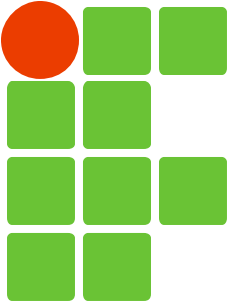


INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Árvores

Prof. Victorio Albani de Carvalho

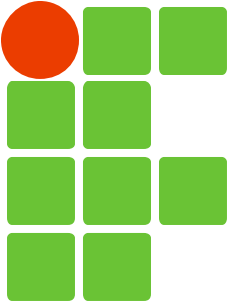


INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Arvores

- Estrutura de dados muito utilizada
 - Permite a representação de dados de maneira hierárquica;
 - Fornece maneiras eficientes de busca;
- Quais são seus usos comuns?
 - Manipular dados hierárquicos
 - Manipular listas ordenadas de dados
 - Em algoritmos de roteamento de pacotes

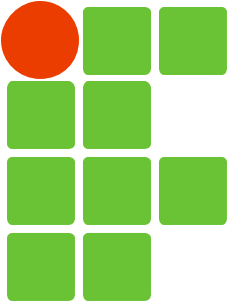


INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Árvores

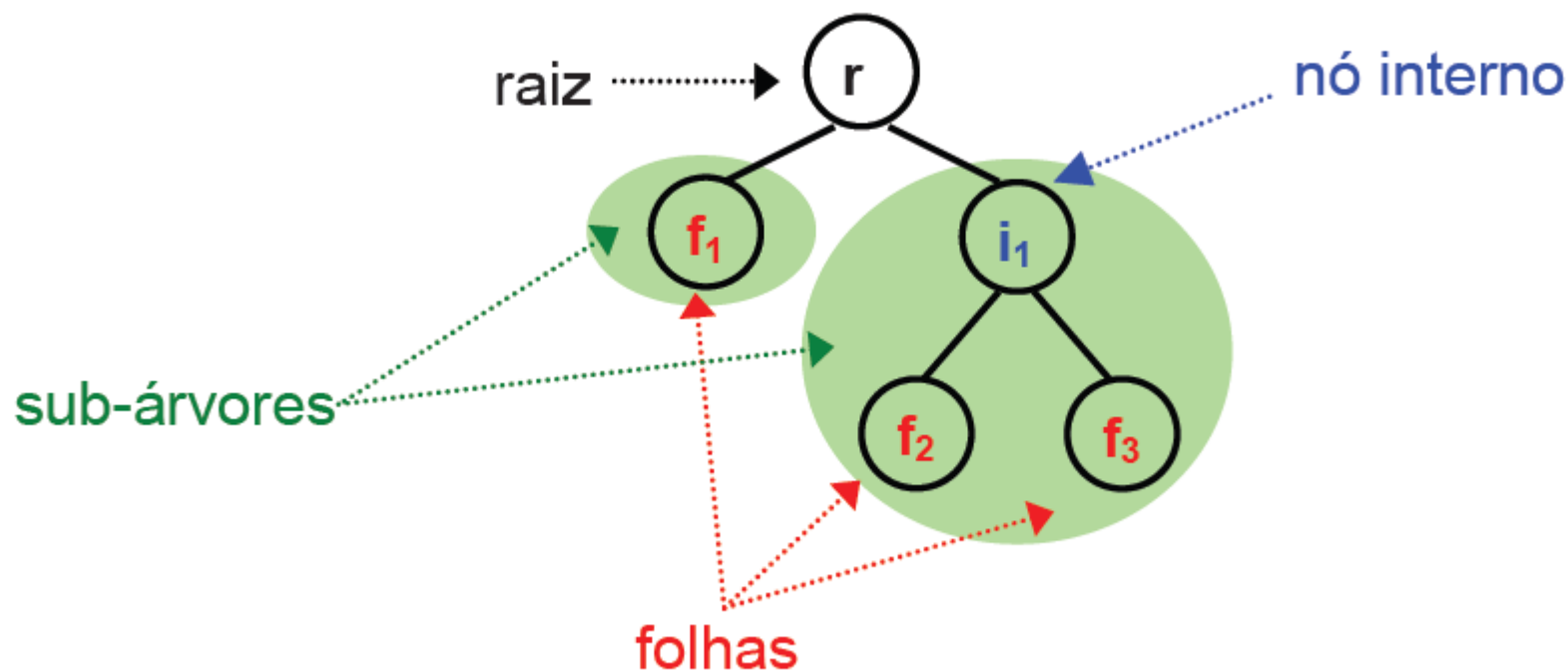
- Listas Encadeadas: Lineares
- Tabelas Hash: Acesso direto com estruturas lineares em caso de colisão
- Árvores: Estruturas Hierárquicas

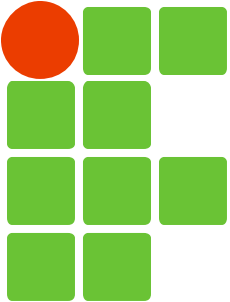


INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Definição de Arvore

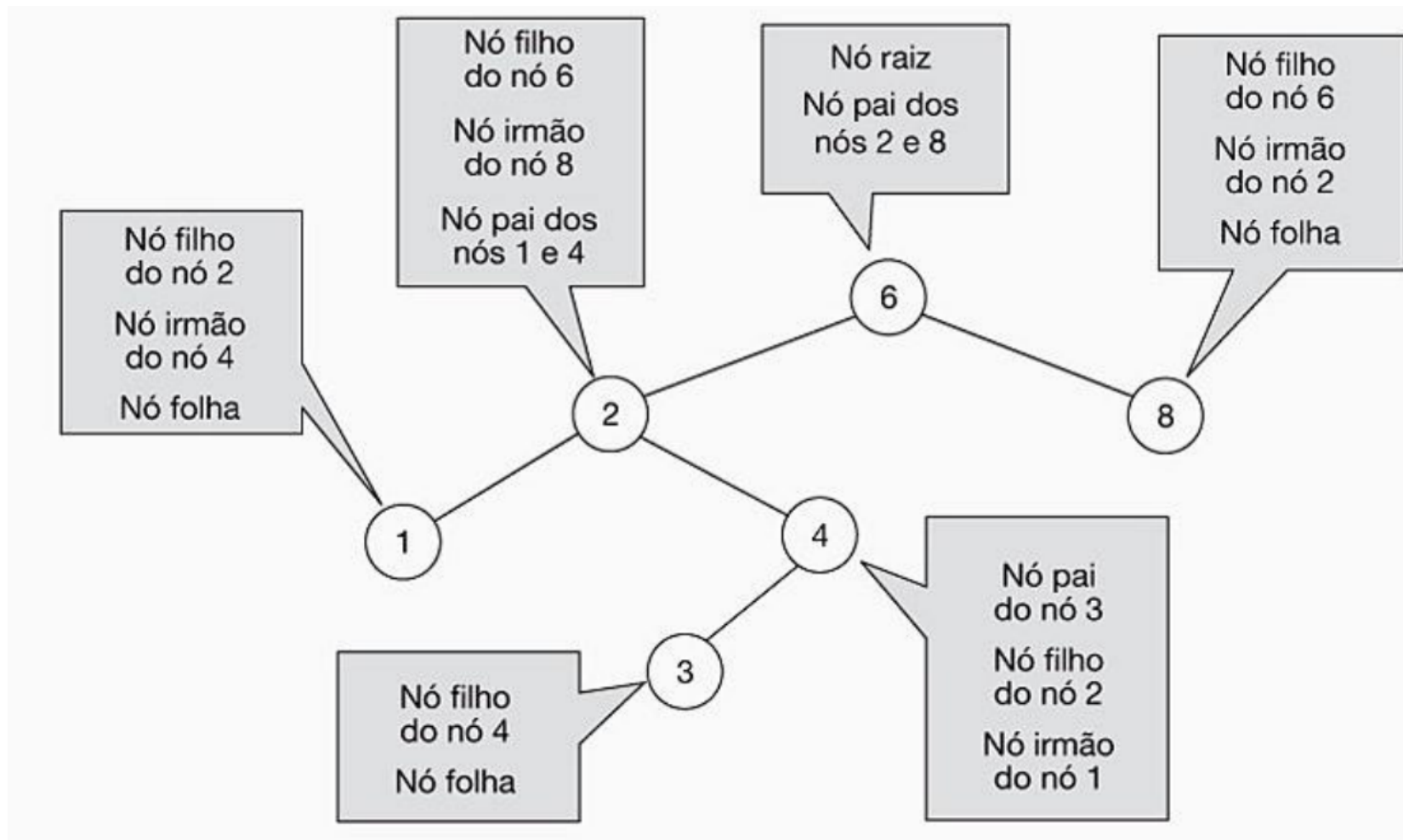


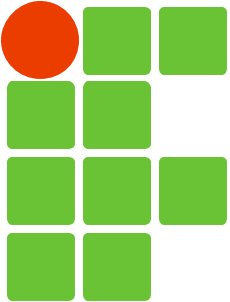


INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Parentesco

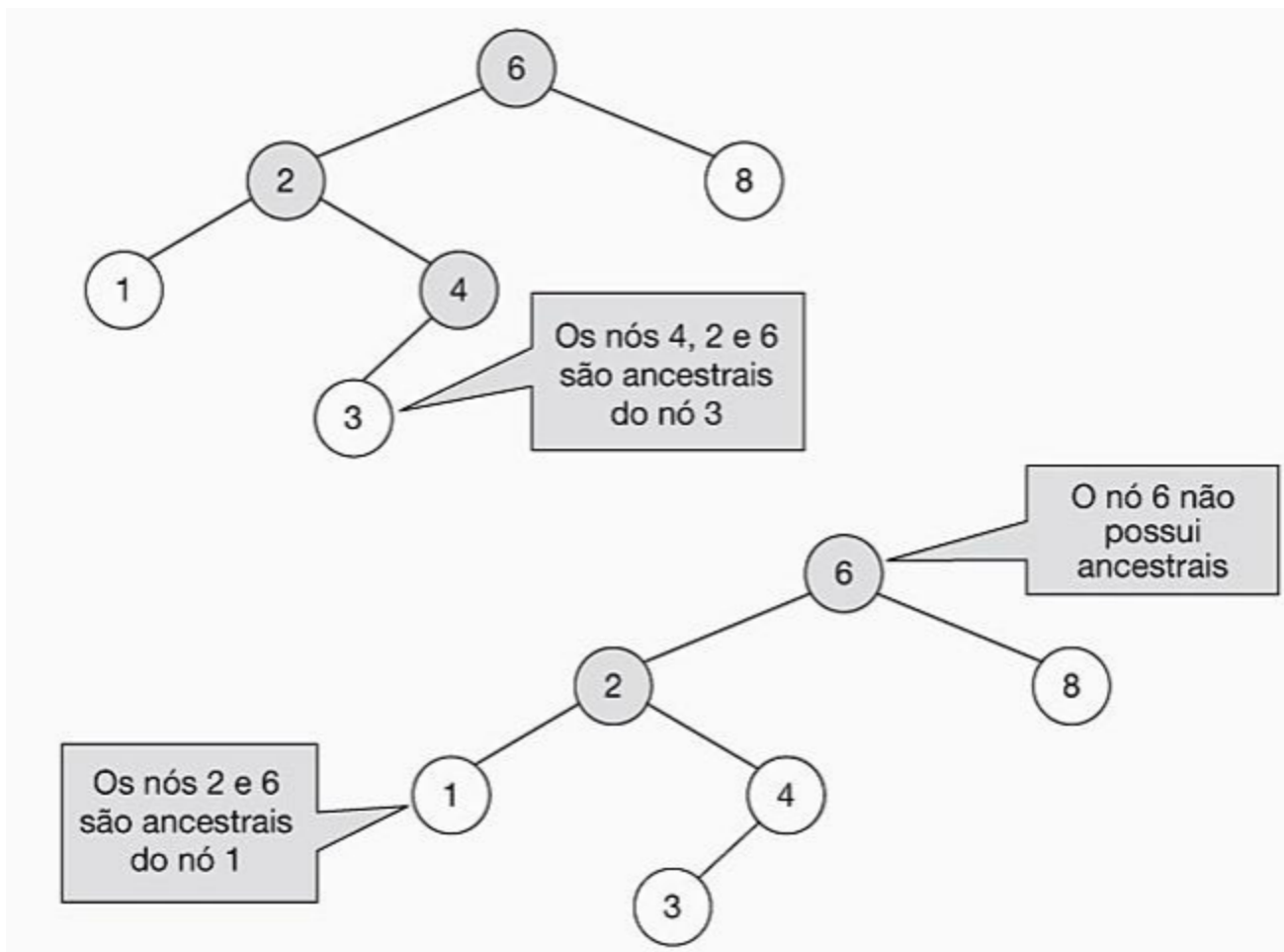


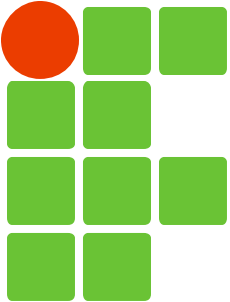


INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Ancestrais

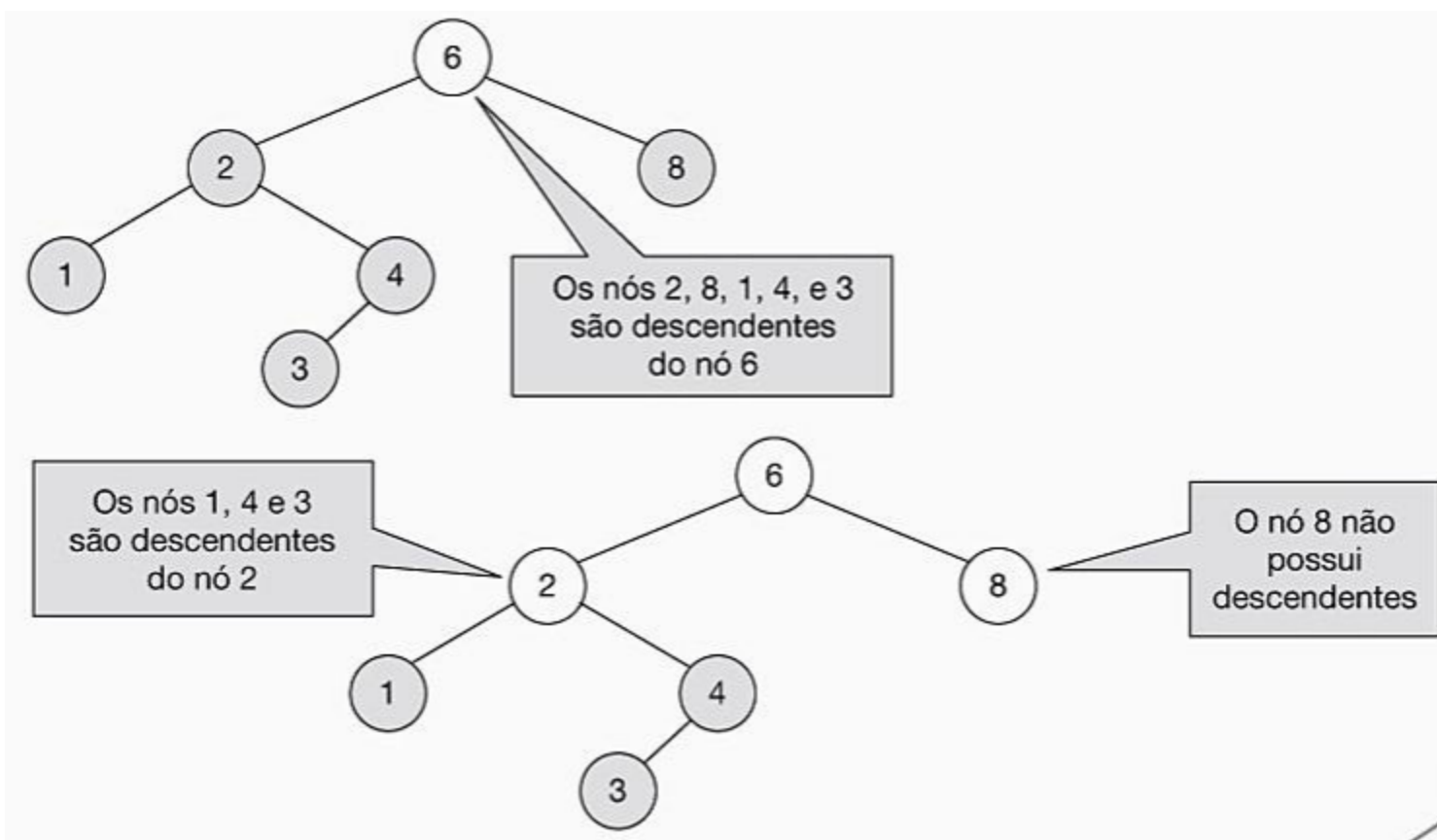


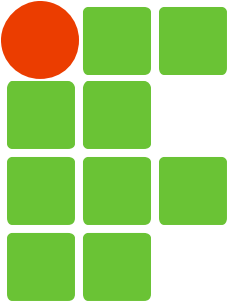


INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Descendentes

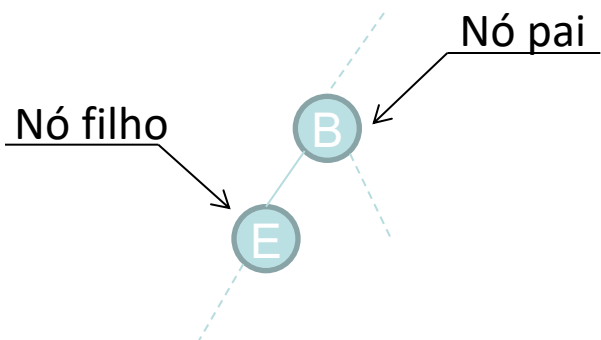
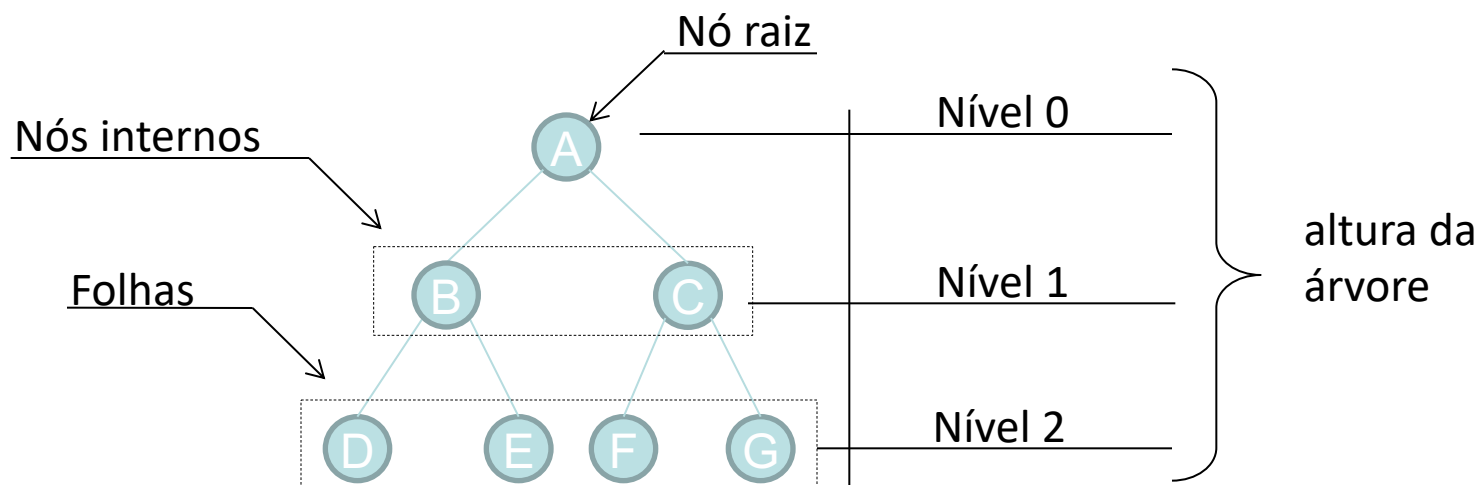


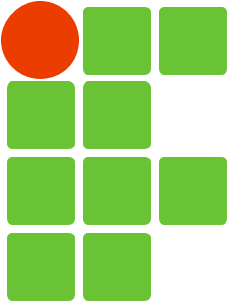


INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Estrutura



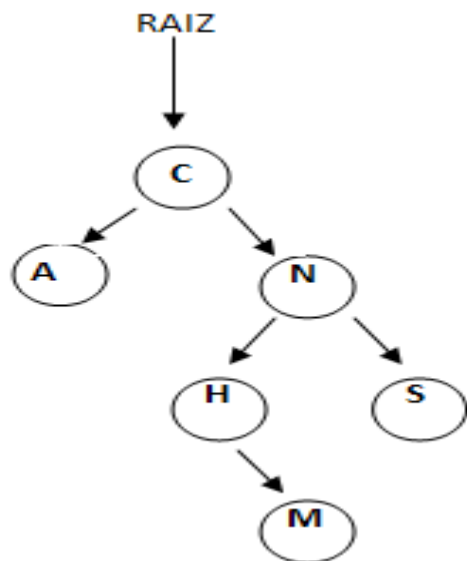


INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Grau de um nó

- **Grau** representa o número de **subarvores** de um **nó**.
- Exemplo:

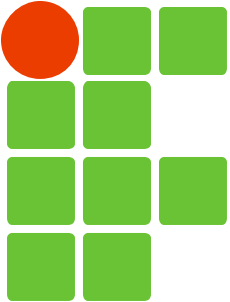


O nó "C" tem grau 2.

O nó "A" tem grau zero.

O nó "H" tem grau 1.

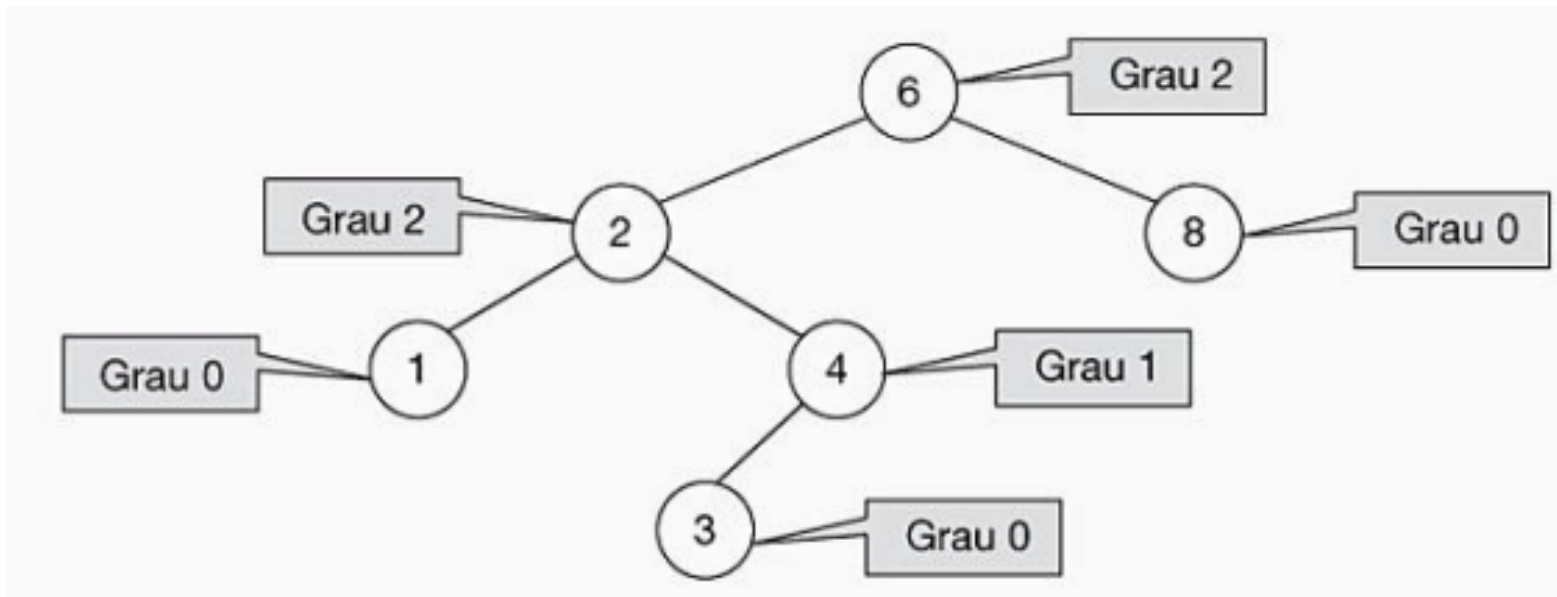
O nó "N" tem grau 2.



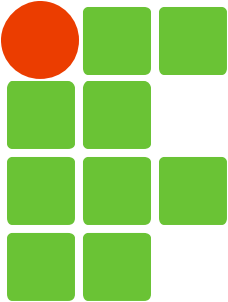
INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Graus de um nó



O Grau da árvore é o máximo dos graus de seus nós

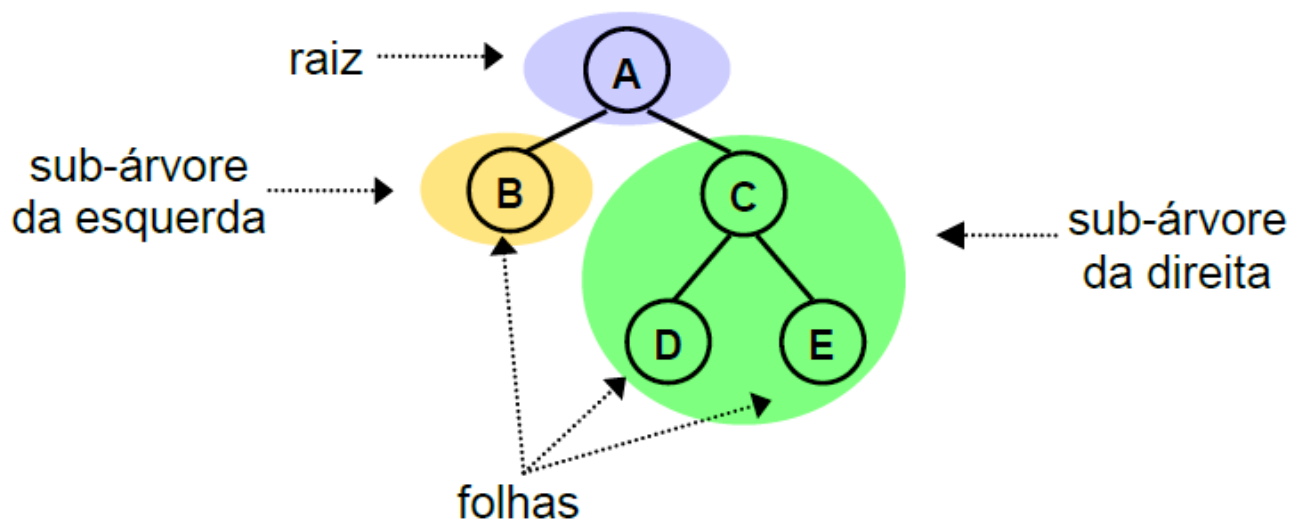


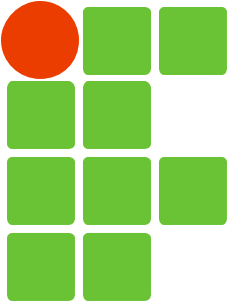
INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Árvores Binárias

- Árvore de grau 2
- É muito útil para modelar situações em que precisam ser tomadas decisões bidirecionais em cada ponto de um processo



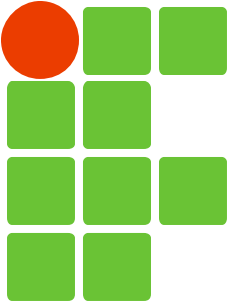


INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

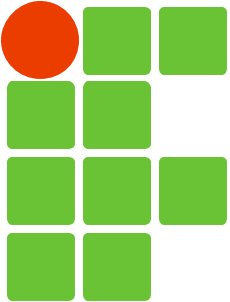
Árvore Binária

Nível (<u>d</u>)	n. max. nós	Potência
0	1	2^0
1	2	2^1
2	4	2^2
3	8	2^3
4	16	2^4
5	32	2^5



Árvore Binária

- Relação entre altura(h) e número de nós (n) : $n-1 \geq h \geq \text{piso}(\log(n))$
- Uma **árvore binária** que armazene **n** nós possui uma **altura máxima $n - 1$** .
- Uma **árvore binária** que armazene **n** nós possui uma **altura mínima $\text{piso}(\log(n))$** .
- Uma **árvore binária** de altura **h** armazena **no mínimo $h+1$** e **no máximo $2^{(h+1)}-1$** nós

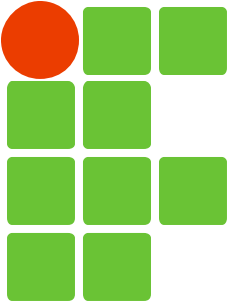


INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Árvore Binária

Altura (h)	n. min. nós	n. max. nós
0	1	1
1	2	3
2	3	7
3	4	15
4	5	31
5	6	63

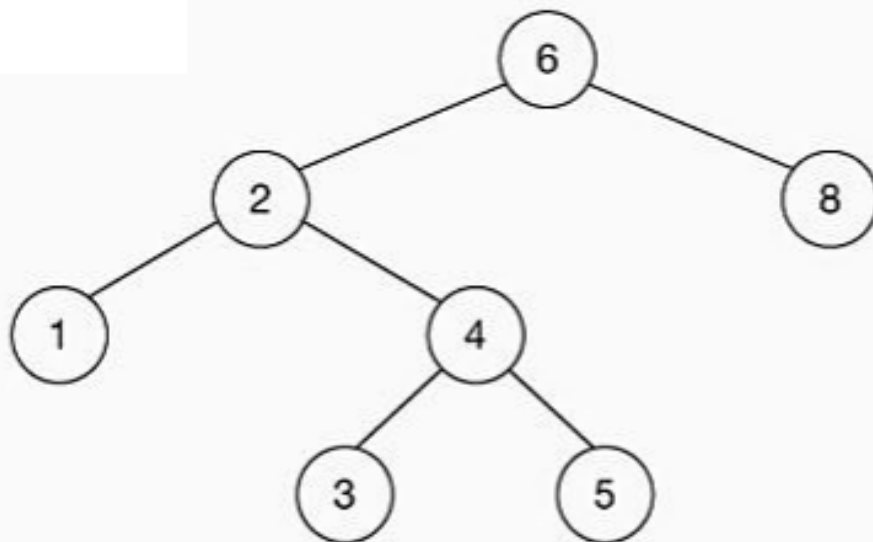


INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Árvore Estritamente Binária

Árvore estritamente binária



Quantidade de nós
folha = 4.

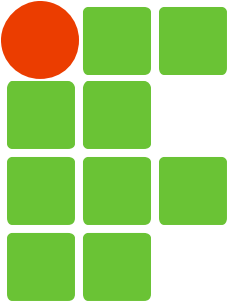
Os nós folha são:
1, 3, 5 e 8.

Número de nós desta
árvore estritamente
binária = $2.n - 1$, onde
 n é o número de folhas
 $2.4 - 1 = 7$ nós

Árvore Estritamente Binária é uma
árvore em que cada nó tem 0 ou 2 filhos.

**Toda Árvore Binária Completa é uma
Árvore Estritamente Binária!**

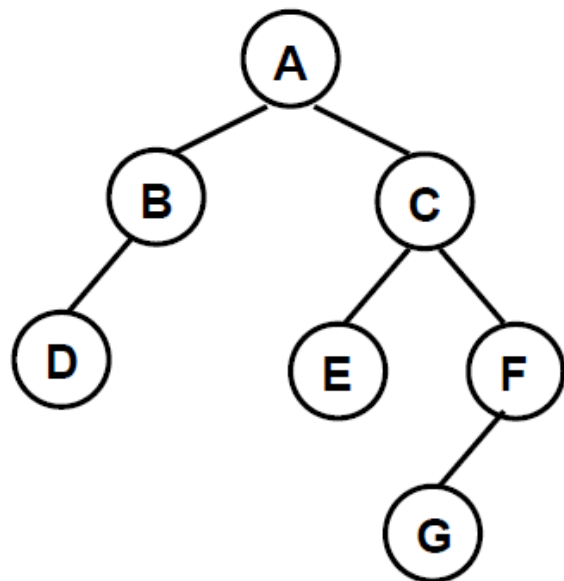
Árvore Completa: Uma árvore de grau d é
uma árvore completa se possui o número
máximo de nós, isto é, todos os nós têm
número máximo de filhos exceto as folhas,
e todas as folhas estão na mesma altura.



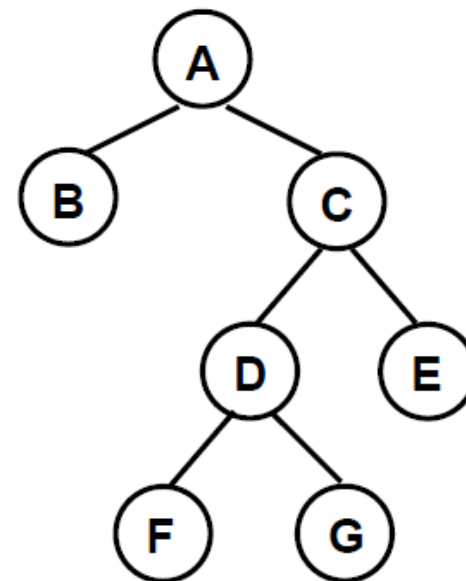
INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

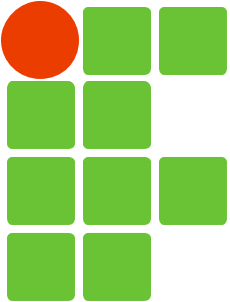
Estritamente Binária



Árvore Binária



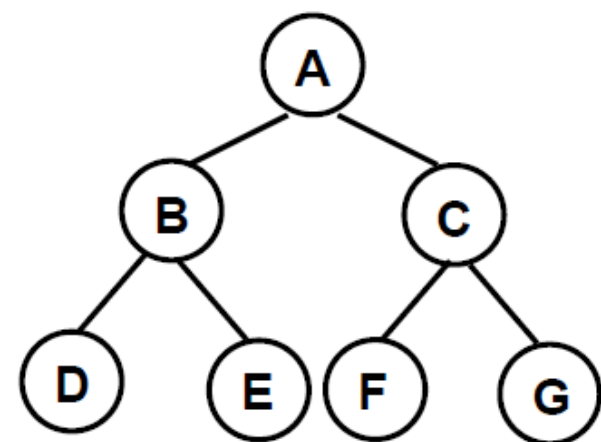
Árvore Estritamente Binária



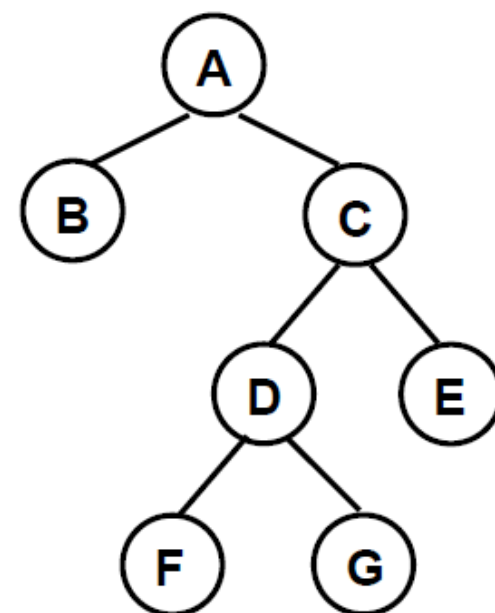
INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Árvore Binária Completa

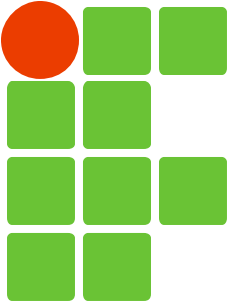


Árvore Binária Completa



Árvore Binária Não Completa

Uma **árvore binária completa** de altura **h** armazena **$2^{(h+1)}-1$** nós.



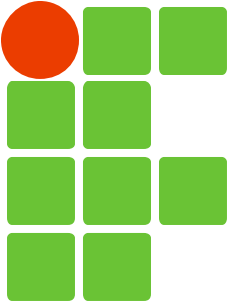
INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Árvore binária de busca

Árvores de busca (ou de pesquisa) binárias aplicam uma regra de inserção a fim de facilitar buscas:

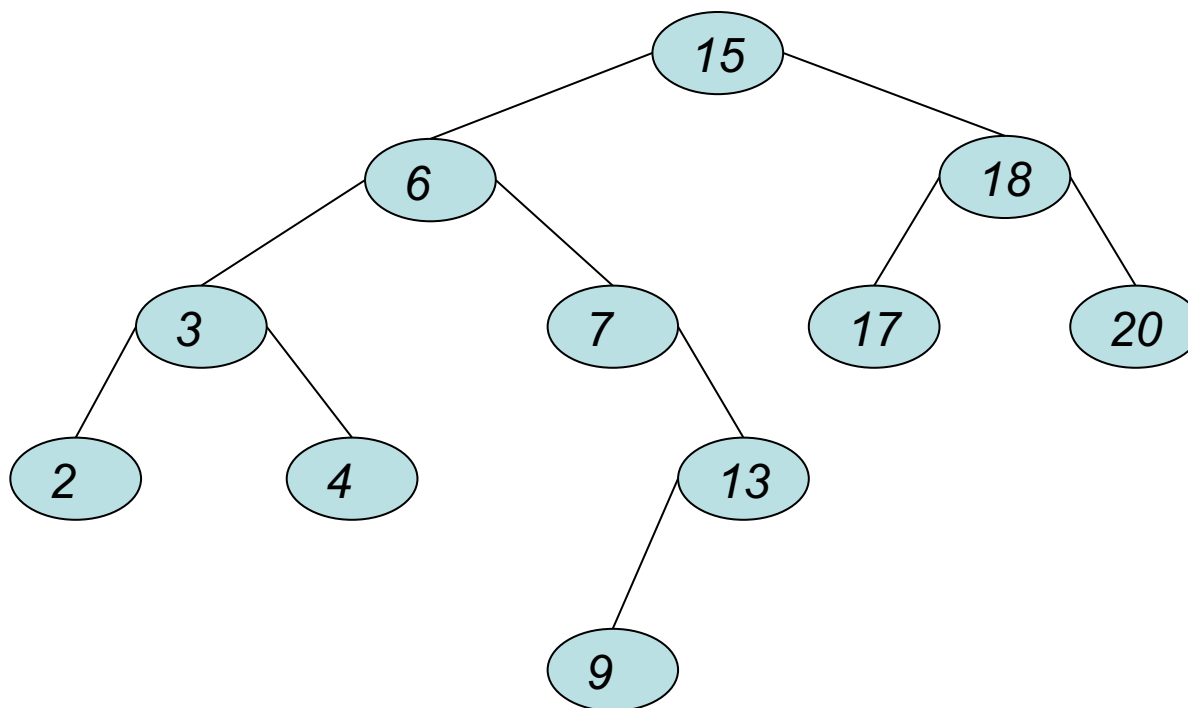
- Seja x um nó em uma árvore de pesquisa binária, se y é um nó da subárvore da esquerda de x , então $\text{chave}(y) \leq \text{chave}(x)$*
- Se y é um nó da subárvore da direita de x , então $\text{chave}(y) \geq \text{chave}(x)$*



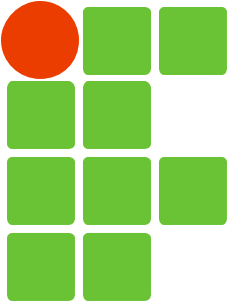
INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Operações Binária



1. *Mínimo* = Nó mais a esquerda = 2
2. *Máximo* = Nó mais à direita = 20
3. *Sucessor de um nó* = Nó mais a esquerda da subárvore direita – Sucessor (15) = 17

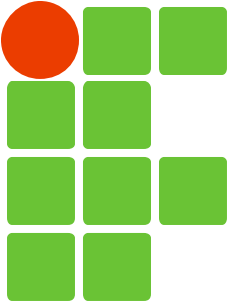


INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

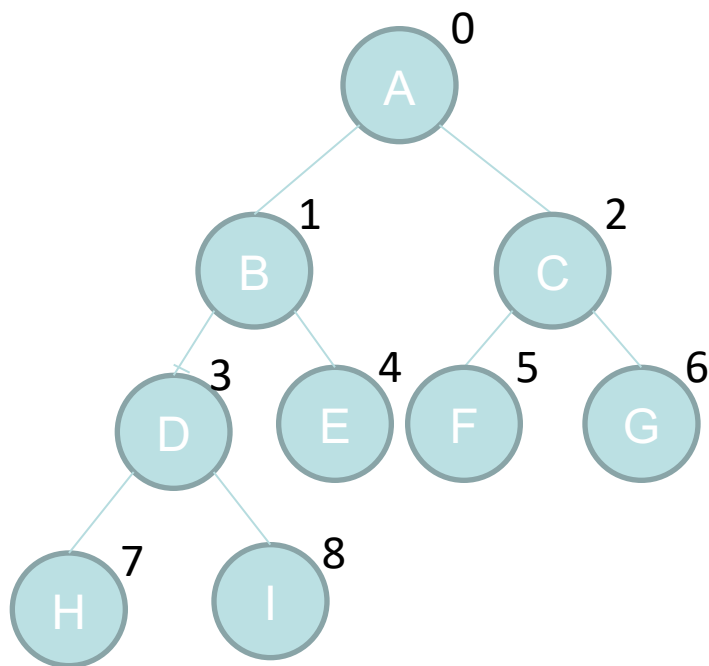
Implementação

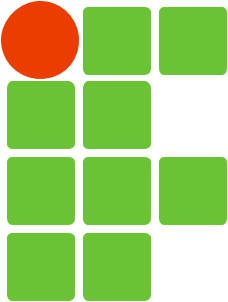
- Lista Seqüencial (Vetores)
- Alocação Dinâmica



Numeração de Nós

- Os nós de uma árvore binária podem ser numerados
- 0 para a raiz
- Filho esquerdo = dobro do n. do pai + 1
- Filho direito = dobro do n. do pai + 2

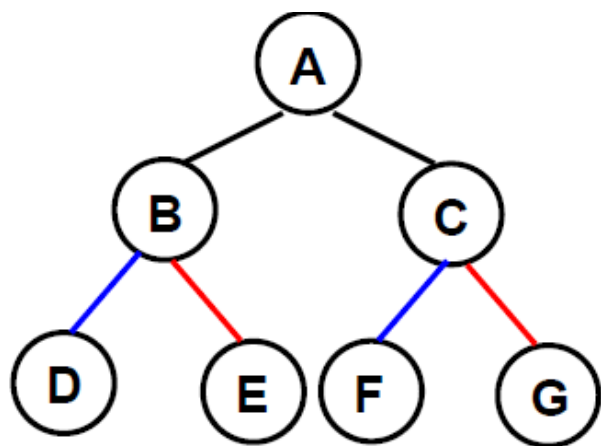




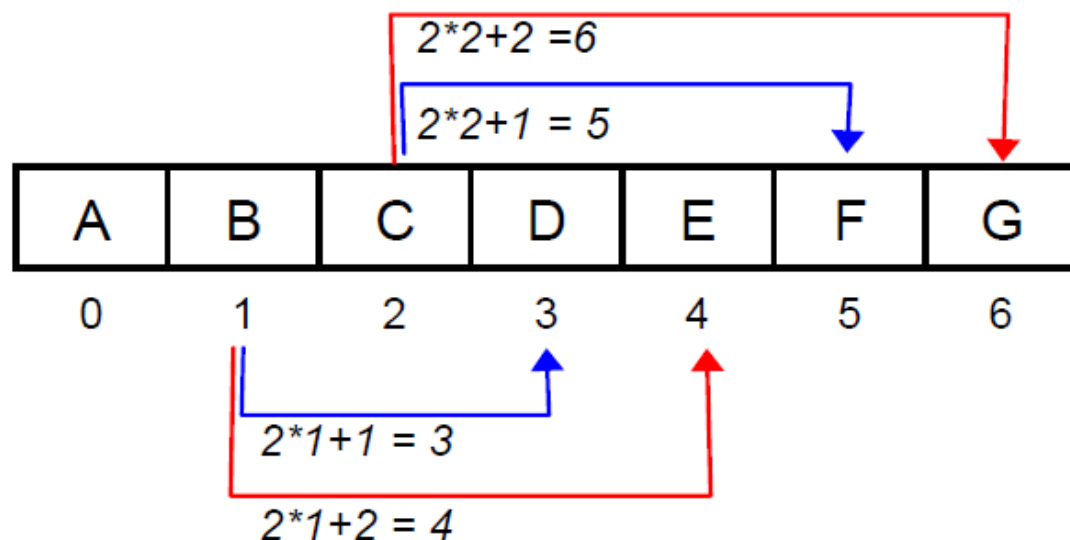
INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

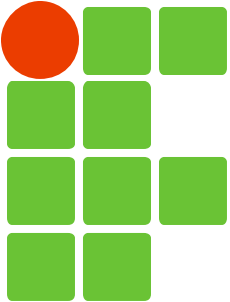
Implementação Sequencial



Árvore Binária Completa



- Armazenar os nós, por nível, num vetor;
- Assim: se um nó ocupa a posição i , então seus filhos diretos estão nas posições $2i+1$ e $2i+2$



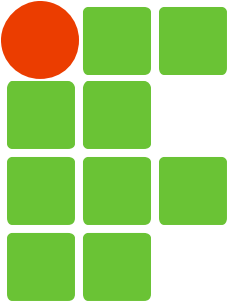
INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Implementação Sequencial

Vantagem: espaço só para armazenar conteúdo; ligações implícitas

Desvantagens: espaços vazios se a árvore não for completa por níveis, ou se sofrer eliminação. *Necessidade de se determinar previamente a altura da árvore.*

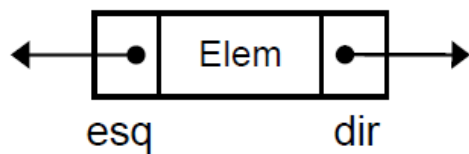


INSTITUTO FEDERAL
ESPIRITO SANTO

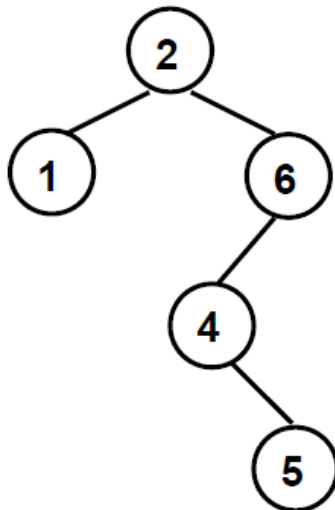
CAMPUS COLATINA

Implementação: Alocação Dinâmica

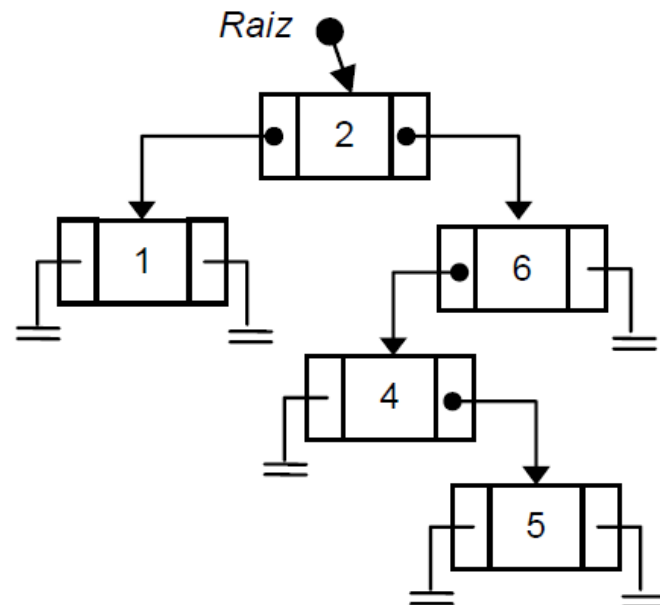
Estrutura do Nó

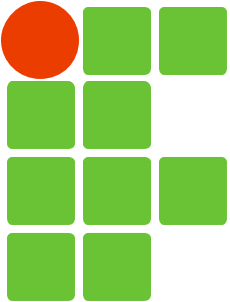


Representação Gráfica



Representação com nós





INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

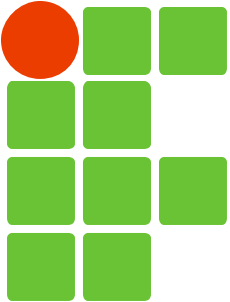
Percorrendo Árvores

Passeio em Profundidade (3 tipos):

- Pré-ordem (*Pre-order*) ou profundidade;
- Em-ordem (*In-order*) ou ordem simétrica;
- Pós-ordem (*Post-order*).

Passeio em Largura ou em Nível

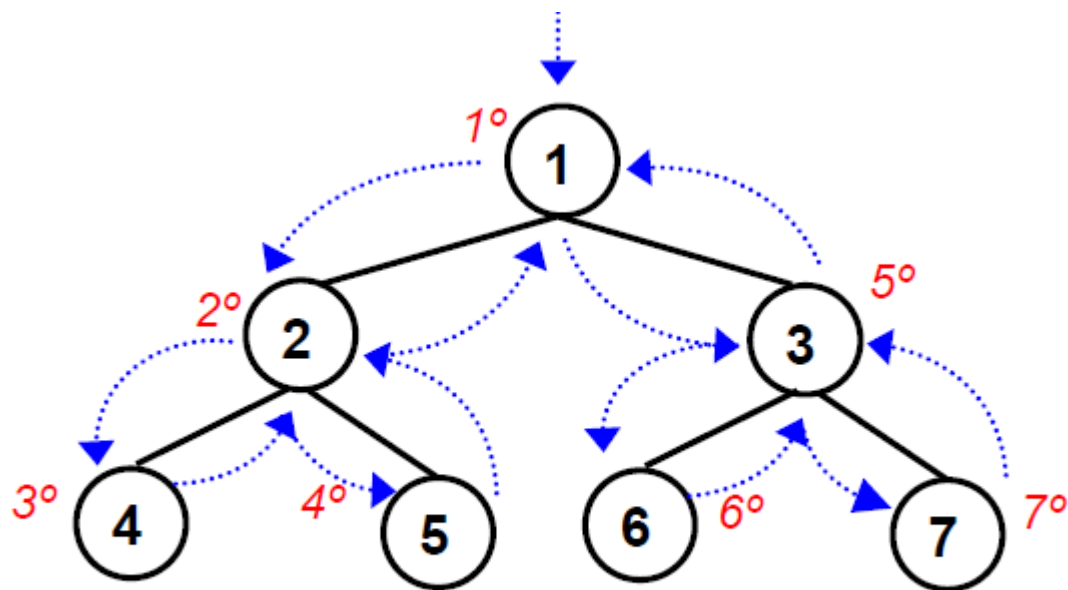
A diferença entre esses percursos é, basicamente, a ordem em que os nós são “visitados”.



INSTITUTO FEDERAL
ESPIRITO SANTO

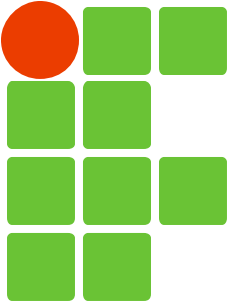
CAMPUS COLATINA

Pré-Ordem ou Profundidade



Resultado do Percurso: 1,2,4,5,3,6,7

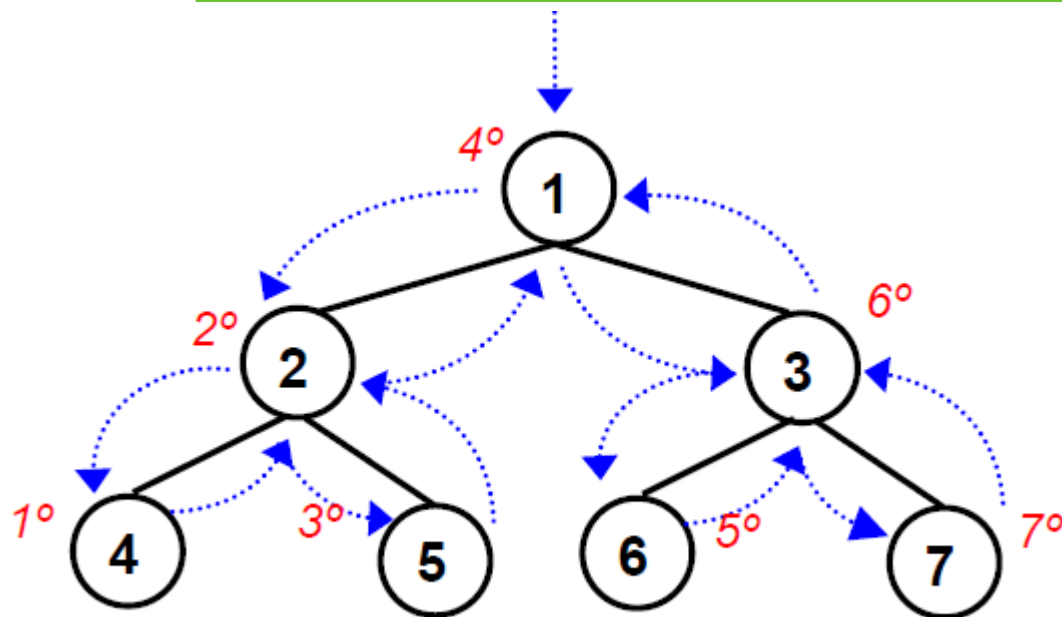
1. Mostra o valor do nó;
2. Visita o nó esquerdo;
3. Visita o nó direito;



INSTITUTO FEDERAL
ESPIRITO SANTO

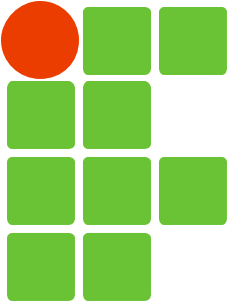
CAMPUS COLATINA

Em-Ordem ou Ordem Simétrica



Resultado do Percurso: 4,2,5,1,6,3,7

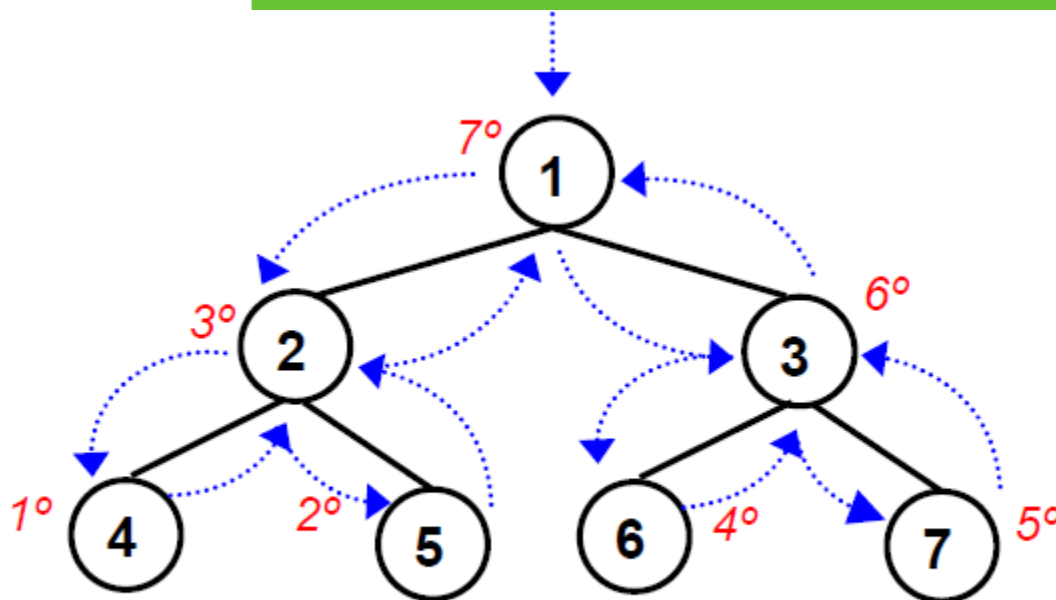
1. Visita o nó esquerdo;
2. Mostra o valor do nó;
3. Visita o nó direito;



INSTITUTO FEDERAL
ESPIRITO SANTO

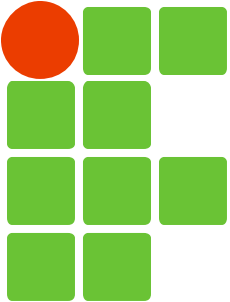
CAMPUS COLATINA

Pós-Ordem



Resultado do Percurso: 4,5,2,6,7,3,1

1. Visita o nó esquerdo;
2. Visita o nó direito;
3. Mostra o valor do nó;



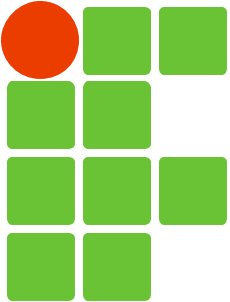
INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Remoção de Nó

3 Possibilidades

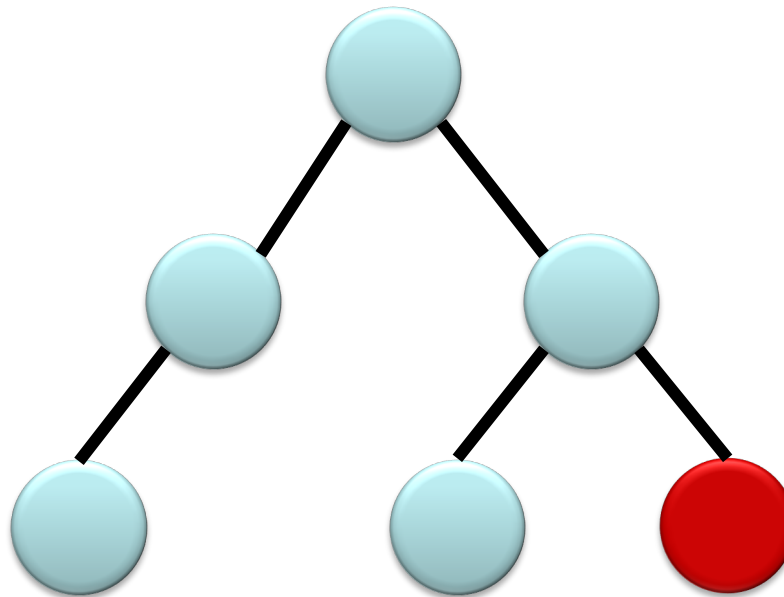
1. Não tem filho
2. Somente 1 Filho
3. 2 Filhos

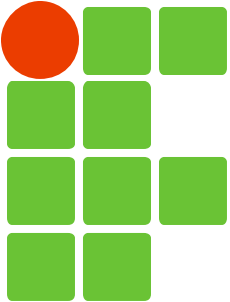


INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Remoção de Nó Sem Filhos

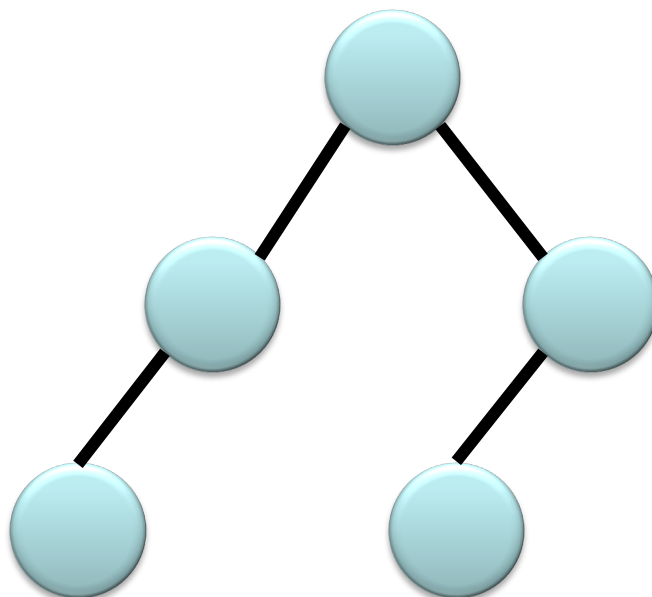


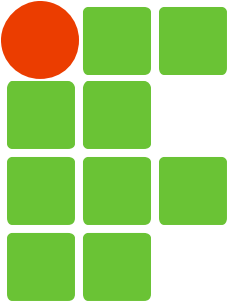


INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Remoção de Nó Sem Filhos

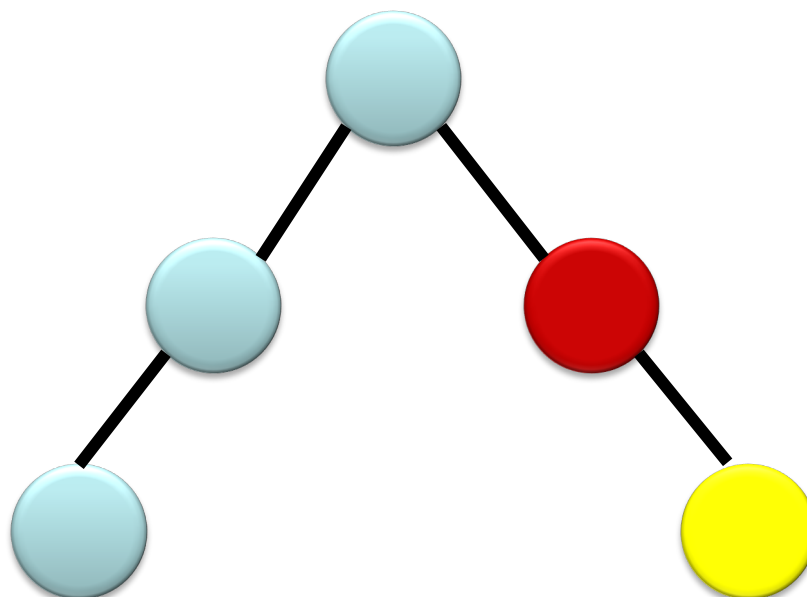


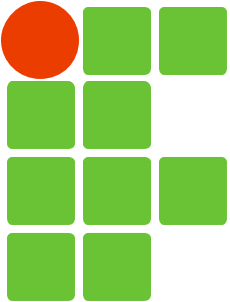


INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Remoção de Nó Com 1 Filho

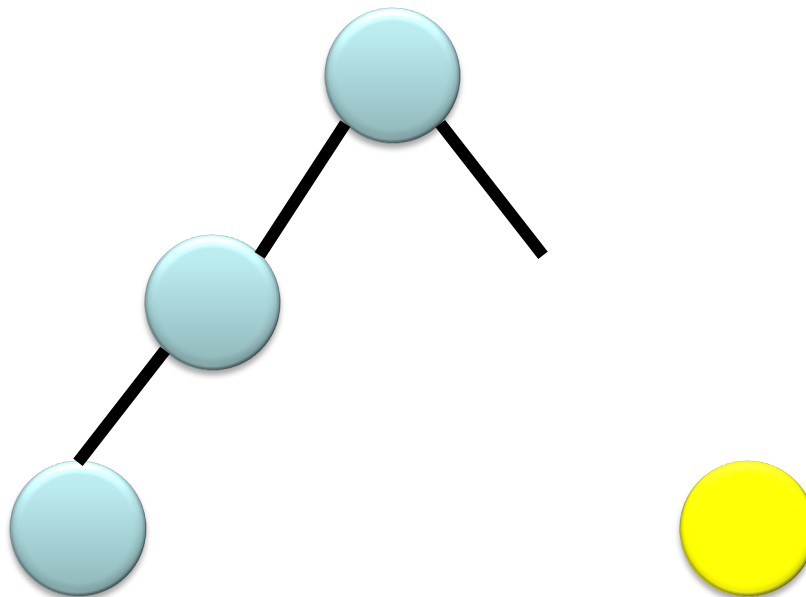


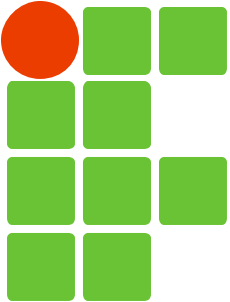


**INSTITUTO FEDERAL
ESPIRITO SANTO**

CAMPUS COLATINA

Remoção de Nó Com 1 Filho

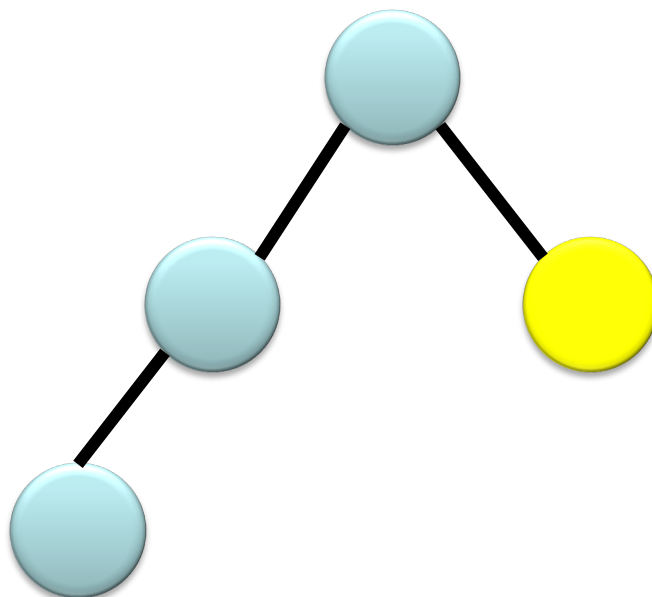


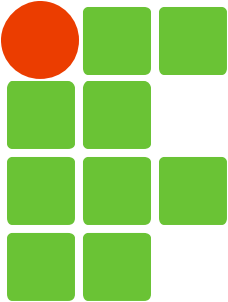


INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Remoção de Nó Com 1 Filho





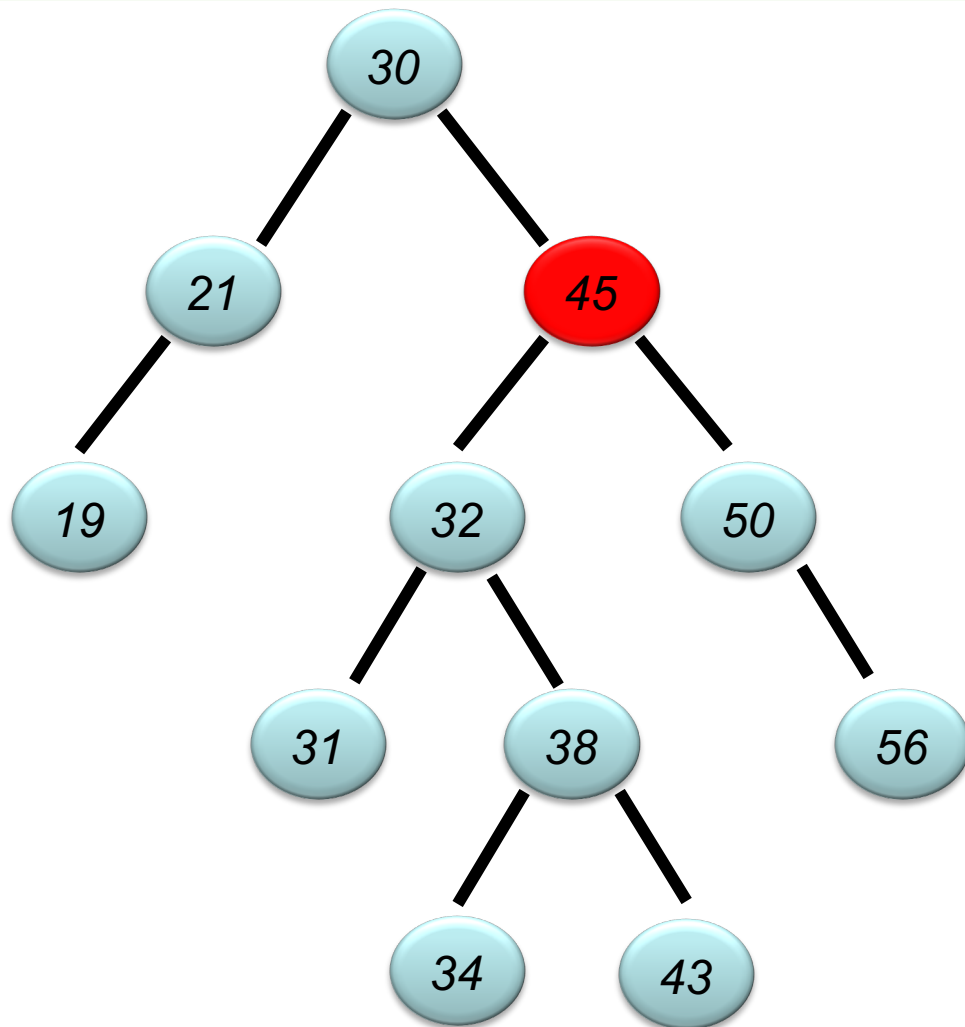
INSTITUTO FEDERAL
ESPIRITO SANTO

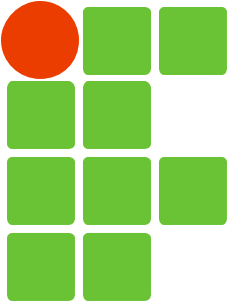
CAMPUS COLATINA

Remoção de Nó Com 2 Filhos

2 Opções:

- Máximo da árvore esquerda vai pra raiz;
- Mínimo da árvore direita vai pra raiz;





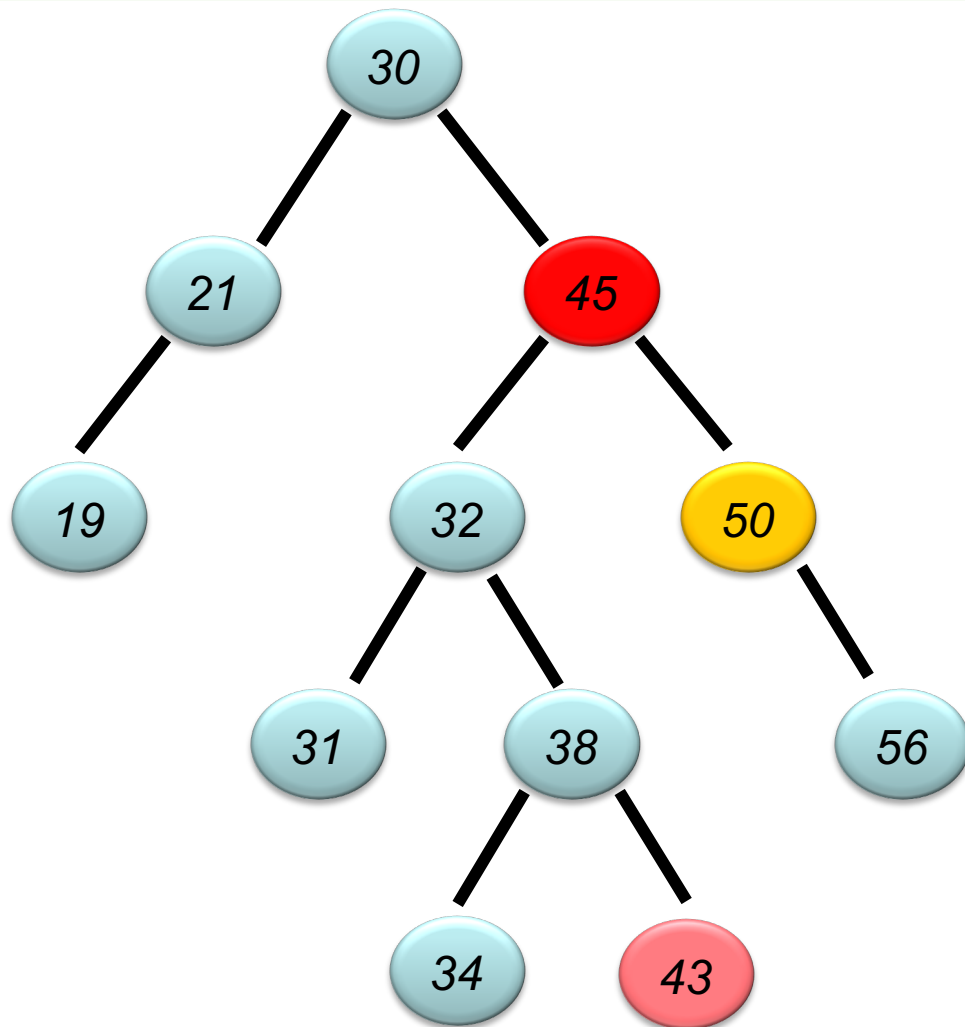
INSTITUTO FEDERAL
ESPIRITO SANTO

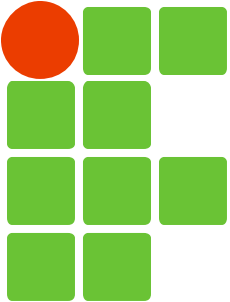
CAMPUS COLATINA

Remoção de Nó Com 2 Filhos

2 Opções:

- Máximo da árvore esquerda vai pra raiz;
- Mínimo da árvore direita vai pra raiz;





INSTITUTO FEDERAL
ESPIRITO SANTO

CAMPUS COLATINA

Próximos passos

- Implementar uma biblioteca de árvores para armazenar estruturas de contatos telefônicos, como a implementada no trabalho de listas.
- Assim como fizemos no trabalho de lista, a estrutura deverá ser indexada pelo nome do contato.
- Sua biblioteca deve ter métodos para inserir um elemento na árvore, remover um elemento, pesquisar por um elemento, imprimir os 3 caminhamentos clássicos