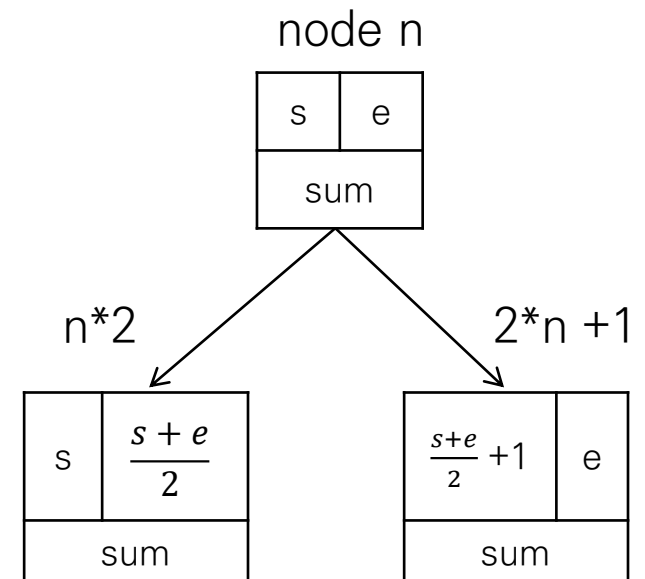


세그먼트 트리

■ n개의 정수에 대한 일정 구간의 합을 포화이진 트리에 저장.

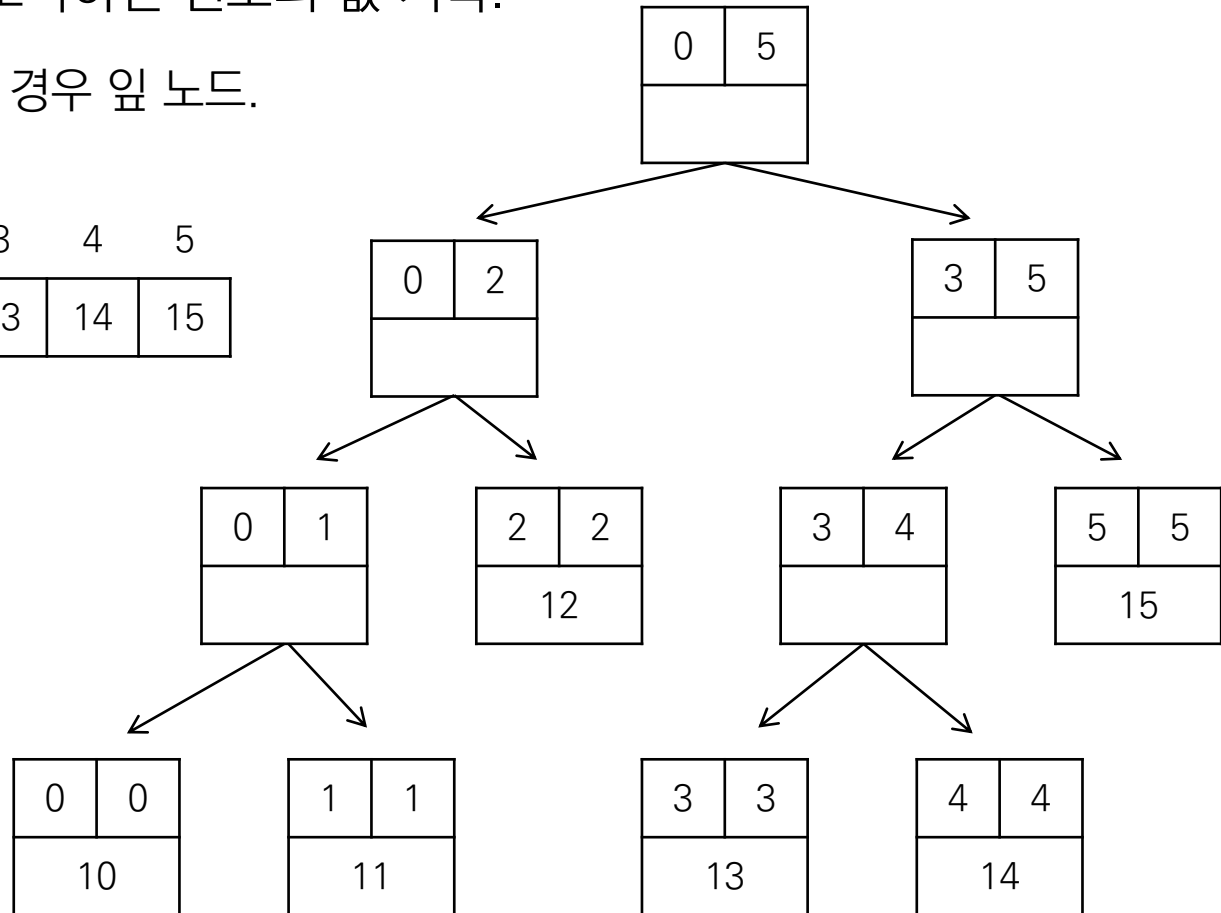
- 각 노드에는 구간의 시작과 끝 인덱스와 구간의 합을 저장한다.
- 루트 노드에 전 구간 인덱스와 합이 저장된다.
- 자식 노드에는 부모가 가진 구간의 절반씩을 저장.
 - 부모 노드의 구간이 s, e 인 경우
 - 왼쪽 자식 노드의 구간 $s, (s+e)/2$.
 - 오른쪽 자식 노드의 구간 $(s+e)/2+1, e$.
- $s == e$ 인 경우 잎 노드.
 - 구간의 길이가 1.



■ 전위 순회로 구간을 나눔.

- 앞 노드에 도착하면 원소의 값 기록.
- $s == e$ 인 경우 앞 노드.

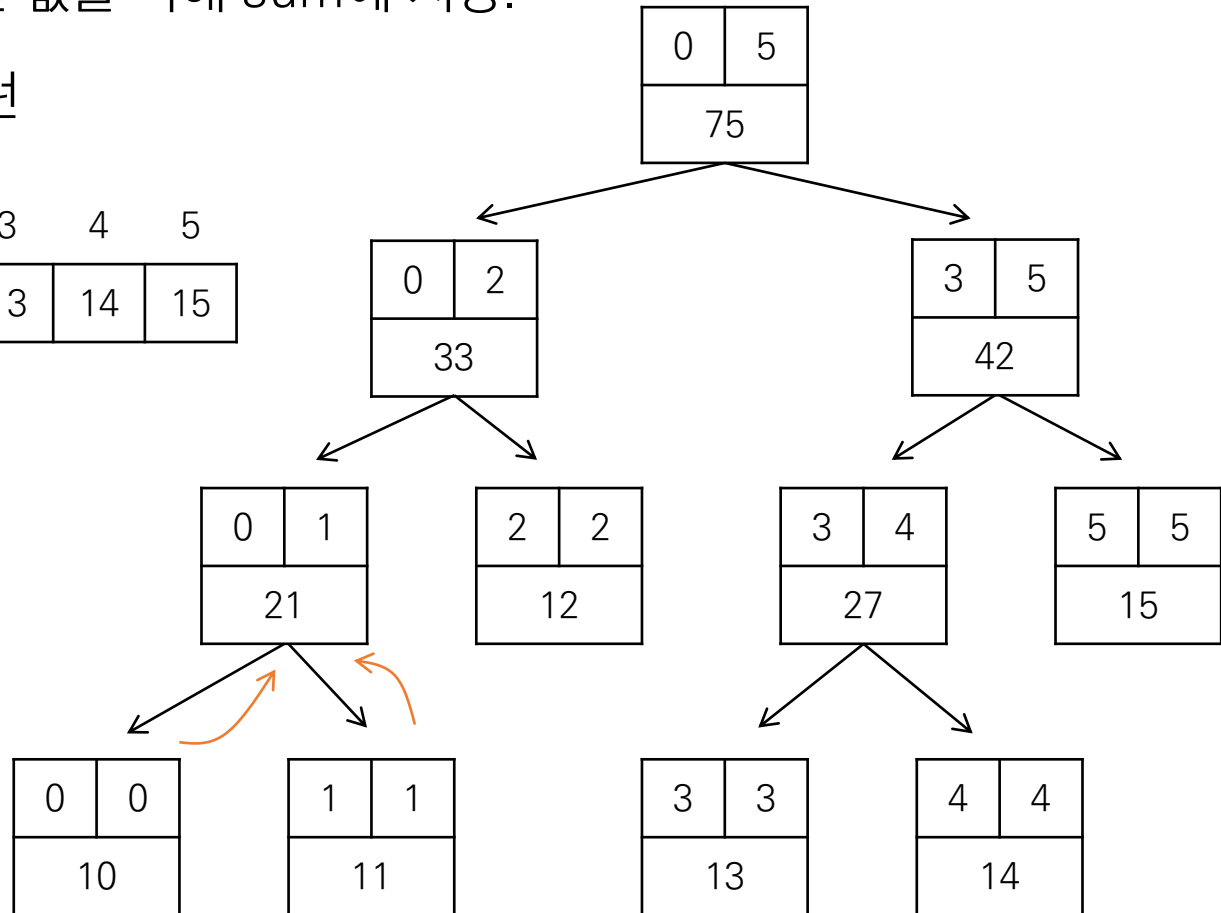
0	1	2	3	4	5
10	11	12	13	14	15



■ 후위 순회로 자식 노드의 구간 합을 계산.

- 양쪽의 리턴 값을 더해 sum에 저장.
- sum을 리턴

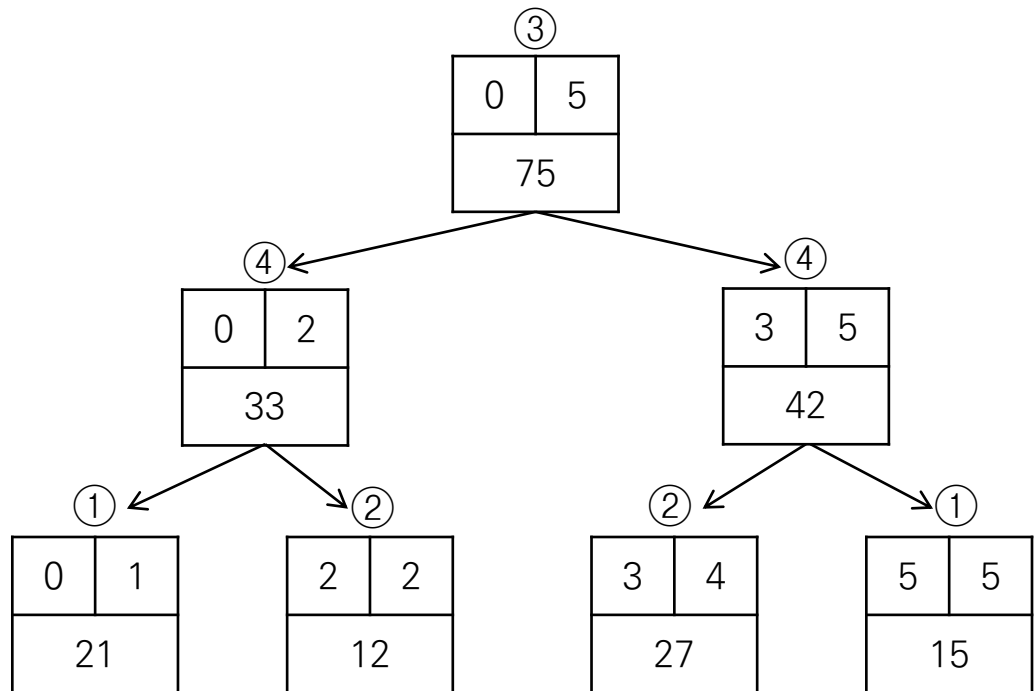
0	1	2	3	4	5
10	11	12	13	14	15



■ L-R 구간의 합을 구하기 위한 탐색 과정.

- ① if ($R < s$ || $e < L$) return 0 : 범위 밖.
- ② if ($L \leq s$ && $e \leq R$) return sum : 유효한 구간
- ③ if ($s \leq L$ && $R \leq e$) 계속 탐색 후 리턴 값끼리 더함.
- ④ 일부 구간 겹침 : 계속 탐색 후 리턴 값끼리 더함.

L			R		
0	1	2	3	4	5
10	11	12	13	14	15



- i 값을 m 으로 변경.

- $\text{diff} = m - p[i]$

```

change( n, i, diff)
  if (i < s || e < i) // ①
    return
  else if (s == i && i == e) // ③
    sum += diff
    return
  else if (s <= i && i <= e) // ②
    sum += diff
    change(n*2, i, diff)
    change(n*2+1, i, diff)
    return
  
```

