

Árvores

O que é uma
árvore?

Estruturas lineares:

- Listas
- Pilhas
- Filas
- Listas encadeadas circulares
- Listas duplamente encadeadas

Estruturas não-lineares

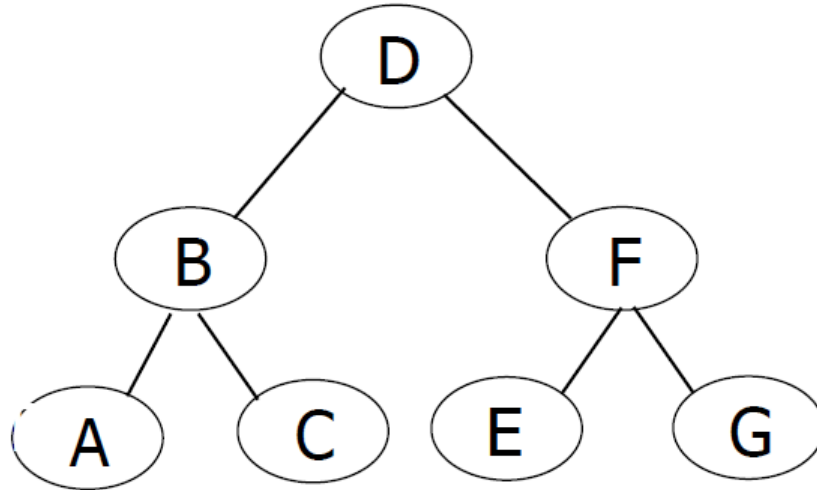
- Estrutura em grafos
- Estrutura em árvores



Hyphaene Compressa - Doum Palm

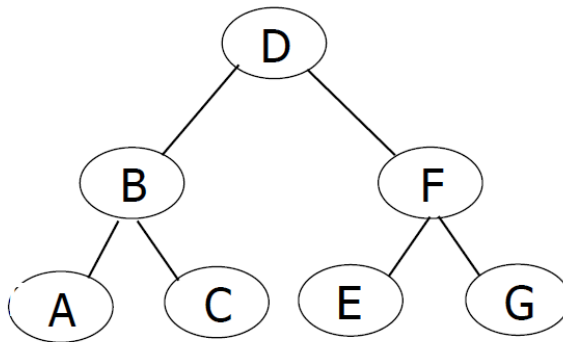
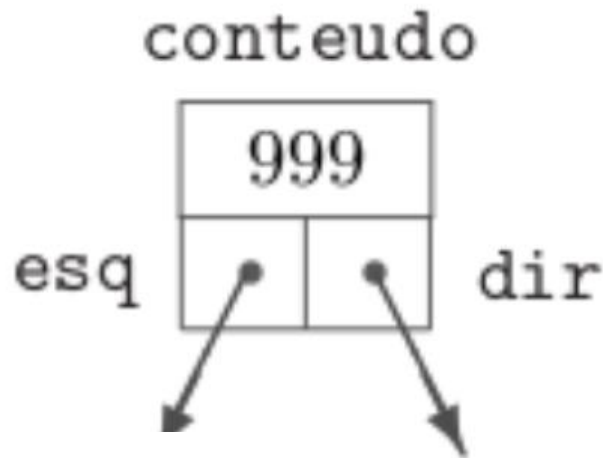
© Shlomit Pinter

Estrutura em árvore: organização dos dados de forma não-linear, mantendo um relacionamento hierárquico entre os elementos



Nós

```
typedef struct reg
{
    int conteudo; //
    conteúdo
    noh *esq;
    noh *dir;
} noh; // nó
```

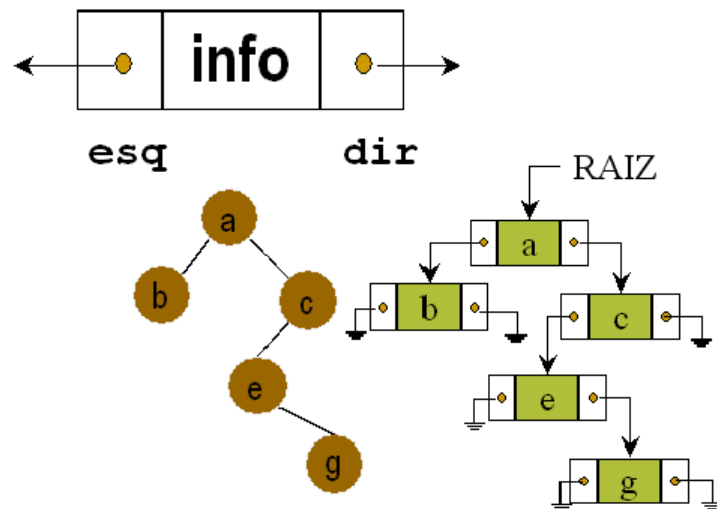


- Para qualquer árvore, cada nó é do tipo

```
typedef struct {  
    int chave;  
    char valor;  
} INFO;
```

```
typedef struct NO {  
    INFO info;  
    struct NO *fesq;  
    struct NO *fdir;  
} NO;
```

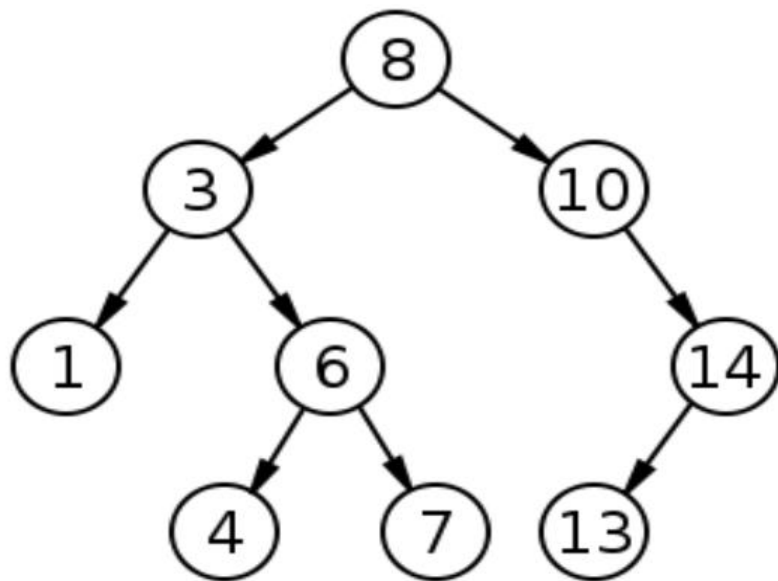
```
typedef struct {  
    NO *raiz;  
} ARVORE_BINARIA;
```



Motivação/Vantagens:

- Representatividade no relacionamento entre os dados
- Favorece a extração de informação de forma eficiente
- Quem são os filhos de Maria?
- Onde está o capítulo de estrutura em árvores?
- Quem é o diretor da seção de financeiro?
- -> **Busca** <-

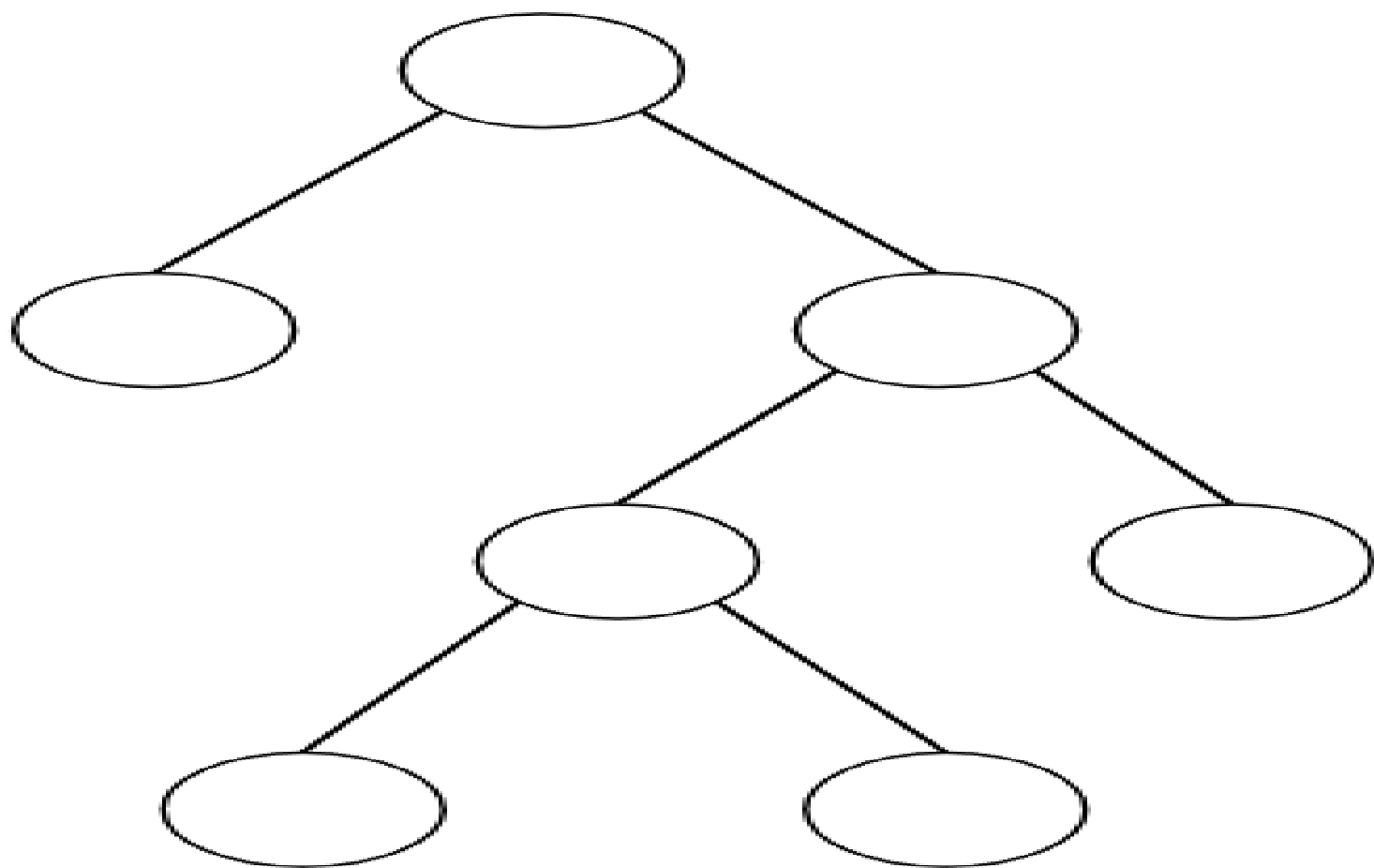
Exemplo:



Variedade de
árvores

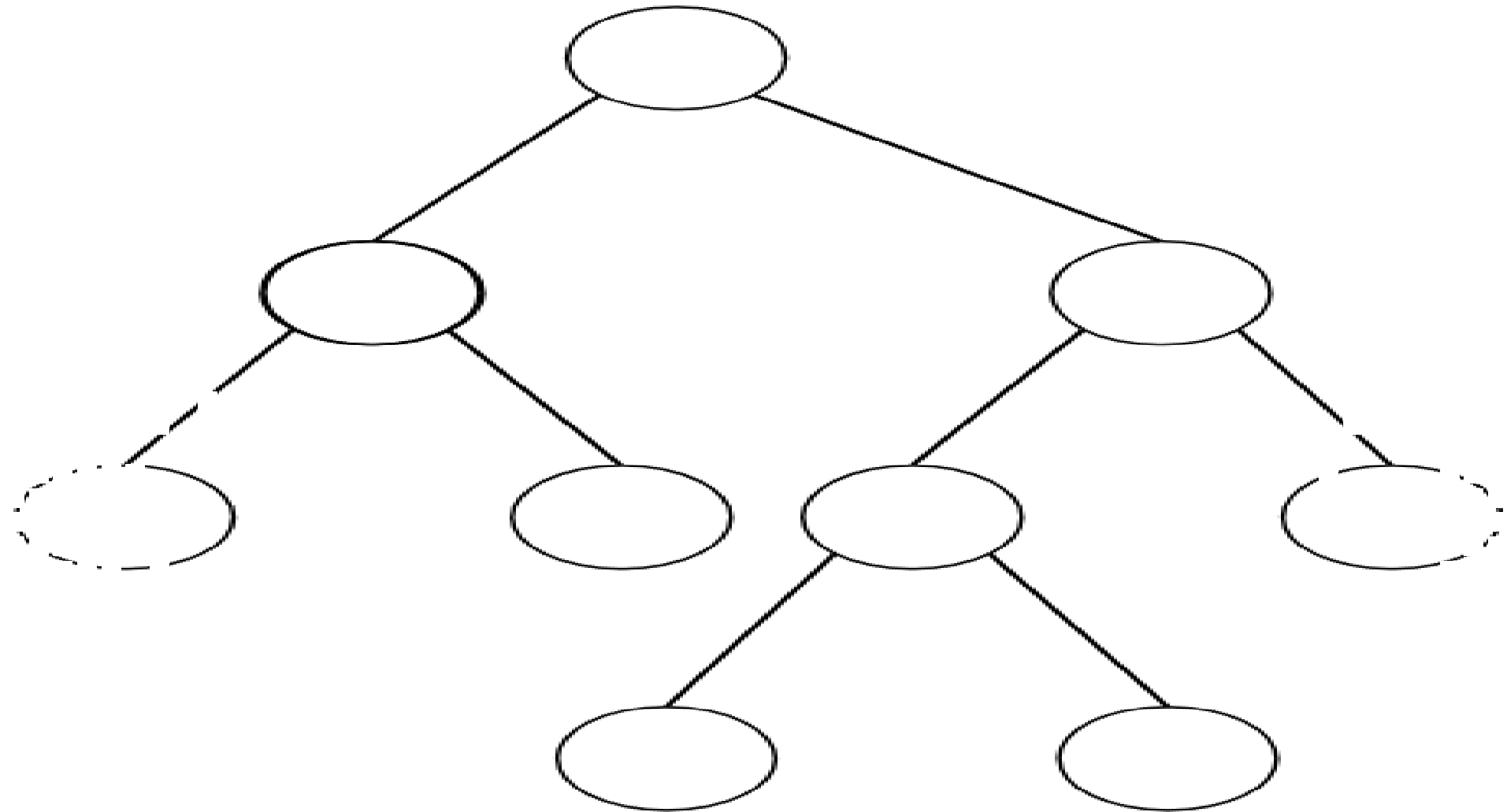
Árvore estritamente binária

- Todos os nós possuem 0 ou 2 “filhos”
- Nós interiores sempre possuem 2 “filhos”



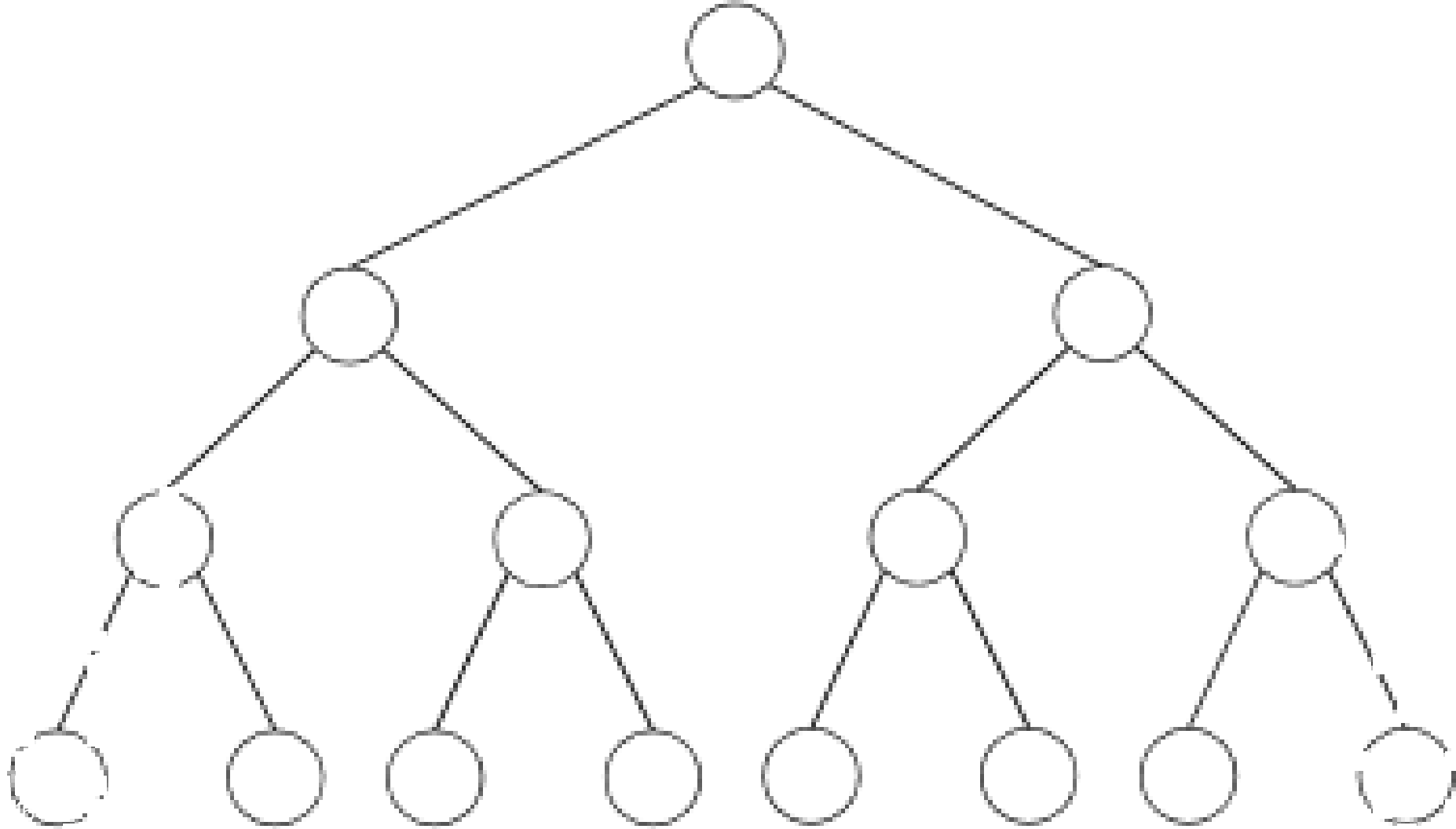
Árvore completamente binária

- Se o nó V tem alguma subárvore vazia, então V está no último ou penúltimo nível da árvore



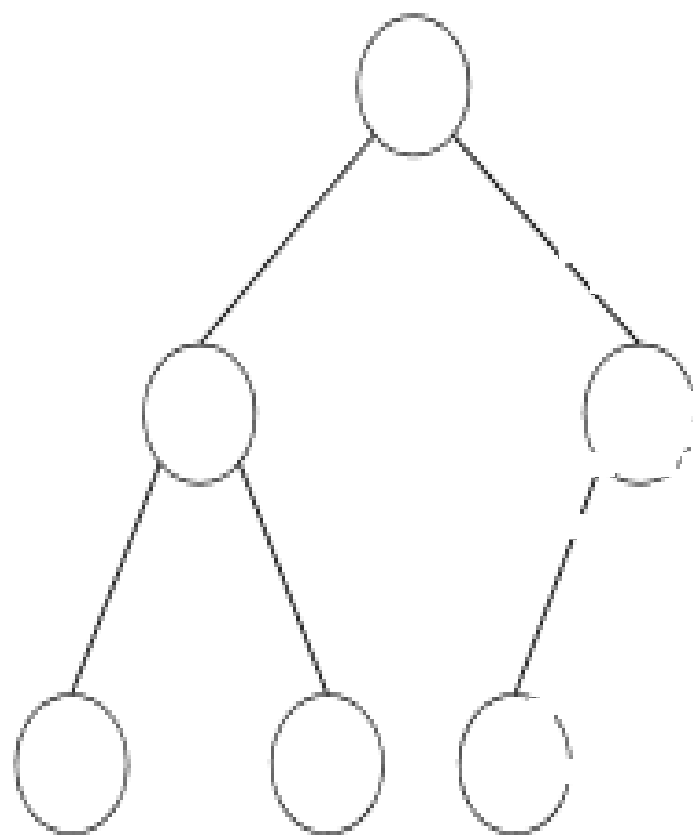
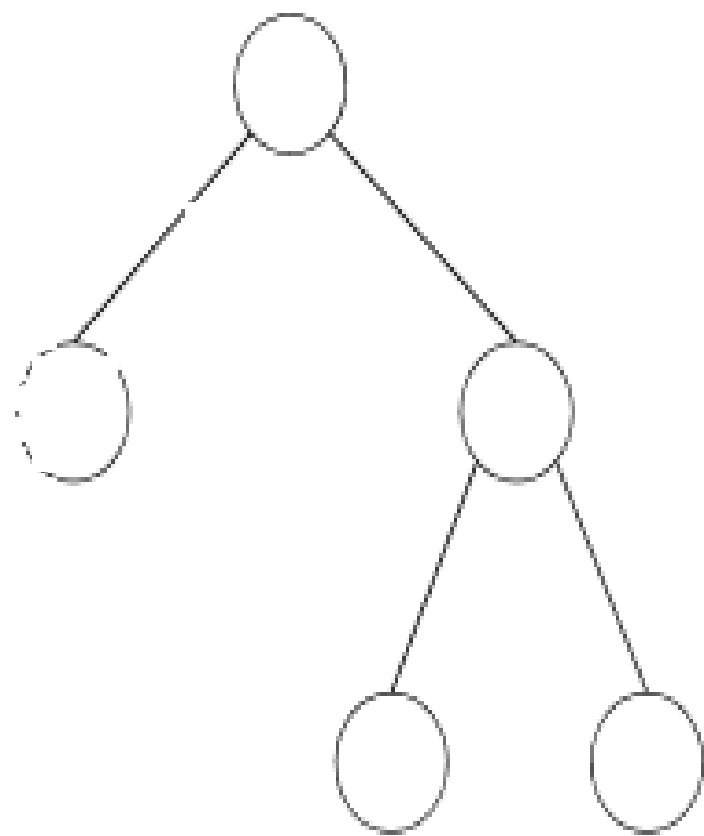
Árvore binária cheia

- Se o nó V tem alguma subárvore vazia, então V está no último nível da árvore
- Uma árvore binária cheia é uma árvore binária completa e estritamente binária



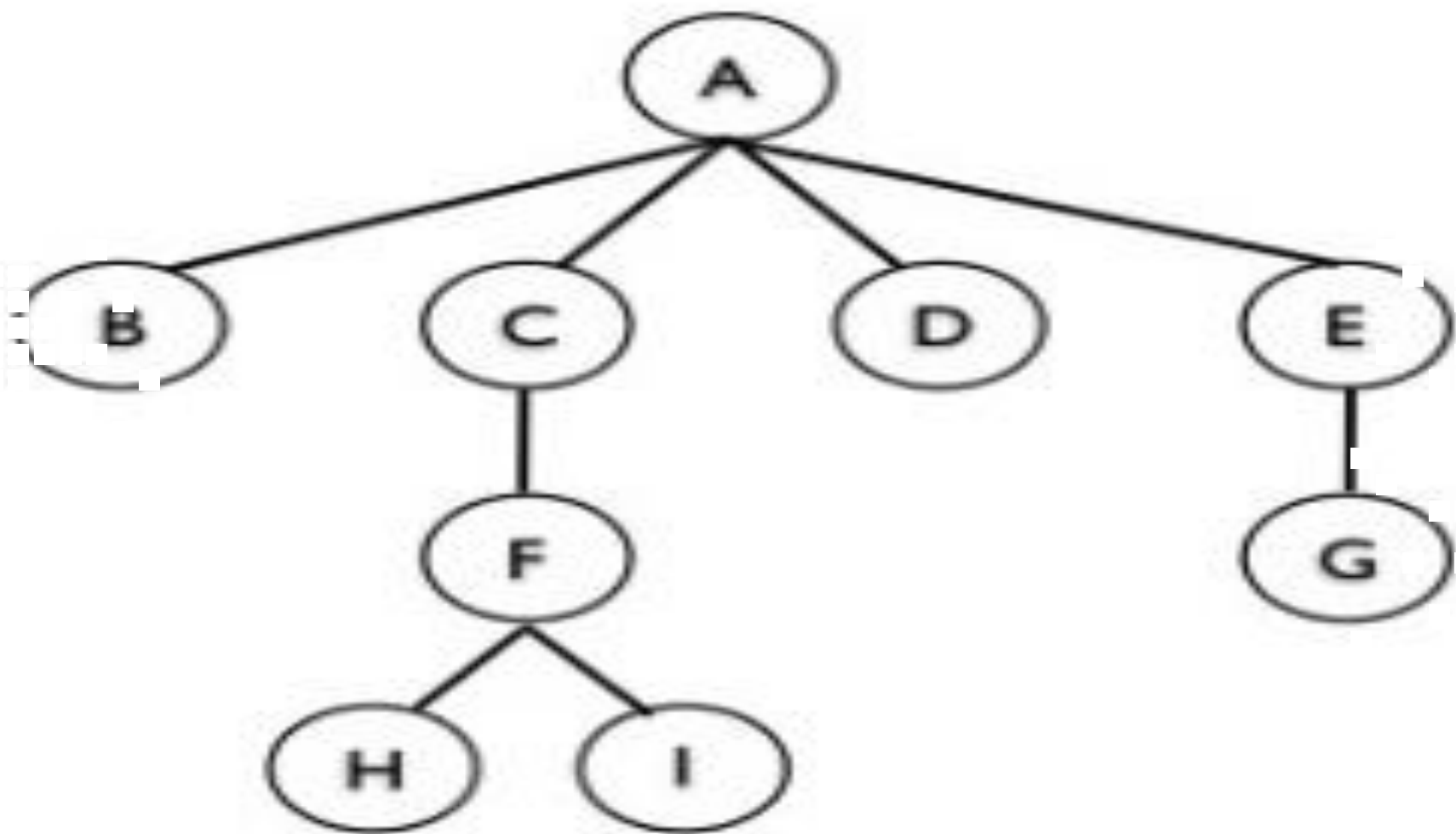
Árvore binária balanceada

- Para cada nó, as alturas de suas duas subárvores diferem de, no máximo, 1



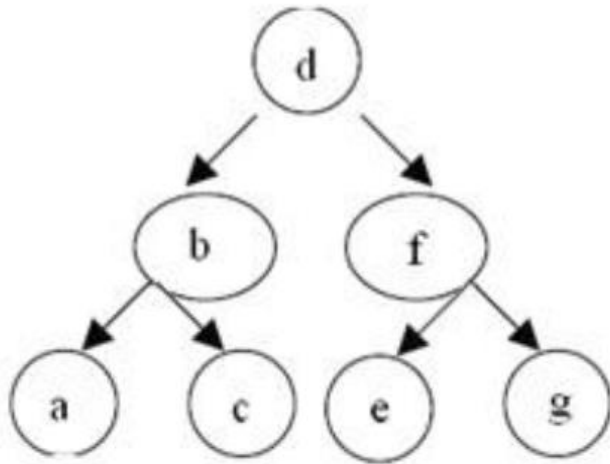
Árvore n-ária

- Cada nó pode conter n “filhos”

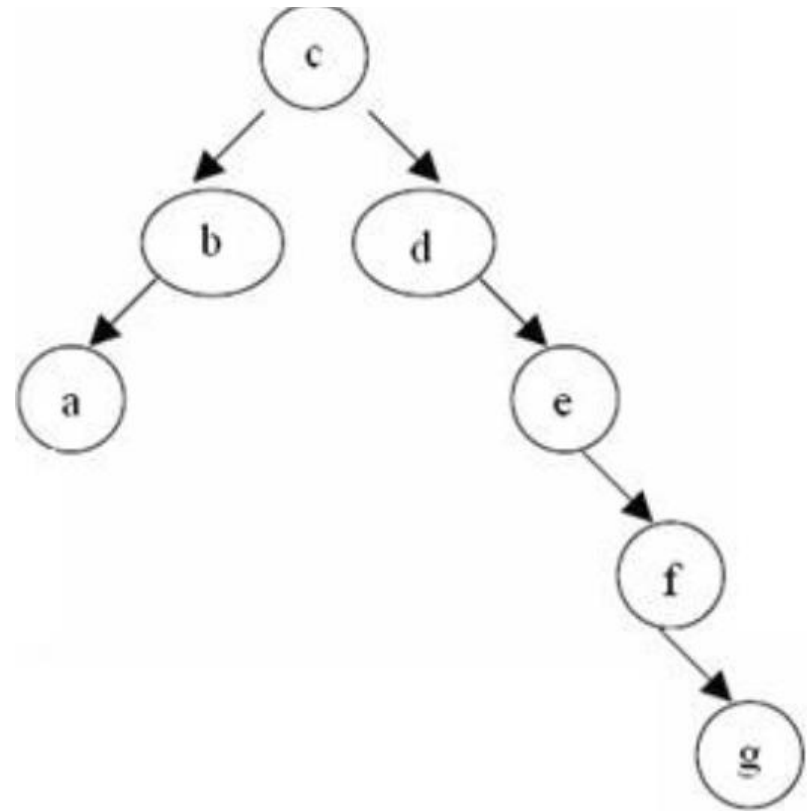


Verificação de balanceamento

Árvores Balanceadas



Árvore Binária Balanceada

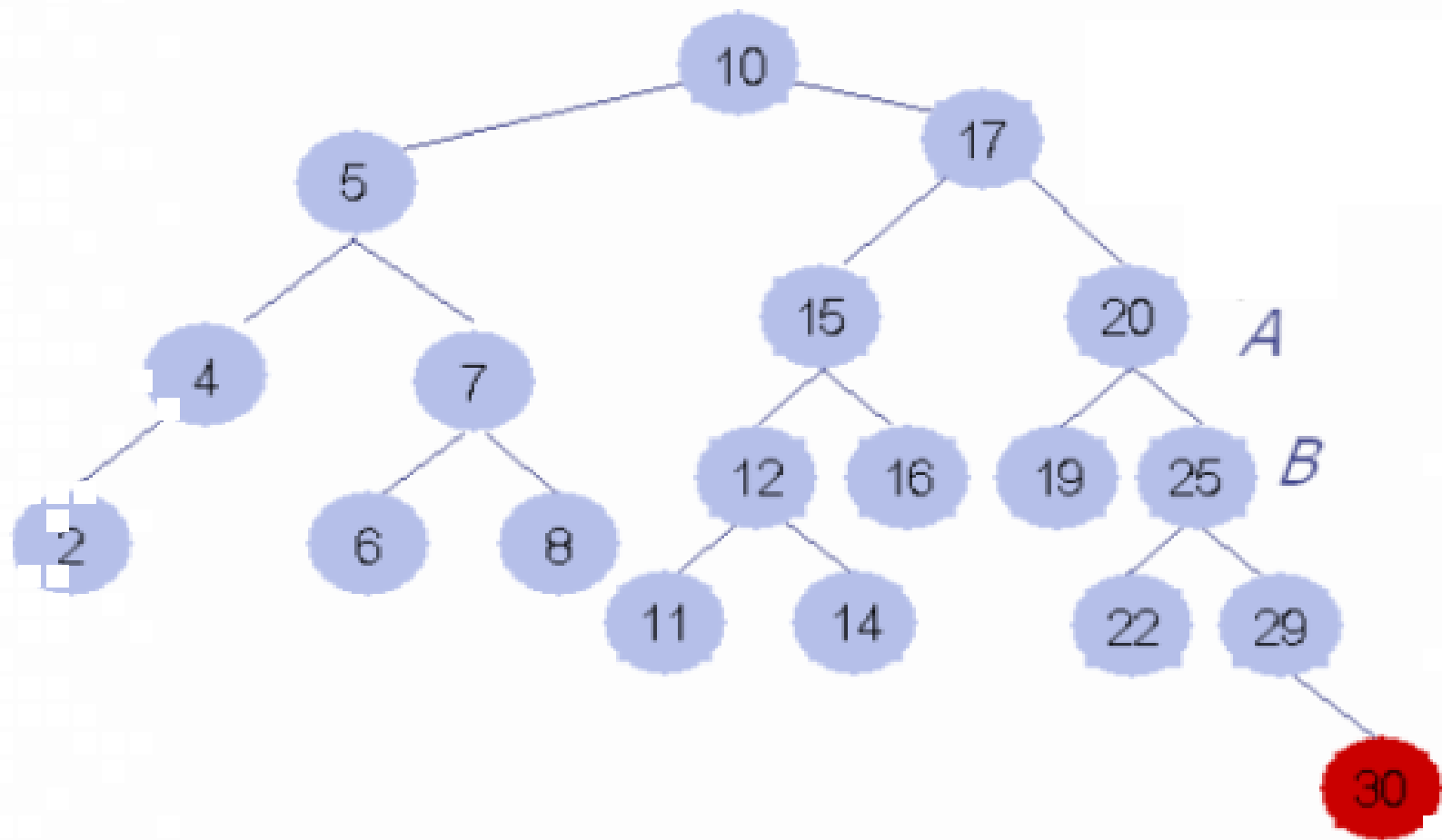


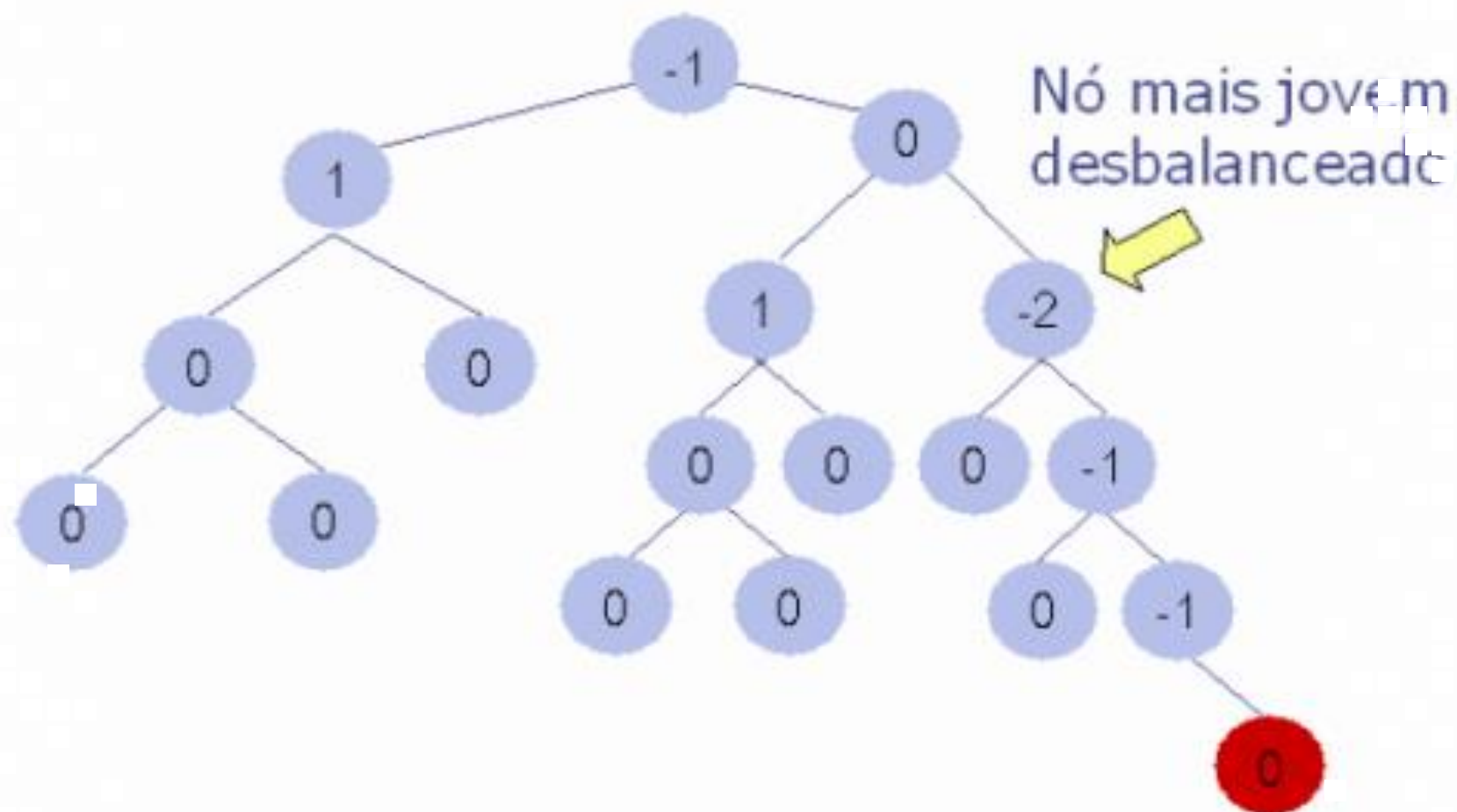
Árvore Binária Degenerada

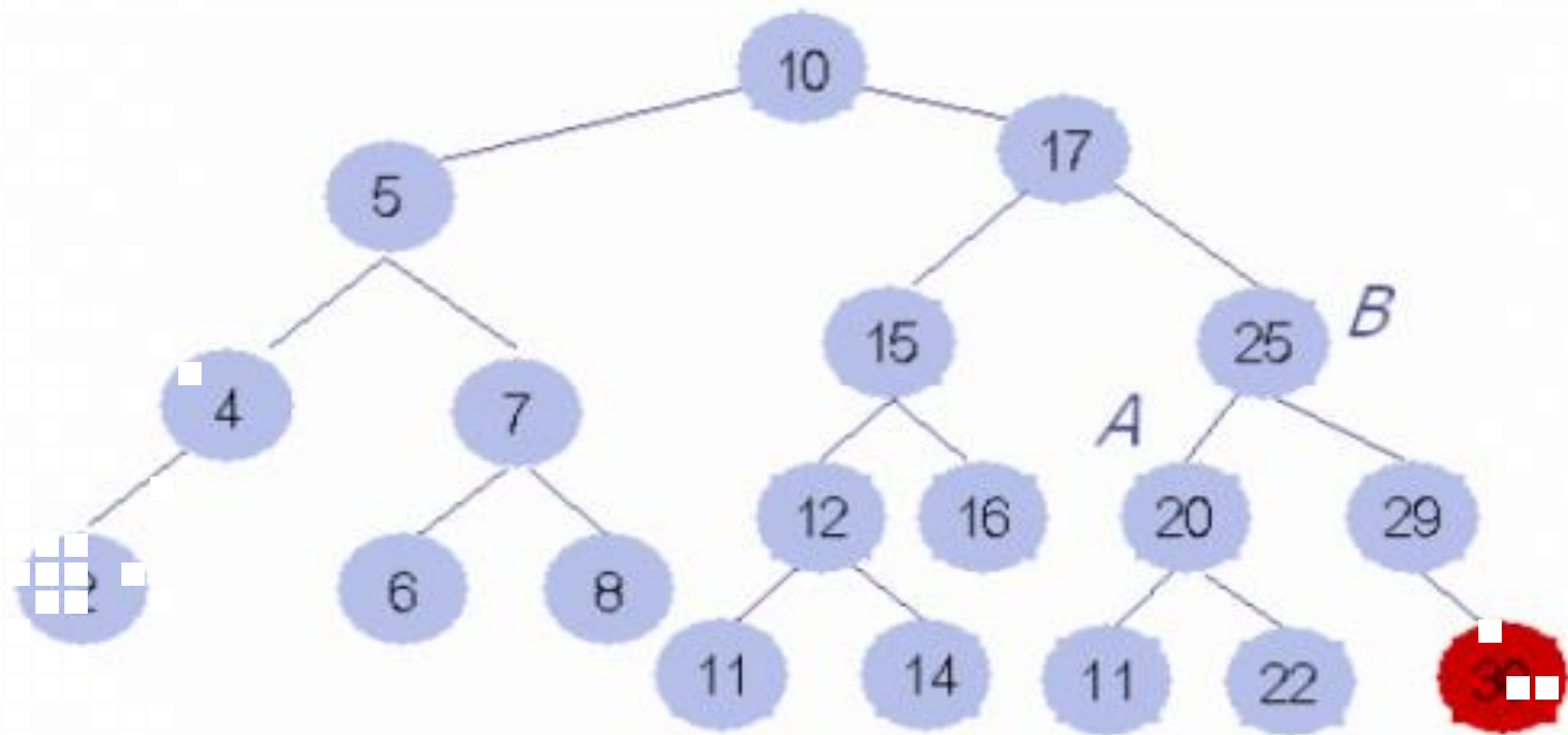
Em uma árvore não balanceada com 10.000 nós, são necessários (no pior cenário) **5.000 comparações** para efetuar uma busca, já uma árvore binária balanceada **este número cai para 14.**

Mecanismos de balanceamento

Rotação à esquerda

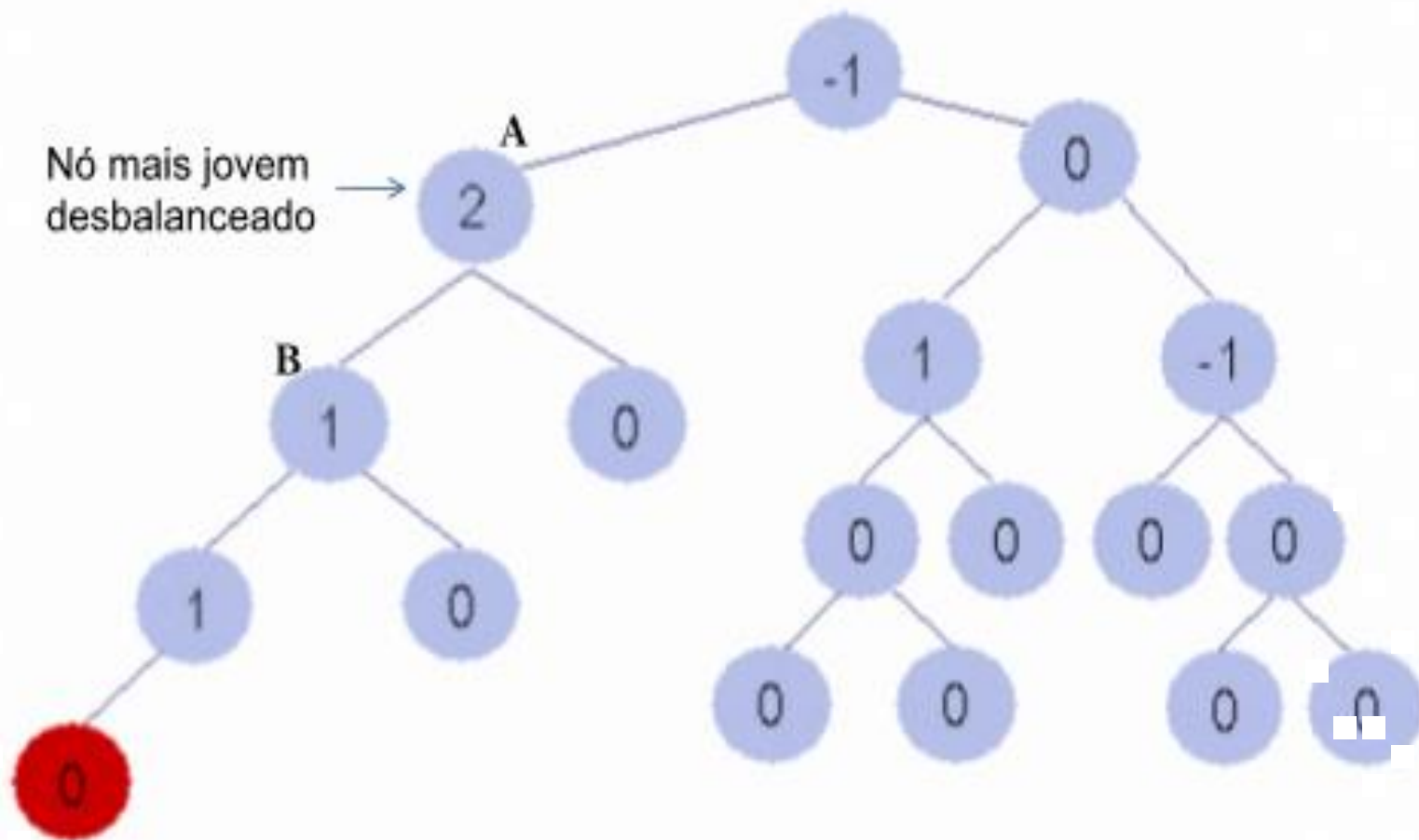






Rotação à direita

Nó mais jovem
desbalanceado



Exercício

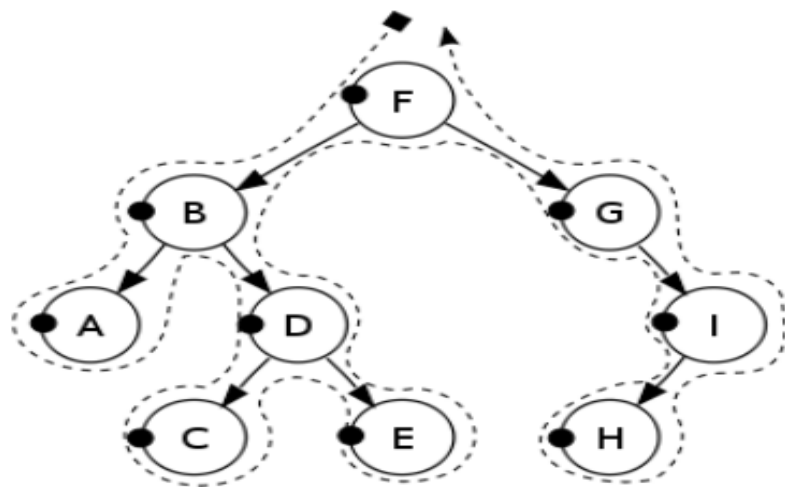
Vocês terão que criar uma árvore e um menu com as seguintes opções:

1) Inserir

2) Remover

3) Mostrar

Exemplo de leitura Pré, Em e Pós Ordem da árvore:



PreOrdem (RED): F,B,A,D,C,E,G,I,H

InOrdem (ERD): A,B,C,D,E,F,G,H,I

PosOrdem (EDR): A,C,E,D,B,H,I,G,F
