

**Árvore Binária de Busca – valor: 4 pontos**

- 1) **Resolução de problemas com árvore - Criando um índice remissivo:** Dado um texto **T** e um conjunto de palavras **p**, um gerador de índices remissivos associa a cada palavra **p** as posições nas quais ela é referenciada em **T**.

Por exemplo, a partir do texto na Tabela 1 e das palavras-chaves {espelho, achar, de, quem}, obtemos o índice remissivo da Tabela 2. Nesse índice, as palavras-chaves aparecem em ordem alfabética e suas referências (linhas) em ordem crescente.

1. Cantiga para Tom Jobim
- 2.
3. Quem for além simplesmente
4. Deste espelho transparente
5. há de sumir? Ou se ver?
6. relembrar? Ou esquecer?
7. Quem for além simplesmente
8. deste espelho transparente
9. há de sentir? ou sonhar?
10. progredir? ou regressar?
11. Mas quem achar uma seta
12. que lhe apontar o sentido
13. neste espelho, há de se achar
14. no paraíso, perdido,
15. onde achar o poeta,
16. de repente ou devagar.

Tabela 1 – Texto qualquer

A impressão deve estar em ordem alfabética. Lembrando, caminhar de forma simétrica permite imprimir em ordem alfabética.

Achar, 11,13,15  
De,5,9,13,16  
Espelho,4,8,13  
Quem,3,7,11

Tabela 2 – Índice remissivo gerado

Crie um arquivo conforme Tabela 3 de palavras chaves para auxiliar a montagem do índice remissivo. Esse arquivo será gerado pelo usuário com as palavras que ele desejar.

Espelho  
Quem  
Achar  
De  
Vitoria

Tabela 3 – Arquivo de palavras chaves

Um índice remissivo será representado por uma árvore de busca binária, representado na Figura 1, em que cada nó armazena uma palavra-chave e sua respectiva lista de referências, além dos ponteiros para as subárvores esquerda e direita. O tipo Arvore, bem como da lista de referências, está definido na Tabela 4. Use o TAD Lista para a variável referência.

```
Estrutura arvore{  
    Estrutura arvore* esq;  
    Estrutura arvore* dir;  
    String palavra;  
    Lista Referencia ;  
}  
Tipo definido estrutura arvore Arvore;
```

Tabela 4 – Estrutura de dados

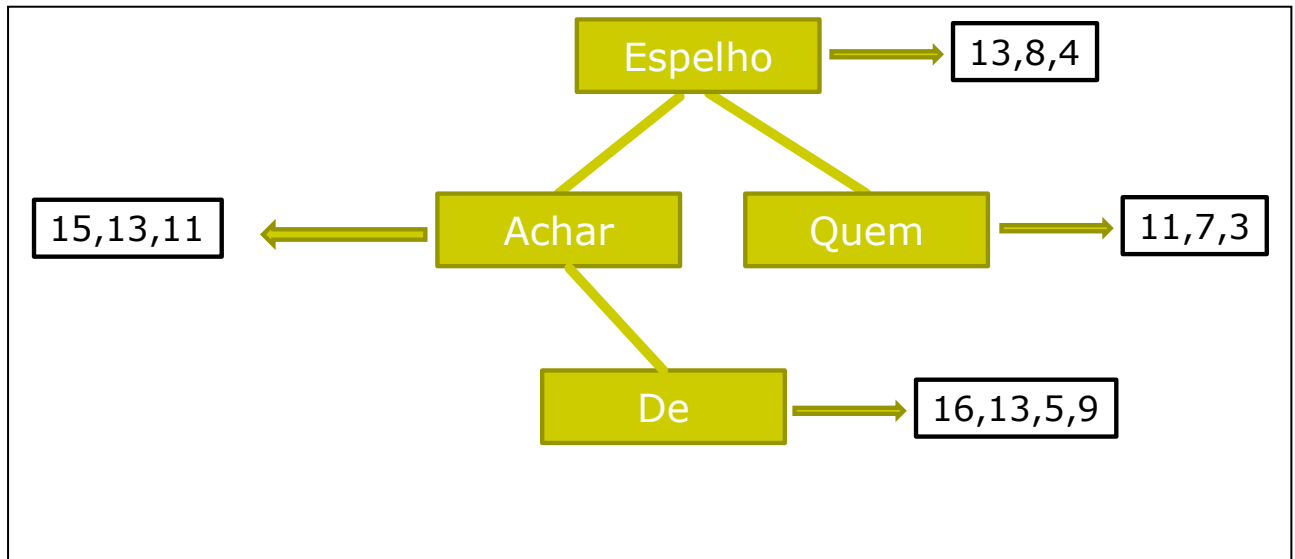


Figura 1: Árvore gerada

- Crie uma função auxiliar usada para extrair palavras de uma linha de texto. (1 ponto)
- Crie uma função para gerar o arquivo indexador a partir da árvore. Siga as dicas para criar o arquivo indexador=> (1,5 ponto)
  - Inicie com uma árvore de busca binária R vazia.
  - Para cada palavra lida do arquivo de palavras chaves C, verificar se existe no arquivo T
    - Se existir => insira em R um nó contendo essa palavra e sua respectiva lista de referências vazia.
    - Se não existir=> descartar a palavra.
  - Para cada linha lida do arquivo T, extraia as palavras dessa linha e atualize suas listas de referências em R, caso elas sejam palavras-chave. Para manter o controle de referências, use um contador de linhas, inicialmente igual a 1, e incremente-o a cada linha de texto lida.
- Crie uma função para exibição do índice remissivo gerado. (0,5 ponto)
- Faça um programa principal que teste todas as funções anteriores. (0,5 ponto)
- USE TAD para resolver esse problema. (0,5)