



01. Prefix Sum & Queries

2019 Summer / 20141574 임지환



이 수업은...

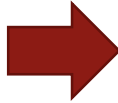
1. 중급 스터디입니다.
2. 여러 개념이 섞인 문제들을 소개합니다.
3. 능동적인 참여를 원합니다.



Prefix Sum?

ex) 수열이 주어지고 i 번째 ~ j 번째까지의 합을 구하려 한다면?

```
for(int idx = i; idx <= j; idx++)  
    sum += a[idx];
```

 $O(j - i)$



#11659 구간 합 구하기 4

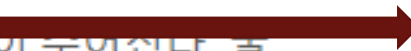
문제

수 N 개가 주어졌을 때, i 번째 수부터 j 번째 수까지 합을 구하는 프로그램을 작성하시오.

입력

$O(\text{구간 길이})$

$O(\text{구간 길이} \times M)$

첫째 줄에 수의 개수 N ($1 \leq N \leq 100,000$), 합을 구해야 하는 횟수 M ($1 \leq M \leq 100,000$)이 주어진다. 둘째 줄에는 N 개의 수가 주어진다. 수는 1,000보다 작거나 같은 자연수이다. 셋째 줄부터 M 개의 줄에는 합을 구해야 하는 구간 i 와 j 가 주어진다.  Query

출력

총 M 개의 줄에 입력으로 주어진 i 번째 수부터 j 번째 수까지 합을 출력한다.



#11659 구간 합 구하기 4

- 쿼리 수를 줄일 수는 없다.
- 쿼리 당 소요 시간을 줄여야 하는데 How?



Prefix Sum (부분 합)

수열 a 에 대하여 $\sum_{k=i}^j a_k$ 를 구한다면?

1. 수열 a 가 잘 정의되어 있어 합공식을 유도할 수 있을 때

2. $S_n = \sum_{k=1}^n a_k$ 을 알고 있을 때



Prefix Sum (부분 합)

$$S_n = S_{n-1} + a_n$$

$$\sum_{k=i}^j a_k = S_j - S_{i-1}$$



Prefix Sum (부분 합)

```
int arr[100010], psum[100010];
```

```
//arr[1 ... 100000]
```

```
for(int i = 1; i <= 100000; i++)
```

```
    psum[i] = psum[i-1] + arr[i];
```

```
int Sum = psum[j] - psum[i-1]; // sum of ith~jth (i<=j)
```




#11659 구간 합 구하기 4

```
int arr[100001], psum[100001];

int main() {
    int N, M, l, r;
    scanf("%d%d", &N, &M);
    for (int i = 1; i <= N; i++) {
        scanf("%d", arr + i);
        psum[i] = psum[i - 1] + arr[i];
    }
    for (int i = 0; i < M; i++) {
        scanf("%d%d", &l, &r);
        printf("%d\n", psum[r] - psum[l - 1]);
    }
}
```



Building Palindromes (Round B 2019 – Kick Start)

팰린드롬(palindrome)은 앞에서 읽었을 때와 뒤에서 읽었을 때 모두 같은 단어를 말한다. 'ANNA', 'RACECAR', 'AAA', 'X'는 모두 팰린드롬이다. 'AB', 'FROG', 'YOYO' 등은 팰린드롬이 아니다.

길이가 $N(\leq 10^5)$ 인 문자열이 주어지고 $[L, R]$ 형태의 쿼리 Q 개($\leq 10^5$)가 주어질 때 ($1 \leq L \leq R \leq N$) L 번째부터 R 번째까지의 문자를 모두 이용하여 팰린드롬을 만들 수 있는지 판별하여 가능한 경우의 수를 출력하라.



Building Palindromes (Round B 2019 – Kick Start)



Example)

7 5

ABAACCA

3 6

AACC

CAAC

4 4

A

A

2 5

BAAC

6 7

CC

3 7

AACCA

CAAAC



Building Palindromes (Round B 2019 – Kick Start)

- 구간 길이가 홀수인 경우?

홀수개인 알파벳은 1개, 다른 알파벳은 짝수개

- 구간 길이가 짝수인 경우?

등장하는 모든 알파벳의 개수가 짝수개



```
int psum[26][100001];

void input() {
    cin >> n >> q >> str;
    memset(psum, 0, sizeof(psum));

    for (int i = 0; i < 26; i++) {
        for (int j = 0; j < str.size(); j++) {
            psum[i][j + 1] = psum[i][j];
            if (str[j] - 'A' == i)
                psum[str[j] - 'A'][j + 1]++;
        }
    }
}
```

```
bool query(int a, int b) {
    int alp[26] = { 0 };
    for (int i = 0; i < 26; i++)
        alp[i] = psum[i][b] - psum[i][a - 1];
    bool flag = false;
    for (int i = 0; i < 26; i++) {
        if (alp[i] % 2 == 0) continue;
        else {
            if ((b - a + 1) % 2 == 0) return false;
            else {
                if (!flag) flag = true;
                else return false;
            }
        }
    }
    return true;
}
```



RGB Substring (Codeforces Ro.575 – D2)

문자 'R', 'G', 'B'로만 구성된 길이 n 의 문자열이 있다.

문자열을 적절히 수정하여 길이 k 의 "RGBRGB...", "GBRBGR...", 혹은 "BRGBRG..." 형태의 문자열이 등장하도록 하려 한다.

문자열의 수정은 아무렇게나 해도 되지만 수정을 최소한으로 한다.

q 개의 쿼리에 대해 이때 최소 수정횟수를 구하여라. 이때 모든 쿼리의 n 에 대해 $\sum n \leq 200,000$ 이 보장된다. ($1 \leq k \leq n \leq 200,000$)



RGB Substring (Codeforces Ro.575 – D2)

1. RGB...일 때, GBR...일 때, BRG...일 때의 케이스를 적용시켜본다.
2. 주어진 문자열의 각 자리에 대해 해당 문자를 바꿔야 하는지 확인한다.



RGB Substring (Codeforces Ro.575 – D2)

1. RGB...일 때, GBR...일 때, BRG...일 때의 케이스를 적용시켜본다.

편의상 R,G,B가 아닌 0, 1, 2로 바뀌어서 생각해본다. 왜?

R은 0, 0+3, 0+6, ... 에서 등장해야 하고

G는 1, 1+3, 1+6, ... 에서 등장해야 하고

B는 2, 2+3, 2+6, ... 에서 등장해야 하기 때문.



RGB Substring (Codeforces Ro.575 – D2)

2. 주어진 문자열의 각 자리에 대해 해당 문자를 바꿔야 하는지 확인한다.

$i, i+3, i+6, \dots$ 번자리가 0인지 확인.

Generization : $i\%3$ 번 자리가 0인지 확인.



RGB Substring (Codeforces Ro.575 – D2)

3. 구간합 배열 정의

$\text{psum}[i] := i\text{번 index까지의 바뀌어야 하는 문자의 총 개수}$

$$\text{ans} = \min(\text{ans}, \text{psum}[i] - \text{psum}[i-k])$$



Painting the Fence (Codeforces Educational Ro.61 – C)

n 칸의 울타리가 있고, q 명의 페인트공을 불러 페인트칠을 하려 한다.

i 번째 페인트공은 $l_i \leq x \leq r_i$ 의 구간을 칠한다.

페인트공 모두를 고용할 예산이 부족하여 두명은 고용할 수 없다. 즉, $q-2$ 명만을 고용하여 페인트칠을 하려 한다. 이때 최대로 칠할 수 있는 울타리의 개수를 구하여라.

$(3 \leq n, q \leq 5000), (1 \leq l_i \leq r_i \leq n)$



Painting the Fence (Codeforces Educational Ro.61 – C) $O(q^2n)$

Naïve solution

2중 for문을 통해 2명을 걸러내고,

➡ $O(q^2)$

남은 $q-2$ 명에 대해 칠을 하여 그 개수를 세어본다?

➡ $O(n)$

$$O(q^2n)$$



Painting the Fence (Codeforces Educational Ro.61 – C)

Prefix Sum Solution

1. 아무튼 페인트칠은 해보아야 한다.
2. 각 케이스에 대해 몇 개의 울타리가 칠해졌는지 구해보아야 한다.



Painting the Fence (Codeforces Educational Ro.61 – C)

Prefix Sum Solution

1. 아무튼 페인트칠은 해보아야 한다.
 - 1) for문을 돌려서 페인트칠의 정보를 처리할 수 없다.
 - 2) 2명의 페인트공에 대해 처리해야 한다.



페인트공의 정보 처리

- 구간 양끝에 +1, -1을 한다면?

ex) [1, 4], [3, 7]





페인트공의 정보 처리

- 구간 양끝에 +1, -1을 한다면?

ex) [1, 4], [3, 7]

1	1	2	2	1	1	1	0	0



페인트공의 정보 처리

- 구간 양끝에 +1, -1을 한다면?

ex) **[1, 4]**, [3, 7]

1	1	2	2	1	1	1	0	0
+1				-1				



페인트공의 정보 처리

- 구간 양끝에 +1, -1을 한다면?

ex) [1, 4], **[3, 7]**

1	1	2	2	1	1	1	0	0
+1		+1		-1			-1	



페인트공의 정보 처리

- 구간 양끝에 +1, -1을 한다면?

ex) [1, 4], [3, 7]





페인트공의 정보 처리

- 구간 양끝에 +1, -1을 한다면?

ex) [1, 4], [3, 7]





페인트공의 정보 처리

- 구간 양끝에 +1, -1을 한다면?

ex) [1, 4], [3, 7]

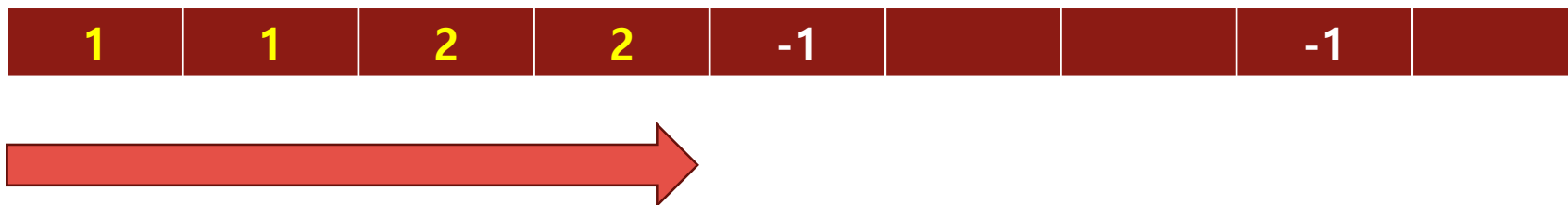




페인트공의 정보 처리

- 구간 양끝에 +1, -1을 한다면?

ex) [1, 4], [3, 7]

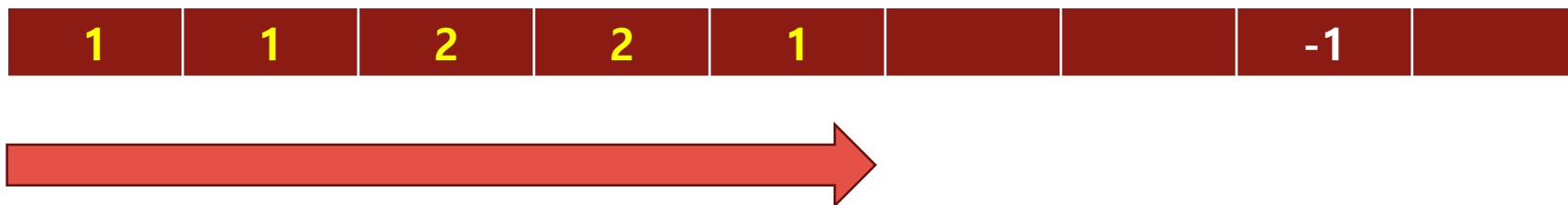




페인트공의 정보 처리

- 구간 양끝에 +1, -1을 한다면?

ex) [1, 4], [3, 7]

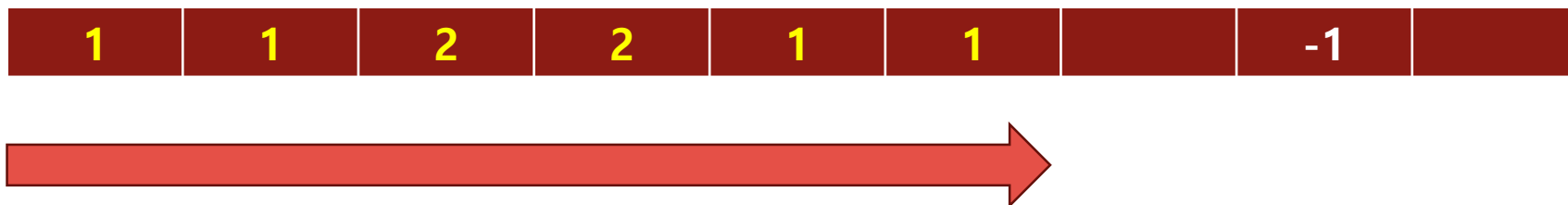




페인트공의 정보 처리

- 구간 양끝에 +1, -1을 한다면?

ex) [1, 4], [3, 7]

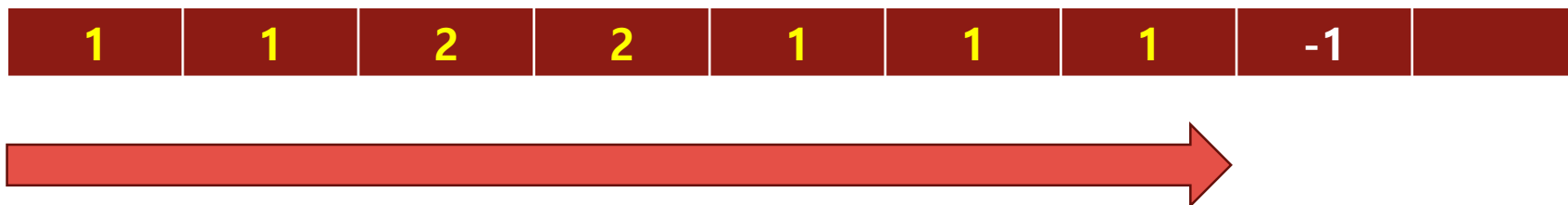




페인트공의 정보 처리

- 구간 양끝에 +1, -1을 한다면?

ex) [1, 4], [3, 7]

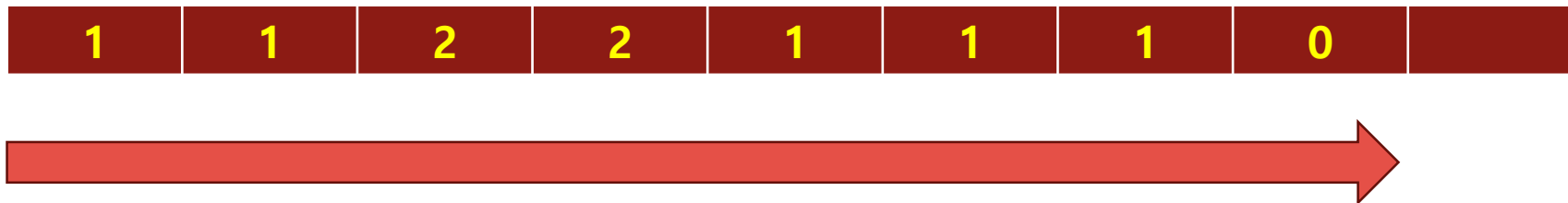




페인트공의 정보 처리

- 구간 양끝에 +1, -1을 한다면?

ex) [1, 4], [3, 7]

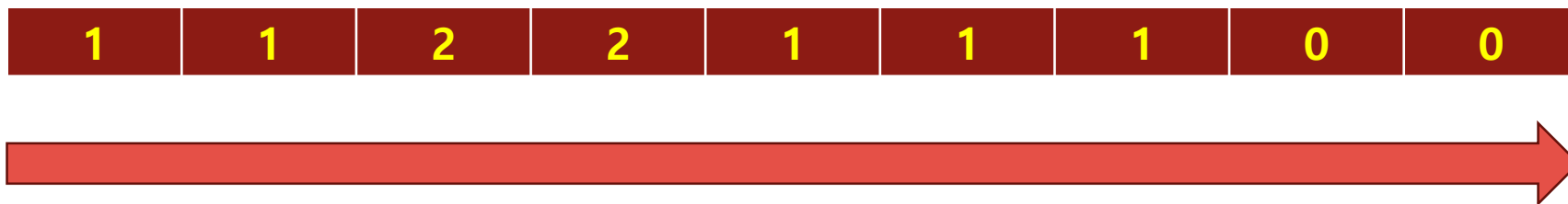




페인트공의 정보 처리

- 구간 양끝에 +1, -1을 한다면?

ex) [1, 4], [3, 7]






2명의 페인트공에 대한 처리


1. 한명은 무조건 제외하고 시작
⇒ q번 반복
2. 한명을 제외한 나머지 페인트공의 구간처리를 하였을 때 구간 값이 1인 경우 찾기
=> 구간처리에 (q-1), 울타리길이 n에 대해 적용하기 때문에 $\max(q-1, n)$ 만큼의 시간

$$O(q * (\max(q, n)))$$




Problem Set

 11659 구간 합 구하기 4

 11660 구간 합 구하기 5

 11969 Breed Counting

 KickStart Ro.B Building Palindromes

 1806 부분합

 CF 1196D2 RGB Substring

 CF 1132C Painting the Fence

 15589 Lifeguards (Silver)

 CF 1076E Vasya and a Tree

 12745 Traffic (Small)

 12746 Traffic (Large)