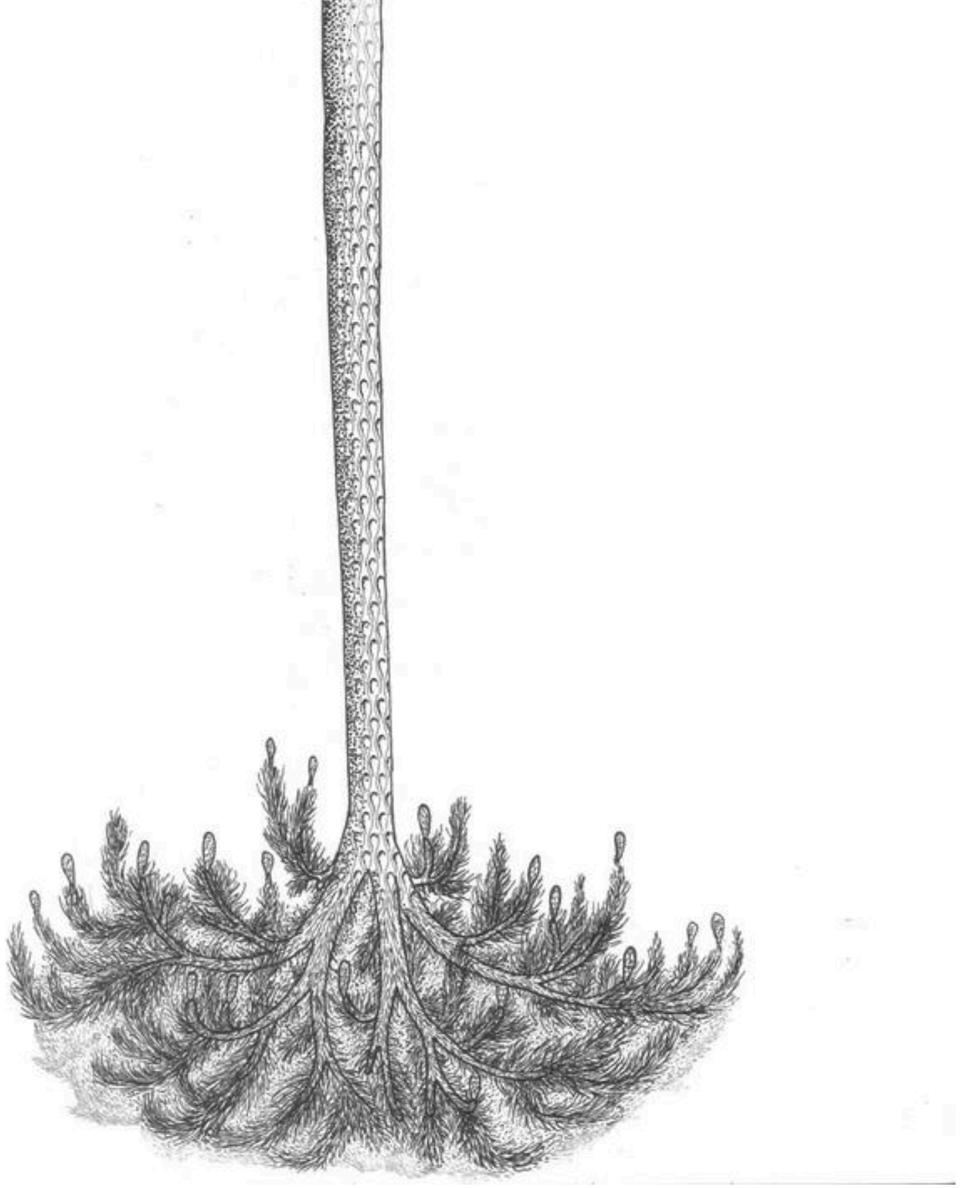
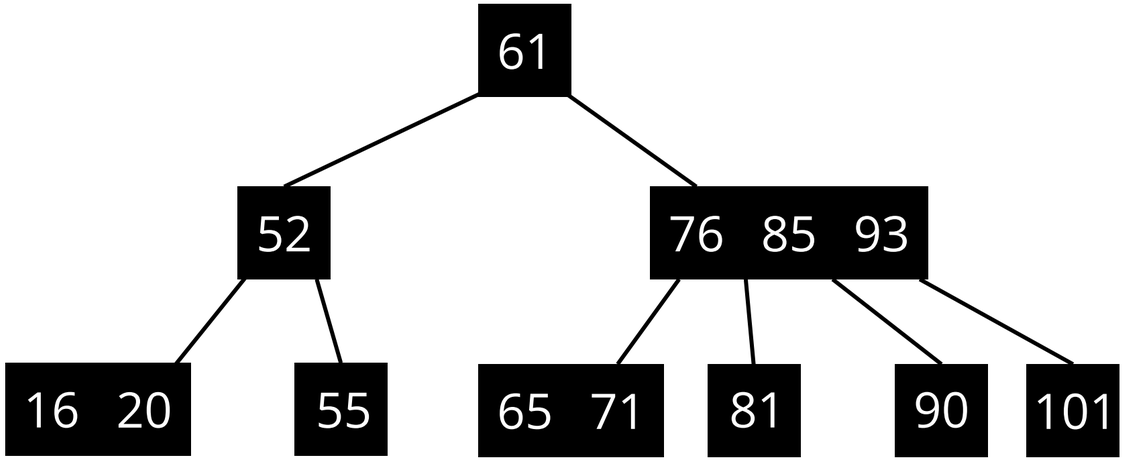
Árvore 2-3-4

Aluno: Elivan Vieira
Disciplina: Estruturas de Dados II
Professor: Adriano Antunes



Índice(n);

- Definição da Árvore 2-3-4();
- Características();
- Exemplo();
- Operações_básicas();
- // Implementado_na_linguagem_C();

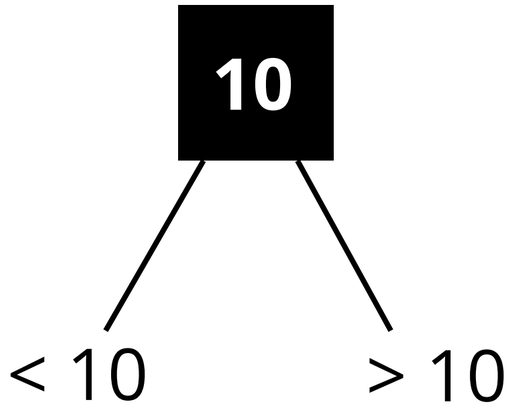


Definição();

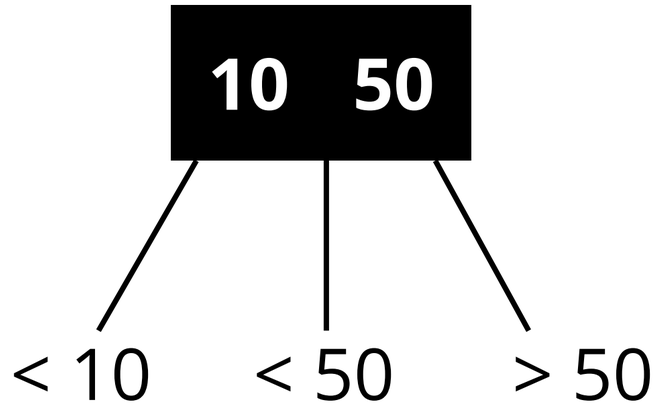
Uma árvore 2-3-4 balanceada é uma árvore de busca na qual todos os nós folha estão na mesma profundidade, garantindo que a distância da raiz até qualquer folha seja igual.

Características();

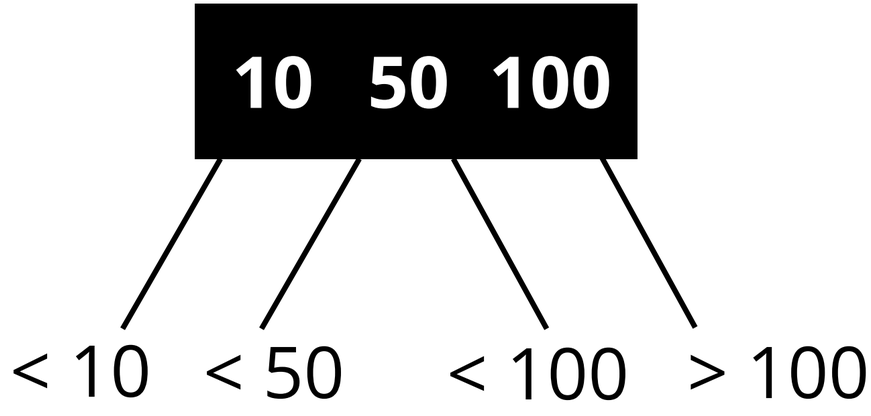
- Todos os “nós-folha” estão no mesmo nível (árvore balanceada);
- Busca eficiente;
- Remoção um “pouco” complicada
- Usada em bancos de dados;
- Complexidade O(log n);



- Uma chave com dois ponteiros



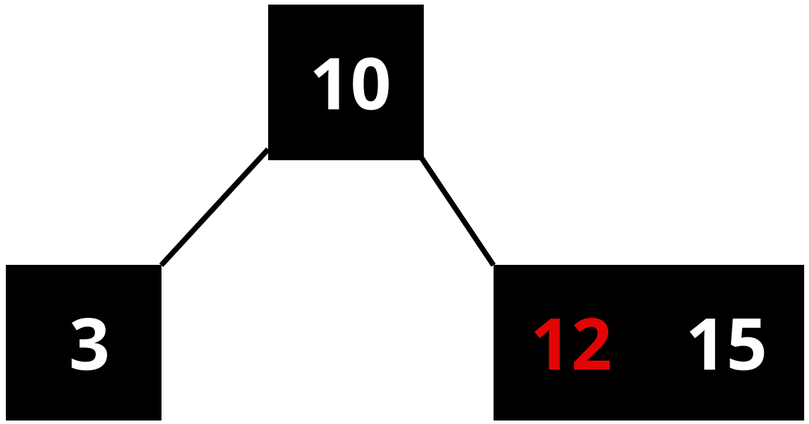
- Duas chaves com três ponteiros



- Três chaves com quatro ponteiros

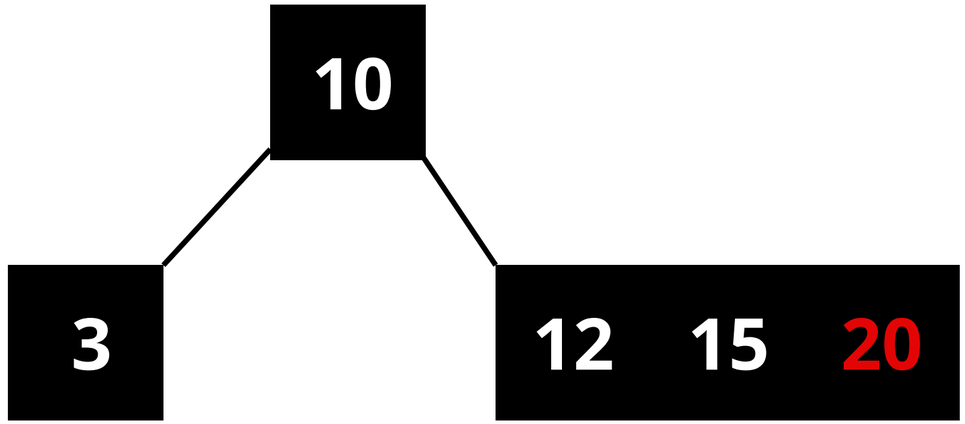
Exemplo(3, 10, 15, 12, 20, 22, 25, 30);

Exemplo(3, 10, 15, 12, 20, 22, 25, 30);

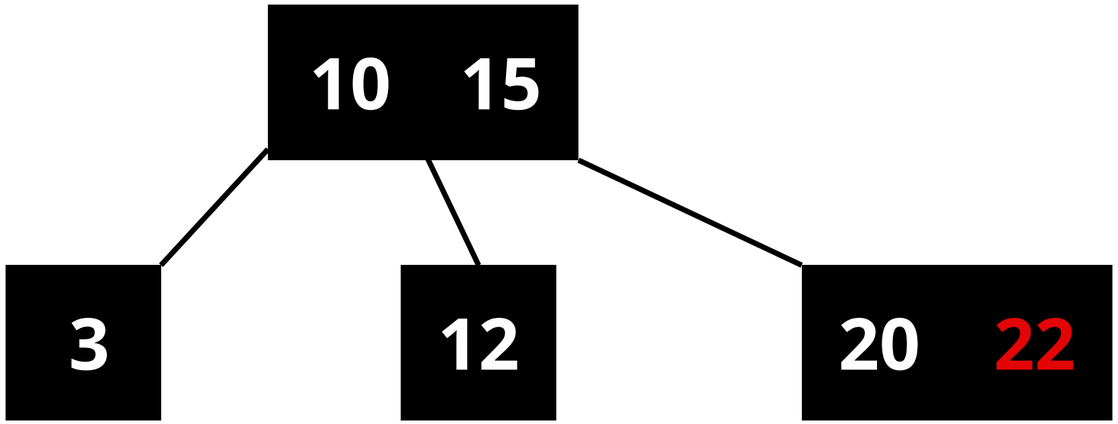


Split();

Exemplo(3, 10, 15, 12, 20, 22, 25, 30);

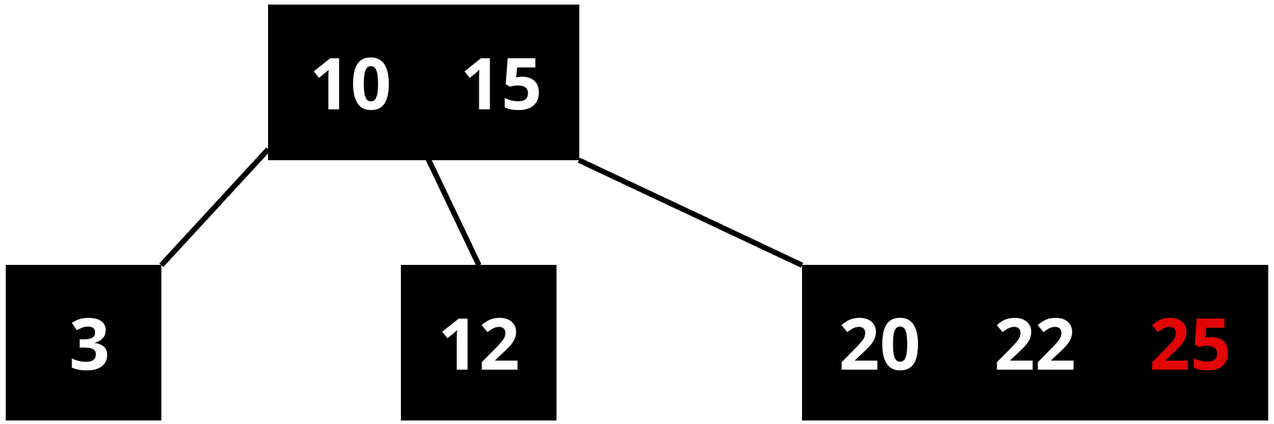


Exemplo(3, 10, 15, 12, 20, 22, 25, 30);

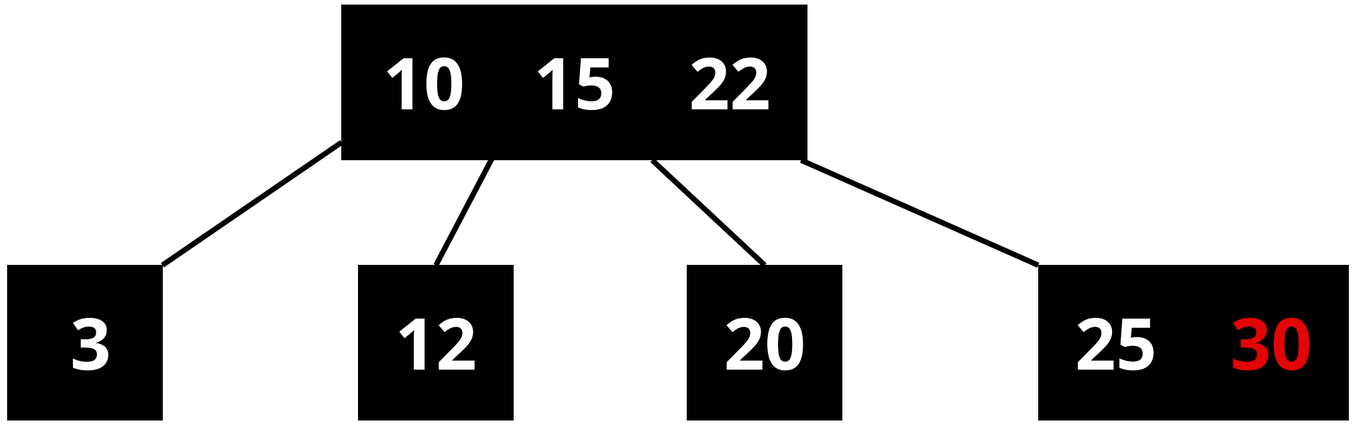


Split();

Exemplo(3, 10, 15, 12, 20, 22, 25, 30);

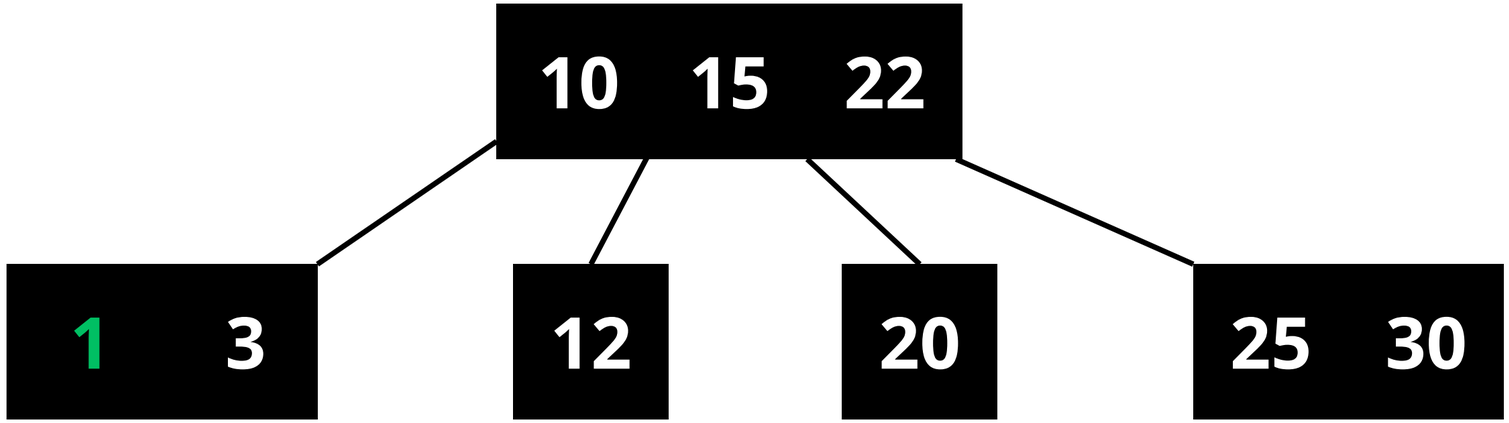


Exemplo(3, 10, 15, 12, 20, 22, 25, 30);

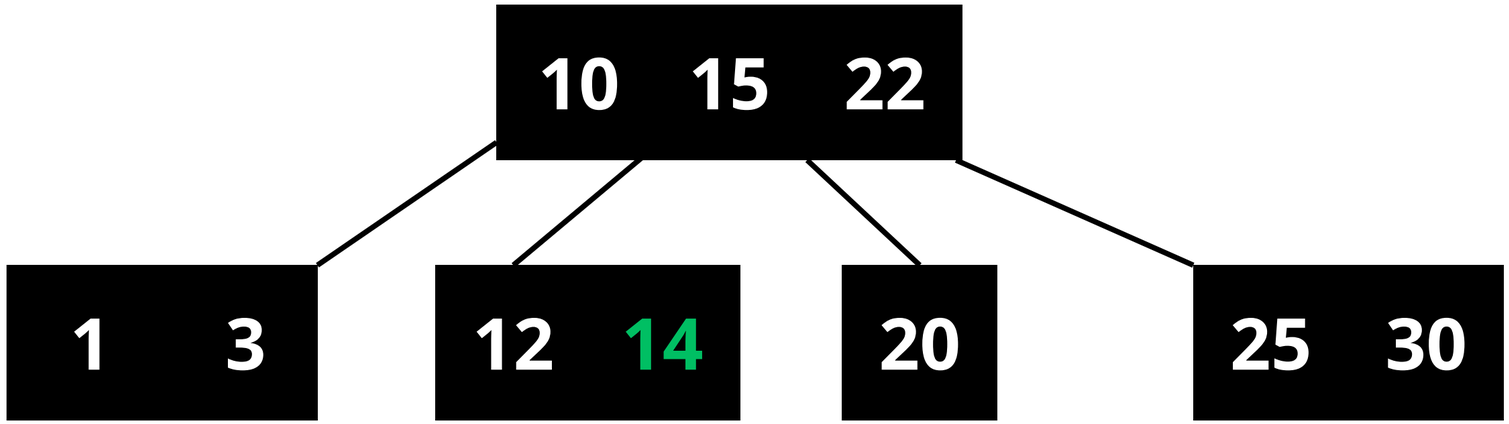


Split();

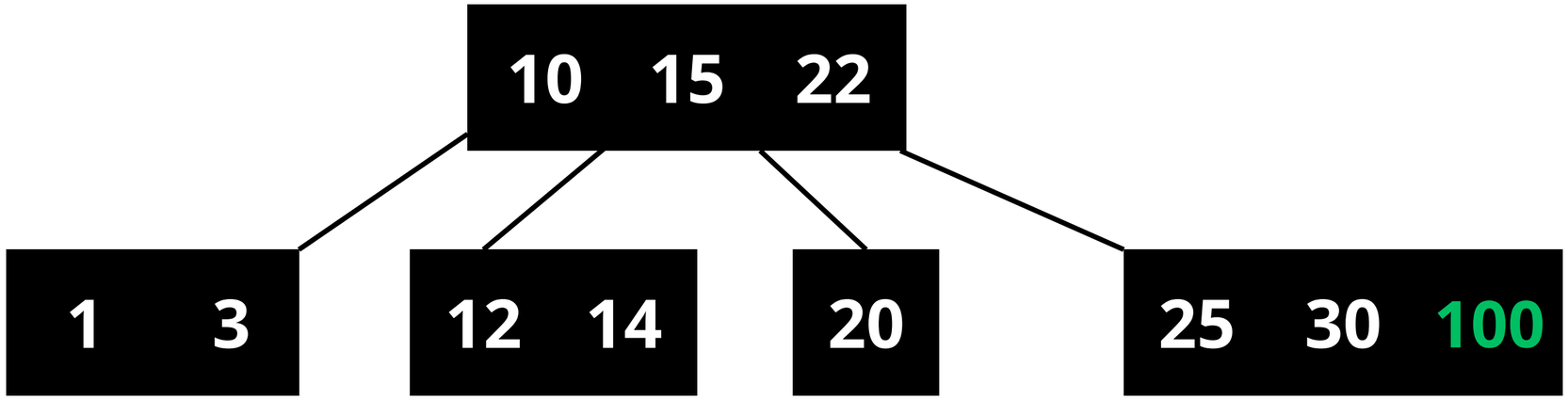
Inserindo(1, 14, 100, 21, 31, 60);



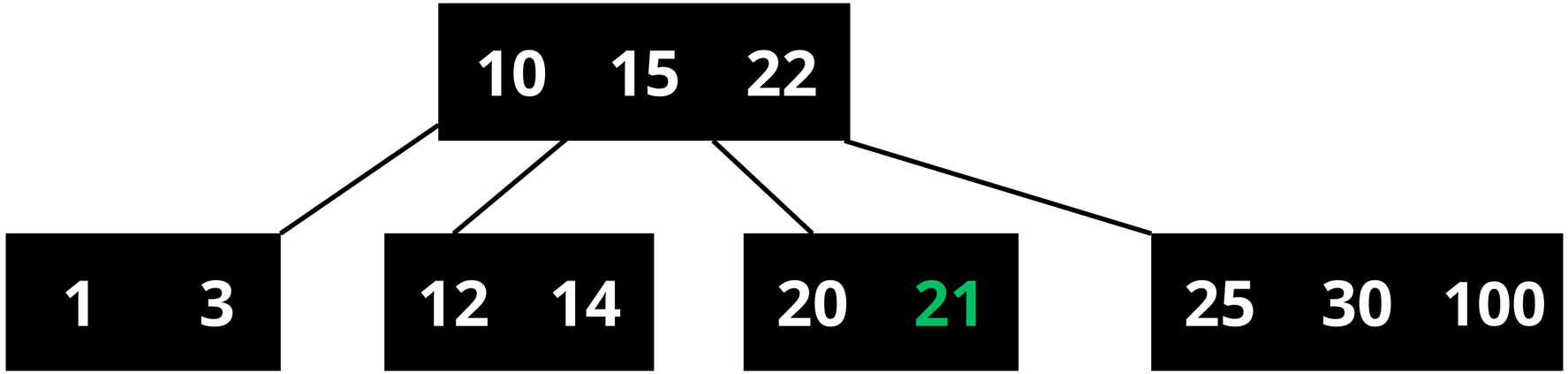
Inserindo(1, 14, 100, 21, 31, 60);



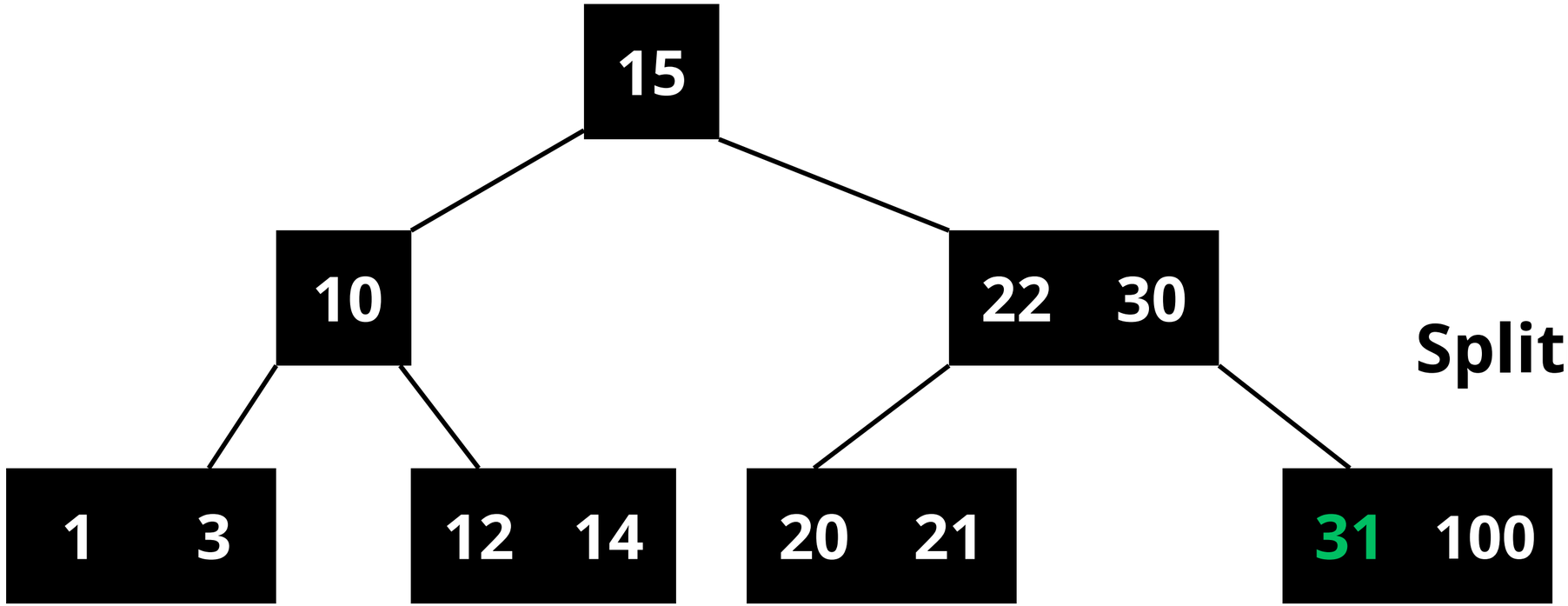
Inserindo(1, 14, 100, 21, 31, 60);



Inserindo(1, 14, 100, 21, 31, 60);

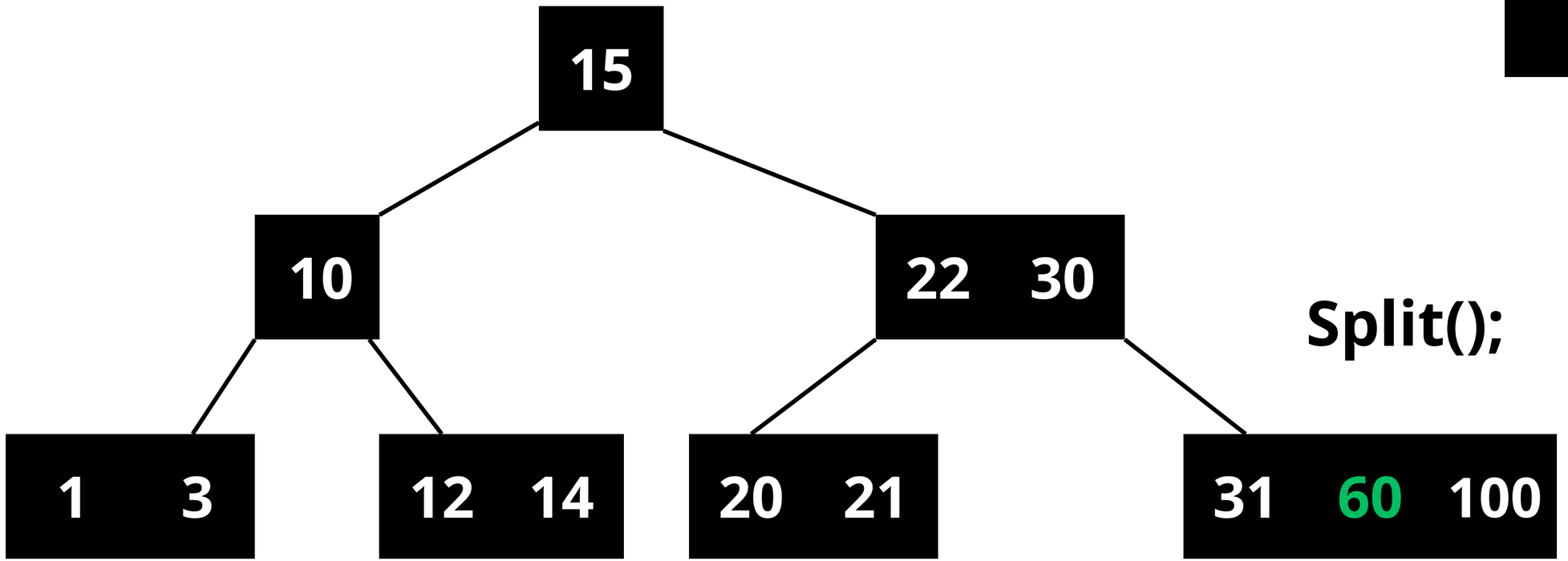


Inserindo(1, 14, 100, 21, 31, 60);



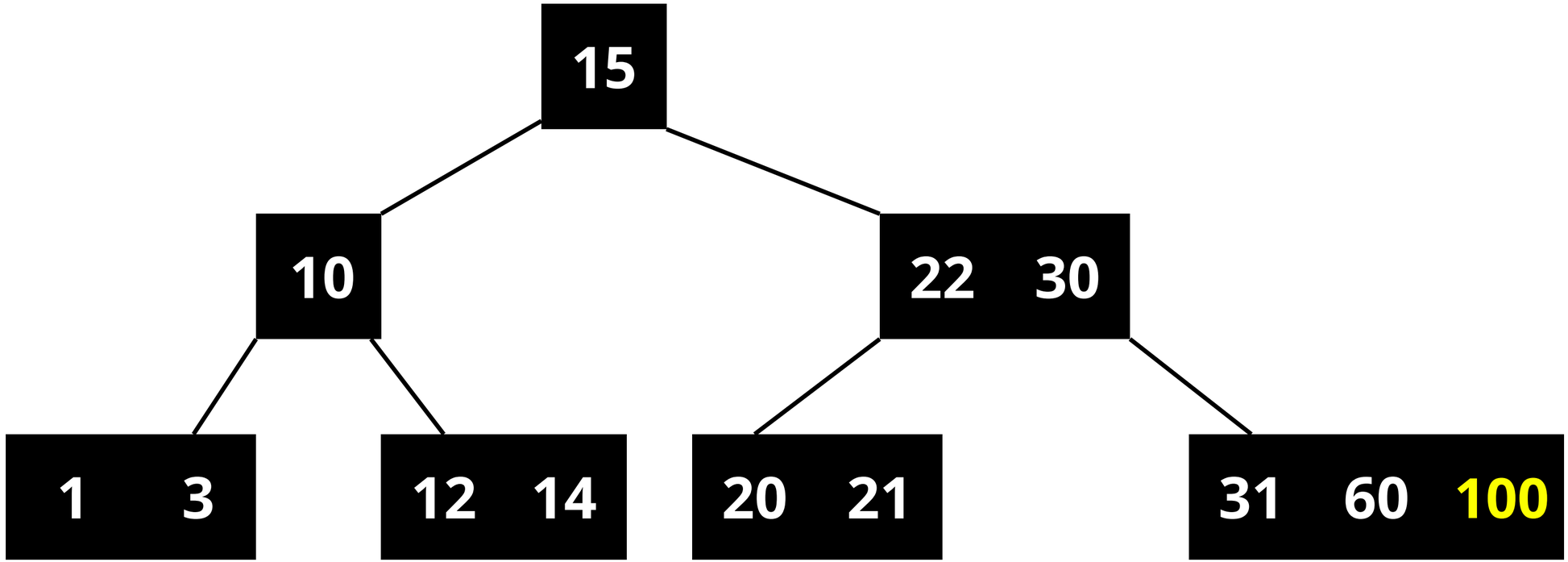
Split();

Inserindo(1, 14, 100, 21, 31, 60);

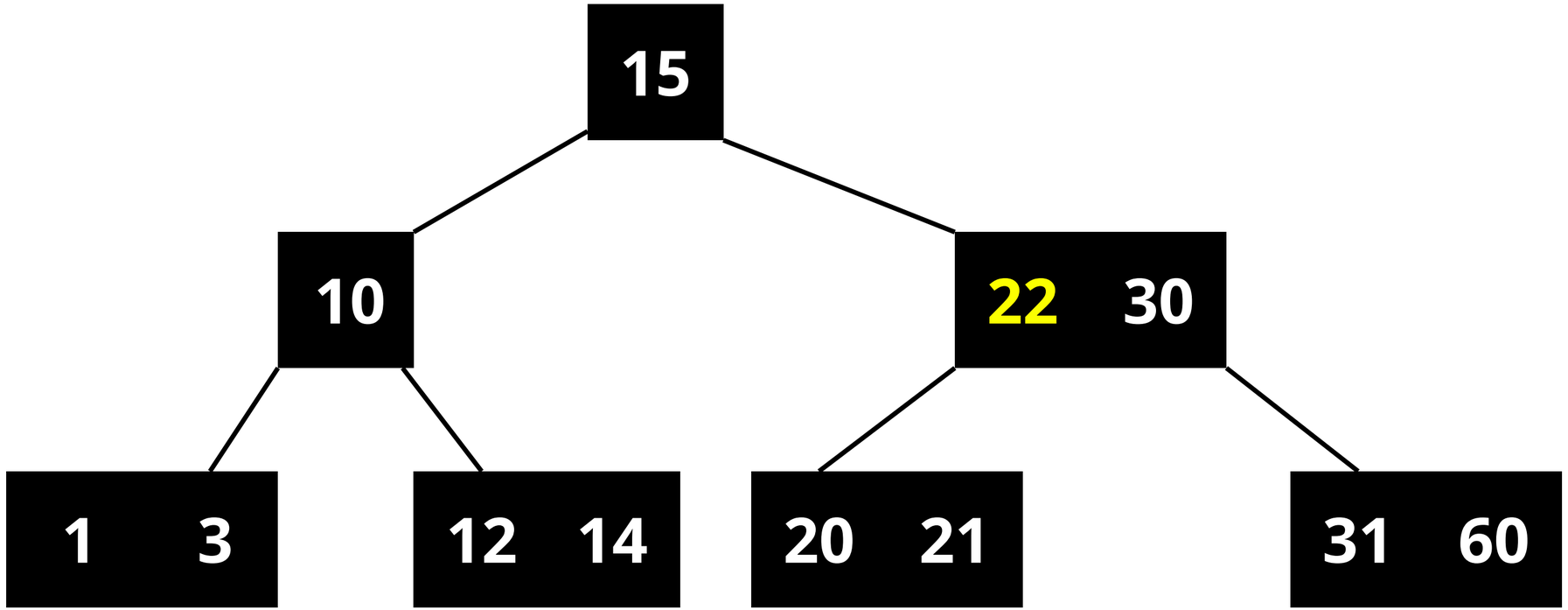


Split();

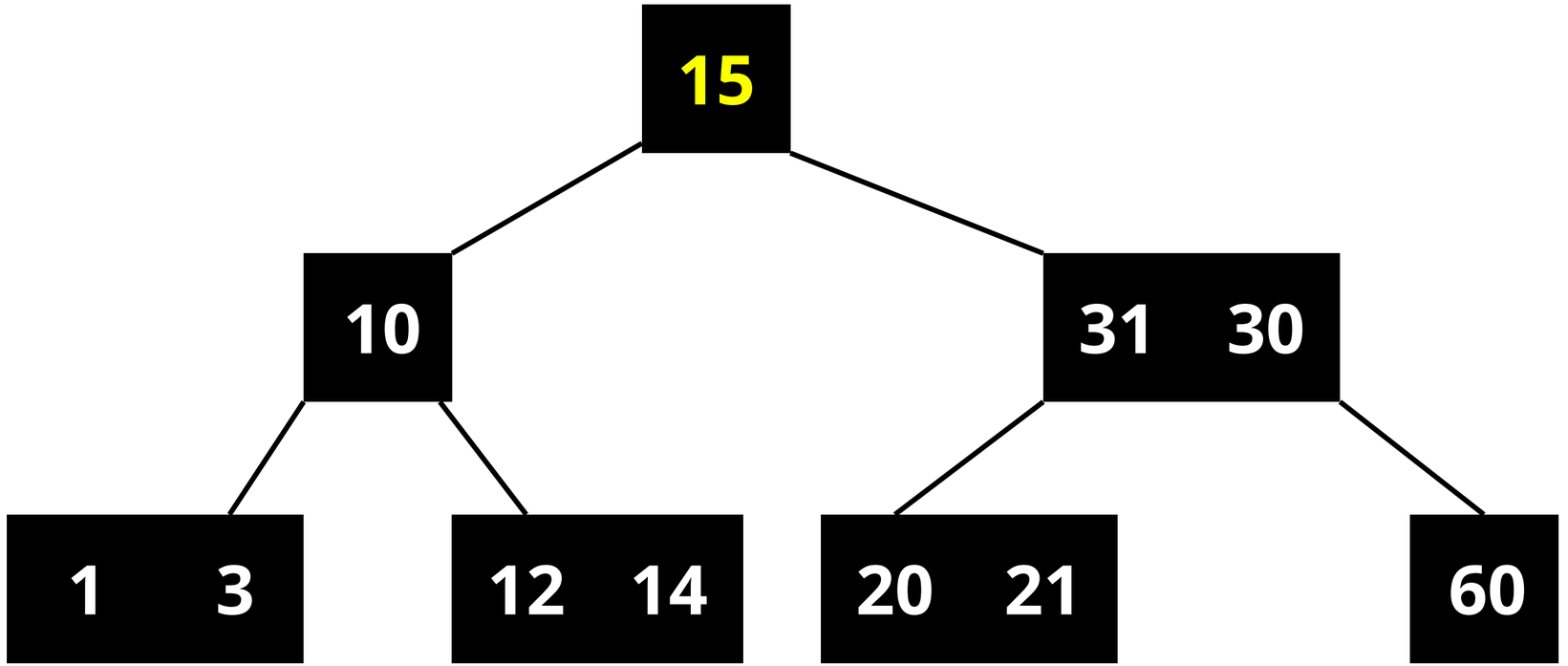
Removendo(100, 22, 15);



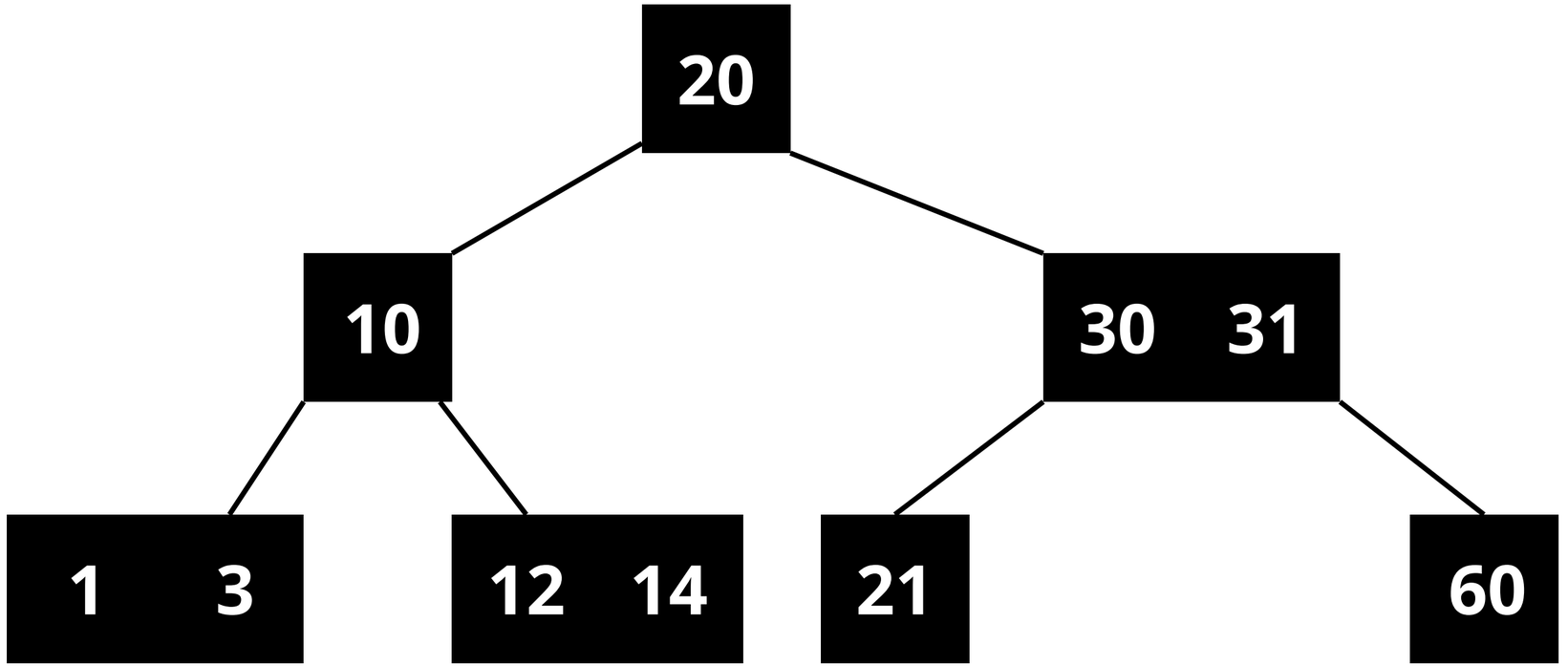
Removendo(100, 22, 15);



Removendo(100, 22, 15);



Removendo(100, 22, 15);



- **Caso 1: Removendo uma chave em um nó intermediário. Busca o maior dos menores ou o menor dos maiores e troca com a chave a ser removida. O maior ou o menor sempre estarão em uma folha.**

- **Caso 2: Removendo uma chave em um nó intermediário quando o maior dos menores e o menor dos maiores está sozinho no nó folha. Remove o elemento e faz uma junção com os filhos.**

- **Caso 3: Remoção em nó folha com mais de uma chave. Remove a chave e verifica a ordenação.**

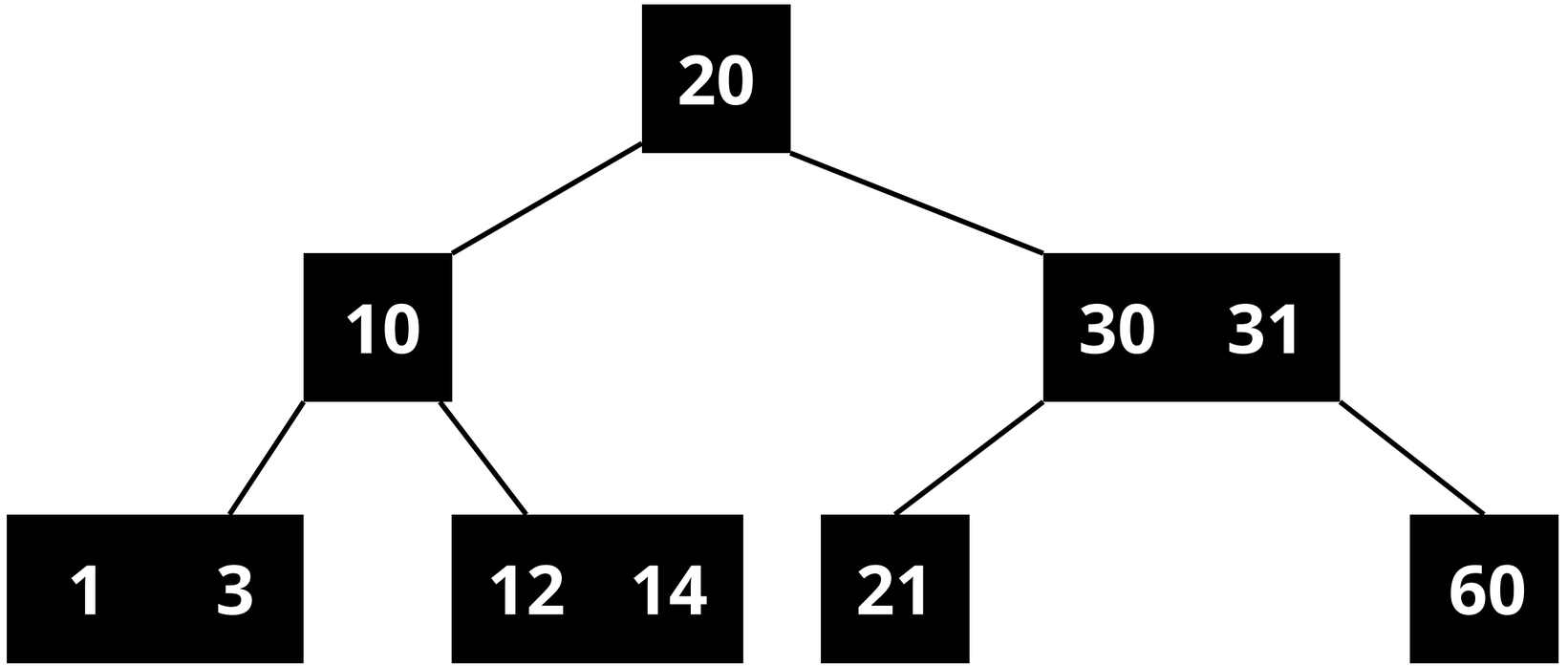
- **Caso 4: Remoção em nó folha com apenas uma chave e raiz da árvore. Remove e libera a memória do nó.**

- **Caso 5: Remoção em nó folha com apenas uma chave, não é raiz e irmão adjacente tem mais de 1 chave. Pai desce para o nó onde foi removido a chave e sobe uma chave do irmão adjacente.**

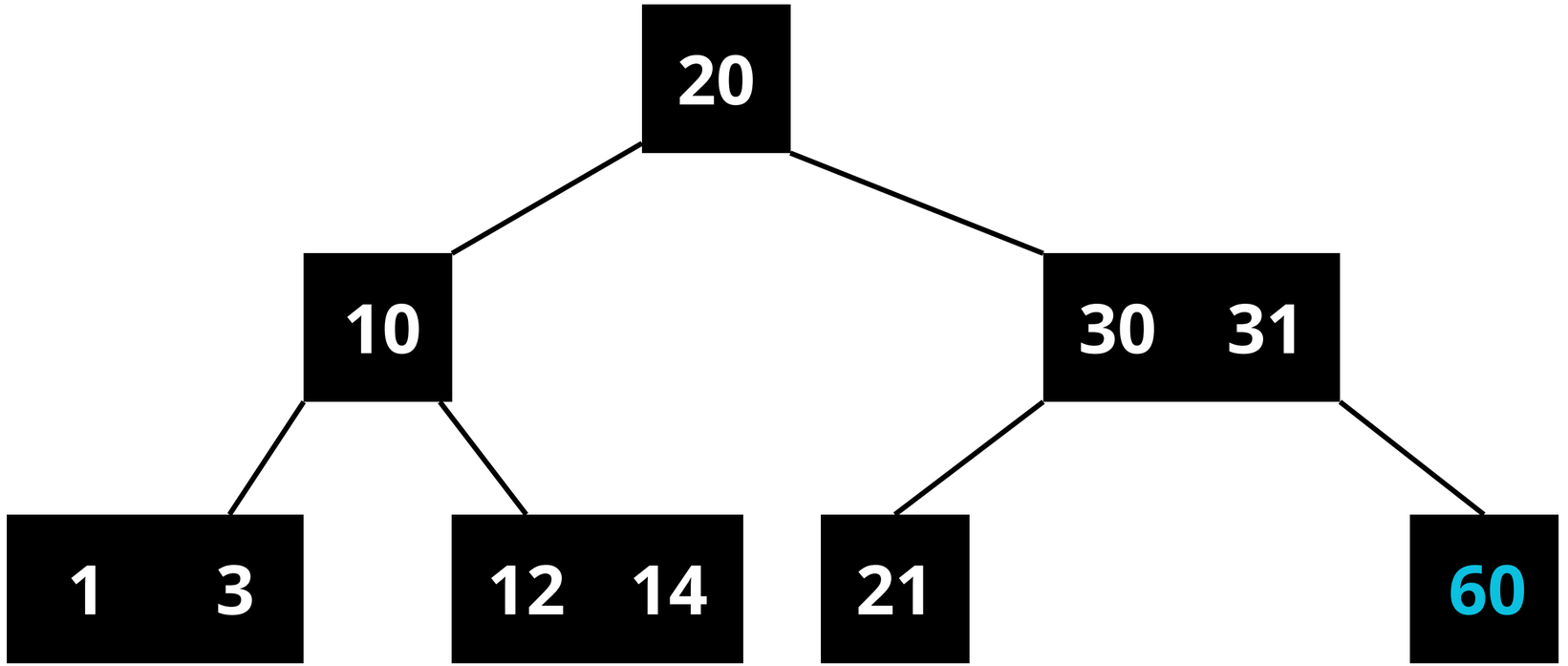
- **Caso 6: Remoção em nó folha com apenas uma chave e não é raiz. Pai e irmão com apenas 1 chave. Remove o nó e faz uma junção.**

- **Caso 7: Remoção em nó folha. No pai tee 1 chave e os irmãos têm apenas 1 chave. Junção e rotação. Uma chave do pai desce para um dos filhos e este filho ocupa a posição do nó removido.**

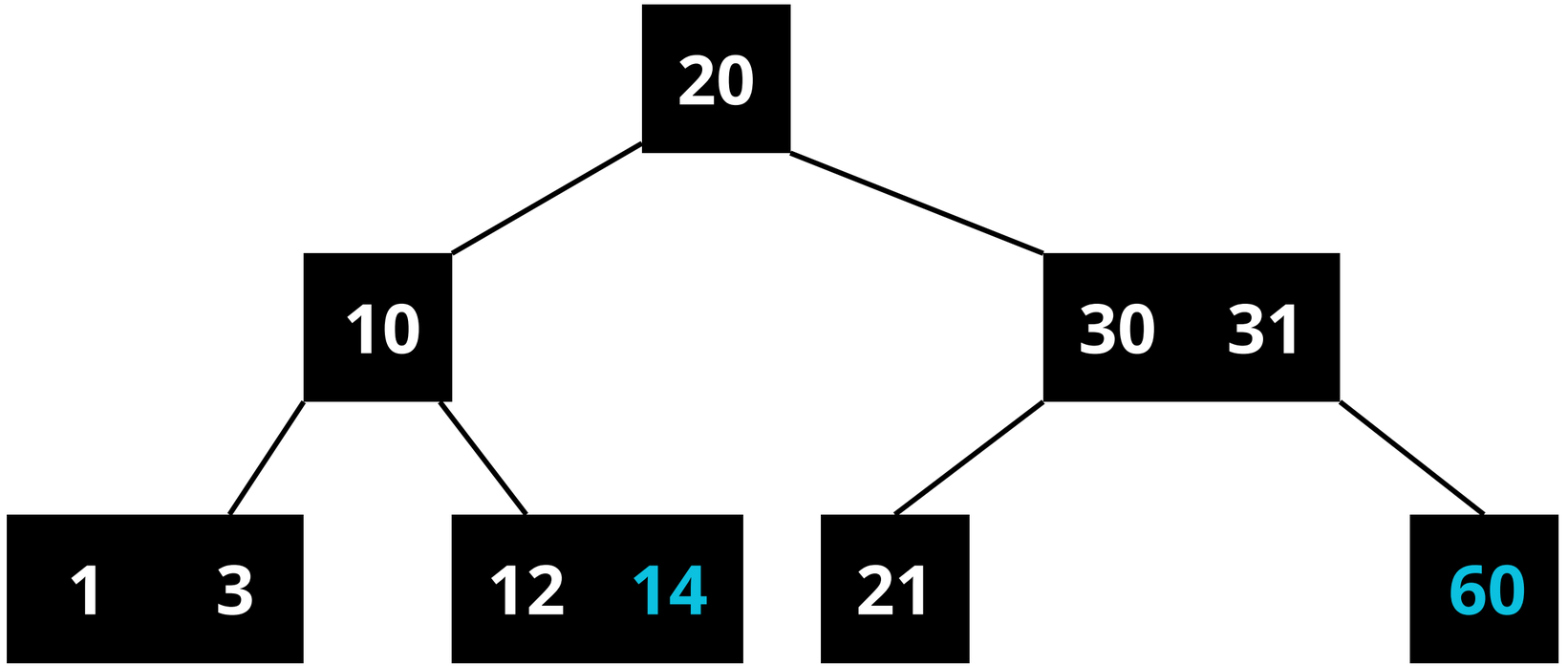
Buscando(60, 14, 999);



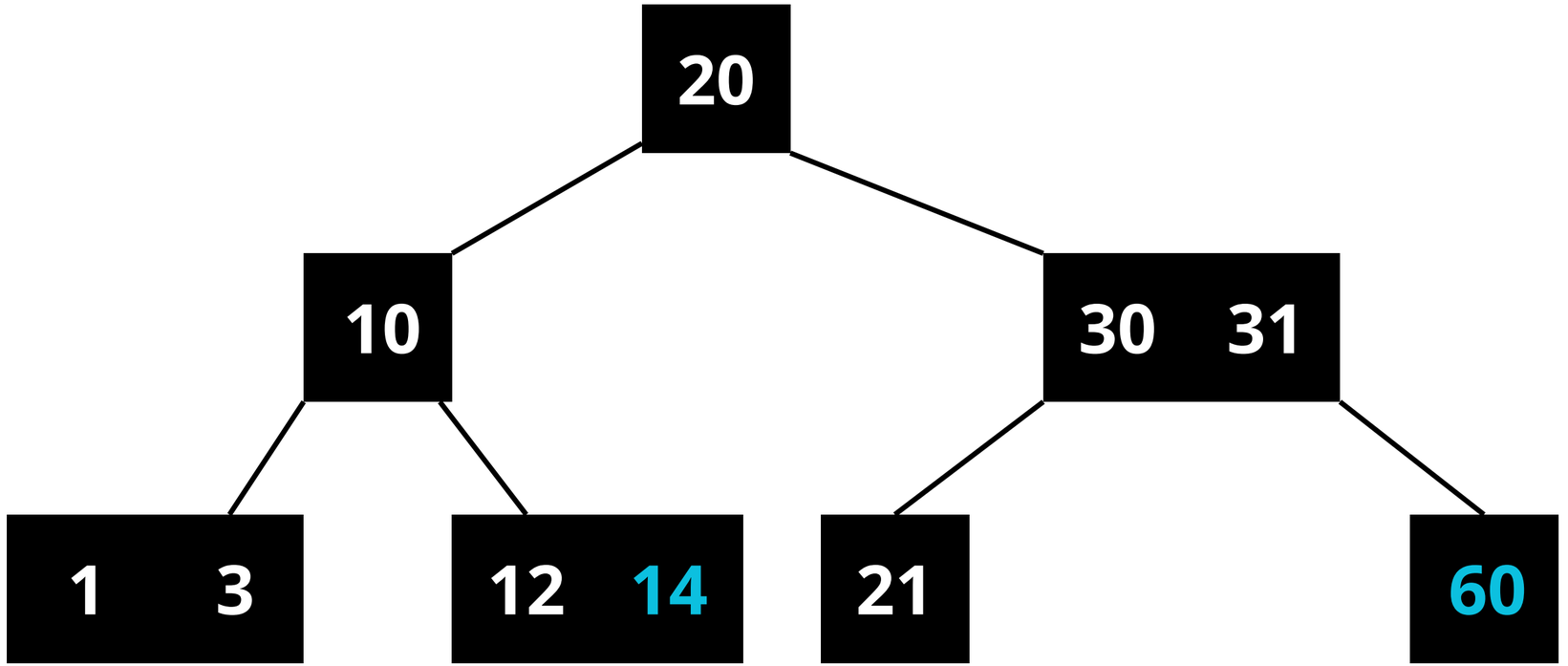
Buscando(60, 14, 999);



Buscando(60, 14, 999);



Buscando(60, 14, 999);



Obrigado pelo tempo/atenção!