

 <p>INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA SÃO PAULO</p>	<p><b>CTII 1xx</b> <b>ALGORITMOS</b> Prof. Me Luiz Henrique Kiehn</p>	<p><b>CAMPUS CUBATÃO</b></p>
--	---	----------------------------------

### Trabalho B2T1 – 2º Bimestre 2024

- Este trabalho poderá ser realizado em dupla, SEM consulta entre grupos.
- É **permitida** consulta aos materiais de aula e à Internet, bem como a qualquer bibliografia que auxilie na realização do trabalho.
- A critério do professor, os trabalhos que forem considerados semelhantes, mesmo que com nomes de variáveis distintas e linhas eventualmente trocadas de posição, serão invalidados, ficando os grupos envolvidos sem nota.

### Bom trabalho!

Este trabalho é continuação do trabalho B1T1. Agora, deverá ser acrescentado código para realizar, além das instruções já implementadas (mostrar a tabela com os 12 produtos na tela), os seguintes requisitos:

1. Antes de prosseguirmos, vamos efetuar alguns ajustes nos códigos, para deixá-los preparados para este e os próximos trabalhos.

**\*\*\* ATENÇÃO! Texto acrescentado posteriormente.**

- 1.1. Após apresentar a tabela com os 12 produtos na tela, acrescente os comandos abaixo para, respectivamente, limpar a tela e pausar a execução até o usuário teclar F9.

```
limpatela // Comando para limpar a tela
pausa    // Comando para pausar a execução do programa
```

**\*\*\* Fim do texto acrescentado posteriormente.**

1.2. As variáveis que guardam os dados dos produtos devem seguir o padrão indicado abaixo.

codProdNN, descrProdNN, undProdNN: caractere  
precoProdNN, saldoEstoqueNN: real

onde:

NN: numeração sequencial iniciando de 01 (01, 02, 03...).

Por exemplo:

codProd01, codProd02, precoProd01, precoProd02...

1.3. Crie (**declare**) variáveis adicionais de código, descrição, unidade, preço e saldo de estoque para armazenar os dados de 12 produtos, como na imagem abaixo. **Nota importante:** OS CÓDIGOS E DESCRIÇÕES DEVEM SER EXATAMENTE OS AQUI INDICADOS.



Código	Descrição	Preço	Quant.	Estoque
7891025101604	Leite	R\$ 3.00	15.00	l
7891000120101	Creme de leite	R\$ 3.70	28.00	lt
7891000100103	Leite condensado	R\$ 4.50	23.00	cx
7891000105016	Barra cereal	R\$ 3.20	18.00	und
7897236904805	Água	R\$ 2.15	32.00	gf
7891150036567	Caldo de galinha	R\$ 1.35	12.00	cx
7891999000538	Iogurte	R\$ 2.60	25.00	und
7896051126041	Leite fermentado	R\$ 1.80	33.00	und
7622300830083	Biscoito recheado	R\$ 1.90	13.00	und
4005900036728	Desodorante	R\$ 11.10	14.00	und
7896185989819	Vitamina C	R\$ 35.20	16.00	und
7898113811452	Lanche	R\$ 9.50	19.00	und
Valor total do estoque:				R\$ 1442.90

2. Crie (**declare**) variáveis adicionais para armazenar os dados (descrição, unidade, preço e saldo de estoque) do produto informado pelo usuário. Por exemplo:

codigoProduto, descricaoProduto, unidadeProduto: caractere  
precoProduto, saldoEstoque: real

3. Crie uma variável para armazenar o valor total da compra. Por exemplo:

`valorTotalCompra: real`

4. Crie (**declare**) uma variável (por exemplo, `produtoExiste` ou `produtoEncontrado`) do tipo lógico que servirá para indicar se o código informado pelo usuário existe na tabela (verdadeiro) ou não (falso).
5. Inicialize com zero o valor da variável de total da compra criada no item 3 acima e apresente na tela um cabeçalho de cupom fiscal conforme a figura abaixo.



```

C:\ Console simulando o modo texto do MS-DOS
-----
| SUPERMERCADO FEDERAL |
-----
| Código Prod. | Descrição Produto | Preço (R$) | Quant. | Und | Valor item |
-----

```

6. Vamos implementar agora uma estrutura de repetição para que o usuário possa informar códigos de produtos mais de uma vez. A repetição deverá ocorrer até que o código do produto (variável `codigoProduto`) informado pelo usuário seja igual à palavra “fim”, ou enquanto esse código for diferente da palavra “fim”.

**NOTA:** Atentem para o fato de que, logo após informado o código do produto pelo usuário, se ele escreveu a palavra “fim”, as demais instruções de tratamento e impressão do item não deverão ser executadas, devendo o programa sair da repetição e apresentar o rodapé do cupom, com o valor total da compra. **NÃO USEM O COMANDO INTERROMPA!**

7. Dentro da repetição:

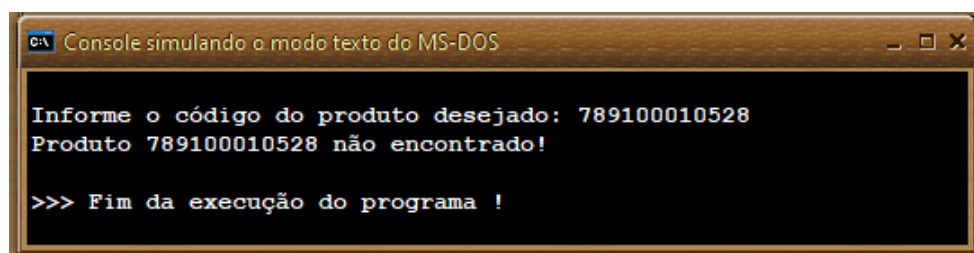
7.1. Solicite o código do produto (`codigoProduto`) para o usuário.

7.2. Inicialize a variável (criada no item 4) que indica se o código informado pelo usuário existe na tabela de produtos com o valor **verdadeiro**.

7.3. Verifique a qual dos 12 produtos o código informado pelo usuário corresponde. Use uma estrutura escolha-caso para isso. Em cada caso, atribua os conteúdos das variáveis do produto correspondente às respectivas variáveis criadas no item 2. Na cláusula outrocaso, apenas atribua **falso** à variável de verificação da existência do produto. Por exemplo:

```
escolha codigoProduto
caso codProd01:
    descricaoProduto <- descrProd01
    unidadeProduto  <- undProd01
    precoProd       <- precoProd01
    saldoEstoque    <- saldoEstoque01
```

7.4. Após a estrutura escolha-caso, use a estrutura se (1º se) para verificar se o produto foi encontrado. Em caso negativo, apresente uma mensagem indicando que o produto não foi encontrado na tabela; informe nessa mensagem o código fornecido pelo usuário. Caso contrário (1º senao), solicite que o usuário informe a quantidade desejada para aquele produto (item 7.5).



7.5. Dentro do 1º senao (do item 7.4), solicite que o usuário informe a quantidade desejada para aquele produto. Use outra estrutura se (2º se, dentro do 1º senao e logo após solicitar a quantidade) para verificar se há estoque suficiente para atender a quantidade solicitada; se não for suficiente, apresente uma mensagem para o usuário informando-o sobre isso. Caso contrário (2º senao), significa que, neste momento, o produto foi **encontrado** e que sua quantidade em estoque supre a quantidade solicitada pelo usuário, devendo-se executar os próximos passos.

```
C:\> Console simulando o modo texto do MS-DOS

Informe o código do produto desejado: 7891000100103
Informe a quantidade desejada: 30
Não há quantidade suficiente em estoque.

>>> Fim da execução do programa !
```

7.6. Mostre, em duas linhas e usando formatação, os dados do produto **encontrado** (que estão nas variáveis adicionais **declaradas** no item 2 e **atualizadas** no item 7.3). A primeira linha deve apresentar, nesta ordem: o código, a descrição, o preço, a quantidade, a unidade e o produto (multiplicação) do preço pela quantidade (correspondente ao valor total do item). A segunda linha deve apresentar o saldo em estoque e, novamente, a unidade; o saldo apresentado deve corresponder ao saldo inicial subtraída a quantidade informada pelo usuário. Efetue o acúmulo do valor total do item na variável de totalização do valor da compra criada no item 3. A figura abaixo apresenta uma sugestão de leiaute.

```
C:\> Console simulando o modo texto do MS-DOS

-----
| SUPERMERCADO FEDERAL                                     |
-----
| Código Prod. | Descrição Produto      | Preço (R$) | Quant.  Und | Valor item |
-----
Informe o código do produto desejado: 7891000100103
Informe a quantidade desejada: 6

| 7891000100103 | Leite condensado      | R$   4.50 |   6.00 cx | R$   27.00 |
Estoque atualizado:   17.00 cx

Informe o código do produto desejado: |
```

- Quando o usuário informar o **fim** da inserção de itens, apresente na tela duas linhas tracejadas para delimitar o fim do cupom. Entre essas linhas, apresente o valor total da compra; esse valor deve compreender apenas os itens que passaram pelas verificações efetuadas nos itens 7.4 e 7.5.

A próxima imagem mostra um exemplo de como fica a tela após a execução do programa informando 3 itens.

```

C:\> Console simulando o modo texto do MS-DOS

-----
| SUPERMERCADO FEDERAL                                     |
-----
| Código Prod. | Descrição Produto      | Preço (R$) | Quant.  Und | Valor item |
-----
Informe o código do produto desejado: 7891000100103
Informe a quantidade desejada: 5

| 7891000100103 | Leite condensado      | R$   4.50 |   5.00 cx  | R$   22.50 |
Estoque atualizado:   18.00 cx

Informe o código do produto desejado: 7897236904805
Informe a quantidade desejada: 2

| 7897236904805 | Água                  | R$   2.15 |   2.00 gf  | R$    4.30 |
Estoque atualizado:   30.00 gf

Informe o código do produto desejado: 7891000120101
Informe a quantidade desejada: 3

| 7891000120101 | Creme de leite        | R$   3.70 |   3.00 lt  | R$   11.10 |
Estoque atualizado:   25.00 lt

Informe o código do produto desejado: fim

-----
|                                                    Total da compra: R$   37.90 |
-----

>>> Fim da execução do programa !

```

9. O nome de cada programa deverá estar no padrão abaixo. **Não se esqueçam de fazer constar os nomes de todos os membros do grupo no programa, na parte de comentários iniciais.** O arquivo entregue deverá estar compactado (.zip ou .rar), seguindo o padrão de nome indicado abaixo, usando-se, porém, B2T1.

IFSP2024\_ALG\_CTII<sup>nnn</sup>\_B2T1\_Gmm.alg (arquivo do programa)

IFSP2024\_ALG\_CTII<sup>nnn</sup>\_B2T1\_Gmm.rar (arquivo compactado)

sendo:

**nnn**: número da turma (117 ou 148).

**mm**: número do grupo dentro da turma (01, 02, 03...); a turma deverá definir o número de cada grupo.

Por exemplo: IFSP2024\_ALG\_CTII<sup>117</sup>\_B2T1\_G01.zip

10. O arquivo gerado deverá ser submetido, na atividade **ALG – Avaliação B2T1**, no Moodle, até as 23h59min do dia **08/09/2024**.