





Outline

- Discussion

- Problem

- Fenwick Tree

- Complexity

- 2D Fenwick Tree

- Implementation Detail

- Sum Query

- Update Query



Problem

- Suppose, there are several boxes
Labeled from 1 to N
- We can
Add N marble(s) into ith box
We say box ith has frequency N
- Now, we want to know
Total number of marbles in boxes i to j

Fenwick Tree

- Each element in the array stores prefix sum of consecutive list of boxes.
- Range of boxes that is stored is related to “binary value” of the index.




Sum Query

The diagram illustrates the conversion of decimal numbers to binary and then to Gray code. It consists of four tables labeled A, T, A, and T, each with 15 rows. The first table (A) shows decimal values and their 4-bit binary representations. The second table (T) shows the same decimal values and binary representations, but with the first bit of each binary value highlighted in red. The third table (A) shows the decimal values and their 4-bit binary representations, with the first bit of each binary value highlighted in red. The fourth table (T) shows the decimal values and their 4-bit binary representations, with the first bit of each binary value highlighted in red. A red arrow points from the first table to the second, and another red arrow points from the third table to the fourth. A red arrow also points from the second table to the fourth, indicating a direct conversion from binary to Gray code.

A	T	A	T
5	5	5	5
2	2	2	2
9	9	9	9
-3	-3	-3	-3
5	5	5	5
20	20	20	20
10	10	10	10
-7	-7	-7	-7
2	2	2	2
3	3	3	3
-4	-4	-4	-4
0	0	0	0
-2	-2	-2	-2
15	15	15	15
5	5	5	5

Update Prefix Sum Array

A				T			
5	1	00001]	5			
2	2	00010		7			
9	3	00011		9			
-3	4	00100		13			
5	5	00101]	5			
20	6	00110		25			
10	7	00111]	10			
-7	8	01000		41			
2	9	01001]	2			
3	10	01010		5			
-4	11	01011]	-4			
0	12	01100		1			
-2	13	01101]	-2			
15	14	01110		13			
5	15	01111]	5			



A				T			
5	1	00001]	5			
2	2	00010		7			
9	3	00011		9			
7	4	00100		23			
5	5	00101]	5			
20	6	00110		25			
10	7	00111]	10			
-7	8	01000		51			
2	9	01001]	2			
3	10	01010		5			
-4	11	01011]	-4			
0	12	01100		1			
-2	13	01101]	-2			
15	14	01110		13			
5	15	01111]	5			

add(4, 10)

$$T[00100] = 13 + 10 = 23$$

$$T[01000] = 41 + 10 = 51$$



Complexity

getSum	$O(\lg n)$
update	$O(\lg n)$



2D Fenwick Tree

- Multi-dimensional data structure.
- Array of 1D BITs.
- Suppose we have:

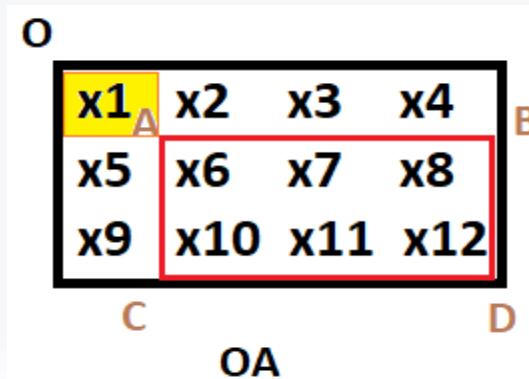
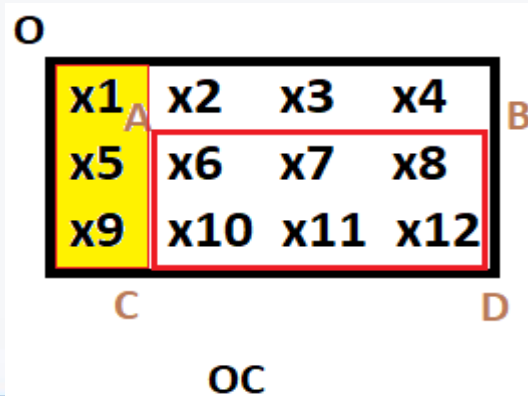
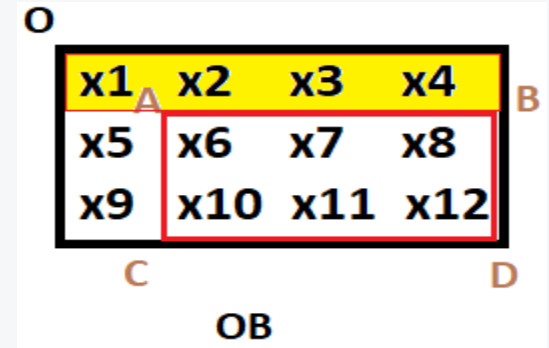
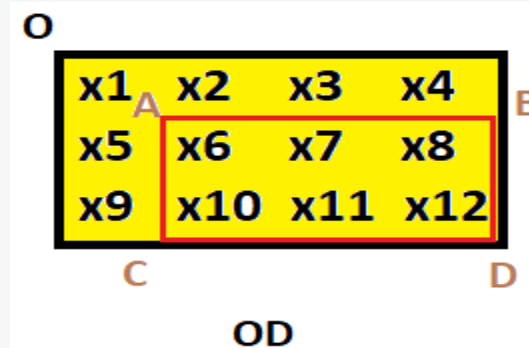
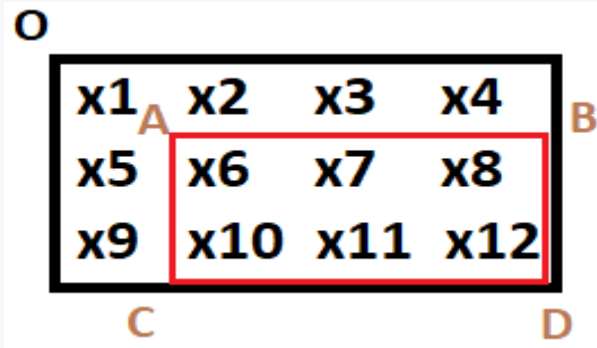
x1	x2	x3	x4
x5	x6	x7	x8
x9	x10	x11	x12

To Find:

x1	x2	x3	x4
x5	x6	x7	x8
x9	x10	x11	x12



Principle of Inclusion Exclusion

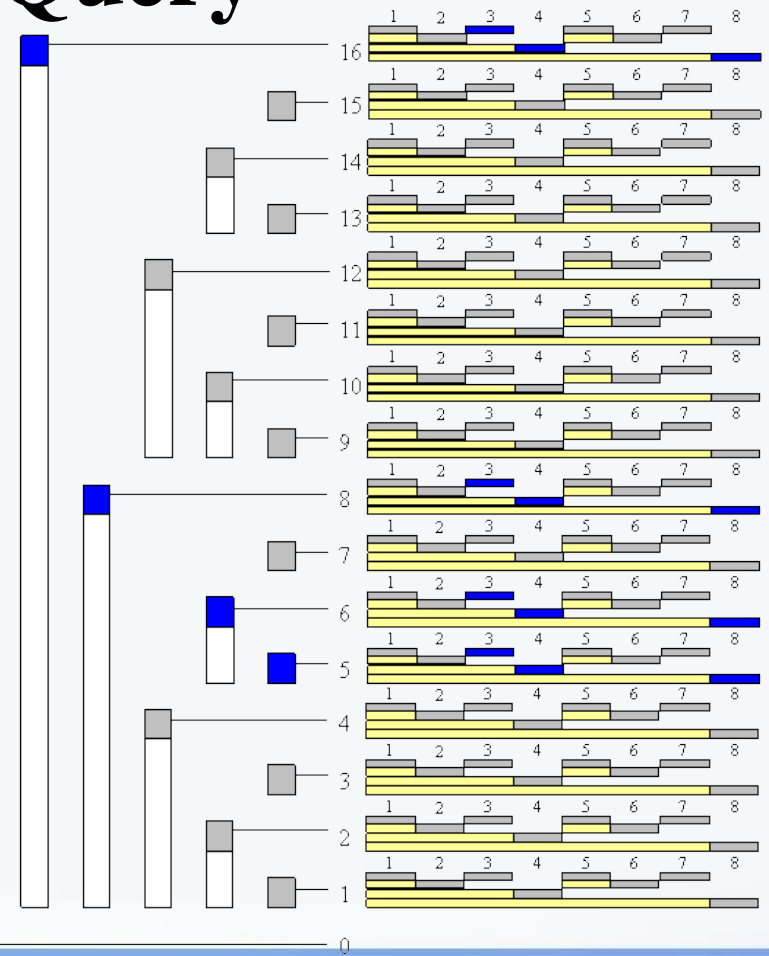


$$\begin{aligned} &\text{Sum(OD)} - \text{Sum(OB)} - \\ &\text{Sum(OC)} + \text{Sum(OA)} \\ &= \\ &\text{Sum of Area} \end{aligned}$$

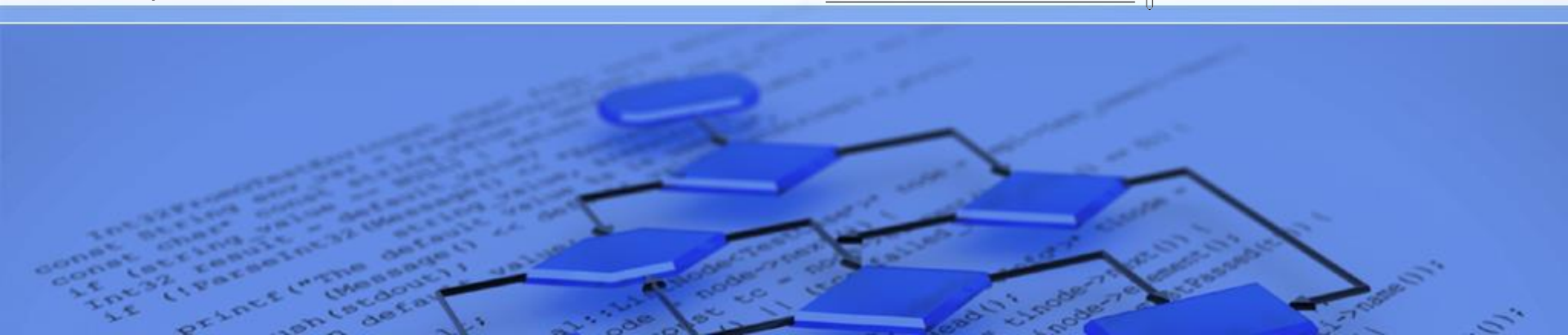


Sum/Update Query

- This is 2D BIT of size (16 x 8).
- Updating index (5 , 3).



Source: topcoder.com



Complexity

getSum	$O(\lg MN)$
update	$O(\lg MN)$



Thank You

