

### Modalidade:

- assíncrono com os horários de aula
- em grupo (no máximo 3 alunos por grupo)

**Data de entrega:** 19/julho até 23:59 na tarefa do Microsoft Teams. Entregar UM ÚNICO ARQUIVO COMPACTADO .ZIP CONTENDO APENAS OS ARQUIVOS FONTES, e um relatório em PDF, informando os nomes dos alunos do grupo. São descontados 25% da nota por dia de atraso.

Uma loja de materiais precisa manter um cadastro de seus produtos. Implemente na linguagem C uma **árvore B em arquivo binário** de acordo com as especificações abaixo:

- considerar que todas as *strings* são sem acentos.
- todas as opções do programa devem ser mostradas em um menu no *shell* do sistema operacional. Usar somente bibliotecas ANSI C. **NÃO implementar interfaces gráficas!**
- documentar o código fonte, colocando para cada função implementada as pré-condições e pós-condições, tanto nos arquivos *.h* quanto nos *.c*.
- não podem ser usadas bibliotecas prontas dessas estruturas de dados, devendo todas elas serem implementadas pelo grupo.
- as seguintes operações devem ser implementadas:
  - inserir: insere as informações de um produto no cadastro:
    - \* código do produto: representado por um número inteiro
    - \* nome: cadeia de caracteres (50 no máximo)
    - \* quantidade em estoque: número inteiro
    - \* preço unitário: número real com 2 casas decimais
    - \* localização: *string* contendo a descrição do local em que o produto se encontra armazenado (100 caracteres no máximo)

Todas as informações para a operação de inserção são obrigatórias.

- remover: dado o código de um produto, remove-o do cadastro
  - alterar estoque: altera a quantidade em estoque
  - alterar preço: altera o preço unitário
  - alterar localização: altera a localização
  - carregar arquivo texto: faz a carga de um arquivo texto, cujo caminho é informado pelo usuário, contendo operações e respectivas informações a serem aplicadas ao cadastro
  - informações do produto: a partir do código de um produto específico, imprime suas informações
  - listar produtos: lista as informações dos produtos do cadastro por ordem crescente de código
  - imprimir árvore: imprime a árvore B por níveis (apenas os códigos dos produtos) de maneira que seja possível identificar os nós existentes com as respectivas chaves.
  - imprimir posições livres do arquivo de dados: imprime a lista de posições livres do arquivo de dados
  - imprimir posições livres do arquivo de índices: imprime a lista de posições livres do arquivo de índices
- As informações devem ser separadas em 2 arquivos distintos, conforme esquema da figura 1:

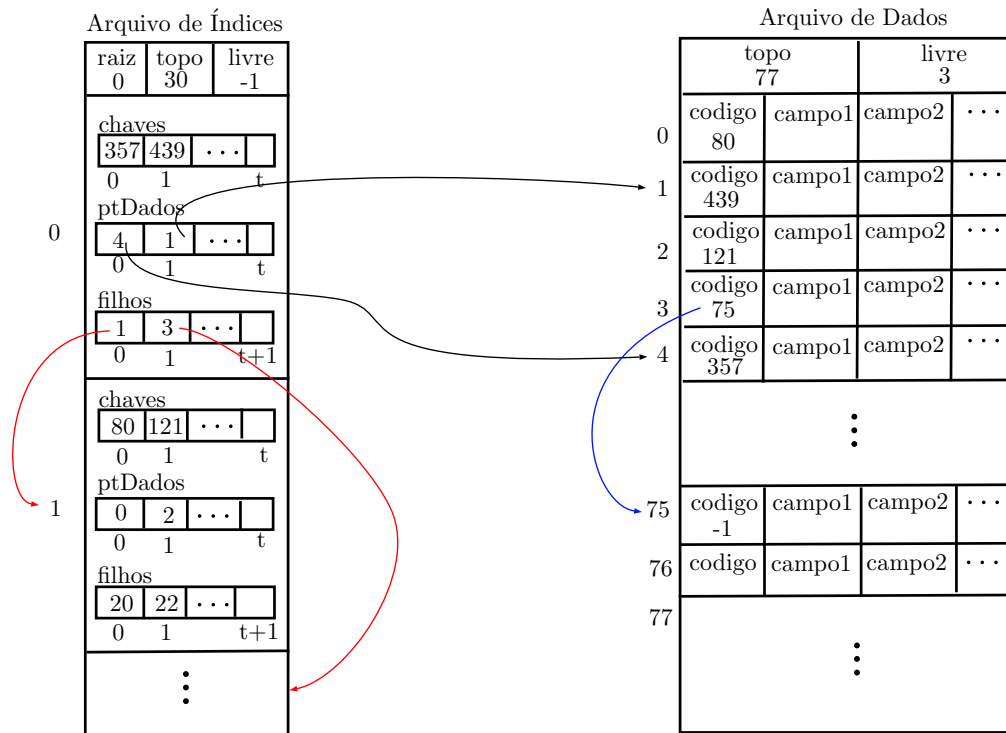


Figura 1: Esquema de arquivos para árvore B

1. um arquivo para os registros de dados: contendo código, nome, quantidade em estoque, preço unitário, e localização. E um campo adicional para encadeamento de registros livres.
  2. um arquivo para registros de índices, organizado na forma de árvore B, sendo o campo código usado como chave. Além de campos para encadeamento dos nós da árvore, devem haver campos para armazenar as posições dos registros de dados no arquivo de dados.
- tanto o arquivo de dados quanto o arquivo de índices devem ser arquivos binários.
  - a árvore B NÃO DEVE SER CARREGADA INTEIRAMENTE na memória principal, apenas as informações do registro sendo manipulado no momento.
  - no início do arquivo de índices (árvore B) deve haver um cabeçalho contendo o endereço do registro raiz, a posição do topo (primeira posição livre do arquivo) e o endereço da cabeça de nós (páginas) livres.
  - no início do arquivo de dados deve haver um cabeçalho contendo o endereço do topo (primeira posição livre do arquivo) e o endereço da cabeça de registros de dados livres
  - No caso de remoção de um produto do cadastro, a posição correspondente dentro do arquivo binário deve ser colocada em uma lista de nós livres para reaproveitamento em futuras inserções. Deve existir uma lista de livres para o arquivo de índices e outra para o arquivo de dados. Pode ser reaproveitado algum campo do registro para fazer o encadeamento da lista de livres.
  - a implementação deve considerar a possibilidade de escolher a ordem da árvore B alterando-se em um único local do código fonte. Para efeitos de teste, iremos considerar árvore B de ordem 5.
  - o arquivo texto a ser carregado contém uma operação por linha conforme exemplo a seguir:

```
I;20;parafuso 3mm;500;2,00;prateleira 5A
I;7;bucha 2p;100;1,20;prateleira 1B
I;11;alicate fino;20;30,00;prateleira 3C
I;5;chave inglesa;80;8,00;prateleira 5C
I;13;chave de fenda 1p;30;15,00;prateleira 7A
R;7
```

I;120;lixa 2mm;300;1,50;prateleira 3A  
I;80;parafuso 5mm;250;3,00;prateleira 5B  
A;7;200;1,30;prateleira 1C  
R;55  
A;30;;25,00;prateleira 2B  
I;70;broca 8p;140;5,00;prateleira 5C  
R;13  
A;5;;;prateleira 2B

- as informações são separadas por ponto-e-vírgula
  - brancos no início e no fim de cada campo devem ser retirados antes da aplicação da operação
  - I indica inserção, e as informações que se seguem são nome do produto, quantidade em estoque, preço unitário e localização
  - A indica alteração, e as informações que se seguem são quantidade em estoque, preço unitário e localização. Não é possível alterar o nome de um produto. Quando alguma informação é omitida, significa que o respectivo campo não deve ser alterado.
  - R indica remoção, e a informação que se segue é o código do produto a ser removido
  - no caso da operação de inserção, caso seja informado um código já existente no cadastro, a operação de inserção deve ser ignorada.
  - no caso das operações de alteração e remoção, caso seja informado um código inexistente no cadastro, as operações devem ser ignoradas.
  - as operações devem ser aplicadas na mesma ordem em que aparecem no arquivo texto
- entregar um relatório em PDF com as seguintes informações:
    - nome dos componentes do grupo
    - instruções para execução do programa
    - descrição sucinta das principais estruturas de dados implementadas
  - o trabalho deverá ser apresentado oralmente em grupo para o professor no horário síncrono de aula, constando de execução do programa, 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.

Critérios de avaliação:

- documentação/organização do código fonte: peso 1
- relatório PDF: peso 1
- implementação correta das funcionalidades: peso 8, assim distribuído:
  - 70% para funcionamento correto
  - 30% para qualidade do código implementado na funcionalidade

A funcionalidade de alteração (preço, quantidade em estoque e localização) serão avaliadas como uma única funcionalidade.

Trabalhos copiados ou plagiados receberão nota ZERO!