

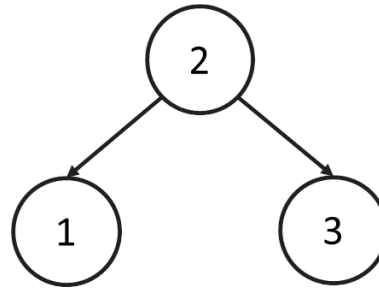
Estrutura de Dados II

Ciência da Computação

Prof. André Kishimoto
2023

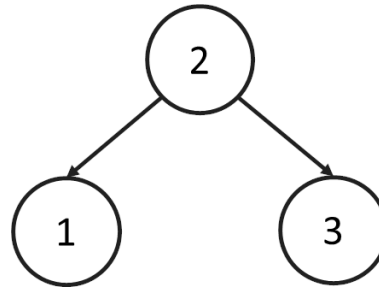
Árvore binária de busca [binária] (BST)

- BST: *Binary Search Tree*.
- Árvore binária com uma nova regra:
 - *Dado um nó T da árvore, todas as subárvores que se encontram à **esquerda de T** possuem nós com valores menores que T e todas as subárvores que se encontram à **direita de T** possuem nós com valores maiores que T .*



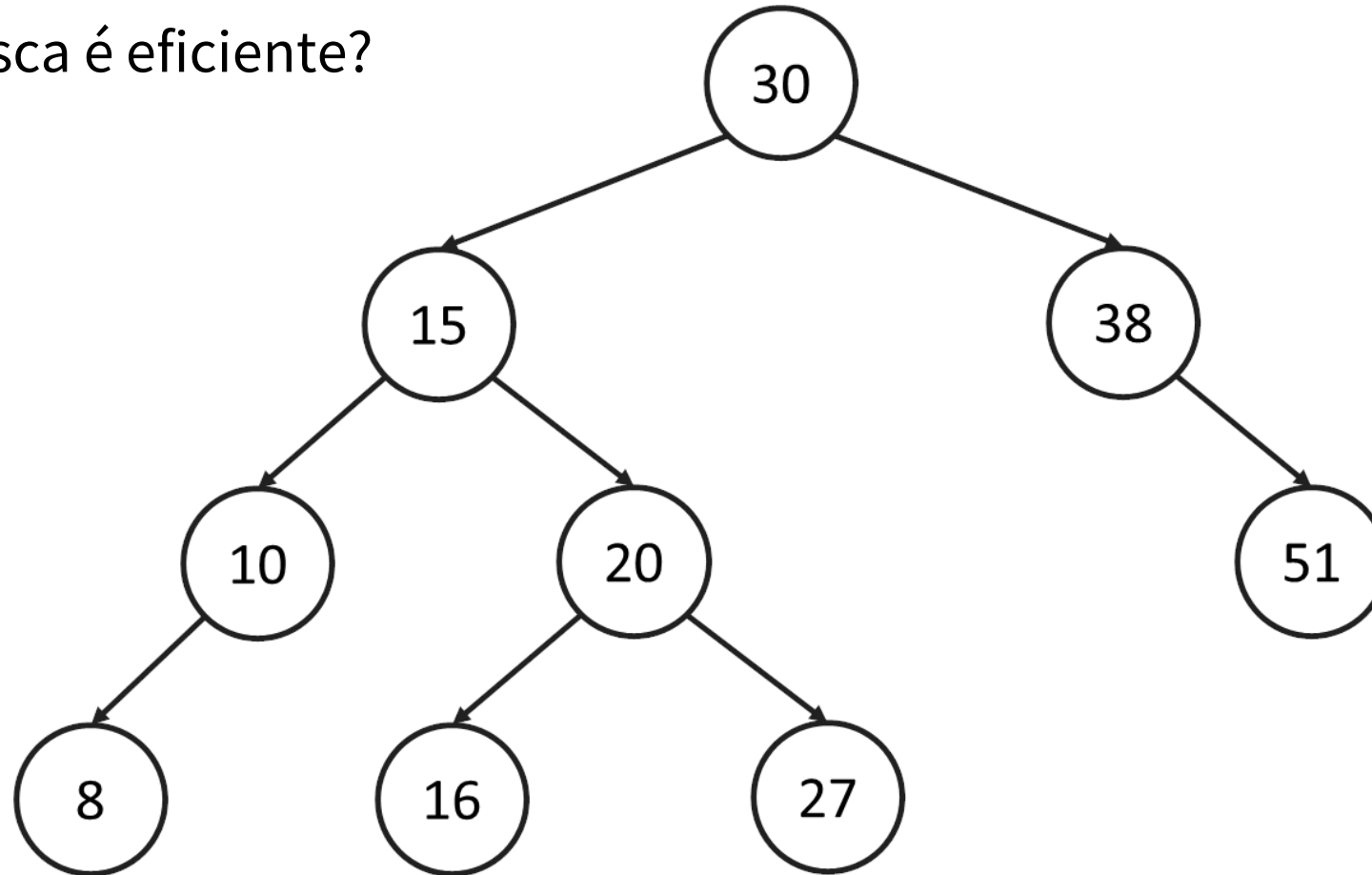
Árvore binária de busca (BST)

- O que define um nó ter um valor (chave) menor ou maior do que outro nó?
 - Depende do contexto, desde que seja possível comparar se um valor é menor, maior ou igual a outro.
 - Igualdade: geralmente, uma BST não tem nós com valores duplicados.
 - Exemplo mais simples: números inteiros e positivos.



Árvore binária de busca (BST)

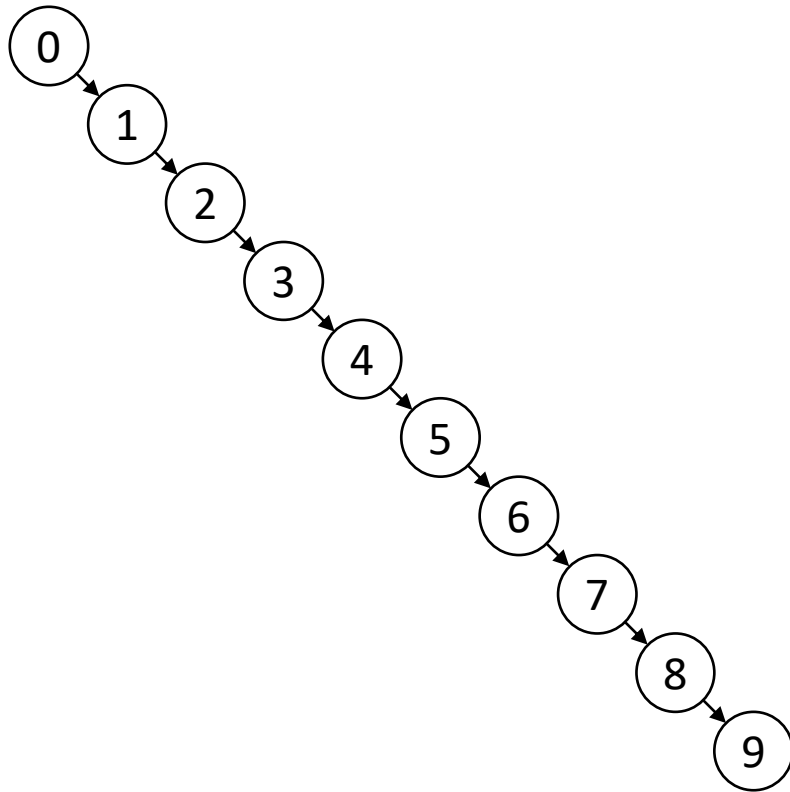
Por que a busca é eficiente?



Árvore binária de busca (BST)

- Vamos inserir os seguintes elementos em uma BST vazia, na ordem apresentada:
- 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.
- Como fica a representação visual da árvore?

Árvore binária de busca (BST)



- A ordem da inserção dos elementos afeta a construção da BST...
- Nas próximas aulas, veremos uma outra árvore que resolve esse problema (de balanceamento de árvore).

BST – Busca

- BST é uma árvore binária para realizar buscas binárias.
 - Divisão e conquista.
 - Lembra da busca binária em um vetor de elementos ordenados?
 - A ideia de busca em uma BST é a mesma!
- Começamos pela raiz da árvore.
 - Se a chave da raiz é a que buscamos, fim.
 - Senão, comparamos a chave a ser buscada com a raiz.
 - É menor do que a raiz? Continua a busca pela subárvore esquerda.
 - É maior do que a raiz? Continua a busca pela subárvore direita.
 - Continua até encontrar o nó com a chave OU encontrar um nó nulo (chave não existe na árvore).

BST – Busca

Exemplo:

Buscar o elemento 20.

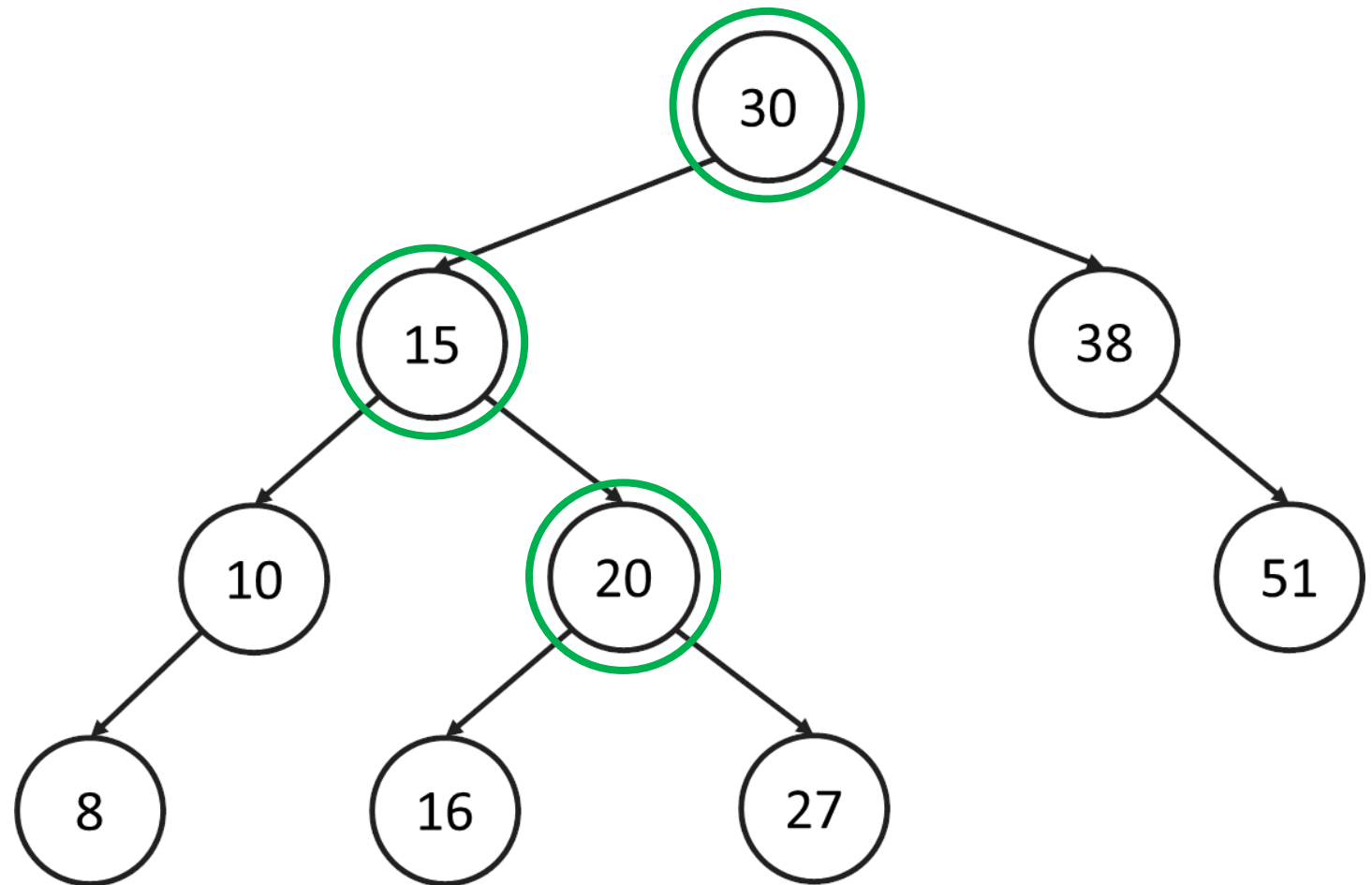
Quantas comparações?

3 comparações.

Complexidade:

- Pior caso, $O(n)$
- Média, $O(\log n)$

n : quantidade de elementos



BST – Inserção

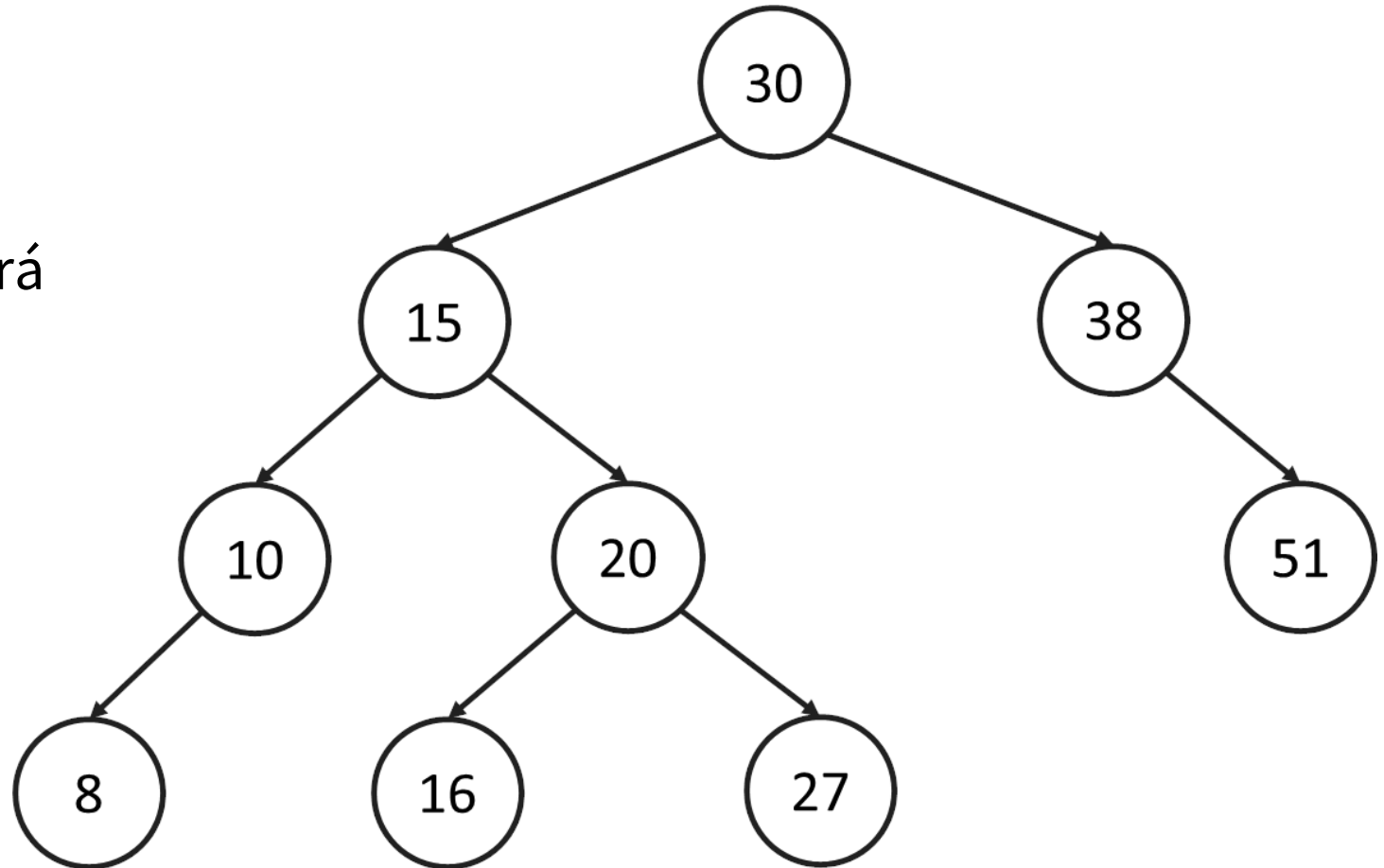
- Primeira etapa é similar à busca: procuramos uma posição na árvore para inserir o novo nó, seguindo a regra da BST.
 - Chave menor que um nó T deve ser inserido à esquerda.
 - Chave maior que um nó T deve ser inserido à direita.
- Percorrer a árvore e encontrar um nó nulo:
 - Para a operação de busca: busca concluída sem encontrar a chave especificada.
 - Para a operação inserção: encontramos a posição para a nova chave.

BST – Inserção

Exemplo:

Inserir o elemento 35.

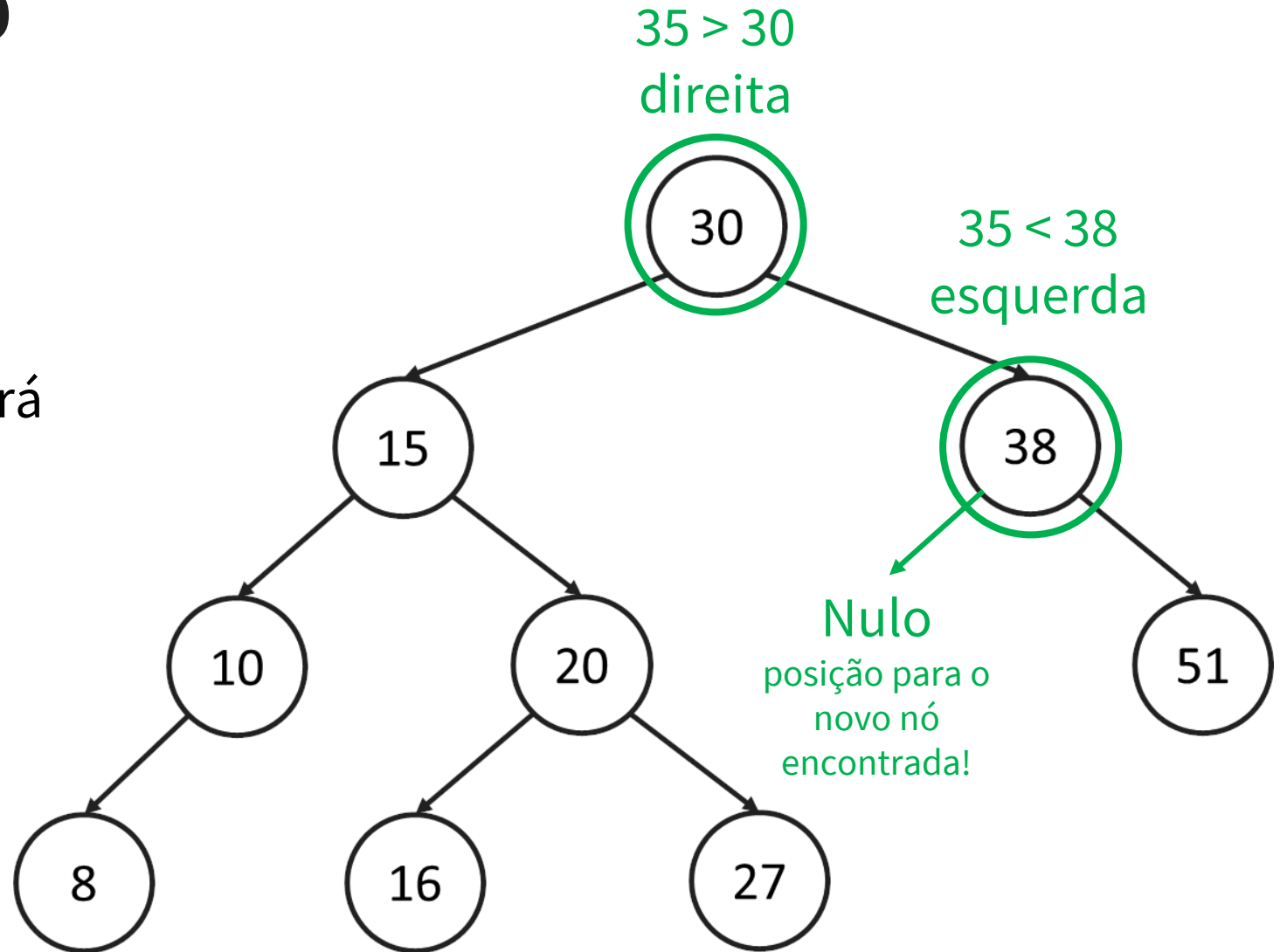
Onde o nó com a chave 35 será inserido na árvore ao lado?



BST – Inserção

Exemplo:
Inserir o elemento 35.

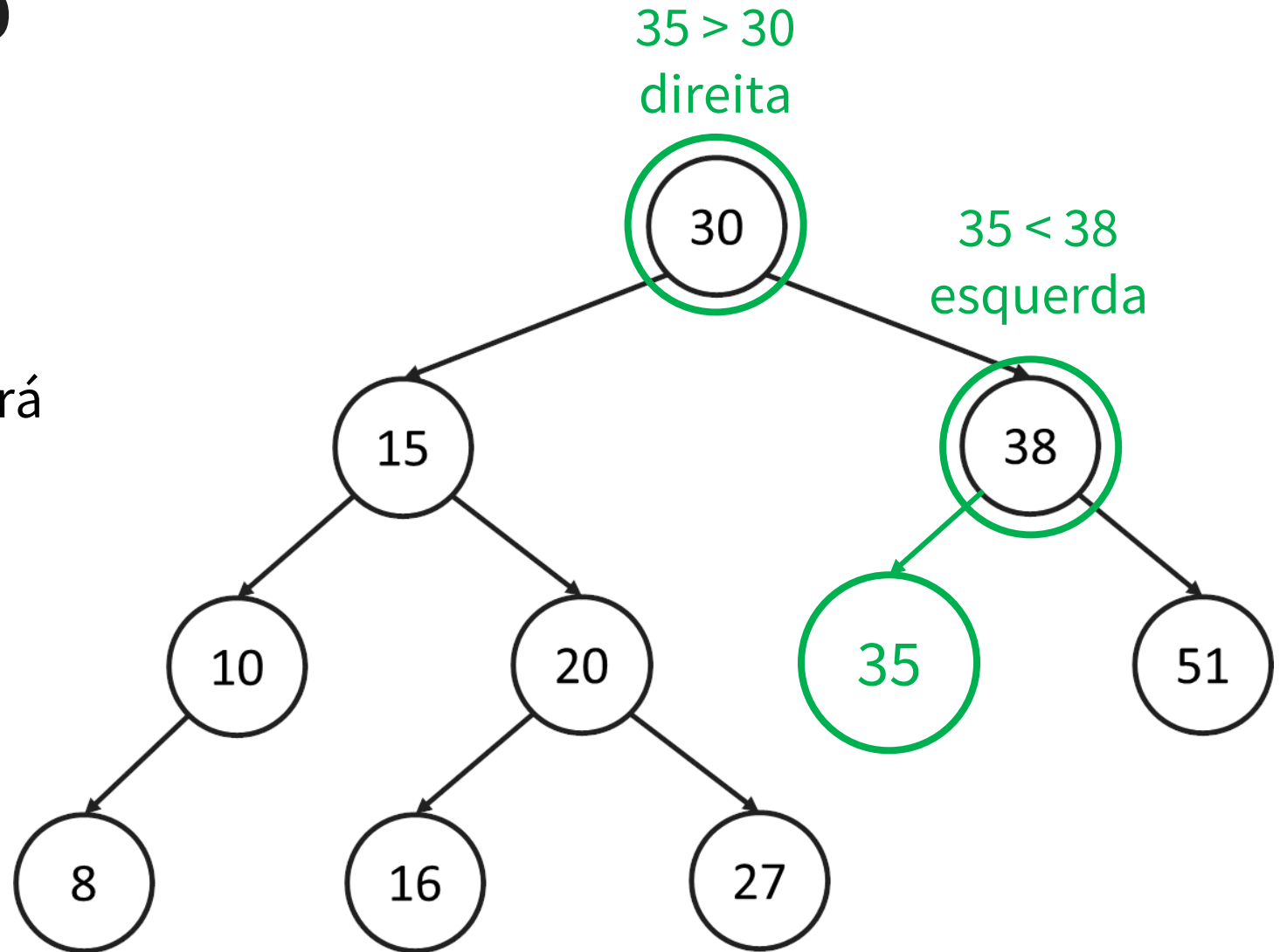
Onde o nó com a chave 35 será
inserido na árvore ao lado?



BST – Inserção

Exemplo:
Inserir o elemento 35.

Onde o nó com a chave 35 será
inserido na árvore ao lado?



BST – Remoção

- Assim como na busca e inserção, precisamos realizar uma busca na BST para encontrar o nó a ser removido.
- Diferente da busca e inserção, a operação de remoção possui algumas condições que precisamos analisar.
 - Precisamos garantir que, após um nó ser removido, a árvore continue sendo uma BST.

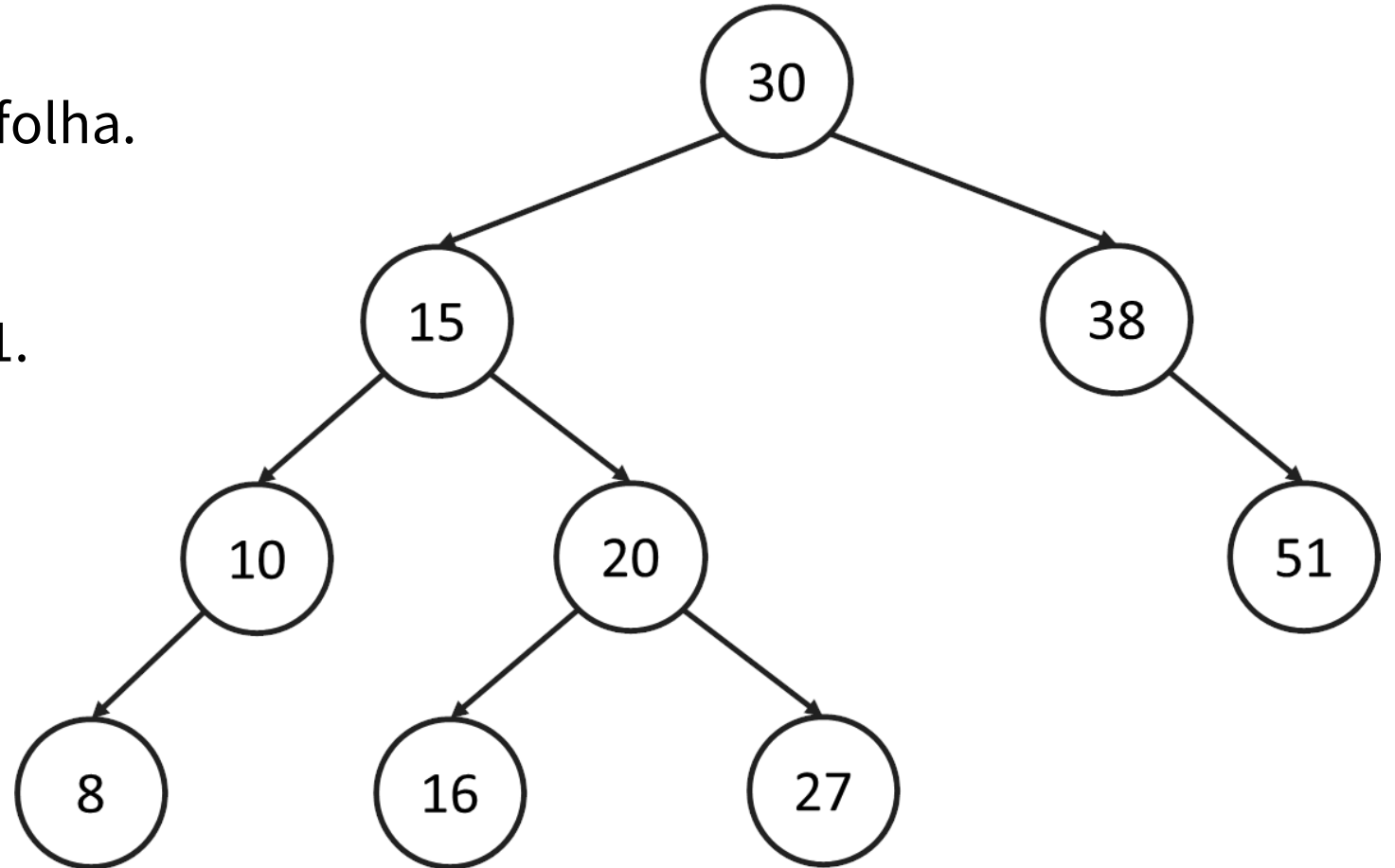
BST – Remoção

- Existem quatro casos para analisar a remoção de um nó da BST:
 1. O nó a ser removido é uma folha.
 2. O nó a ser removido não possui filho esquerdo (não possui subárvore da esquerda).
 3. O nó a ser removido não possui filho direito (não possui subárvore da direita).
 4. O nó a ser removido possui os filhos esquerdo e direito (possui subárvores da esquerda e da direita).

BST – Remoção

Caso 1:
O nó a ser removido é uma folha.

Exemplo:
Remover o nó com chave 51.



BST – Remoção

Caso 1:

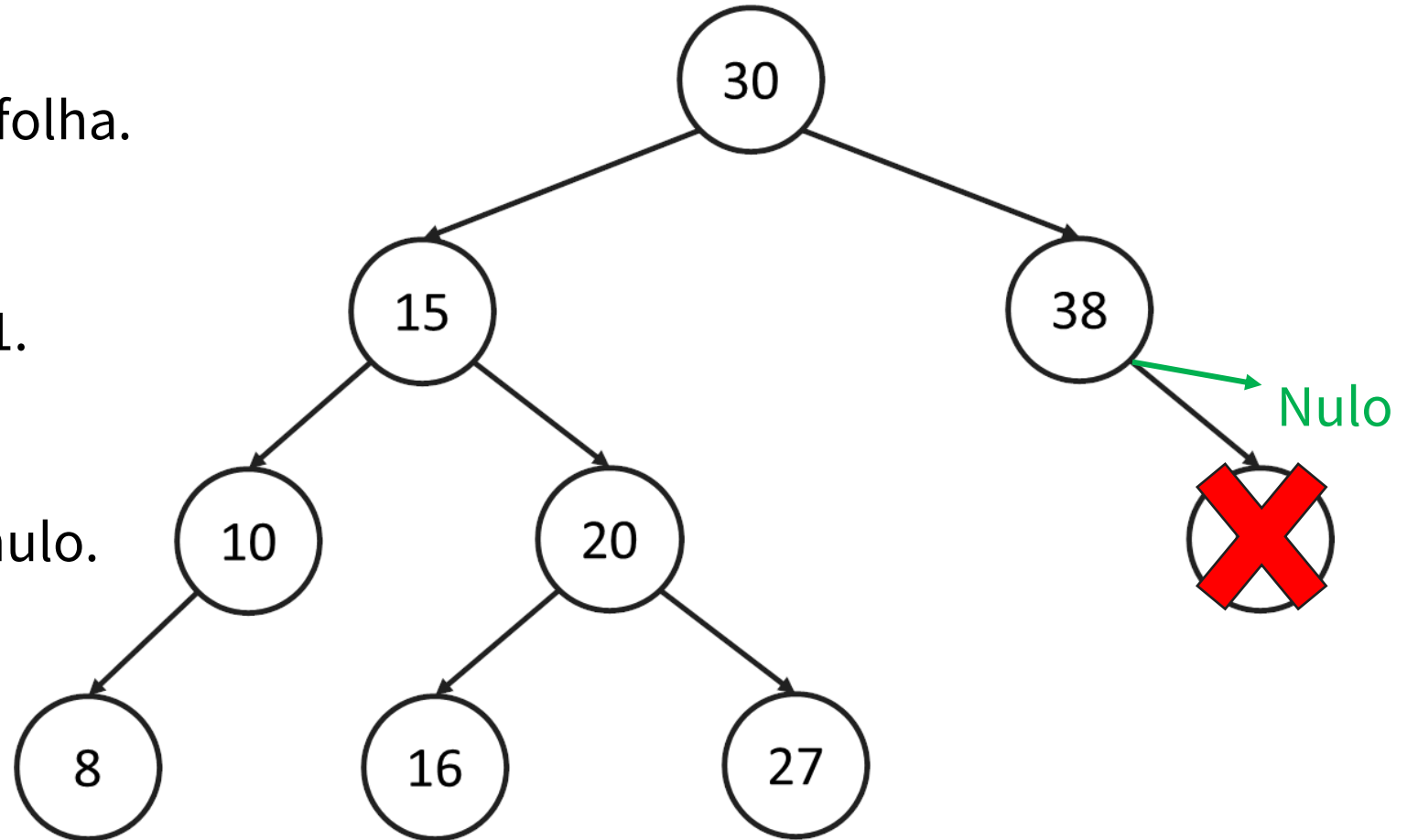
O nó a ser removido é uma folha.

Exemplo:

Remover o nó com chave 51.

Resultado:

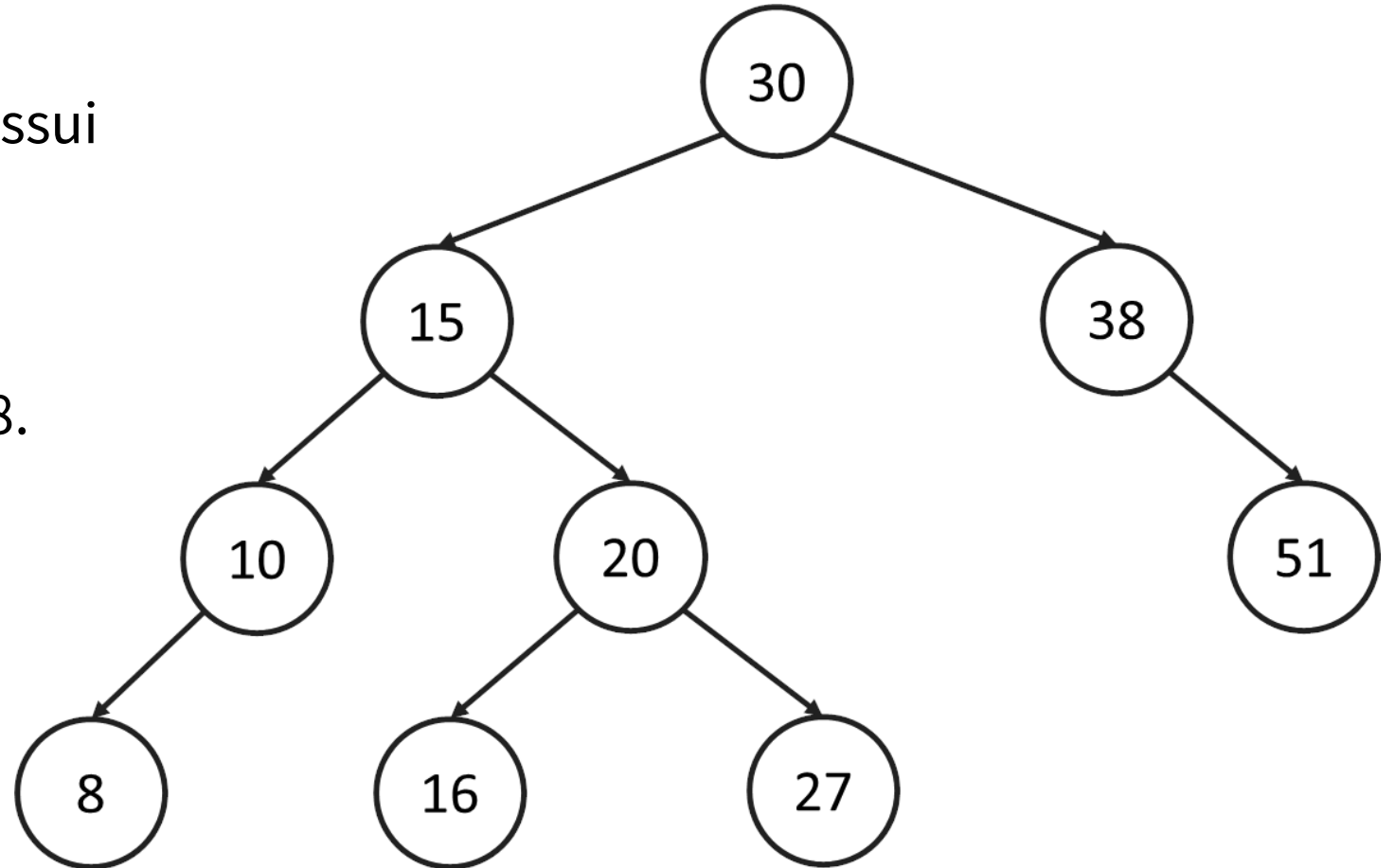
Filho direito de 38 agora é nulo.



BST – Remoção

Caso 2:
O nó a ser removido não possui
subárvore esquerda.

Exemplo:
Remover o nó com chave 38.



BST – Remoção

Caso 2:

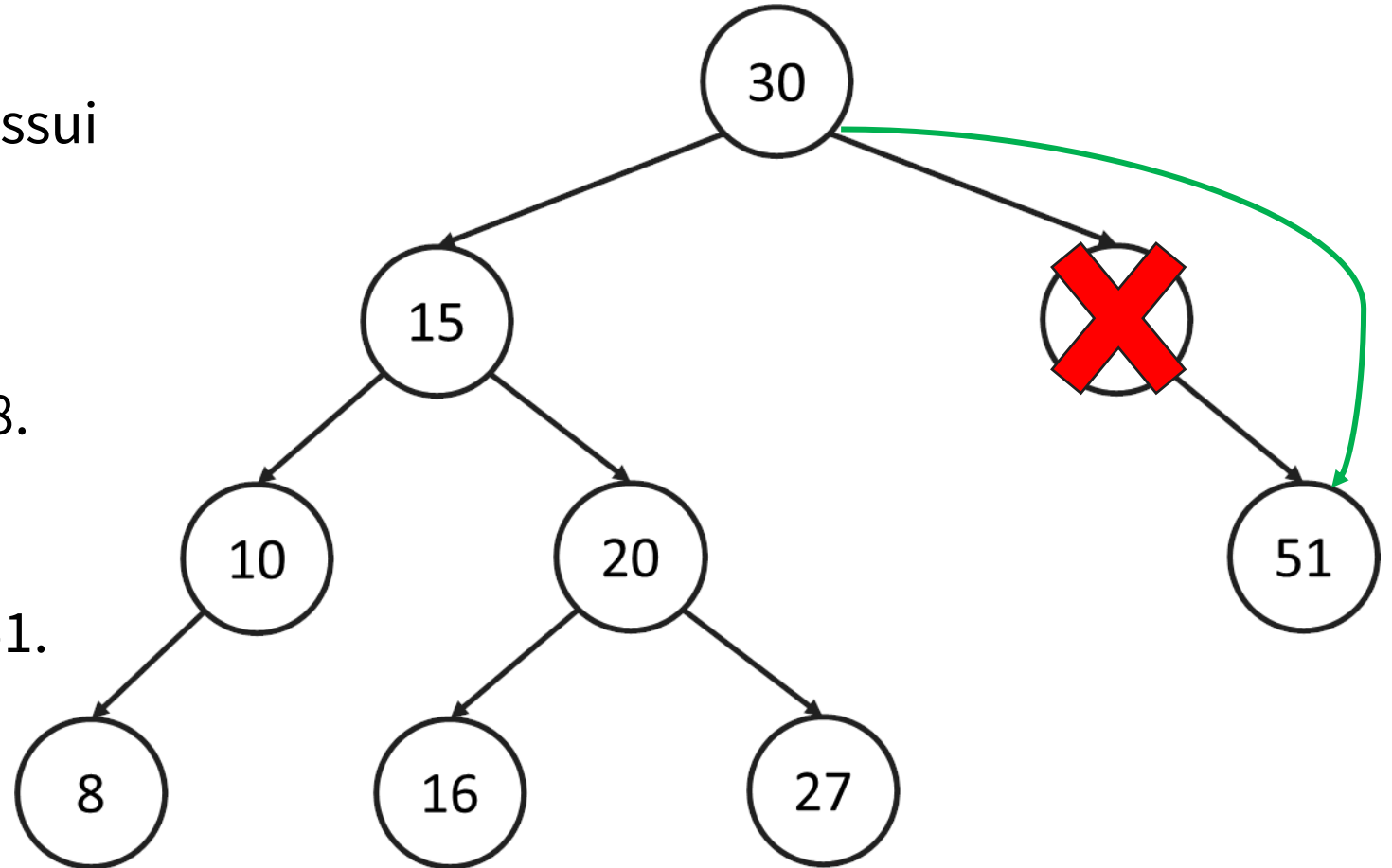
O nó a ser removido não possui subárvore esquerda.

Exemplo:

Remover o nó com chave 38.

Resultado:

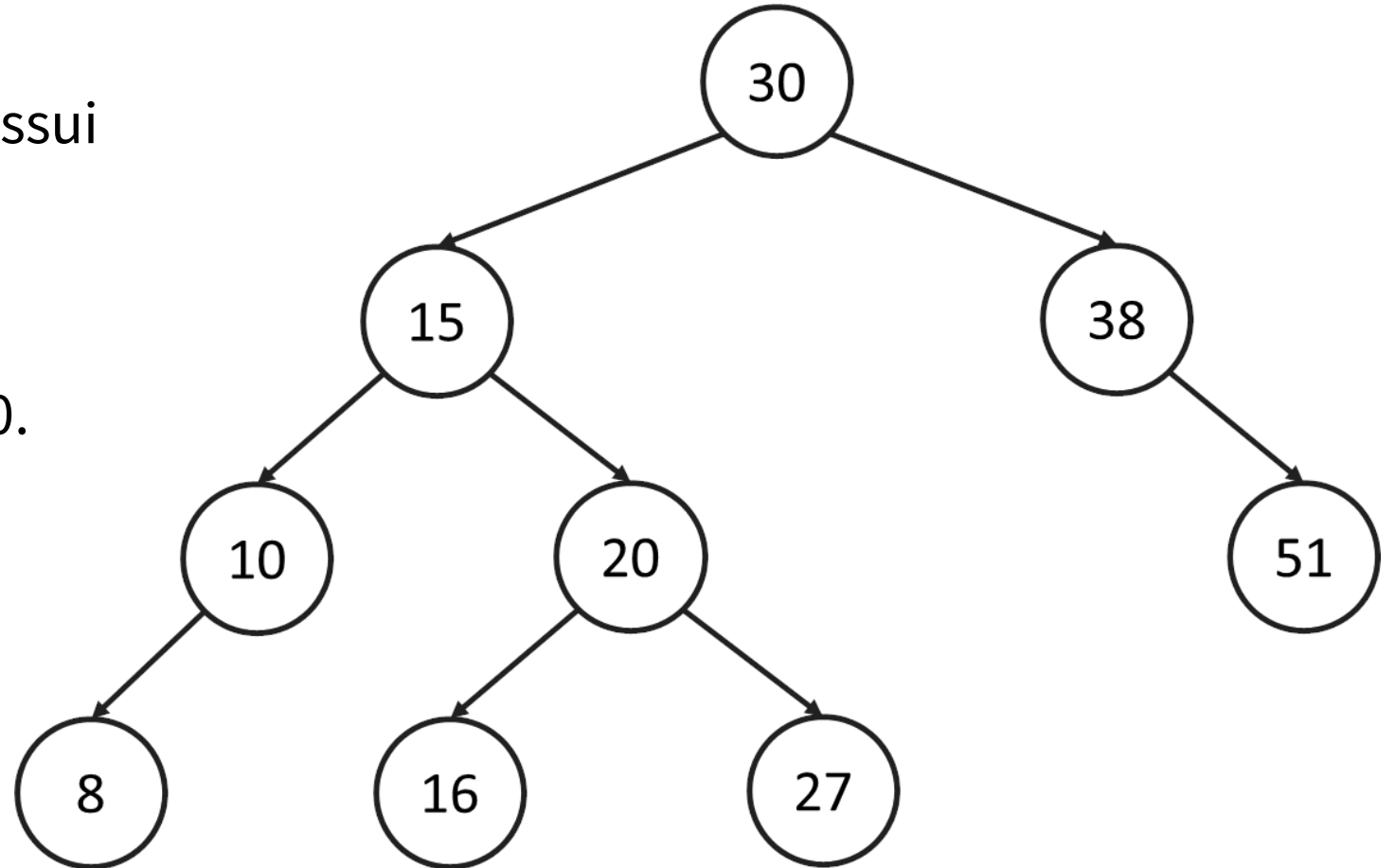
Filho direito de 30 agora é 51.



BST – Remoção

Caso 3:
O nó a ser removido não possui
subárvore direita.

Exemplo:
Remover o nó com chave 10.

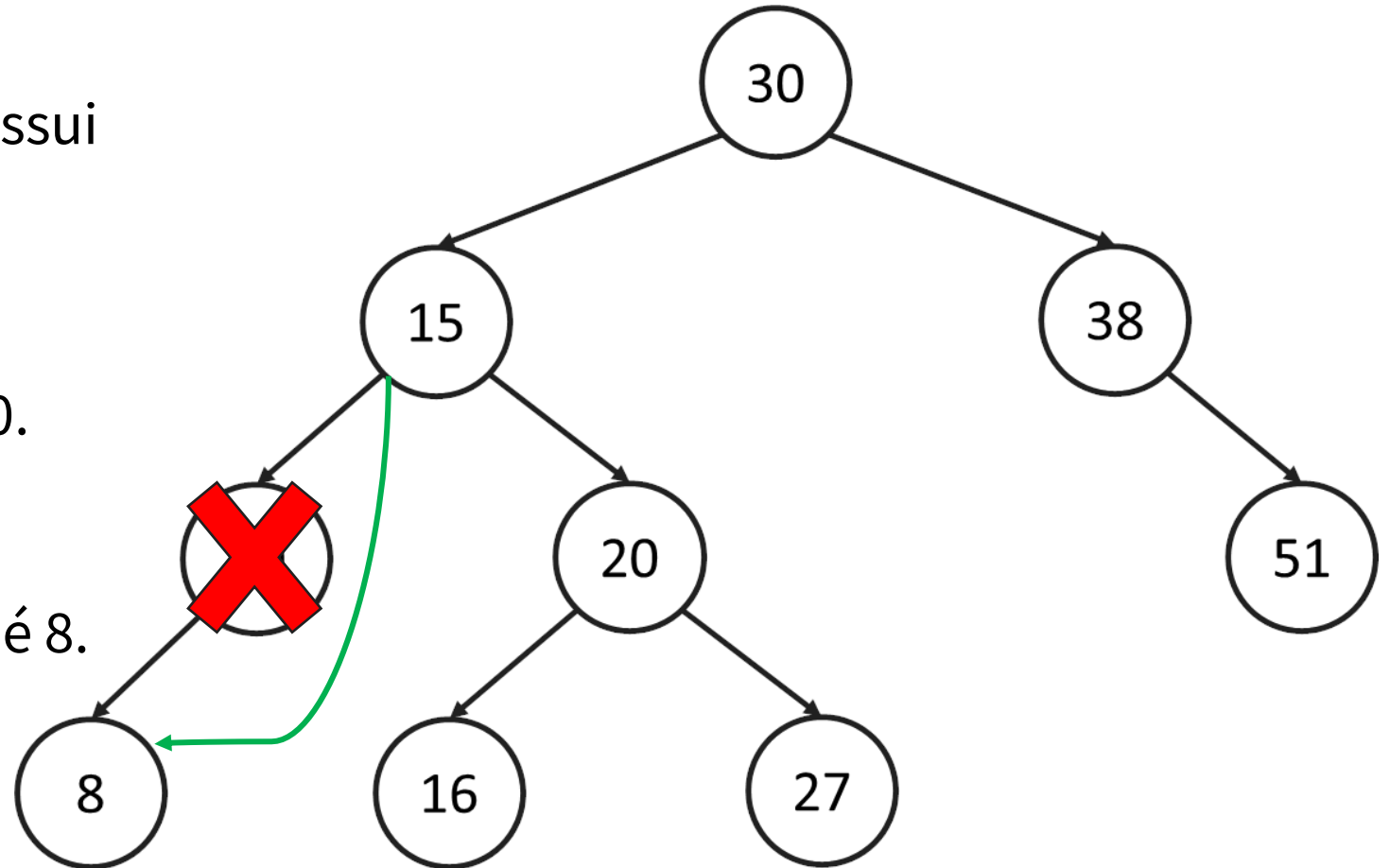


BST – Remoção

Caso 3:
O nó a ser removido não possui
subárvore direita.

Exemplo:
Remover o nó com chave 10.

Resultado:
Filho esquerdo de 15 agora é 8.



BST – Remoção

Caso 4:

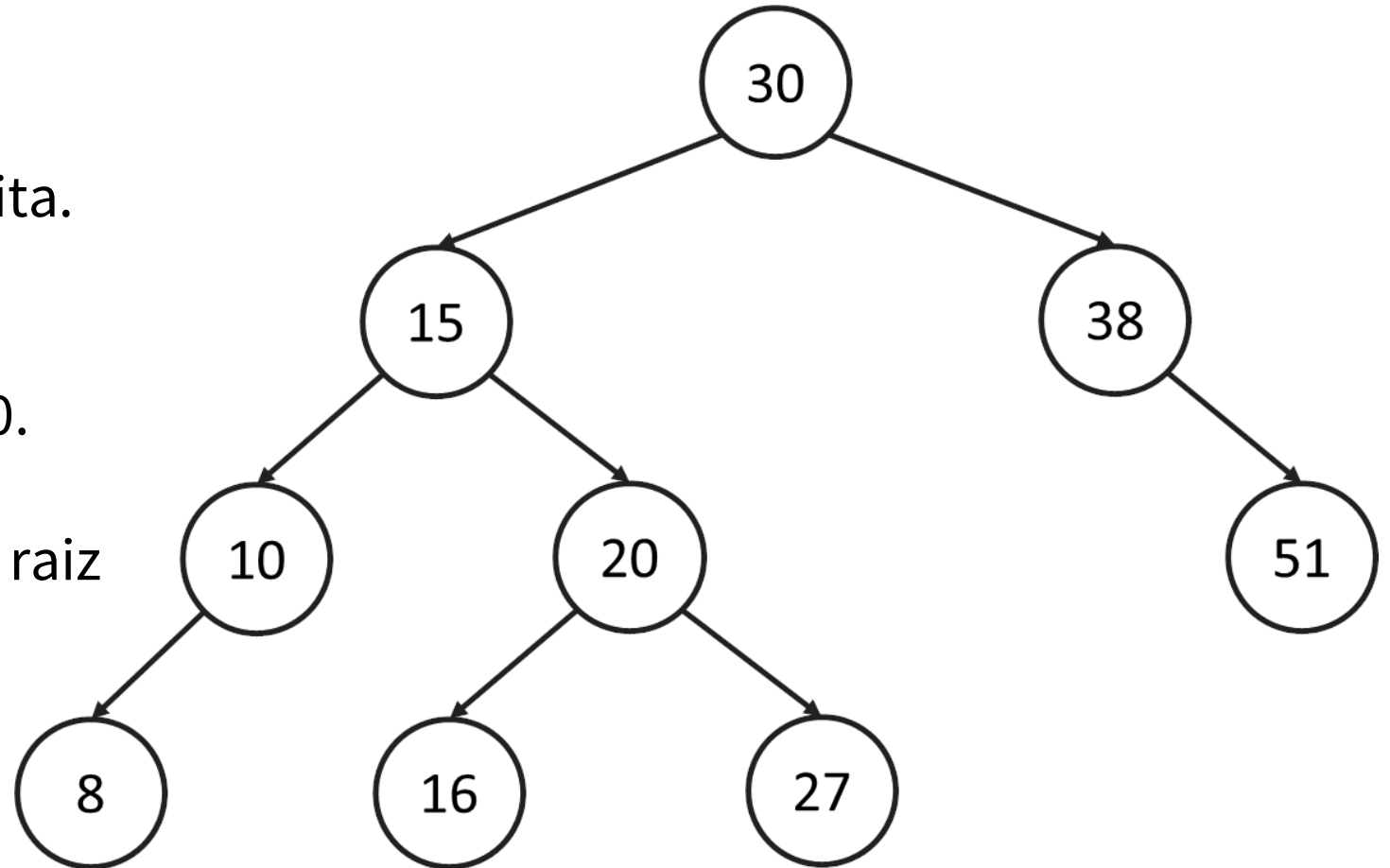
O nó a ser removido possui subárvores esquerda e direita.

Exemplo:

Remover o nó com chave 30.

O nó com chave 30 é um nó raiz de uma [sub]árvore.

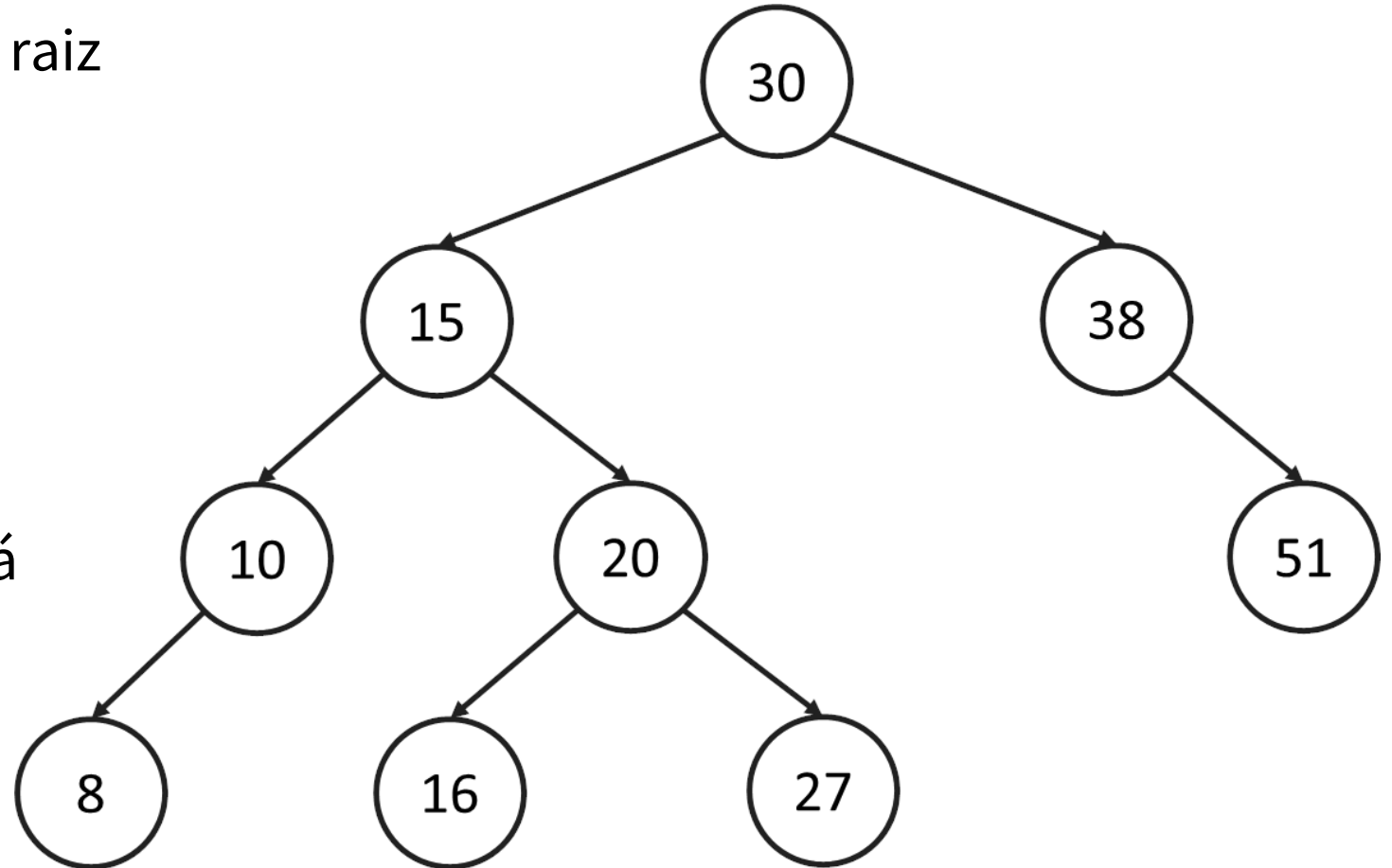
Qual nó deve virar a raiz da [sub]árvore?



BST – Remoção

O nó com chave 30 é um nó raiz de uma [sub]árvore.
Qual nó deve virar a raiz da [sub]árvore?

Podemos escolher um nó antecessor (27) ou um nó sucessor (38) ao nó que será removido.

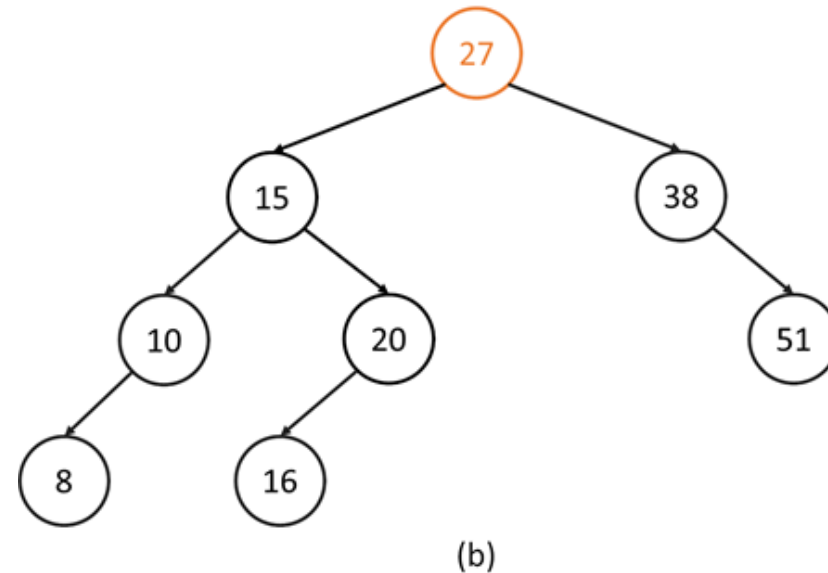
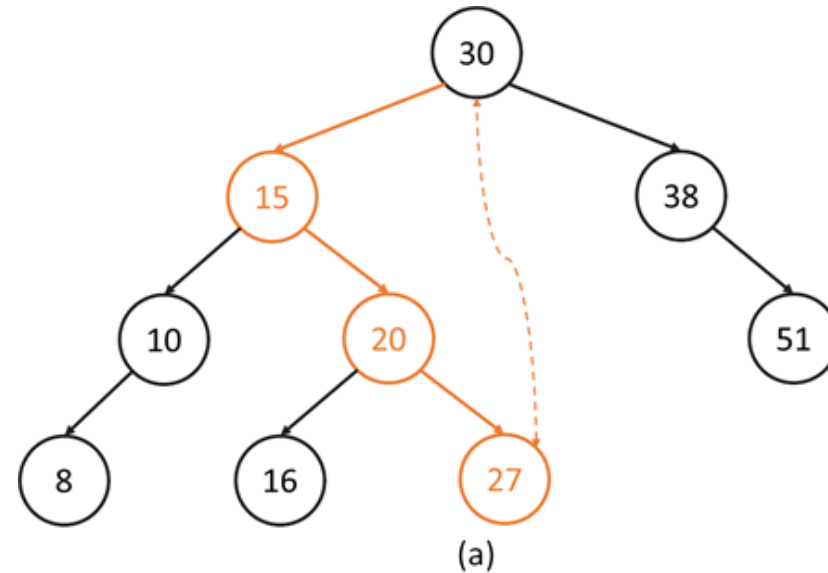


BST – Remoção

Na figura ao lado (a), escolhemos o antecessor do nó a ser removido para se tornar raiz. Nesse exemplo, o nó com chave 27.

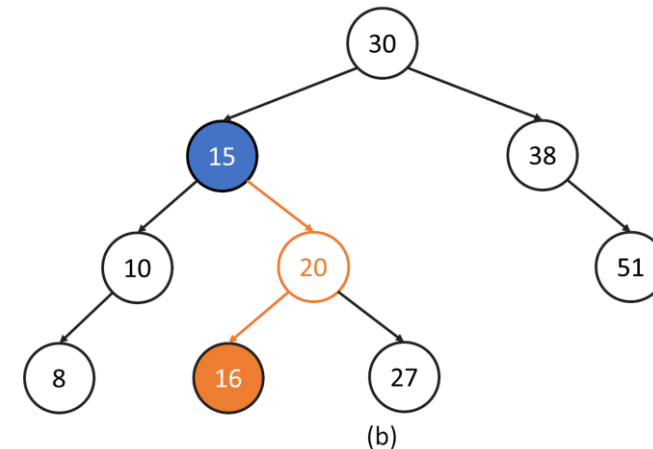
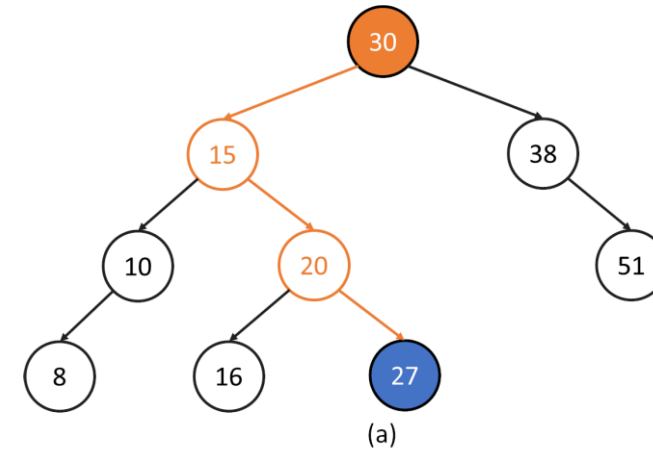
Em (b), o nó 27 saiu de sua posição original e tomou o lugar do nó 30, que foi removido.

Observe que a árvore continua sendo uma BST.



BST – Antecessor de um nó T

- (a) Se o nó T possui subárvore esquerda:
- A chave de maior valor dentre as chaves de menor valor (comparado com a chave de T).
 - Busca na subárvore esquerda de T, até encontrar um nó sem filho direito.
- (b) Se o nó T não possui uma subárvore esquerda:
- Devemos “subir na árvore” até encontrar um nó que possui valor menor que T.
- (c) Se o nó T é o menor valor da BST, então não há antecessor de T.



(laranja: nó a ser removido; azul: nó antecessor)

BST – Sucessor de um nó T

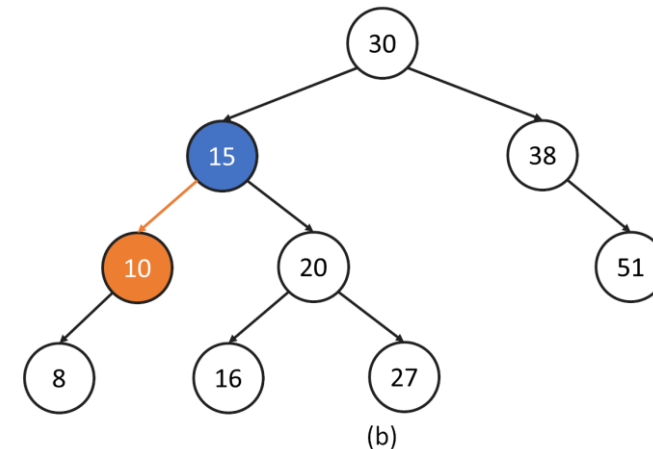
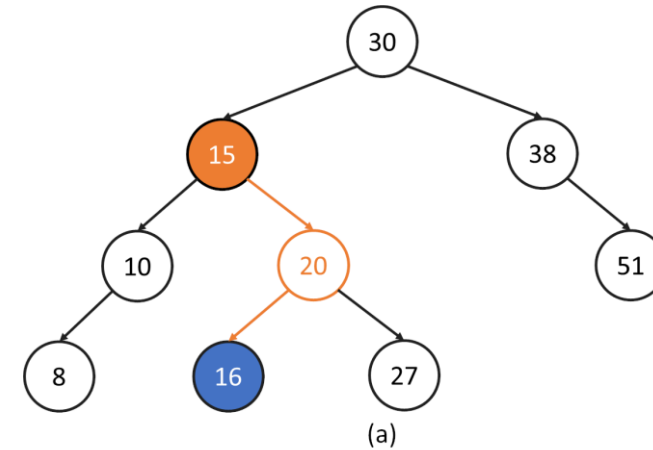
(a) Se o nó T possui uma subárvore direita:

- A chave de menor valor dentre as chaves de maior valor (comparado com a chave de T).
- Busca na subárvore direita de T, até encontrar um nó sem filho esquerdo.

(b) Se o nó T não possui uma subárvore direita:

- Devemos “subir na árvore” até encontrar um nó que possui valor maior que T.

(c) Se o nó T é o maior valor da BST, então não há sucessor de T.



(laranja: nó a ser removido; azul: nó sucessor)