

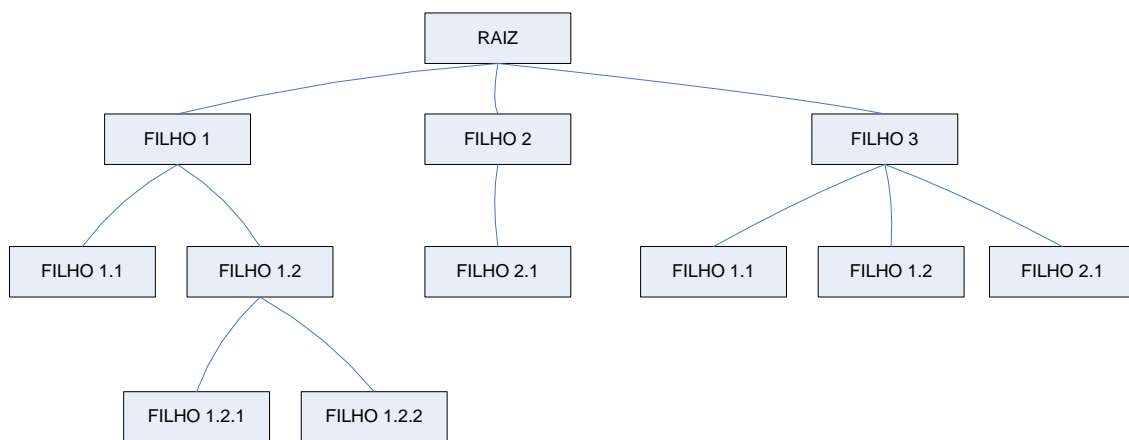


TAD ÁRVORE N-ÁRIA

-- TEORIA --

Introdução

A **Árvore N-Ária**, também conhecida como **Árvore de Estrutura Genérica** ou, dependendo de como seja implementada, **Árvore Hiperbólica**, é um tipo de árvore que possibilita que um dado nó tenha uma quantidade ilimitada de filhos.



Este tipo de árvore geralmente é utilizado para representar estruturas hierárquicas como organogramas empresariais, composição das partes de um equipamento, estruturas de categorias botânicas, animais etc. Essas estruturas podem ser de natureza estática, ou seja, determinado nó é filho de outro e não muda, como também para estruturas dinâmicas, onde há movimentação entre os nós (um filho muda de pai) como pessoas ou equipes de trabalho que são realocadas em uma estrutura de organograma de um projeto de grande porte.

Professor

Marcio Feitosa



Esse tipo de árvore pode ser armazenado em memória de várias formas. Vejamos duas delas:

Array de 2 colunas:

Neste array podemos armazenar na primeira coluna o identificador do pai e na segunda o identificador do filho.

Tabela 1

PAI	FILHO
RAIZ	FILHO 1
RAIZ	FILHO 2
RAIZ	FILHO 3
FILHO 1	FILHO 1.1
FILHO 1	FILHO 1.2
FILHO 2	FILHO 2.1
...	...

As funcionalidades de interesse seriam:

- identificar a raiz
 - percorre a coluna "pai" e verifica qual deles não aparece como filho.
- identificar filhos de um dado nó
 - percorre a coluna "pai", localiza as linhas em que o nó aparece como pai e lê a coluna "filho".
- identificar pai de um nó
 - percorre a coluna "filho", localiza o nó e lê na coluna "pai"
- identificar a profundidade de um nó (quantos níveis o nó está abaixo da raiz)
 - (a) identifica o raiz; (b) localiza o pai, a seguir o pai do pai e assim por diante até chegar ao nó raiz (vai contabilizando a cada pai encontrado)
- identificar a profundidade de um subárvore (dado um nó, quantos níveis tem abaixo)
 - (a) identifica os filhos; (b) para cada filho, identifica os filhos deste; (c) descarta os que não tem filhos e repete item "b" até nenhum nó ter mais filhos (vai contabilizando os níveis)
- localizar um elemento na árvore
 - pesquisar na coluna "filho" (se não for encontrado -> é o nó raiz)
- imprimir a árvore ou uma subárvore (dado um nó)
 - uma forma é criar uma lista ligada onde cada elemento tem o valor identificador do nó e o nível em que se encontra. Esta lista tem que estar na ordem da raiz para o nó mais profundo, para que, ao percorrê-la, se vá imprimindo (na tela) e aplicando as indentações de acordo com o nível do nó.

Professor Marcio Feitosa



Lista encadeada:

Neste modelo cada nó é representado por uma classe com os atributos de dados, um atributo indicativo do nível (0 é o nível raiz) e um ponteiro para o próximo nó.

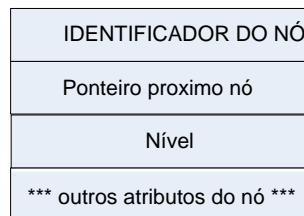


Figura 1



Figura 2

Observação: Caso seja necessário interagir com a árvore por diversas vezes em uma mesma instância, descendo e/ou subindo de nível, pode ser necessário implementar com lista duplamente ligada.

Para construir esta lista a partir da array de 2 colunas:

1. Identificar nó raiz (nó da coluna pai que não aparece na coluna filho). Criar o nó cabeça. Nível = 0.
2. Identificar os filhos do nó raiz. Criar uma lista auxiliar e instanciar cada nó filho, apontando para o próximo. Nível = 1. Inserir esta lista auxiliar após nó raiz.
3. Incrementar nível (0+1). Correr a lista do início localizando os nós deste próximo nível. Para cada nó (que estão na mesma sequência) localizar os filhos deste, criando uma lista auxiliar. Inserir esta lista após o nó pai.
4. Correr os filhos deste último nó à procura do próximo de mesmo nível. Achar os filhos. Construir uma lista auxiliar. Inserir a lista após o pai.
5. Assim por diante até que não sejam mais encontrados filhos.

Professor Marcio Feitosa



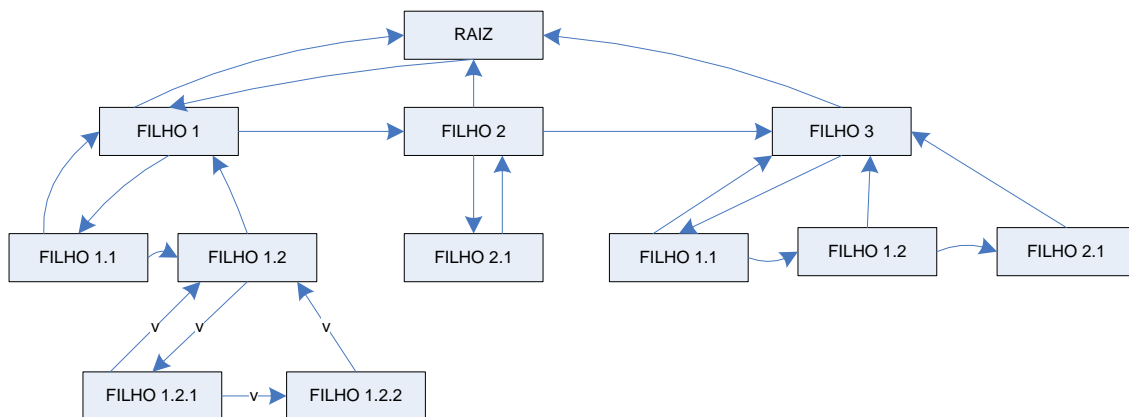
Grafo de listas:

Neste modelo, que se aproxima mais da chamada Árvore Hiperbólica, cada nó é representado por uma classe com 3 ponteiros: pai, filho e irmão.

IDENTIFICADOR DO NÓ
Ponteiro PAI
Ponteiro IRMÃO
Ponteiro FILHO
Nível
*** outros atributos do nó ***

Figura 3

Neste modelo o ponteiro "irmão" aponta para uma lista encadeada de irmãos (o pai aponta para um filho e do segundo em diante se conectam pelo ponteiro "irmão").



Não é condição obrigatória que se tenha o ponteiro "pai" para se acessar todos os nós da rede, mas sem este não há possibilidade de se fazer a navegação reversa (os algoritmos ficam bastante burocráticos, pois voltar a um nível anterior obrigaria a reiniciar pelo nó raiz).

-O-O-O-O-O-O-O-O-O-O-O-