

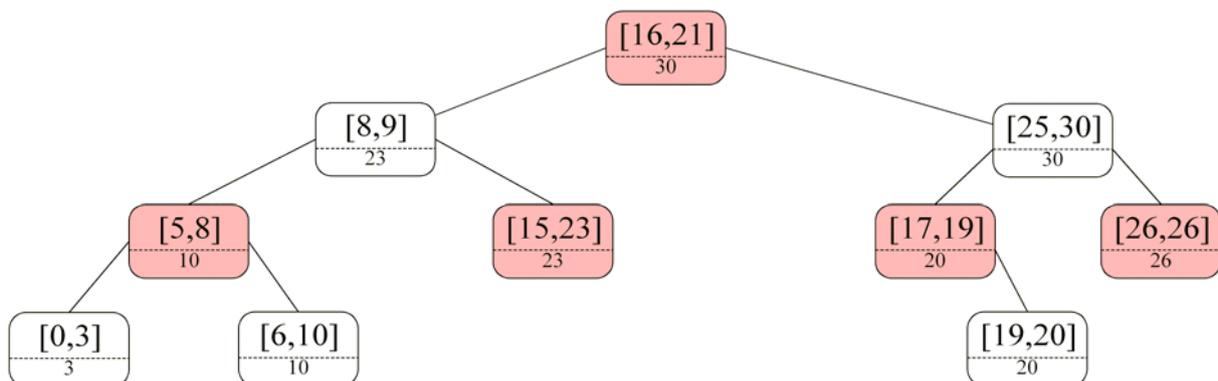
## Arbeitsblatt 5: Intervallbäume

INTERVAL-SEARCH( $T, i$ )

```
1   $x \leftarrow \text{root}[T]$ 
2  while  $x \neq \text{nil}[T]$  and  $i$  und  $\text{int}[x]$  überschneiden sich nicht
3      do if  $\text{left}[x] \neq \text{nil}[T]$  and  $\text{max}[\text{left}[x]] \geq \text{low}[i]$ 
4          then  $x \leftarrow \text{left}[x]$ 
5          else  $x \leftarrow \text{right}[x]$ 
6  return  $x$ 
```

### Aufgabe 1

Der Pseudocode INTERVAL-SEARCH( $T, i$ ) wählt in Abhängigkeit von dem Ergebnis der Abfrage in Zeile 3 für die weitere Suche nach einer Überlappung den linken oder den rechten Teilbaum des Knotens  $x$ .



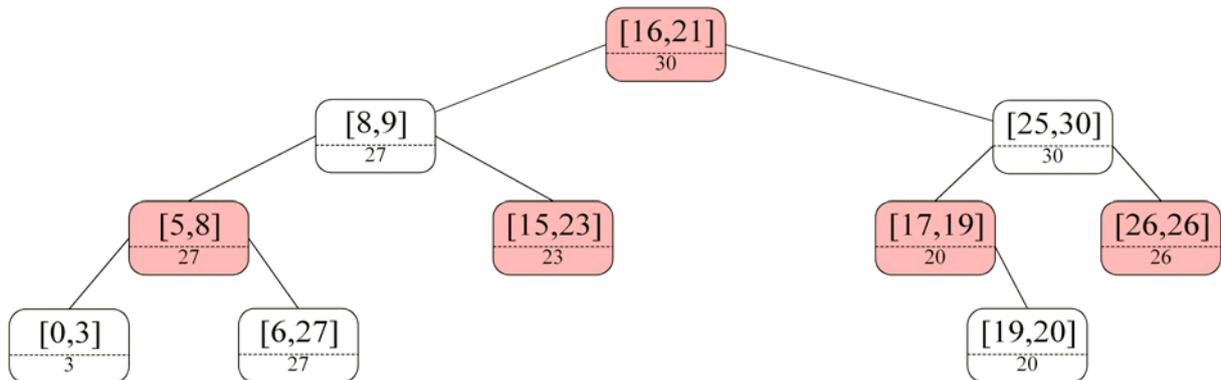
Unter welchen Bedingungen kann eine Intervallüberschneidung im linken und im rechten Teilbaum existieren? Diskutieren Sie diese Frage sowohl für die **erfolgreiche Suche**, z.B.  $i = [22, 27]$ , als auch für die **erfolglose Suche**, z.B.  $i = [11, 14]$  von INTERVAL-SEARCH.

## Aufgabe 2

Führen Sie die Operation INTERVAL-SEARCH(T, i) für den folgenden Baum T und das Intervall  $i = [25, 31]$  aus. Verwenden Sie beim Tracen den Pseudocode:

```

INTERVAL-SEARCH(T, i)
1    $x \leftarrow \text{root}[T]$ 
2   while  $x \neq \text{nil}[T]$  and  $i$  und  $\text{int}[x]$  überschneiden sich nicht
3       do if  $\text{left}[x] \neq \text{nil}[T]$  and  $\text{max}[\text{left}[x]] \geq \text{low}[i]$ 
4           then  $x \leftarrow \text{left}[x]$ 
5           else  $x \leftarrow \text{right}[x]$ 
6   return  $x$ 
    
```



Notieren jeden Schritt des Algorithmus in der folgenden Tabelle.

Nr	x	$[25, 31] \cap \text{int}[x]$	left[x]	right[x]	$\text{max}[\text{left}[x]] \geq \text{low}[i]$ ?
1	16	$[25, 31] \cap [16, 21] = \emptyset$	8	25	$27 \geq 25 \quad \checkmark$
2					
3					
4					
5					
6					