UFSC-CTC-INE

INE5384 - Estruturas de Dados

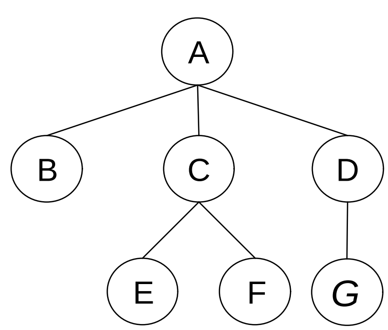
Árvores

Prof. Ronaldo S. Mello

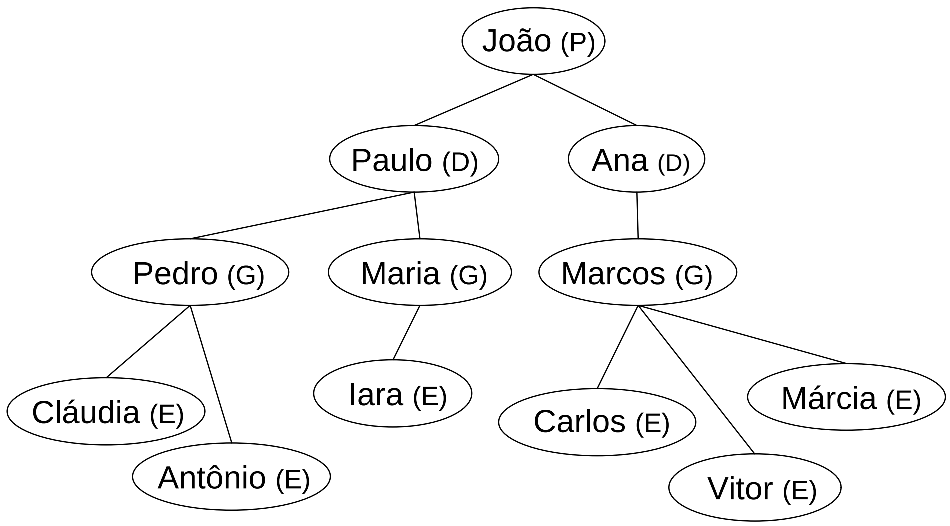
2002/2

Árvore

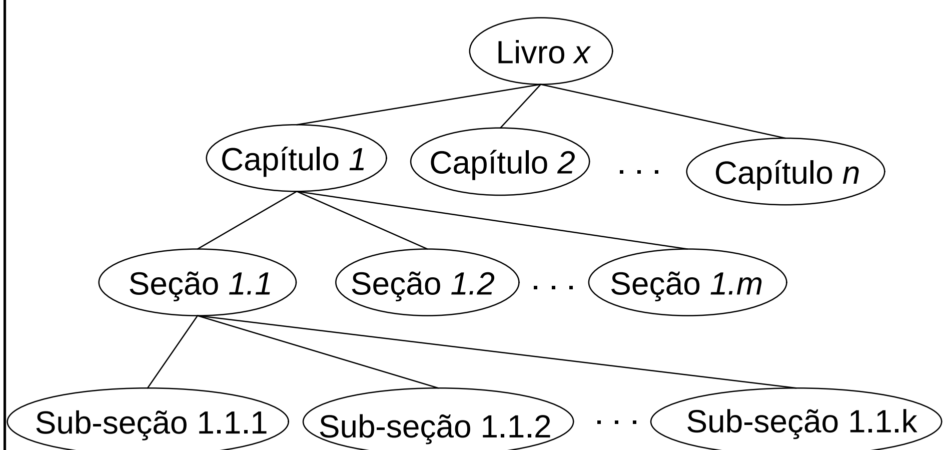
- Uma árvore é uma estrutura que mantém uma relação de hierarquia ou composição entre os dados



Exemplo



Exemplo



Definição de Árvore

Uma árvore *A* é uma ED tal que:

– existe um nodo raiz *r*

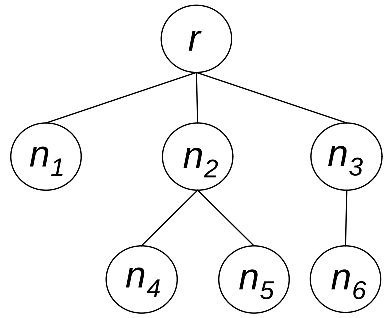
– existem outros nodos $n_1, n_2, ..., n_m$, $m >= 0$, associados a *r* que são raízes de subárvores disjuntas $A_1, A_2, ..., A_m$

Definição de Árvore

Uma árvore *A* é uma ED tal que:

– existe um nodo raiz *r*

– existem outros nodos $n_1, n_2, ..., n_m$, $m >= 0$, associados a *r* que são raízes de subárvores disjuntas $A_1, A_2, ..., A_m$

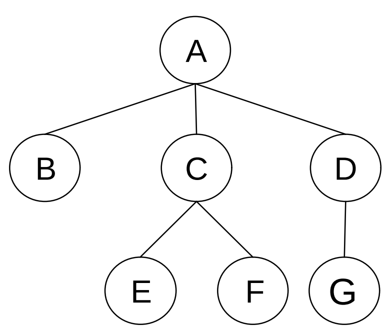


Terminologia

- Grau de um nodo
- Nodo folha e nodo interno
- Caminho
- Profundidade ou Nível de um nodo
- Nodo pai, nodo filho e nodo irmão
- Nodo ancestral e nodo descendente
- Altura de uma árvore
- Árvore ordenada e árvore orientada
- Floresta

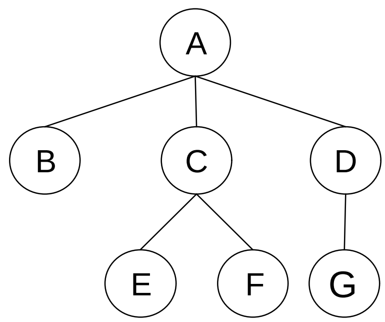
Grau de um Nodo

- Denotado por *g(n)*
- É o número de subárvores de um nodo



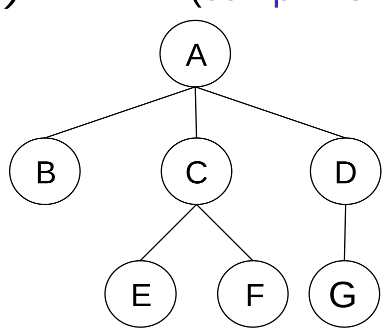
Nodo Folha e Nodo Interno

- *n* é um nodo folha se $g(n) = 0$
- *m* é um nodo interno se $g(m) > 0$



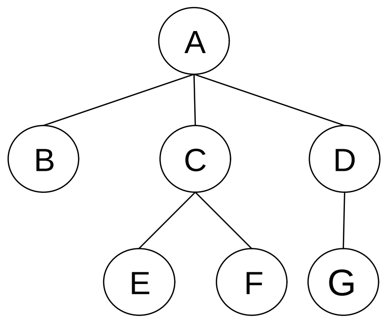
Caminho

- Um caminho *C* em uma árvore é uma seqüência de nodos relacionados na forma $C = n_1, n_2, ..., n_m$, $m > 0$, sendo n_i hierarquicamente superior a n_{i+1}
- *lenght(C)* $= m - 1$ (comprimento de *C*)



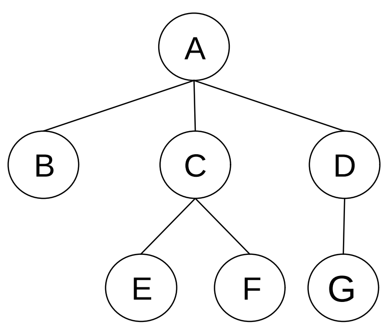
Profundidade de um Nodo

- Denotado por $p(n)$
- $p(n) = lenght(C)$, sendo $C = r, ..., n$



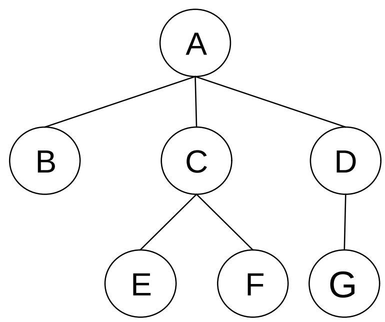
Nodo Pai e Nodo Filho

- dados 2 nodos n_i e n_j, se existe um caminho a partir da raiz que passa por n_i e n_j e $p(n_j) = p(n_i) + 1$, então n_i é nodo pai de n_j e, conseqüentemente, n_j é nodo filho de n_i



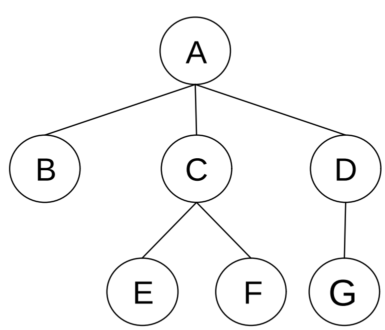
Nodo Irmão

- Dois nodos n_i *e* n_j são irmãos se possuem o mesmo nodo pai



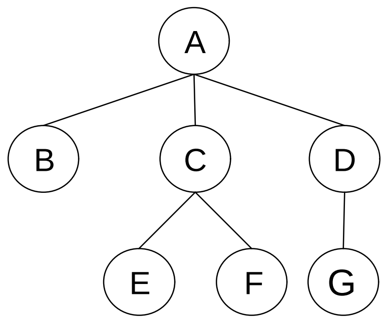
Nodos Ancestral e Descendente

- dados 2 nodos n_i *e* n_j, se existe um caminho a partir da raiz que passa por n_i e n_j e $p(n_j) <= p(n_i)$, então n_i é ancestral de n_j e, conseqüentemente, n_j é descendente de n_i



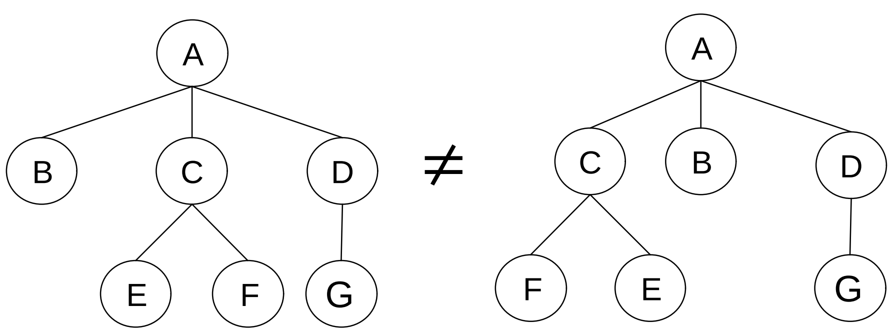
Altura de uma Árvore

- Denotado por *h(A)*
- $h(A) = \max(p(n_i))$, para $n_i \in A$



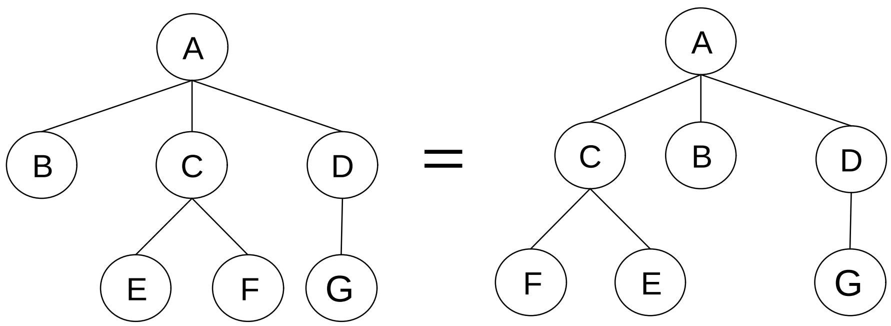
Árvore Ordenada

- Árvore na qual a ordem das subárvores é significativa



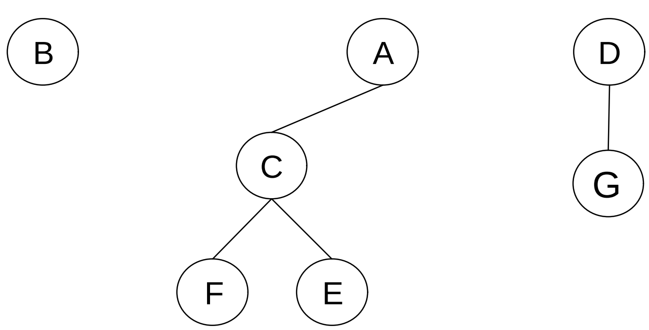
Árvore Orientada

- Árvore na qual a ordem das subárvores não é significativa



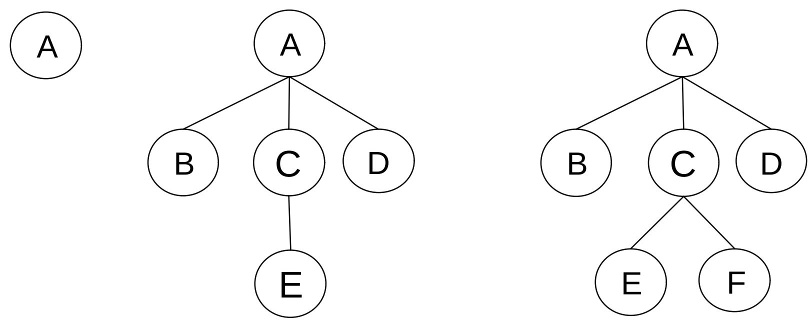
Floresta

- Uma floresta $F = \{A_1, A_2, ..., A_n\}$ é um conjunto de $n >= 0$ árvores disjuntas



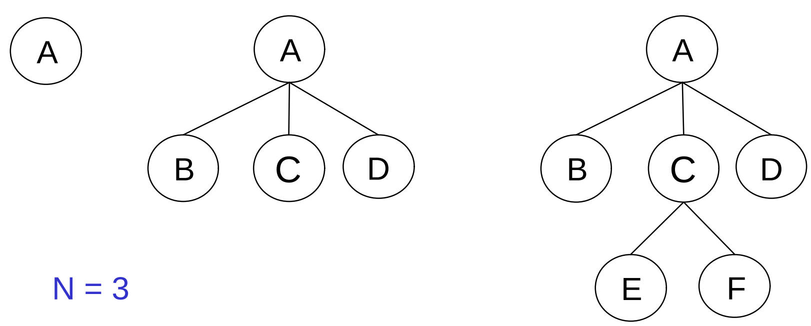
Árvore N-ária

- Uma árvore *A* é dita N-ária se
  $\forall\, n_i \in A\ (g(n_i) \in [0,N])$
- Exemplo: N = 3



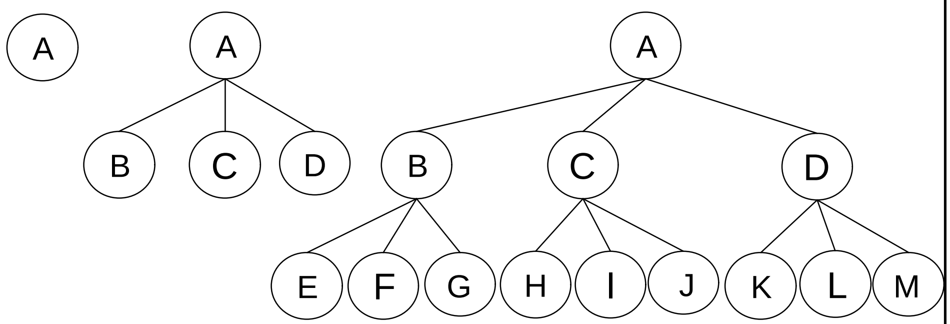
Propriedade (P1)

- "Em uma árvore N-ária *A* com altura *h(A)*, o número máximo de nodos folha é $N^{h(A)}$"



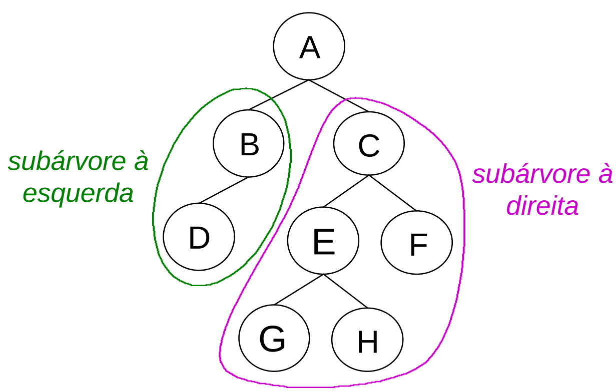
Árvore Cheia

- Árvore N-ária *A* cujo número de nodos folha é $N^{h(A)}$"



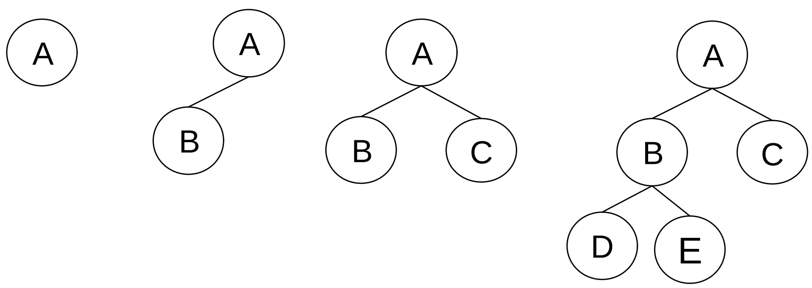
Árvore Binária

- Uma árvore binária é uma árvore N-ária com N = 2
- Uma árvore binária é ordenada por *default*



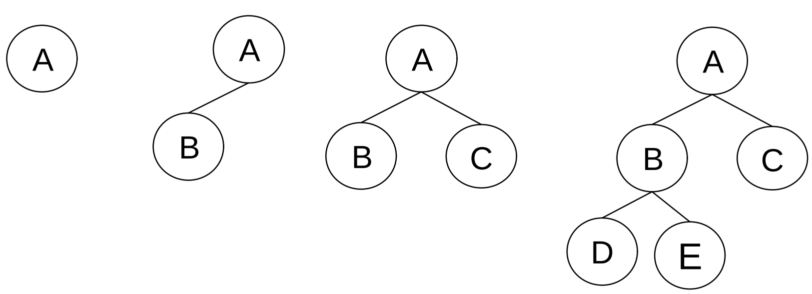
Propriedade (P2)

- "O número máximo de nodos em uma árvore binária *A* com altura *h(A)* é $2^{h(a)+1}-1$"



Propriedade (P3)

- "A altura *h(A)* de uma árvore binária *A* com *n* nodos é $\lfloor log_2\, n \rfloor$"

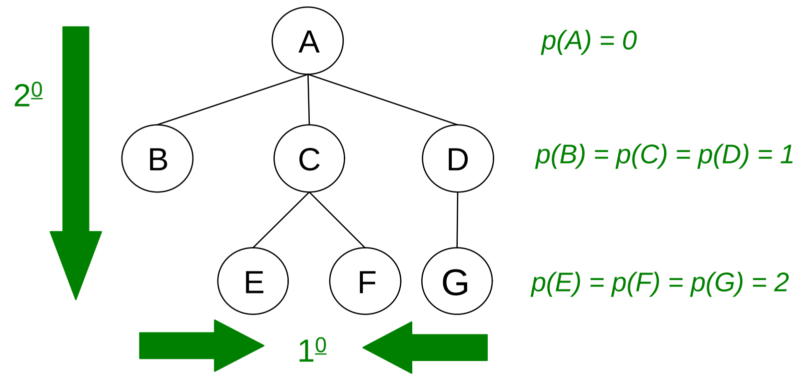


Caminhamento em Árvores

- Métodos de pesquisa em uma árvore para fins de consulta e/ou atualização de dados
- Existem dois métodos:
  - busca em largura *(breadth-first-traversal)*
  - busca em profundidade *(depth-first-traversal)*

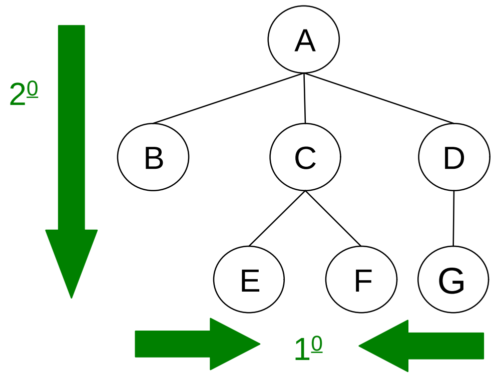
Busca em Largura

- Percorre a árvore por ordem de profundidade de nodo



Busca em Largura

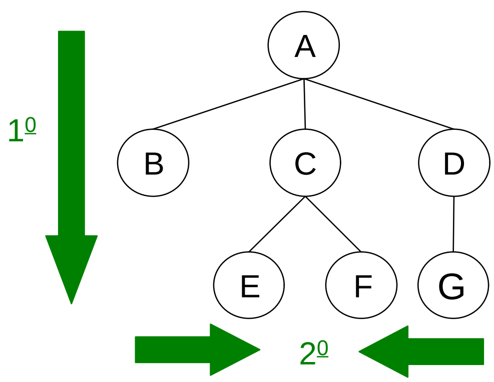
- Percorre a árvore por ordem de profundidade de nodo



busca pela ESQ:
A-B-C-D-E-F-G

Busca em Profundidade

- Percorre a árvore por ordem de sub-árvore (recursivamente)

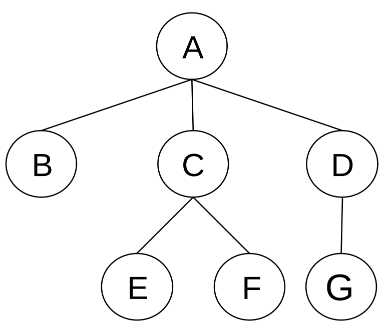


Busca em Profundidade

- Existem 3 tipos de pesquisa:
  - pré-ordem (ou pré-fixada)
  - pós-ordem (ou pós-fixada)
  - in-orem (ou in-fixada ou central)

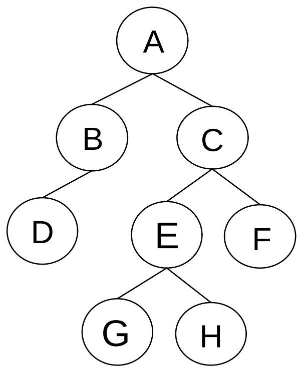
Pesquisa em Pré-Ordem

- Passos:
  - Visita o nodo raiz
  - Pesquisa em pré-ordem cada uma das subárvores (pela ESQ (*default*) ou pela DIR)



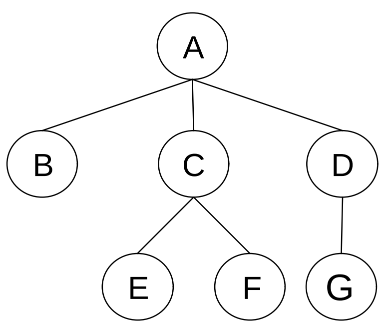
Pré-Ordem em Árvore Binária

- Passos:
  - Visita o nodo raiz
  - Pesquisa em pré-ordem a subárvore à ESQ
  - Pesquisa em pré-ordem a subárvore à DIR



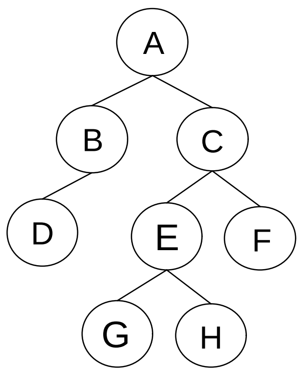
Pesquisa em Pós-Ordem

- Passos:
  - Pesquisa em pós-ordem cada uma das subárvores (pela ESQ (*default*) ou pela DIR)
  - Visita o nodo raiz



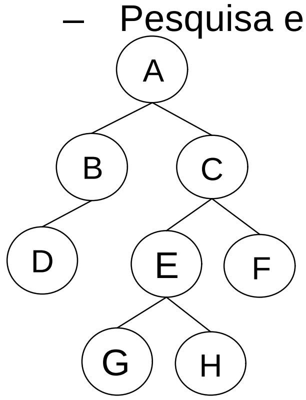
Pós-Ordem em Árvore Binária

- Passos:
  - Pesquisa em pós-ordem a subárvore à ESQ
  - Pesquisa em pós-ordem a subárvore à DIR
  - Visita o nodo raiz



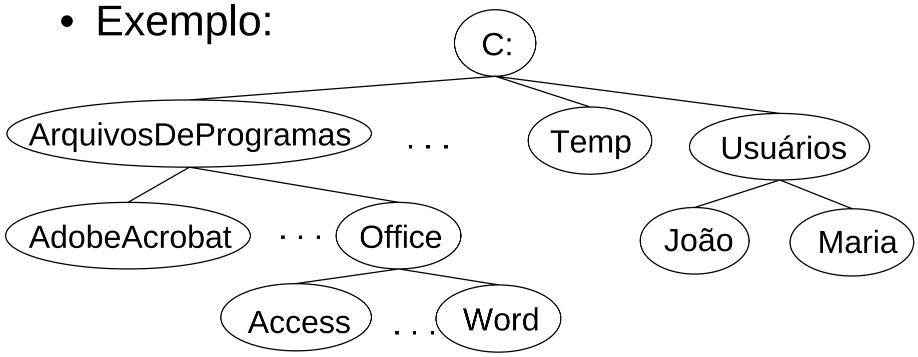
Pesquisa em In-Ordem

- Aplica-se apenas a árvores binárias
- Passos:
  - Pesquisa em in-ordem a subárvore à ESQ
  - Visita o nodo raiz
  - Pesquisa em in-ordem a subárvore à DIR



Pesquisa em Árvores

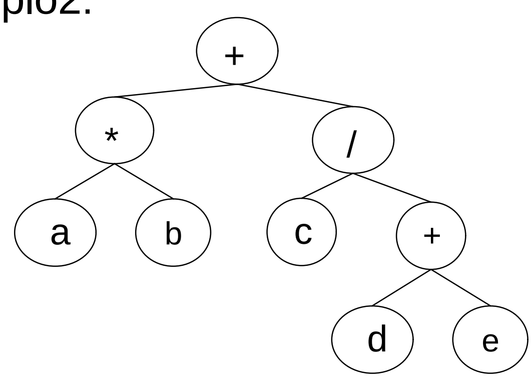
- O método de pesquisa a ser utilizado depende da intenção da aplicação
- Exemplo:



C: (ArquivosDeProgramas (AdobeAcrobat; ...; Office (Access; ...; Word)); Temp; Usuários (João; Maria))

Pesquisa em Árvores

- Exemplo2:



a * b + c / (d + e)