

## Exercício 7 - ABB

### Sobre a entrega:

- exercício **INDIVIDUAL**
- submissão no **run.codes** – veja instruções no arquivo **run\_codes.pdf**
- até **20/11/2020**, 23:59h

**Objetivos:** revisar conceitos básicos de árvore binária de busca.

### Parte 1: Implementação

**Problema:** implementar, em linguagem C, o acervo de filmes no sistema Netflix, usando o TAD ABB visto em aula.

#### Simplificações:

- Considere que a árvore armazena elementos do tipo **ITEM**, definido da seguinte maneira:
  - `int` chave; /\*identificador numérico do filme \*/
  - `char` nome[20]; /\*nome do filme\*/
- A chave passada é sempre um inteiro;
- O nome do filme não possui espaços;
- O final do caso teste é indicado por um 0.

**Implemente** o TAD ABB, visto nas aulas sobre Árvore Binárias, utilizando a abordagem Dinâmica. O TAD deve conter, no mínimo, as seguintes funções, onde o seu número indicará a operação na entrada:

- 1) **criar árvore**
- 2) **apagar árvore**
- 3) **inserir item**
- 4) **remover item informando a chave**
- 5) **buscar chave**
- 6) **imprimir ordenado (pela chave)**
- 7) **altura**

OBS: a altura da árvore deve ser armazenada como campo da estrutura, e atualizado a cada inserção e remoção.

Utilizando o TAD implementado, desenvolva um programa para armazenar e manipular a lista de filmes do Netflix, considerando a estrutura definida em ITEM.

#### Requisitos (obrigatórios):

- *Makefile* compilado com gcc C99, flag -Wall

Entregar o arquivo no formato .zip contendo:

- *Makefile*
- main.c
- abb.c
- abb.h

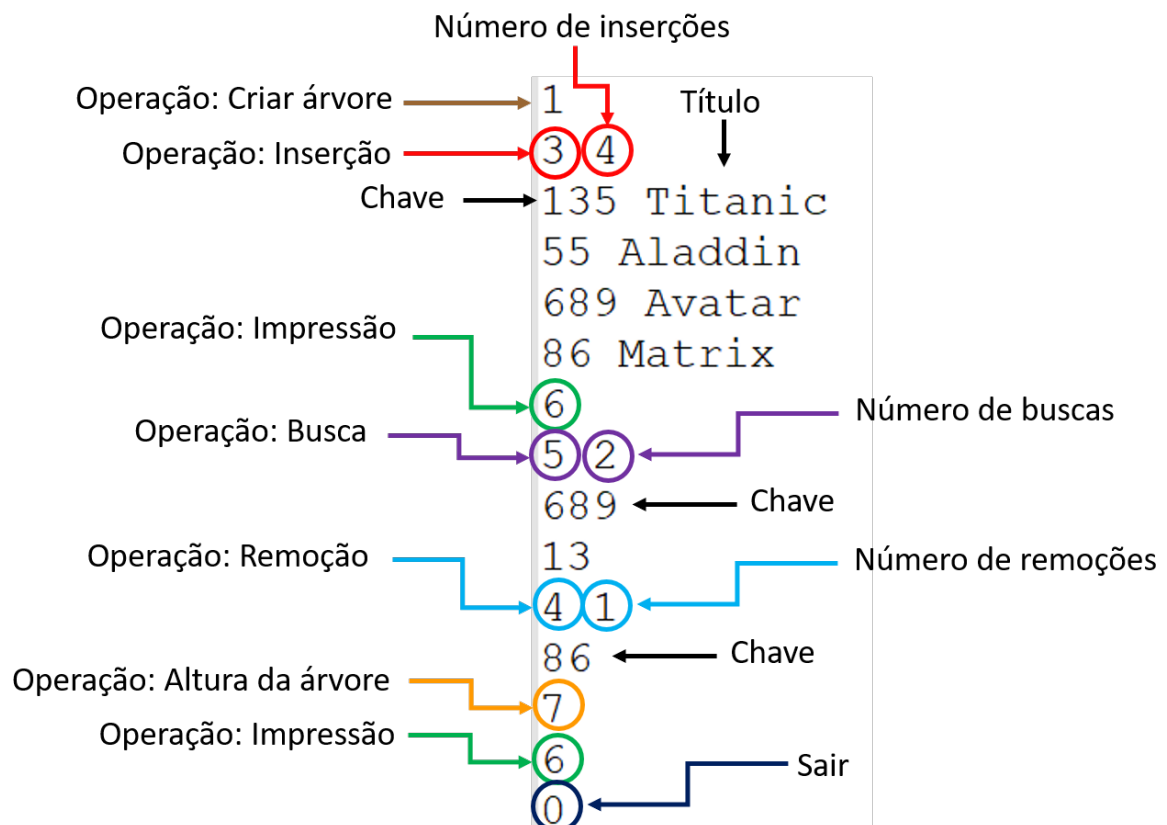
### Tratamento de exceção:

- Operações em árvore não alocada devem retornar ERRO (exceto **criar**);
- Elemento (chave) inexistente:
  - Busca deve retornar ERRO;
  - Remoção deve retornar ERRO.
- Inserção de elemento repetido (chave repetida) deve retornar ERRO.

### Exemplo de Entrada e Saída Esperada

OBS: seguir o padrão de entrada e saída é OBRIGATÓRIO. Exercícios com entrada/saída fora do esperado não serão corrigidos. Atenção para os números que identificam cada operação.

Entrada



#### Saída

```

1 ← Árvore criada
1 ← Inseriu
1
1
1
55 Aladdin
86 Matrix
135 Titanic
689 Avatar
Avatar ← Chave buscada está na árvore
-32000 ← Chave inexistente
1 ← Removeu
2 ← Altura da árvore
55 Aladdin
135 Titanic
689 Avatar
  
```

Impressão

Impressão

#### IMPORTANTE:

- ✓ Lembre-se e aplique as “boas práticas de programação”, tais como: organizar e documentar bem o código, definir adequadamente os nomes de variáveis e funções, modularizar o código, identificar autor do código em todos os arquivos, etc...
- ✓ Nomeie os arquivos de modo a facilitar correção do código.