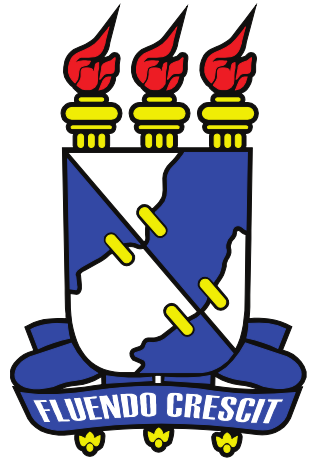


Estrutura de Dados I

Árvores binárias

Bruno Prado

Departamento de Computação / UFS

Introdução

- O que é uma árvore (em computação)?
  - Representação natural de informações hierárquicas
    - Árvore genealógica
    - Organograma
    - Estruturas de diretórios
    - ...
  - Eficiência e simplicidade

Introdução

- Tipos de árvores
  - Árvore livre ou não enraizada (unrooted) ou grafo
  - Árvore enraizada (rooted)
  - Árvore k-ária

Introdução

- Grafo
  - Conjunto de vértices V e arestas E
  - Os vértices são conectados pelas arestas
  - As arestas podem conter pesos associados

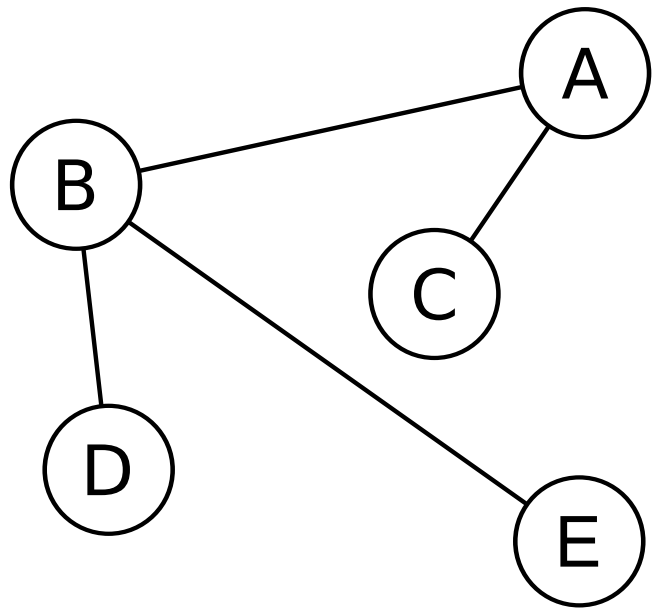
$$G = (V, E)$$

$$V = \{v_1, v_2, \ldots, v_n\}$$

$$E = \{e_1, e_2, \ldots, e_m\}$$

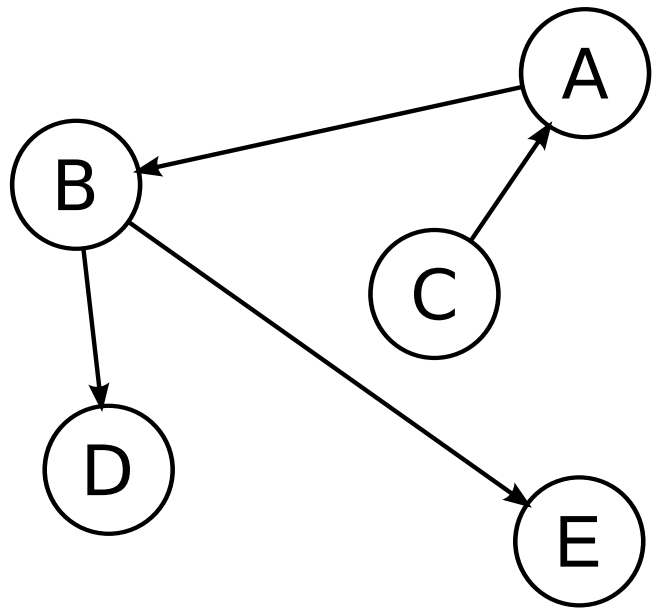
Introdução

- Grafo não direcionado
  - Vértices $V = \{A, B, C, D, E\}$
  - Arestas $E = \{AB, AC, BA, BD, BE, CA, DB, EB\}$



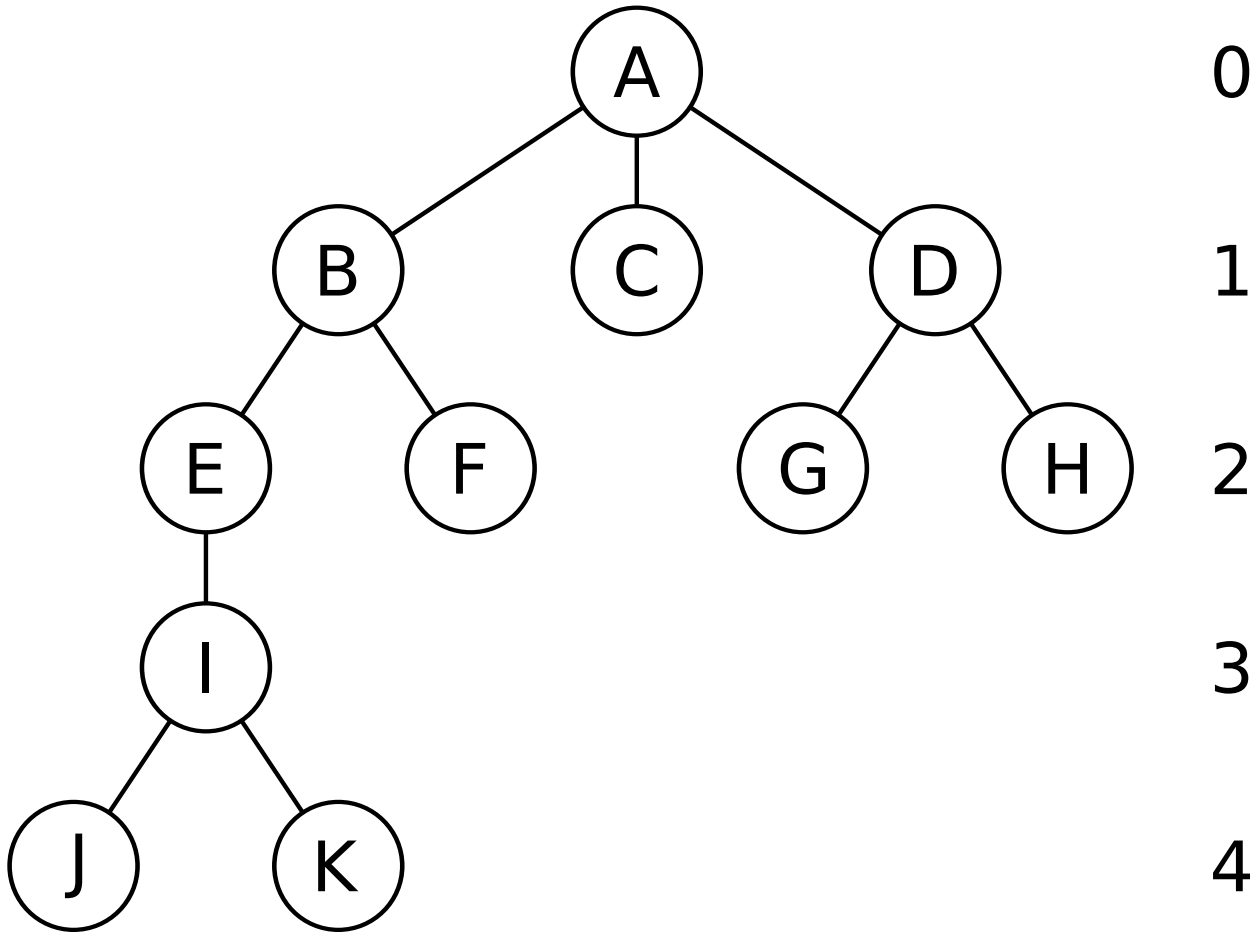
Introdução

- Grafo direcionado (dígrafo)
  - Vértices $V = \{A, B, C, D, E\}$
  - Arestas $E = \{AB, BD, BE, CA\}$



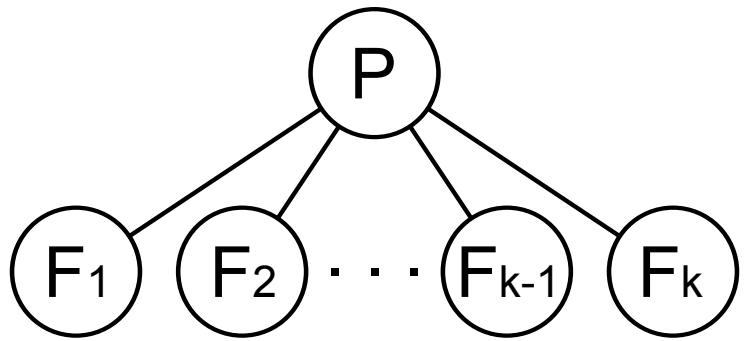
Introdução

- Árvore enraizada
  - Conjunto de nós ou vértices, com um nó especial que é denominado de raiz
  - Possuem níveis e as partições geram subárvores



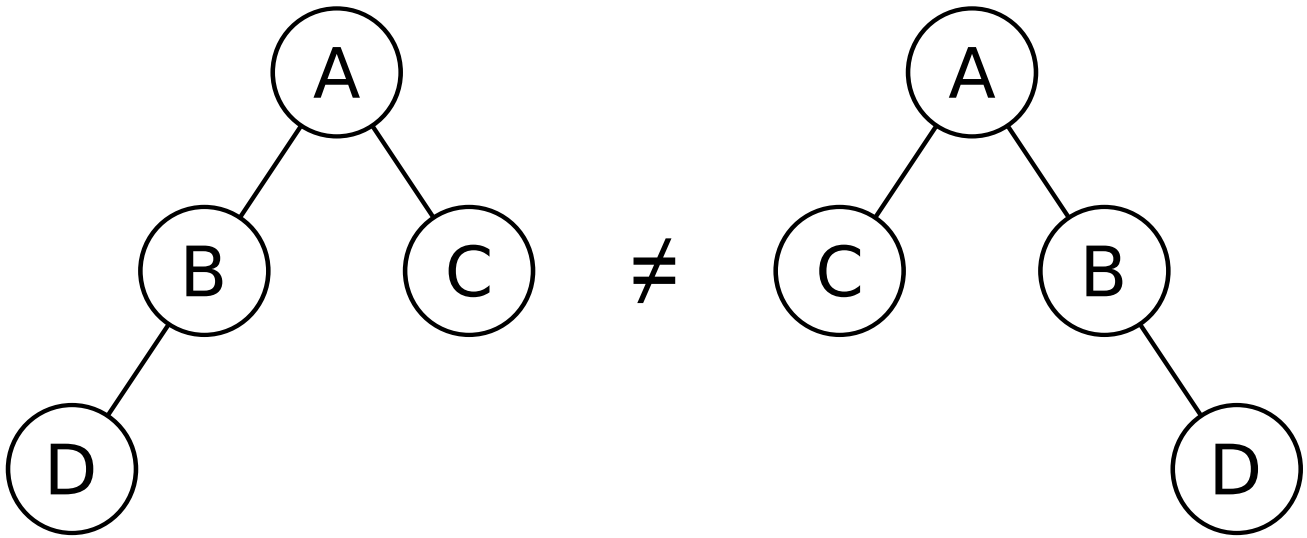
Introdução

- Árvore *k*-ária
  - Cada nó ou vértice tem no máximo *k* subárvores
  - O número de subárvores define o grau do vértice
  - Conceito de pai *P* e filho F_i



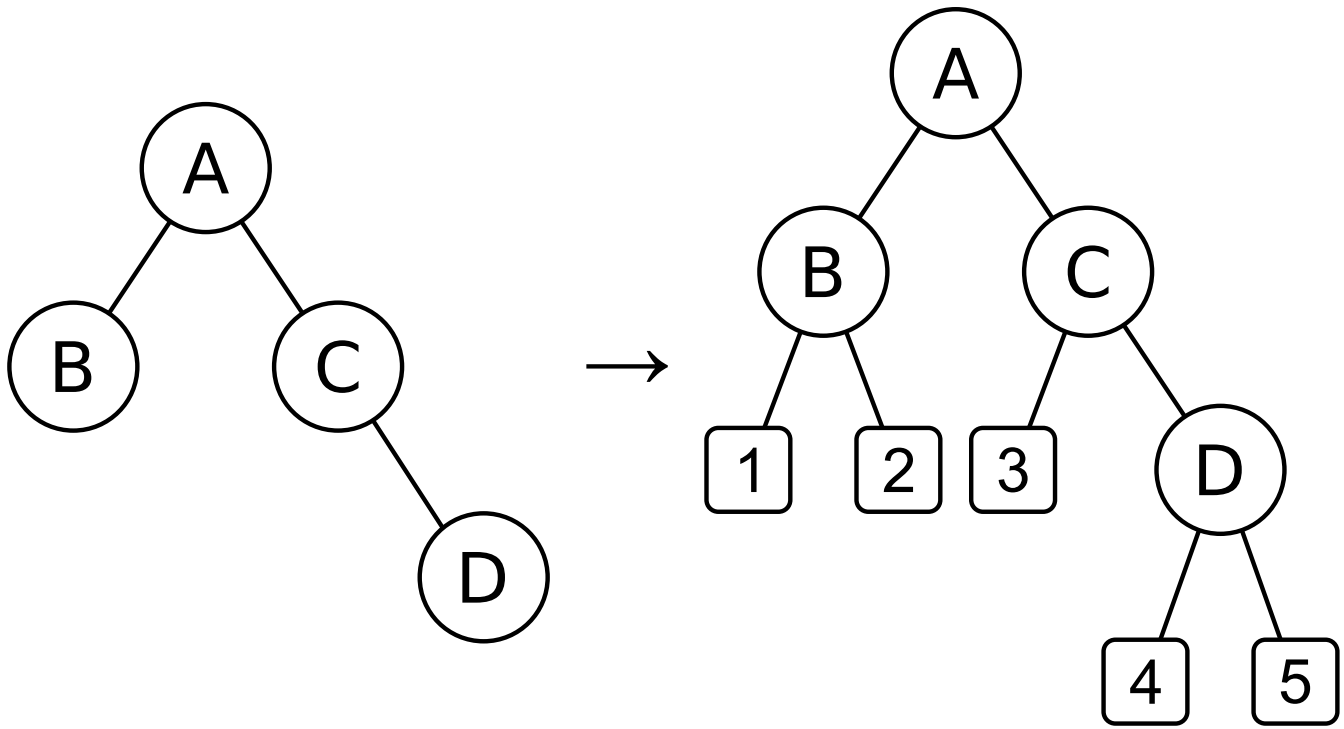
Árvores Binárias

- O que é uma árvore binária?
  - É uma árvore enraizada cujos nós pai referenciam até 2 nós filho ou 2 subárvores
  - Assimetria dos nós e dos ramos



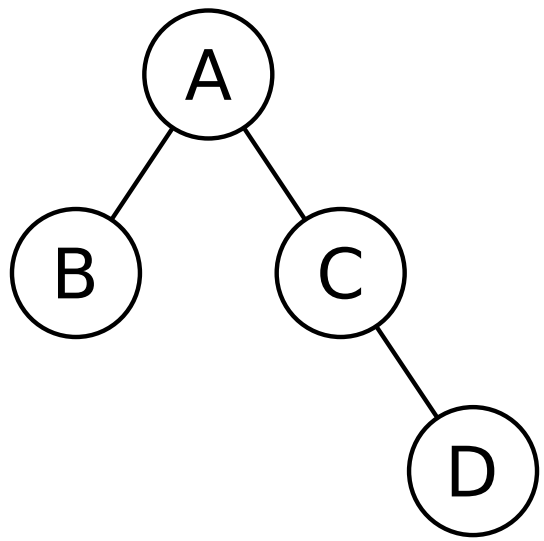
Árvores Binárias

- Propriedades
  - Uma árvore binária com n nós internos (não folhas) possui $n + 1$ nós externos (folhas)
    - Cada nó interno possui exatamente dois filhos, utilizando um total de $2n$ ramos
    - Existem $n - 1$ ramos para nós internos
    - Restam $n + 1$ ramos para cada nó externo



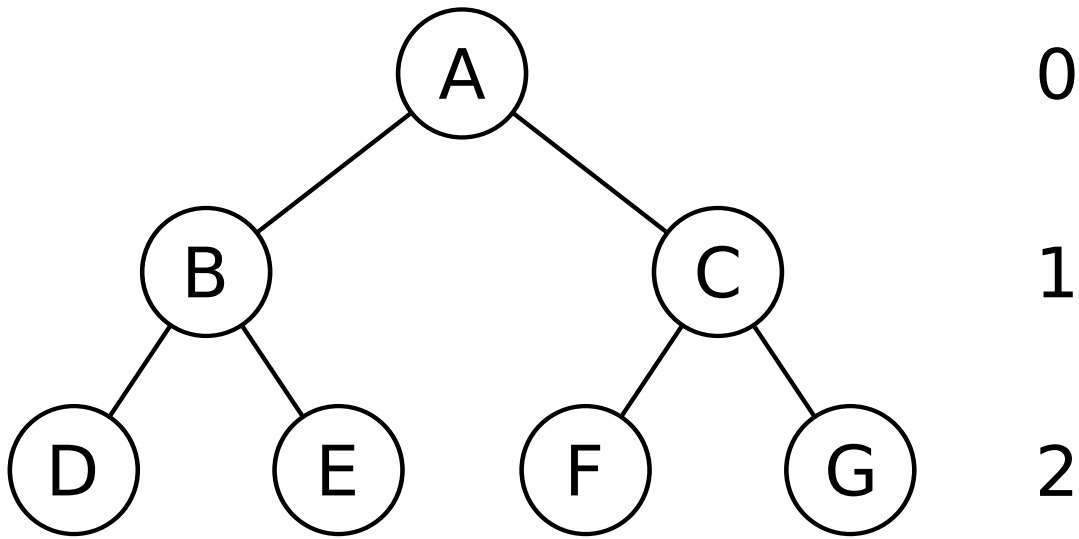
Árvores Binárias

- Propriedades
  - Qualquer árvore binária não vazia com n_0 nós folha e n_2 nós com grau 2, temos que $n_0 = n_2 + 1$
    - Total de nós é $n = n_0 + n_1 + n_2$
    - A árvore possui $n - 1$ ramos, todos partindo de nós que possuem grau 1 ou 2
    - Implicando em $n - 1 = n_1 + 2n_2 \rightarrow n = n_1 + 2n_2 + 1$



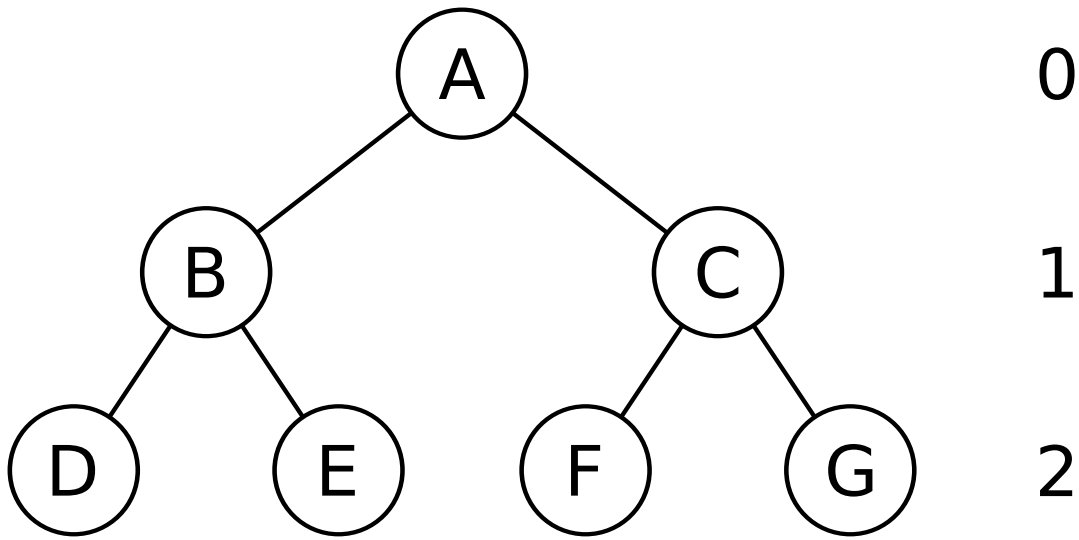
Árvores Binárias

- Propriedades
  - O número máximo de nós em um nível i é 2^i para $i \geq 0$
    - Progressão geométrica
    - Cada passo duplica o número máximo de nós



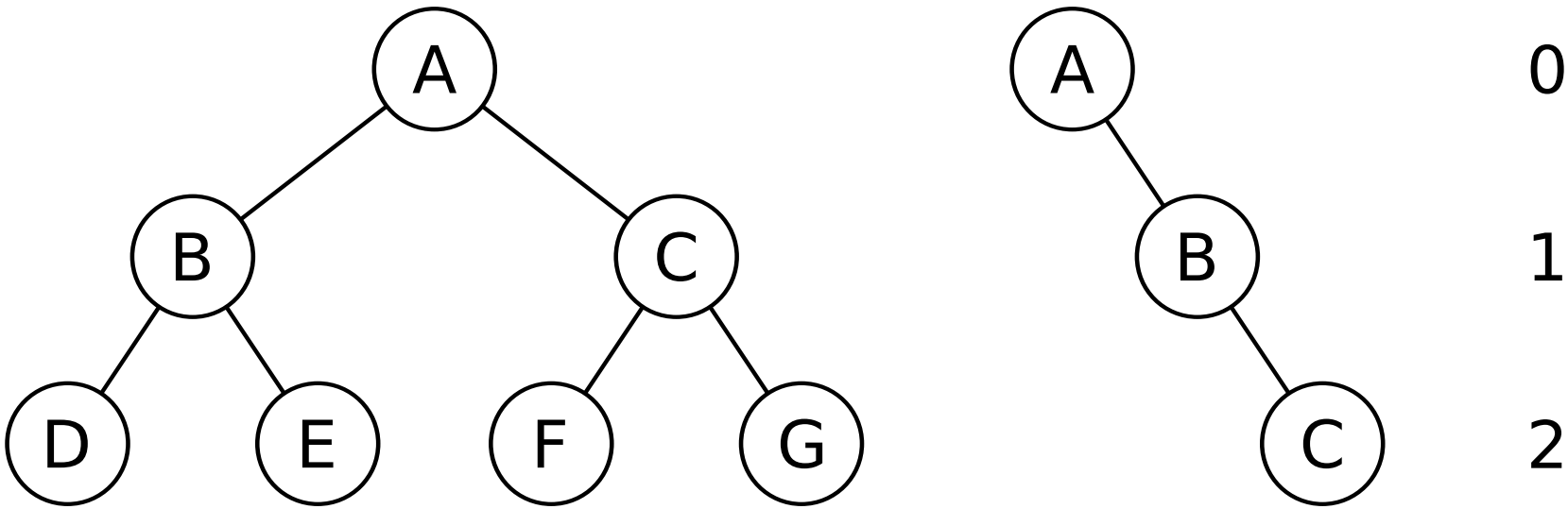
Árvores Binárias

- Propriedades
  - O número máximo de nós de uma árvore de altura h é $2^{h+1} - 1$
    - Cada nível possui no máximo 2^i nós
    - Temos que $0 \leq i \leq h$
    - Total de nós é $\sum_{i=0}^{h} 2^i$ e utilizando o somatório de progressão geométrica $\sum_{k=1}^{n} ar^{k-1} = \frac{a(1-r^n)}{1-r}$, é obtido $\sum_{k=1}^{h+1} 2^{k-1} = \frac{1(1-2^{h+1})}{1-2} = 2^{h+1} - 1$



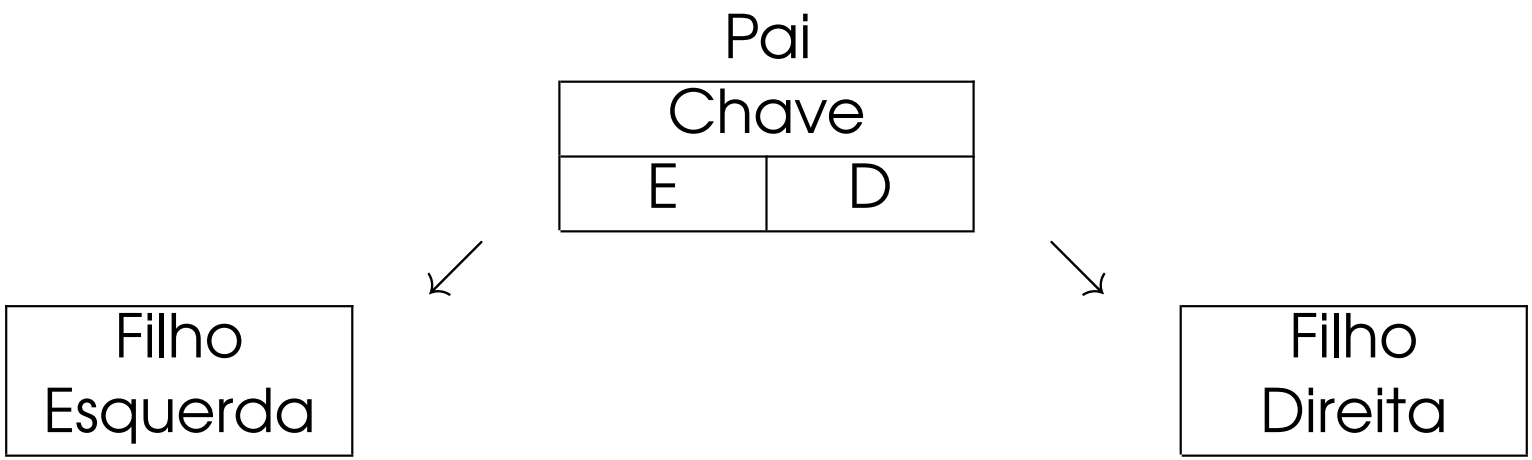
Árvores Binárias

- Propriedades
  - A altura h de uma árvore binária com n nós internos é pelo menos $\log_2 n - 1$ e no máximo $n - 1$
    - No caso inferior, com níveis com 2^i nós e total máximo de nós igual a $n \leq 2^{h+1} - 1$
    - No caso superior, o número de nós externos é $n + 1$ e que temos $n - 1$ ramos que é igual a altura h



Árvores Binárias

- Implementação em C
  - Estruturas e ponteiros



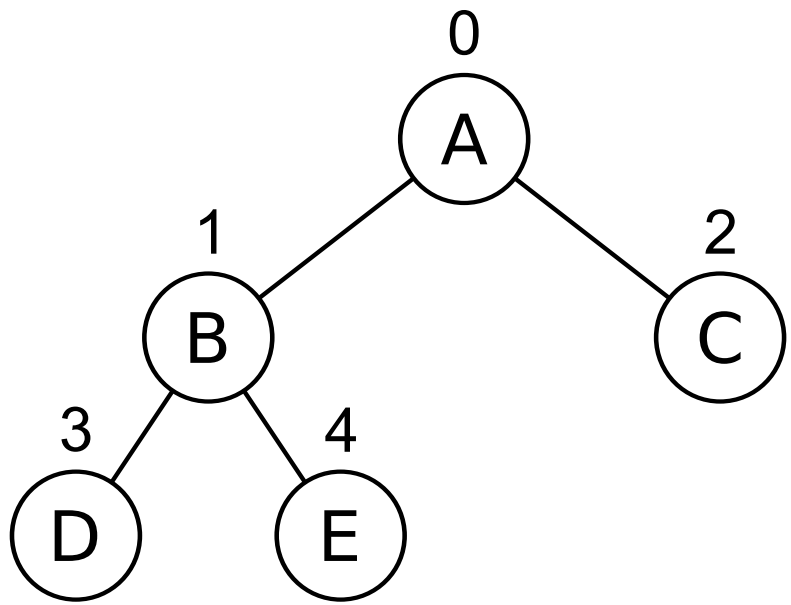
Árvores Binárias

- Implementação em C
  - Estruturas e ponteiros

```
// Representação da árvore binária
typedef struct no {
        // Chave do nó
        int chave;
        // Filho da esquerda
        struct no* E;
        // Filho da direita
        struct no* D;
} no;
```

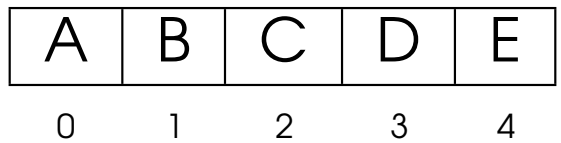
Árvores Binárias

- Implementação em C
  - Vetores e índices



Árvores Binárias

- Implementação em C
  - Vetores e índices
  - Nó de índice i
    - Filho da esquerda é $2i + 1$
    - Filha da direita é $2i + 2$

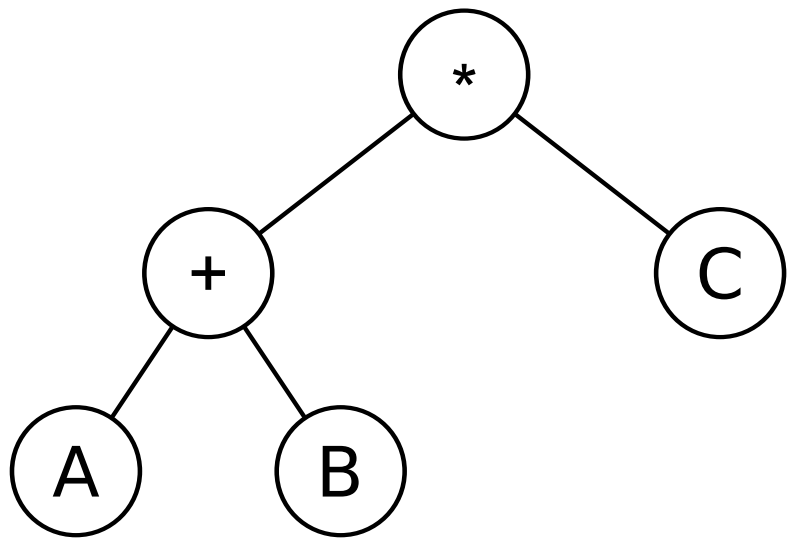


Árvores Binárias

- Percursos na árvore
  - Definem como cada nó da árvore será visitado
    - Em ordem
    - Pré-ordem
    - Pós-ordem

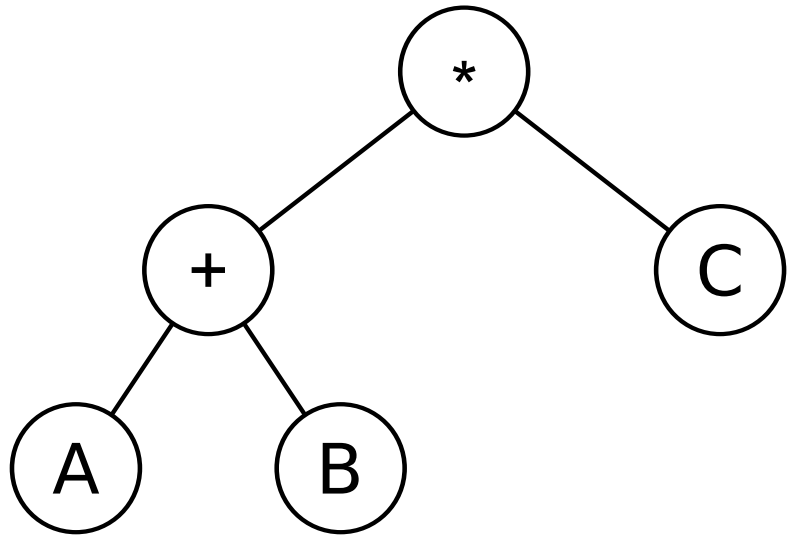
Árvores Binárias

- Percurso em ordem
  - Expressão $(A + B) * C$
  - Regra E P D



Árvores Binárias

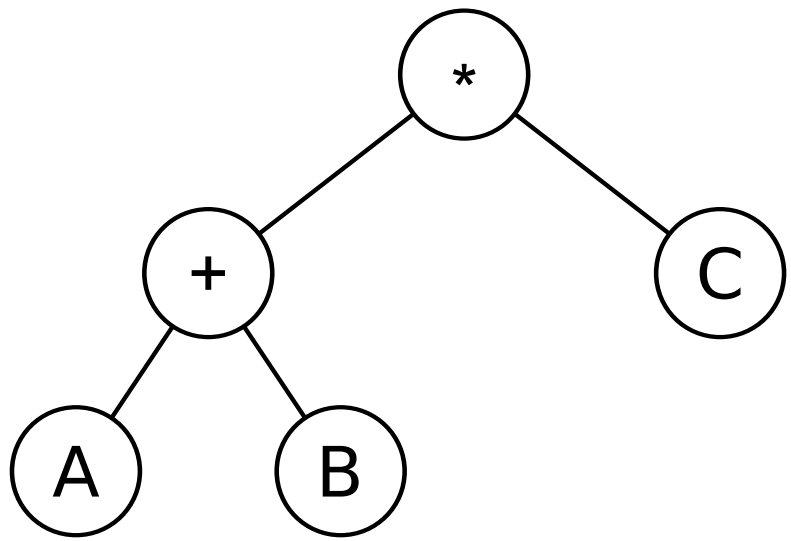
- Percurso em ordem
  - Expressão $(A + B) * C$
  - Regra E P D



O percurso realizado foi A+B*C

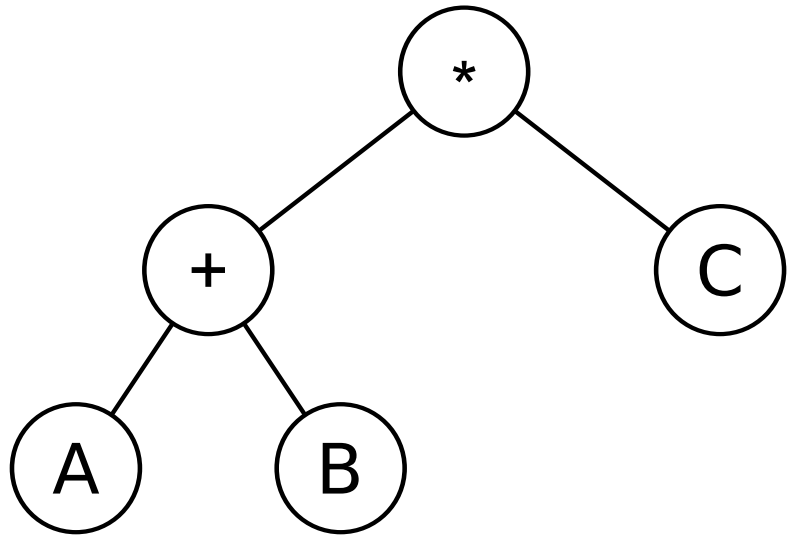
Árvores Binárias

- Percurso pré-ordem
  - Expressão $(A + B) * C$
  - Regra P E D



Árvores Binárias

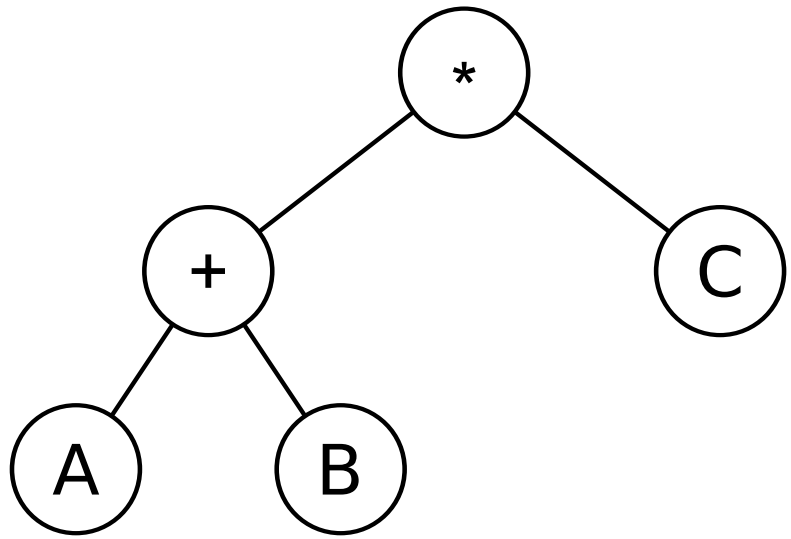
- Percurso pré-ordem
  - Expressão $(A + B) * C$
  - Regra P E D



O percurso realizado foi *+ABC

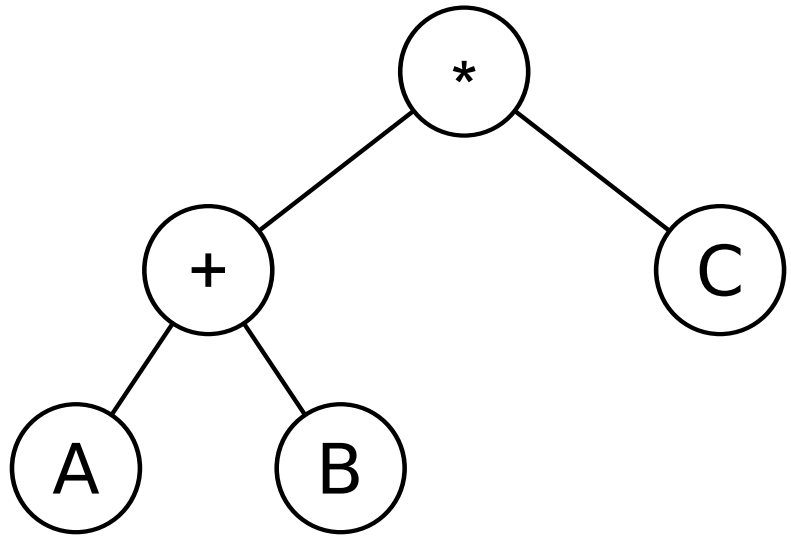
Árvores Binárias

- Percurso pós-ordem
  - Expressão $(A + B) * C$
  - Regra E D P



Árvores Binárias

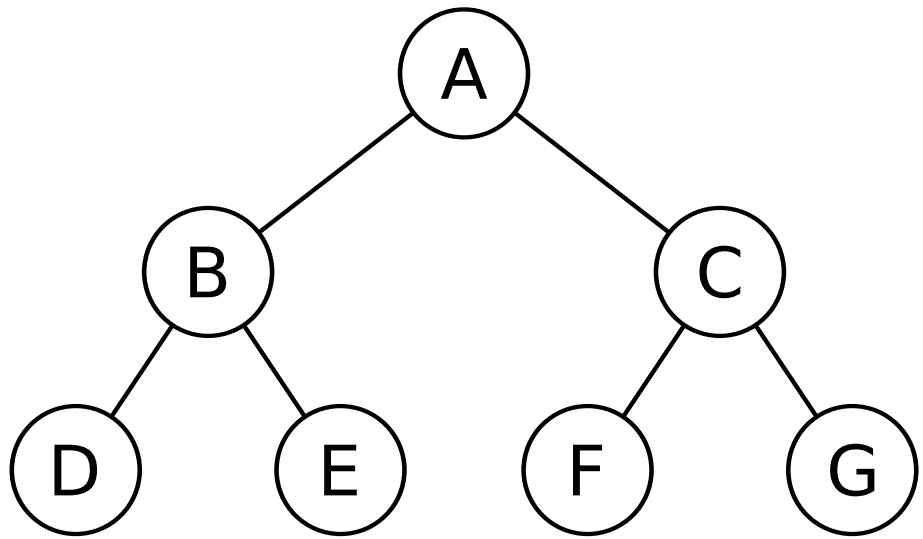
- Percurso pós-ordem
  - Expressão $(A + B) * C$
  - Regra E D P



O percurso realizado foi AB+C*

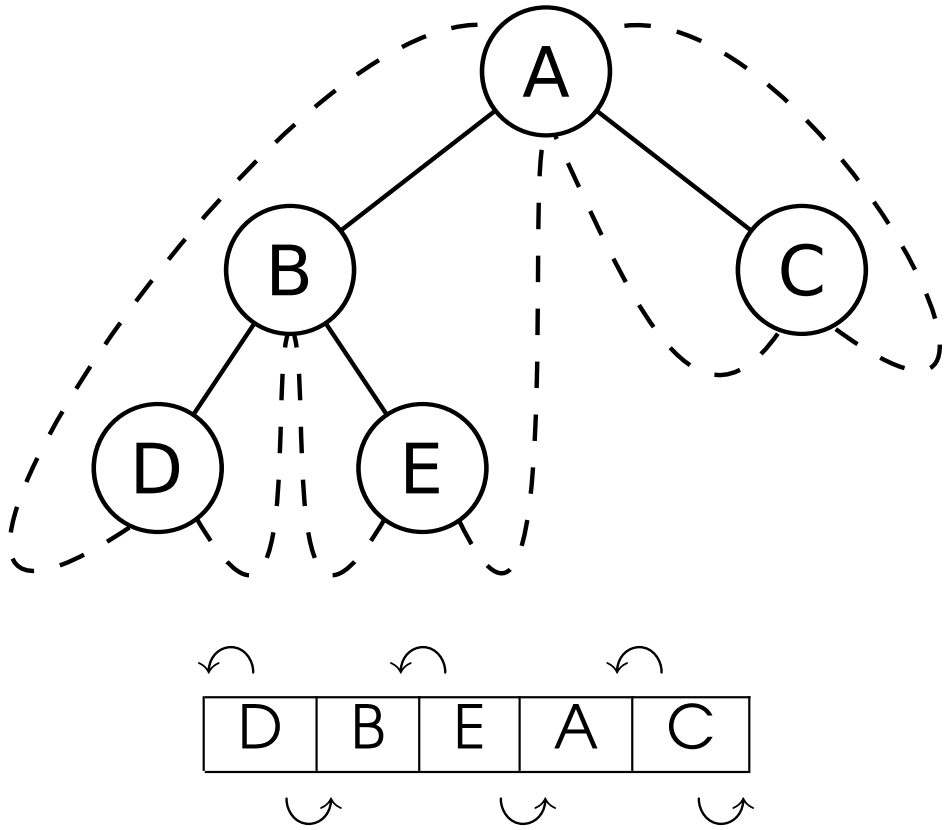
Exemplo

- Realize o percurso em ordem, pré-ordem e pós-ordem na árvore binária



Árvores Binárias

- Árvore binária costurada
  - Uma árvore binária com n nós possui $n + 1$ ponteiros nas folhas com valores nulos
  - Os ponteiros direito e esquerdo do nó folha referenciam seu sucessor e predecessor no percurso em ordem, respectivamente



Exemplo

- Defina os ponteiros de costura da árvore binária