Difference Array

Concepts & Qns

Video - 1

DSA — Easy, M, H.
Sys → Basic, ...
Gym → 5kg/10kg.

Motivation :-

Until or unless you push your limits,
You won't be able to grow.
It's when you challenge yourself
that true strength emerges.

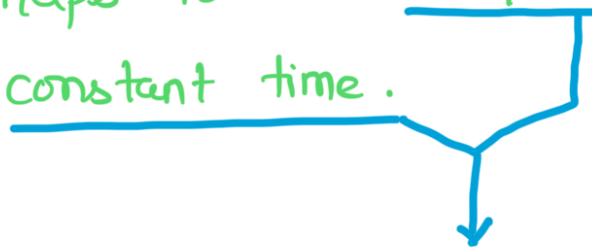
MIK...

INTRODUCTION

What is a "Difference Array" ??

Famous technique use to efficiently apply range updates to an array.

It helps to do multiple updates in constant time.



Ideal for Range Modification Problems.

Let's take an example to understand :-

$n = 5$

nums = {0, 0, 0, 0, 0} ⇐

update Queries = [(1, 3, 2), (2, 4, 3), (0, 2, -2)]

(1, 3, 2) → { 0, 2, 0, 0, -2 }

(2, 4, 3) → { 0, 2, 3, 0, -2 }

(0, 2, -2) → { -2, 2, 3, 2, -2 }

{ -2, 0, 3, 5, 3 } ⇐

[L, R, x] → arr[L] += x

arr[R+1] -= x :→ x+1 < n

Leetcode-370 " Range Addition "

Intuition

Why this works ???

0	1	2	3	4
5	0	0	-5	0

Comsum → 5 5 5 0 0

Job1 = (0, 2, 5)

(0, 0, 0, 0, 0) ←

Code :-

```
for (auto & query : queries) {

    int start = query[0];
    int end   = query[1];
    int x     = query[2];

    arr[start] += x;
    if (end+1 < n)
        arr[end+1] -= x;
}
```

We already have "segment Tree" for that.

Why not use it???

1. Difference array is easier to implement.

2. Space
   - Difference array ~ O(n)
   - Segment Tree ~ O(4*n)

3. Time Complexity for each range update (L, R, x)
   - arr[L] += x
   - arr[R+1] -= x

Difference array = O(1) constant

Then you perform single pass O(n)

to apply all the changes. (consum).

Segment Tree = O(log n)

(4) When to use ???

Difference Array

- Best suited for problems where you need to perform multiple range updates (like adding or subtracting a value over a range).
- you don't need to query the array frequently after updates.
- Highly efficient when you only need to apply a sequence of range operations and get done.

Segment Tree :-

- When you need both range query

range queries

and updates frequently.

(•) Diff. Array

(•) (L, R, x) = num[L] += x

num[R] -= x

(•) Leet-370 → Range Addition :-