

Trabalho Prático 2 - Sistema para controle de estoque

O objetivo desse Trabalho Prático é treinar a manipulação de arquivos em C e a criação de funções para execução de determinados procedimentos.

O trabalho prático consiste na implementação de um sistema de controle de estoque, que receberá como entrada a lista de produtos em um arquivo de extensão TXT.

O programa inicialmente deve ler as entradas contendo informações sobre os produtos, armazenar essas entradas em alguma estrutura interna (em memória) e realizar as operações solicitadas.

Seu programa deve ser capaz de receber dados por parâmetro de linha de comando, bem como trabalhar com leitura e gravação de arquivos de texto.

ENTRADA: A entrada possui várias linhas, onde cada linha tem detalhes sobre determinado produto.

- A primeira linha possui a quantidade (Q) de produtos que constam na lista ($1 \leq Q \leq 1000$);
- As próximas $5 * Q$ linhas representam os dados de cada produto, na seguinte ordem (Um dado por linha):
 - Código
 - Nome
 - Quantidade
 - Preço
 - Estado

Execução: O programa deve ser executado a partir da linha de comando com os seguintes parâmetros:

.\tp2 [param1] [param2] [param3]

[param1] (Obrigatório): Nome do arquivo de entrada.

Ex.: input.txt

IMPORTANTE: O arquivo TXT deve estar no mesmo diretório do programa.

[param2] (Obrigatório): Número correspondente a função a ser executada, a saber:

1. Gerar relatório de estoque
2. Pesquisar por produto pelo código
3. Listar dados do produto com menor quantidade em estoque
4. Listar produtos por estado
5. Encontrar produto com menor quantidade em estoque do estado

6. Calcular a quantidade total de itens no estoque

[param3] (Opcional): Nome do arquivo de saída (se função 1), código do produto desejado (se função 2), ou sigla do estado desejado (funções 4 e 5).

Exemplo de cada chamada, considerando o arquivo input.txt como arquivo de entrada:

1. Gerar relatório de estoque: **.\tp2 input.txt 1 output.txt**
2. Pesquisar por produto pelo código **.\tp2 input.txt 2 0123**
3. Listar dados do produto com menor quantidade em estoque **.\tp2 input.txt 3**
4. Listar produtos por estado **.\tp2 input.txt 4 MG**
5. Encontrar produto com menor quantidade em estoque do estado **.\tp2 input.txt 5 SP**
6. Calcular a quantidade total de itens no estoque: **.\tp2 input.txt 6**

SAÍDA: A saída esperada para cada uma das opções é:

1. Gerar relatório de estoque:
O programa deve gerar um arquivo TXT, com o nome passado pelo **param3**, salvo no mesmo diretório de execução. Deve-se salvar a lista de todos os produtos COM OS NOMES EM ORDEM ALFABÉTICA, com os mesmos dados da entrada, um dado por linha (Código, Nome, Quantidade, Preço e Estado).
2. Pesquisar por produto pelo código:
O programa deverá imprimir na saída padrão (tela) todos os dados do produto que tenha o código igual ao informado no **param3**, um dado por linha (Código, Nome, Quantidade, Preço e Estado).
3. Listar dados do produto com menor quantidade em estoque:
O programa deverá imprimir na saída padrão (tela) todos os dados do produto que possui a menor quantidade em estoque, um dado por linha (Código, Nome, Quantidade, Preço e Estado).
4. Pesquisar por produto pelo estado:
O programa deverá imprimir na saída padrão (tela) a lista de produtos do estado informado no **param3**, EM ORDEM ALFABÉTICA, com todos os dados de cada produto, um dado por linha (Código, Nome, Quantidade, Preço e Estado).
5. Encontrar produto com menor quantidade em estoque do estado:
O programa deverá imprimir na saída padrão (tela) o produto do estado informado no **param3** que possui a menor quantidade em estoque, um dado por linha (Código, Nome, Quantidade, Preço e Estado).
6. Calcular a quantidade total de itens no estoque:
O programa deverá imprimir na saída padrão (tela) a quantidade total de itens e, em seguida, o preço total (R\$) de todos os produtos presentes no estoque. Para o cálculo da quantidade total de itens do estoque deve-se somar a quantidade de cada produto, recebida no arquivo de entrada.

Cada dado deve ser impresso por linha.

Ex.: 125

EXEMPLOS:

Entrada	Linha de Execução	Saída
4 //Quantidade de produtos 109791 BOMBOM GAROTO SORTIDO 250G 6 10.98 PI 97818 MOLHO SHOYU TRAD.AZUMA 8ML 25 0.98 AP 78942 MOLHO SHOYU SAKURA 99 78 PI 92129 LEITE COND.ITALAC 395G 4 3.78 RO	.\tp2 input.txt 1 output.txt	109791 BOMBOM GAROTO SORTIDO 250G 6 10.98 PI 92129 LEITE COND.ITALAC 395G 4 3.78 RO 78942 MOLHO SHOYU SAKURA 99 78.00 PI 97818 MOLHO SHOYU TRAD.AZUMA 8ML 25 0.98 AP
4 //Quantidade de produtos 109791 BOMBOM GAROTO SORTIDO 250G 6 10.98 PI 97818 MOLHO SHOYU TRAD.AZUMA 8ML 25 0.98 AP 78942 MOLHO SHOYU SAKURA 99 78 PI 92129 LEITE COND.ITALAC 395G 4 3.78 RO	.\tp2 input.txt 2 97818	97818 MOLHO SHOYU TRAD.AZUMA 8ML 25 0.98 AP
4 //Quantidade de produtos 109791 BOMBOM GAROTO SORTIDO 250G 6 10.98	.\tp2 input.txt 3	92129 LEITE COND.ITALAC 395G 4 3.78 RO

PI 97818 MOLHO SHOYU TRAD.AZUMA 8ML 25 0.98 AP 78942 MOLHO SHOYU SAKURA 99 78 PI 92129 LEITE COND.ITALAC 395G 4 3.78 RO		
4 //Quantidade de produtos 109791 BOMBOM GAROTO SORTIDO 250G 6 10.98 PI 97818 MOLHO SHOYU TRAD.AZUMA 8ML 25 0.98 AP 78942 MOLHO SHOYU SAKURA 99 78 PI 92129 LEITE COND.ITALAC 395G 4 3.78 RO	.\tp2 input.txt 4 PI	109791 BOMBOM GAROTO SORTIDO 250G 6 10.98 PI 78942 MOLHO SHOYU SAKURA 99 78.00 PI
4 //Quantidade de produtos 109791 BOMBOM GAROTO SORTIDO 250G 6 10.98 PI 97818 MOLHO SHOYU TRAD.AZUMA 8ML 25 0.98 AP 78942 MOLHO SHOYU SAKURA 99 78 PI 92129 LEITE COND.ITALAC 395G 4 3.78	.\tp2 input.txt 5 PI	109791 BOMBOM GAROTO SORTIDO 250G 6 10.98 PI

RO		
4 //Quantidade de produtos 109791 BOMBOM GAROTO SORTIDO 250G 6 10.98 PI 97818 MOLHO SHOYU TRAD.AZUMA 8ML 25 0.98 AP 78942 MOLHO SHOYU SAKURA 99 78 PI 92129 LEITE COND.ITALAC 395G 4 3.78 RO	.\tp2 input.txt 6	134

DICA: Para imprimir na tela o preço do produto utilize printf("%.2f ", preco);

ENTREGA: Cópias de trabalho são inaceitáveis, podem ser detectadas automaticamente e serão repassadas ao conselho disciplinar, além de receberem nota 0 (ZERO). Seu código deve ser desenvolvido na linguagem C e entregue através do Moodle.