

Unidade V: Árvore Binária - Inserção



PUC Minas

Instituto de Ciências Exatas e Informática
Departamento de Ciência da Computação

Agenda

- Funcionamento básico
- Exemplo
- Inserção em Java com retorno de referência
- Inserção em Java com passagem de pai
- Análise de complexidade

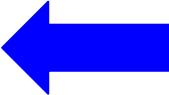
Agenda

- **Funcionamento básico** ←
- Exemplo
- Inserção em Java com retorno de referência
- Inserção em Java com passagem de pai
- Análise de complexidade

Funcionamento Básico da Inserção

- (1) Se a raiz estiver vazia, insere-se o elemento nela
- (2) Senão, se o novo elemento for **menor** que o da raiz, chama-se recursivamente a inserção para a subárvore da **esquerda**
- (3) Senão, se o novo elemento for **maior** que o da raiz, chama-se recursivamente a inserção para a subárvore da **direita**
- (4) Senão, se o novo elemento for **igual** ao da raiz, não inserir um elemento repetido

Agenda

- Funcionamento básico
- **Exemplo** 
- Inserção em Java com retorno de referência
- Inserção em Java com passagem de pai
- Análise de complexidade

Exemplo

- Inserir, na ordem, os elementos 3, 5, 1, 8, 2, 4, 7 e 6

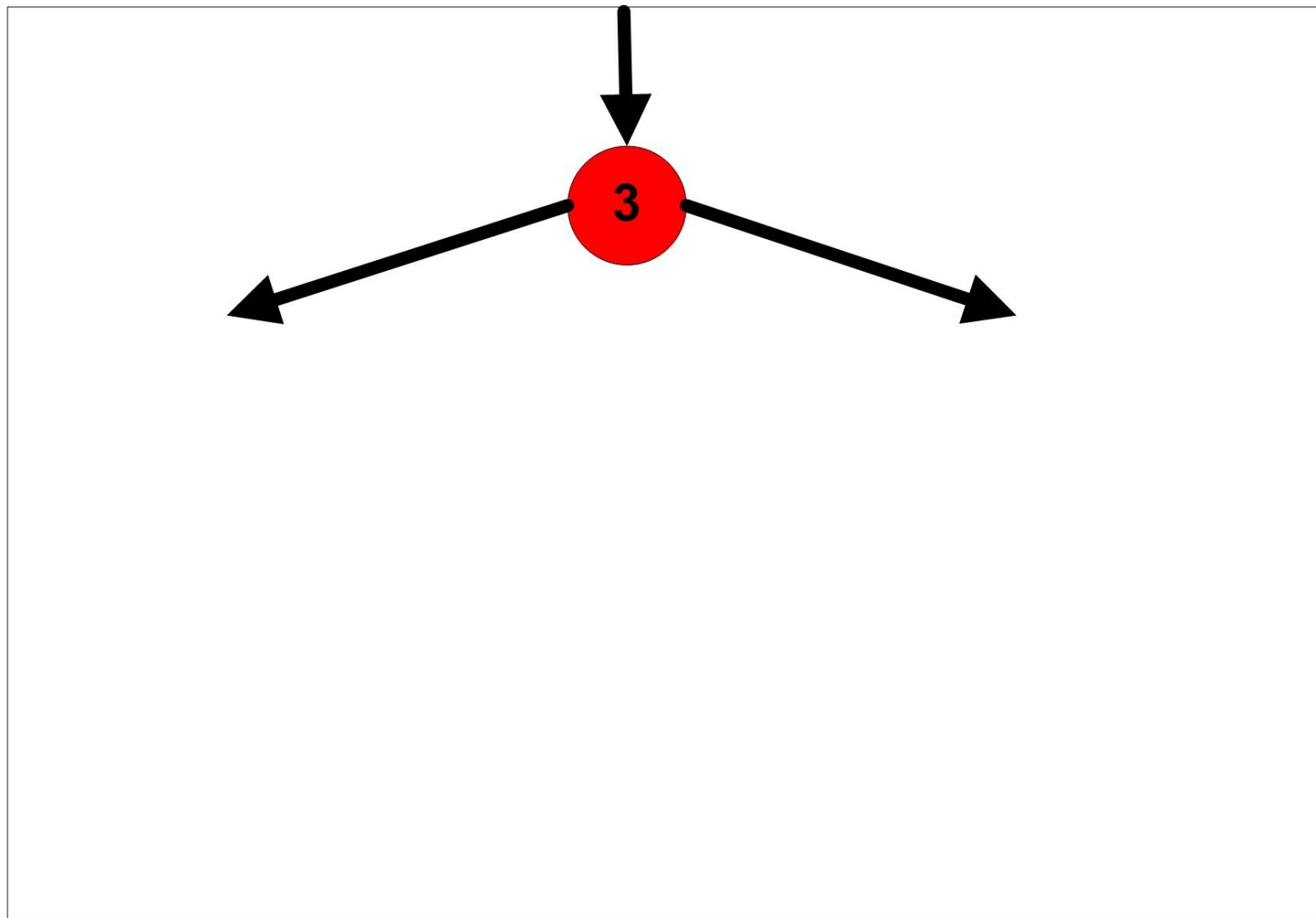
Exemplo

- Inserir, na ordem, os elementos 3, 5, 1, 8, 2, 4, 7 e 6



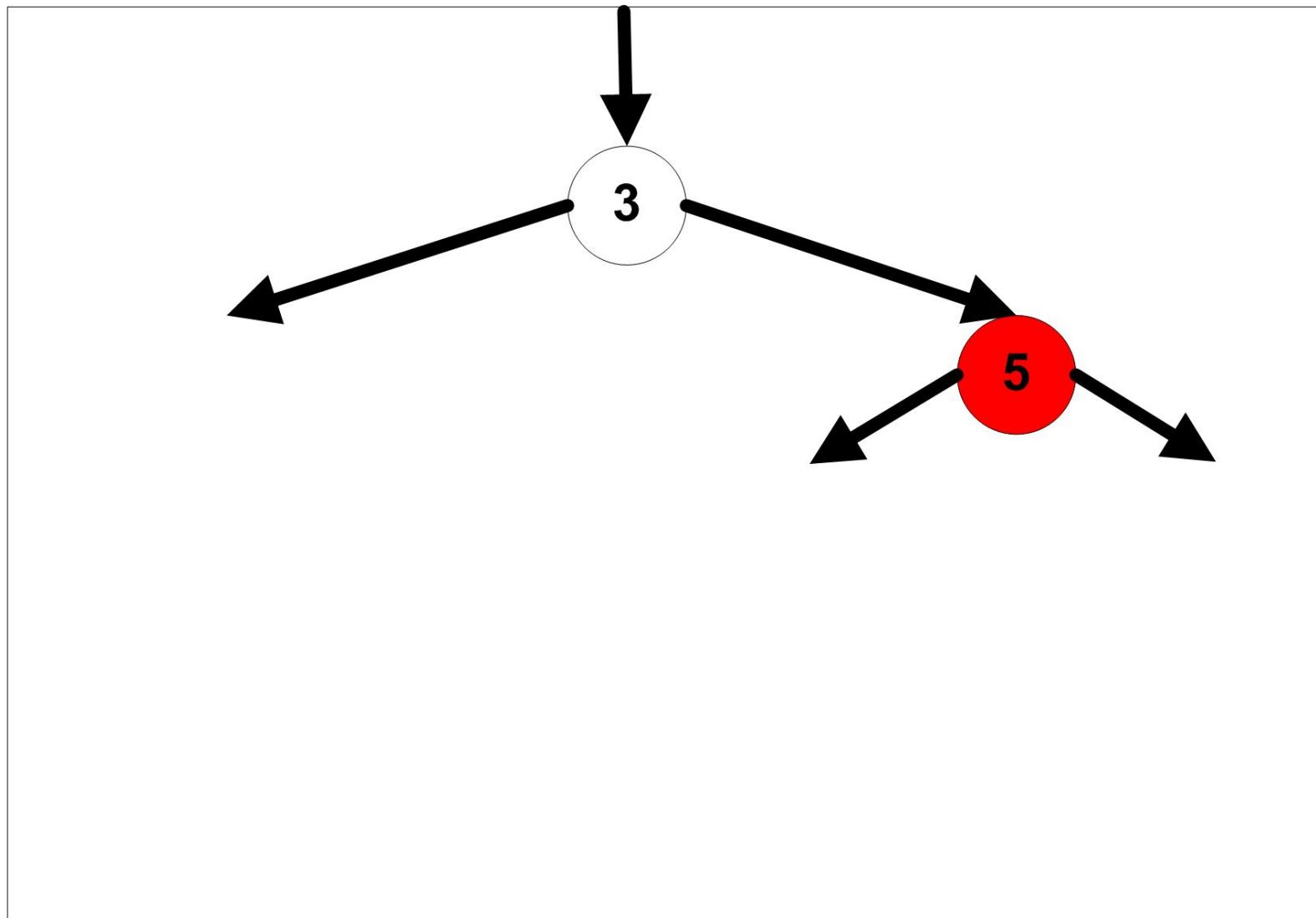
Exemplo

- Inserir, na ordem, os elementos **3, 5, 1, 8, 2, 4, 7 e 6**



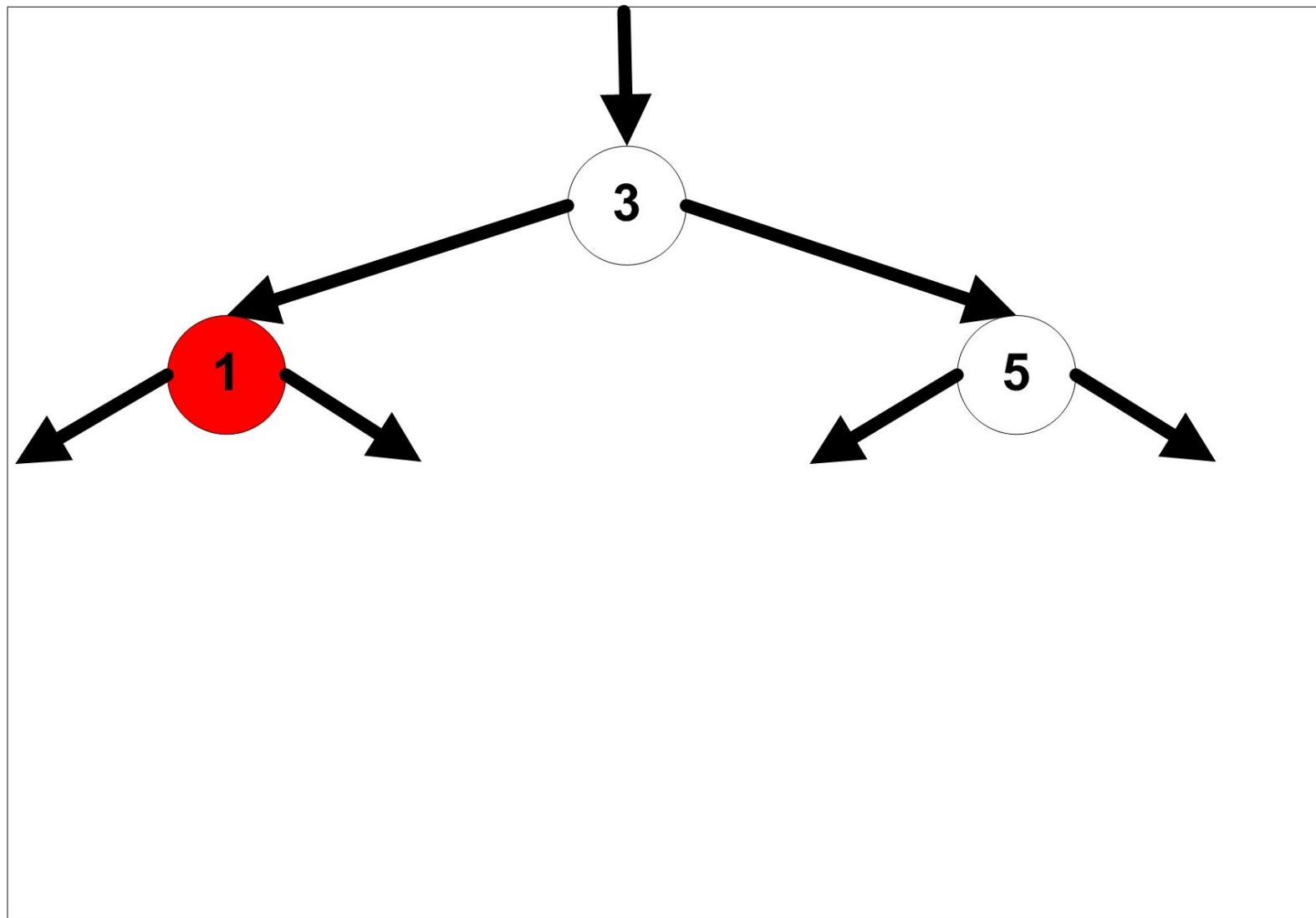
Exemplo

- Inserir, na ordem, os elementos 3, **5**, 1, 8, 2, 4, 7 e 6



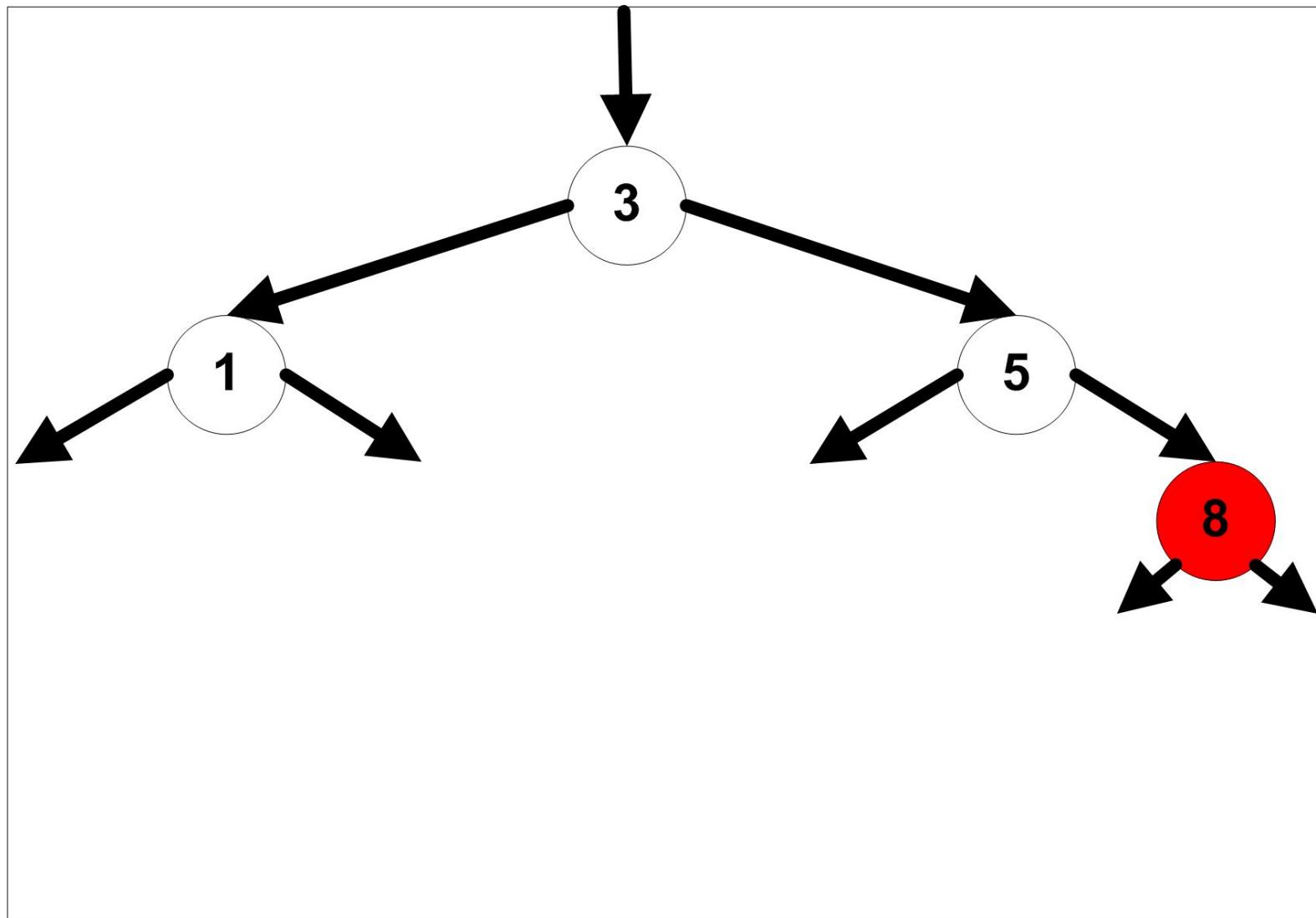
Exemplo

- Inserir, na ordem, os elementos 3, 5, **1**, 8, 2, 4, 7 e 6



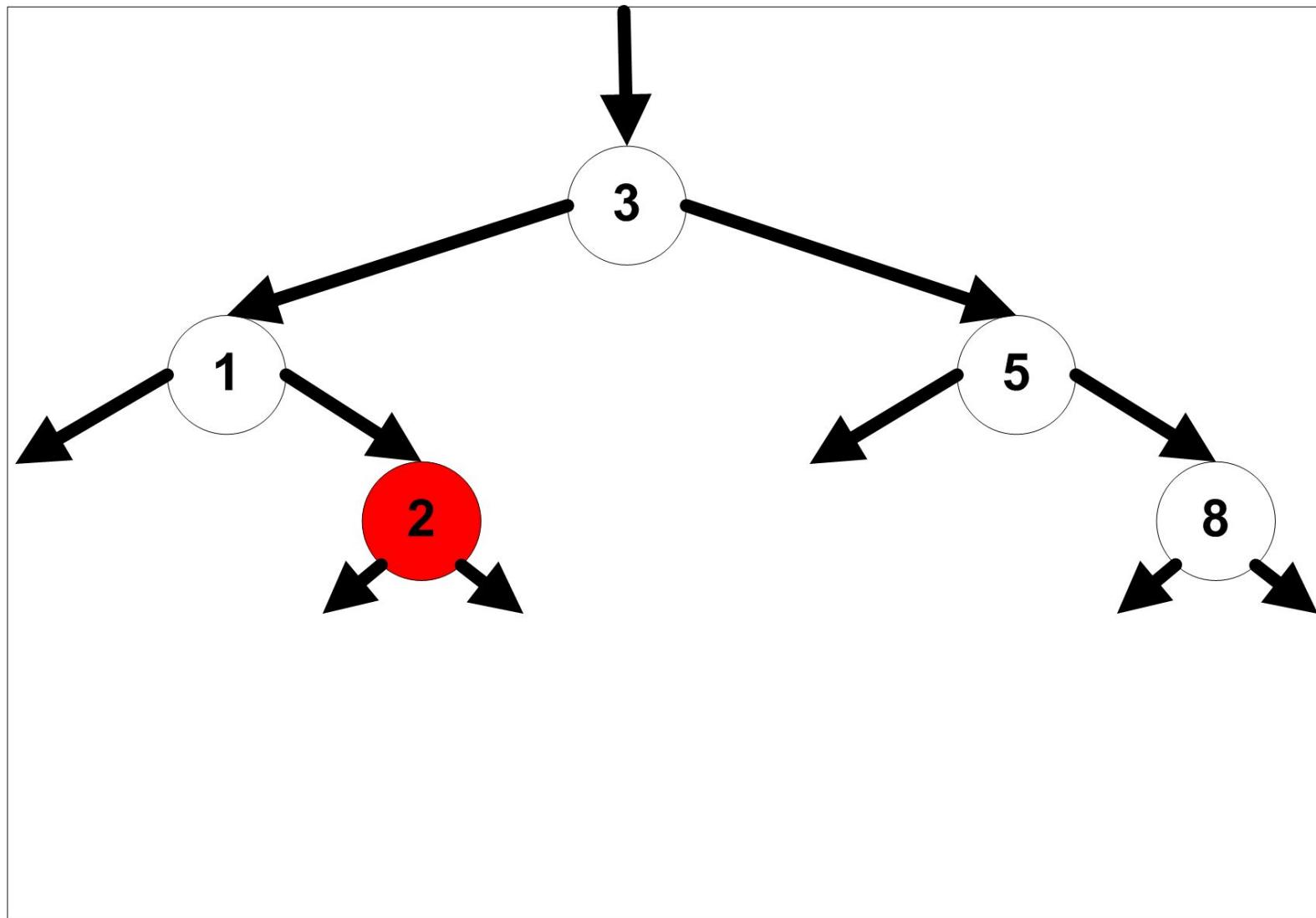
Exemplo

- Inserir, na ordem, os elementos 3, 5, 1, **8**, 2, 4, 7 e 6



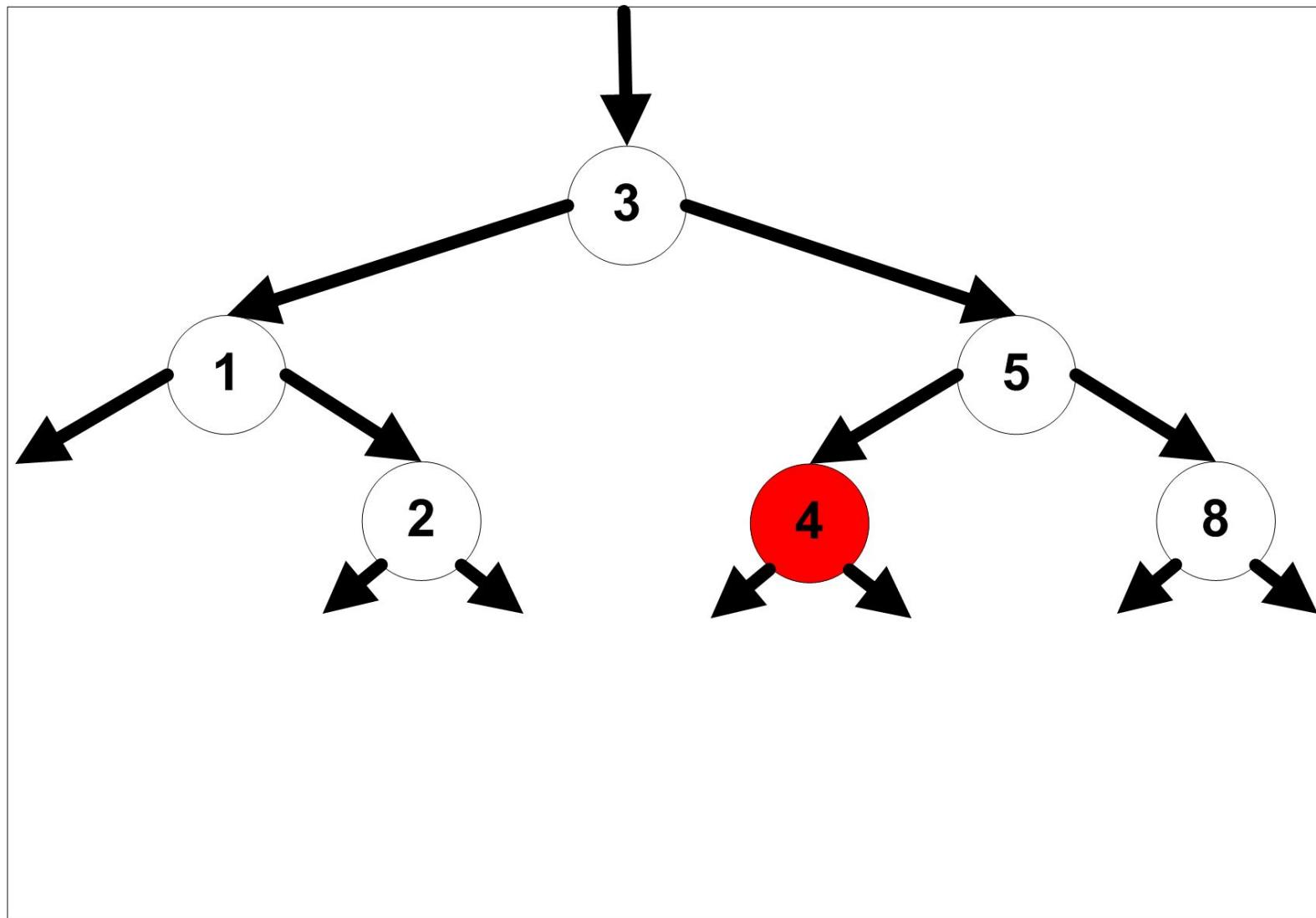
Exemplo

- Inserir, na ordem, os elementos 3, 5, 1, 8, **2**, 4, 7 e 6



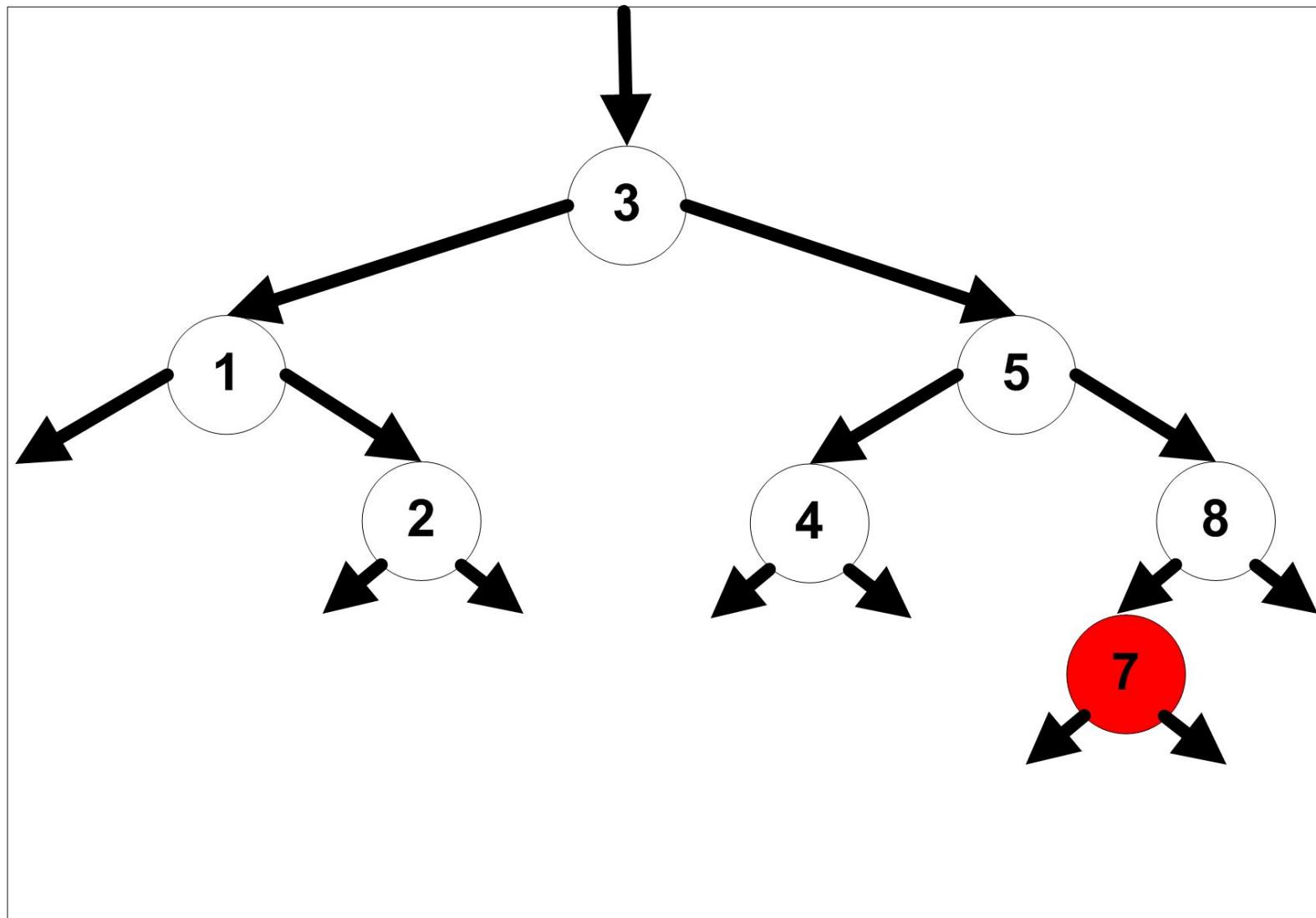
Exemplo

- Inserir, na ordem, os elementos 3, 5, 1, 8, 2, **4**, 7 e 6



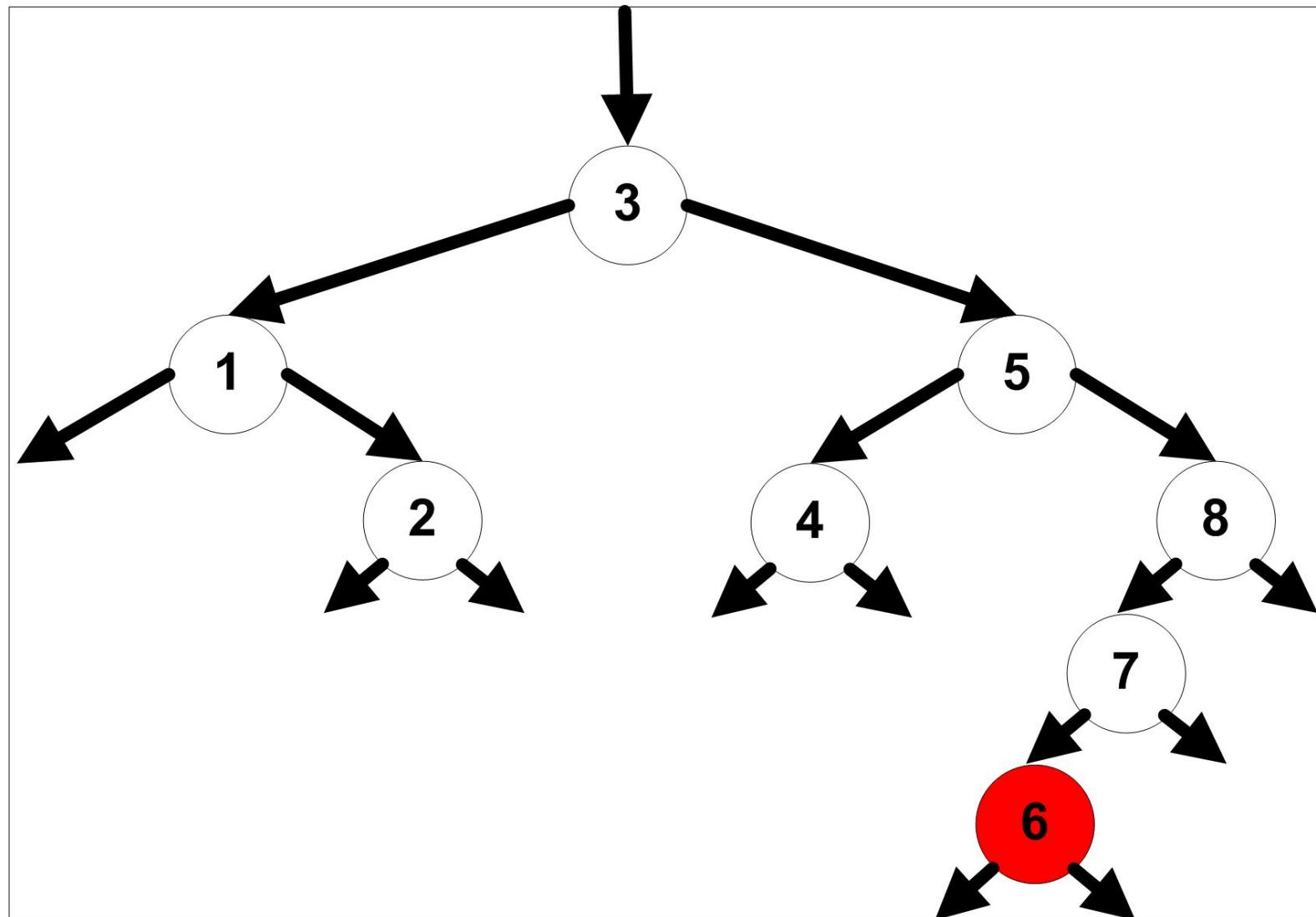
Exemplo

- Inserir, na ordem, os elementos 3, 5, 1, 8, 2, 4, **7** e 6



Exemplo

- Inserir, na ordem, os elementos 3, 5, 1, 8, 2, 4, 7 e **6**



Exercício

- Inserir, na ordem, os elementos 1, 2, 3, 4, 5, 6, 7 e 8

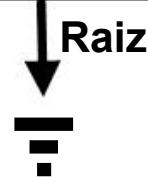
Agenda

- Funcionamento básico
- Exemplo
- **Inserção em Java com retorno de referência** 
- Inserção em Java com passagem de pai
- Análise de complexidade

Inserção em Java com Retorno de Referência

```
class ArvoreBinaria {  
    No raiz;  
    ArvoreBinaria() { raiz = null; }  
    void inserir(int x) { }  
    void inserirPai(int x) { }  
    boolean pesquisar(int x) { }  
    void caminharCentral() { }  
    void caminharPre() { }  
    void caminharPos() { }  
    void remover(int x) { }  
}
```

raiz
null



Inserção em Java com Retorno de Referência

```
class ArvoreBinaria {  
    No raiz;  
    ArvoreBinaria() { raiz = null; }  
    void inserir(int x) { }  
    void inserirPai(int x) { }  
    boolean pesquisar(int x) { }  
    void caminharCentral() { }  
    void caminharPre() { }  
    void caminharPos() { }  
    void remover(int x) { }  
}
```

raiz
null



Vamos inserir os elementos

3, 5, 1, 8, 2, 4, 7 e 6

(várias chamadas do inserir)

Inserção em Java com Retorno de Referência

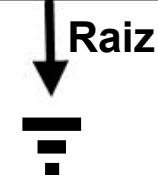
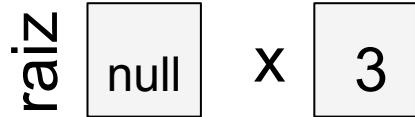
```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
```

```
    raiz = inserir(x, raiz);  
}
```

```
No inserir(int x, No i) throws Exception {
```

```
    if (i == null) {  
        i = new No(x);  
    } else if (x < i.elemento) {  
        i.esq = inserir(x, i.esq);  
    } else if (x > i.elemento) {  
        i.dir = inserir(x, i.dir);  
    } else {  
        throw new Exception("Erro!");  
    }  
    return i;  
}
```



Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {  
    raiz = inserir(x, raiz);  
}
```

```
No inserir(int x, No i) throws Exception {  
    if (i == null) {  
        i = new No(x);  
    } else if (x < i.elemento) {  
        i.esq = inserir(x, i.esq);  
    } else if (x > i.elemento) {  
        i.dir = inserir(x, i.dir);  
    } else {  
        throw new Exception("Erro!");  
    }  
    return i;  
}
```



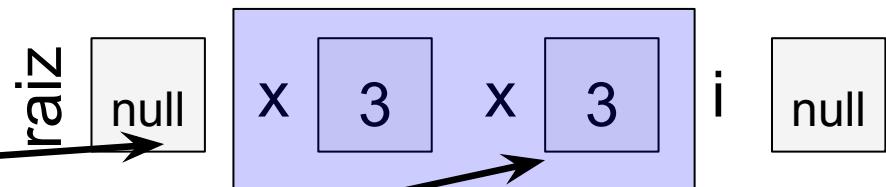
Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
```

```
if (i == null) {
    i = new No(x);
} else if (x < i.elemento) {
    i.esq = inserir(x, i.esq);
} else if (x > i.elemento) {
    i.dir = inserir(x, i.dir);
} else
    throw ...
}
return i
}
```



Cada chamada do inserir cria novas variáveis
e, por isso, temos duas variáveis com nome x

Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
```

```
if (i == null) {
    i = new No(x);
} else if (x < i.elemento) {
    i.esq = inserir(x, i.esq);
} else if (x > i.elemento) {
    i.dir = inserir(x, i.dir);
} else
    throw ...
}
return i
}
```



O valor inicial do ponteiro *i* é o mesmo
do ponteiro *raiz*, ou seja, *null*

Inserção em Java com Retorno de Referência

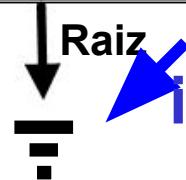
```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
```

```
if (i == null) {
    i = new No(x);
} else if (x < i.elemento) {
    i.esq = inserir(x, i.esq);
} else if (x > i.elemento) {
    i.dir = inserir(x, i.dir);
} else {
    throw new Exception("Erro!");
}
return i;
}
```

true: null == null



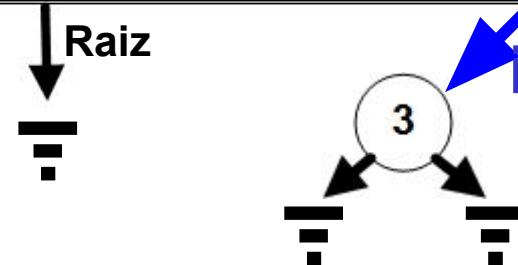
Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
    if (i == null) {
        i = new No(x);
    } else if (x < i.elemento) {
        i.esq = inserir(x, i.esq);
    } else if (x > i.elemento) {
        i.dir = inserir(x, i.dir);
    } else {
        throw new Exception("Erro!");
    }
    return i;
}
```

raiz null x 3 x 3 i n(3)



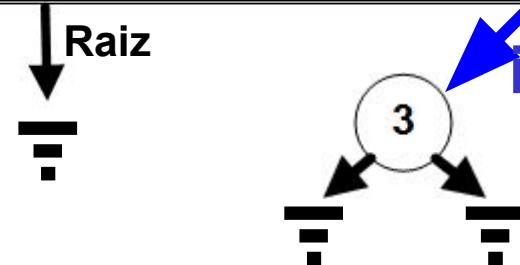
Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
    if (i == null) {
        i = new No(x);
    } else if (x < i.elemento) {
        i.esq = inserir(x, i.esq);
    } else if (x > i.elemento) {
        i.dir = inserir(x, i.dir);
    } else {
        throw new Exception("Erro!");
    }
    return i;
}                                retorna o endereço de n(3)
```

raiz null x 3 x 3 i n(3)



Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
    if (i == null) {
        i = new No(x);
    } else if (x < i.elemento) {
        i.esq = inserir(x, i.esq);
    } else if (x > i.elemento) {
        i.dir = inserir(x, i.dir);
    } else {
        throw new Exception("Erro!");
    }
    return i;
}
```

raiz

null

 x

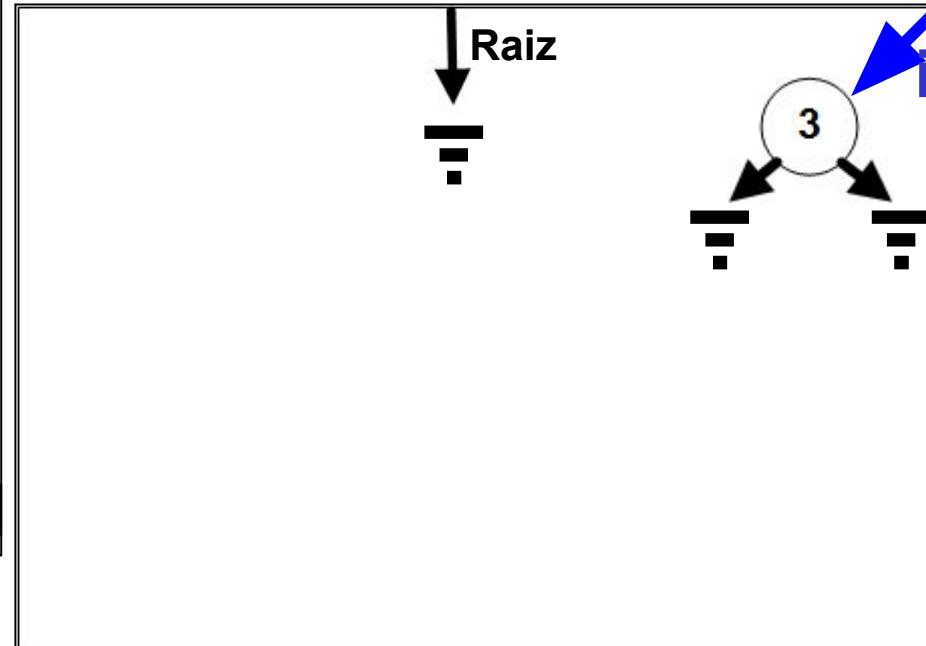
3

 x

3

 i

n(3)



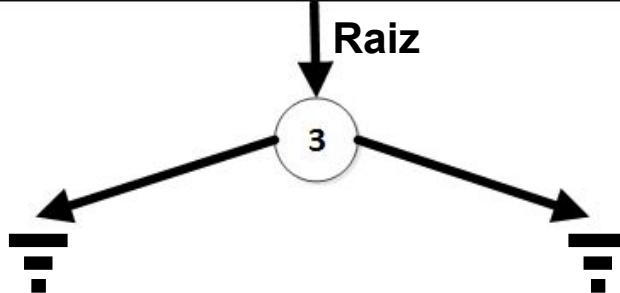
Inserção em Java com Retorno de Referência

//Inserir 3, 5, 1, 8, 2, 4, 7 e 6

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
    if (i == null) {
        i = new No(x);
    } else if (x < i.elemento) {
        i.esq = inserir(x, i.esq);
    } else if (x > i.elemento) {
        i.dir = inserir(x, i.dir);
    } else {
        throw new Exception("Erro!");
    }
    return i;
}
```

raiz n(3) x 3



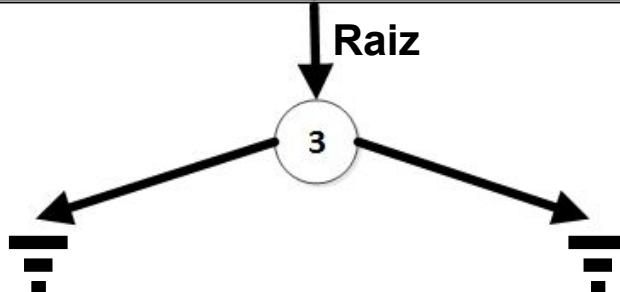
Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {  
    raiz = inserir(x, raiz);  
}
```

```
No inserir(int x, No i) throws Exception {  
    if (i == null) {  
        i = new No(x);  
    } else if (x < i.elemento) {  
        i.esq = inserir(x, i.esq);  
    } else if (x > i.elemento) {  
        i.dir = inserir(x, i.dir);  
    } else {  
        throw new Exception("Erro!");  
    }  
    return i;  
}
```

raiz
n(3)



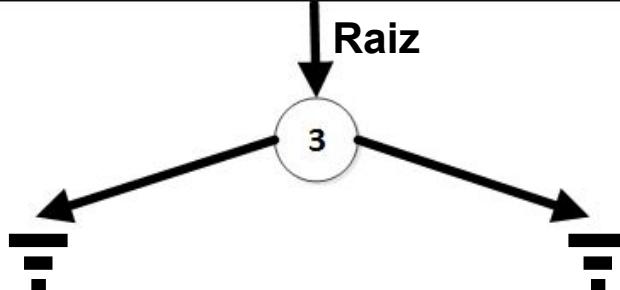
Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {  
    raiz = inserir(x, raiz);  
}
```

```
No inserir(int x, No i) throws Exception {  
    if (i == null) {  
        i = new No(x);  
    } else if (x < i.elemento) {  
        i.esq = inserir(x, i.esq);  
    } else if (x > i.elemento) {  
        i.dir = inserir(x, i.dir);  
    } else {  
        throw new Exception("Erro!");  
    }  
    return i;  
}
```

raiz n(3) X 5



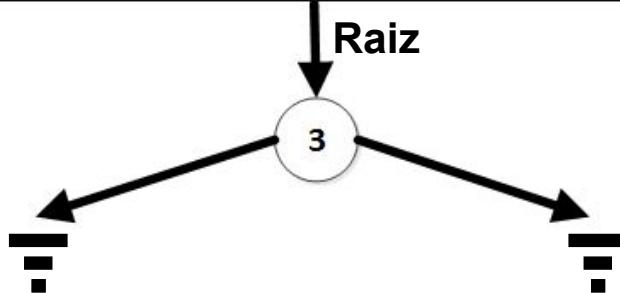
Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {  
    raiz = inserir(x, raiz);  
}
```

```
No inserir(int x, No i) throws Exception {  
    if (i == null) {  
        i = new No(x);  
    } else if (x < i.elemento) {  
        i.esq = inserir(x, i.esq);  
    } else if (x > i.elemento) {  
        i.dir = inserir(x, i.dir);  
    } else {  
        throw new Exception("Erro!");  
    }  
    return i;  
}
```

raiz n(3) X 5



Inserção em Java com Retorno de Referência

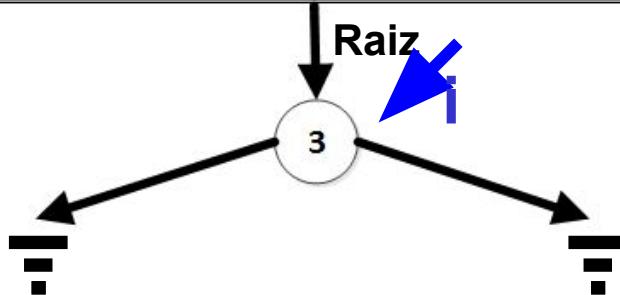
```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
```

```
if (i == null) {
    i = new No(x);
} else if (x < i.elemento) {
    i.esq = inserir(x, i.esq);
} else if (x > i.elemento) {
    i.dir = inserir(x, i.dir);
} else {
    throw new Exception("Erro!");
}
return i;
}
```

raiz n(3) x 5 x 5 i n(3)



Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

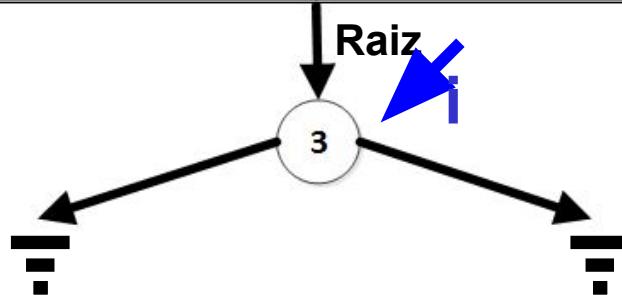
```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
```

```
if (i == null) {
    i = new No(x);
} else if (x < i.elemento) {
    i.esq = inserir(x, i.esq);
} else if (x > i.elemento) {
    i.dir = inserir(x, i.dir);
} else {
    throw new Exception("Erro!");
}
return i;
}
```

false: n(3) == null

raiz n(3) X 5 X 5 i n(3)



Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
```

```
if (i == null) {
    i = new No(x);
}
```

```
} else if (x < i.elemento) {
```

```
    i.esq = inserir(x, i.esq);
```

```
} else if (x > i.elemento) {
```

```
    i.dir = inserir(x, i.dir);
```

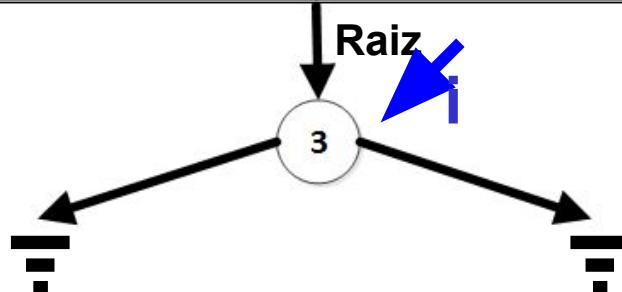
```
} else {
```

```
    throw new Exception("Erro!");
```

```
}
```

```
return i;    false: 5 < 3
```

raiz n(3) x 5 x 5 i n(3)



Inserção em Java com Retorno de Referência

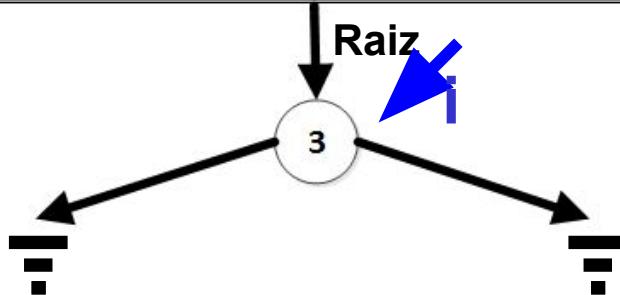
```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
    if (i == null) {
        i = new No(x);
    } else if (x < i.elemento) {
        i.esq = inserir(x, i.esq);
    } else if (x > i.elemento) {
        i.dir = inserir(x, i.dir);
    } else {
        throw new Exception("Erro!");
    }
    return i;
}
```

true: 5 > 3

raiz n(3) x 5 x 5 i n(3)



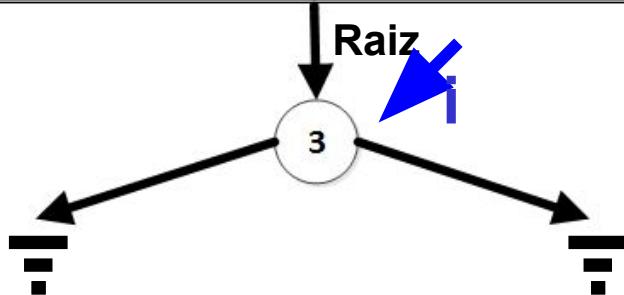
Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
    if (i == null) {
        i = new No(x);
    } else if (x < i.elemento) {
        i.esq = inserir(x, i.esq);
    } else if (x > i.elemento) {
        i.dir = inserir(x, i.dir);
    } else {
        throw new Exception("Erro!");
    }
    return i;
}
```

raiz n(3) x 5 x 5 i n(3)



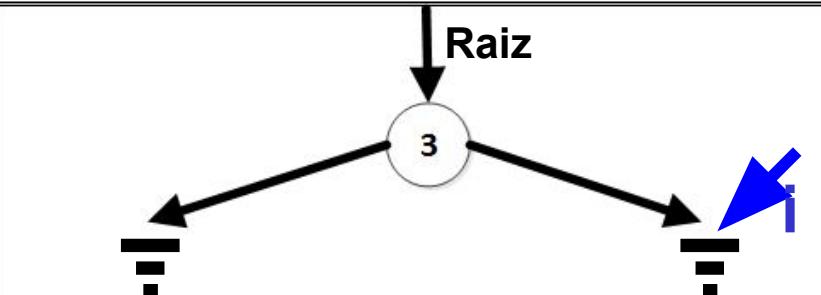
Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
```

```
if (i == null) {
    i = new No(x);
} else if (x < i.elemento) {
    i.esq = inserir(x, i.esq);
} else if (x > i.elemento) {
    i.dir = inserir(x, i.dir);
} else {
    throw new Exception("Erro!");
}
return i;
}
```

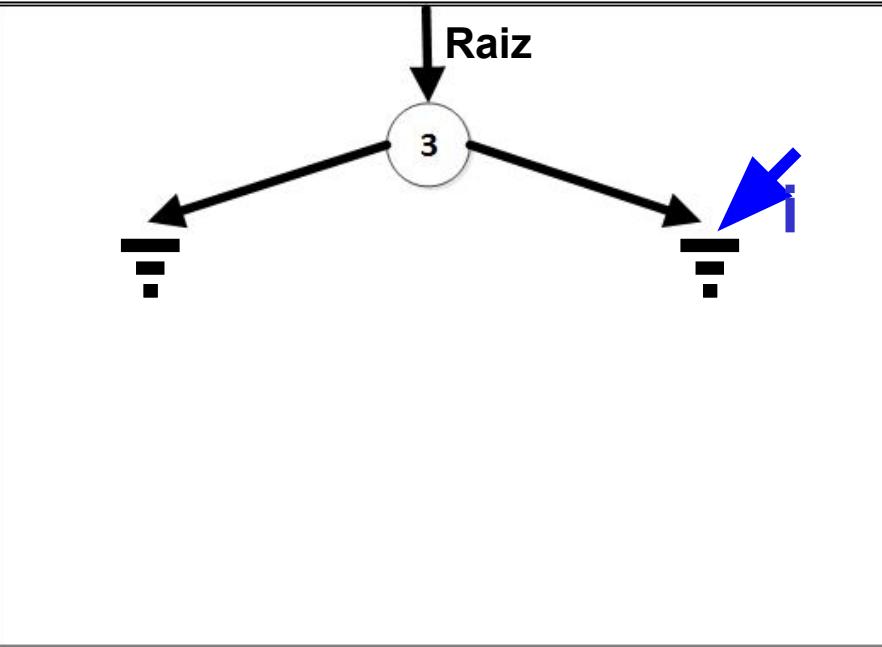
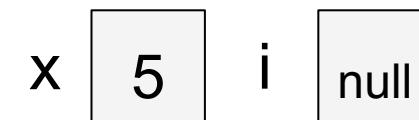


Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6

void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}

No inserir(int x, No i) throws Exception {
    if (i == null) {
        i = new No(x);
    } else if (x < i.elemento) {
        i.esq = inserir(x, i.esq);
    } else if (x > i.elemento) {
        i.dir = inserir(x, i.dir);
    } else {
        throw new Exception("Erro!");
    }
    return i;
}
true: null == null
```



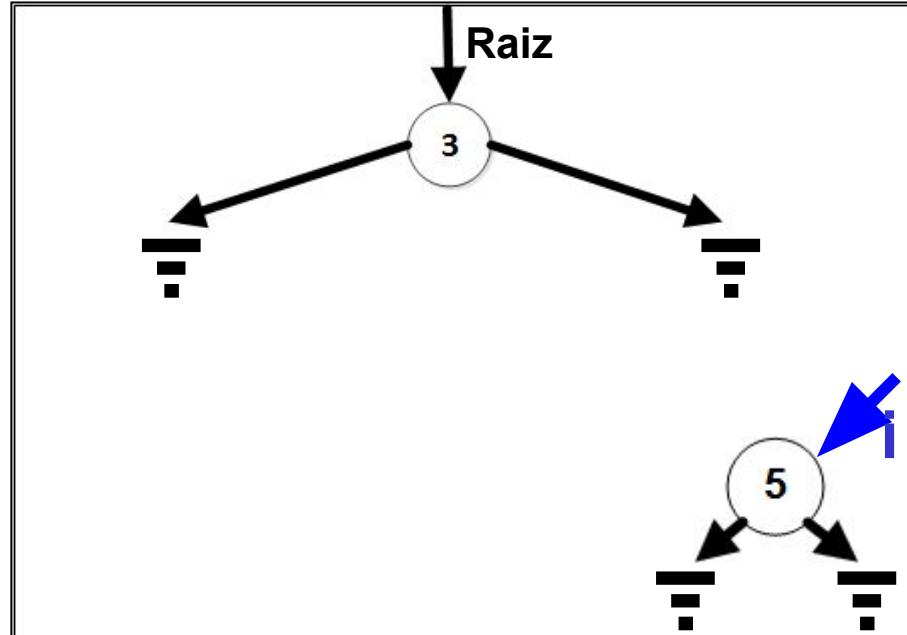
Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
```

```
if (i == null) {
    i = new No(x);
} else if (x < i.elemento) {
    i.esq = inserir(x, i.esq);
} else if (x > i.elemento) {
    i.dir = inserir(x, i.dir);
} else {
    throw new Exception("Erro!");
}
return i;
}
```

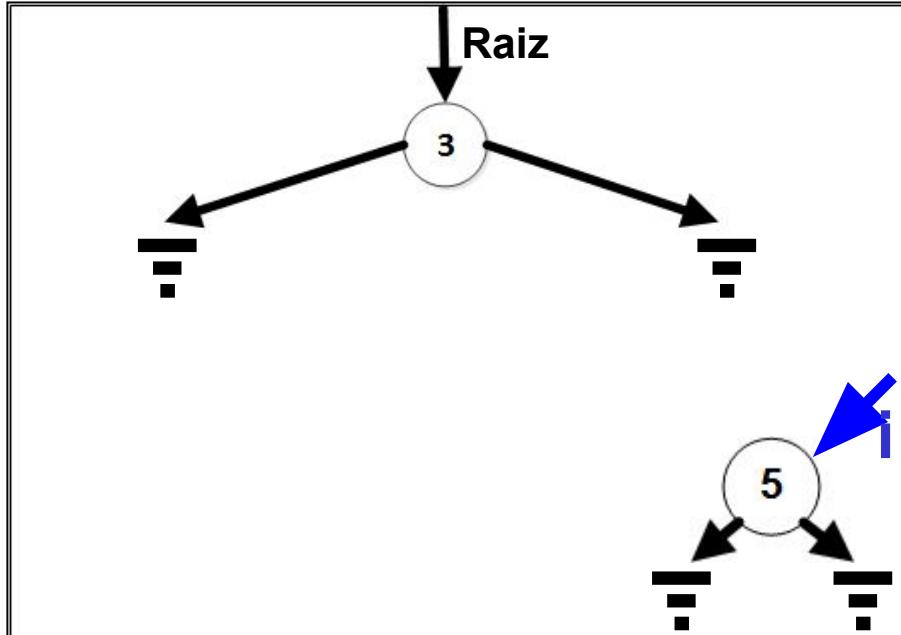
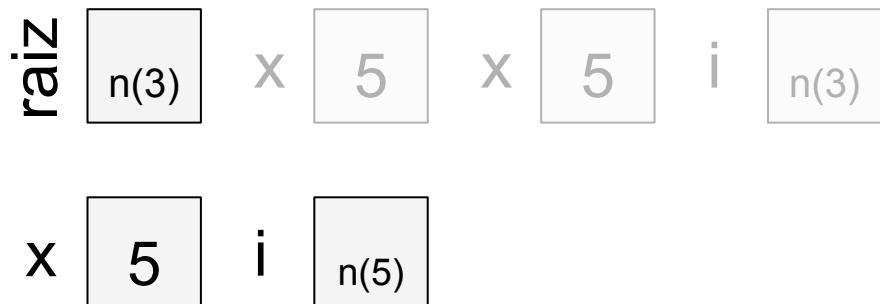


Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
    if (i == null) {
        i = new No(x);
    } else if (x < i.elemento) {
        i.esq = inserir(x, i.esq);
    } else if (x > i.elemento) {
        i.dir = inserir(x, i.dir);
    } else {
        throw new Exception("Erro!");
    }
    return i;
}                                retorna o endereço de n(5)
```

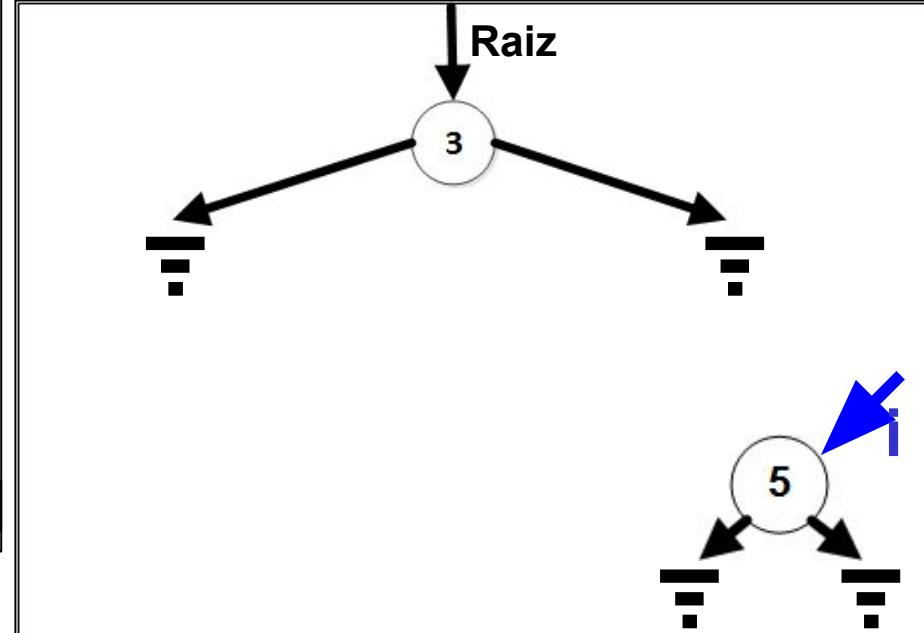


Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
    if (i == null) {
        i = new No(x);
    } else if (x < i.elemento) {
        i.esq = inserir(x, i.esq);
    } else if (x > i.elemento) {
        i.dir = inserir(x, i.dir);
    } else {
        throw new Exception("Erro!");
    }
    return i;
}
```



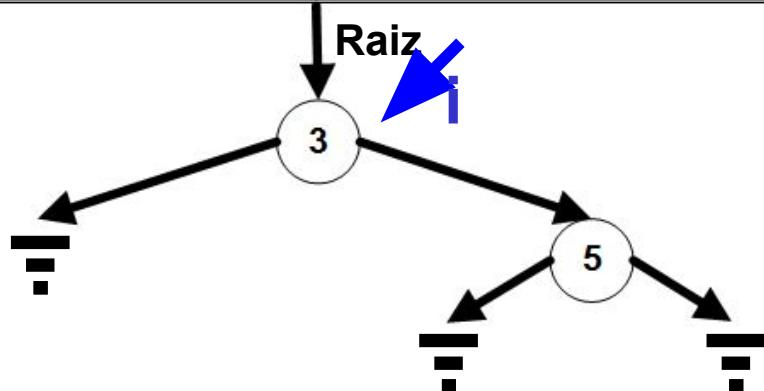
Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
    if (i == null) {
        i = new No(x);
    } else if (x < i.elemento) {
        i.esq = inserir(x, i.esq);
    } else if (x > i.elemento) {
        i.dir = inserir(x, i.dir);
    } else {
        throw new Exception("Erro!");
    }
    return i;
}
```

raiz n(3) x 5 x 5 i n(3)



Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

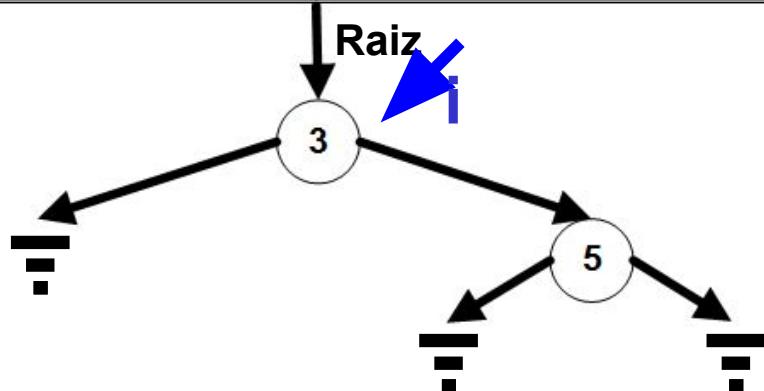
```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
    if (i == null) {
        i = new No(x);
    } else if (x < i.elemento) {
        i.esq = inserir(x, i.esq);
    } else if (x > i.elemento) {
        i.dir = inserir(x, i.dir);
    } else {
        throw new Exception("Erro!");
    }
}
```

```
return i;
```

retorna o endereço de n(3)

raiz n(3) x 5 x 5 i n(3)



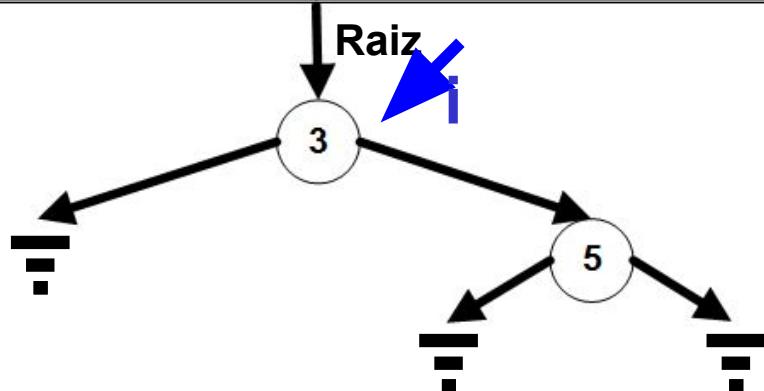
Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
    if (i == null) {
        i = new No(x);
    } else if (x < i.elemento) {
        i.esq = inserir(x, i.esq);
    } else if (x > i.elemento) {
        i.dir = inserir(x, i.dir);
    } else {
        throw new Exception("Erro!");
    }
    return i;
}
```

raiz n(3) x 5 x 5 i n(3)



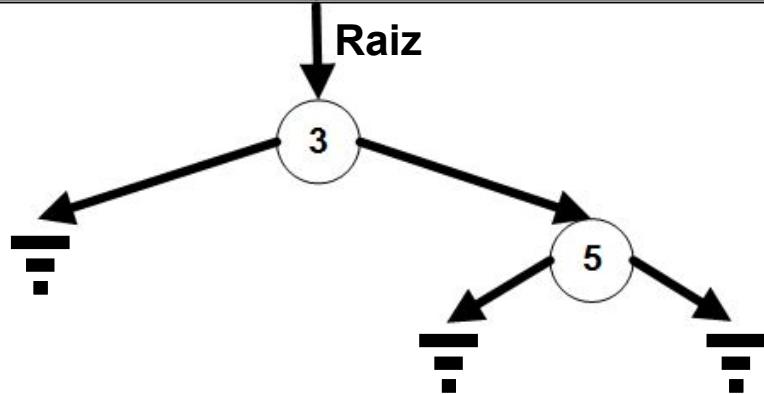
Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
    if (i == null) {
        i = new No(x);
    } else if (x < i.elemento) {
        i.esq = inserir(x, i.esq);
    } else if (x > i.elemento) {
        i.dir = inserir(x, i.dir);
    } else {
        throw new Exception("Erro!");
    }
    return i;
}
```

raiz n(3) x 5



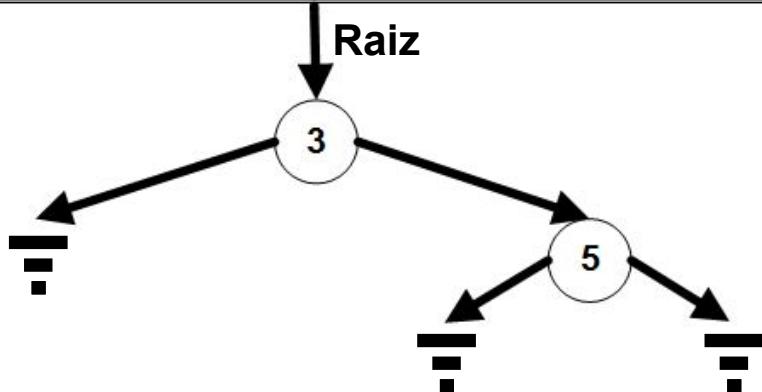
Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {  
    raiz = inserir(x, raiz);  
}
```

```
No inserir(int x, No i) throws Exception {  
    if (i == null) {  
        i = new No(x);  
    } else if (x < i.elemento) {  
        i.esq = inserir(x, i.esq);  
    } else if (x > i.elemento) {  
        i.dir = inserir(x, i.dir);  
    } else {  
        throw new Exception("Erro!");  
    }  
    return i;  
}
```

raiz
n(3)



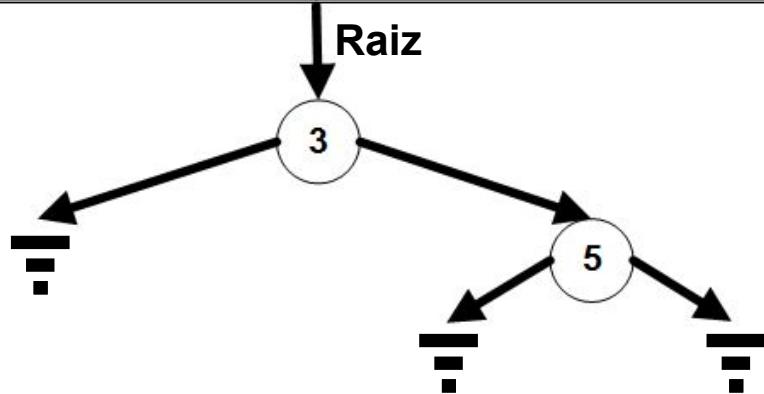
Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
    if (i == null) {
        i = new No(x);
    } else if (x < i.elemento) {
        i.esq = inserir(x, i.esq);
    } else if (x > i.elemento) {
        i.dir = inserir(x, i.dir);
    } else {
        throw new Exception("Erro!");
    }
    return i;
}
```

raiz n(3) x 1



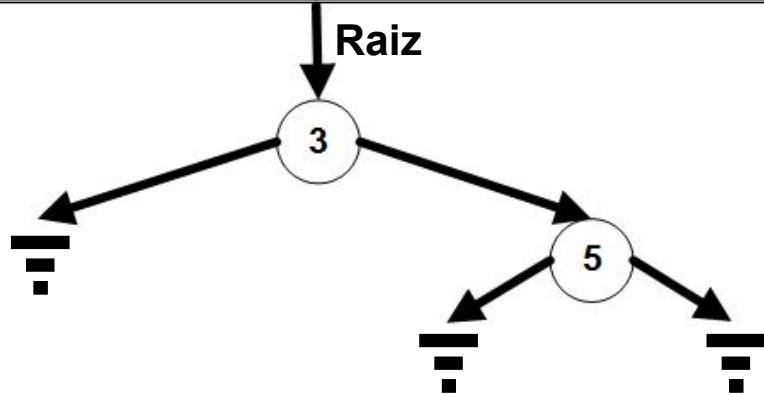
Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
    if (i == null) {
        i = new No(x);
    } else if (x < i.elemento) {
        i.esq = inserir(x, i.esq);
    } else if (x > i.elemento) {
        i.dir = inserir(x, i.dir);
    } else {
        throw new Exception("Erro!");
    }
    return i;
}
```

raiz n(3) x 1



Inserção em Java com Retorno de Referência

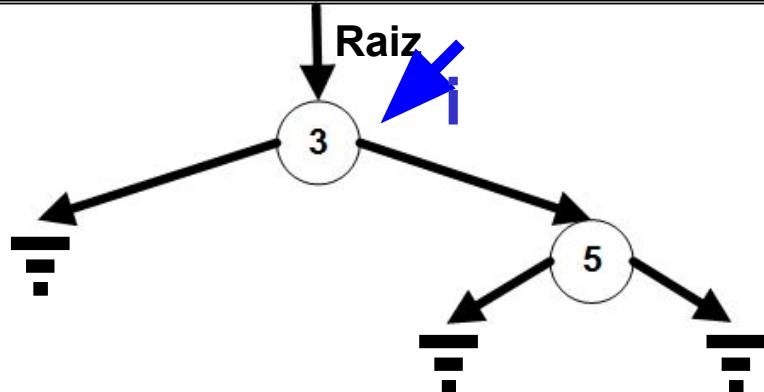
```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
```

```
if (i == null) {
    i = new No(x);
} else if (x < i.elemento) {
    i.esq = inserir(x, i.esq);
} else if (x > i.elemento) {
    i.dir = inserir(x, i.dir);
} else {
    throw new Exception("Erro!");
}
return i;
}
```

raiz n(3) x 1 x 1 i n(3)



Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

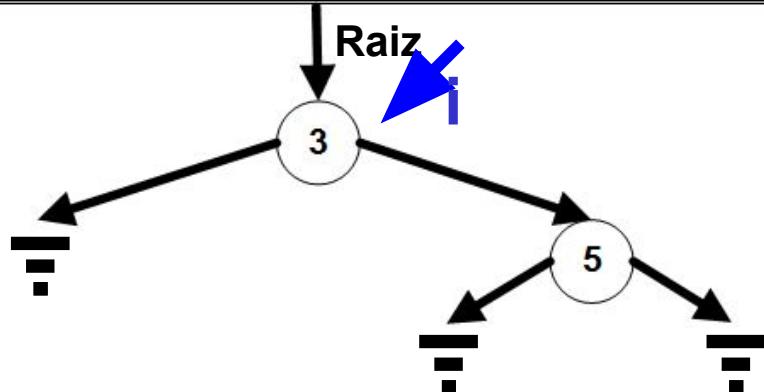
```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
```

```
if (i == null) {
    i = new No(x);
} else if (x < i.elemento) {
    i.esq = inserir(x, i.esq);
} else if (x > i.elemento) {
    i.dir = inserir(x, i.dir);
} else {
    throw new Exception("Erro!");
}
return i;
}
```

false: n(3) == null

raiz n(3) X 1 X 1 i n(3)



Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
```

```
if (i == null) {
    i = new No(x);
}
```

```
} else if (x < i.elemento) {
```

```
    i.esq = inserir(x, i.esq);
```

```
} else if (x > i.elemento) {
```

```
    i.dir = inserir(x, i.dir);
```

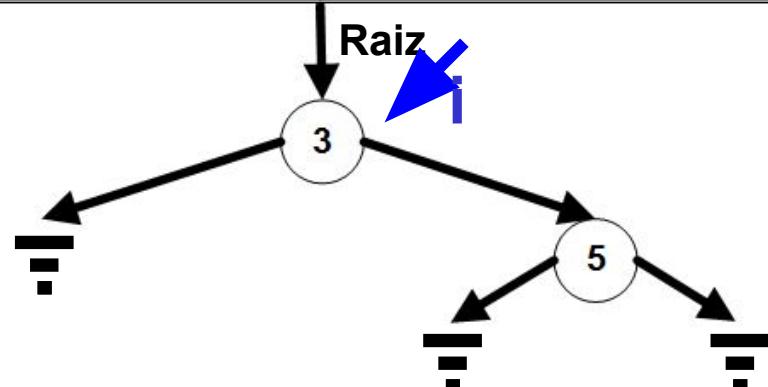
```
} else {
```

```
    throw new Exception("Erro!");
```

```
}
```

```
return i;    true: 1 < 3
}
```

raiz n(3) x 1 x 1 i n(3)



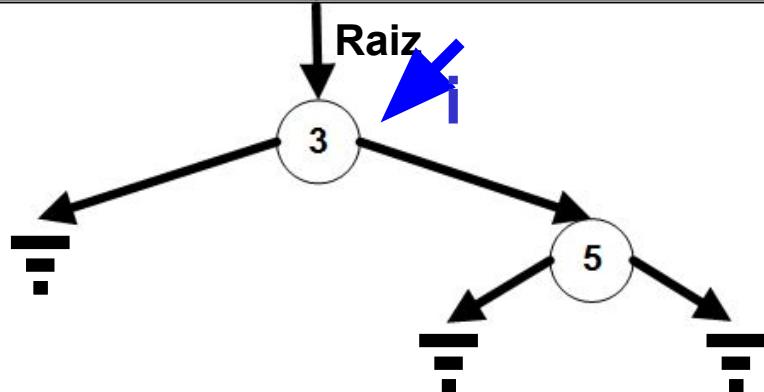
Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
    if (i == null) {
        i = new No(x);
    } else if (x < i.elemento) {
        i.esq = inserir(x, i.esq);
    } else if (x > i.elemento) {
        i.dir = inserir(x, i.dir);
    } else {
        throw new Exception("Erro!");
    }
    return i;
}
```

raiz n(3) x 1 x 1 i n(3)



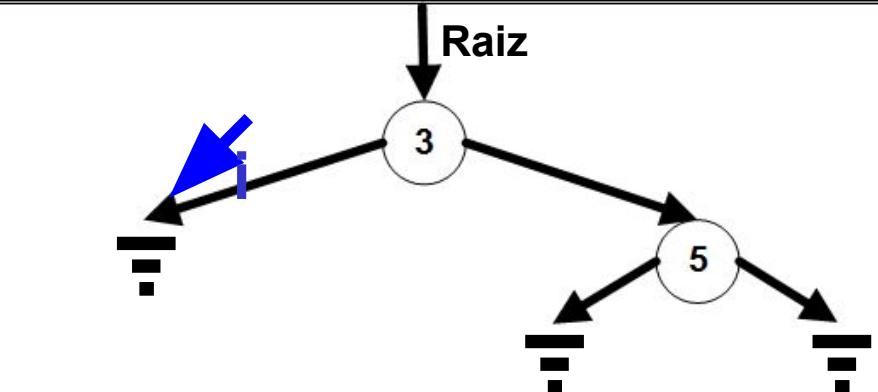
Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
```

```
if (i == null) {
    i = new No(x);
} else if (x < i.elemento) {
    i.esq = inserir(x, i.esq);
} else if (x > i.elemento) {
    i.dir = inserir(x, i.dir);
} else {
    throw new Exception("Erro!");
}
return i;
}
```

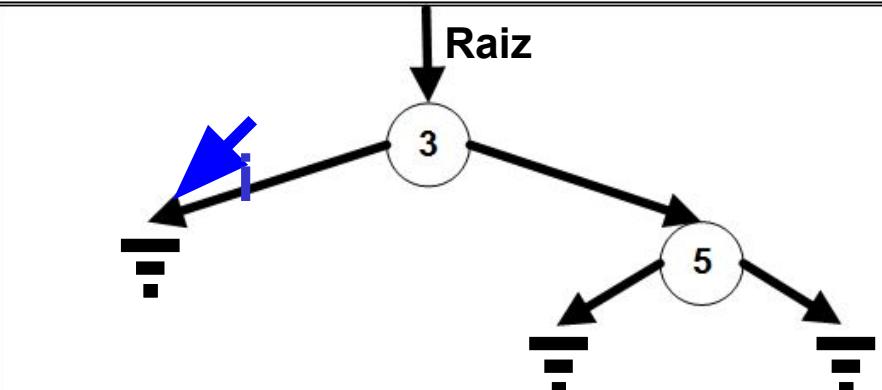
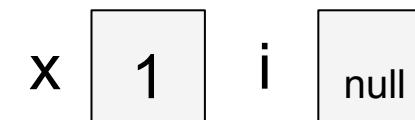


Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6

void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}

No inserir(int x, No i) throws Exception {
    if (i == null) {
        i = new No(x);
    } else if (x < i.elemento) {
        i.esq = inserir(x, i.esq);
    } else if (x > i.elemento) {
        i.dir = inserir(x, i.dir);
    } else {
        throw new Exception("Erro!");
    }
    return i;
}
true: null == null
```



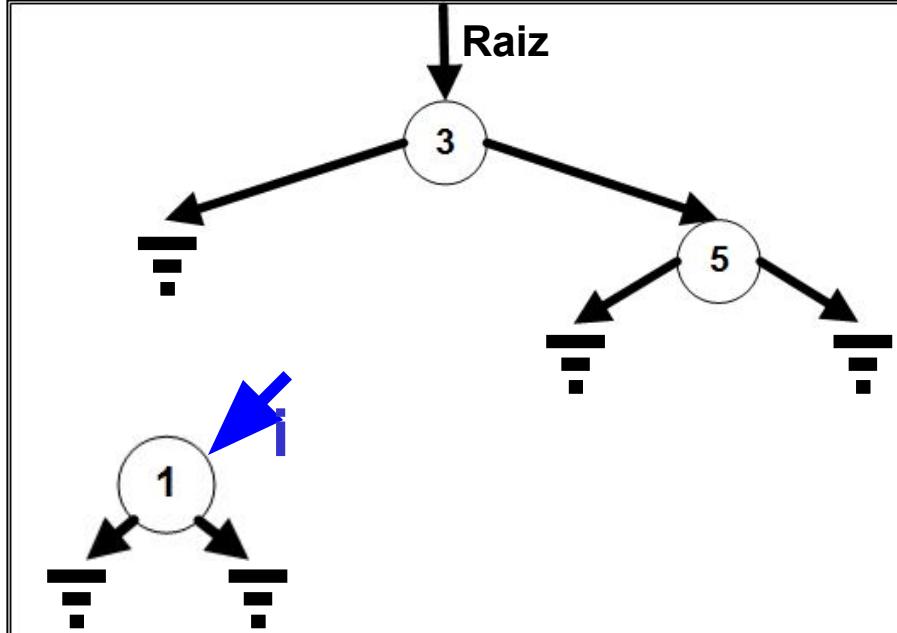
Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
```

```
if (i == null) {
    i = new No(x);
} else if (x < i.elemento) {
    i.esq = inserir(x, i.esq);
} else if (x > i.elemento) {
    i.dir = inserir(x, i.dir);
} else {
    throw new Exception("Erro!");
}
return i;
}
```



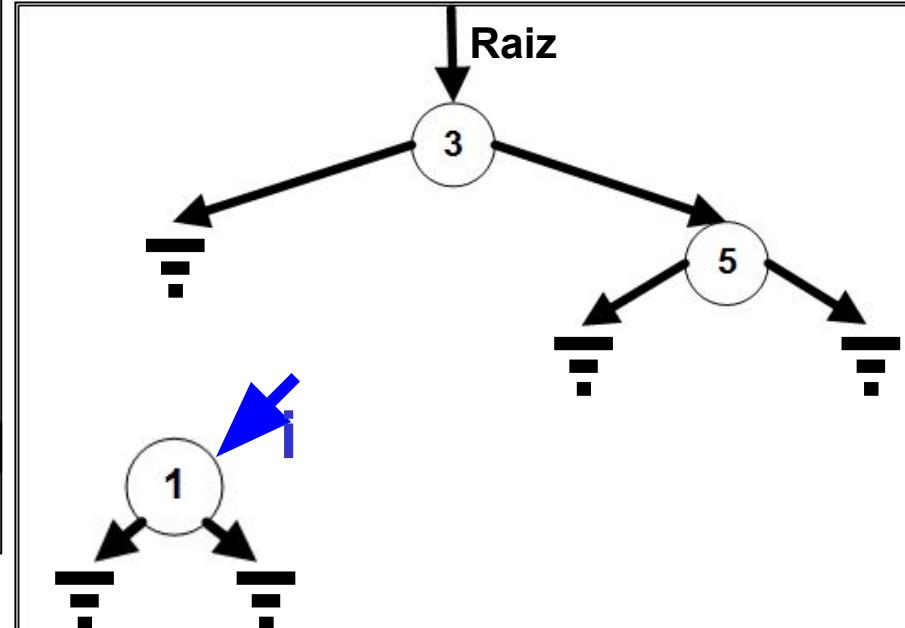
Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
    if (i == null) {
        i = new No(x);
    } else if (x < i.elemento) {
        i.esq = inserir(x, i.esq);
    } else if (x > i.elemento) {
        i.dir = inserir(x, i.dir);
    } else {
        throw new Exception("Erro!");
    }
}
```

return i; retorna o endereço de n(1)

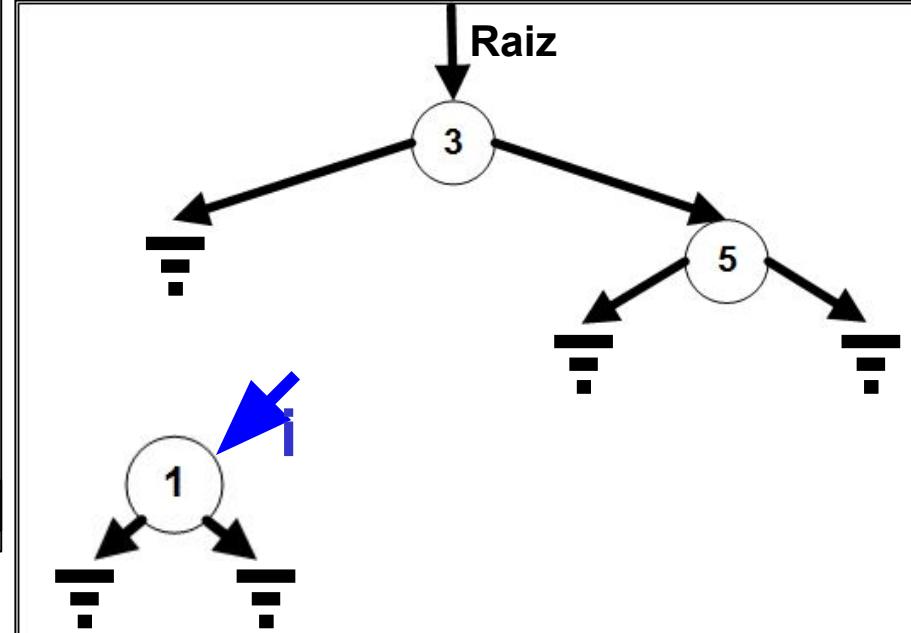


Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
    if (i == null) {
        i = new No(x);
    } else if (x < i.elemento) {
        i.esq = inserir(x, i.esq);
    } else if (x > i.elemento) {
        i.dir = inserir(x, i.dir);
    } else {
        throw new Exception("Erro!");
    }
    return i;
}
```



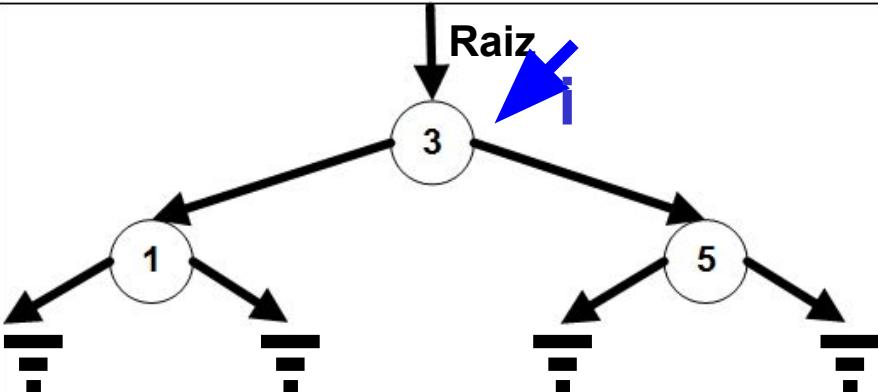
Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
    if (i == null) {
        i = new No(x);
    } else if (x < i.elemento) {
        i.esq = inserir(x, i.esq);
    } else if (x > i.elemento) {
        i.dir = inserir(x, i.dir);
    } else {
        throw new Exception("Erro!");
    }
    return i;
}
```

raiz n(3) x 1 x 1 i n(3)



Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

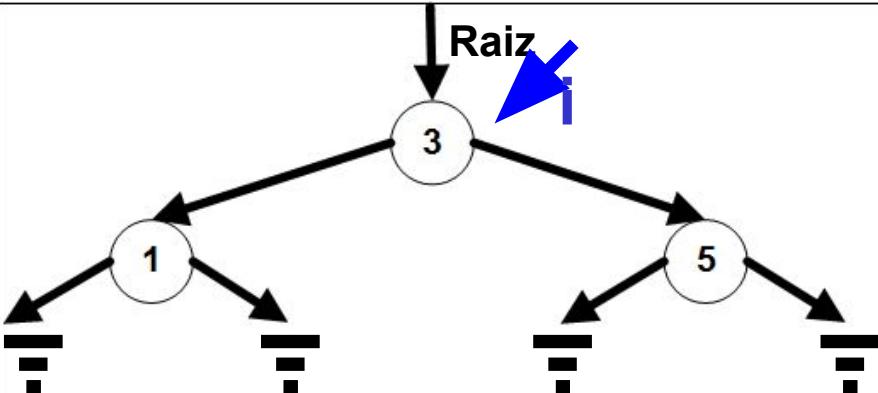
```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
    if (i == null) {
        i = new No(x);
    } else if (x < i.elemento) {
        i.esq = inserir(x, i.esq);
    } else if (x > i.elemento) {
        i.dir = inserir(x, i.dir);
    } else {
        throw new Exception("Erro!");
    }
}
```

```
return i;
```

retorna o endereço de n(3)

raiz n(3) x 1 x 1 i n(3)



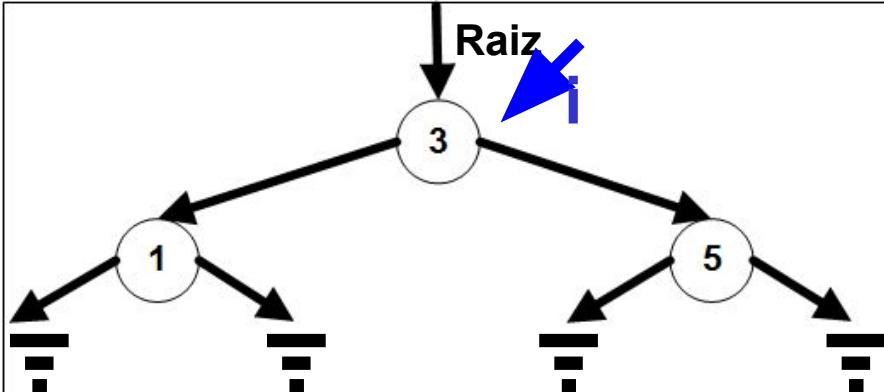
Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
    if (i == null) {
        i = new No(x);
    } else if (x < i.elemento) {
        i.esq = inserir(x, i.esq);
    } else if (x > i.elemento) {
        i.dir = inserir(x, i.dir);
    } else {
        throw new Exception("Erro!");
    }
    return i;
}
```

raiz n(3) x 1 x 1 i n(3)



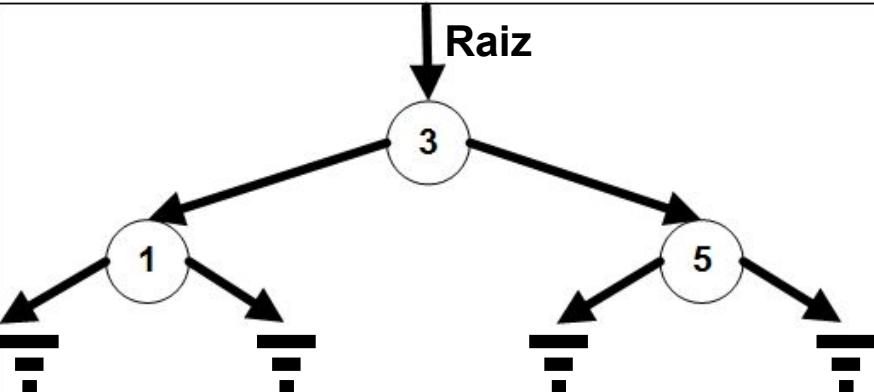
Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {
    raiz = inserir(x, raiz);
}
```

```
No inserir(int x, No i) throws Exception {
    if (i == null) {
        i = new No(x);
    } else if (x < i.elemento) {
        i.esq = inserir(x, i.esq);
    } else if (x > i.elemento) {
        i.dir = inserir(x, i.dir);
    } else {
        throw new Exception("Erro!");
    }
    return i;
}
```

raiz n(3) X 1



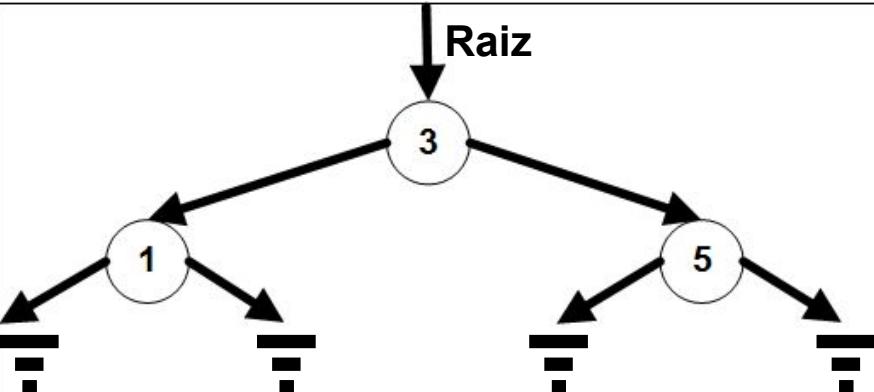
Inserção em Java com Retorno de Referência

```
//Inserir 3, 5, 1, 8, 2, 4, 7 e 6
```

```
void inserir(int x) throws Exception {  
    raiz = inserir(x, raiz);  
}
```

```
No inserir(int x, No i) throws Exception {  
    if (i == null) {  
        i = new No(x);  
    } else if (x < i.elemento) {  
        i.esq = inserir(x, i.esq);  
    } else if (x > i.elemento) {  
        i.dir = inserir(x, i.dir);  
    } else {  
        throw new Exception("Erro!");  
    }  
    return i;  
}
```

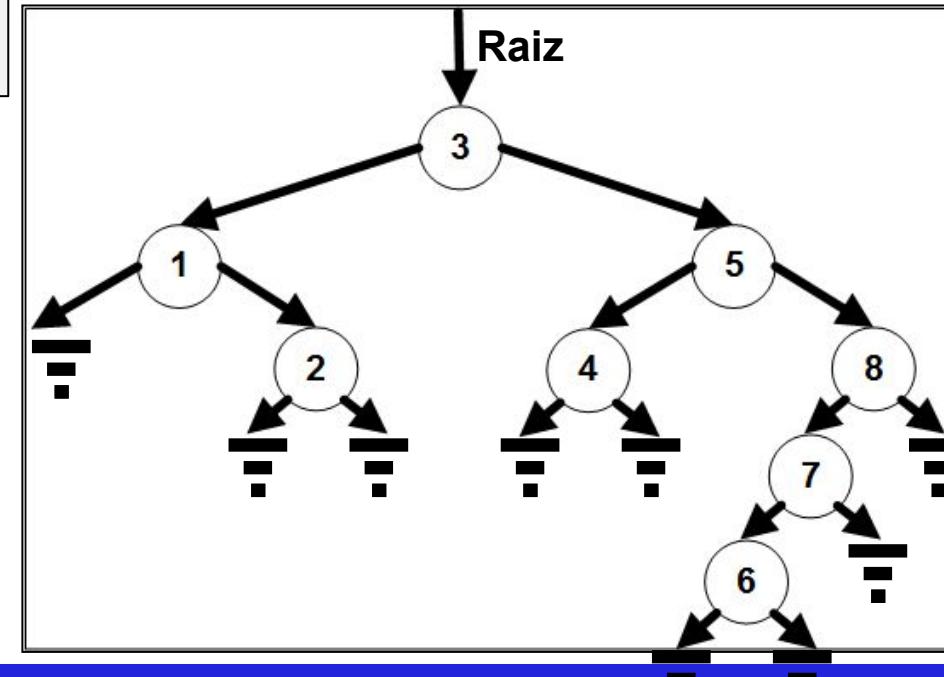
raiz
n(3)



Inserção em Java com Retorno de Referência

```
class ArvoreBinaria {  
    No raiz;  
    ArvoreBinaria() { raiz = null; }  
    void inserir(int x) { }  
    void inserirPai(int x) { }  
    boolean pesquisar(int x) { }  
    void caminharCentral() { }  
    void caminharPre() { }  
    void caminharPos() { }  
    void remover(int x) { }  
}
```

Após a inserção do 8, 2, 4, 7 e 6, temos:



Agenda

- Funcionamento básico
- Exemplo
- Inserção em Java com retorno de referência
- **Inserção em Java com passagem de pai** ←
- Análise de complexidade

Inserção em Java com Passagem de Pai

```
class ArvoreBinaria {  
    No raiz;  
    ArvoreBinaria() { raiz = null; }  
    void inserir(int x) { }  
    void inserirPai(int x) { }  
    boolean pesquisar(int x) { }  
    void caminharCentral() { }  
    void caminharPre() { }  
    void caminharPos() { }  
    void remover(int x) { }  
}
```



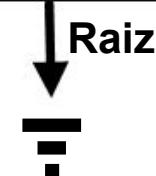
Vamos inserir os elementos
3, 5, 1 e 8
(várias chamadas do inserir)

Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

```
void inserirPai(int x) throws Exception {  
    if (raiz == null) {  
        raiz = new No(x);  
    } else if (x < raiz.elemento) {  
        inserirPai(x, raiz.esq, raiz);  
    } else if (x > raiz.elemento) {  
        inserirPai(x, raiz.dir, raiz);  
    } else {  
        throw new Exception("Erro!");  
    }  
}
```

```
void inserirPai(int x, No i, No pai) throws Exception {  
    ■ ■ ■  
}
```



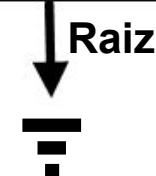
Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

```
void inserirPai(int x) throws Exception {
```

```
    if (raiz == null) {  
        raiz = new No(x);  
    } else if (x < raiz.elemento) {  
        inserirPai(x, raiz.esq, raiz);  
    } else if (x > raiz.elemento) {  
        inserirPai(x, raiz.dir, raiz);  
    } else {  
        throw new Exception("Erro!");  
    }  
}
```

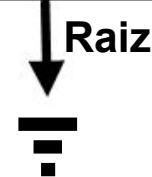
```
void inserirPai(int x, No i, No pai) throws Exception {  
    ■ ■ ■  
}
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

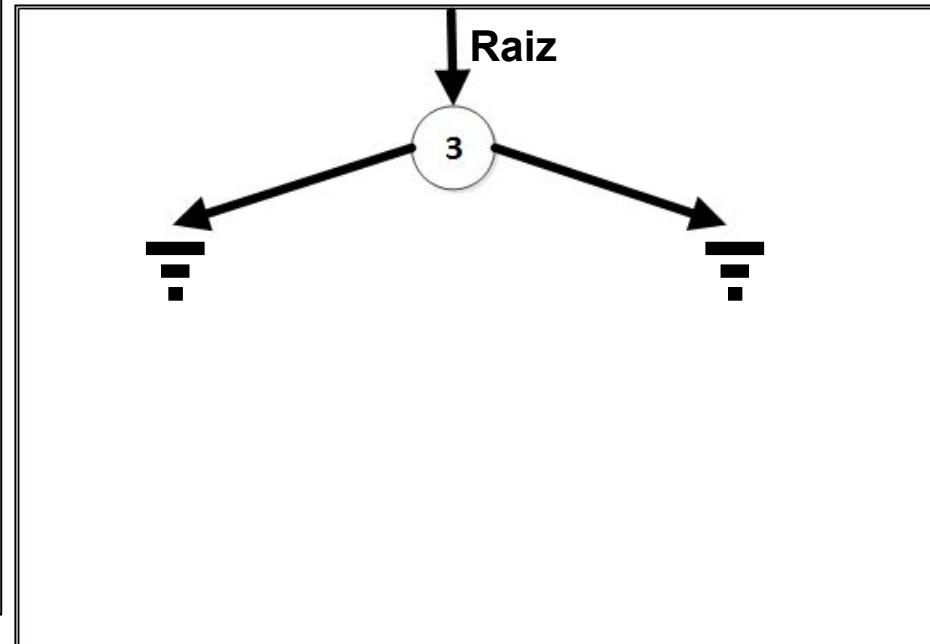
```
void inserirPai(int x) throws Exception {  
    if (raiz == null) {  
        raiz = new No(x);  
    } else if (x < raiz.elemento) {  
        inserirPai(x, raiz.esq, raiz);  
    } else if (x > raiz.elemento) {  
        inserirPai(x, raiz.dir, raiz);  
    } else {  
        throw new Exception("Erro!");  
    }  
}  
true: null == null  
void inserirPai(int x, No i, No pai) throws Exception {  
    ■ ■ ■  
}
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

```
void inserirPai(int x) throws Exception {  
    if (raiz == null) {  
        raiz = new No(x);  
    } else if (x < raiz.elemento) {  
        inserirPai(x, raiz.esq, raiz);  
    } else if (x > raiz.elemento) {  
        inserirPai(x, raiz.dir, raiz);  
    } else {  
        throw new Exception("Erro!");  
    }  
}  
  
void inserirPai(int x, No i, No pai) throws Exception {  
    ...  
}
```

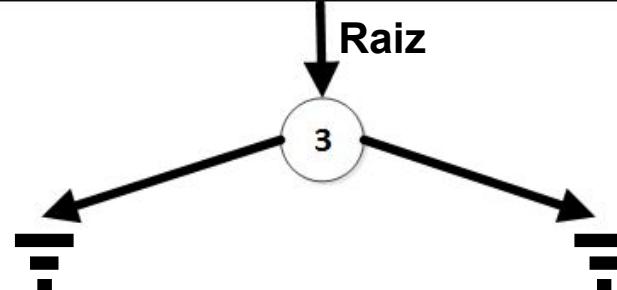


Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

```
void inserirPai(int x) throws Exception {  
    if (raiz == null) {  
        raiz = new No(x);  
    } else if (x < raiz.elemento) {  
        inserirPai(x, raiz.esq, raiz);  
    } else if (x > raiz.elemento) {  
        inserirPai(x, raiz.dir, raiz);  
    } else {  
        throw new Exception("Erro!");  
    }  
}
```

```
void inserirPai(int x, No i, No pai) throws Exception {  
    ■ ■ ■  
}
```

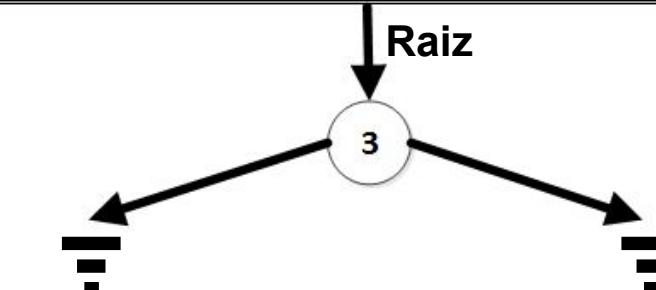


Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

```
void inserirPai(int x) throws Exception {  
    if (raiz == null) {  
        raiz = new No(x);  
    } else if (x < raiz.elemento) {  
        inserirPai(x, raiz.esq, raiz);  
    } else if (x > raiz.elemento) {  
        inserirPai(x, raiz.dir, raiz);  
    } else {  
        throw new Exception("Erro!");  
    }  
}
```

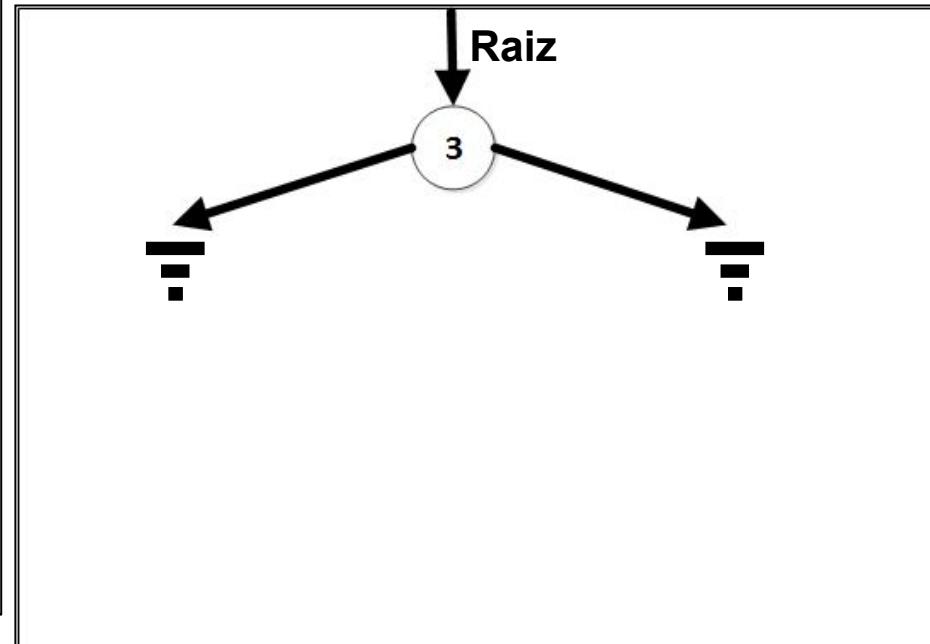
```
void inserirPai(int x, No i, No pai) throws Exception {  
    ...  
}
```



Inserção em Java com Passagem de Pai

//Inserir 3, 5, 1, 8

```
void inserirPai(int x) throws Exception {  
    if (raiz == null) {  
        raiz = new No(x);  
    } else if (x < raiz.elemento) {  
        inserirPai(x, raiz.esq, raiz);  
    } else if (x > raiz.elemento) {  
        inserirPai(x, raiz.dir, raiz);  
    } else {  
        throw new Exception("Erro!");  
    }  
}  
  
false: n(3) ≠ null  
  
void inserirPai(int x, No i, No pai) throws Exception {  
  
    ■ ■ ■  
}
```



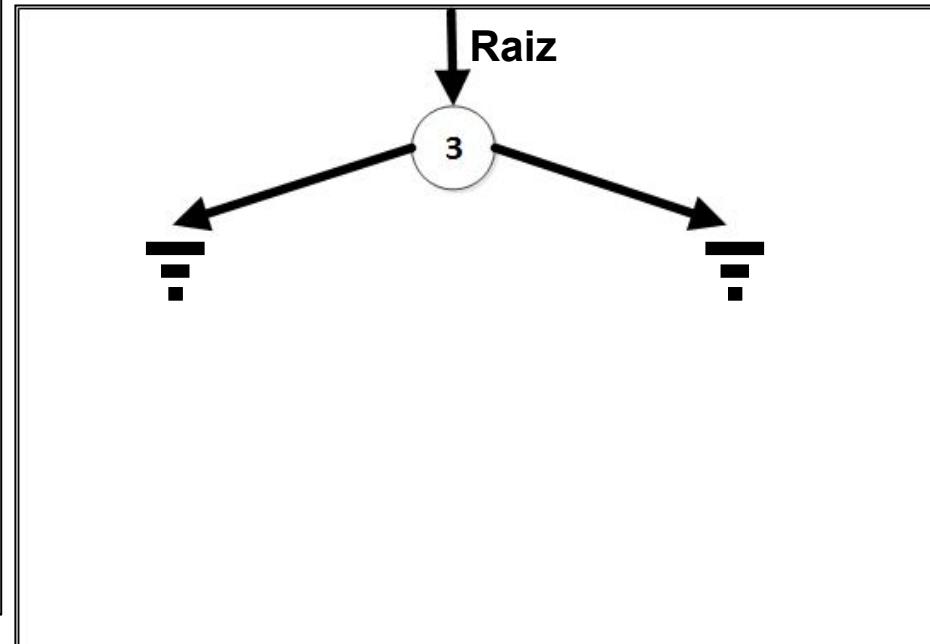
Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

```
void inserirPai(int x) throws Exception {  
    if (raiz == null) {  
        raiz = new No(x);  
    } else if (x < raiz.elemento) {  
        inserirPai(x, raiz.esq, raiz);  
    } else if (x > raiz.elemento) {  
        inserirPai(x, raiz.dir, raiz);  
    } else {  
        throw new Exception("Erro!");  
    }  
}
```

false: 5 < 3

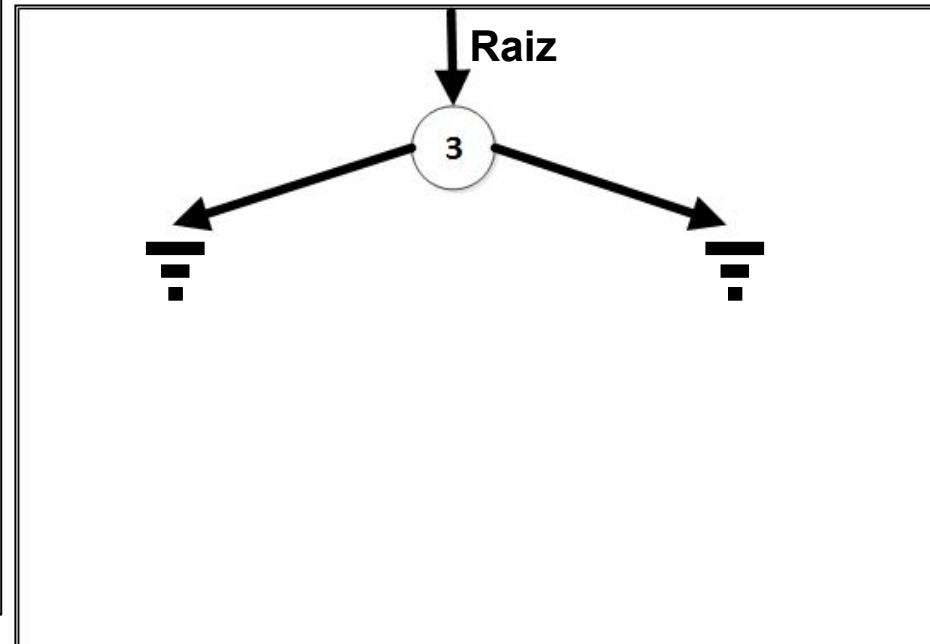
```
void inserirPai(int x, No i, No pai) throws Exception {  
    ...  
}
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

```
void inserirPai(int x) throws Exception {  
    if (raiz == null) {  
        raiz = new No(x);  
    } else if (x < raiz.elemento) {  
        inserirPai(x, raiz.esq, raiz);  
    } else if (x > raiz.elemento) {  
        inserirPai(x, raiz.dir, raiz);  
    } else {  
        throw new Exception("Erro!");  
    }  
}  
  
true: 5 > 3  
  
void inserirPai(int x, No i, No pai) throws Exception {  
    ...  
}
```

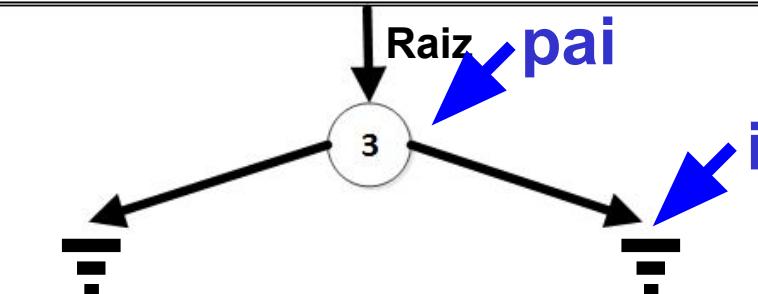


Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

```
void inserirPai(int x) throws Exception {  
    if (raiz == null) {  
        raiz = new No(x);  
    } else if (x < raiz.elemento) {  
        inserirPai(x, raiz.esq, raiz);  
    } else if (x > raiz.elemento) {  
        inserirPai(x, raiz.dir, raiz);  
    } else {  
        throw new Exception("Erro!");  
    }  
}
```

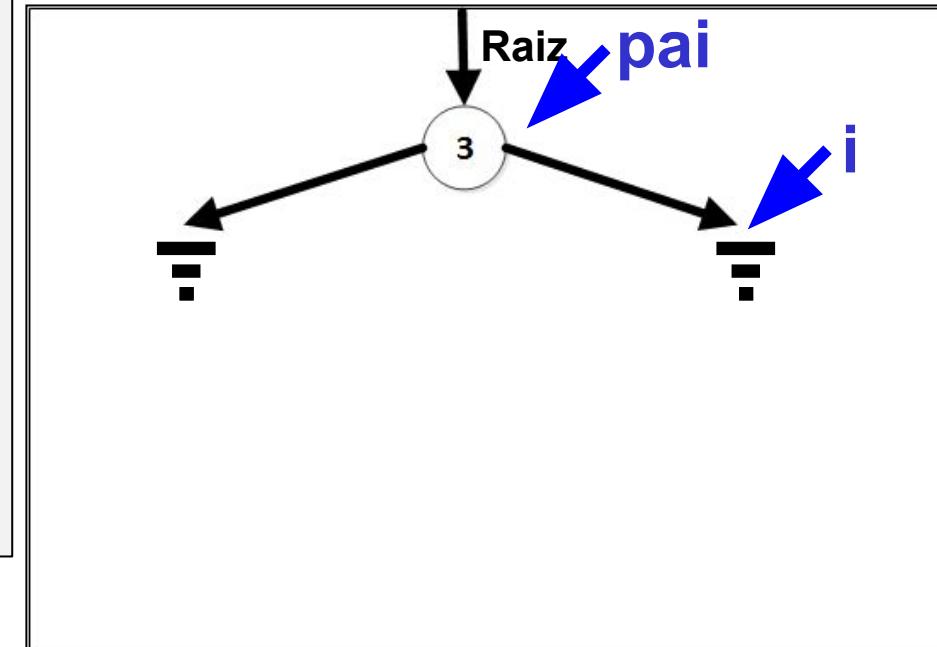
```
void inserirPai(int x, No i, No pai) throws Exception {  
    ...  
}
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

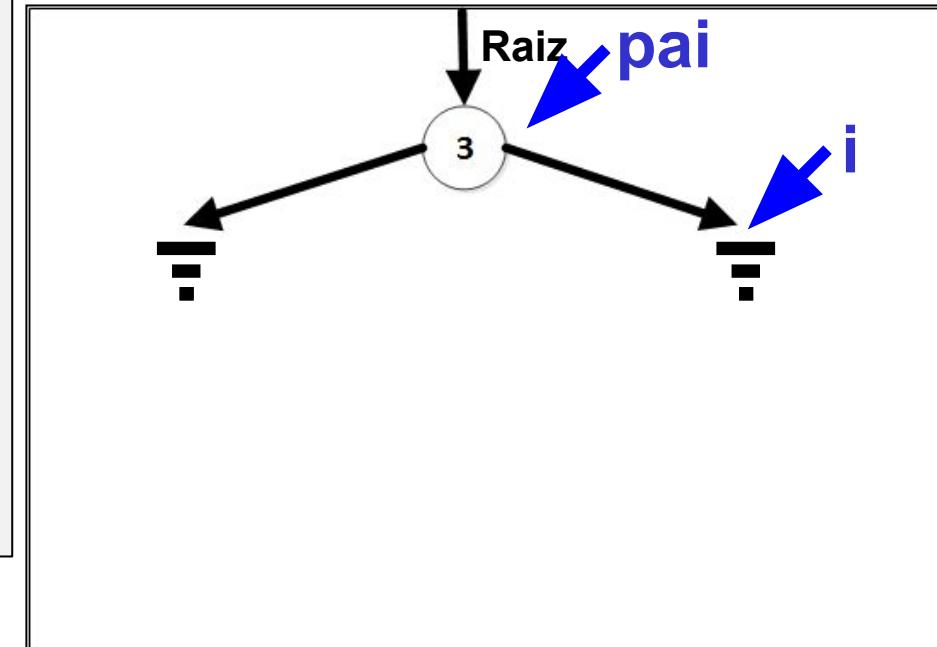
```
void inserirPai(int x, No i, No pai) throws Exception {  
    if (i == null) {  
        If (x < pai.elemento){  
            pai.esq = new No(x);  
        } else {  
            pai.dir = new No(x);  
        }  
    } else if (x < i.elemento) {  
        inserirPai(x, i.esq, i);  
    } else if (x > i.elemento) {  
        inserirPai(x, i.dir, i);  
    } else {  
        throw new Exception("Erro!");  
    }  
}
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

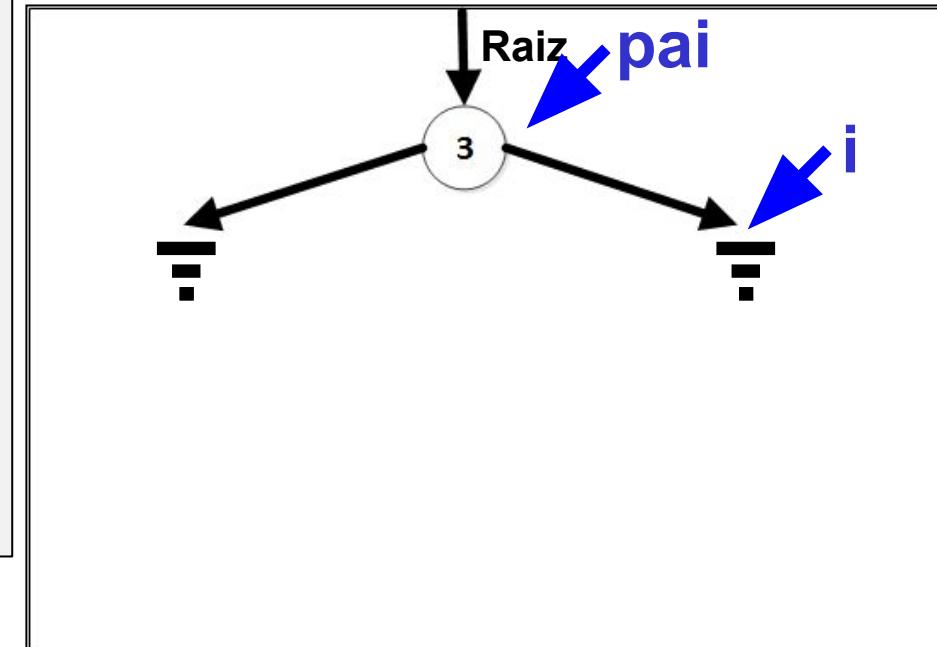
```
void inserirPai(int x, No i, No pai) throws Exception {  
    if (i == null) {  
        If (x < pai.elemento){  
            pai.esq = new No(x);  
        } else {  
            pai.dir = new No(x);  
        }  
    } else if (x < i.elemento) {  
        inserirPai(x, i.esq, i);  
    } else if (x > i.elemento) {  
        inserirPai(x, i.dir, i);  
    } else {  
        throw new Exception("Erro!");  
    }  
}  
true: null == null
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

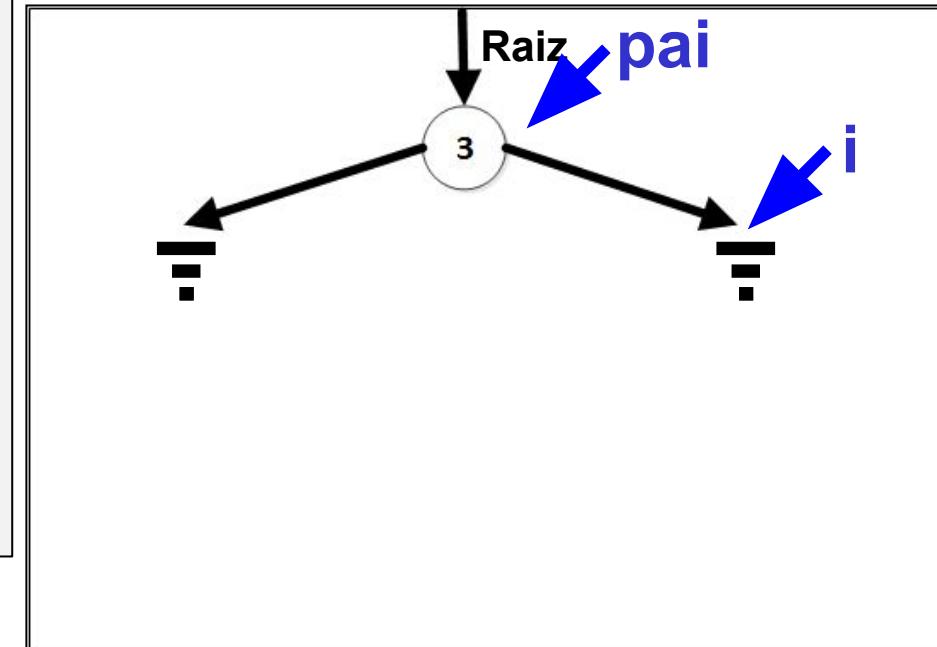
```
void inserirPai(int x, No i, No pai) throws Exception {  
    if (i == null) {  
        If (x < pai.elemento){  
            pai.esq = new No(x);  
        } else {  
            pai.dir = new No(x);  
        }  
    } else if (x < i.elemento) {  
        inserirPai(x, i.esq, i);  
    } else if (x > i.elemento) {  
        inserirPai(x, i.dir, i);  
    } else {  
        throw new Exception("Erro!");  
    }  
}  
  
false: 5 < 3
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

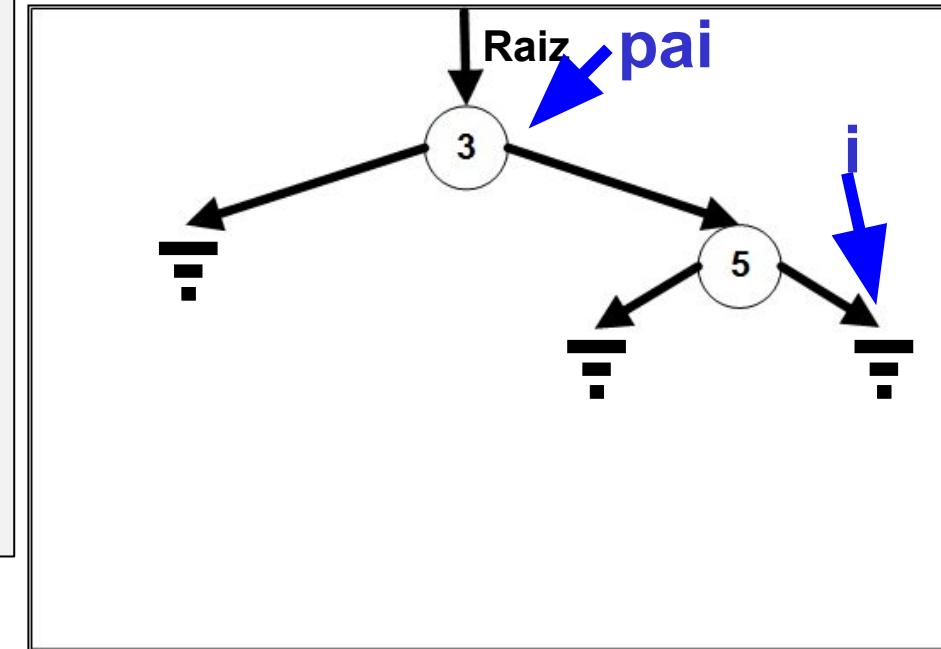
```
void inserirPai(int x, No i, No pai) throws Exception {  
    if (i == null) {  
        If (x < pai.elemento){  
            pai.esq = new No(x);  
        } else {  
            pai.dir = new No(x);  
        }  
    } else if (x < i.elemento) {  
        inserirPai(x, i.esq, i);  
    } else if (x > i.elemento) {  
        inserirPai(x, i.dir, i);  
    } else {  
        throw new Exception("Erro!");  
    }  
}  
  
true: 5 > 3
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

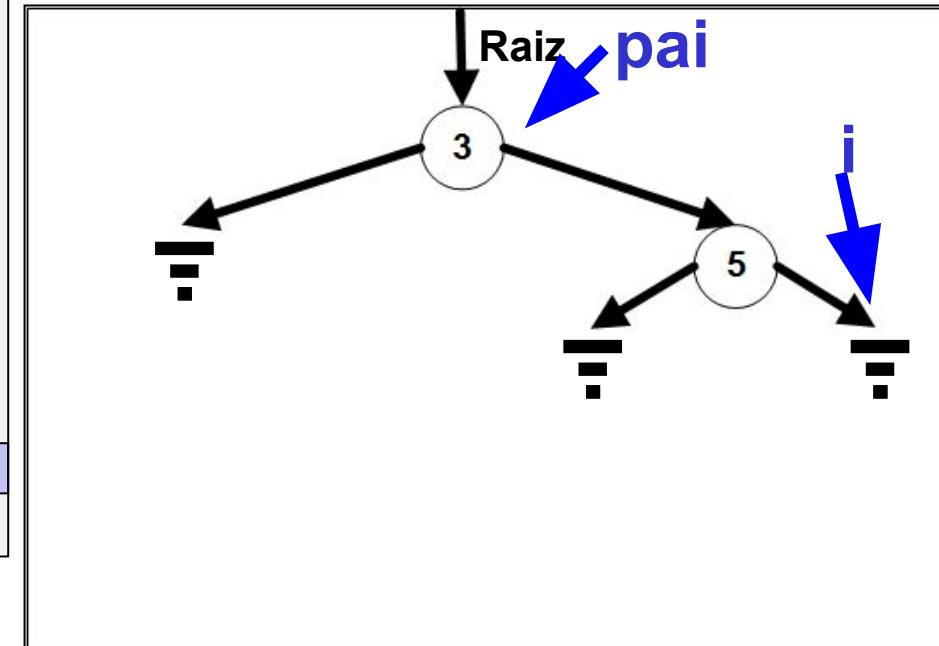
```
void inserirPai(int x, No i, No pai) throws Exception {  
    if (i == null) {  
        If (x < pai.elemento){  
            pai.esq = new No(x);  
        } else {  
            pai.dir = new No(x);  
        }  
    } else if (x < i.elemento) {  
        inserirPai(x, i.esq, i);  
    } else if (x > i.elemento) {  
        inserirPai(x, i.dir, i);  
    } else {  
        throw new Exception("Erro!");  
    }  
}
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

```
void inserirPai(int x, No i, No pai) throws Exception {  
    if (i == null) {  
        If (x < pai.elemento){  
            pai.esq = new No(x);  
        } else {  
            pai.dir = new No(x);  
        }  
    } else if (x < i.elemento) {  
        inserirPai(x, i.esq, i);  
    } else if (x > i.elemento) {  
        inserirPai(x, i.dir, i);  
    } else {  
        throw new Exception("Erro!");  
    }  
}
```



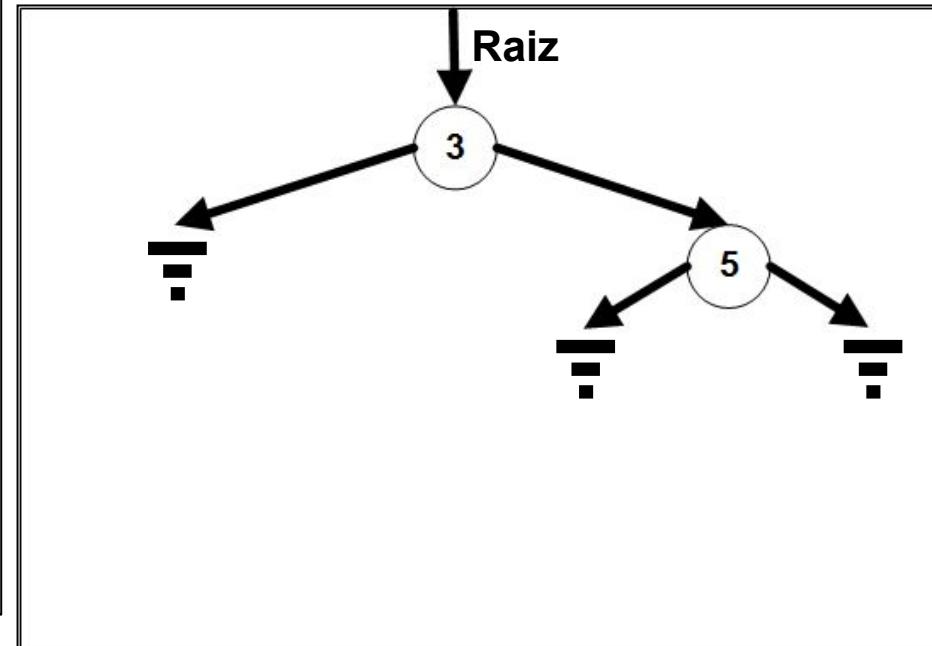
Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

```
void inserirPai(int x) throws Exception {
```

```
    if (raiz == null) {  
        raiz = new No(x);  
    } else if (x < raiz.elemento) {  
        inserirPai(x, raiz.esq, raiz);  
    } else if (x > raiz.elemento) {  
        inserirPai(x, raiz.dir, raiz);  
    } else {  
        throw new Exception("Erro!");  
    }  
}
```

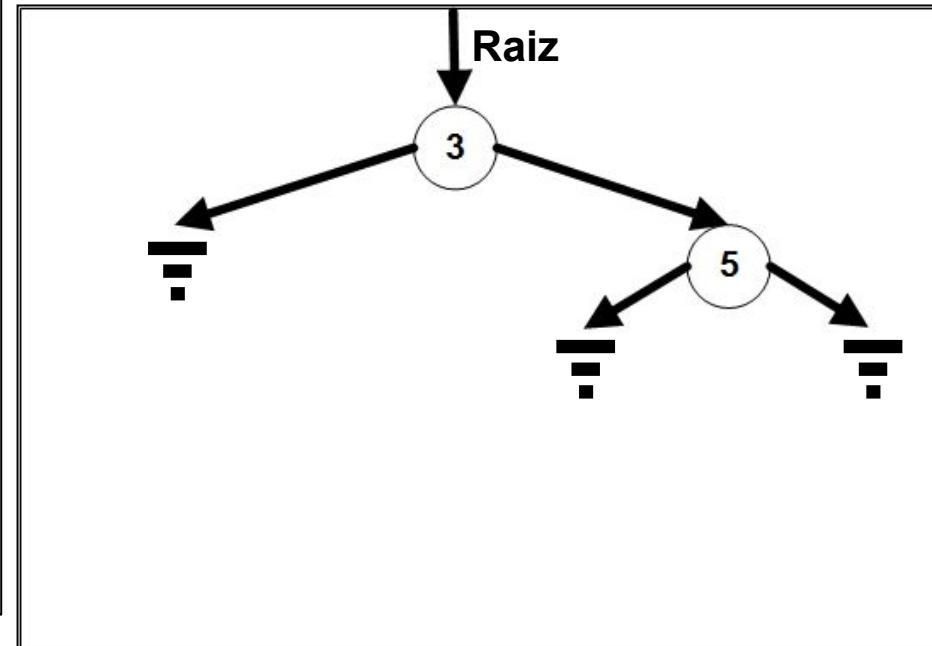
```
void inserirPai(int x, No i, No pai) throws Exception {  
    ■ ■ ■  
}
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

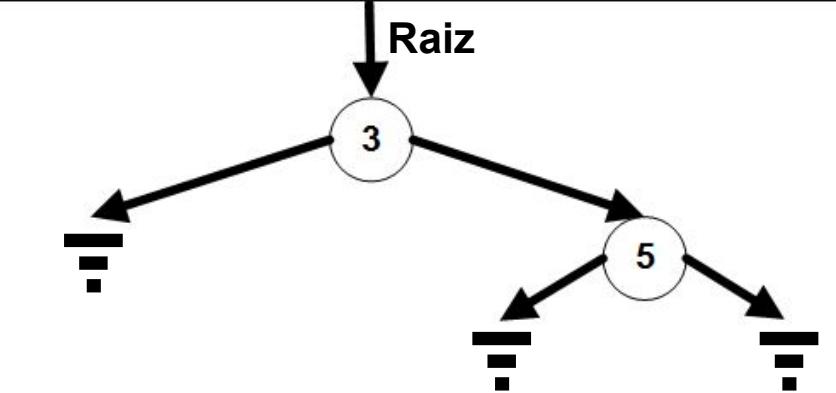
```
void inserirPai(int x) throws Exception {  
    if (raiz == null) {  
        raiz = new No(x);  
    } else if (x < raiz.elemento) {  
        inserirPai(x, raiz.esq, raiz);  
    } else if (x > raiz.elemento) {  
        inserirPai(x, raiz.dir, raiz);  
    } else {  
        throw new Exception("Erro!");  
    }  
}  
  
false: n(3) ≠ null  
  
void inserirPai(int x, No i, No pai) throws Exception {  
    ...  
}
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

```
void inserirPai(int x) throws Exception {  
    if (raiz == null) {  
        raiz = new No(x);  
    } else if (x < raiz.elemento) {  
        inserirPai(x, raiz.esq, raiz);  
    } else if (x > raiz.elemento) {  
        inserirPai(x, raiz.dir, raiz);  
    } else {  
        throw new Exception("Erro!");  
    }  
}  
  
true: 1 < 3  
  
void inserirPai(int x, No i, No pai) throws Exception {  
    ...  
}
```

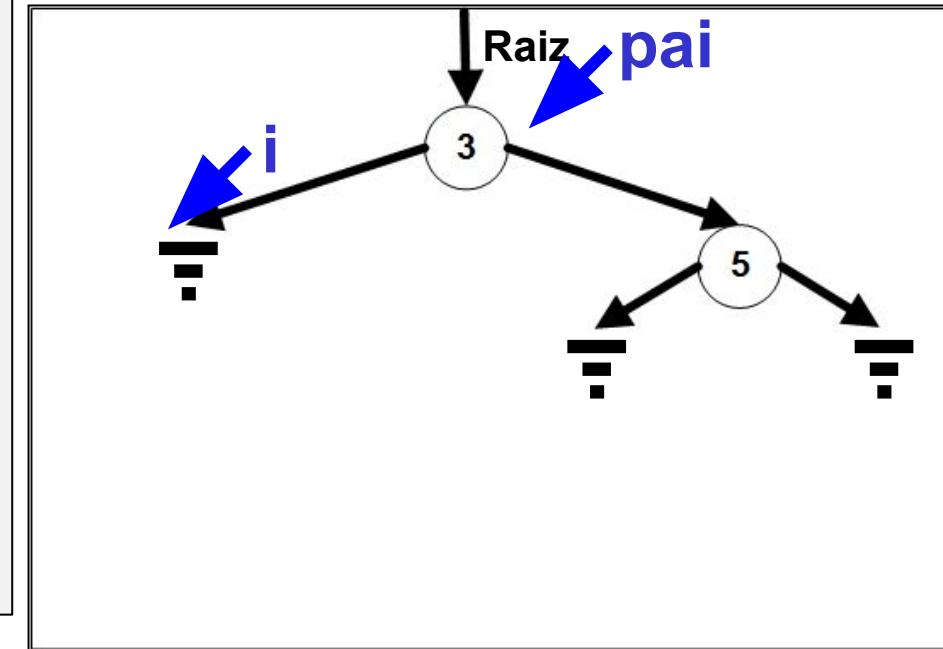


Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

```
void inserirPai(int x) throws Exception {
    if (raiz == null) {
        raiz = new No(x);
    } else if (x < raiz.elemento) {
        inserirPai(x, raiz.esq, raiz);
    } else if (x > raiz.elemento) {
        inserirPai(x, raiz.dir, raiz);
    } else {
        throw new Exception("Erro!");
    }
}

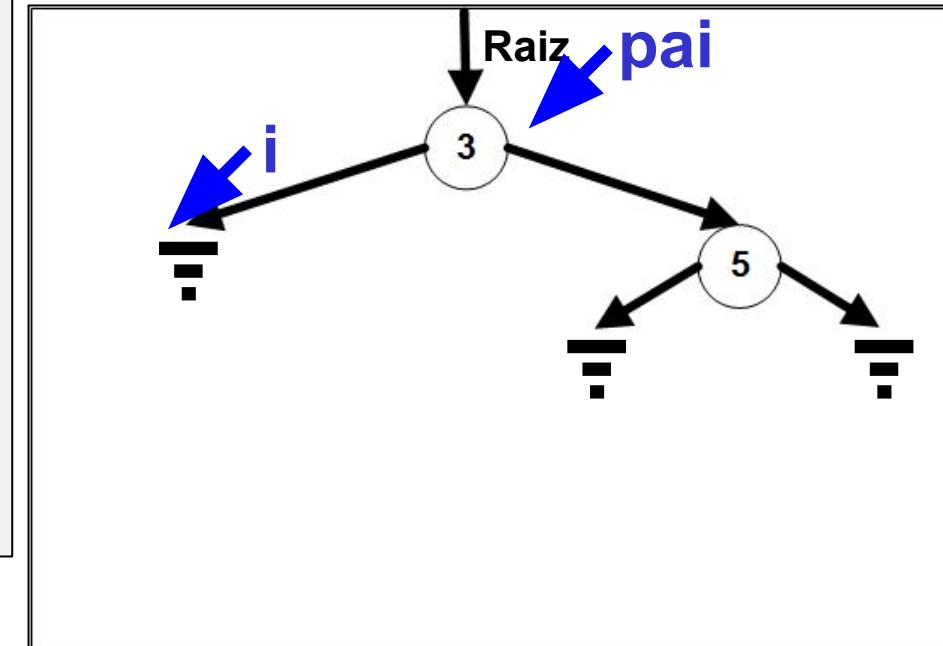
void inserirPai(int x, No i, No pai) throws Exception {
    ...
}
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

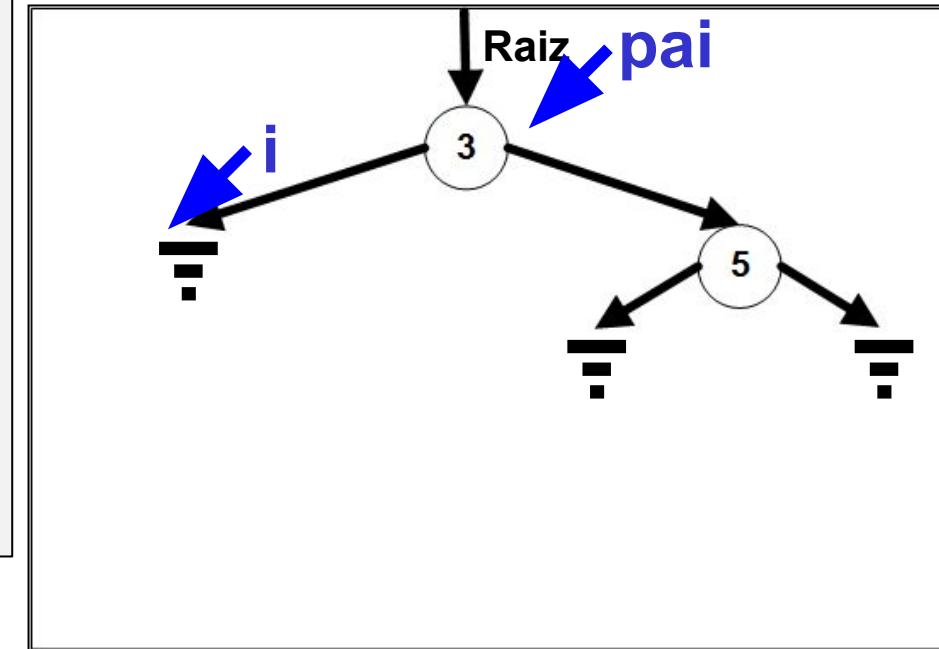
```
void inserirPai(int x, No i, No pai) throws Exception {  
    if (i == null) {  
        If (x < pai.elemento){  
            pai.esq = new No(x);  
        } else {  
            pai.dir = new No(x);  
        }  
    } else if (x < i.elemento) {  
        inserirPai(x, i.esq, i);  
    } else if (x > i.elemento) {  
        inserirPai(x, i.dir, i);  
    } else {  
        throw new Exception("Erro!");  
    }  
}
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

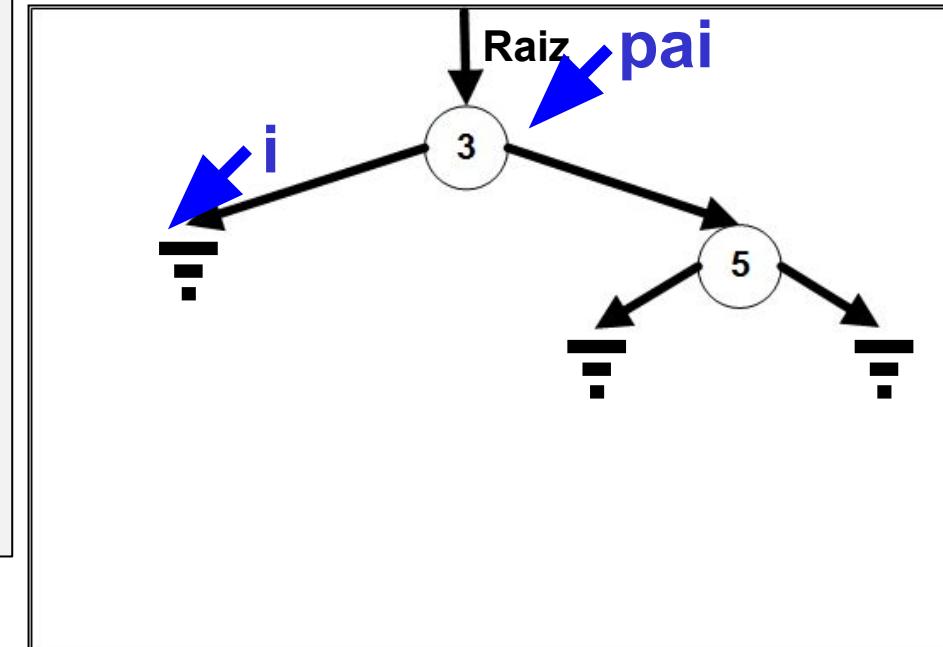
```
void inserirPai(int x, No i, No pai) throws Exception {
    if (i == null) {
        If (x < pai.elemento){
            pai.esq = new No(x);
        } else {
            pai.dir = new No(x);
        }
    } else if (x < i.elemento) {
        inserirPai(x, i.esq, i);
    } else if (x > i.elemento) {
        inserirPai(x, i.dir, i);
    } else {
        throw new Exception("Erro!");
    }
}
true: null == null
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

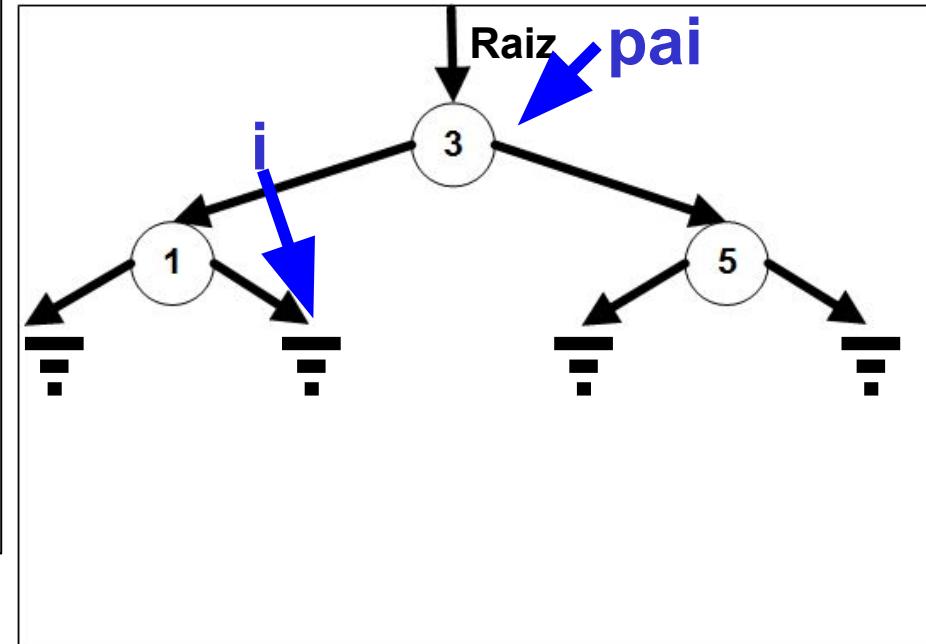
```
void inserirPai(int x, No i, No pai) throws Exception {  
    if (i == null) {  
        If (x < pai.elemento){  
            pai.esq = new No(x);  
        } else {  
            pai.dir = new No(x);  
        }  
    } else if (x < i.elemento) {  
        inserirPai(x, i.esq, i);  
    } else if (x > i.elemento) {  
        inserirPai(x, i.dir, i);  
    } else {  
        throw new Exception("Erro!");  
    }  
}  
  
true: 1 < 3
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

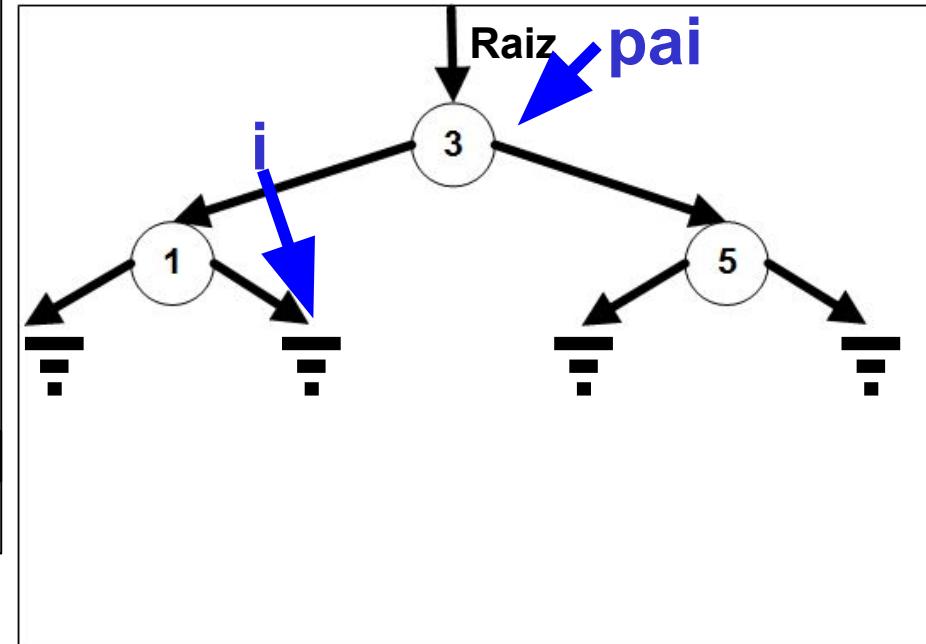
```
void inserirPai(int x, No i, No pai) throws Exception {
    if (i == null) {
        If (x < pai.elemento){
            pai.esq = new No(x);
        } else {
            pai.dir = new No(x);
        }
    } else if (x < i.elemento) {
        inserirPai(x, i.esq, i);
    } else if (x > i.elemento) {
        inserirPai(x, i.dir, i);
    } else {
        throw new Exception("Erro!");
    }
}
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

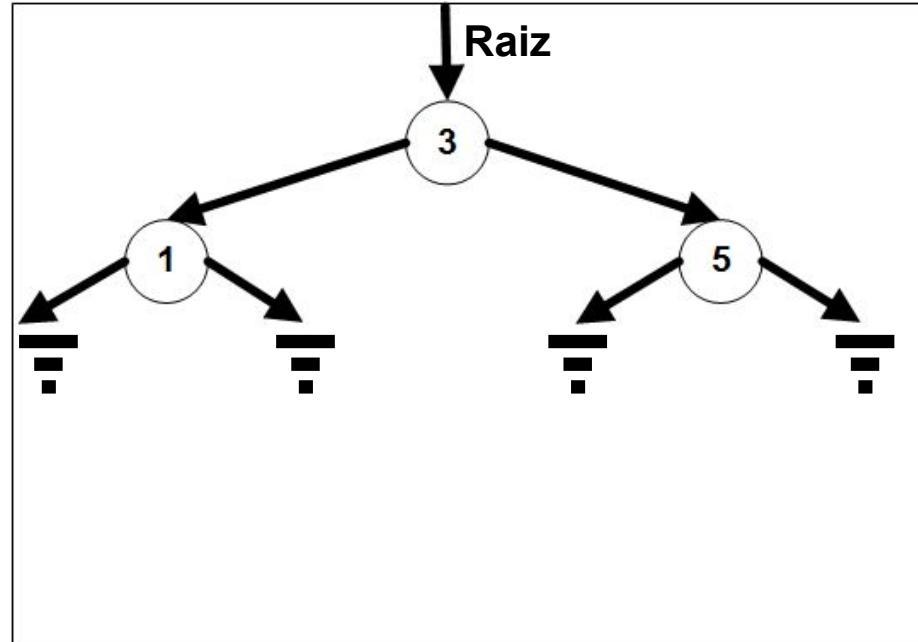
```
void inserirPai(int x, No i, No pai) throws Exception {  
    if (i == null) {  
        If (x < pai.elemento){  
            pai.esq = new No(x);  
        } else {  
            pai.dir = new No(x);  
        }  
    } else if (x < i.elemento) {  
        inserirPai(x, i.esq, i);  
    } else if (x > i.elemento) {  
        inserirPai(x, i.dir, i);  
    } else {  
        throw new Exception("Erro!");  
    }  
}
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

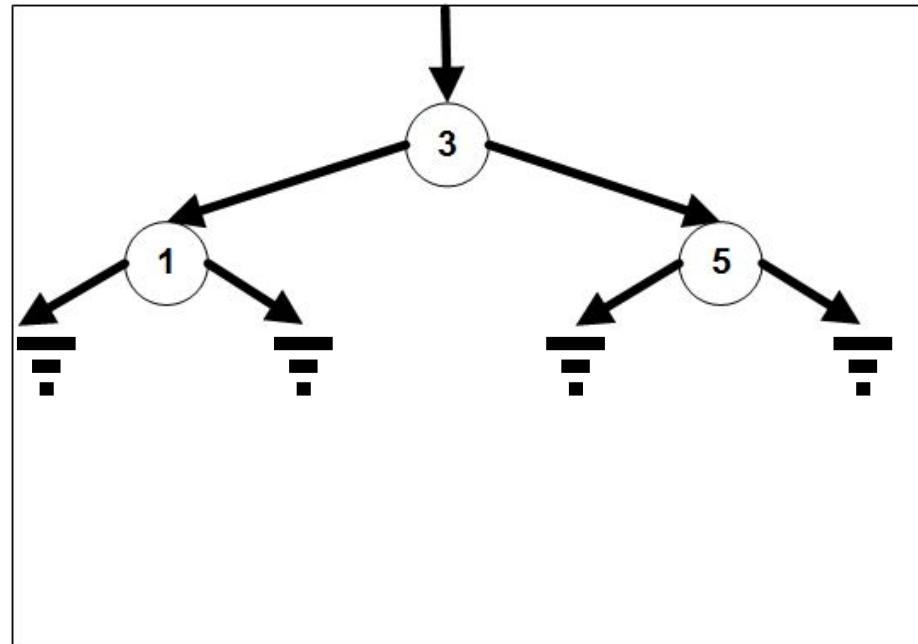
```
void inserirPai(int x) throws Exception {  
    if (raiz == null) {  
        raiz = new No(x);  
    } else if (x < raiz.elemento) {  
        inserirPai(x, raiz.esq, raiz);  
    } else if (x > raiz.elemento) {  
        inserirPai(x, raiz.dir, raiz);  
    } else {  
        throw new Exception("Erro!");  
    }  
}  
  
void inserirPai(int x, No i, No pai) throws Exception {  
    ...  
}
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

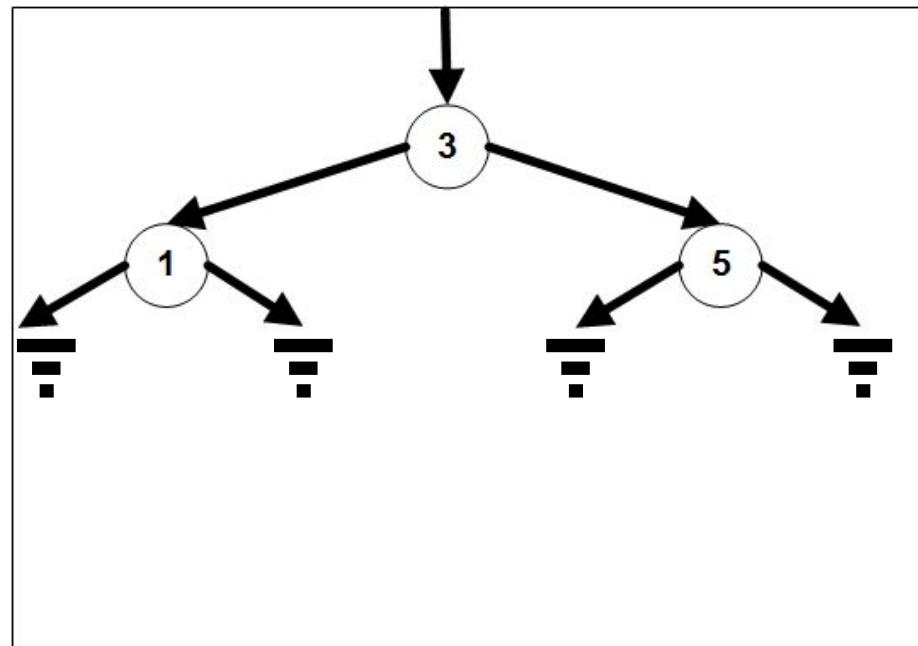
```
void inserirPai(int x) throws Exception {  
    if (raiz == null) {  
        raiz = new No(x);  
    } else if (x < raiz.elemento) {  
        inserirPai(x, raiz.esq, raiz);  
    } else if (x > raiz.elemento) {  
        inserirPai(x, raiz.dir, raiz);  
    } else {  
        throw new Exception("Erro!");  
    }  
}  
  
false: n(3) ≠ null  
  
void inserirPai(int x, No i, No pai) throws Exception {  
    ...  
}
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

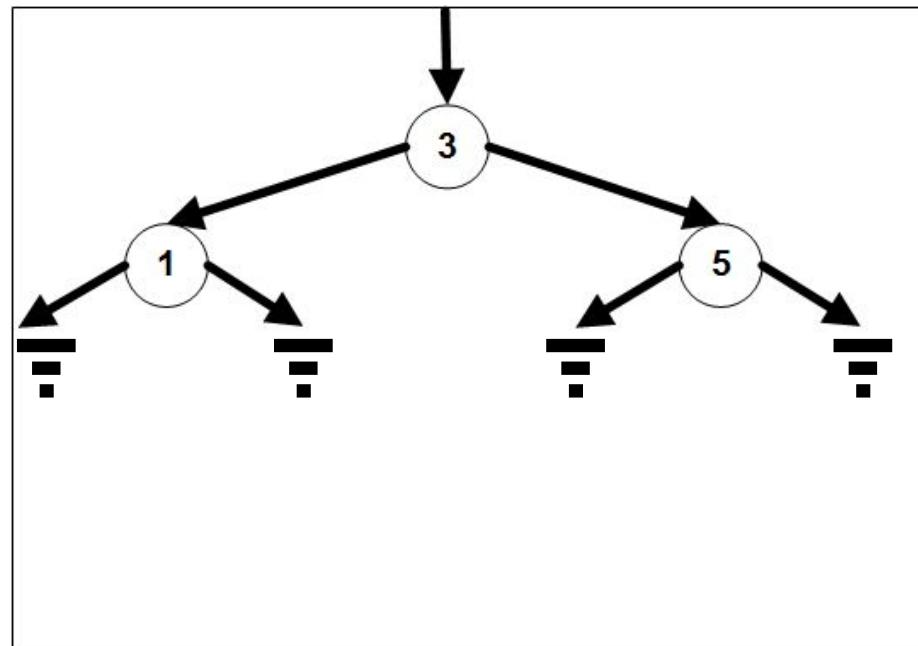
```
void inserirPai(int x) throws Exception {  
    if (raiz == null) {  
        raiz = new No(x);  
    } else if (x < raiz.elemento) {  
        inserirPai(x, raiz.esq, raiz);  
    } else if (x > raiz.elemento) {  
        inserirPai(x, raiz.dir, raiz);  
    } else {  
        throw new Exception("Erro!");  
    }  
}  
  
false: 8 < 3  
  
void inserirPai(int x, No i, No pai) throws Exception {  
    ...  
}
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

```
void inserirPai(int x) throws Exception {  
    if (raiz == null) {  
        raiz = new No(x);  
    } else if (x < raiz.elemento) {  
        inserirPai(x, raiz.esq, raiz);  
    } else if (x > raiz.elemento) {  
        inserirPai(x, raiz.dir, raiz);  
    } else {  
        throw new Exception("Erro!");  
    }  
}  
  
true: 8 > 3  
  
void inserirPai(int x, No i, No pai) throws Exception {  
    ...  
}
```

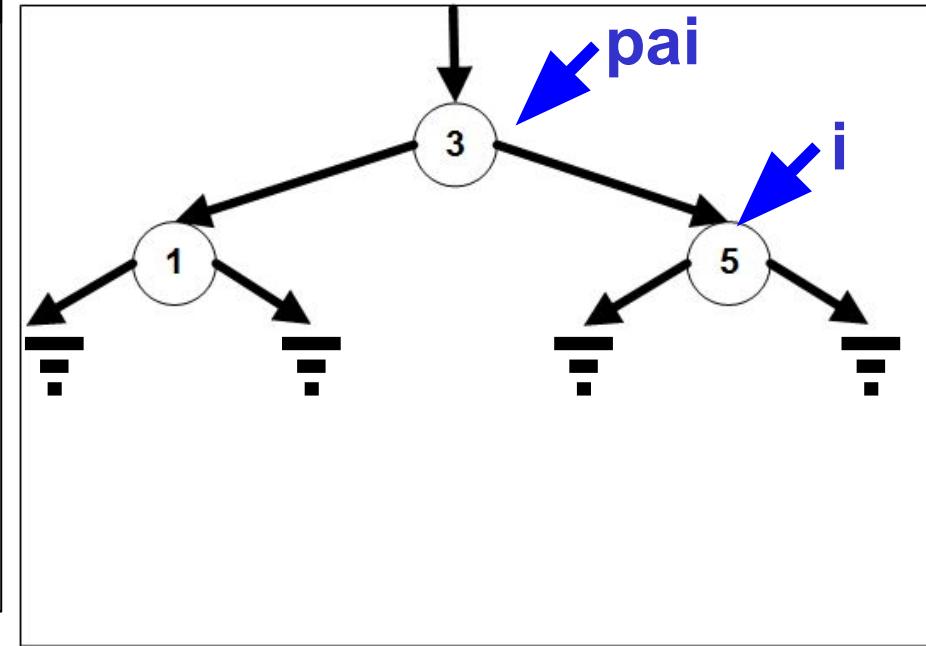


Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

```
void inserirPai(int x) throws Exception {
    if (raiz == null) {
        raiz = new No(x);
    } else if (x < raiz.elemento) {
        inserirPai(x, raiz.esq, raiz);
    } else if (x > raiz.elemento) {
        inserirPai(x, raiz.dir, raiz);
    } else {
        throw new Exception("Erro!");
    }
}

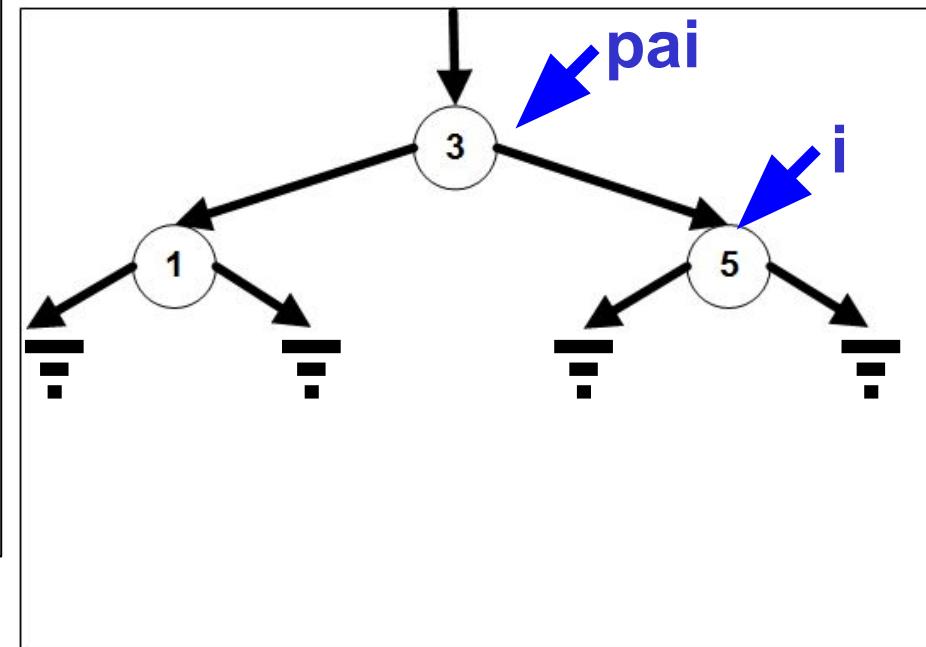
void inserirPai(int x, No i, No pai) throws Exception {
    ...
}
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

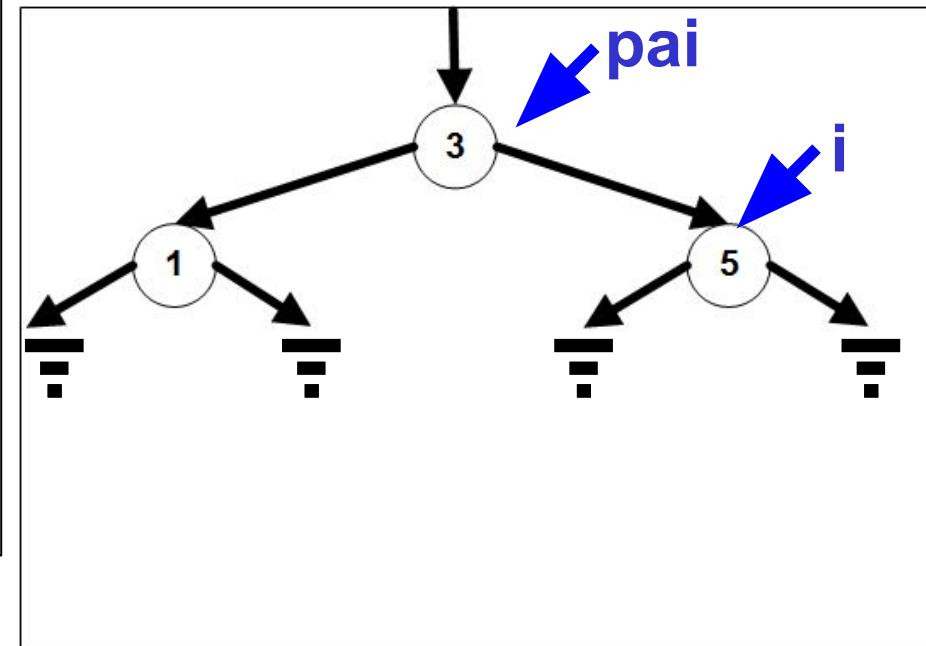
```
void inserirPai(int x, No i, No pai) throws Exception {  
    if (i == null) {  
        If (x < pai.elemento){  
            pai.esq = new No(x);  
        } else {  
            pai.dir = new No(x);  
        }  
    } else if (x < i.elemento) {  
        inserirPai(x, i.esq, i);  
    } else if (x > i.elemento) {  
        inserirPai(x, i.dir, i);  
    } else {  
        throw new Exception("Erro!");  
    }  
}
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

```
void inserirPai(int x, No i, No pai) throws Exception {  
    if (i == null) {  
        If (x < pai.elemento){  
            pai.esq = new No(x);  
        } else {  
            pai.dir = new No(x);  
        }  
    } else if (x < i.elemento) {  
        inserirPai(x, i.esq, i);  
    } else if (x > i.elemento) {  
        inserirPai(x, i.dir, i);  
    } else {  
        throw new Exception("Erro!");  
    }  
}  
  
false: n(5) ≠ null
```

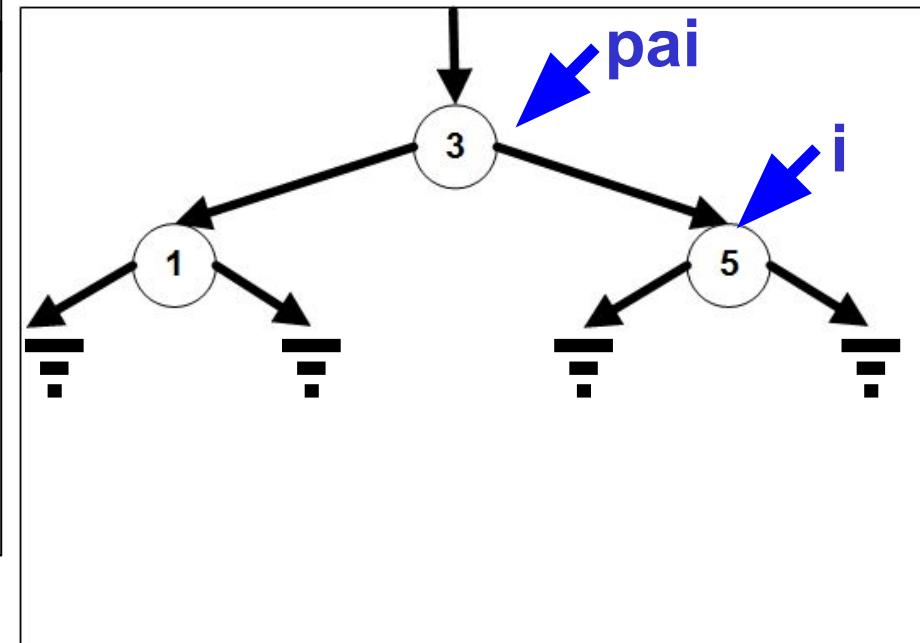


Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

```
void inserirPai(int x, No i, No pai) throws Exception {
    if (i == null) {
        If (x < pai.elemento){
            pai.esq = new No(x);
        } else {
            pai.dir = new No(x);
        }
    } else if (x < i.elemento) {
        inserirPai(x, i.esq, i);
    } else if (x > i.elemento) {
        inserirPai(x, i.dir, i);
    } else {
        throw new Exception("Erro!");
    }
}
```

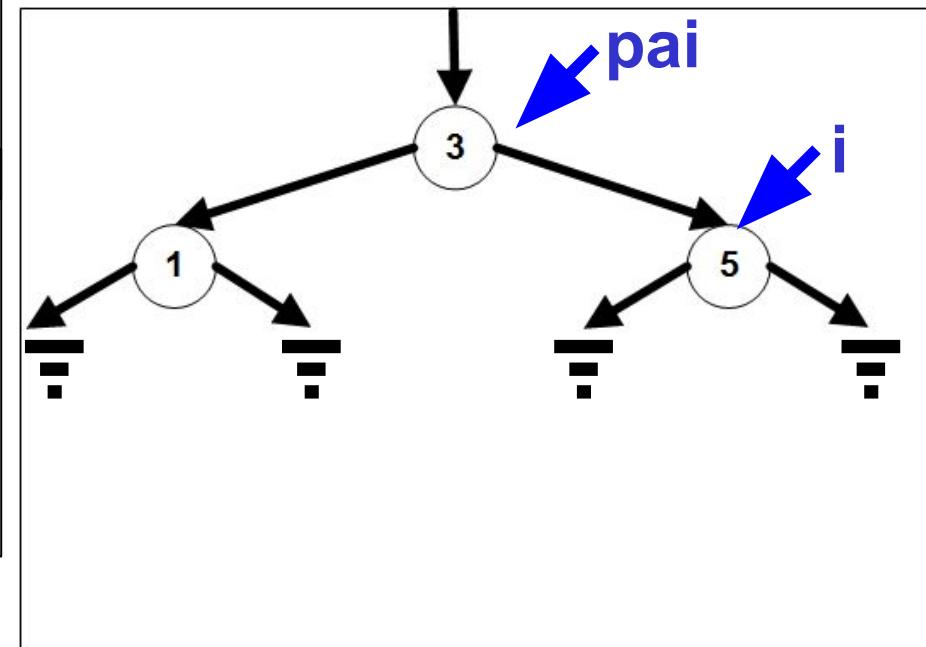
false: 8 < 5



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

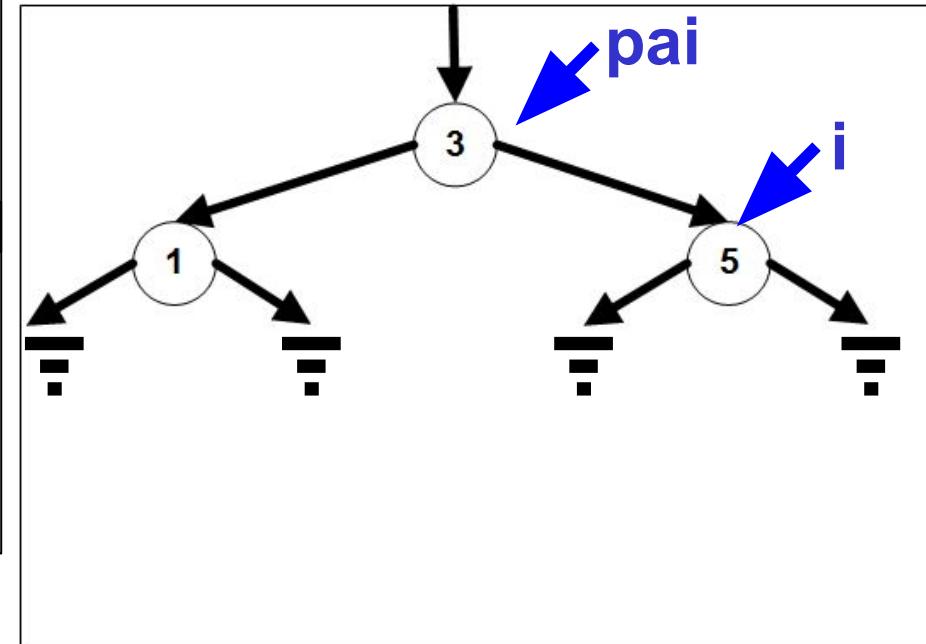
```
void inserirPai(int x, No i, No pai) throws Exception {  
    if (i == null) {  
        If (x < pai.elemento){  
            pai.esq = new No(x);  
        } else {  
            pai.dir = new No(x);  
        }  
    } else if (x < i.elemento) {  
        inserirPai(x, i.esq, i);  
    } else if (x > i.elemento) {  
        inserirPai(x, i.dir, i);  
    } else {  
        throw new Exception("Erro!");  
    }  
}  
  
true: 8 > 5
```



Inserção em Java com Passagem de Pai

//Inserir 3, 5, 1, **8**

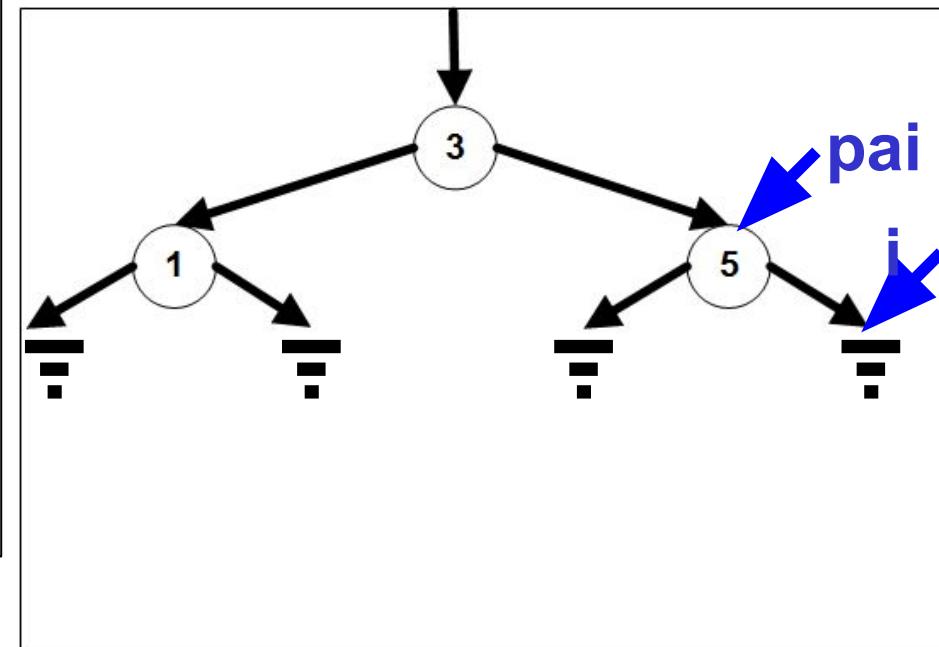
```
void inserirPai(int x, No i, No pai) throws Exception {  
    if (i == null) {  
        If (x < pai.elemento){  
            pai.esq = new No(x);  
        } else {  
            pai.dir = new No(x);  
        }  
    } else if (x < i.elemento) {  
        inserirPai(x, i.esq, i);  
    } else if (x > i.elemento) {  
        inserirPai(x, i.dir, i);  
    } else {  
        throw new Exception("Erro!");  
    }  
}
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

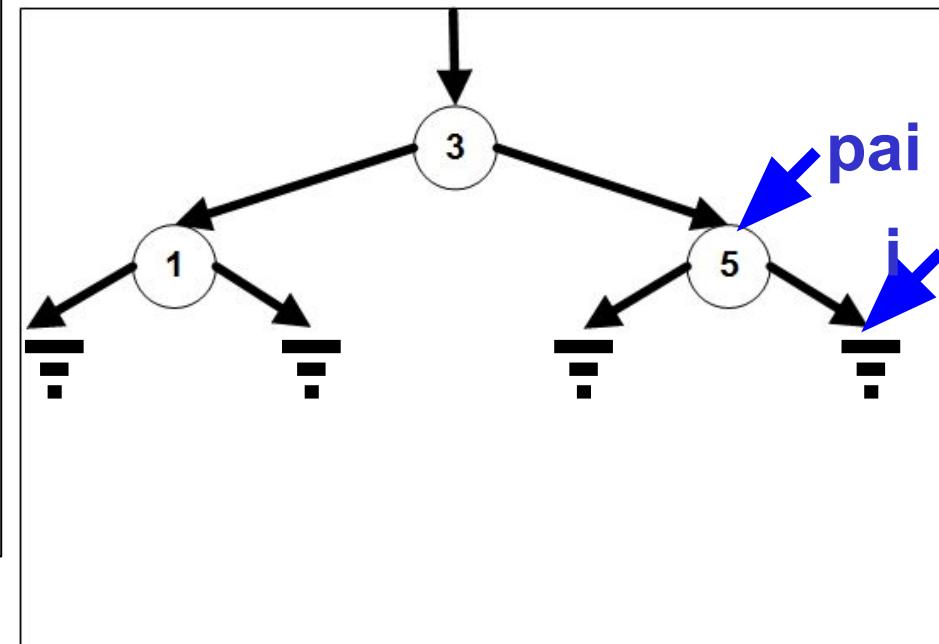
```
void inserirPai(int x, No i, No pai) throws Exception {  
    if (i == null) {  
        If (x < pai.elemento){  
            pai.esq = new No(x);  
        } else {  
            pai.dir = new No(x);  
        }  
    } else if (x < i.elemento) {  
        inserirPai(x, i.esq, i);  
    } else if (x > i.elemento) {  
        inserirPai(x, i.dir, i);  
    } else {  
        throw new Exception("Erro!");  
    }  
}
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

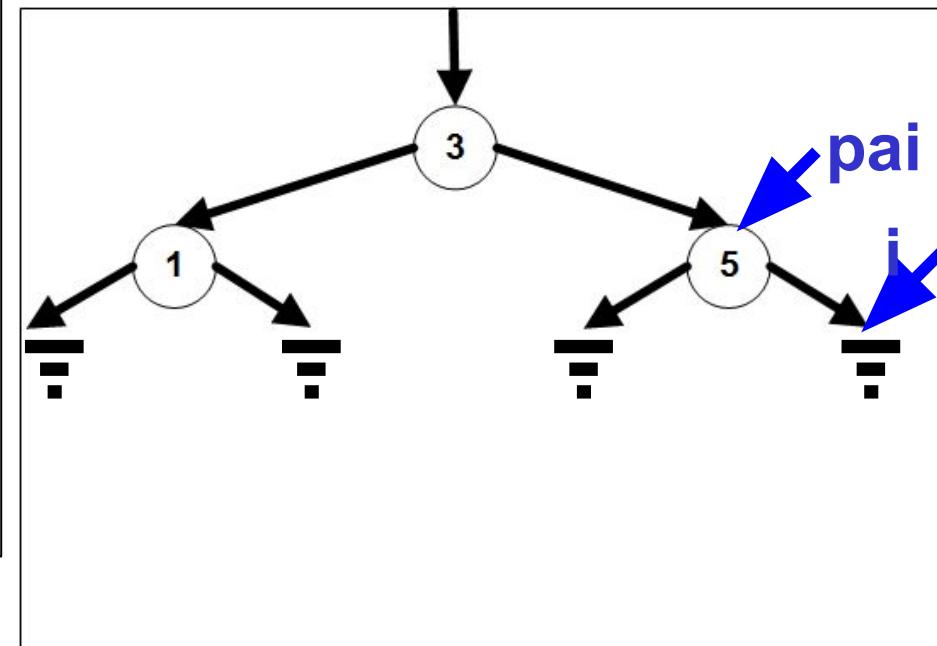
```
void inserirPai(int x, No i, No pai) throws Exception {  
    if (i == null) {  
        If (x < pai.elemento){  
            pai.esq = new No(x);  
        } else {  
            pai.dir = new No(x);  
        }  
    } else if (x < i.elemento) {  
        inserirPai(x, i.esq, i);  
    } else if (x > i.elemento) {  
        inserirPai(x, i.dir, i);  
    } else {  
        throw new Exception("Erro!");  
    }  
}  
  
true: null == null
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

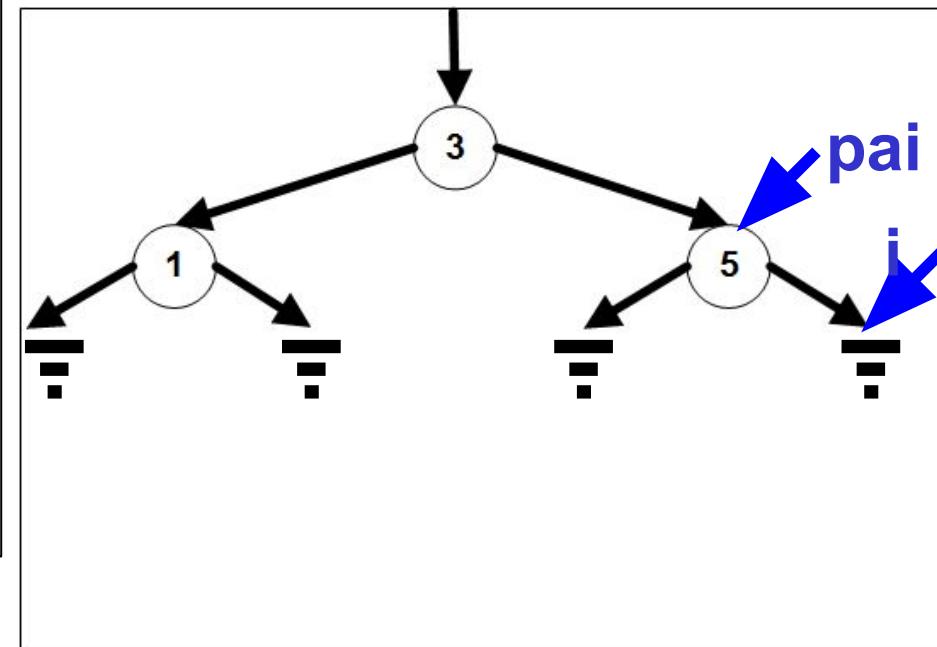
```
void inserirPai(int x, No i, No pai) throws Exception {  
    if (i == null) {  
        If (x < pai.elemento){  
            pai.esq = new No(x);  
        } else {  
            pai.dir = new No(x);  
        }  
    } else if (x < i.elemento) {  
        inserirPai(x, i.esq, i);  
    } else if (x > i.elemento) {  
        inserirPai(x, i.dir, i);  
    } else {  
        throw new Exception("Erro!");  
    }  
}  
  
false: 8 < 5
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

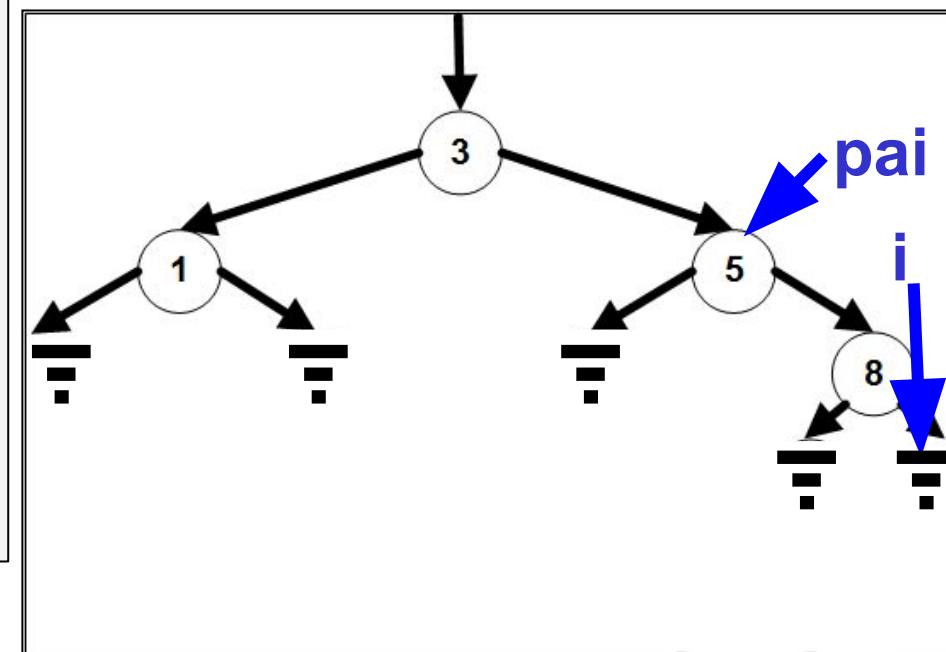
```
void inserirPai(int x, No i, No pai) throws Exception {  
    if (i == null) {  
        If (x < pai.elemento){  
            pai.esq = new No(x);  
    } else {  
        pai.dir = new No(x);  
    }  
    } else if (x < i.elemento) {  
        inserirPai(x, i.esq, i);  
    } else if (x > i.elemento) {  
        inserirPai(x, i.dir, i);  
    } else {  
        throw new Exception("Erro!");  
    }  
}
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

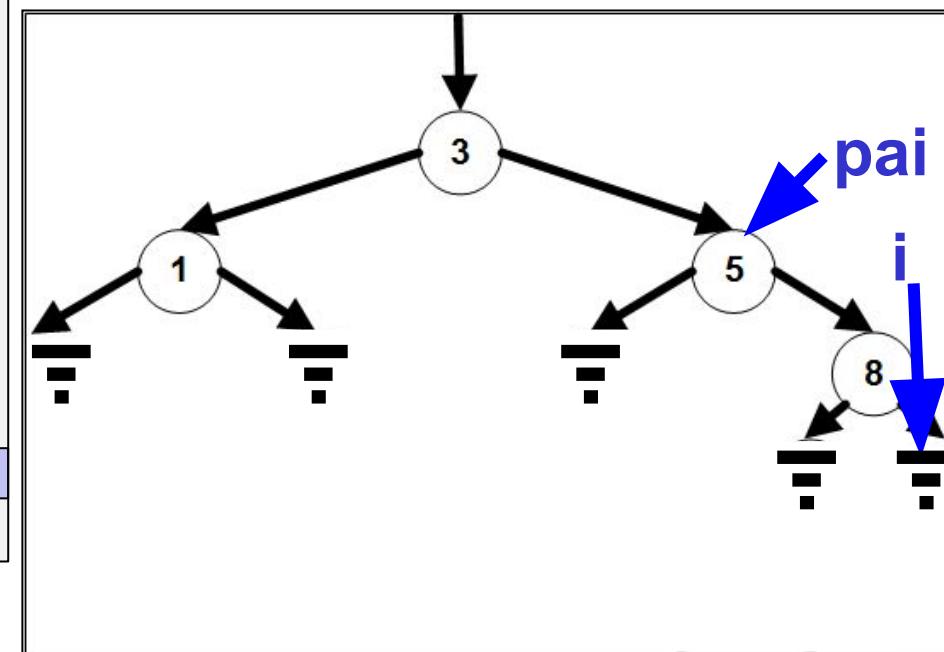
```
void inserirPai(int x, No i, No pai) throws Exception {  
    if (i == null) {  
        If (x < pai.elemento){  
            pai.esq = new No(x);  
        } else {  
            pai.dir = new No(x);  
        }  
    } else if (x < i.elemento) {  
        inserirPai(x, i.esq, i);  
    } else if (x > i.elemento) {  
        inserirPai(x, i.dir, i);  
    } else {  
        throw new Exception("Erro!");  
    }  
}
```



Inserção em Java com Passagem de Pai

```
//Inserir 3, 5, 1, 8
```

```
void inserirPai(int x, No i, No pai) throws Exception {  
    if (i == null) {  
        If (x < pai.elemento){  
            pai.esq = new No(x);  
        } else {  
            pai.dir = new No(x);  
        }  
    } else if (x < i.elemento) {  
        inserirPai(x, i.esq, i);  
    } else if (x > i.elemento) {  
        inserirPai(x, i.dir, i);  
    } else {  
        throw new Exception("Erro!");  
    }  
}
```



Agenda

- Funcionamento básico
- Exemplo
- Inserção em Java com retorno de referência
- Inserção em Java com passagem de pai
- Análise de complexidade 

Análise de Complexidade da Inserção

- **Melhor Caso:** $\Theta(1)$ comparações e acontece, por exemplo, inserindo na raiz
- **Pior Caso:** $\Theta(n)$ comparações e acontece, por exemplo, quando inserimos os elementos na ordem crescente ou decrescente
- **Caso Médio:** $\Theta(\lg(n))$ comparações e acontece, por exemplo, quando inserimos um elemento na folha de uma árvore balanceada. Lembrando que a altura da árvore balanceada é $\Theta(\lg(n))$

Análise de Complexidade da Inserção

- **Melhor Caso:** $\Theta(1)$ comparações e acontece, por exemplo, inserindo na raiz
- **Pior Caso:** $\Theta(n)$ comparações e acontece, por exemplo, quando inserimos os elementos na ordem crescente ou decrescente
- **Caso Médio:** $\Theta(\lg(n))$ comparações e acontece, por exemplo, quando inserimos um elemento na folha de uma árvore balanceada. Lembrando que a altura da árvore balanceada é $\Theta(\lg(n))$

Observação (1): Dependência do formato da árvore

Observação (2): Na inserção aleatória $\approx 1,39 \times \lg(n)$ comparações