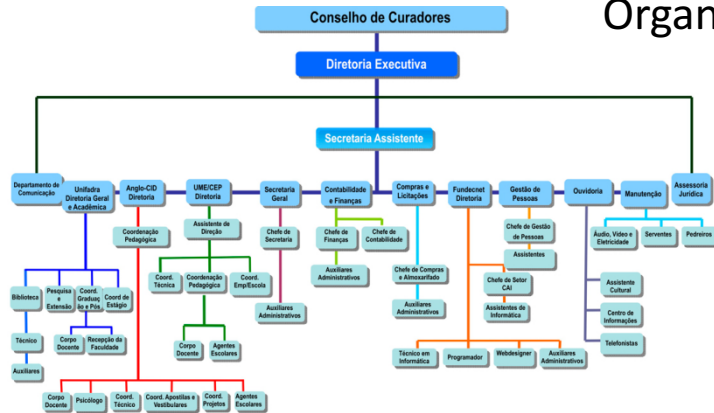


Estrutura de Dados e Algoritmos

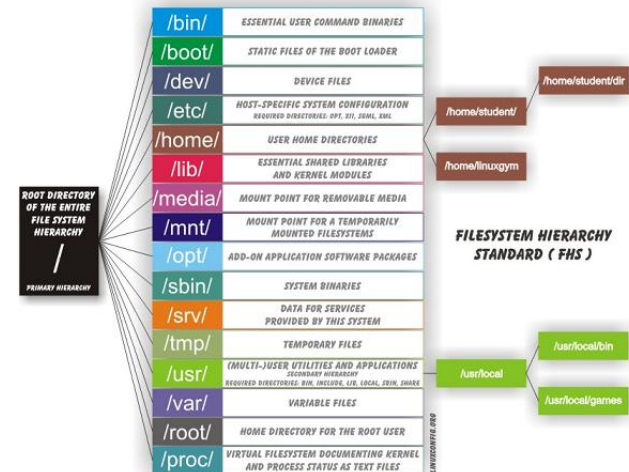
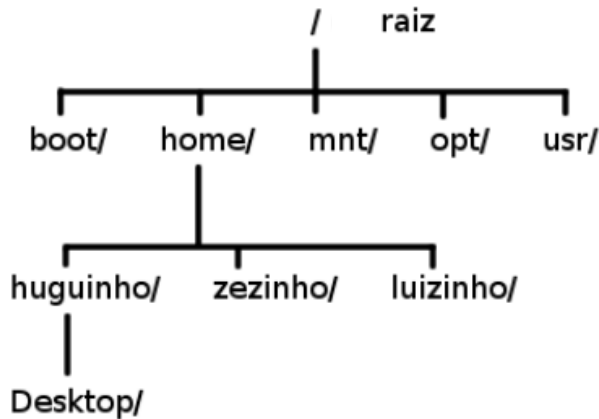
Árvores

- Importância de estruturas unidimensionais ou lineares (vetores e listas) é inegável.
- Contudo, elas não são adequadas para representar dados que devem ser dispostos de maneira **hierárquica**.
- Da mesma forma que listas, árvores são estruturas de dados que caracterizam uma relação entre seus dados (denominados nós). Neste caso, hierárquica.

Organograma



Árvores de diretórios

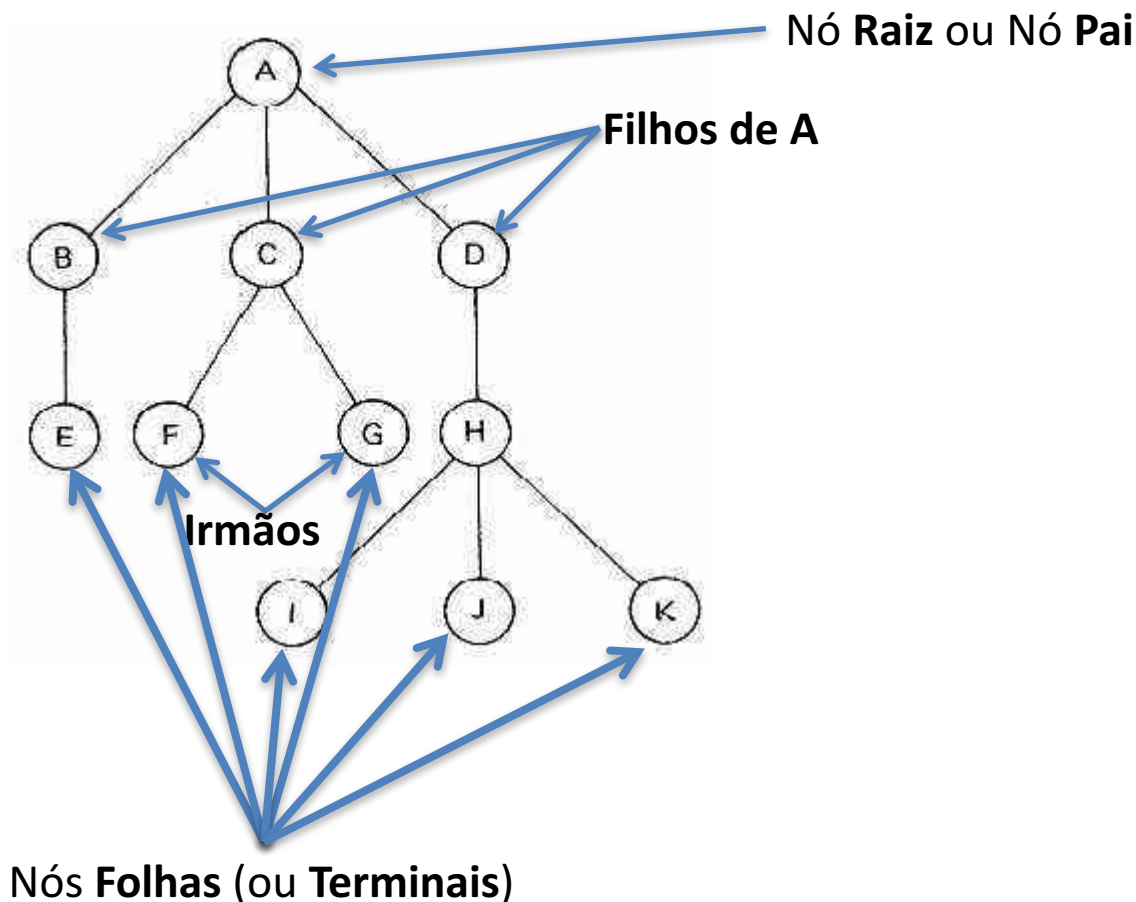


- Terminologias

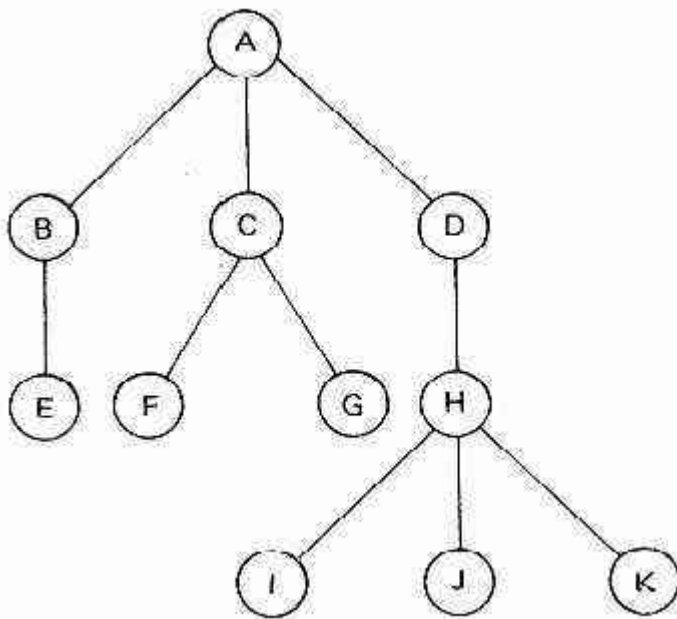
1. **Nó Raiz** ou **Nó Pai**: Cada nó de uma árvore é a raiz/pai de uma subárvore;
2. **Nó Filho**: As raízes das subárvores de um nó P são ditos nós filhos de P;
3. **Nós Irmãos**: Os filhos de um mesmo nó P são ditos nós irmãos;
4. **Grau**: A quantidade de subárvores de um nó define seu grau;
5. **Nó Folha** (ou **Terminal**): Nós com grau igual a zero;
6. **Nível**: É relativo ao posicionamento hierárquico de um nó dentro da árvore (sua distância em relação à raiz). A raiz tem nível zero, seus filhos tem nível 1, o filho de seus filhos tem nível 2 e assim sucessivamente;
7. **Altura**: A altura de uma árvore é definida como sendo o seu nível mais alto.
8. **Floresta**: Conjunto de zero ou mais árvores disjuntas.

- Formalmente, uma árvore é um conjunto finito T de um ou mais nós, tais que:
 - Existe um nó raiz R ;
 - Os demais nós formam $m \geq 0$ conjuntos disjuntos S_1, \dots, S_m , onde cada elemento é uma subárvore de R .

- Terminologias:

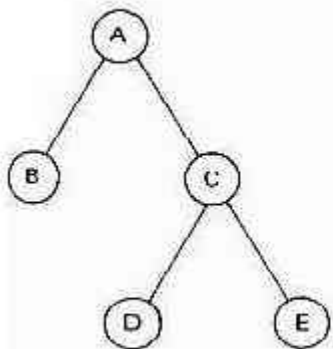


- Terminologias:

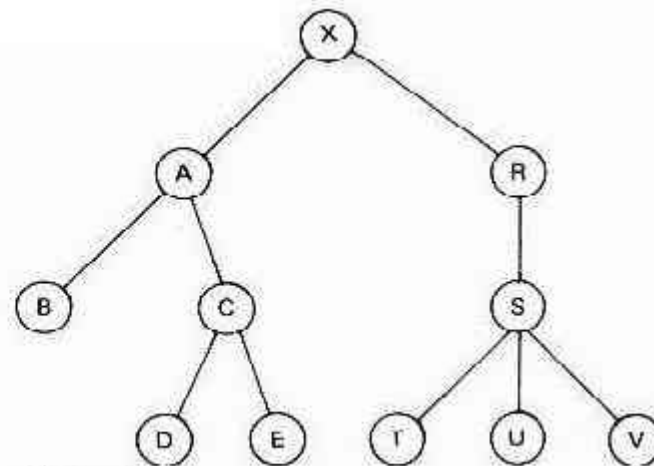


Nodo	Grau	Nível	Observações
A	3	0	Raiz da árvore
B	1	1	
C	2	1	
D	1	1	
E	0	2	Nó terminal (ou folha)
F	0	2	Nó terminal (ou folha)
G	0	2	Nó terminal (ou folha)
H	3	2	
I	0	3	Nó terminal (ou folha)
J	0	3	Nó terminal (ou folha)
K	0	3	Nó terminal (ou folha)

- Terminologias:

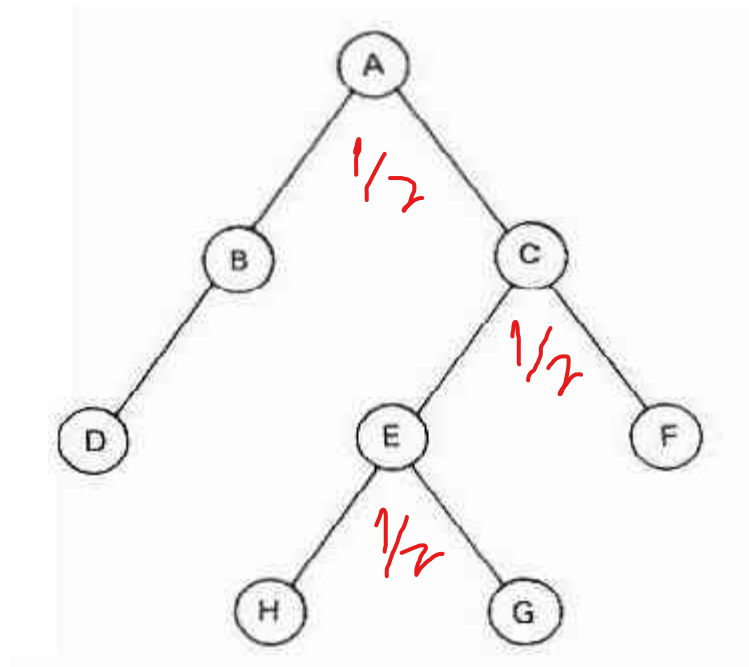


a) Uma floresta de duas árvores



b) Uma árvore

- Árvores binárias são estruturas do tipo árvore, onde o grau de cada nó é menor ou igual a dois.
- No caso de árvores binárias, distingue-se as subárvores de um nó em: subárvore da esquerda e subárvore da direita. Assim, se o grau de um nó é igual a 1, deve-se especificar se sua subárvore é a da esquerda ou da direita.
- Uma árvore binária pode ser vazia. Isto é, não possuir nenhum nó.



- Árvore X Árvore Binária:



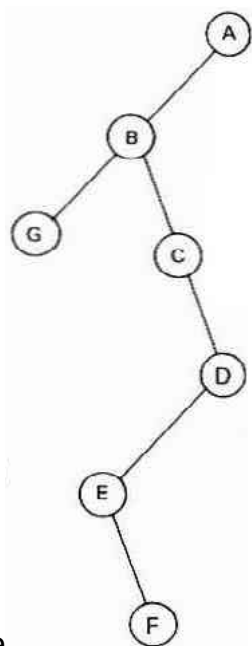
a) Duas árvores iguais



b) Duas árvores binárias diferentes.

- Implementações:
 - Alocação Estática (Vetores);
 - Alocação Dinâmica.

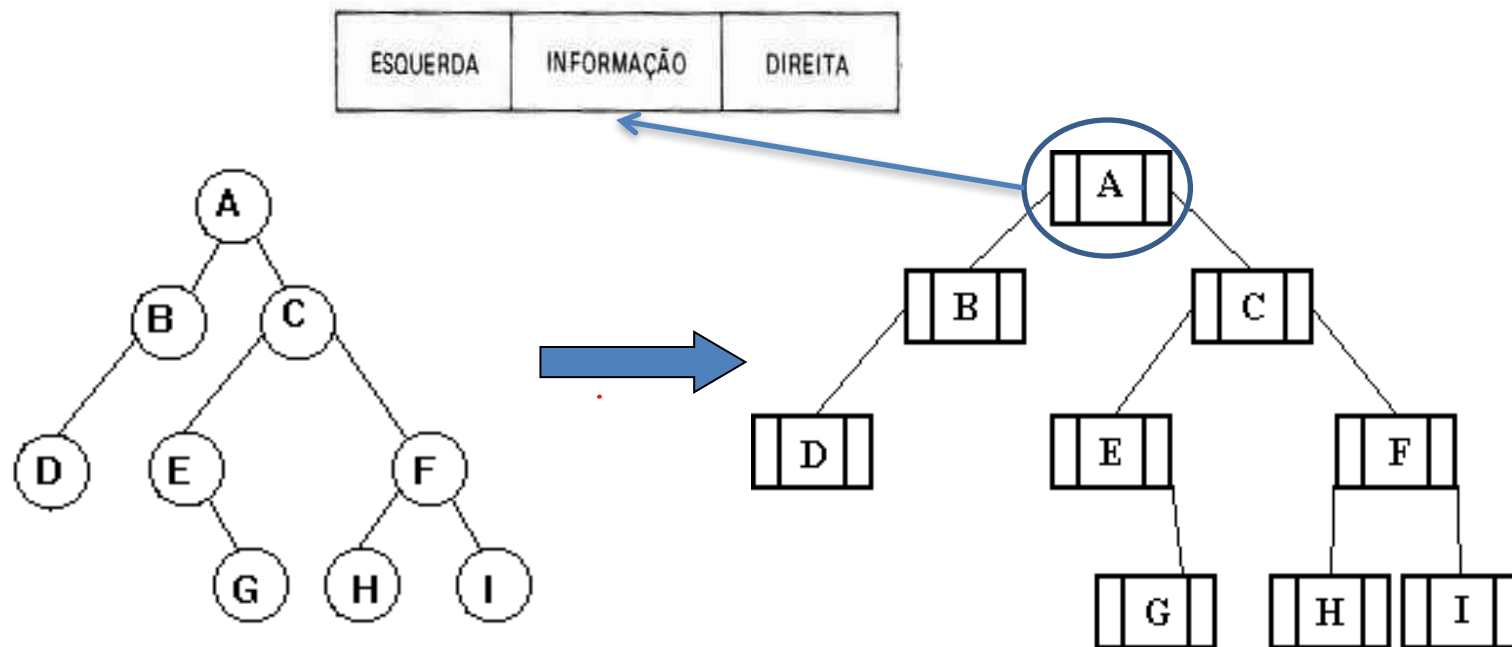
- Alocação Estática (Vetores):
 - Árvores são armazenadas sequencialmente na memória em uma estrutura de vetor.
 - Exemplo:



ABG..C.DE.F....

- Alocação Estática (Vetores):
 - Formato de representação dificulta manipulação da árvore (percorrimento, inclusões e exclusões);
 - Indicada para o armazenamento permanente.

- Alocação Dinâmica:
 - Área de memória (para os Nós) é alocada sob demanda:



- Alocação Dinâmica:

```
#ifndef TNOBIN_H
#define TNOBIN_H
struct TNoBin
{
    int info;
    TNoBin *Esquerda;
    TNoBin *Direita;
};
#endif
```