

펜윅 트리

최백준 choi@startlink.io

누적합

누적합

Prefix Sum

- 수열 $A[1], A[2], \dots, A[N]$ 이 있을 때
- $A[i] + \dots + A[j]$ 를 구하는 문제
- $S[i] = A[1] + A[2] + \dots + A[i]$

누적합

Prefix Sum

- $A[i] + \dots + A[j]$ 를 구하는 문제
- $S[j] = A[1] + A[2] + \dots + A[i-1] + A[i] + \dots + A[j]$
- $S[i-1] = A[1] + A[2] + \dots + A[i-1]$
- $S[j] - S[i-1] = A[i] + \dots + A[j]$

구간 합 구하기 4

5

<https://www.acmicpc.net/problem/11659>

- 수 N 개가 주어졌을 때, i 번째 수부터 j 번째 수까지 합을 구하는 문제

구간 합 구하기 4

<https://www.acmicpc.net/problem/11659>

- 소스: <http://codeplus.codes/a14c5adf22764680872d18611ef53ddc>

수들의 합 4

<https://www.acmicpc.net/problem/2015>

- 크기가 N 인 배열 A 이 주어졌을 때, 부분합은 $1 \leq i \leq j \leq N$ 인 정수 i, j 에 대해 $A[i] + \cdots + A[j]$
- $N \times (N+1)/2$ 개의 부분합 중에서 합이 K 인 것의 개수를 찾는 문제
- $1 \leq N \leq 200,000$
- $|K| \leq 2,000,000,000$
- $-10,000 \leq A[i] \leq 10,000$

수들의 합 4

<https://www.acmicpc.net/problem/2015>

- $A[i] + \dots + A[j] == K$ 인 (i, j) 쌍의 개수를 찾는 문제이다.
- $O(N^2)$ 의 방법이 가능하지만, $N \leq 200,000$ 이다.

수들의 합 4

<https://www.acmicpc.net/problem/2015>

- $A[i] + \dots + A[j] == K$ 인 (i, j) 쌍의 개수를 찾는 문제이다.
- $S[j] - S[i-1] == K$ 인 (i, j) 쌍의 개수를 찾는 문제와 같다.
- 각각의 j 에 대해서, $i-1$ 의 개수를 찾으면 된다.

수들의 합 4

<https://www.acmicpc.net/problem/2015>

- $K = 0$ 인 경우
- $S[j] - S[i-1] == K$ 인 (i, j) 쌍의 개수를 찾는 문제
- $j = 1$: $S[i-1] = S[j] - K = 2$ (0개)
- $j = 2$: $S[i-1] = S[j] - K = 0$ (1개)
- $j = 3$: $S[i-1] = S[j] - K = 2$ (1개)
- $j = 4$: $S[i-1] = S[j] - K = 0$ (2개)

i	0	1	2	3	4
A[i]		2	-2	2	-2
S[i]	0	2	0	2	0

수들의 합 4

<https://www.acmicpc.net/problem/2015>

- $\text{cnt}[k] = S[i] == k$ 인 i 의 개수를 저장하면 된다.
- $S[i] < 0$ 이 될 수 있기 때문에, 배열을 사용할 수 없다.
- map을 사용한다.

i	0	1	2	3	4
A[i]		2	-2	2	-2
S[i]	0	2	0	2	0

수들의 합 4

<https://www.acmicpc.net/problem/2015>

- 소스: <http://codeplus.codes/09059c20ec554f08aac14afee3b39c30>

나머지 합

<https://www.acmicpc.net/problem/10986>

- 수 N 개 $A[1], A[2], \dots, A[N]$ 이 주어진다.
- 연속된 부분 구간의 합이 M 으로 나누어 떨어지는 구간의 개수를 구하는 문제
- 즉, $A[i] + \dots + A[j]$ ($i \leq j$)의 합이 M 으로 나누어 떨어지는 (i, j) 쌍의 개수를 구해야 한다.

나머지 합

<https://www.acmicpc.net/problem/10986>

- $S[i] = A[1] + \dots + A[i]$ 라고 하자
- $A[i] + \dots + A[j] = S[j] - S[i-1]$
- $(A[i] + \dots + A[j]) \% M = (S[j] - S[i-1]) \% M$
- $(A[i] + \dots + A[j]) \% M == 0$ 인 것의 개수를 구해야 한다
- $(S[j] - S[i-1]) \% M == 0$ 와 같다
- 나눈 나머지가 0이 되려면
- $S[j] \% M == S[i-1] \% M$ 이 되어야 한다

나머지 합

<https://www.acmicpc.net/problem/10986>

- 이 문제는
- $S[j] \% M == S[i-1] \% M$ 이 되어야 한다
- 를 만족하는 (i, j) 쌍의 개수를 구하는 문제가 된다.
- $\text{cnt}[k]$ 를 $S[i] \% M == k$ 인 i 의 개수라고 하면
- $0 \leq k < M$ 인 k 에 대해서
- $\text{cnt}[k] * (\text{cnt}[k] - 1) / 2$ 의 합을 구하면 된다.

나머지 합

<https://www.acmicpc.net/problem/10986>

- 소스: <http://codeplus.codes/f0d7a3e127b846b598ca2d0e4702a895>

2차원 누적합

구간 합 구하기 5

18

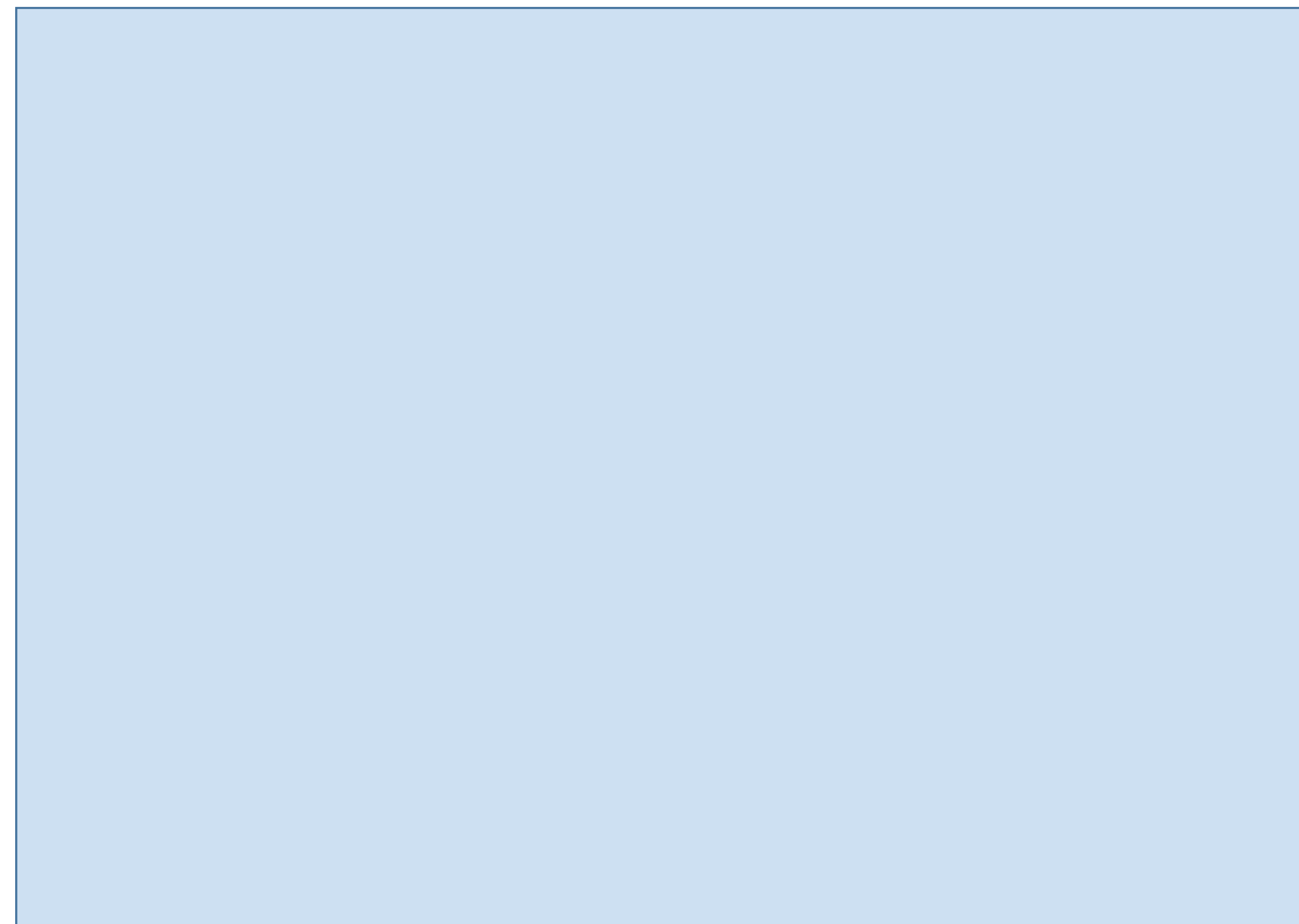
<https://www.acmicpc.net/problem/11660>

- 2차원 배열에서 왼쪽 윗 칸이 $(x1, y1)$, 오른쪽 아랫 칸이 $(x2, y2)$ 인 직사각형에 들어있는 수의 합을 구하는 문제

구간 합 구하기 5

<https://www.acmicpc.net/problem/11660>

- 합을 효율적으로 구하는 방법
- $S[i][j] = (1, 1) \sim (i, j)$ 까지 합
- $S[i][j] = S[i-1][j] + S[i][j-1] - S[i-1][j-1] + A[i][j]$

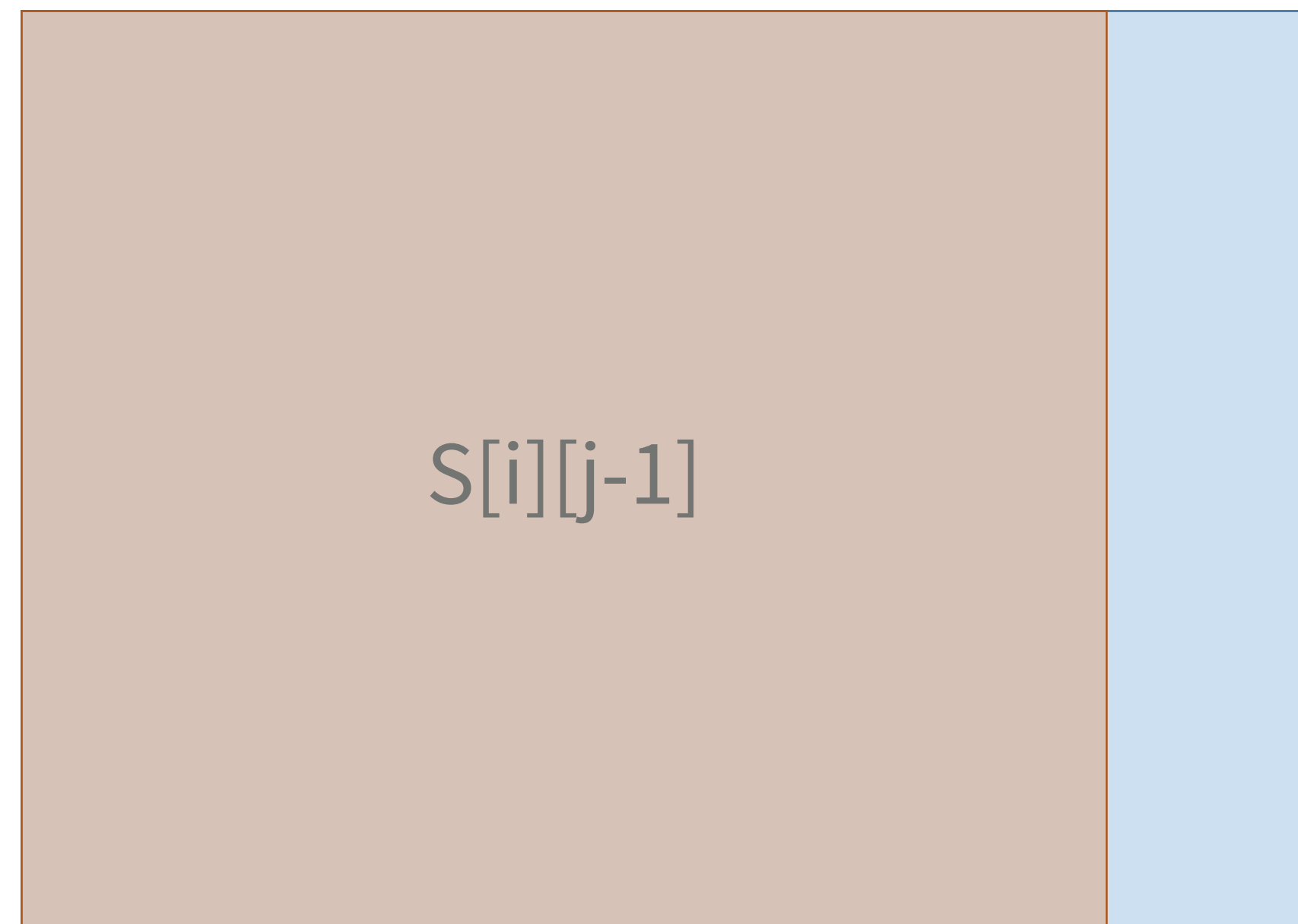


구간 합 구하기 5

20

<https://www.acmicpc.net/problem/11660>

- 합을 효율적으로 구하는 방법
- $S[i][j] = (1, 1) \sim (i, j)$ 까지 합
- $S[i][j] = S[i-1][j] + S[i][j-1] - S[i-1][j-1] + A[i][j]$

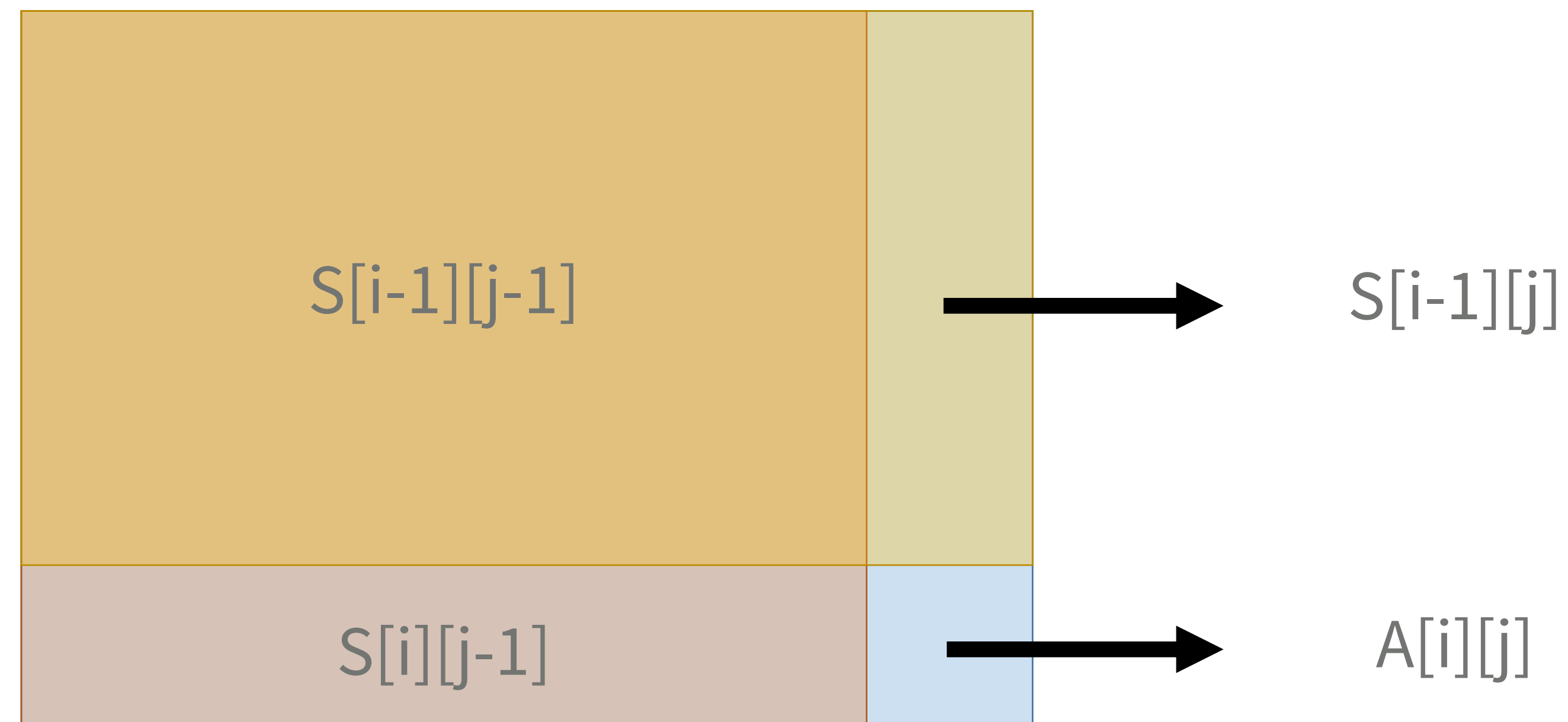


구간 합 구하기 5

21

<https://www.acmicpc.net/problem/11660>

- 합을 효율적으로 구하는 방법
- $S[i][j] = (1, 1) \sim (i, j)$ 까지 합
- $S[i][j] = S[i-1][j] + S[i][j-1] - S[i-1][j-1] + A[i][j]$



구간 합 구하기 5

22

<https://www.acmicpc.net/problem/11660>

- $(a,b) \sim (c,d)$ 합 구하기

	b		d	
a				
c				

구간 합 구하기 5

23

<https://www.acmicpc.net/problem/11660>

- $(a,b) \sim (c,d)$ 합 구하기

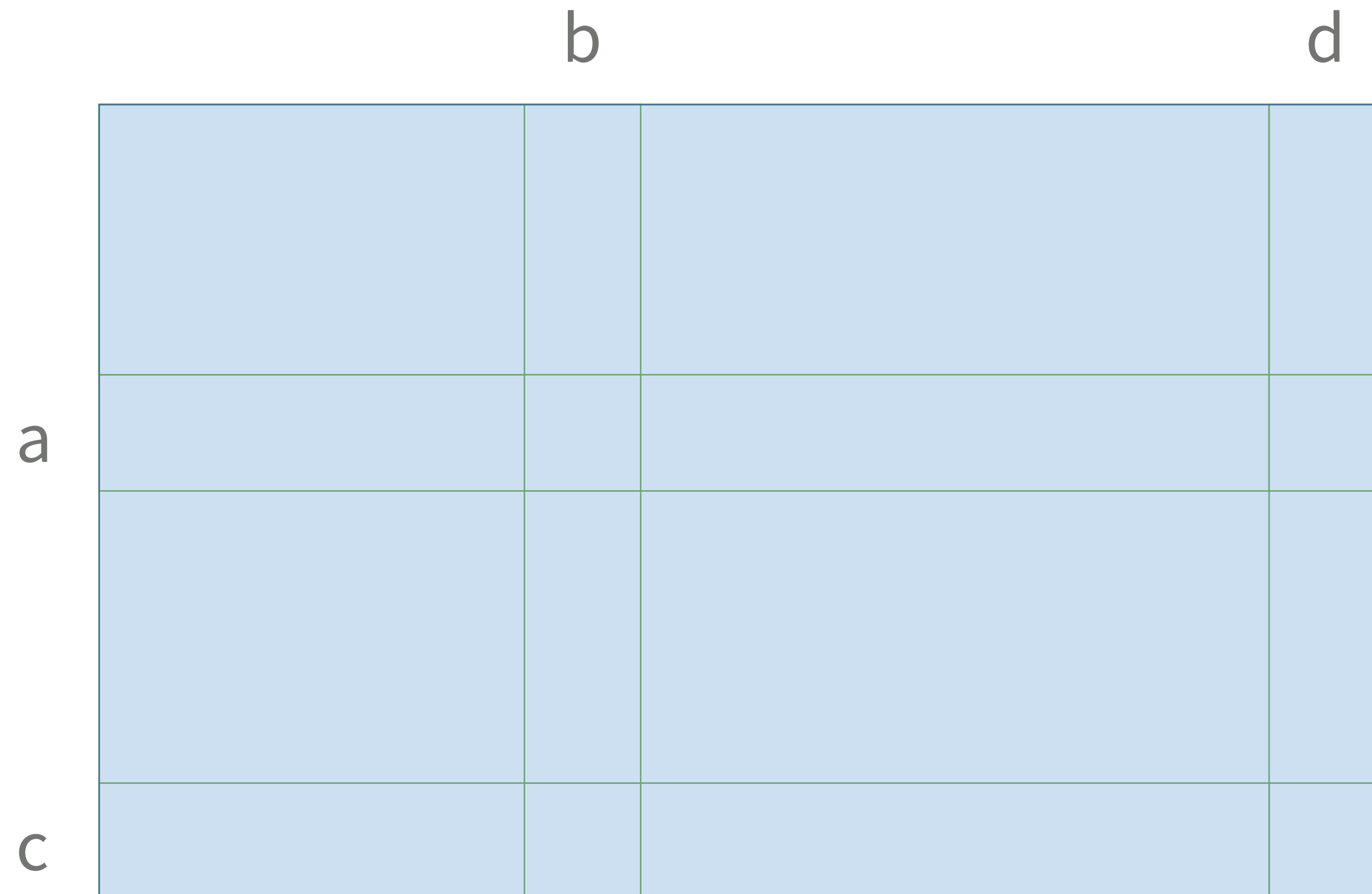
		b		d
a				
c				

구간 합 구하기 5

24

<https://www.acmicpc.net/problem/11660>

- $S[c][d]$

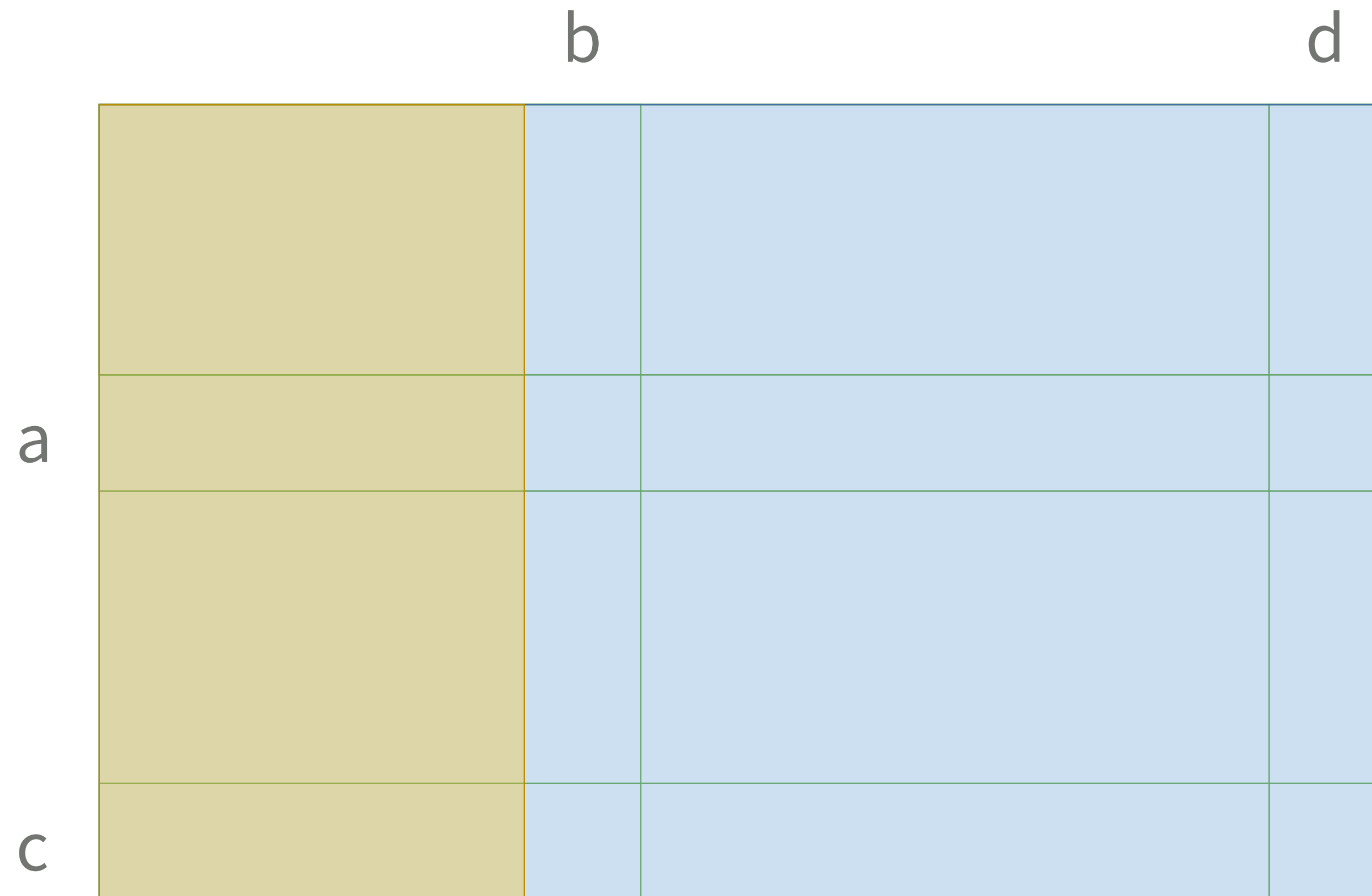


구간 합 구하기 5

25

<https://www.acmicpc.net/problem/11660>

- $S[c][d] - S[c][b-1]$

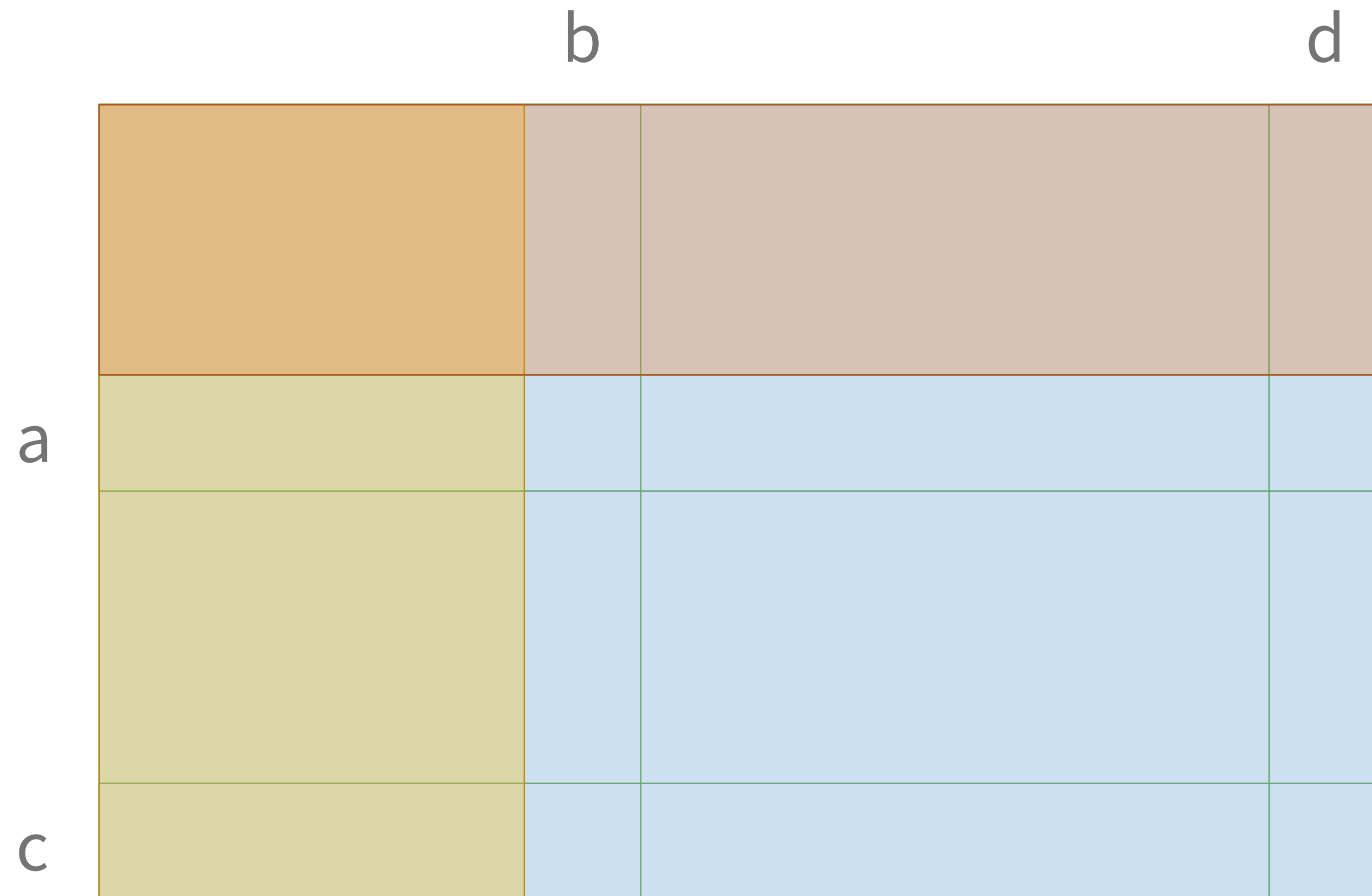


구간 합 구하기 5

26

<https://www.acmicpc.net/problem/11660>

- $S[c][d] - S[c][b-1] - S[a-1][d]$

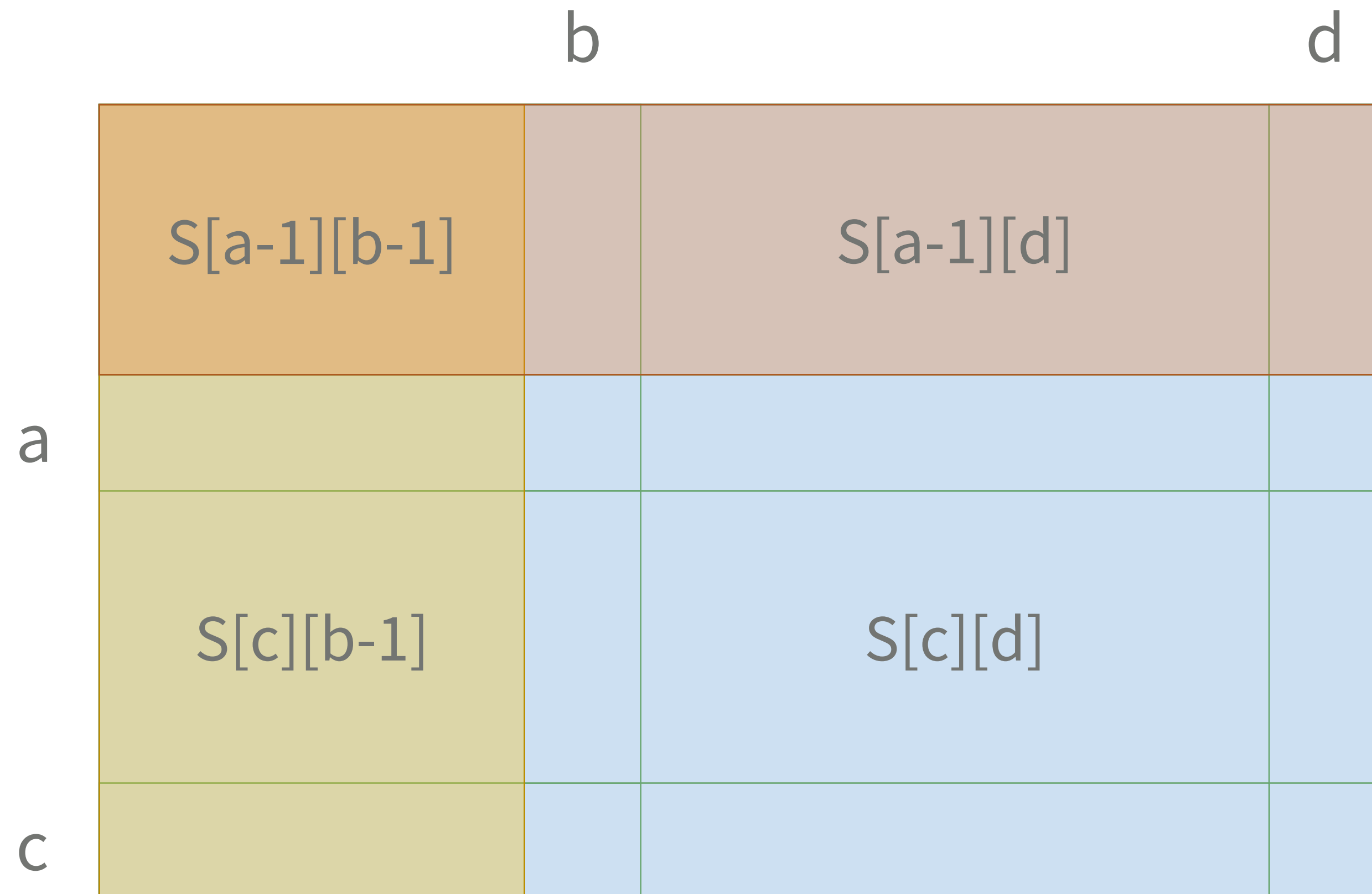


구간 합 구하기 5

27

<https://www.acmicpc.net/problem/11660>

- $S[c][d] - S[c][b-1] - S[a-1][d] + S[a-1][b-1]$



구간 합 구하기 5

28

<https://www.acmicpc.net/problem/11660>

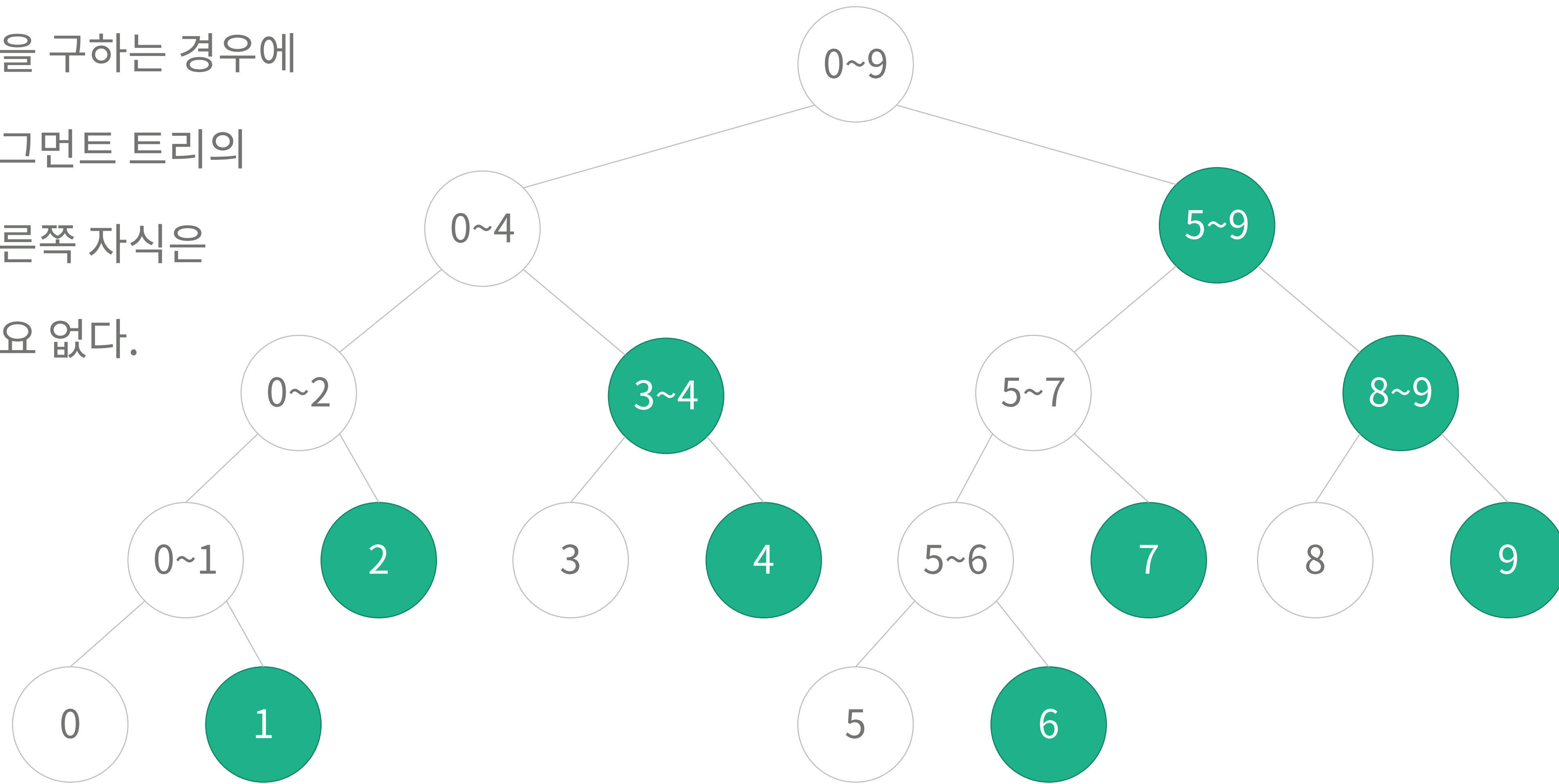
- 소스: <http://codeplus.codes/2a423621e6ee4a05aa24002afa123f68>

Fenwick Tree

Fenwick Tree

Fenwick Tree (BIT)

- 합을 구하는 경우에
- 세그먼트 트리의
- 오른쪽 자식은
- 필요 없다.



Fenwick Tree

Fenwick Tree (BIT)

- $L(i)$: i 를 2진수로 나타냈을 때, 가장 마지막 1이 나타내는 값
- $3 = 11_2$
- $5 = 101_2$
- $6 = 110_2$
- $8 = 1000_2$
- $9 = 1001_2$
- $10 = 1010_2$
- $11 = 1011_2$
- $12 = 1100_2$
- $16 = 10000_2$

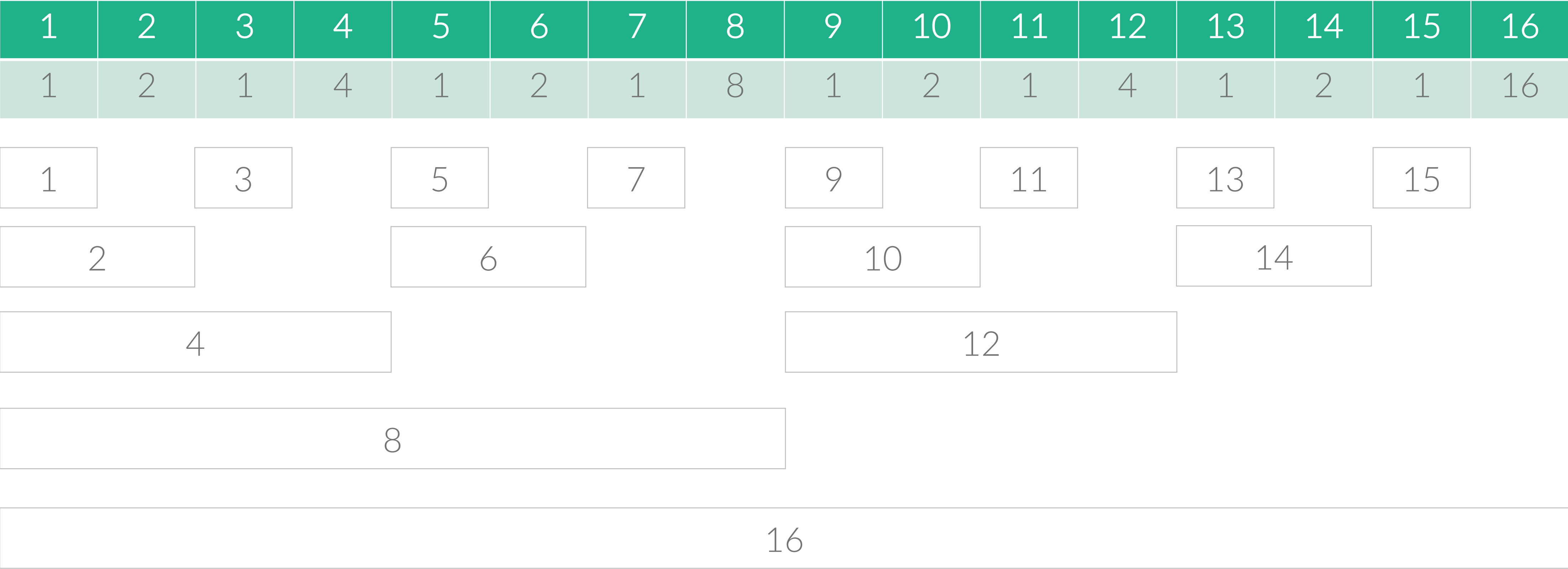
Fenwick Tree

Fenwick Tree (BIT)

- $-\text{num} = \sim\text{num} + 1$
- $\text{num} = 100110101110101100000000000000$
- $\sim\text{num} = 0110010100010100111111111111$
- $-\text{num} = 011001010001010100000000000000$
- $\text{num} \& -\text{num} = 000000000000000000000010000000000000$

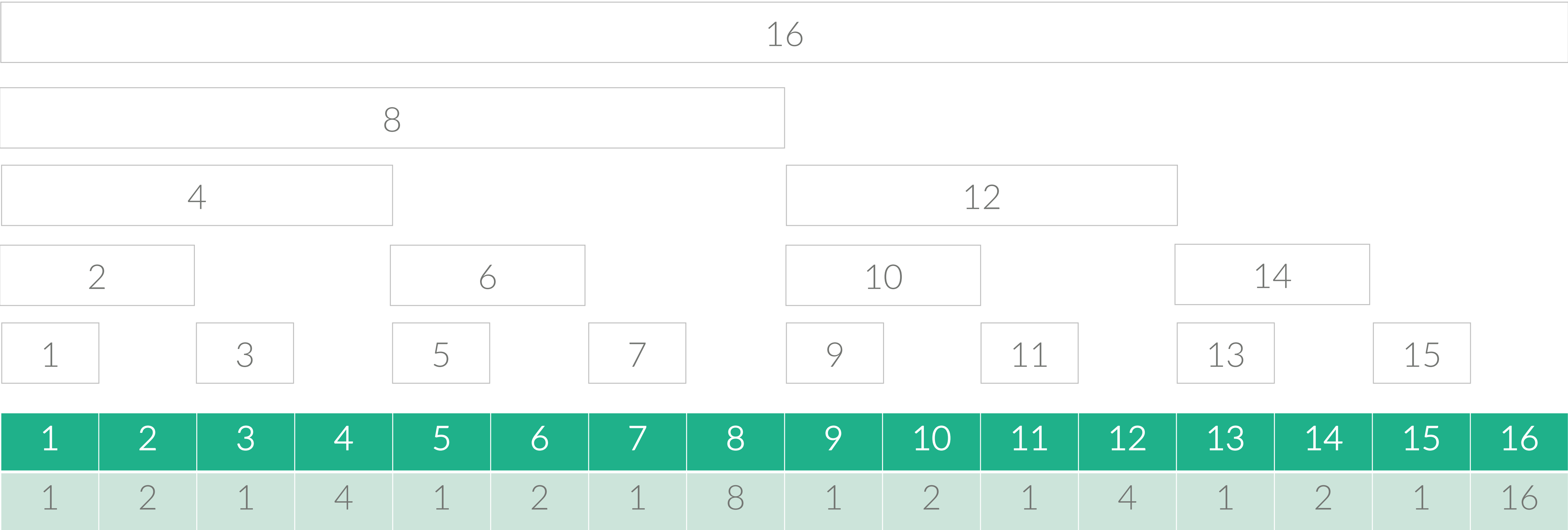
Fenwick Tree

Fenwick Tree (BIT)



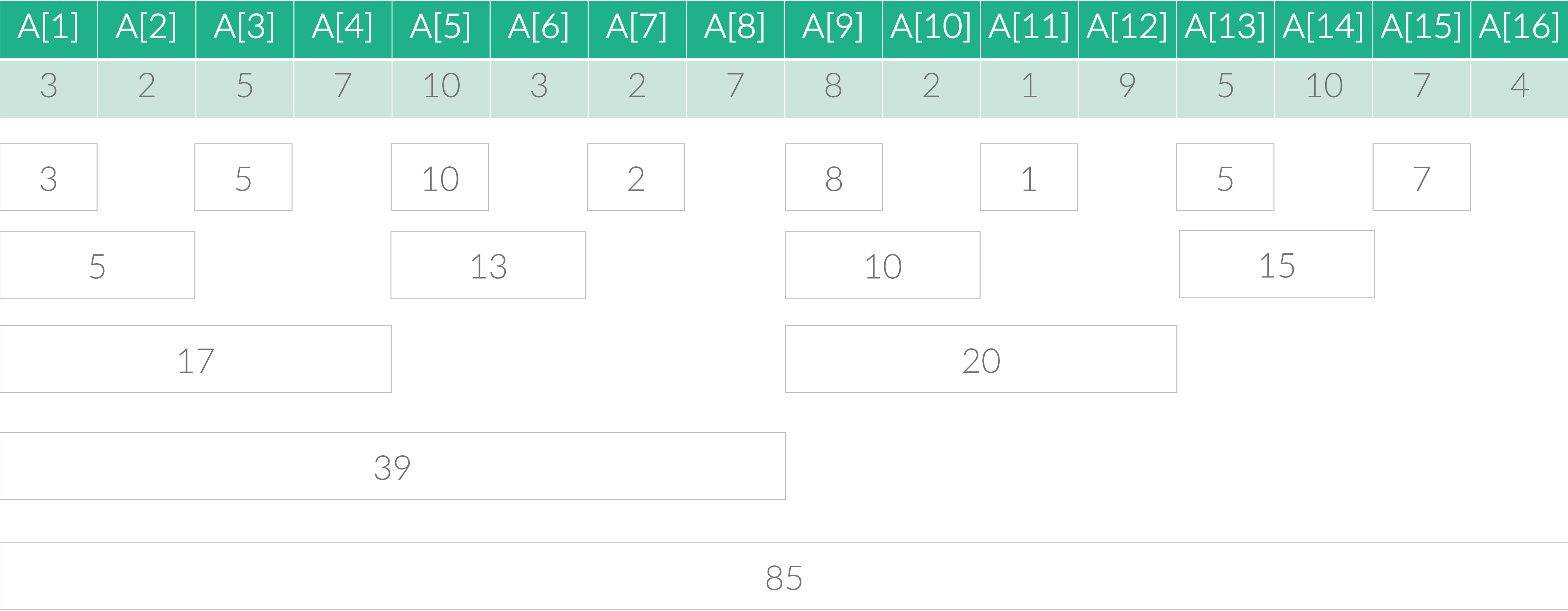
Fenwick Tree

Fenwick Tree (BIT)



Fenwick Tree

Fenwick Tree (BIT)



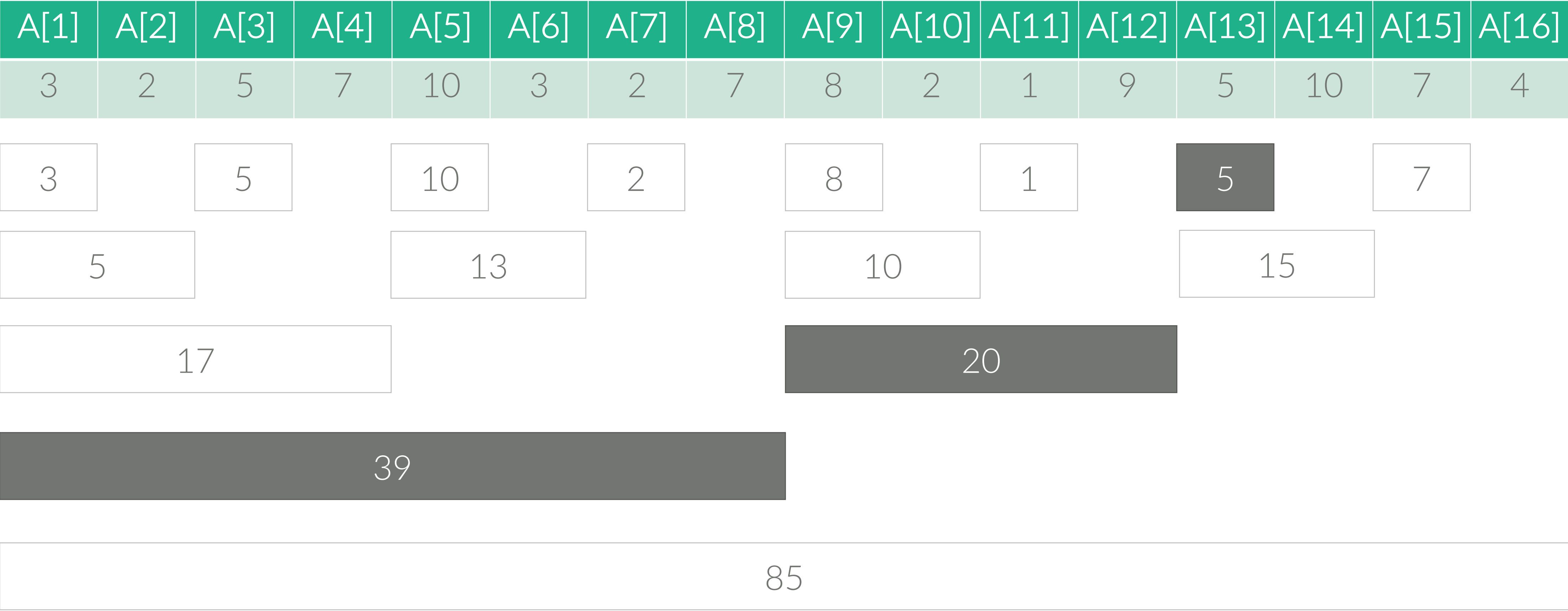
Fenwick Tree

Fenwick Tree (BIT)

- $A[1] + \dots + A[13]$ 을 구하려면
- $13 = 1101_2$
- $\text{tree}[1101_2] + \text{tree}[1100_2] + \text{tree}[1000_2]$

Fenwick Tree

Fenwick Tree (BIT)



Fenwick Tree

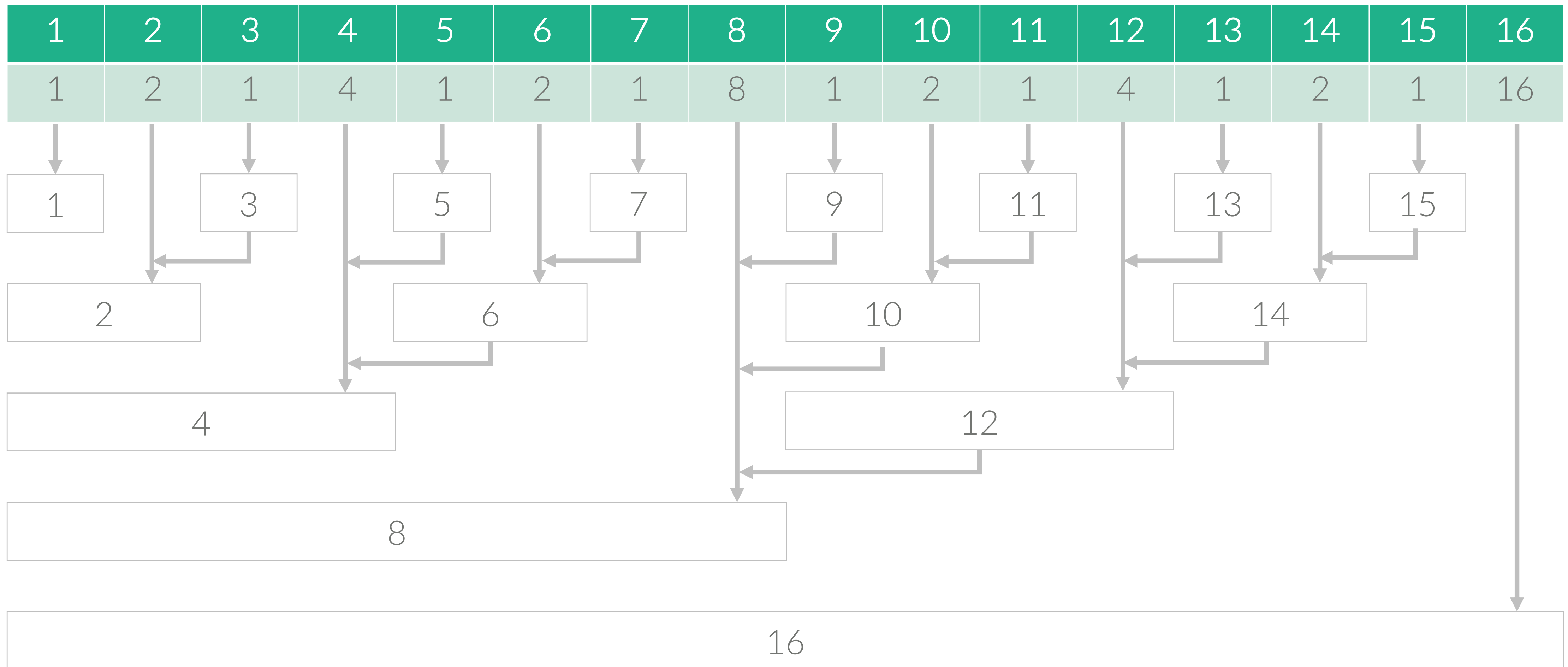
Fenwick Tree (BIT)

```
int sum(int i) {  
    int ans = 0;  
    while (i > 0) {  
        ans += tree[i];  
        i -= (i & -i);  
    }  
    return ans;  
}
```

Fenwick Tree

39

Fenwick Tree (BIT)



Fenwick Tree

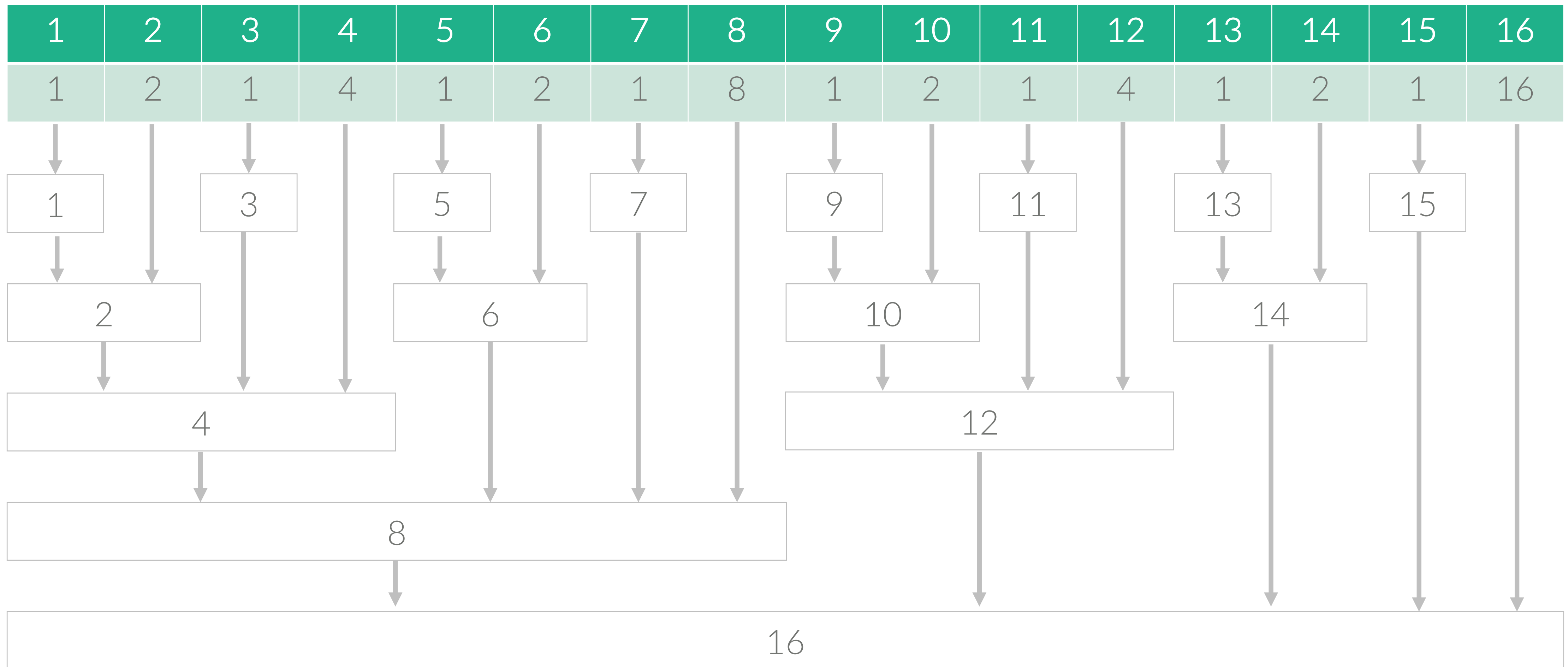
Fenwick Tree (BIT)

```
int update(int i, int num) {  
    while (i <= n) {  
        tree[i] += num;  
        i += (i & -i);  
    }  
}
```


Fenwick Tree

41

Fenwick Tree (BIT)



구간 합 구하기

42

<https://www.acmicpc.net/problem/2042>

- 소스 1: <http://codeplus.codes/092a5402b87f4cfa807a9e9701e4339f>
- 소스 2: <http://codeplus.codes/bf1ccee75f4c47f6a6469edaeb8a8bc9>

2D Fenwick Tree

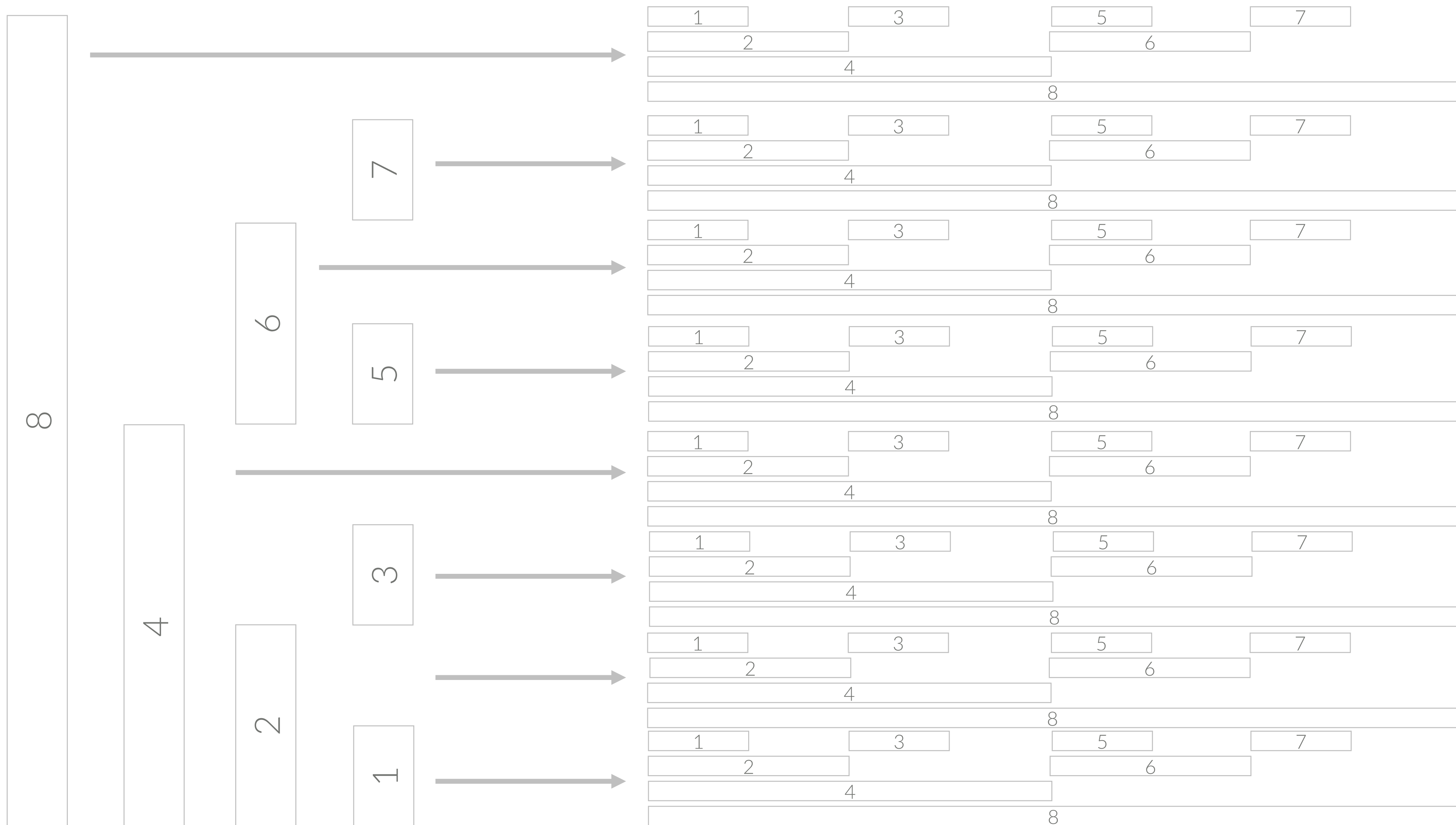
2D Fenwick Tree (BIT)

- 1차원을 2차원으로 확장해서 만들 수 있다.
- 행에 대해서 그리고 열에 대해서 트리를 만들면 된다

2D Fenwick Tree

44

2D Fenwick Tree (BIT)



2D Fenwick Tree

45

2D Fenwick Tree (BIT)

```
void update(int x, int y, int val) {  
    for (int i=x; i<=n; i+=i&-i) {  
        for (int j=y; j<=n; j+=j&-j) {  
            tree[i][j] += val;  
        }  
    }  
}
```

2D Fenwick Tree

2D Fenwick Tree (BIT)

```
int sum(int x, int y) {  
    int ans = 0;  
    for (int i=x; i>0; i-=i&-i) {  
        for (int j=y; j>0; j-=j&-j) {  
            ans += tree[i][j];  
        }  
    }  
    return ans;  
}
```

구간 합 구하기 3

47

<https://www.acmicpc.net/problem/11658>

- 소스: <http://codeplus.codes/1079573df1254a9ebefccf7caf4bfbbc>