



[SWE2015-41] Introduction to Data Structures (자료구조개론)

Segment Tree

Department of Computer Science and Engineering

Instructor: Hankook Lee (이한국)

What is Segment Tree?



- Segment Tree is a binary tree that supports efficient **segment operations**
 - A segment is a collection of consecutive elements in an array
 - Examples of segment operations: sum/max/min/... of all elements in a segment

5	6	1	3	8	2	7	4
---	---	---	---	---	---	---	---

← Segment →

What is Segment Tree?



- Segment Tree is a binary tree that supports efficient **segment operations**
 - A segment is a collection of consecutive elements in an array
 - Examples of segment operations: sum/max/min/... of all elements in a segment

5	6	1	3	8	2	7	4
---	---	---	---	---	---	---	---

↓ Update

5	6	9	3	8	2	7	4
---	---	---	---	---	---	---	---

← Segment →

- This tree can compute such statistics efficiently even after an element is updated

What is Segment Tree?



- Segment Tree is a binary tree that supports efficient **segment operations**
 - A segment is a collection of consecutive elements in an array
 - Examples of segment operations: sum/max/min/... of all elements in a segment

5	6	1	3	8	2	7	4
---	---	---	---	---	---	---	---

↓ Update

5	6	9	3	8	2	7	4
---	---	---	---	---	---	---	---

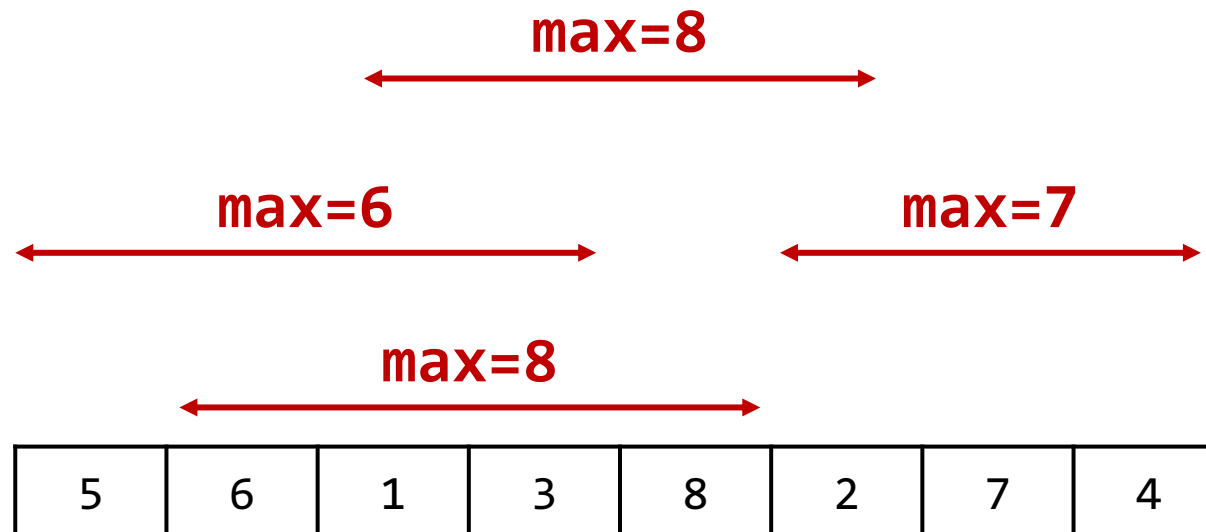
← Segment →

- This tree can compute such statistics efficiently even after an element is updated
 - Each update for an element requires $O(\log N)$ time complexity
 - Each segment operation also requires $O(\log N)$ time complexity

Segment Tree - Construction



- Given an array, **how to construct** its segment tree?
 - If your goal is to compute the **maximum** of elements in a segment, ...



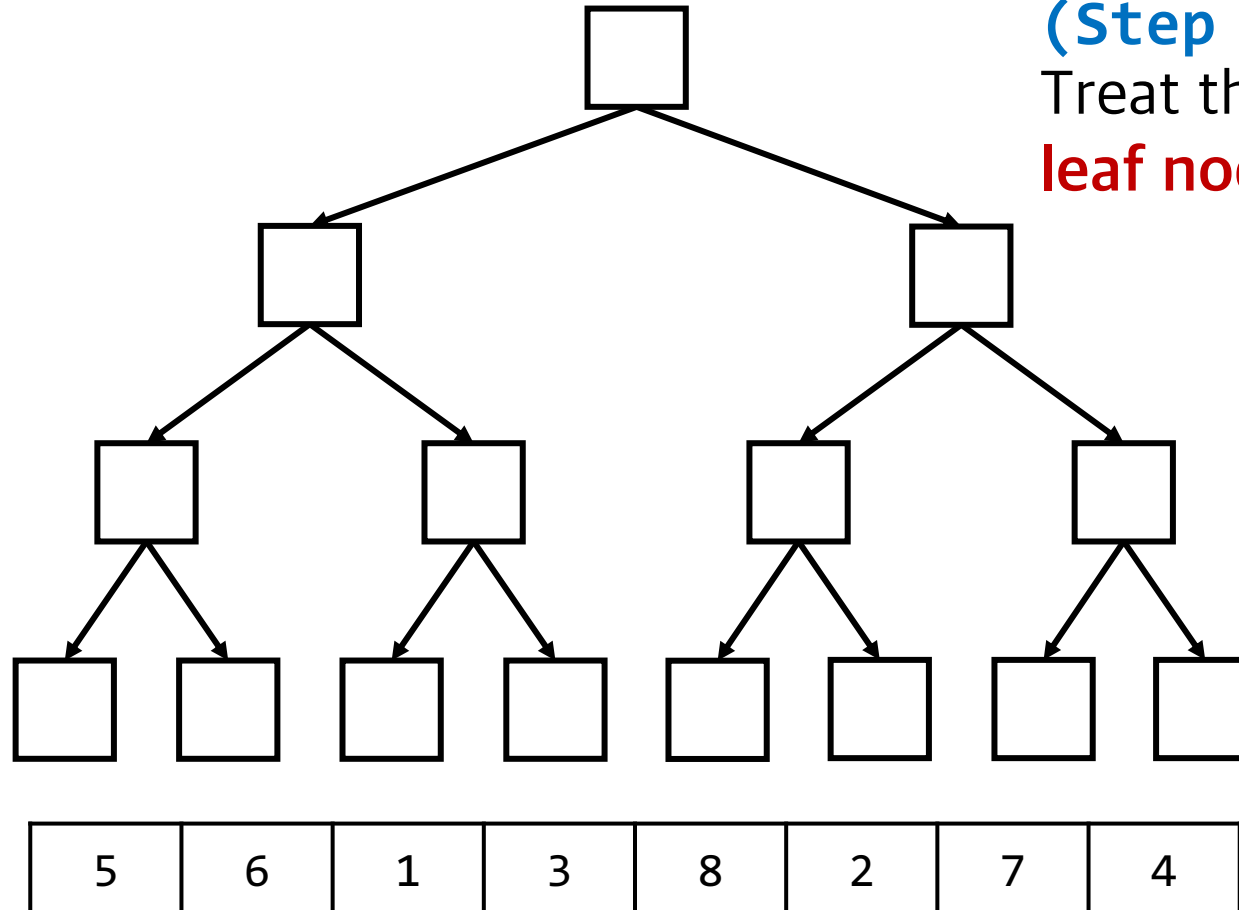
Segment Tree - Construction



- Given an array, **how to construct** its segment tree?
 - If your goal is to compute the **maximum** of elements in a segment,

(Step 1)

Treat the array elements as
leaf nodes in a full binary tree



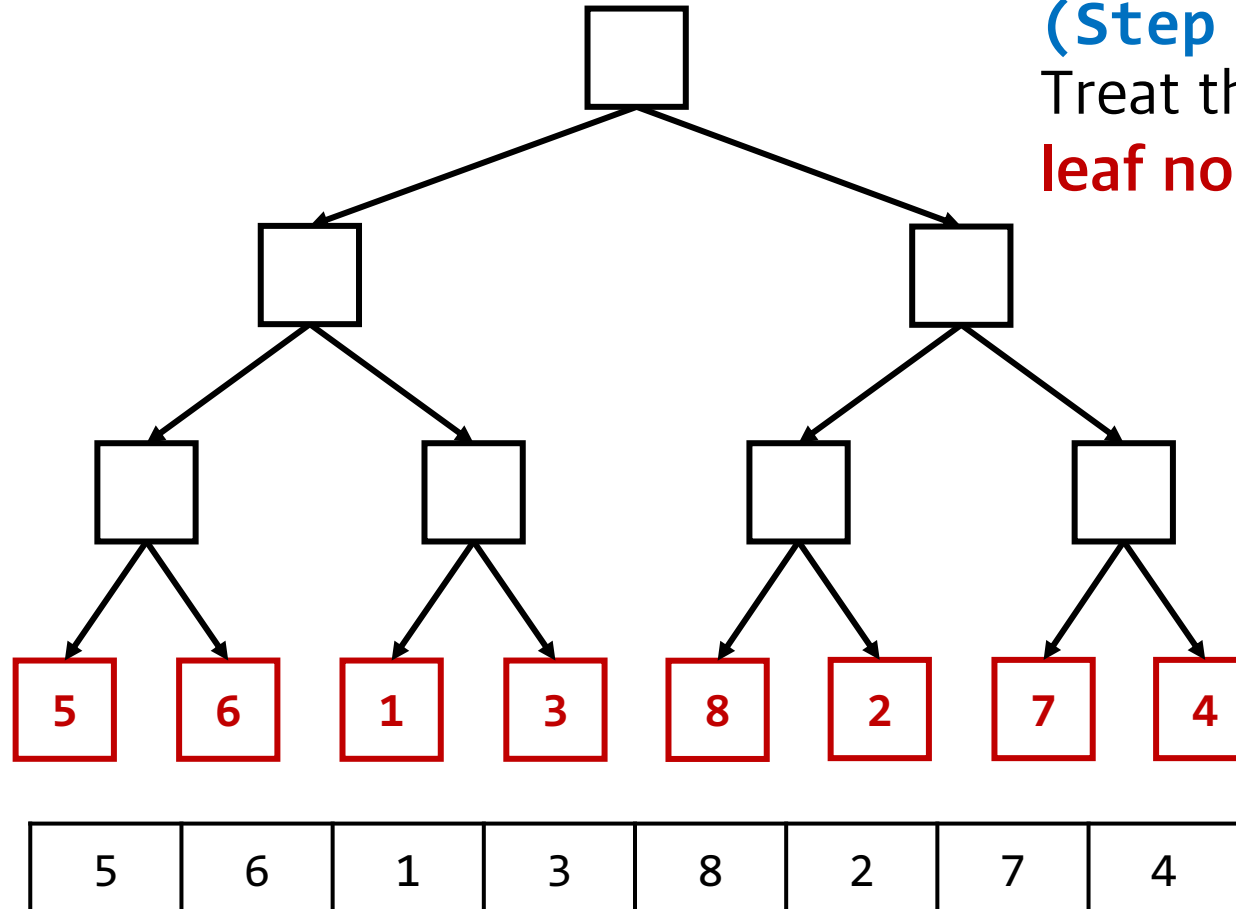
Segment Tree - Construction



- Given an array, **how to construct** its segment tree?
 - If your goal is to compute the **maximum** of elements in a segment,

(Step 1)

Treat the array elements as
leaf nodes in a full binary tree



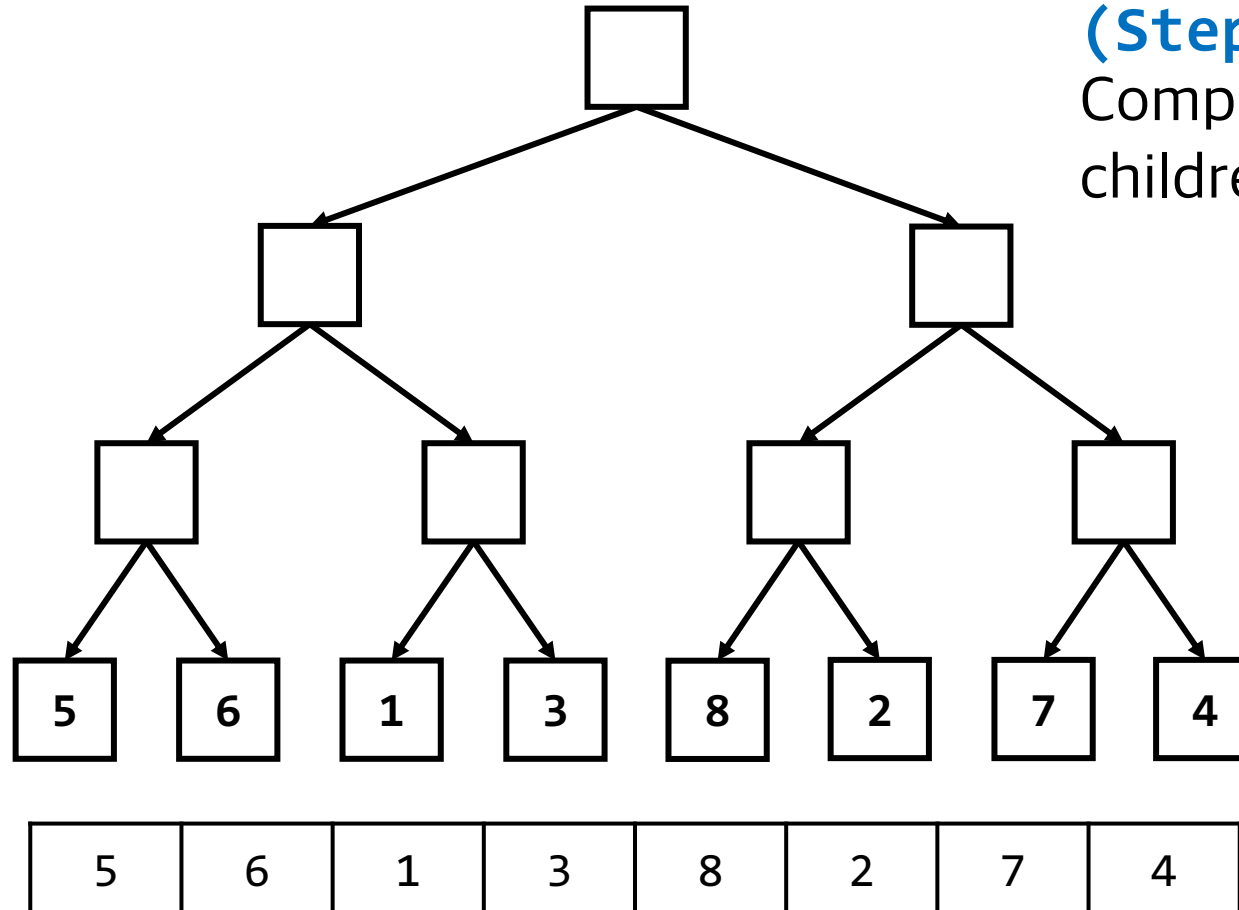
Segment Tree - Construction



- Given an array, **how to construct** its segment tree?
 - If your goal is to compute the **maximum** of elements in a segment,

(Step 2)

Compute the **maximum** of children from bottom to top



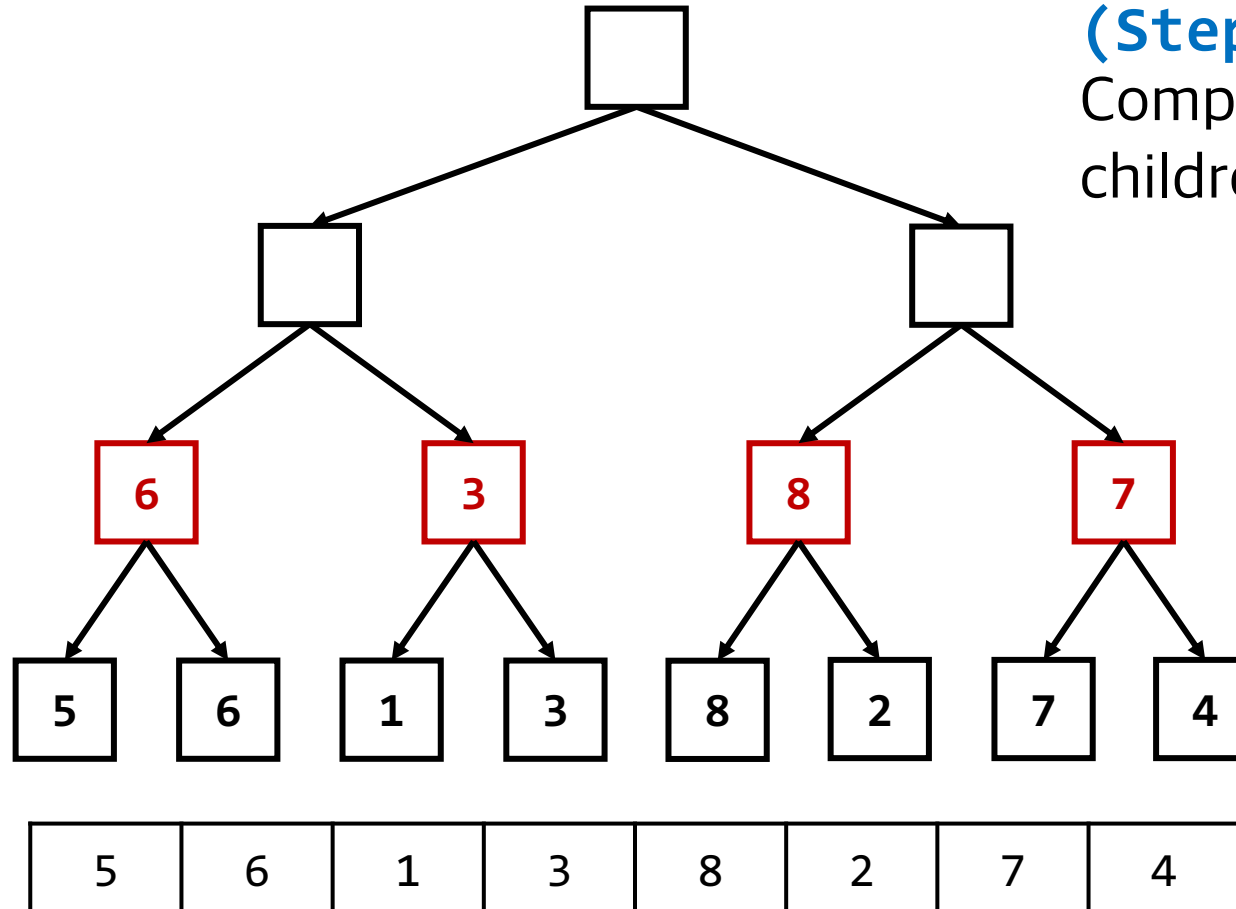
Segment Tree - Construction



- Given an array, **how to construct** its segment tree?
 - If your goal is to compute the **maximum** of elements in a segment,

(Step 2)

Compute the **maximum** of children from bottom to top



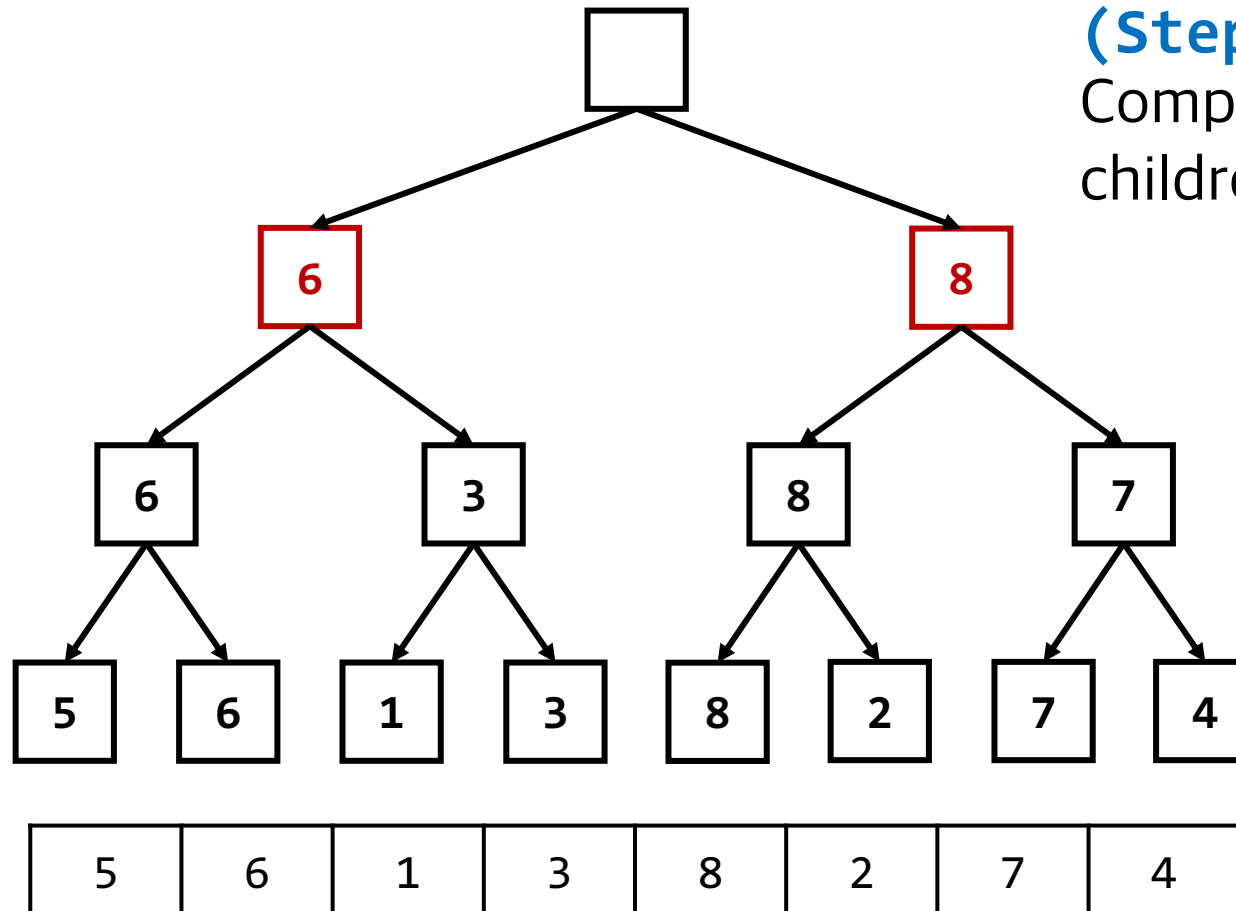
Segment Tree - Construction



- Given an array, **how to construct** its segment tree?
 - If your goal is to compute the **maximum** of elements in a segment,

(Step 2)

Compute the **maximum** of children from bottom to top



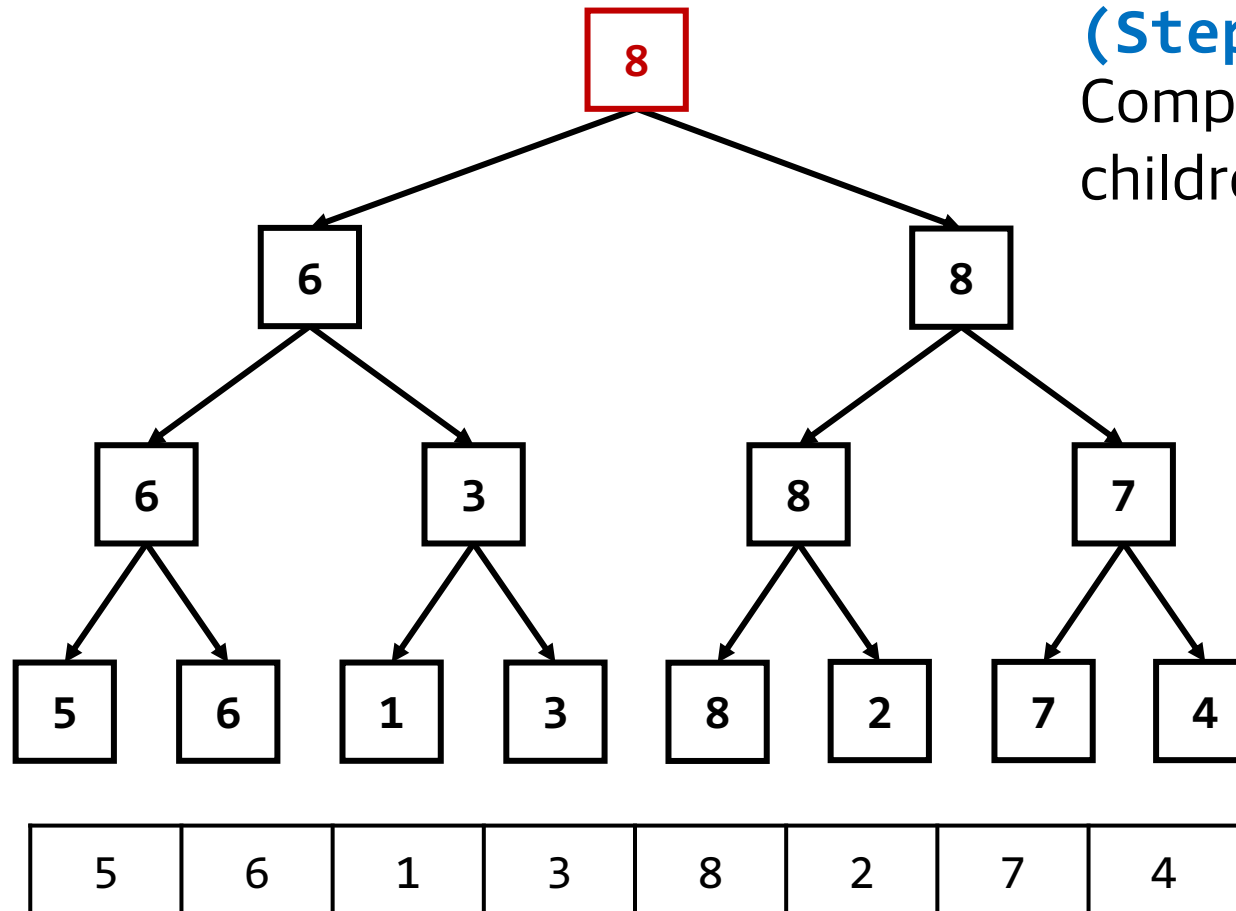
Segment Tree - Construction



- Given an array, **how to construct** its segment tree?
 - If your goal is to compute the **maximum** of elements in a segment,

(Step 2)

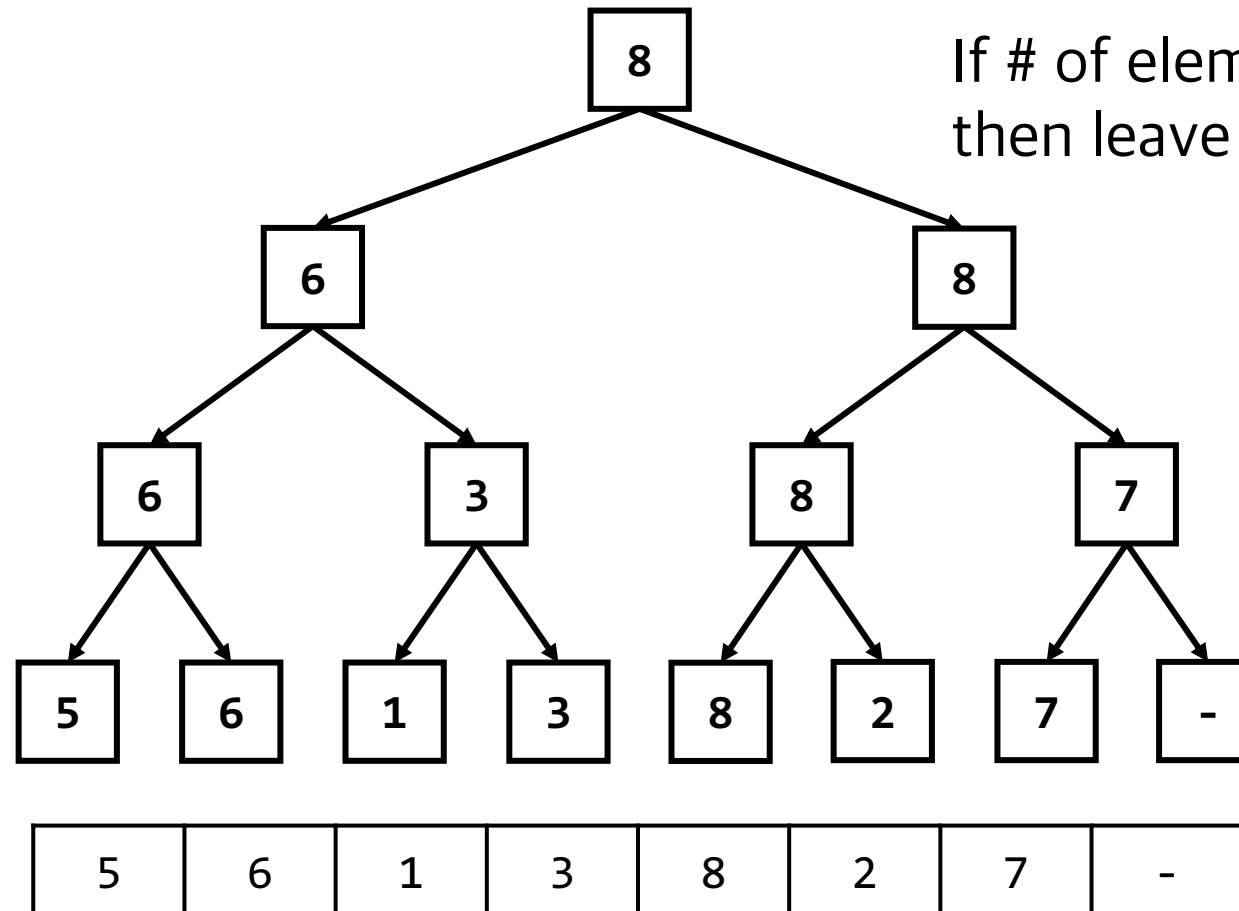
Compute the **maximum** of children from bottom to top



Segment Tree - Construction



- Given an array, **how to construct** its segment tree?
 - If your goal is to compute the **maximum** of elements in a segment,

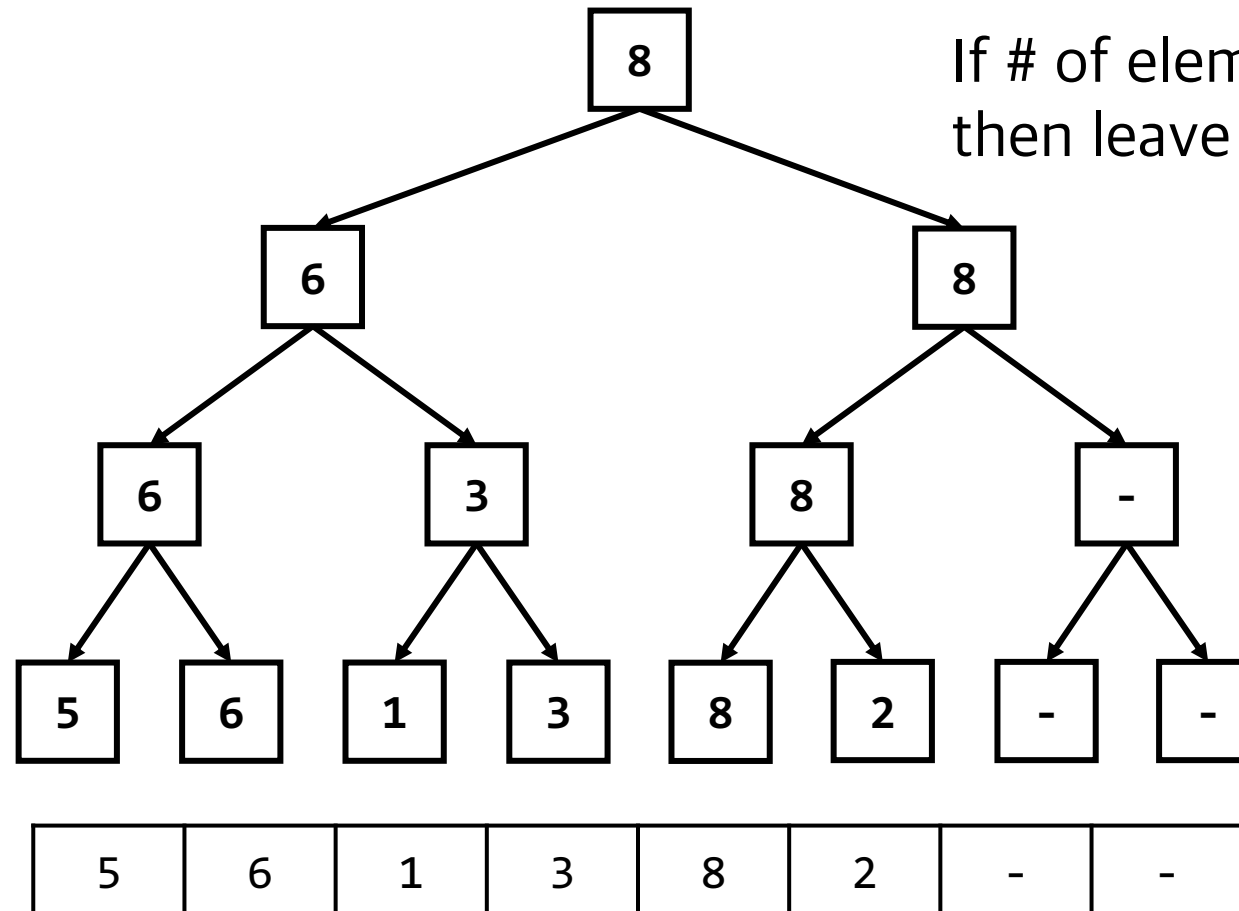


If # of elements \neq # of leaf nodes,
then leave the nodes empty

Segment Tree - Construction



- Given an array, **how to construct** its segment tree?
 - If your goal is to compute the **maximum** of elements in a segment,

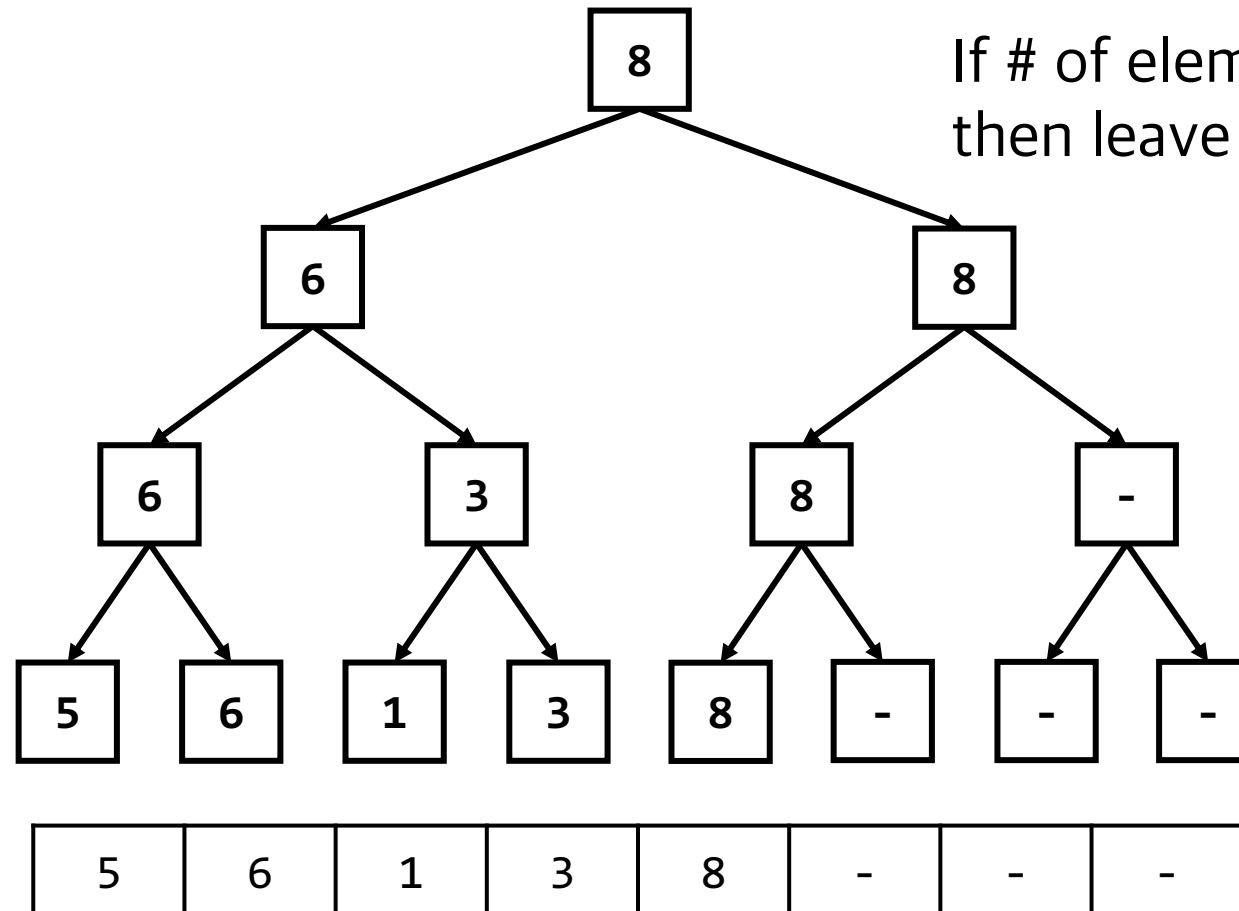


If # of elements \neq # of leaf nodes,
then leave the nodes empty

Segment Tree - Construction



- Given an array, **how to construct** its segment tree?
 - If your goal is to compute the **maximum** of elements in a segment,



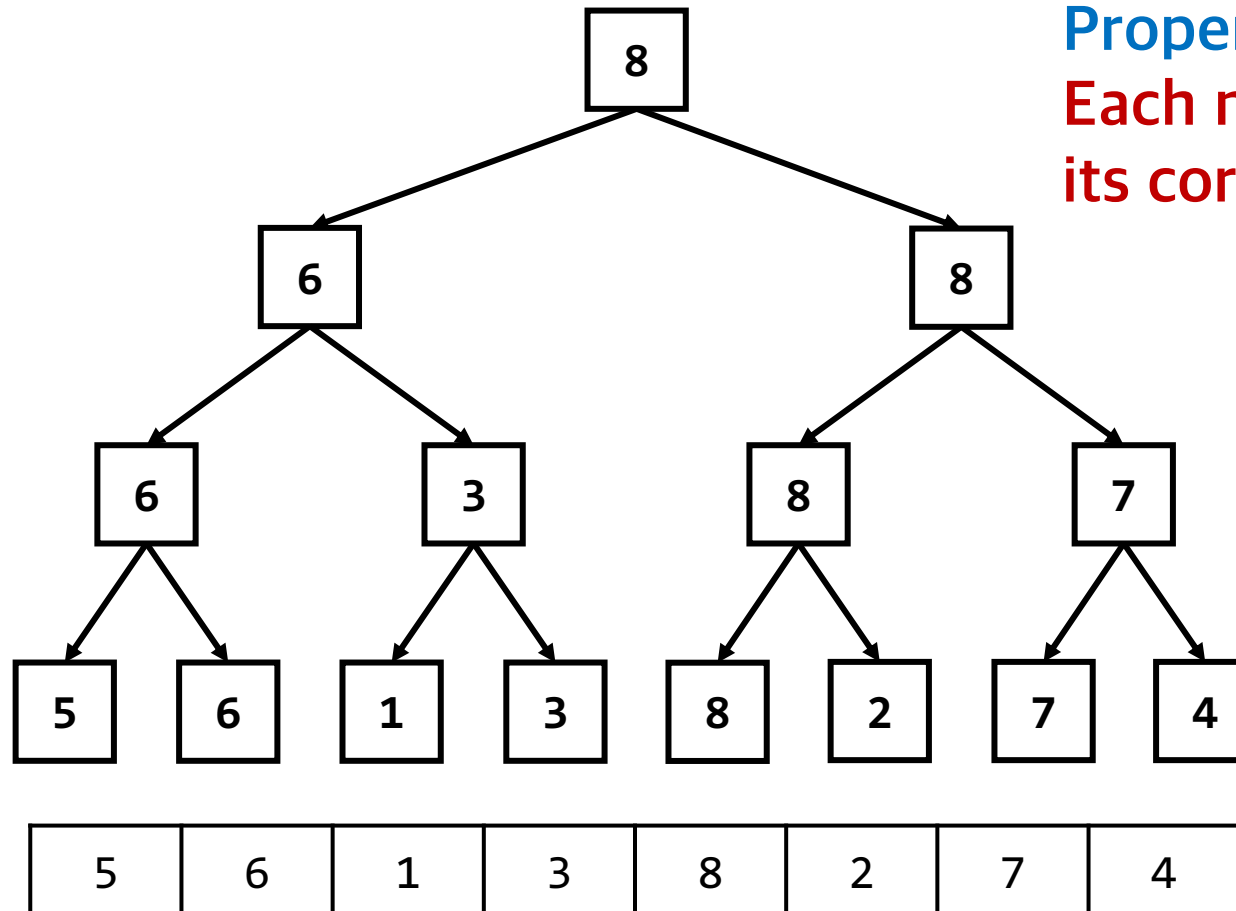
If # of elements \neq # of leaf nodes,
then leave the nodes empty

Segment Tree - Construction



- Given an array, **how to construct** its segment tree?
 - If your goal is to compute the **maximum** of elements in a segment,

Property:
Each node represents
its corresponding segment



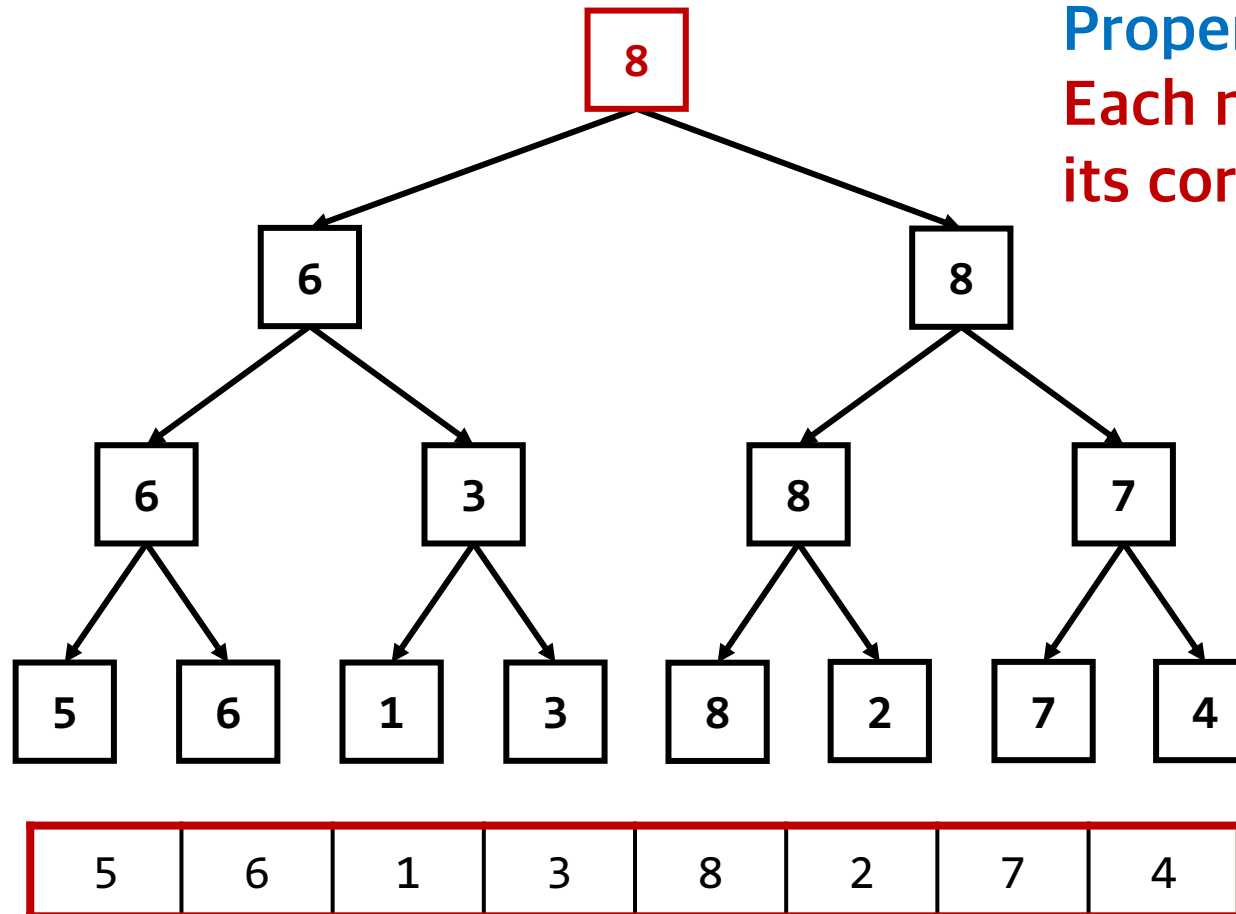
Segment Tree - Construction



- Given an array, **how to construct** its segment tree?
 - If your goal is to compute the **maximum** of elements in a segment,

Property:

Each node represents
its corresponding segment



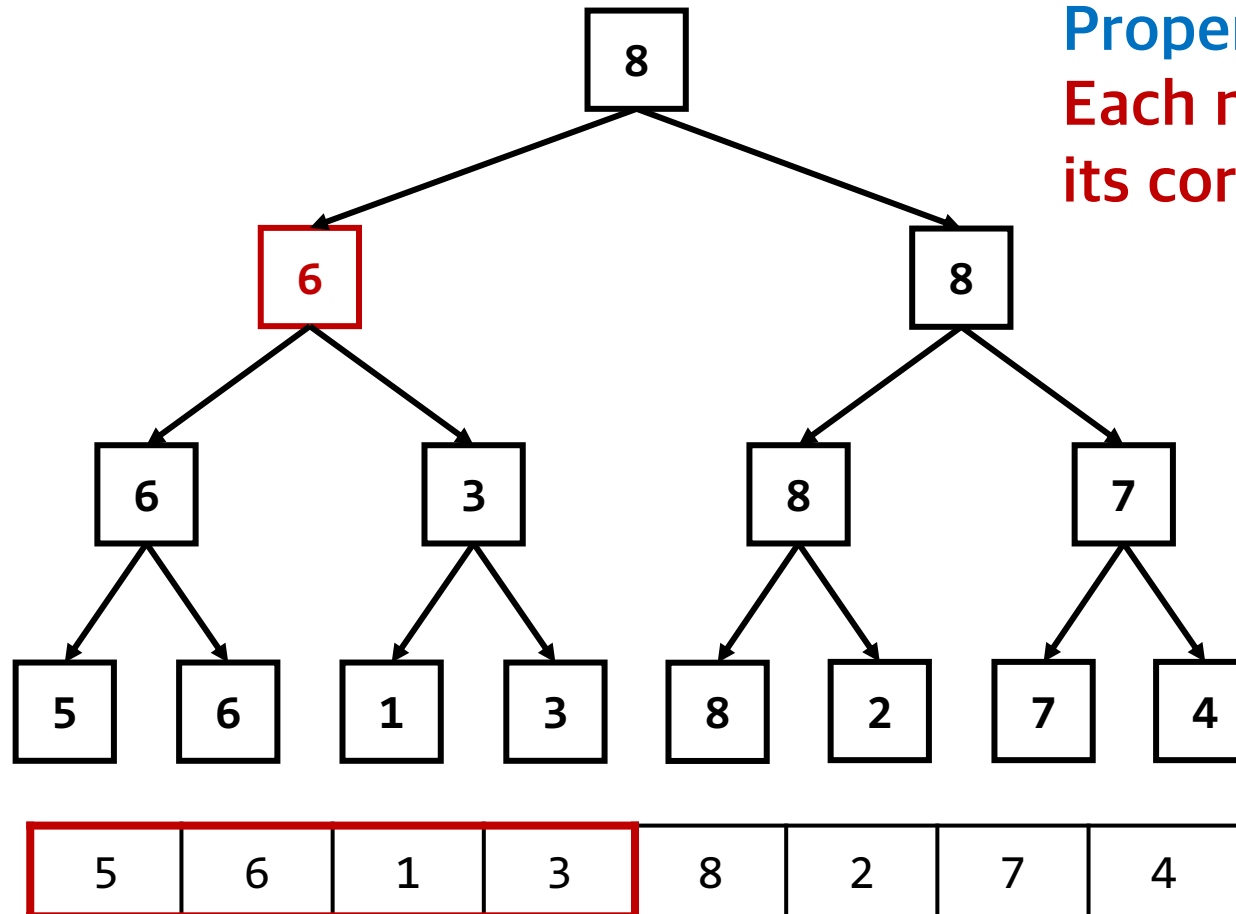
Segment Tree - Construction



- Given an array, **how to construct** its segment tree?
 - If your goal is to compute the **maximum** of elements in a segment,

Property:

Each node represents
its corresponding segment



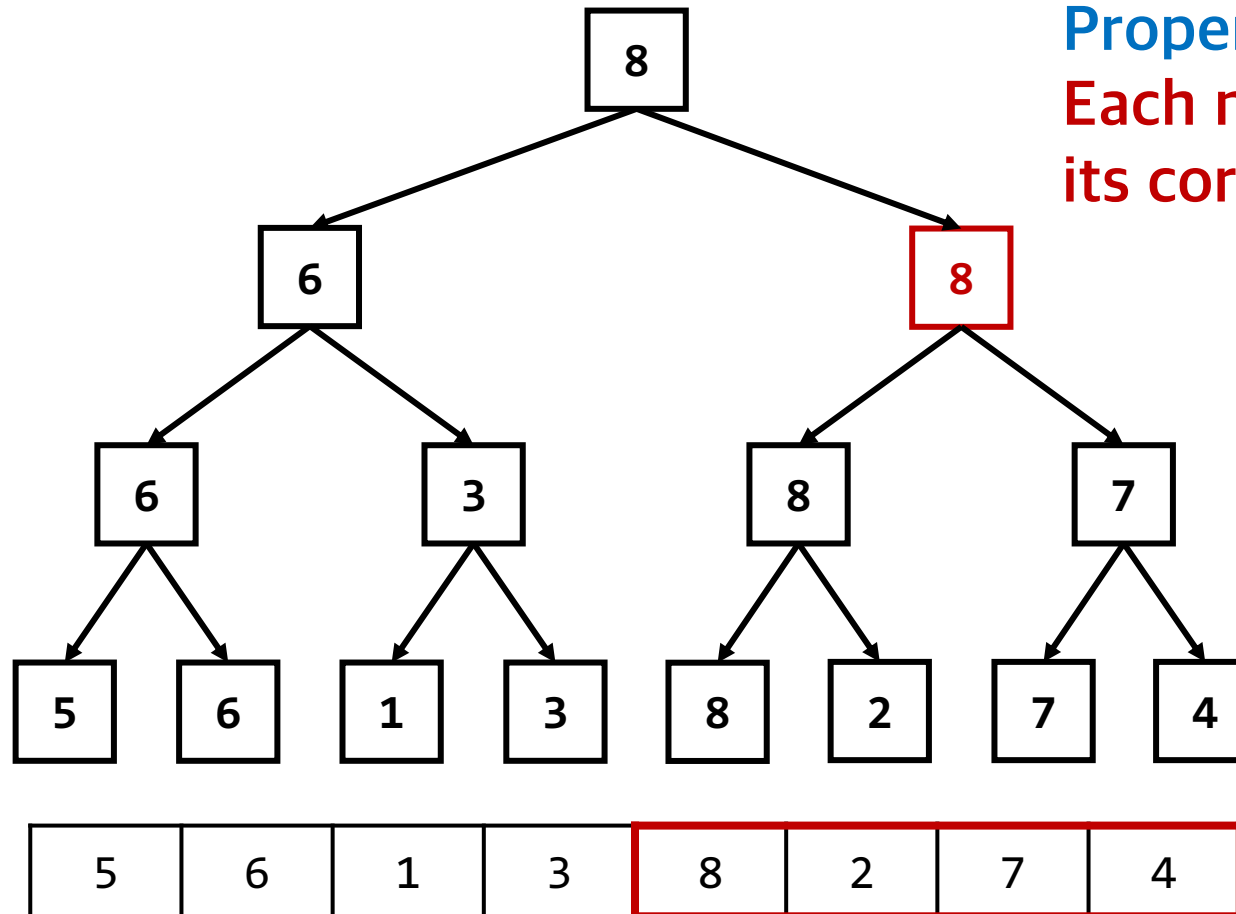
Segment Tree - Construction



- Given an array, **how to construct** its segment tree?
 - If your goal is to compute the **maximum** of elements in a segment,

Property:

Each node represents
its corresponding segment



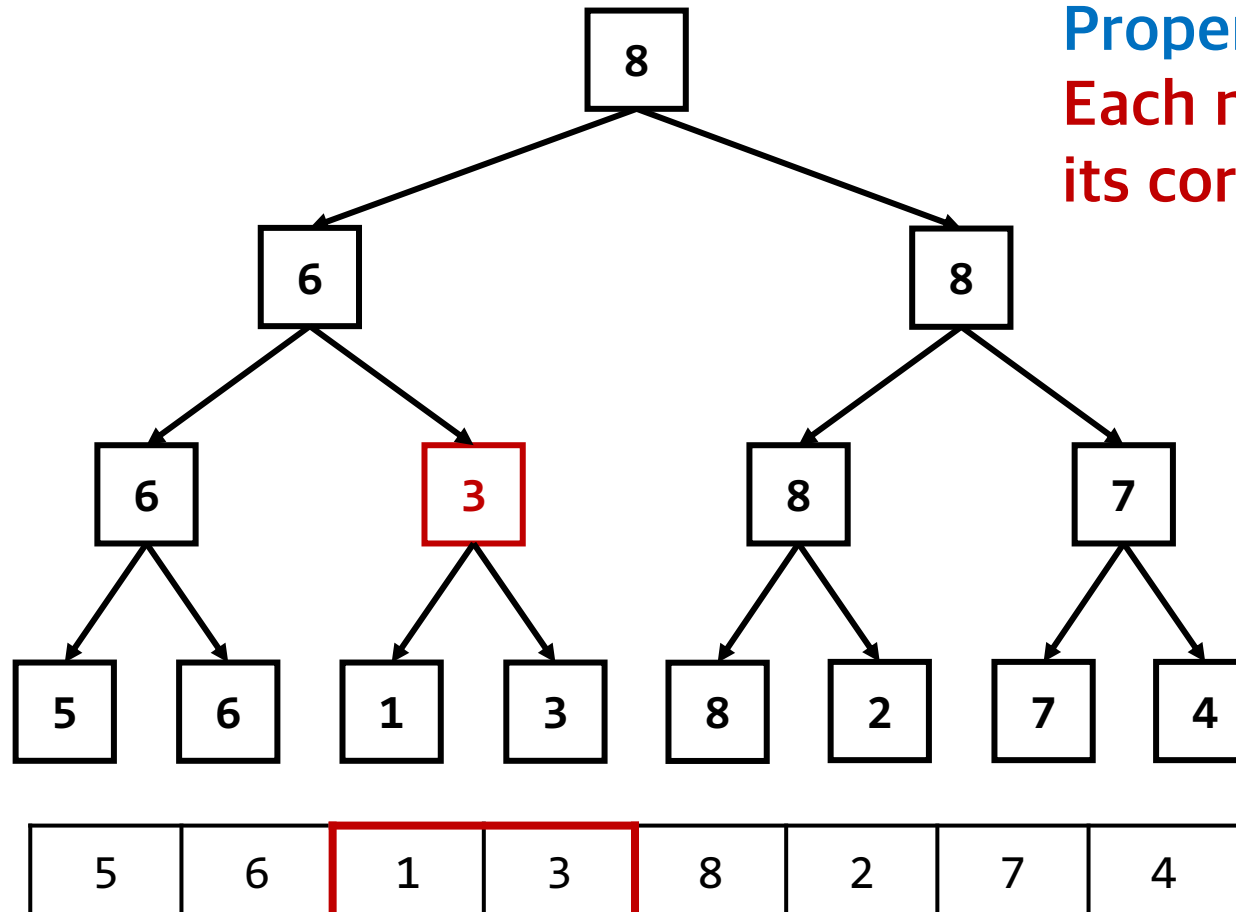
Segment Tree - Construction



- Given an array, **how to construct** its segment tree?
 - If your goal is to compute the **maximum** of elements in a segment,

Property:

Each node represents
its corresponding segment



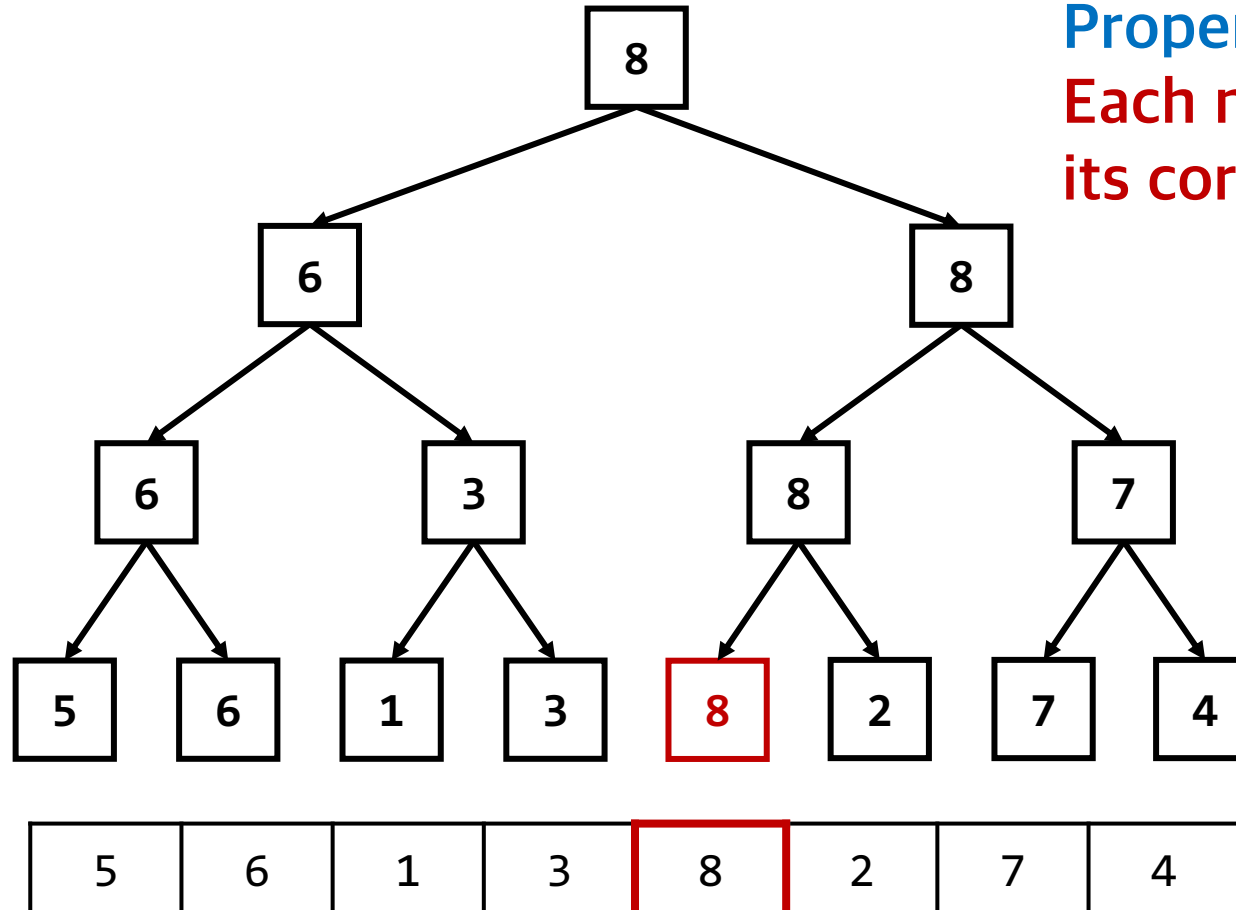
Segment Tree - Construction



- Given an array, **how to construct** its segment tree?
 - If your goal is to compute the **maximum** of elements in a segment,

Property:

Each node represents
its corresponding segment



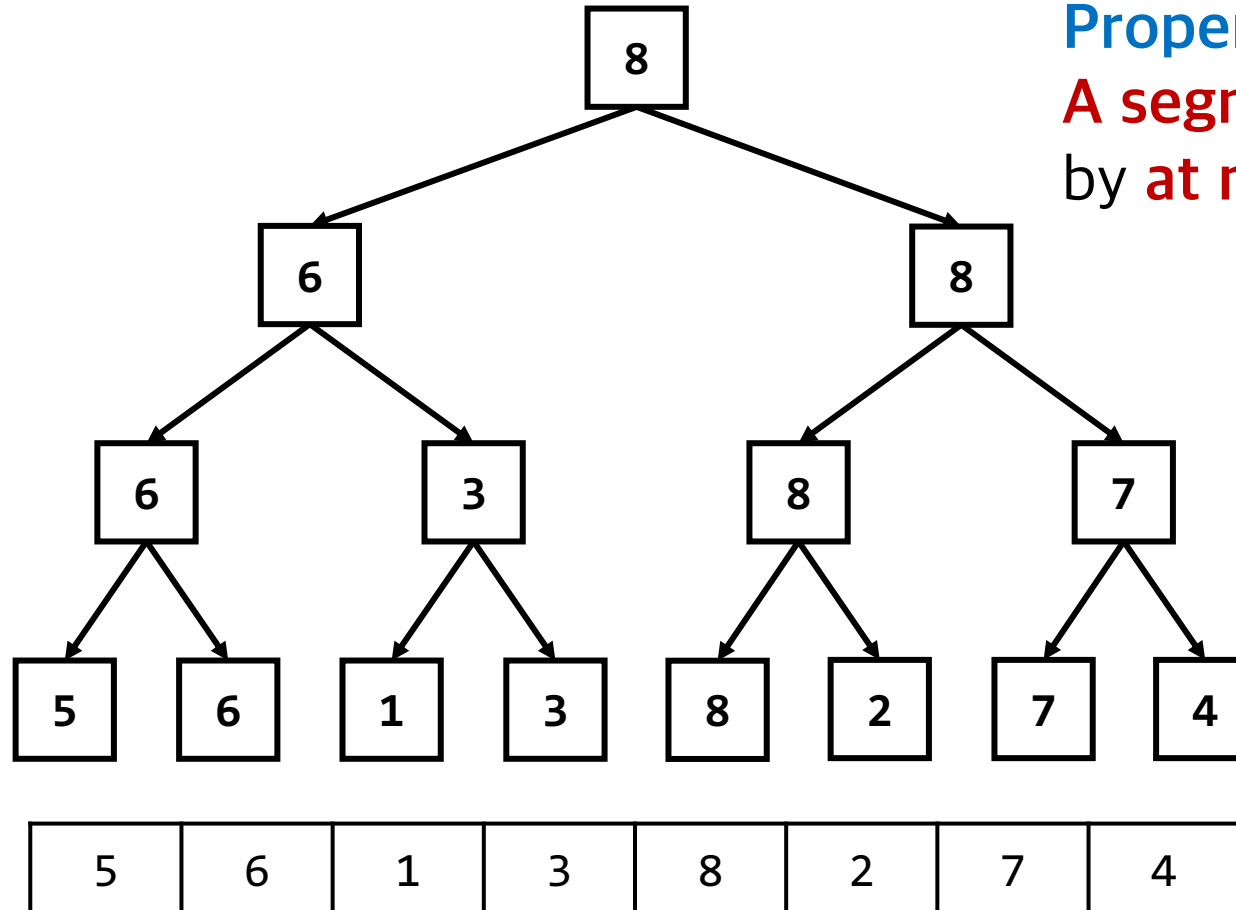
Segment Tree - Construction



- Given an array, **how to construct** its segment tree?
 - If your goal is to compute the **maximum** of elements in a segment,

Property:

A segment can be represented by **at most $O(\log N)$ nodes**



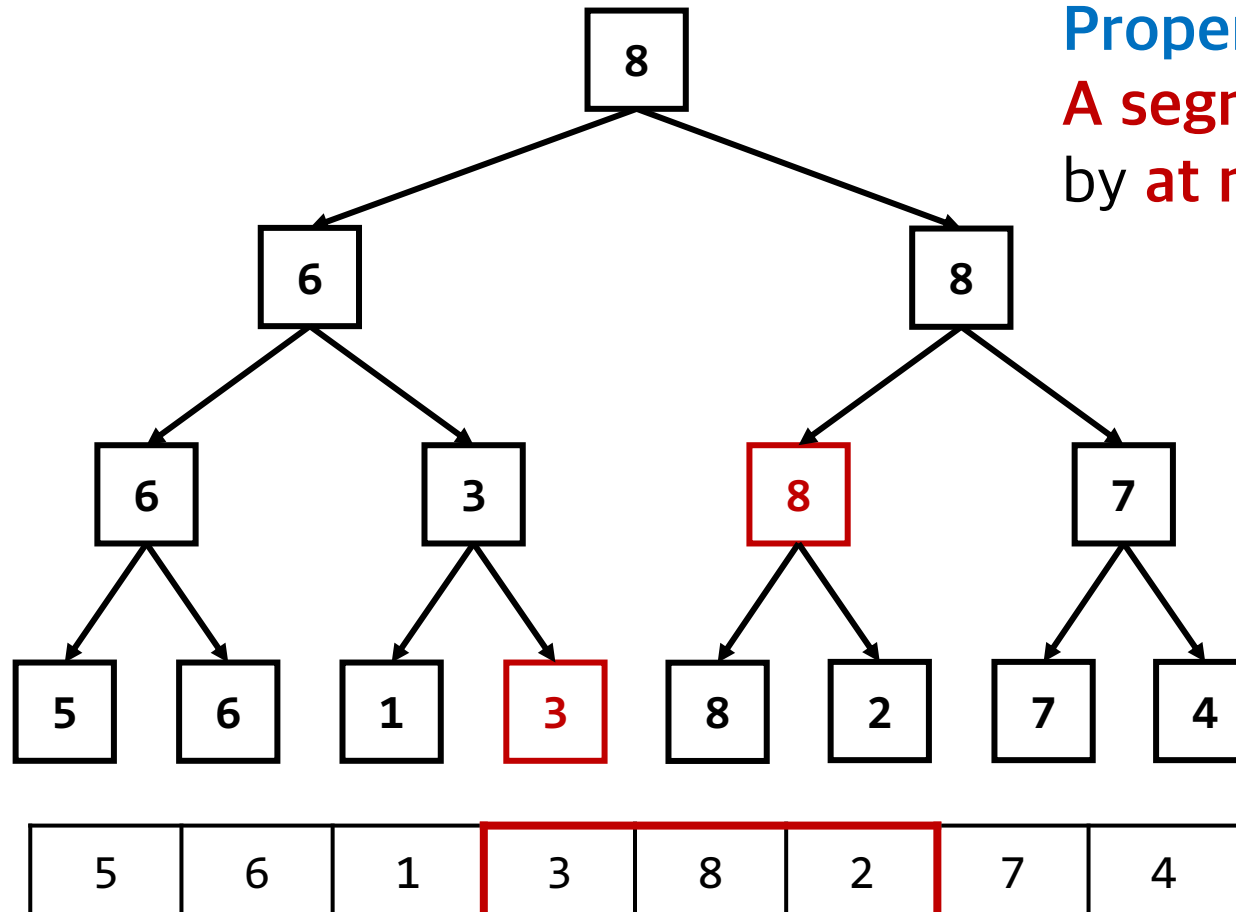
Segment Tree - Construction



- Given an array, **how to construct** its segment tree?
 - If your goal is to compute the **maximum** of elements in a segment,

Property:

A segment can be represented by **at most $O(\log N)$ nodes**



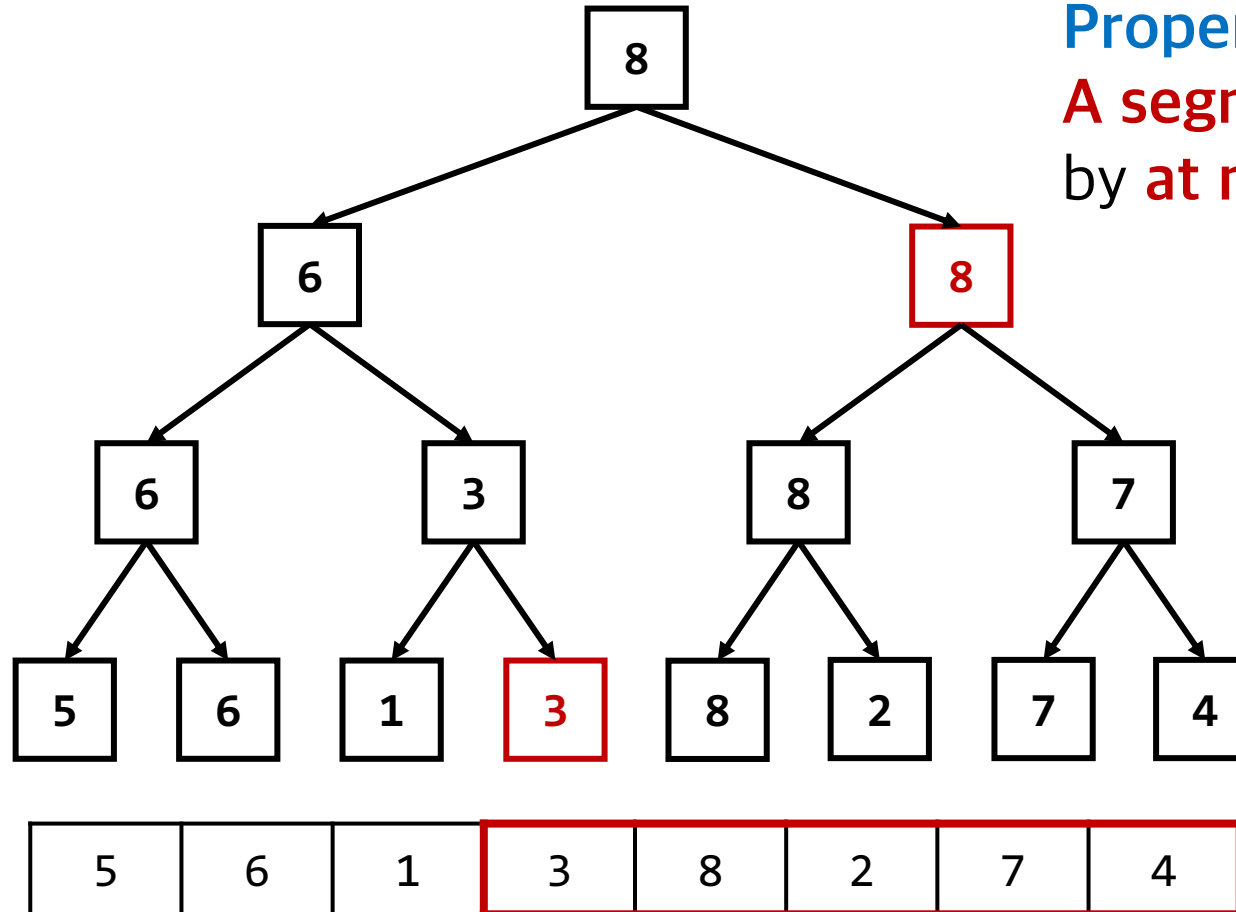
Segment Tree - Construction



- Given an array, **how to construct** its segment tree?
 - If your goal is to compute the **maximum** of elements in a segment,

Property:

A segment can be represented by **at most $O(\log N)$ nodes**



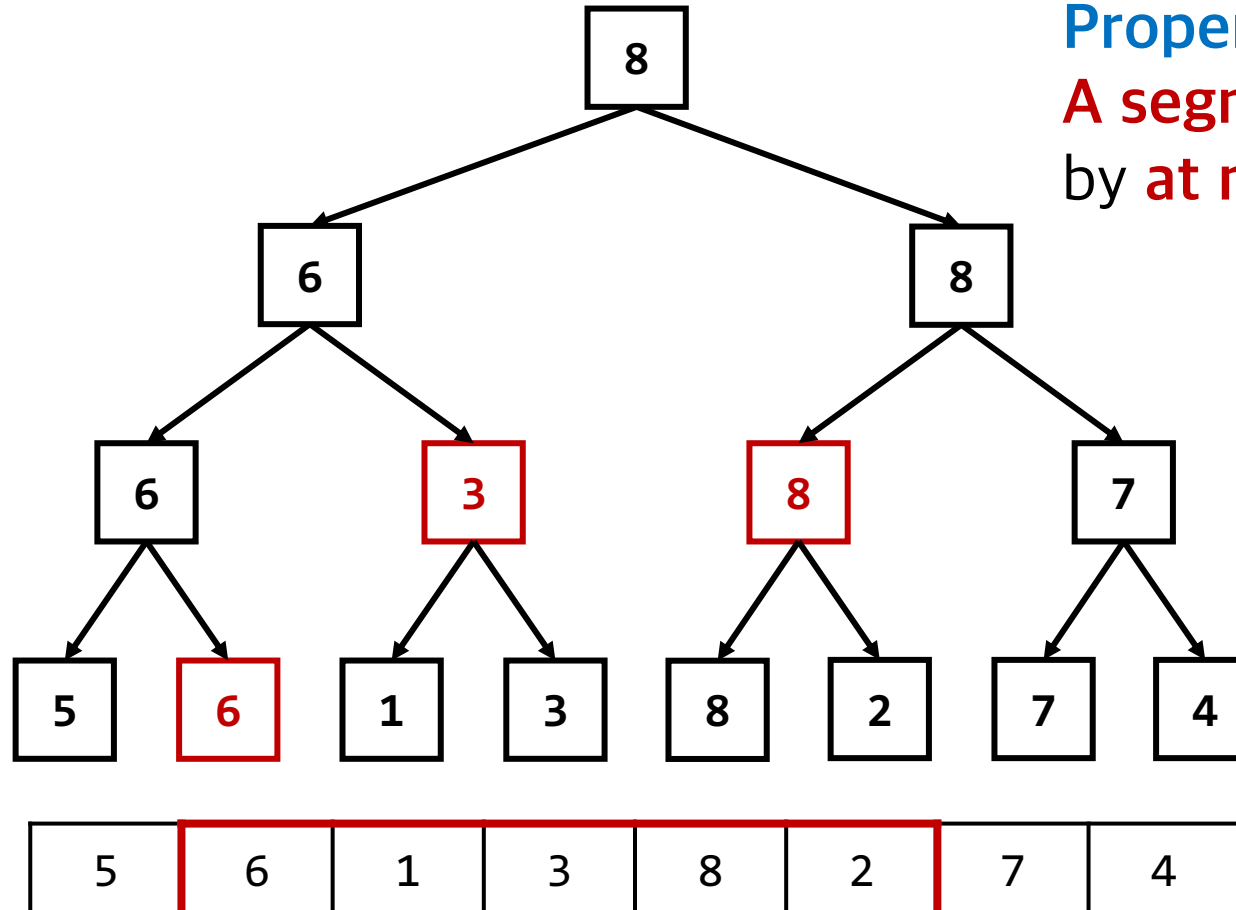
Segment Tree - Construction



- Given an array, **how to construct** its segment tree?
 - If your goal is to compute the **maximum** of elements in a segment,

Property:

A segment can be represented by **at most $O(\log N)$ nodes**



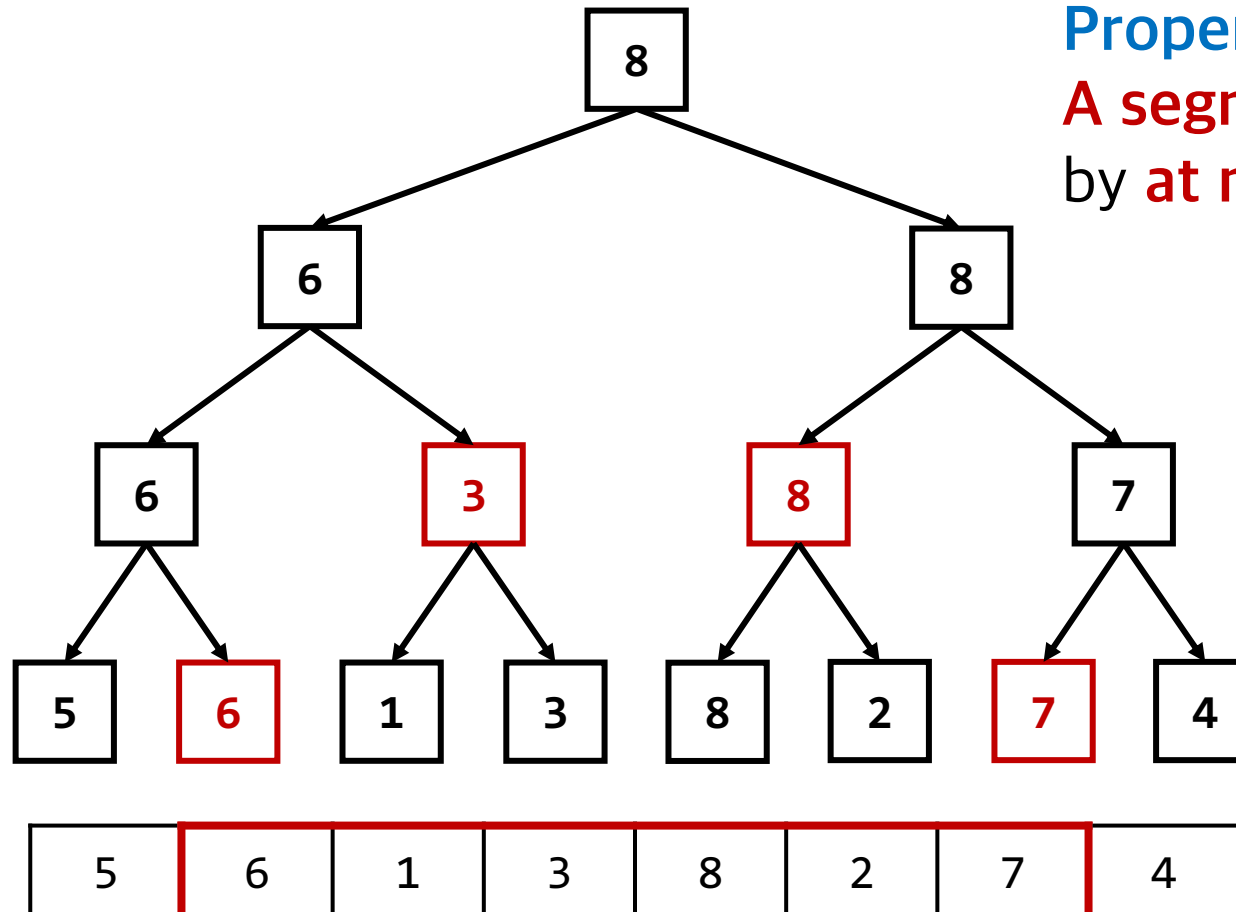
Segment Tree - Construction



- Given an array, **how to construct** its segment tree?
 - If your goal is to compute the **maximum** of elements in a segment,

Property:

A segment can be represented by **at most $O(\log N)$ nodes**



Segment Tree - Construction



- Given an array, **how to construct** its segment tree?
 - (Step 1) Treat the array elements as **leaf nodes in a full binary tree**
 - (Step 2) Compute the maximum of children from bottom to top

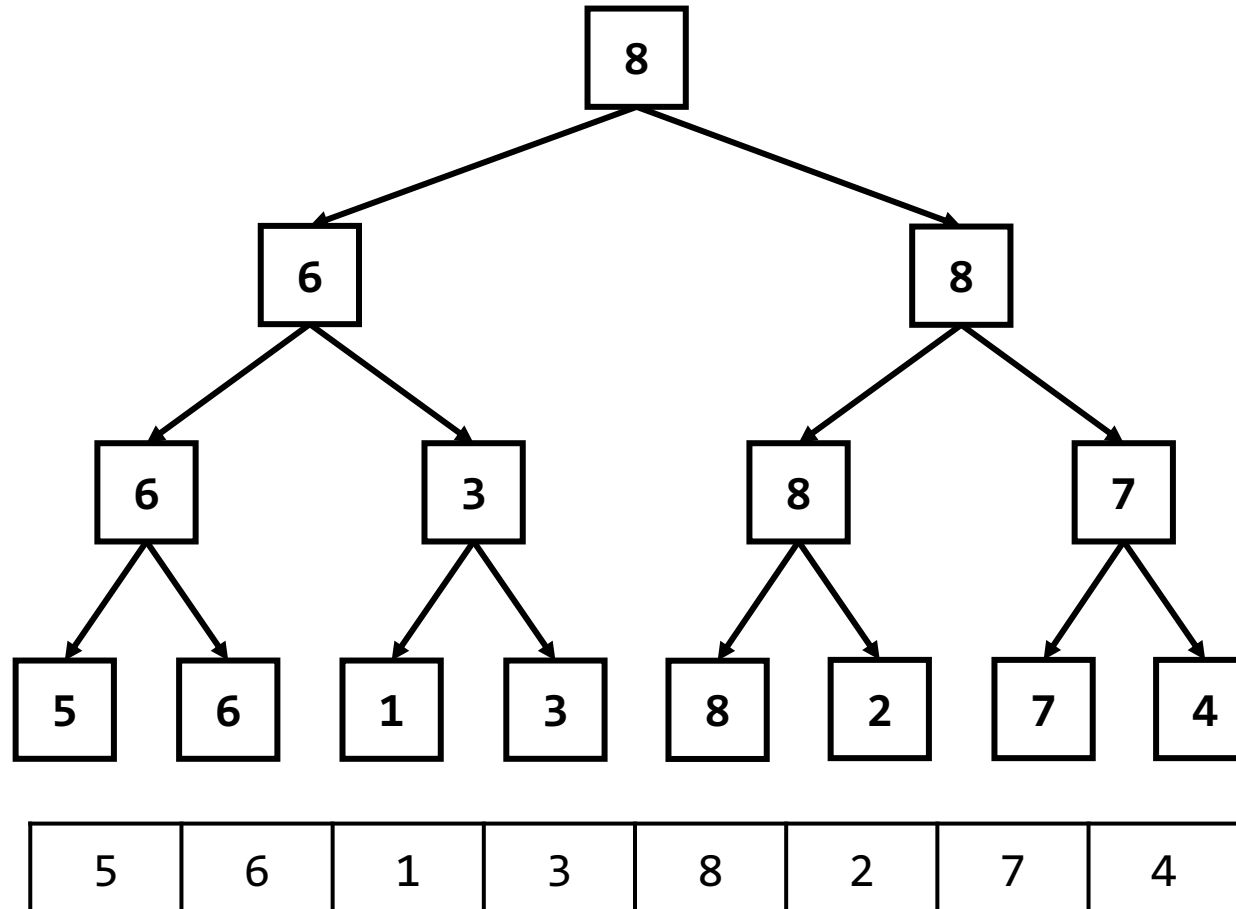
Properties: (where N is the number of array elements)

- **Any segment** can be represented by **at most $O(\log N)$ nodes**
- The maximum of elements in any segment can be computed in $O(\log N)$
 - using the representative nodes
- The construction time = $N + N/2 + N/4 + N/8 + \dots = 2N = O(N)$

Segment Tree - Update



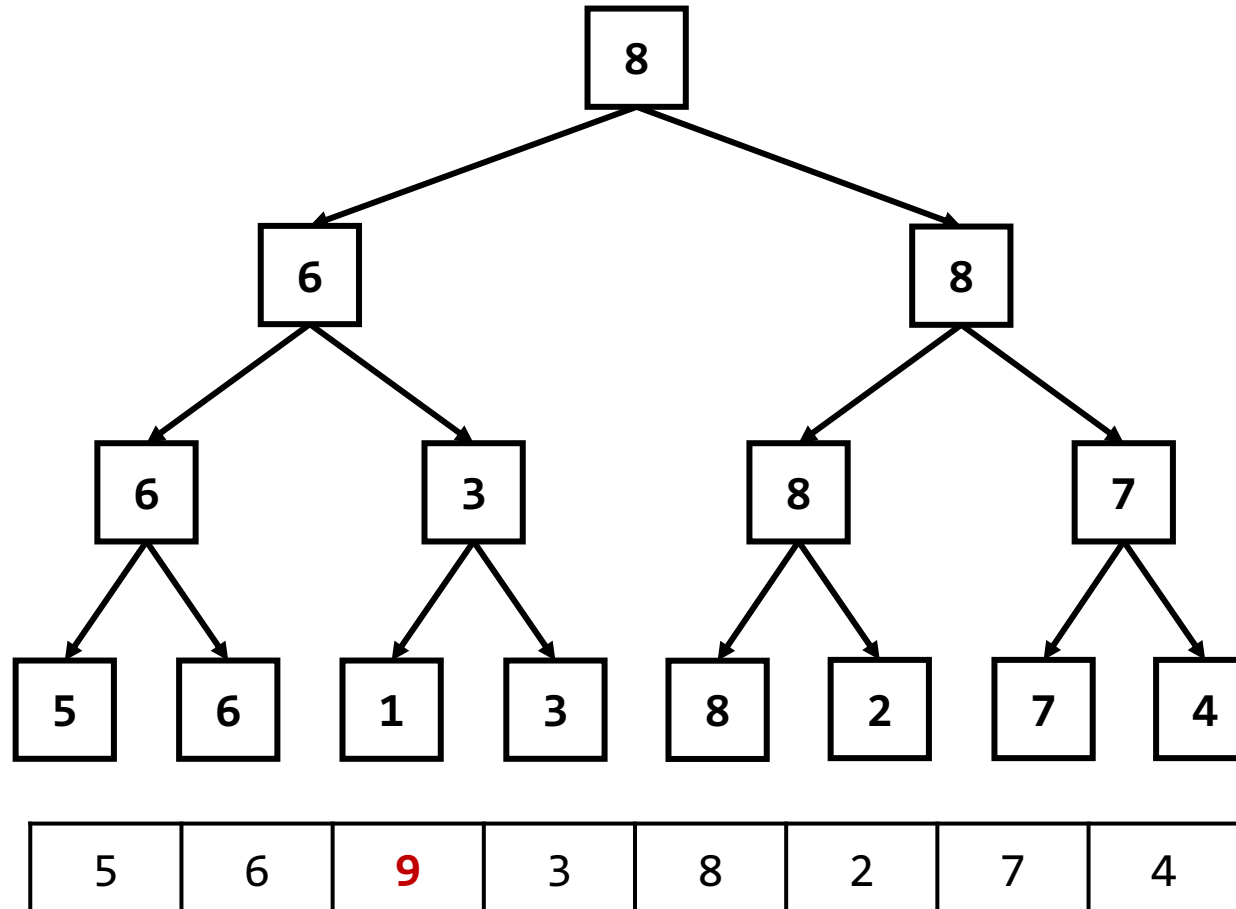
- How to update an element?
 - If the 3rd element is replaced by **9**, which nodes should be updated?



Segment Tree - Update



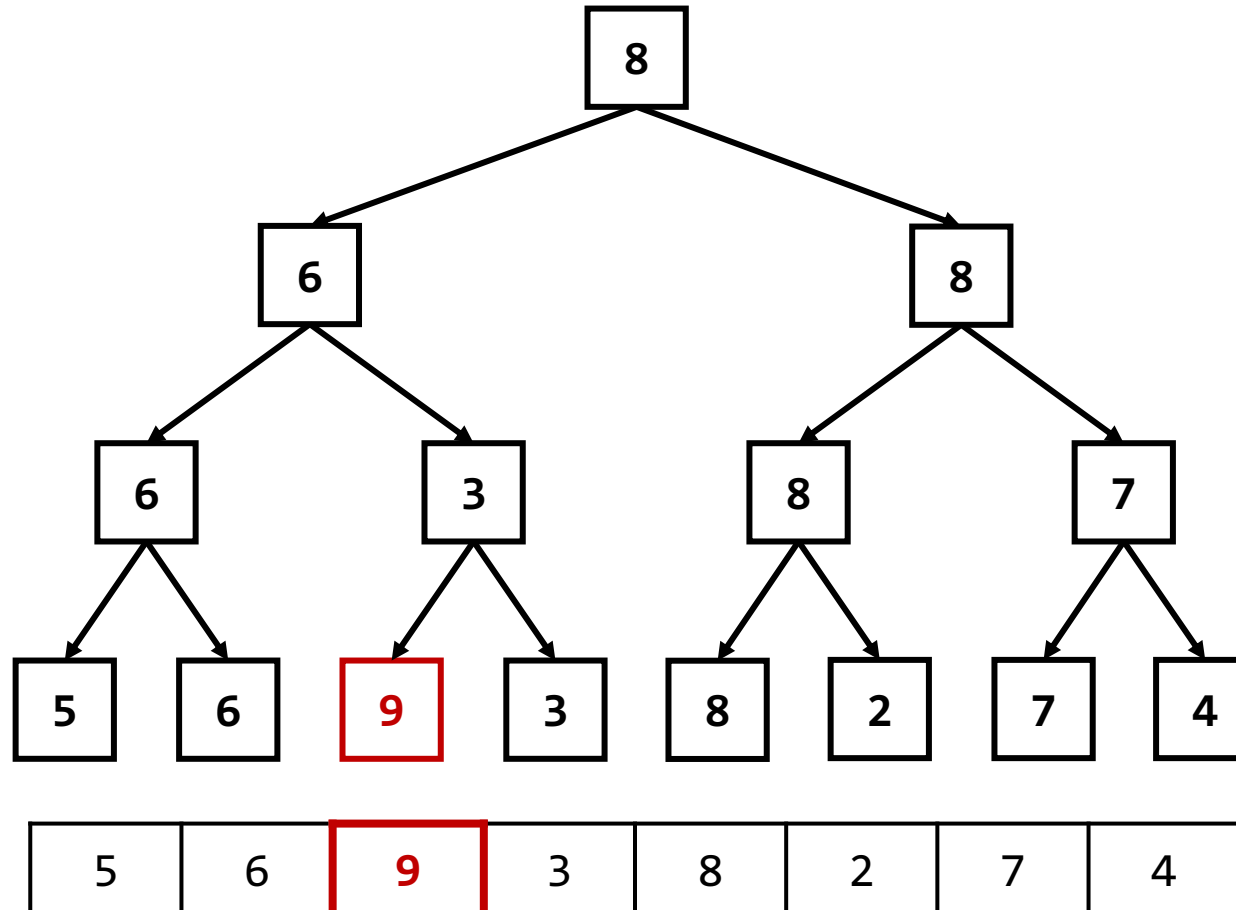
- How to update an element?
 - If the 3rd element is replaced by **9**, which nodes should be updated?



Segment Tree - Update



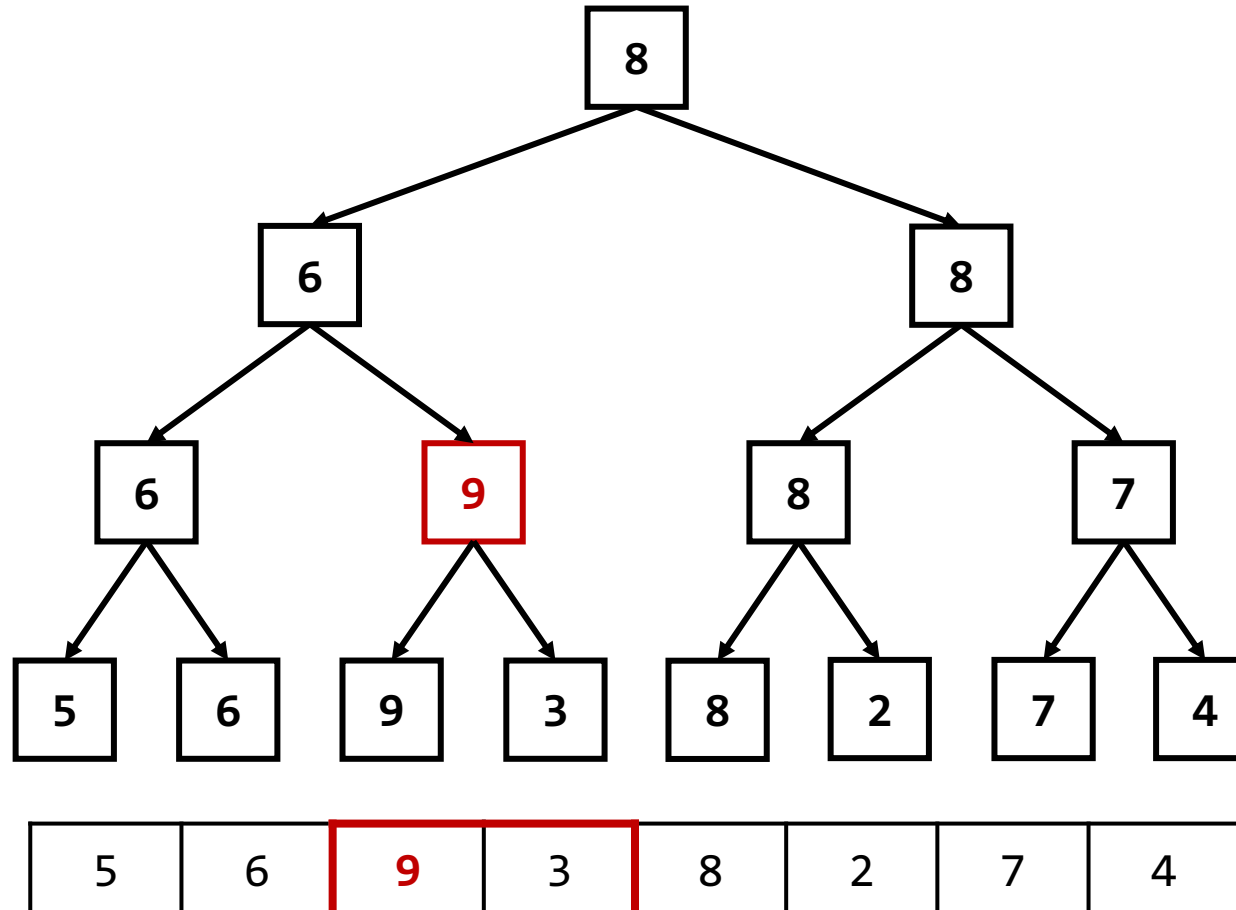
- How to update an element?
 - If the 3rd element is replaced by **9**, which nodes should be updated?



Segment Tree - Update



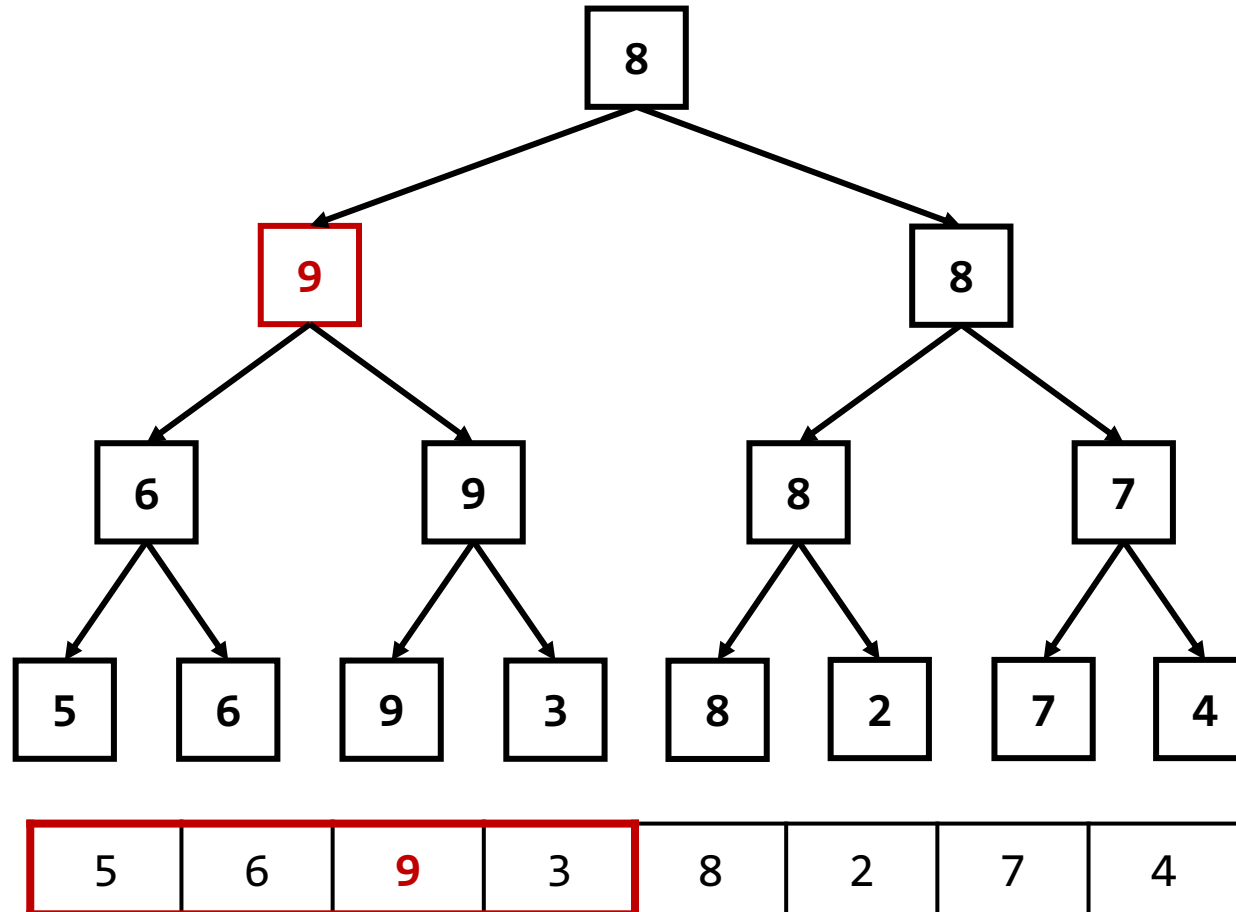
- How to update an element?
 - If the 3rd element is replaced by **9**, which nodes should be updated?



Segment Tree - Update



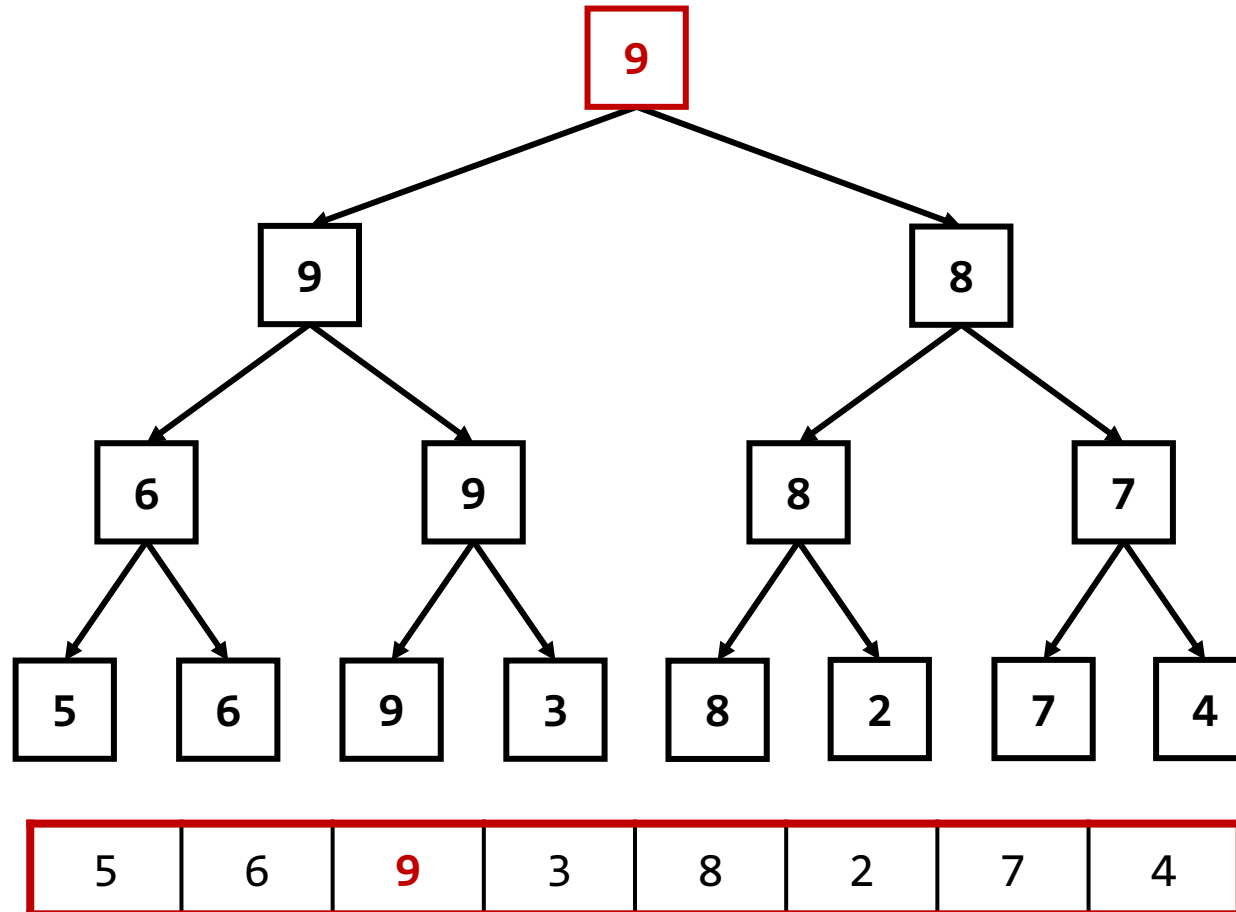
- How to update an element?
 - If the 3rd element is replaced by **9**, which nodes should be updated?



Segment Tree - Update



- How to update an element?
 - If the 3rd element is replaced by **9**, which nodes should be updated?



Segment Tree - Update

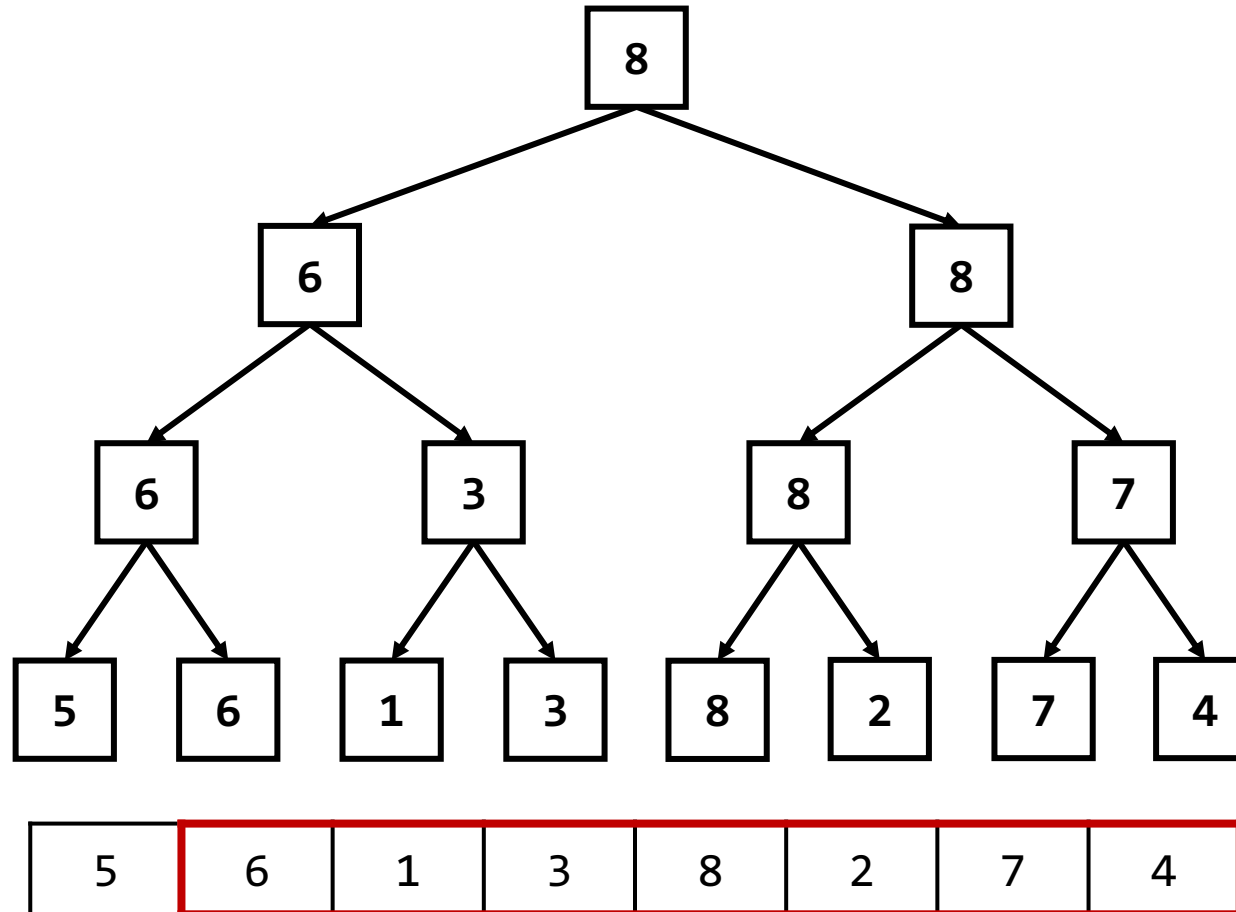


- How to update an element?
 - It is only required to update the corresponding leaf node and its ancestors
 - It takes $O(\log N)$ time since the height of the segment tree is $O(\log N)$

Segment Tree - Segment Operations



- How to compute a cumulative statistic for a segment?
 - If you are interested in the **segment** from the 2nd element to the 8th one,



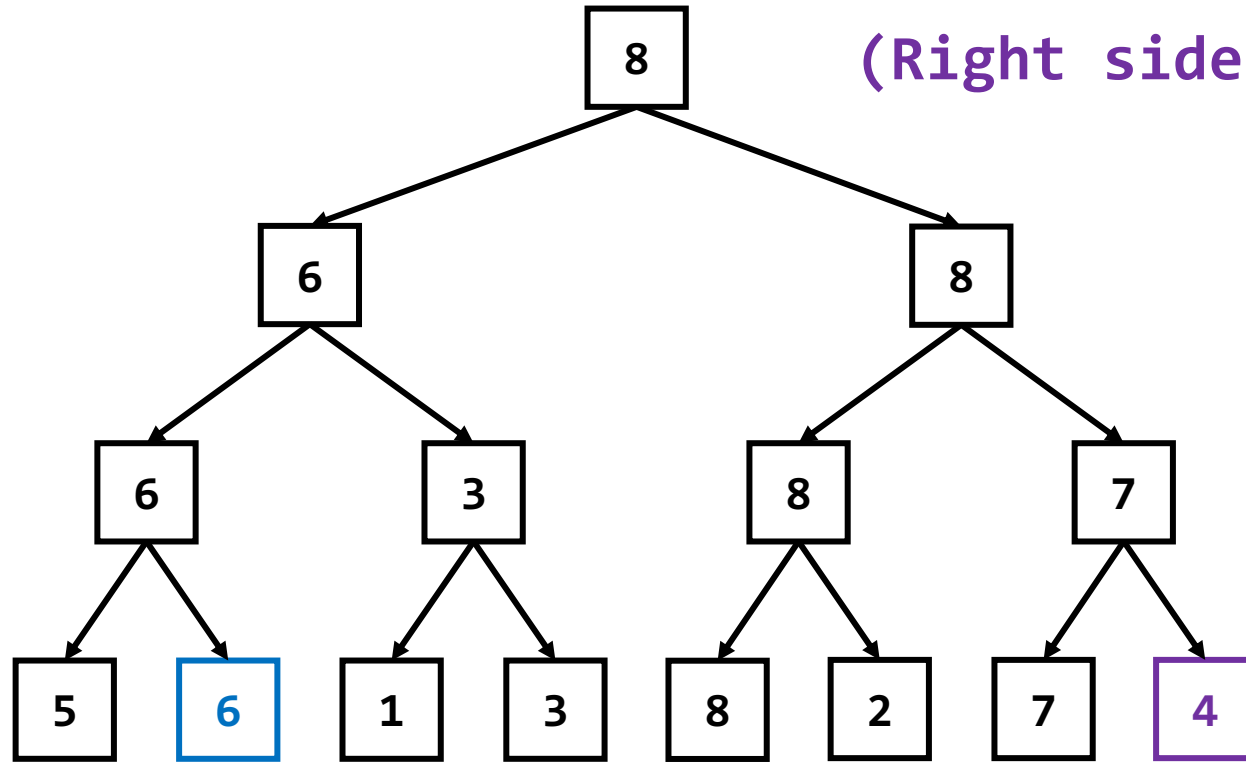
Segment Tree - Segment Operations



- How to compute a cumulative statistic for a segment?
 - If you are interested in the **segment** from the 2nd element to the 8th one,

(Left side)

(Right side)



← Start from bottom

5	6	1	3	8	2	7	4
---	---	---	---	---	---	---	---

Segment Tree - Segment Operations



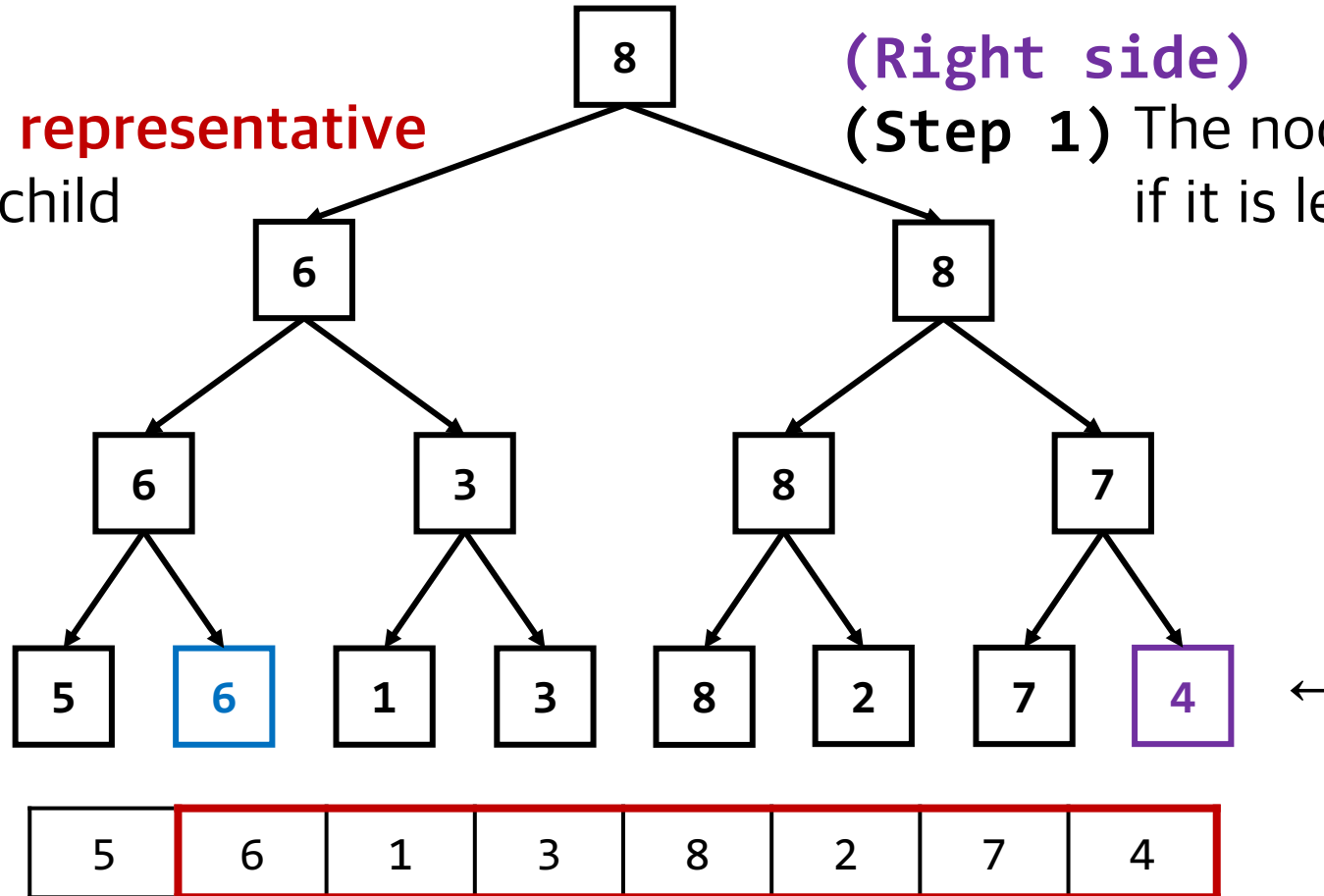
- How to compute a cumulative statistic for a segment?
 - If you are interested in the **segment** from the 2nd element to the 8th one,

(Left side)

(Step 1) The node is **representative** if it is right child

(Right side)

(Step 1) The node is **representative** if it is left child



Segment Tree - Segment Operations



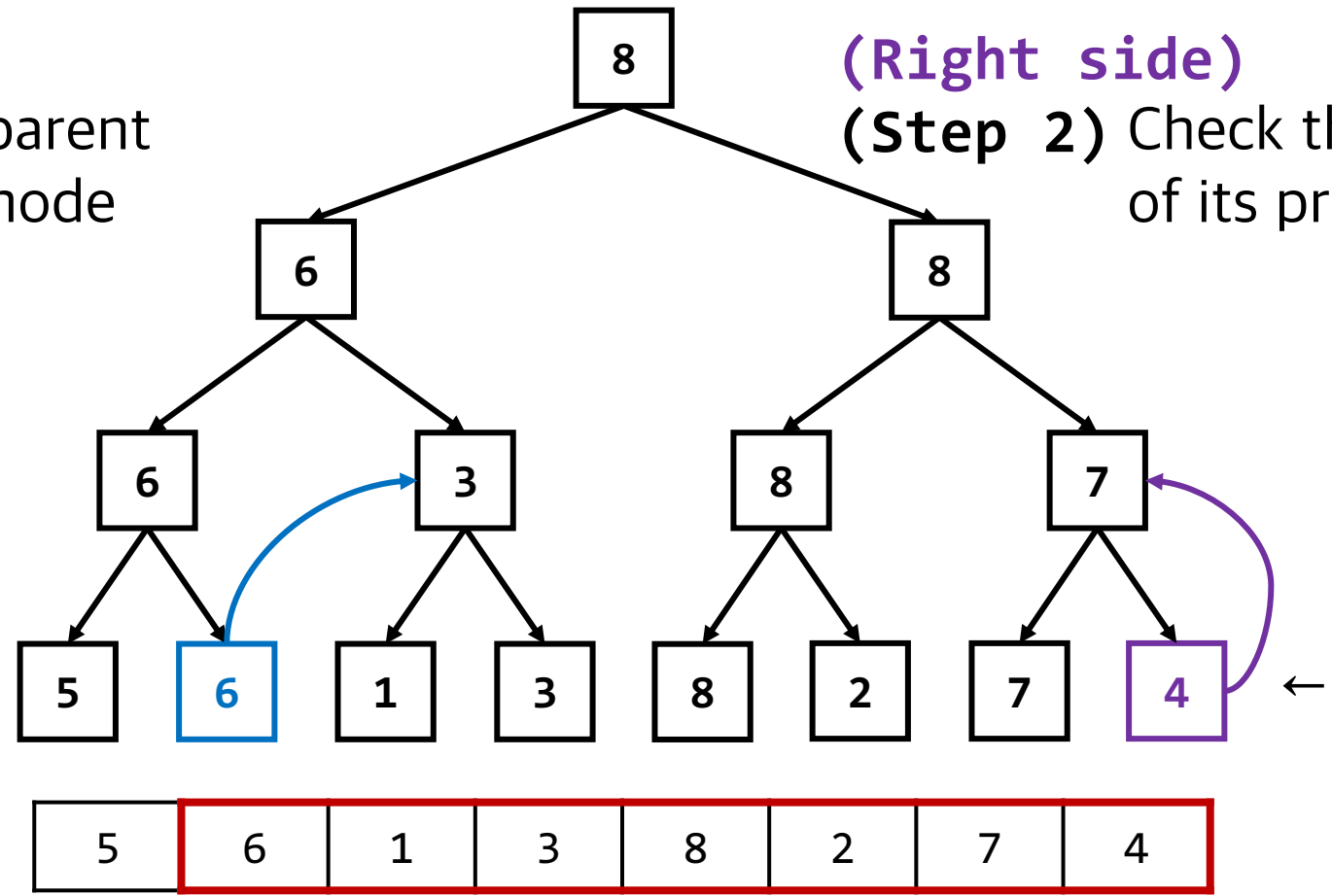
- How to compute a cumulative statistic for a segment?
 - If you are interested in the **segment** from the 2nd element to the 8th one,

(Left side)

(Step 2) Check the parent of its next node

(Right side)

(Step 2) Check the parent of its previous node



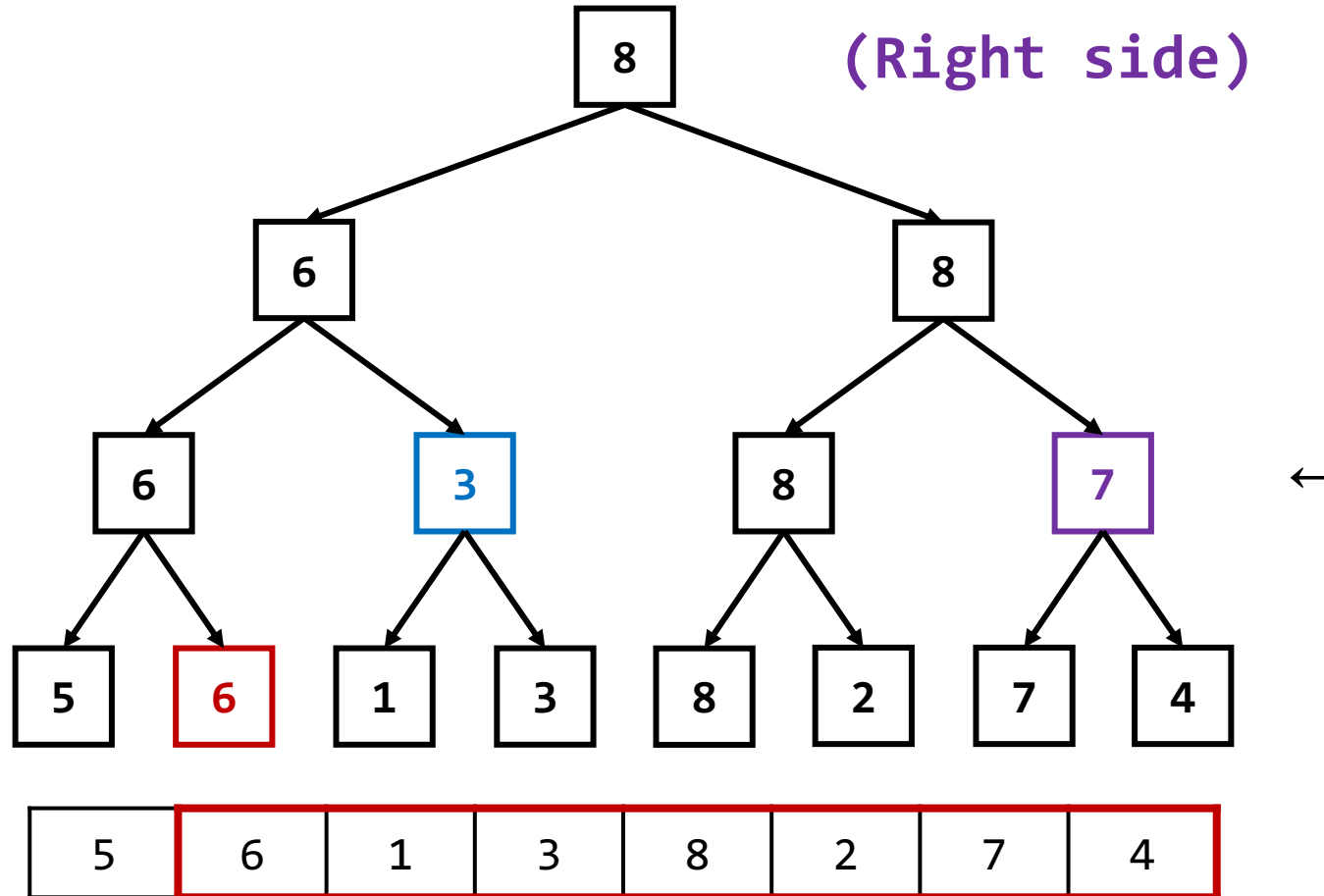
Segment Tree - Segment Operations



- How to compute a cumulative statistic for a segment?
 - If you are interested in the **segment** from the 2nd element to the 8th one,

(Left side)

(Right side)



Segment Tree - Segment Operations



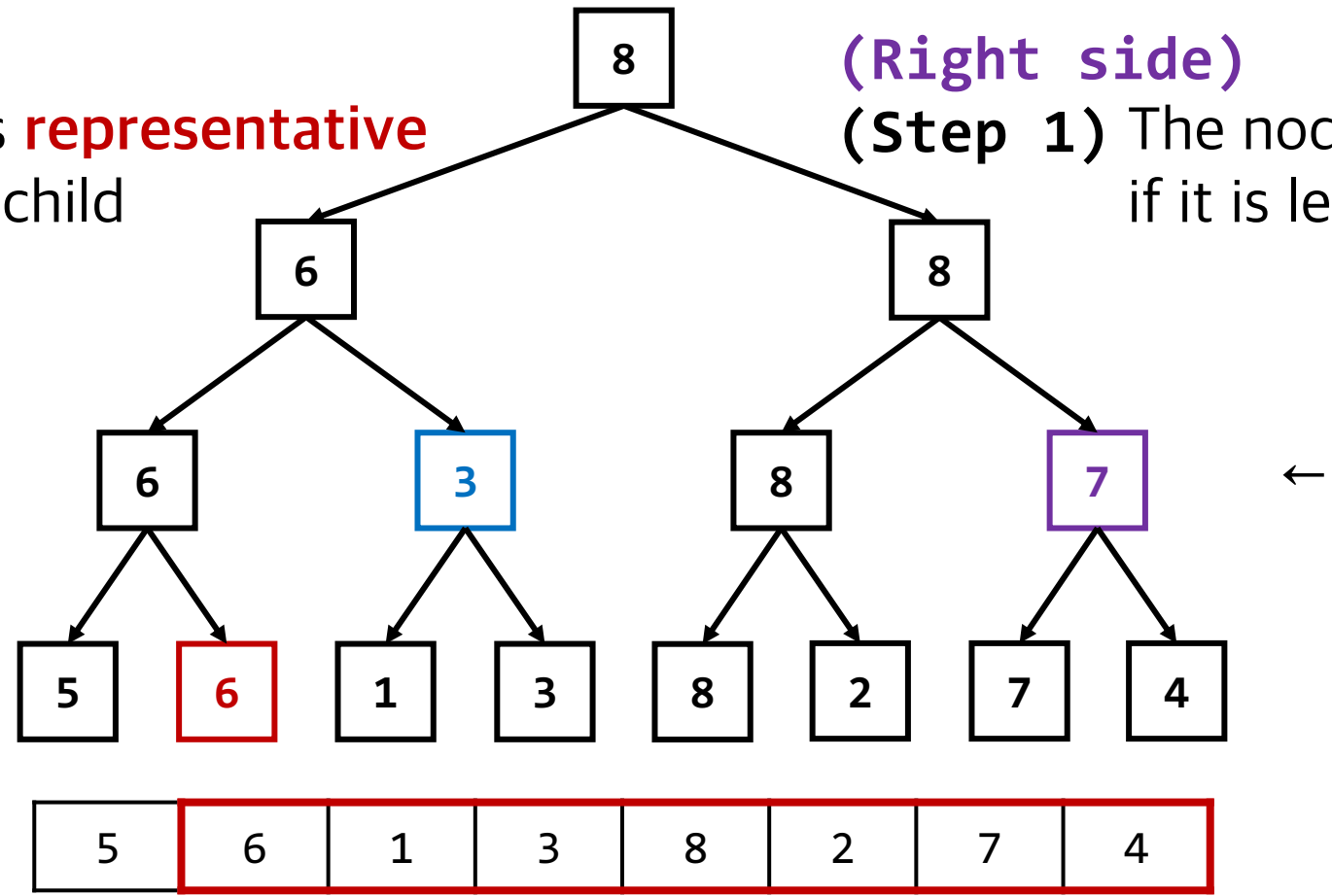
- How to compute a cumulative statistic for a segment?
 - If you are interested in the **segment** from the 2nd element to the 8th one,

(Left side)

(Step 1) The node is **representative** if it is right child

(Right side)

(Step 1) The node is **representative** if it is left child



Segment Tree - Segment Operations



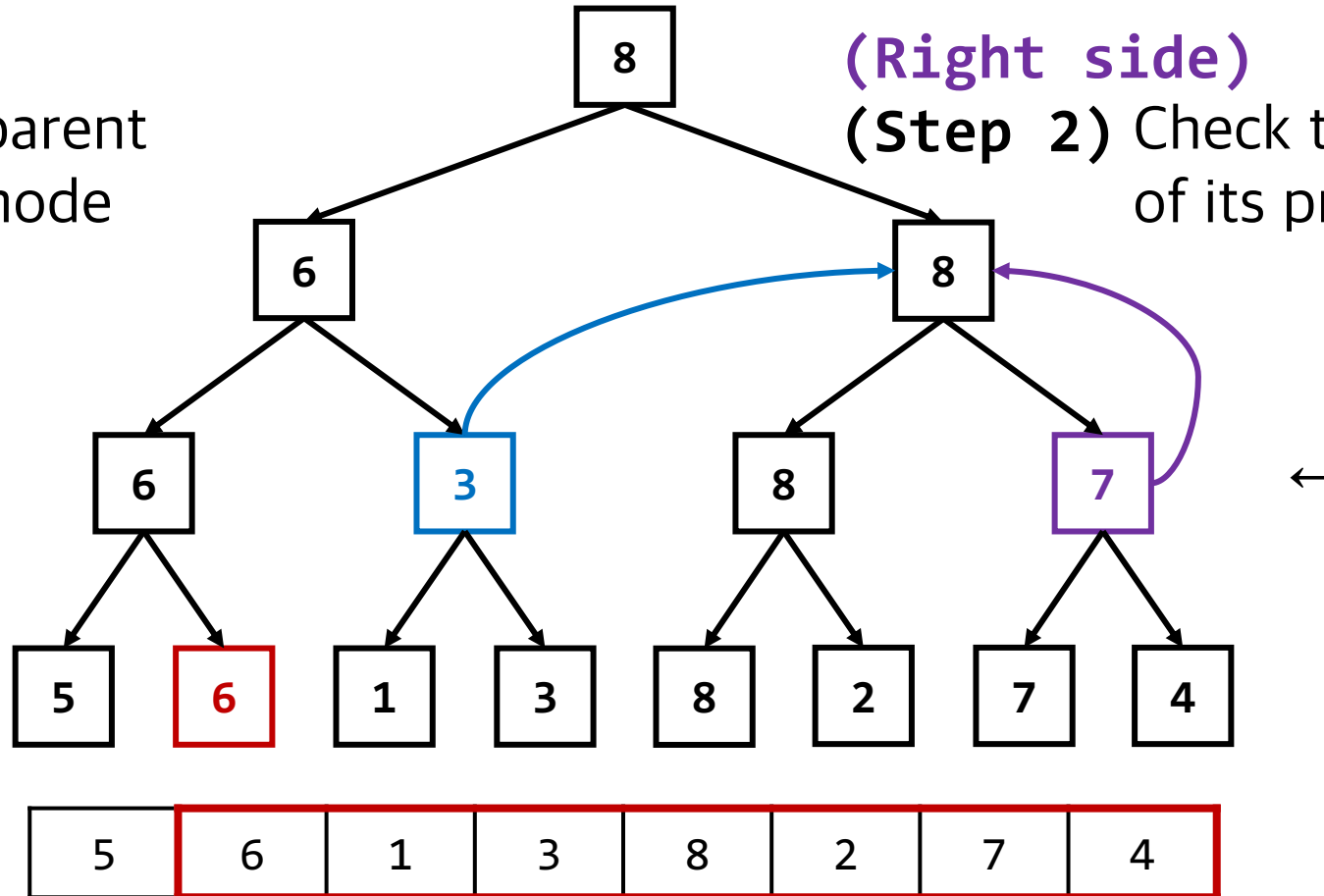
- How to compute a cumulative statistic for a segment?
 - If you are interested in the **segment** from the 2nd element to the 8th one,

(Left side)

(Step 2) Check the parent of its next node

(Right side)

(Step 2) Check the parent of its previous node

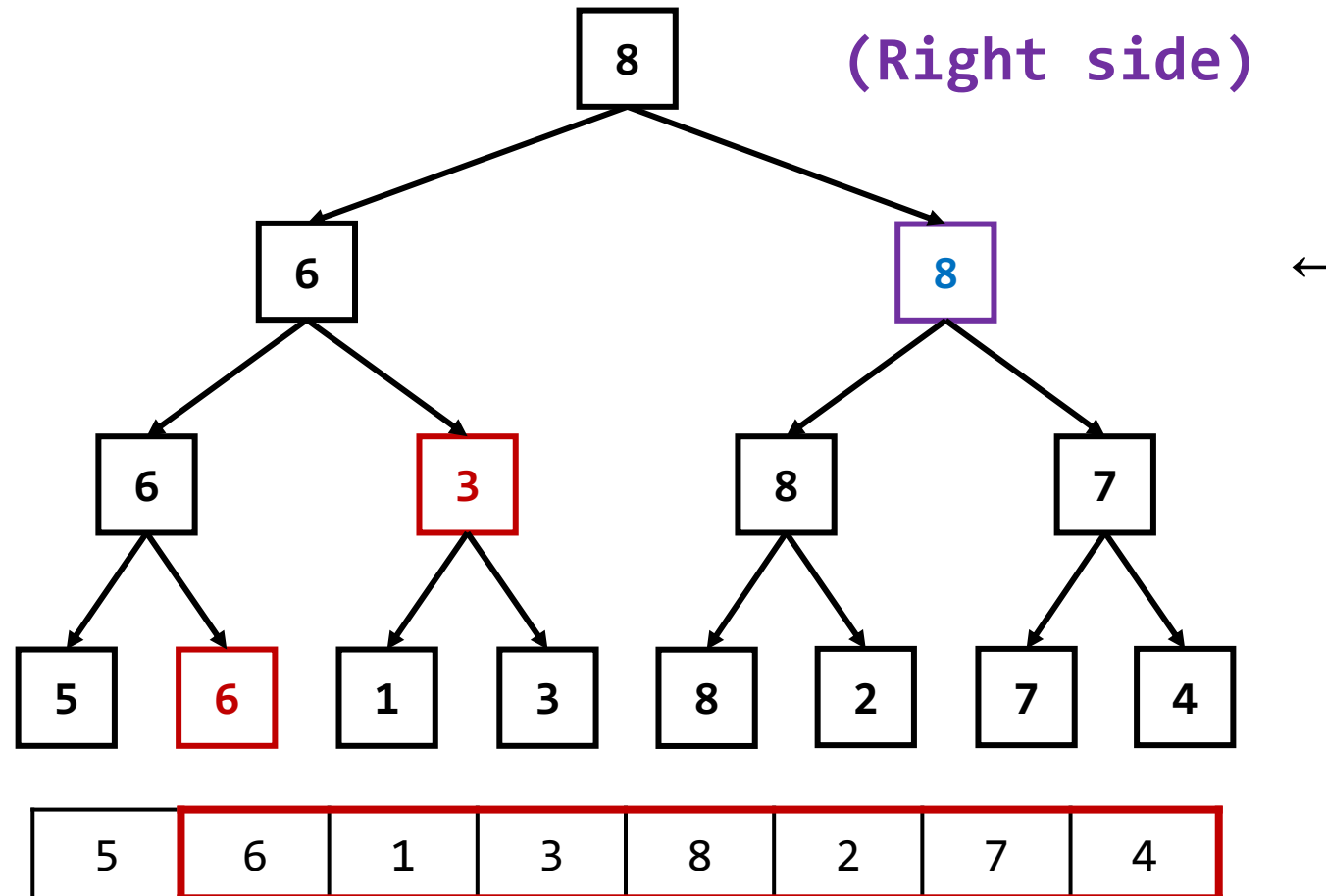


Segment Tree - Segment Operations



- How to compute a cumulative statistic for a segment?
 - If you are interested in the **segment** from the 2nd element to the 8th one,

(Left side)

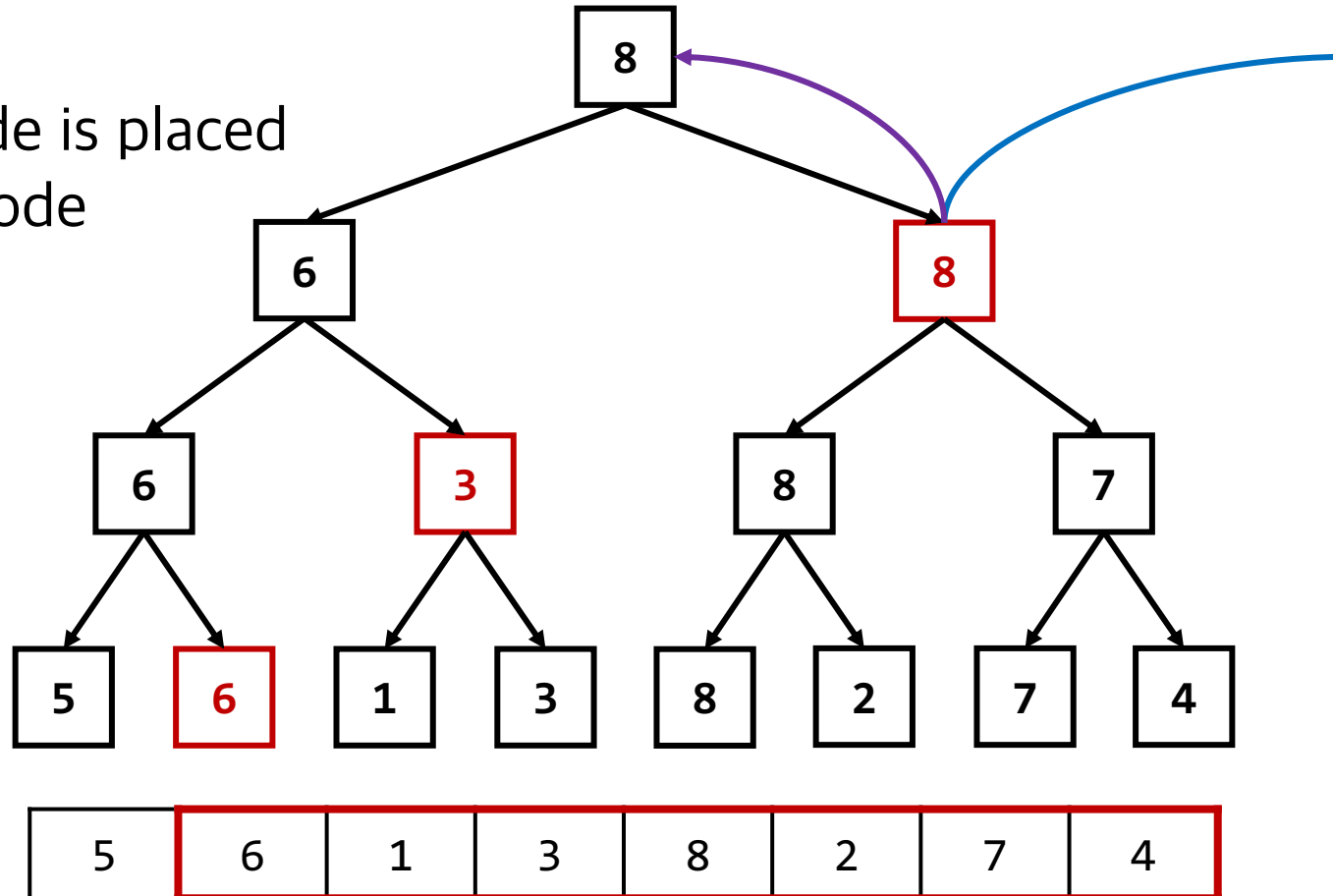


Segment Tree - Segment Operations



- How to compute a cumulative statistic for a segment?
 - If you are interested in the **segment** from the 2nd element to the 8th one,

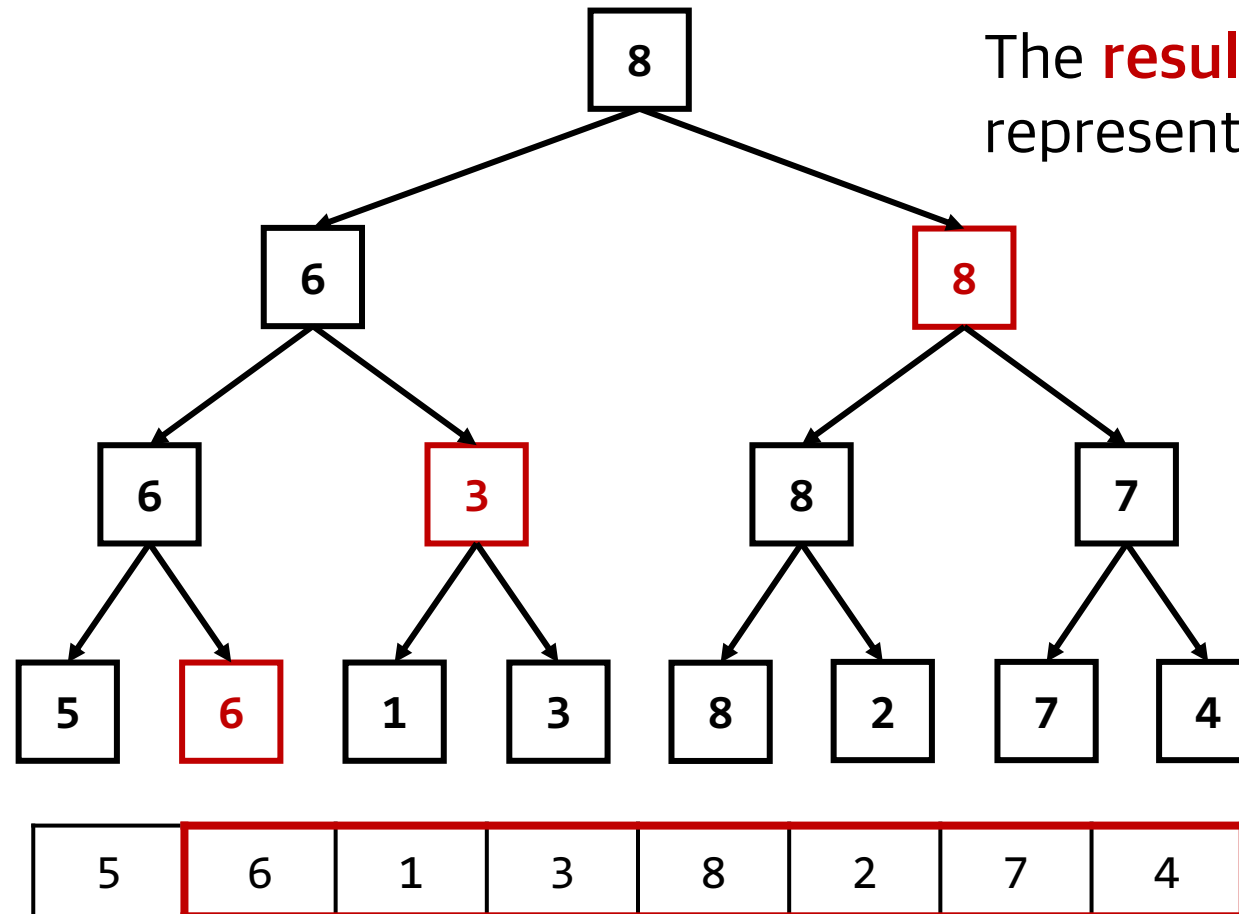
Repeat (**Step 1-2**)
until the **left-side** node is placed
after the **right-side** node



Segment Tree - Segment Operations



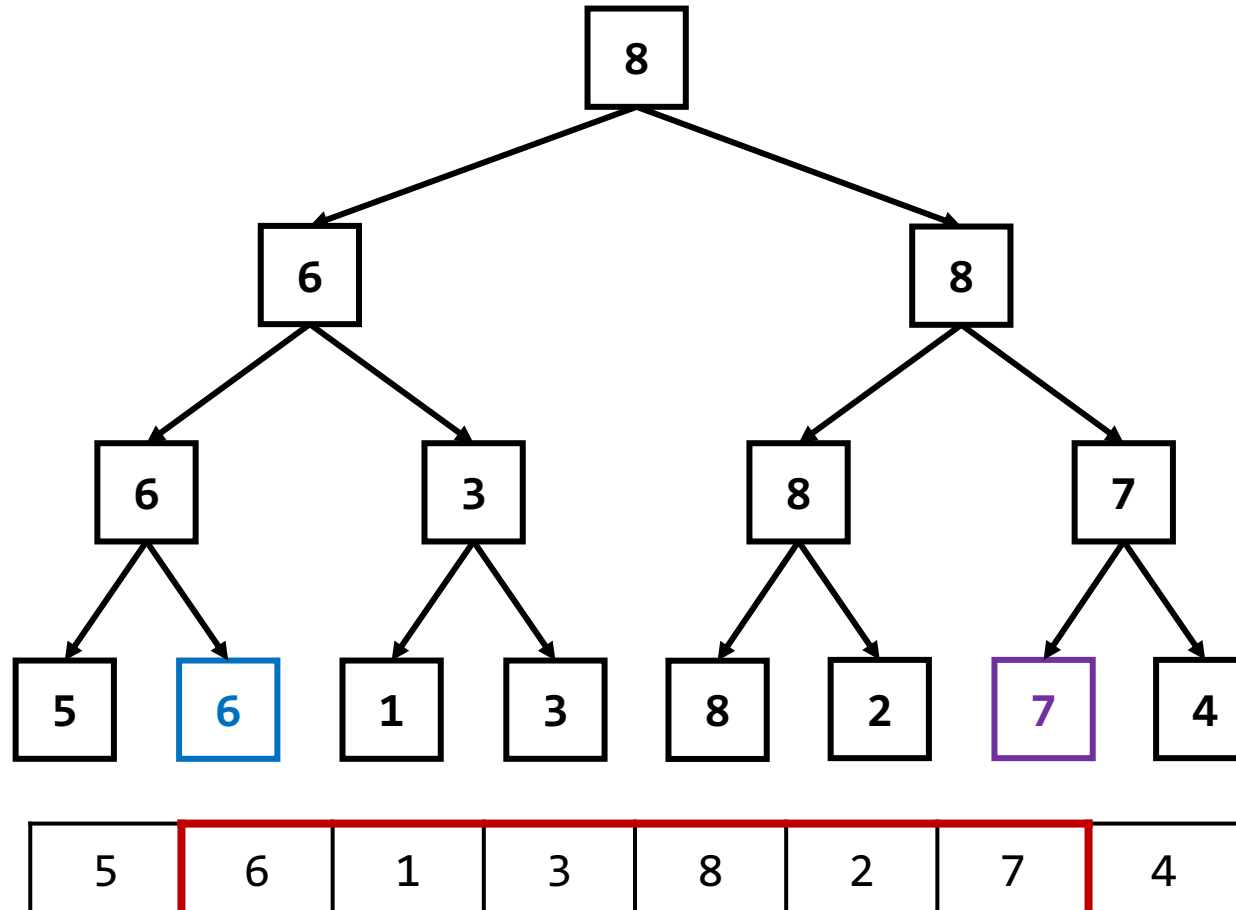
- How to compute a cumulative statistic for a segment?
 - If you are interested in the **segment** from the 2nd element to the 8th one,



Segment Tree - Segment Operations



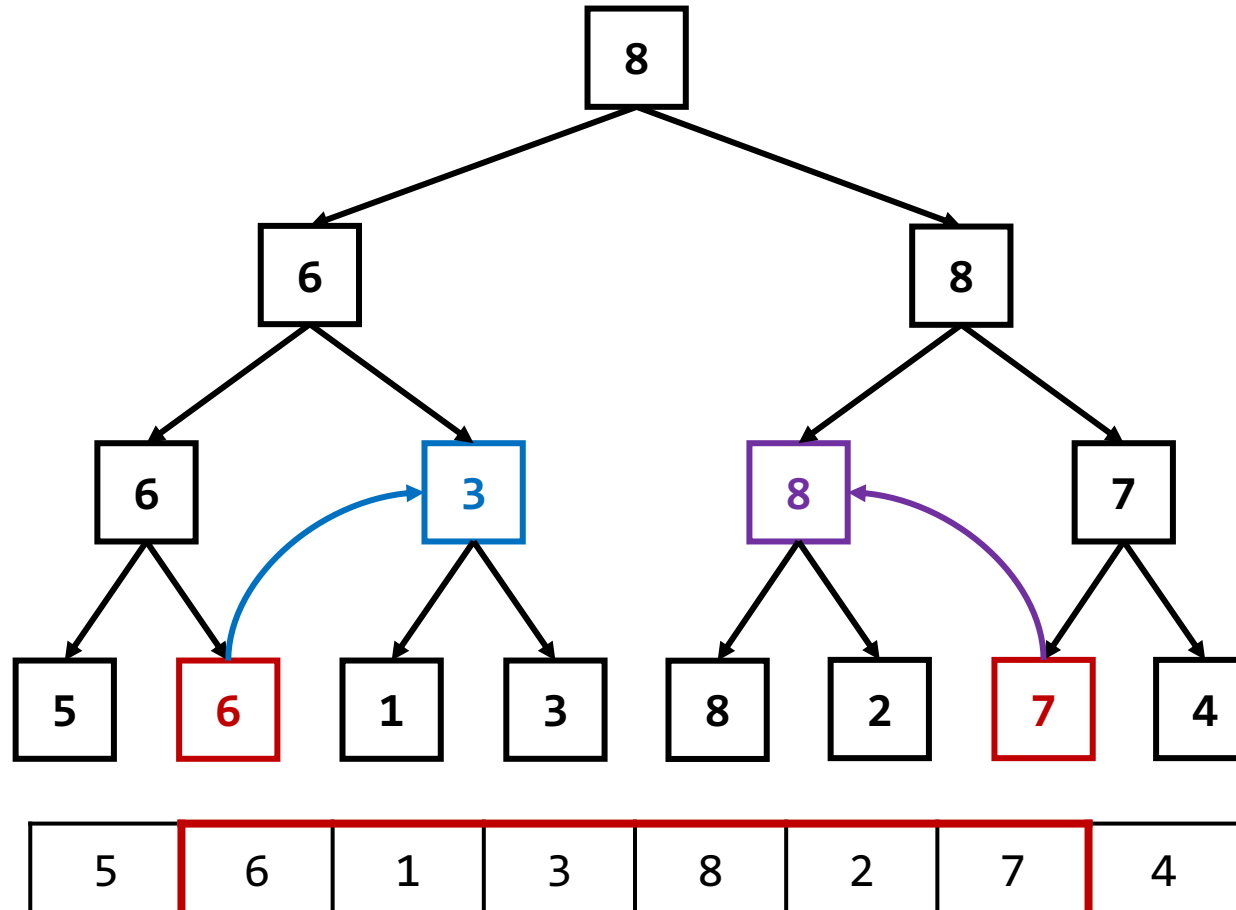
- How to compute a cumulative statistic for a segment?
 - If you are interested in the **segment** from the 2nd element to the 7th one,



Segment Tree - Segment Operations



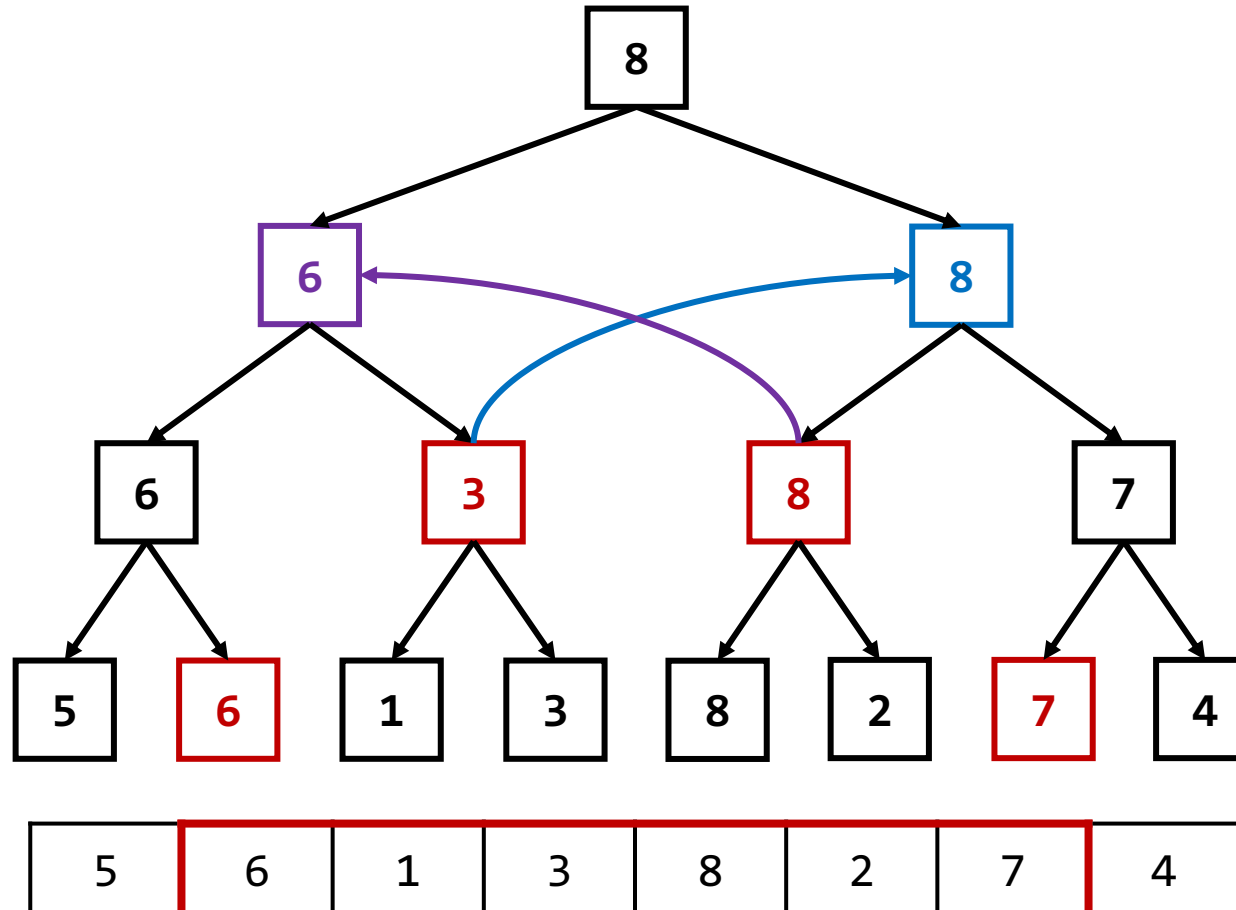
- How to compute a cumulative statistic for a segment?
 - If you are interested in the **segment** from the 2nd element to the 7th one,



Segment Tree - Segment Operations



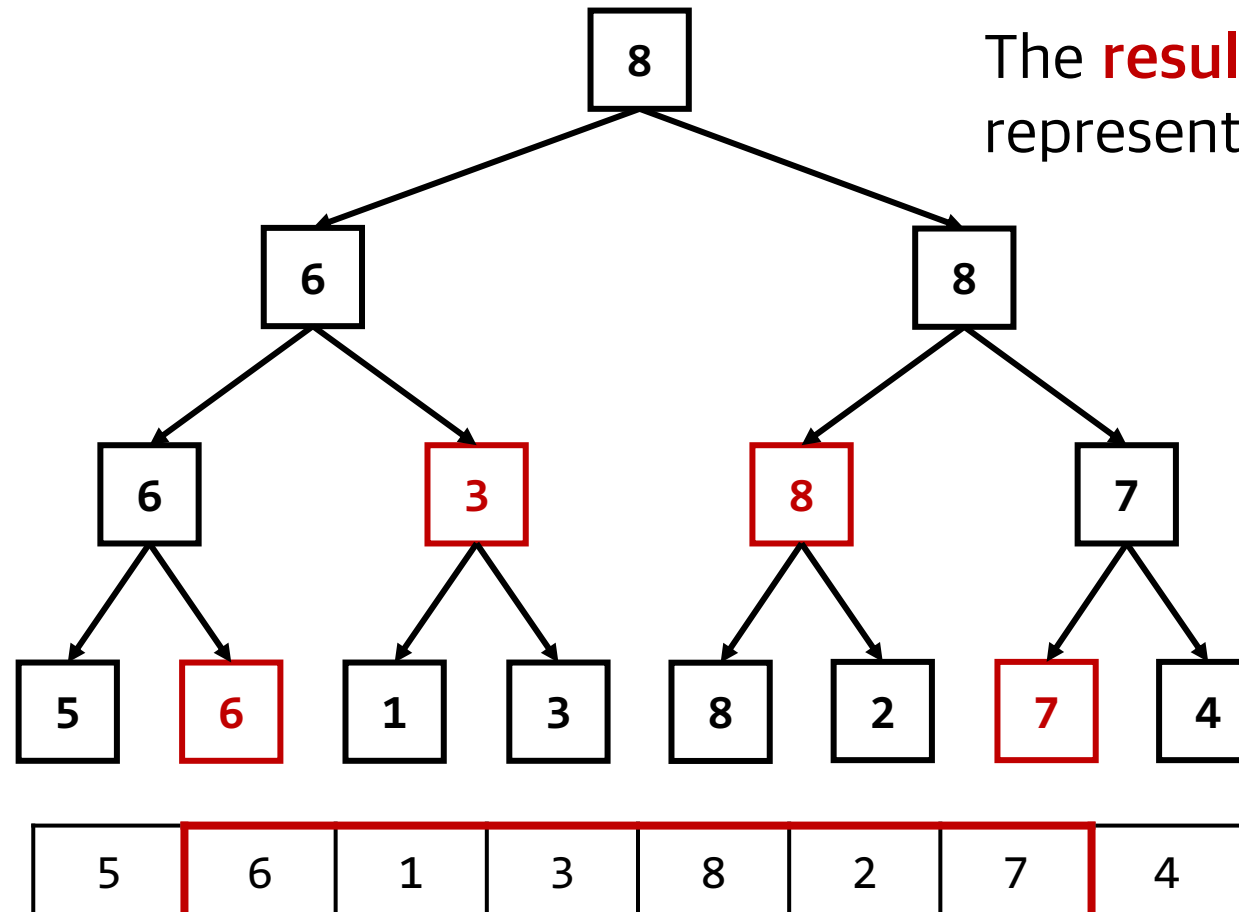
- How to compute a cumulative statistic for a segment?
 - If you are interested in the **segment** from the 2nd element to the 7th one,



Segment Tree - Segment Operations



- How to compute a cumulative statistic for a segment?
 - If you are interested in the **segment** from the 2nd element to the 7th one,



Segment Tree - Segment Operations



- How to compute a cumulative statistic for a segment?
 - The representative nodes for the segment can be found in $O(\log N)$ time
 - The cumulative statistic (e.g., sum, max) can be also computed in $O(\log N)$ time

Segment Tree - Implementation



- Main components for the segment tree structure
 - **Construct** a segment tree from an array of n elements
 - **Update** an element in the segment tree
 - **Compute** the maximum value for a segment

```
typedef struct _SegmentTree {  
    int nodes[2*MAX_SIZE+1];  
    int size;  
} SegmentTree;  
  
void construct(int items[], int n, SegmentTree *tree);  
void update(SegmentTree *tree, int item, int index);  
int computeMax(SegmentTree *tree, int left, int right);
```

Another Use Case of Segment Tree

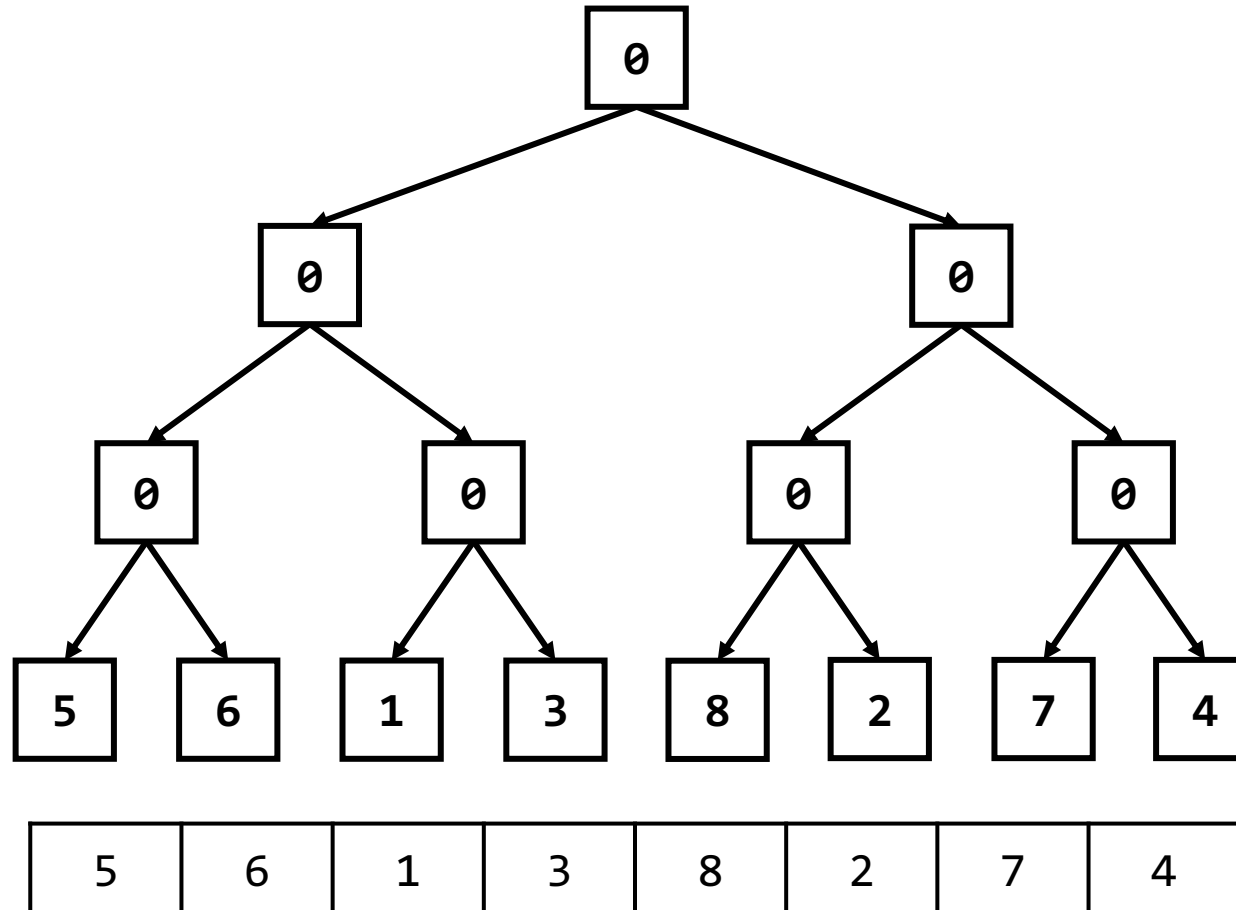


- The previous case is for ...
 - Updating the value for a single element - $O(\log N)$
 - Compute the cumulative value for a segment - $O(\log N)$
- One can use segment tree for ...
 - Updating values of all elements in a segment - $O(\log N)$
 - Computing the value of a single element - $O(\log N)$

Another Use Case of Segment Tree



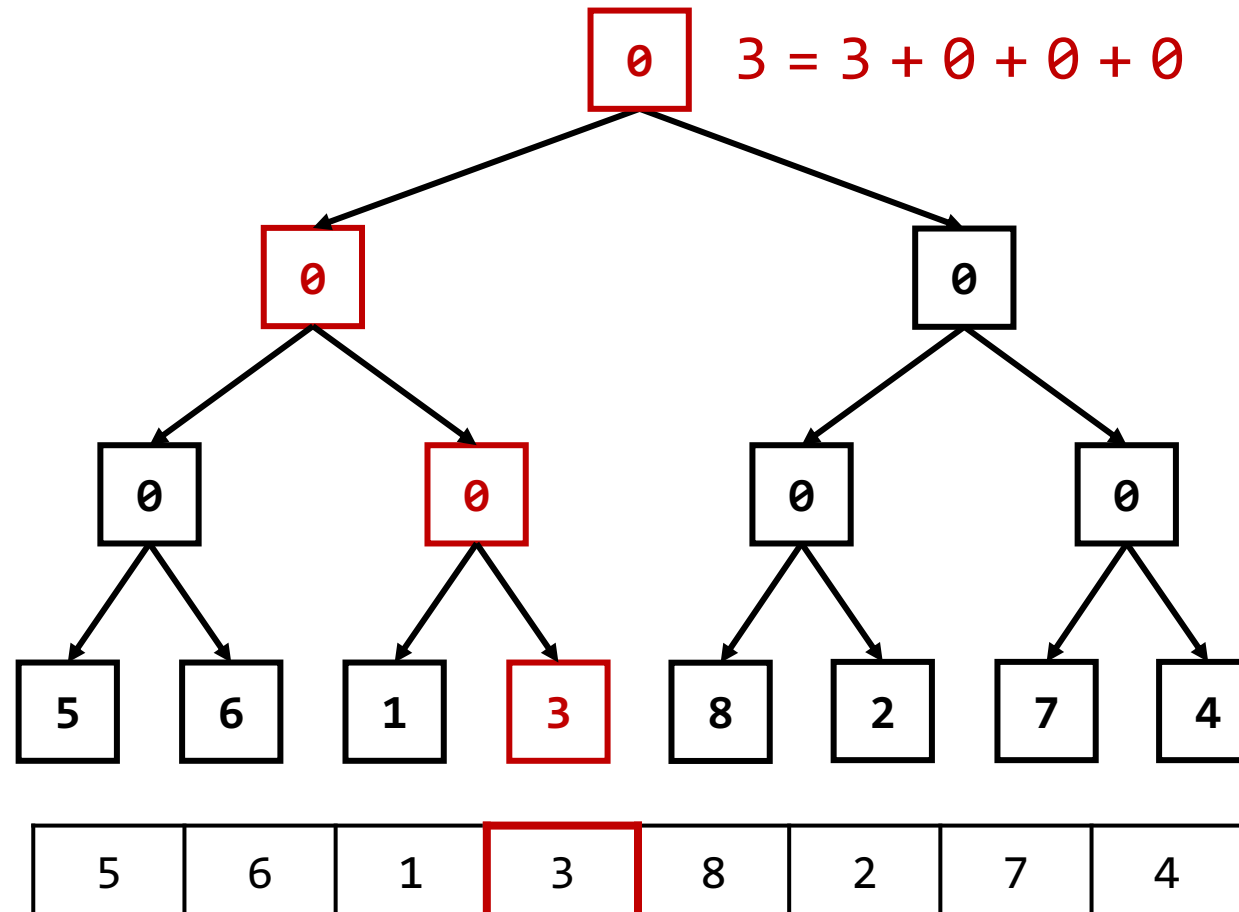
- How to implement the use case?
 - The value of an element is cumulation from its leaf node to the root node



Another Use Case of Segment Tree



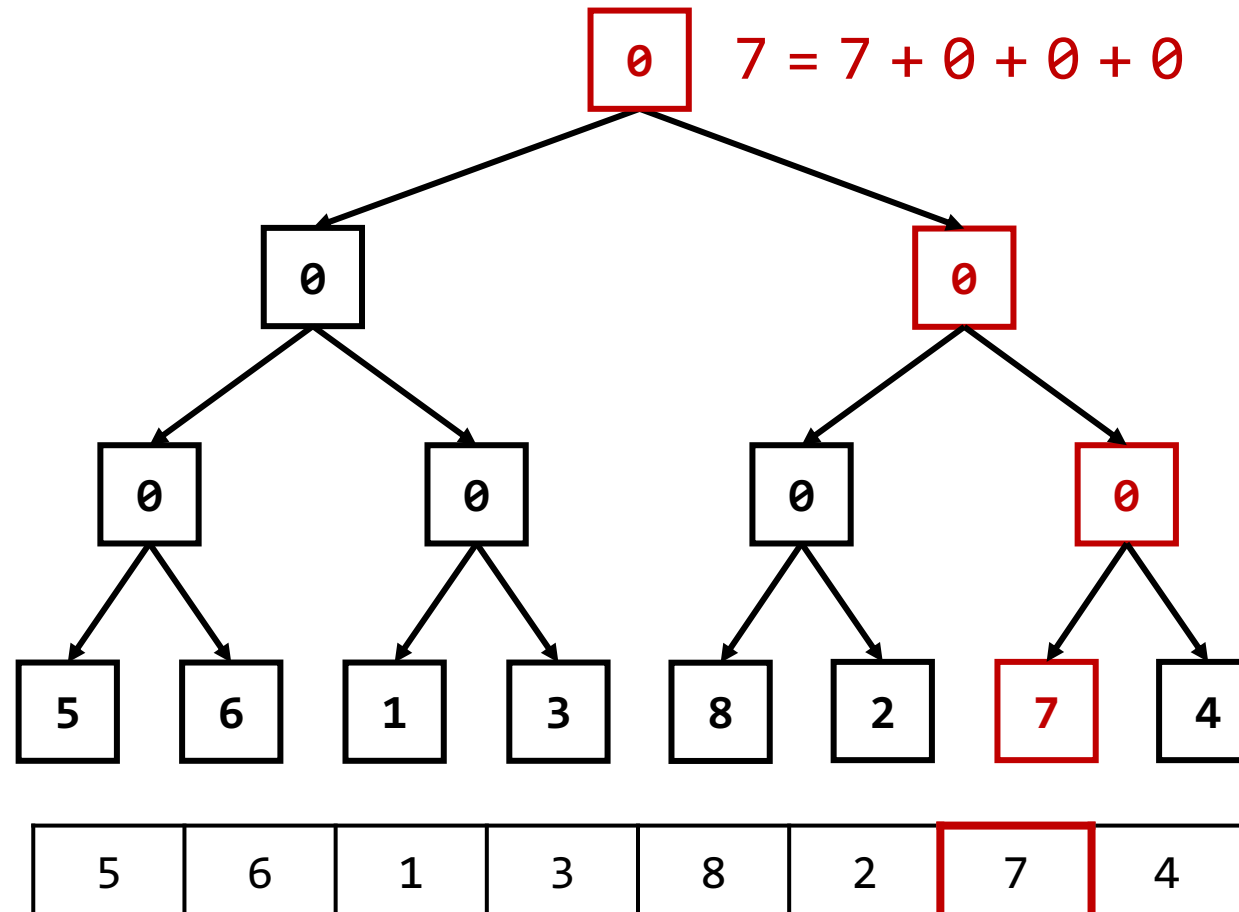
- How to implement the use case?
 - The value of an element is cumulation from its leaf node to the root node



Another Use Case of Segment Tree



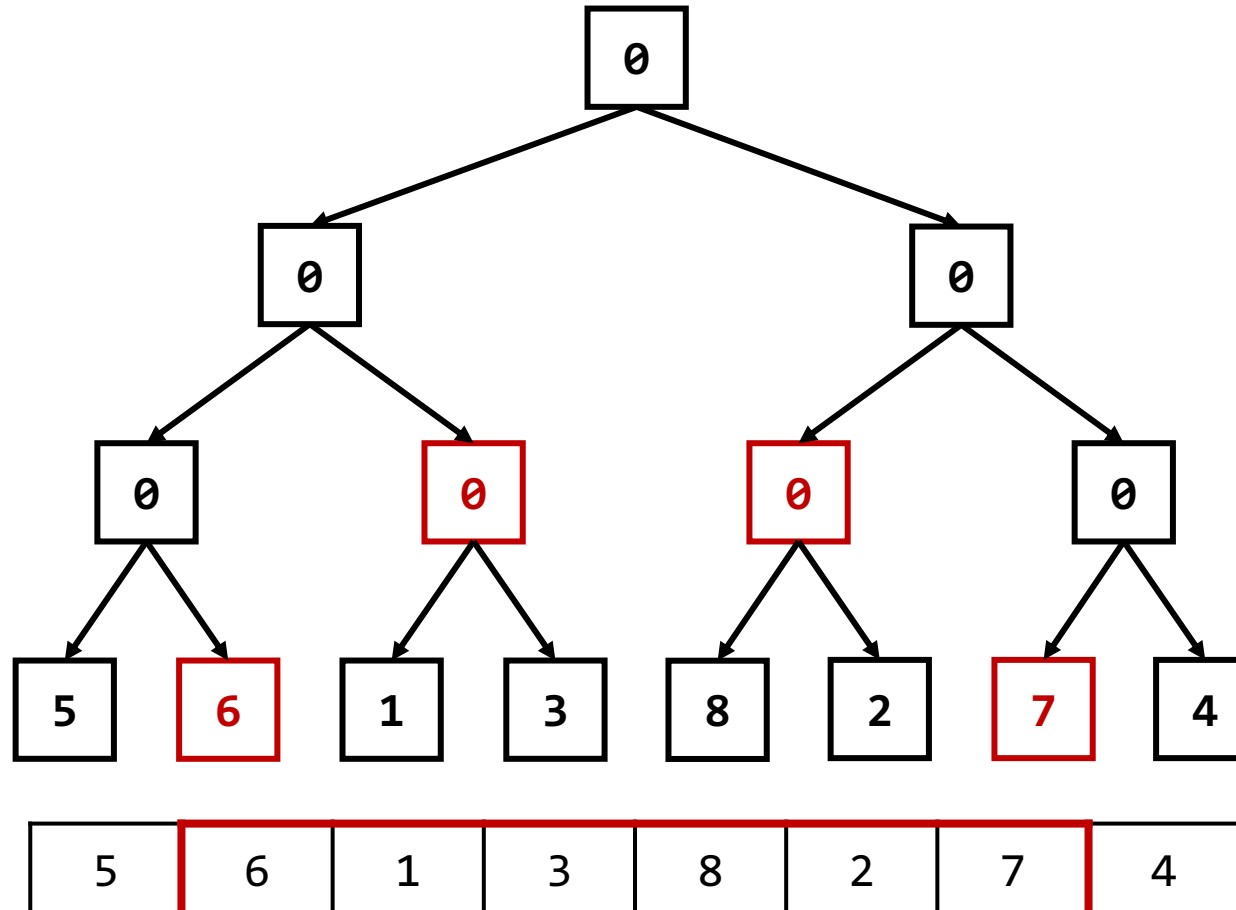
- How to implement the use case?
 - The value of an element is cumulation from its leaf node to the root node



Another Use Case of Segment Tree



- How to implement the use case?
 - Update elements in a **segment** using **representative nodes**

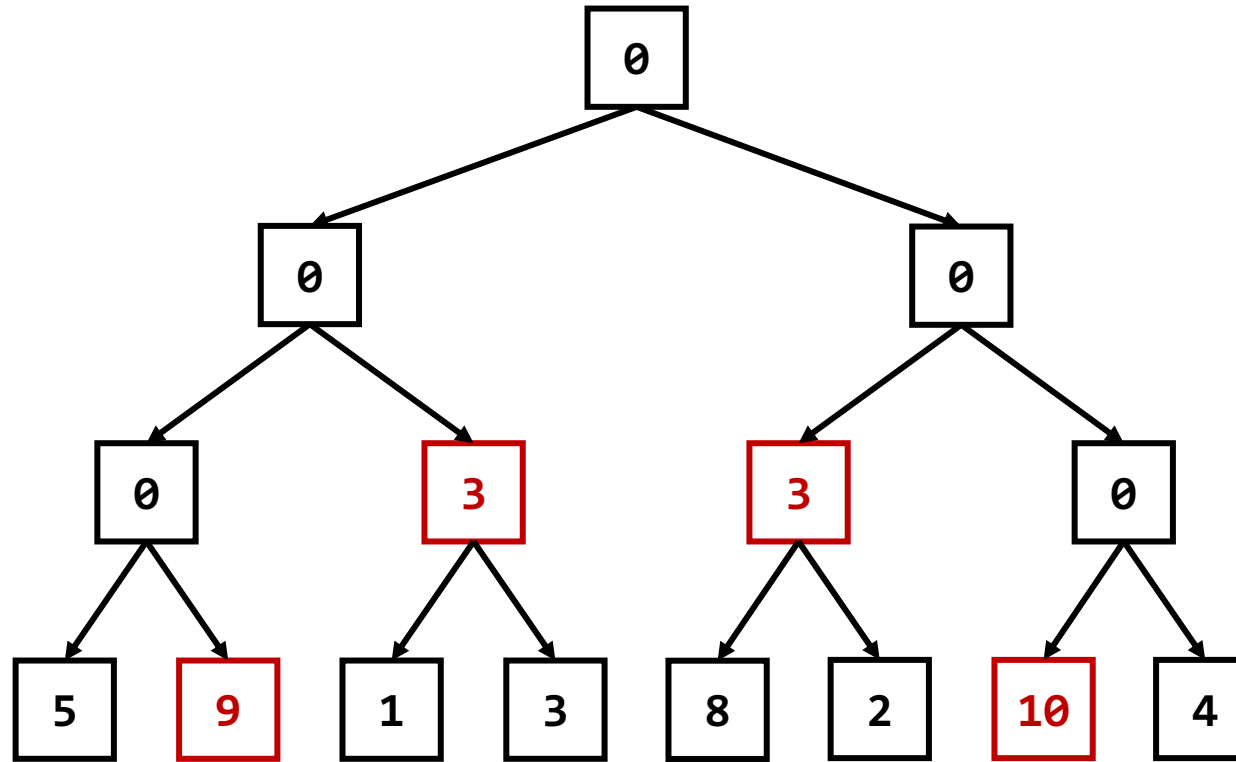


ADD 3

Another Use Case of Segment Tree



- How to implement the use case?
 - Update elements in a **segment** using **representative nodes**

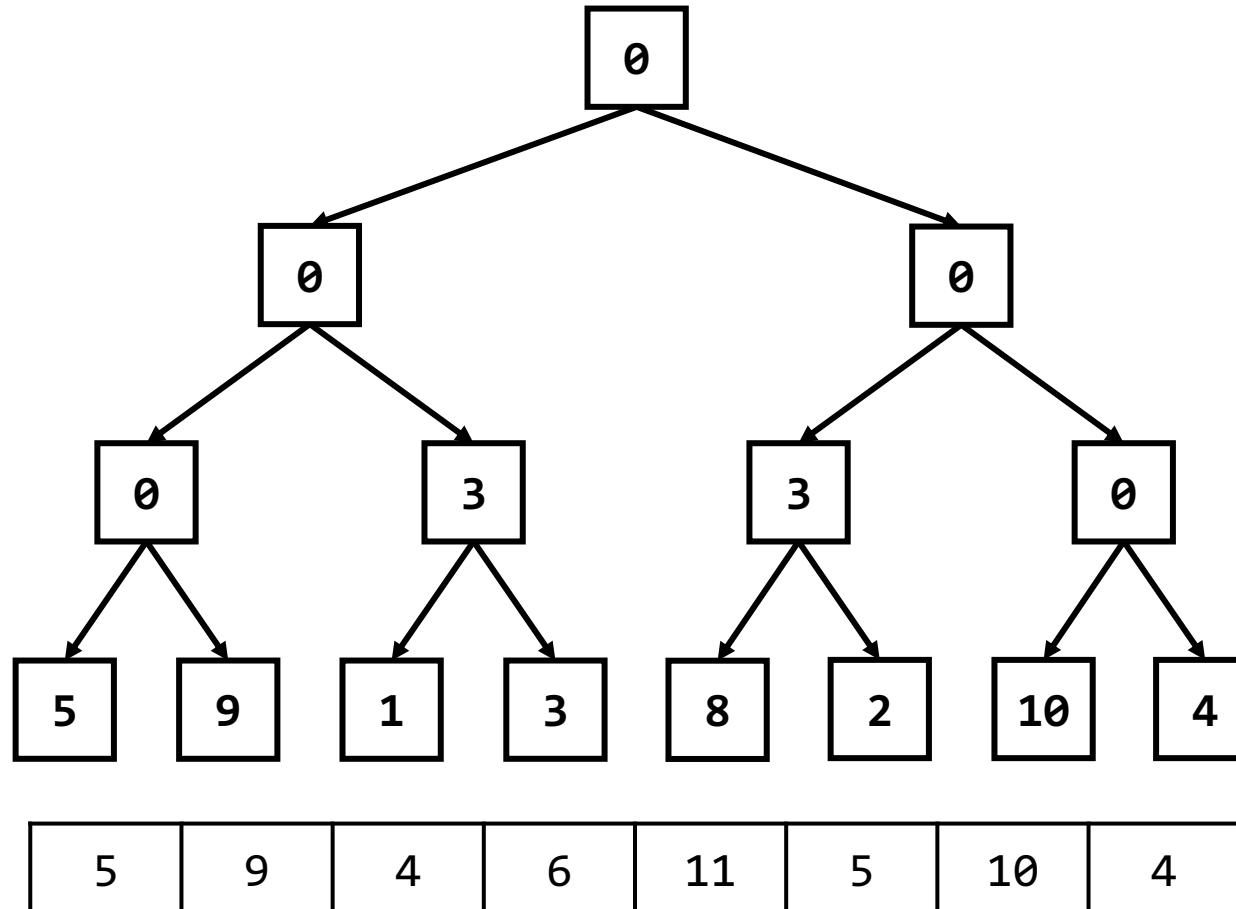


ADD 3

Another Use Case of Segment Tree



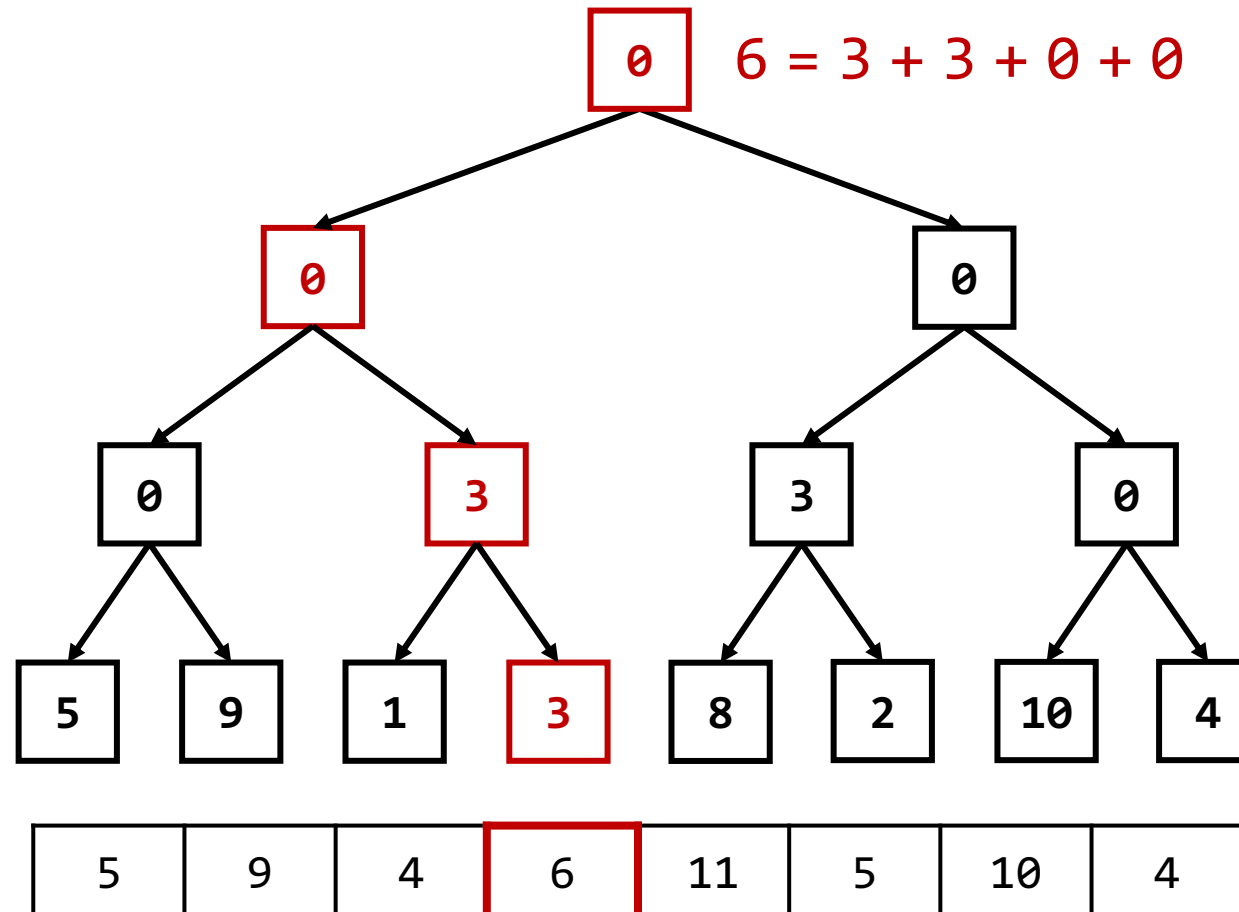
- How to implement the use case?
 - The value of an element is cumulation from its leaf node to the root node



Another Use Case of Segment Tree



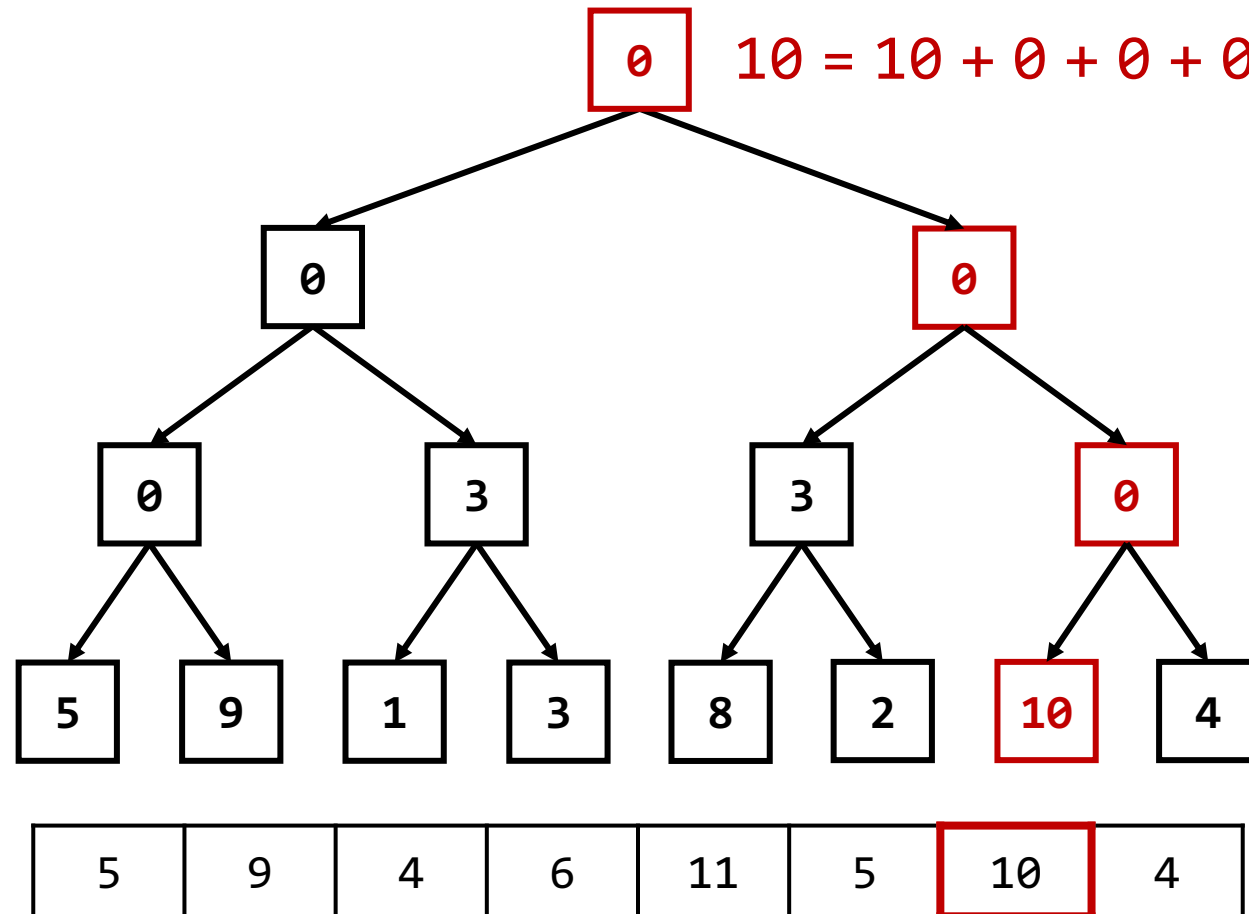
- How to implement the use case?
 - The value of an element is cumulation from its leaf node to the root node



Another Use Case of Segment Tree



- How to implement the use case?
 - The value of an element is cumulation from its leaf node to the root node



- **Segment Tree** is an efficient and easily-implemented structure
 - But, if the number of elements is frequently changed, it might be not useful
 - Some operations might be not supported
 - E.g., average, cross-product, ...
 - For segment tree, the associative property should be satisfied, i.e., $f(A, f(B, C)) = f(f(A, B), C)$
- You must check if the segment tree is suitable for the problem you are considering

Any Questions?

