

Difference Array Technique

차분 배열 테크닉

Algorithm Session

dev.shinkeonkim@gmail.com

김신건

Koa

어디에 쓸 것인가?

어디에 쓸 것인가?

어떤 상황에 쓰이는가?



19951번 : 태상이의 훈련소 생활

<https://www.acmicpc.net/problem/19951>

문제 설명

2020년 5월 14일 논산훈련소에 입대한 태상이는 첫 총기 훈련에서 가스 조절기를 잃어버리는 중대한 실수를 범했다. 그로 인해, 태상이는 조교들에게 눈총을 받게 되었다. 조교들은 태상이에게 연병장(운동장)의 흙을 옮기는 일을 주고 제대로 수행하지 못하면 징계를 내리려고 한다.

연병장은 일렬로 이어진 N 개의 칸으로 이루어져 있으며 각 칸마다 높이를 가지고 있고, 첫 번째 칸부터 순서대로 1번, 2번, 3번, ..., N 번 칸으로 명칭이 붙어있다.

조교들은 총 M 명이 있으며, 각 조교들은 태상이에게 a 번 칸부터 b 번 칸까지 높이 k 만큼 흙을 덮거나 파내라고 지시한다. 흙은 주변 산에서 얼마든지 구할 수 있으므로 절대로 부족하지 않다.

태상이는 각 조교의 지시를 각각 수행하면, 다른 조교의 지시로 흙을 덮어둔 칸을 다시 파내기도 하는 비효율적인 일이 발생하는 것을 깨달았다.

그래서 태상이는 각 조교의 지시를 모아 연병장 각 칸의 최종 높이를 미리 구해 한 번에 일을 수행하려고 한다. 불쌍한 태상이를 위해 조교들의 지시를 모두 수행한 뒤 연병장 각 칸의 높이를 구하자.

어디에 쓸 것인가?

어떤 상황에 쓰이는가?

예제 입력 1 복사

```
10 3
1 2 3 4 5 -1 -2 -3 -4 -5
1 5 -3
6 10 5
2 7 2
```

- 초기 연병장의 상태: 1 2 3 4 5 -1 -2 -3 -4 -5
- 첫번째 지시 (1 5 -3)를 수행 한 뒤: -2 -1 0 1 2 -1 -2 -3 -4 -5
- 두번째 지시 (6 10 5)를 수행한 뒤: -2 -1 0 1 2 4 3 2 1 0
- 세번째 지시 (2 7 2)를 수행한 뒤: -2 1 2 3 4 6 5 2 1 0

예제 출력 1 복사

```
-2 1 2 3 4 6 5 2 1 0
```

기본 원리

기본 원리

차이에 대해서 기록하자

전체 흐름은 간단하다.

1. 차분 배열을 선언한다.
2. 시작점, 끝점에 값의 변화를 기록한다.
3. 마지막에 한번에 모든 변화를 처리한다.

=> 근데, 무슨 소리인지 잘 모르겠다. 예시를 보자.

기본 원리

차이에 대해서 기록하자

1	2	3	4	5	-1	-2	-3	-4	-5
---	---	---	---	---	----	----	----	----	----

Action 1: 1 ~ 5번 칸에 -3 적용

1	2	3	4	5	-1	-2	-3	-4	-5
---	---	---	---	---	----	----	----	----	----

1) 단순히 생각하면 이렇게 저장할 수 있다. 근데,,,

-3	-3	-3	-3	-3	0	0	0	0	0	0
----	----	----	----	----	---	---	---	---	---	---

2) 위와 같이 저장하지 않고 아래와 같이 저장한다면 어떻게 될까?
시작점에서 원래 값대로, 끝점 다음에서 - 원래값을 저장한다. 앞서서부터 변화를 적용한다는 것을 이용하는 것이다.

-3	0	0	0	0	3	0	0	0	0	0
----	---	---	---	---	---	---	---	---	---	---

기본 원리

차이에 대해서 기록하자

Action 1: 1 ~ 5번 칸에 -3 적용



1	2	3	4	5	-1	-2	-3	-4	-5
---	---	---	---	---	----	----	----	----	----

-3	0	0	0	0	3	0	0	0	0	0
----	---	---	---	---	---	---	---	---	---	---

지금까지의 변화를 누적할 cnt (초기값 : 0)가 있다.

배열을 순회하면서,
먼저 변화 값을 cnt에 누적한다.
그리고 배열 값에 cnt를 저장한다.

이때 배열에는 정확한 값이 저장되게 된다.

기본 원리

차이에 대해서 기록하자

문제 상에서 주어지는 쿼리들 모두 적용해보자

Action 1) 1 ~ 5번 칸에 -3 적용

1	2	3	4	5	6	7	8	9	10	11
-3	0	0	0	0	3	0	0	0	0	0

Action 2) 6 ~ 10번 칸에 +5 적용

1	2	3	4	5	6	7	8	9	10	11
-3	0	0	0	0	8 (3 + 5)	0	0	0	0	-5

Action 3) 2 ~ 7번 칸에 +2 적용

1	2	3	4	5	6	7	8	9	10	11
-3	2	0	0	0	8	0	-2	0	0	-5

기본 원리

차이에 대해서 기록하자

	1	2	3	4	5	6	7	8	9	10	11
diff	-3	2	0	0	0	8	0	-2	0	0	-5
cnt	-3	-1	-1	-1	-1	7	7	5	5	5	0

diff를 앞에서부터 누적해가면서 cnt를 구해보면 위와 같다.

arr	1	2	3	4	5	-1	-2	-3	-4	-5
-----	---	---	---	---	---	----	----	----	----	----

그리고 구해진 cnt를 원래 arr과 더하여 답을 구한다.

arr + cnt	-2	1	2	3	4	6	5	2	1	0
-----------------	----	---	---	---	---	---	---	---	---	---

구현 방법

구현 방법

C++ 코드

```
typedef long long ll;
typedef vector<ll> llv1;

ll N, M;
llv1 heights;
llv1 difference_array;

void init() {
    ios::sync_with_stdio(0);
    cin.tie(NULL); cout.tie(NULL);

    cin >> N >> M;

    heights.resize(N + 1);
    difference_array.resize(N + 2, 0);
    for1(1, N + 1) {
        cin >> heights[i];
    }
}
```

```
void solve() {
    init();
    for1(0, M) {
        ll a, b, diff;
        cin >> a >> b >> diff;
        difference_array[a] += diff;
        difference_array[b + 1] -= diff;
    }

    ll crt = 0;
    for1(1, N + 1) {
        crt += difference_array[i];
        heights[i] += crt;
    }

    for1(1, N + 1) {
        cout << heights[i] << " ";
    }
}
```

Thank You

모두의 알고리즘 학습 여정을 응원합니다!