- **`update_sales`**: atualiza as quantidades vendidas de cada produto com base nas informações de venda.
- **`add_product`**: solicita ao usuário as informações de um novo produto e adiciona-o à matriz de estruturas **`Product`**.
- **`remove_product`**: solicita ao usuário o código de um produto e remove-o da matriz de estruturas **`Product`**.

- **edit\_product:** solicita ao usuário o código de um produto e permite que o usuário edite suas informações.
- **search\_product:** solicita ao usuário um termo de pesquisa, nome ou código, e exibe uma lista de produtos correspondentes.
- **generate\_report:** gera um relatório de vendas para cada produto, com base nas informações de venda (Quantidade vendida, data, receita total do produto, assim como a receita de todos os produtos).

As estruturas de dados usadas na implementação são as seguintes:

- **Product:** uma estrutura que contém informações sobre um produto, incluindo seu código, nome, preço e quantidade vendida.
- **Sale:** uma estrutura que contém informações sobre uma venda, incluindo o código do produto, a quantidade vendida e a data da venda.

A interface do módulo principal consiste em chamar as funções nos outros módulos para executar as tarefas necessárias e fornecer uma interface do usuário para selecionar as opções do menu.

Os módulos adicionais incluem:

- **comon.h:** um arquivo de cabeçalho que contém as definições de constantes (estruturas) usadas em todo o programa.
- **file\_io.h:** um arquivo de cabeçalho que contém as definições de funções para ler arquivos CSV.
- **product.h:** um arquivo de cabeçalho que contém as definições de funções para manipular as informações do produto como, adicionar, remover e editar.
- **sale.h:** um arquivo de cabeçalho que contém as definições de funções para gerência as informações de venda, atualizar as vendas e gerar um relatório detalhado.
- **search.h:** um arquivo de cabeçalho que contém as definições de funções para pesquisar produtos com base em um termo de pesquisa.

Em suma, o programa apresenta uma solução para gerenciar as informações de produtos e vendas de uma loja.

