

Modalidade: em grupo (no máximo 3 alunos por grupo)

Data de entrega: 27/março/2023 até 23:59 na tarefa do Microsoft Teams. Entregar UM ÚNICO ARQUIVO COMPACTADO .ZIP CONTENDO APENAS OS ARQUIVOS FONTES e o arquivo *readme*. São descontados 25% da nota por dia de atraso.

Fazer um programa em linguagem C para gerenciar o cadastro de produtos de um *site* de vendas online, utilizando árvore binária implementada em **arquivo binário**, conforme descrito a seguir:

- existem vários produtos cadastrados, contendo as seguintes informações:
 - código: número inteiro que identifica univocamente cada produto
 - nome: nome do produto, tendo no máximo 50 caracteres
 - marca: marca do produto, tendo no máximo 30 caracteres
 - categoria: descreve a categoria do produto, tendo no máximo 50 caracteres
 - estoque: número de unidades disponível no estoque
 - preço: preço unitário do produto.
- O sistema deve ter as seguintes funcionalidades implementadas, organizadas convenientemente em menus:
 - cadastrar produto: cadastra um produto a partir das informações fornecidas pelo usuário
 - remover produto: remove um produto do cadastro, a partir do seu código
 - atualizar preço: atualiza o preço de um produto, a partir de seu código
 - atualizar estoque: atualiza o estoque de um produto, a partir de seu código
 - imprimir informações de um produto, a partir de seu código
 - imprimir lista de todos os produtos: percorrimo *in-ordem* da árvore binária imprimindo os códigos e os respectivos nomes dos produtos
 - imprimir árvore binária: imprime apenas os códigos dos produtos formatados visualmente como uma árvore binária
 - imprimir lista de livres: imprime a lista de nós livres da árvore devido às remoções
 - realizar operações em lote: realiza operações de inserção, alteração e remoção a partir de arquivo texto a ser carregado, cuja sintaxe é descrito mais adiante.

Toda a interface do programa deve ser apresentada em modo texto, sendo executado no *prompt* do *shell* do sistema operacional.

O conteúdo do arquivo texto para realização de operações em lote, deve ter sintaxe conforme abaixo:

`< operacao >; < codigo >; < campo1 >; < campo2 >; ... < campoN >`
onde:

- `< operacao >`: pode ser I (inserção), A (alteração) ou R (remoção)
- `< campo1 >; < campo2 >; ... < campoN >` correspondem aos campos com informações referentes ao respectivo item e operação.

Exemplo de conteúdo de arquivo .txt contendo lote de operações a serem executadas:

```
I;70;Relogio smartwatch;Polar;eletronicos e tecnologia;27;566,70
I;25;Leite;Parmalat;bebidas;358;7,70
I;200;Microondas;LG;eletrodomesticos;53;690,99
I;80;Multiprocessador;Arno;eletrodomesticos;7;299,90
I;50;Guarana;Antartica;bebidas;200;5,50
```

I;30;12 Regras para a Vida: um antídoto para o caos;Alta Books;livro; 54,90
 A;25;340;8,30
 A;80;5;
 A;30;;61,90
 I;11;Impressora Laser;HP;eletronicos e tecnologia;15;779,90
 I;240;Dom Casmurro;Cia das Letras;livro;30;22,90
 I;100;A Condição Humana;Ed. Pensamento; livro;77;96,90
 R;50
 A;100;72;
 I;120;Celular;Apple;eletronicos e tecnologia;25;3200,00
 I;90;Suco de laranja;Del Valle; bebidas; 200;9,90
 R;200

Assim, a linha

I;70;Relógio smartwatch;Polar;eletronicos e tecnologia;27;566,70
 é interpretada como:

- o código do produto é 70
- o nome é Relógio smartwatch
- a marca é Polar
- a categoria é eletronicos e tecnologia
- o estoque tem 27 unidades
- o preço unitário é 566,70.

A linha A;25;340;8,30 é interpretada como:

- o código do produto é 25
- o estoque é alterado para 340 unidades
- o preço unitário é alterado para 8,30

A linha A;30;;61,90 é interpretada como:

- o código do produto é 30
- o estoque permanece inalterado
- o preço unitário é alterado para 61,90

A linha A;80;5; é interpretada como:

- o código do produto é 80
- o estoque é alterado para 5 unidades
- o preço unitário permanece inalterado

Também devem ser observadas as seguintes condições:

- Caso haja tentativa de inserção de um produto com um código já existente, a operação deve ser ignorada.
- Caso haja tentativa de alterar ou remover um produto de código inexistente na árvore, a operação deve ser ignorada.

As informações devem ser armazenadas em arquivo binário, tal que ao se remover nós da árvore, os mesmos devem ser encadeados em uma lista de livres, para serem reaproveitados em inserções futuras.

Entregar um arquivo *readme* contendo:

- nomes dos componentes do grupo
- opções implementadas, seguidas de uma breve explicação das opções do menu para o usuário do programa

Critérios de avaliação:

- o trabalho deverá ser apresentado oralmente em grupo, seguido de arguição sobre o código/teoria relacionada.
 - a nota referente à apresentação oral é individual conforme desempenho do aluno na arguição.
 - o aluno do grupo que não esteja presente durante a apresentação oral, não terá a nota correspondente.
- documentação/organização do código-fonte: peso 1
- arquivo *readme*: peso 1
- apresentação oral: peso 1
- implementação das funcionalidades: peso 7

Em todas as funcionalidades, a avaliação será feita considerando correção e qualidade do código. Trabalhos copiados ou plagiados receberão nota ZERO!