

## Trabalho 4

### LEIA ATENTAMENTE AS REGRAS E OS ENUNCIADOS

<b>R E G R A S</b>
--------------------

- O trabalho deverá ser realizado em grupo (2 a 4 pessoas).
- Não serão aceitos trabalhos individuais ou com mais de 4 pessoas.
- O trabalho deverá ser enviado para o Google classroom até o dia 12/05/2021 (quarta-feira).
- O arquivo solicitado deverá ser zipado em um único arquivo juntamente com o nome dos membros do grupo.
- O programa deverá ter o nome conforme definido no enunciado.
- Não serão aceitos trabalhos enviados por email.
- Trabalhos com estruturas e/ou organizações semelhantes (plágio) serão penalizados com a nota zero.
- O programa que não obedecer às restrições estabelecidas receberá zero.

<b>E N U N C I A D O</b>
--------------------------

1) programa: supermercado.cpp

Crie um programa para administrar o estoque de produtos de um supermercado. O programa deverá ter um menu principal e um menu secundário. Os layouts da interface com o usuário estão definidos no final desse documento e a implementação deve seguir esses layouts.

**Incluir produto:** deverá ser solicitado código, descrição, categoria, quantidade e preço do produto a ser incluído.

- a) Código: número inteiro com 6 dígitos.
- b) Descrição: string com no mínimo 4 e no máximo 30 caracteres.
- c) Categoria: caracter A, B, C, D ou E.
- d) Quantidade: número inteiro  $> 0$ .
- e) Preço: float  $> 0$ .
- f) Não podem existir dois produtos com o mesmo código.

- g) A quantidade máxima de produtos é 50.
- h) Descrição e categoria deverão ser armazenadas em caixa alta.
- i) Apresentar mensagem de sucesso ou a mensagem de erro correspondente.

**Excluir produto:** deverá ser solicitado o código do produto a ser excluído.

→ BUSCA

- a) Código do produto deve corresponder a um produto existente.
- b) Não é possível excluir um produto se quantidade em estoque  $> 0$ .
- c) Apresentar mensagem de sucesso ou a mensagem de erro correspondente (produto não existe ou quantidade em estoque  $> 0$ ).

**Abater do estoque:** deverá ser solicitado o código do produto e a quantidade a abater.

- a) Código do produto deve corresponder a um produto existente.
- b) Quantidade a abater  $> 0$ .
- c) Não é possível abater quantidade maior que a quantidade existente em estoque.
- d) Apresentar mensagem de sucesso ou a mensagem de erro correspondente (produto não existe, quantidade a abater  $\leq 0$  ou quantidade em estoque menor que a quantidade a abater).

→ BUSCA

**Listar todos os produtos:** a ordenação pode ser por descrição ou por categoria/descrição, ou seja, ordena por categoria e, para os produtos com a mesma categoria, ordena por descrição.

**Listar produtos com estoque baixo:** deverá ser solicitada a quantidade mínima de estoque e deverão ser apresentados todos os produtos com quantidade menor que a quantidade mínima.

- a) Quantidade mínima  $\geq 0$ .
- b) O programa deve apresentar mensagem de erro caso quantidade mínima  $< 0$ .

→ BUSCA

**Aplicar aumento/desconto:** deverá ser solicitada a operação desejada (A-aumento ou D-desconto), o código ou a categoria e o percentual.

- a) Operação: deve ser A (aumento) ou D (desconto)
- b) Código: código do produto a ser aumentado/descontado ou 0 (zero) para aumentar/descontar categoria.
- c) Categoria: define a categoria (A, B, C, D ou E) a ser aumentada/descontada ou T para aumentar/descontar todos os produtos. Só deve ser solicitado se código = 0.
- d) Percentual de aumento/desconto  $> 0$ .
- e) Apresentar mensagem de sucesso com a quantidade de produtos aumentados/descontados ou a mensagem de erro correspondente (operação inválida, produto não existe, categoria inválida, percentual de aumento/desconto  $\leq 0$ ).

### Informações gerais:

- a) A implementação deverá seguir os layouts de interface definidos no final desse documento.
- b) Valores monetários devem ser impressos com 2 casas decimais.
- c) Não poderão ser utilizadas variáveis globais. Todos os vetores deverão ser criados na função **main** e passados para as demais funções via parâmetros.

### Dicas:

- Antes de iniciar a programação, crie a(s) struct(s) necessária(s) para resolver o problema e verifique se essas representações atendem às operações descritas no enunciado.
- Crie funções básicas para leitura dos dados e implemente nessas funções a validação dos dados. Exemplos:
  - Função para ler código com 6 dígitos.
  - Função para ler um número positivo.
  - Função para ler uma string com no mínimo n e no máximo k caracteres.
  - etc.
- Crie funções para cada uma das ordenações solicitadas.
- Crie funções para cada um dos menus.
- Crie funções para cada uma das operações do programa descritas anteriormente (incluir produto, excluir produto, listar todos os produtos por descrição, listar todos os produtos por categoria/descrição, etc.).
- Nos exercícios resolvidos no material de aula e nas listas de exercícios encontram-se vários problemas semelhantes aos propostos nesse trabalho. Estude-os!

## Layouts

```
Menu Principal
1-Incluir produto
2-Excluir produto
3-Abater do estoque
4-Listar produtos
5-Aplicar aumento/desconto
6-Fim
```

Layout 1 – Menu principal

```
Menu da Listagem
1-Listar todos os produtos (ordenado por descrição)
2-Listar todos os produtos (ordenado por categoria/descrição)
3-Listar produtos com estoque baixo
5-Voltar
```

Layout 2 – Menu da listagem

```
Codigo: 1234
Descricao: Queijo parmesão
Categoria: A
Qtd.: 100
Preco: 12.5

Produto cadastrado com sucesso!
```

Layout 3 – Inclusão de produto

```
Codigo: 1234
Produto "Queijo parmesão" excluído com sucesso!

OU

Codigo: 2345
Produto não existe!

OU

Codigo: 1234
Produto "Queijo parmesão" tem 50 em estoque!
```

Layout 4 – Exclusão de produto

```

Codigo: 1234
Qtd.: 20
Quantidade abatida. Produto "Queijo parmesão" agora tem 80 em
estoque.

OU

Codigo: 1234
Qtd.: 120
Produto "Queijo parmesão" tem somente 100 em estoque.

OU

Codigo: 2345
Produto não existe!

```

Layout 5 – Abater do estoque

Codigo	Descricao	Categ	Qtd	Preco
999999	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	c	9999	9999,99
999999	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	c	9999	9999,99
999999	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	c	9999	9999,99
999999	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	c	9999	9999,99

Layout 6 – Listagem dos produtos (em qualquer ordem)

Qtd minima: 20				
Codigo	Descricao	Categ	Qtd	Preco
999999	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	c	8	9999,99
999999	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	c	15	9999,99
999999	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	c	2	9999,99
999999	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	c	13	9999,99

Layout 7 – Listagem dos produtos com estoque baixo

Operacao (A-aumento ou D-desconto): A  
Codigo: 0  
Categoria: T  
Percentual: 20  
Operacao realizada com sucesso. Produtos aumentados: 15

OU

Operacao (A-aumento ou D-desconto): D  
Codigo: 0  
Categoria: A  
Percentual: 20  
Operacao realizada com sucesso. Produtos descontados: 3

OU

Operacao (A-aumento ou D-desconto): A  
Codigo: 1234  
Percentual: 5  
Operacao realizada com sucesso. Produtos aumentados: 1

OU

Operacao (A-aumento ou D-desconto): A  
Codigo: 2345  
Produto não existe!

OU

Operacao (A-aumento ou D-desconto): A  
Codigo: 0  
Categoria: X  
Categoria não existe!

Layout 8 – Aplicar aumento/desconto