

Prefix Sums Problem Solving & Difference Arrays

- Harsh Gupta

Goal

- Solving 2 problems on Prefix Sums.
- Learn about Difference Arrays

Problem 1

Given an array of N elements, answer the following Q queries

- Query: L, R
- $\text{Result} = A[L] + 2 * A[L + 1] + 3 * A[L + 2] + \dots (R - L + 1) * A[R]$

$$A = [2 \quad \underline{4} \quad 3 \quad 6 \quad 9]$$

$$(2 \quad 4) \leftarrow$$

$$| \quad \underline{A[2]} + 2 \times \underline{A[3]} + 3 \times A[4]$$

$$(\underline{1}, \underline{4})$$

$$\underline{A[1]} + \cancel{2} \times \underline{A[2]} + \underline{3 \times A[3]} + \underline{4 \times A[4]}$$

L R L

$$1 A_L + 2 \times A_{L+1} + \underbrace{3}_{\uparrow} \times \underbrace{A_{L+2}}_{\text{yellow circle}} + \dots \quad \underline{i = L+2}$$

$$\binom{i-L+1}{L-L+1} = \binom{i-L+1}{1}$$

$$\binom{i-L+1}{1} A_i$$

\downarrow

$$A_L \quad A_{L+1} \quad 3 \times A_{L+2} \quad A_{L+3}$$

$$A_i \quad \binom{i-L+1}{1}$$

$$\binom{i-L+1}{L+2-L+1} A_i$$

$$i-L+1$$

$$L+2-L+1$$

$$\binom{i-L+1}{L+2-L+1}$$

$$\binom{i-L+1}{L+2-L+1}$$

Q

$$\sum_{i=L}^R A_i \times (i-L+1)$$

Diagram illustrating the summation with indices L and R . A yellow arrow points to the summation symbol, and a curved arrow indicates the index i starting from L and increasing to R .

$$\leq a + b$$

$$= \leq a + \leq b$$

$$\sum_{i=L}^R i \times A_i - (L-1) \times A_i$$

Diagram illustrating the transformation of the summation. The terms $i \times A_i$ and $(L-1) \times A_i$ are underlined in yellow. Arrows point from these terms to the corresponding parts of the next equation.

A_1, A_2, A_3, A_n

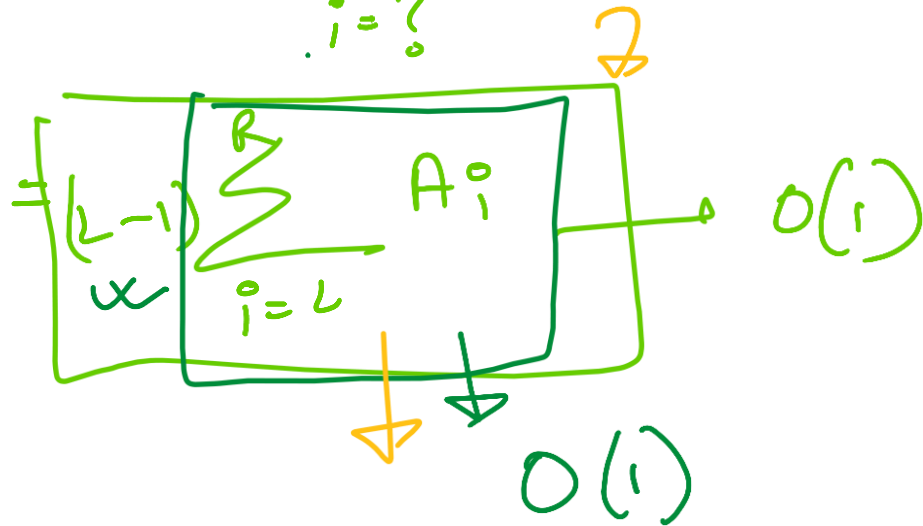
$1 \times A_1, 2 \times A_2, 3 \times A_3, \dots$

$$= \sum_{i=L}^R (i \times A_i) - \sum_{i=L}^R (L-1) A_i$$

Diagram illustrating the final step of the transformation. The term $i \times A_i$ is circled in blue. Wavy lines under the summations are labeled with a question mark and $O(1)$.

$$\sum_{i=L}^R (L-1) A_i$$

$i = ?$



$$i \times A_i$$

$$A_i$$

$N, O, x/2s$

$A_1 \quad A_2 \quad A_3 \quad A_4 \quad A_5 \quad A_6 \quad A_7$

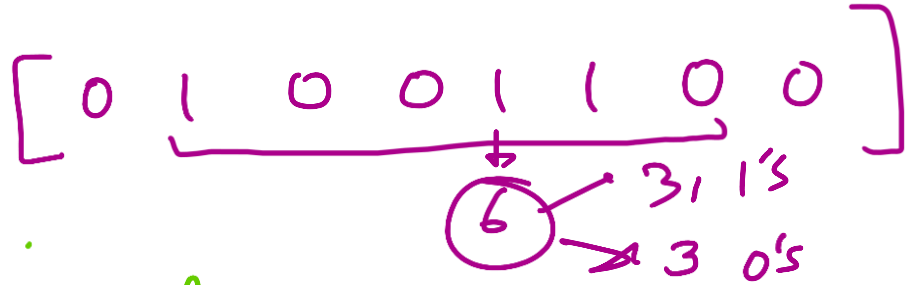
$1 \times A_1 \quad 2A_2 \quad 3A_3 \quad 4A_4 \quad 5A_5 \quad 6A_6 \quad 7A_7$

calculate prefix sum.

$$\underbrace{L=3 \quad R=5}$$

$$\boxed{P[5] - P[2]} \quad \boxed{i \times A_i}$$

Problem 2: Contiguous Array



map with prefix sum

0 \rightarrow -1

1 \rightarrow +1

I will calculate

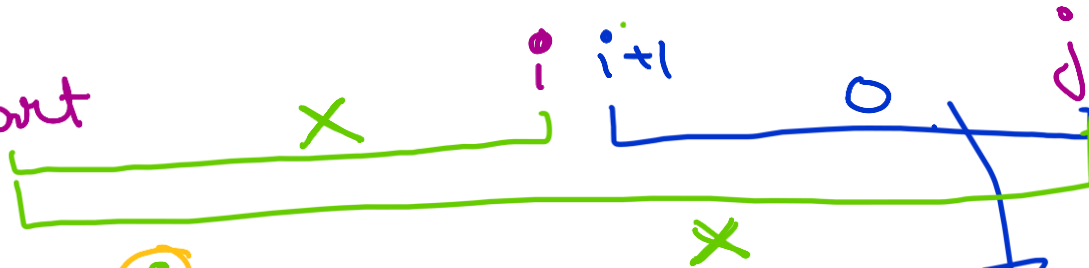
summation of prefixes.

summation = 0

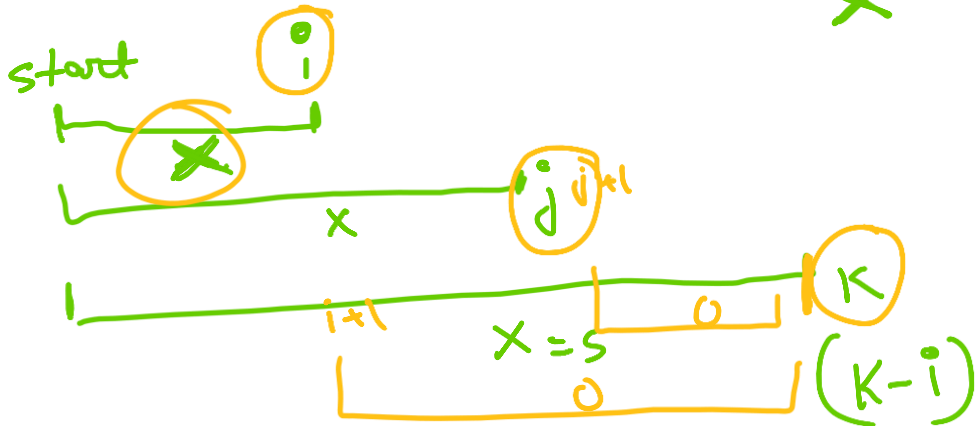
\rightarrow 1's and 0's equal

start

start



start

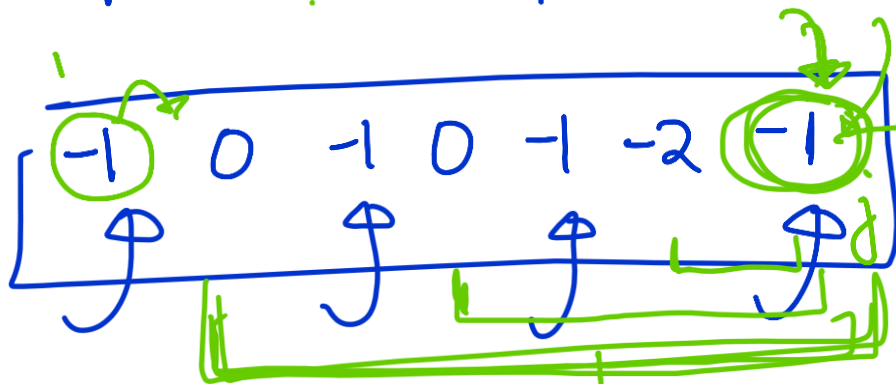


equal #
of 0's and 1's

$(K - i + 1)$

0 1 0 1 0 0 1

-1 +1 -1 +1 -1 -1 +1



map/hashing

$mp[-i] = i$

0's & 1's

~~sum~~ $mp[0] = -1$



$$3 - mp[0]$$

$$3 - (-1) = \underline{\underline{4}}$$

Difference Arrays



A difference array can be used to perform multiple range update where we need to find the final state of the arrays only after performing all the queries.

Difference array helps us achieve the above in $O(N)$ time total time and space.

We can process every range update in $O(1)$

When we need to print our final answer we perform an $O(N)$ computation.

Example

Given an array with all 0s initially perform the following Q queries on it.

In the i th query you will be given 3 integers: L_i , R_i , X_i . You need to add X_i to all the values in the array from index L_i to R_i .

After performing all the queries print the final state of the array.

	0	1	2	3	4	5	6	7
A →	0	3	3	3	3	3	0	0
	.		9	9	9	6		

$\underline{Q} \rightarrow \underline{L_i} \quad \underline{R_i} \quad \underline{X_i}$
(L_i, R_i - 0 based indexing)

• $\underline{1} \quad \underline{4} \quad \underline{3}$

• $\underline{2} \quad \underline{5} \quad \underline{6}$

•
•
•
multiple such queries.

0 1 2 3 4 5 6 7 8
 0 0 0 0 0 0 0 0 0

0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0

Diffⁿ Array



by how much
 every index
 value will
 change.

+3 +2 -3 -2
 0 3 5 5 5 2 0 0 0
 l, h = 8

l, h

~~⊗~~ (X)

• 1, 4, 3 ⁺³ x
 • 2, 5, 2 ✓

diff[l] += x;
 diff[h+1] -= x;

n+1

+3

0 1 2 3 4 5 6 7
x +3 +3 +3 +3

+5, 2 → 5
1 - 4 → +3

0 +3 0 0 0 -5 0 0
+5 -5

Prefix →

0 3 3 3 3 0 0 0
└──────────┘

Prefix

0 3 8 8 8 5 0 0

Code for Difference Array

```
int n;
cin >> n;
vector<int> arr(n);
for(int i = 0; i < n; i++)
    cin >> arr[i];
vector<int> diff(n, 0);
int q;
cin >> q;
while(q--){
    int l, r, x;
    cin >> l >> r >> x;
    diff[l] += x;
    if(r != n - 1)
        diff[r + 1] -= x;
}
for(int i = 1; i < n; i++)
    diff[i] += diff[i - 1];

for(int i = 0; i < n; i++)
    arr[i] += diff[i];
```

Time and Space Complexity of Diff Arrays

Time Complexity: $O(N + Q)$

Space Complexity: $O(N)$

Question: Can we do it in $O(1)$ space complexity?

YES (you are allowed
to change the vector)
H/W

Problem: Greg and Array