

시그 프리와 퍼싱 프리 (도자)

소스코드

최백준 choi@startlink.io

C++14

```
1 #include <cstdio>
2 #include <cmath>
3 #include <vector>
4 #include <algorithm>
5 using namespace std;
6 const int inf = 500001;
7 struct Exam {
8     int first, second, third;
9 };
10 void init(vector<int> &tree, int node, int start, int end) {
11     tree[node] = inf;
12     if (start != end) {
13         init(tree, node*2, start, (start+end)/2);
14         init(tree, node*2+1, (start+end)/2+1, end);
15     }
16 }
17 int minimum(vector<int> &tree, int node, int start, int end, int left, int right) {
18     if (left > end || right < start) {
19         return inf;
20     }
21     if (left <= start && end <= right) {
22         return tree[node];
23     }
24     return min(minimum(tree, node*2, start, (start+end)/2, left, right), minimum(tree, node*2+1,
25 (start+end)/2+1, end, left, right));
26 }
27 void update(vector<int> &tree, int node, int start, int end, int i, int value) {
28     if (i > end || i < start) {
29         return;
30     }
31     tree[node] = min(tree[node], value);
32     if (start != end) {
33         update(tree, node*2, start, (start+end)/2, i, value);
34         update(tree, node*2+1, (start+end)/2+1, end, i, value);
35     }
36 }
37 bool cmp (const Exam &u, const Exam &v) {
38     return u.second < v.second;
39 }
40 int main(){
41     int n;
42     scanf("%d",&n);
43     vector<Exam> a(n+1);
44     for (int i=1; i<=n; i++) {
45         int t;
46         scanf("%d",&t);
47         a[t].first = i;
48     }
49     for (int i=1; i<=n; i++) {
50         int t;
51         scanf("%d",&t);
52         a[t].second = i;
53     }
54     for (int i=1; i<=n; i++) {
55         int t;
56         scanf("%d",&t);
57         a[t].third = i;
58     }
59     sort(a.begin(),a.end(),cmp);
60     int n = (int)ceil(log2(n));
61     int tree_size = (1 << (h+1)) - 1;
62     vector<int> tree(tree_size);
63     init(tree, 1, 1, n);
64     int ans = 0;
65     for (int i=1; i<=n; i++) {
66         int best = minimum(tree, 1, 1, n, 1, a[i].first);
67         if (best > a[i].third) {
68             ans += 1;
69         }
70         update(tree, 1, 1, n, a[i].first, a[i].third);
71     }
72     printf("%d\n",ans);
73     return 0;
74 }
```

$N+1$
 $N \leq 50000$

시행동작대로
작성

B4413

[1,0]

A

-1

P

1, a[i].first

∞

A

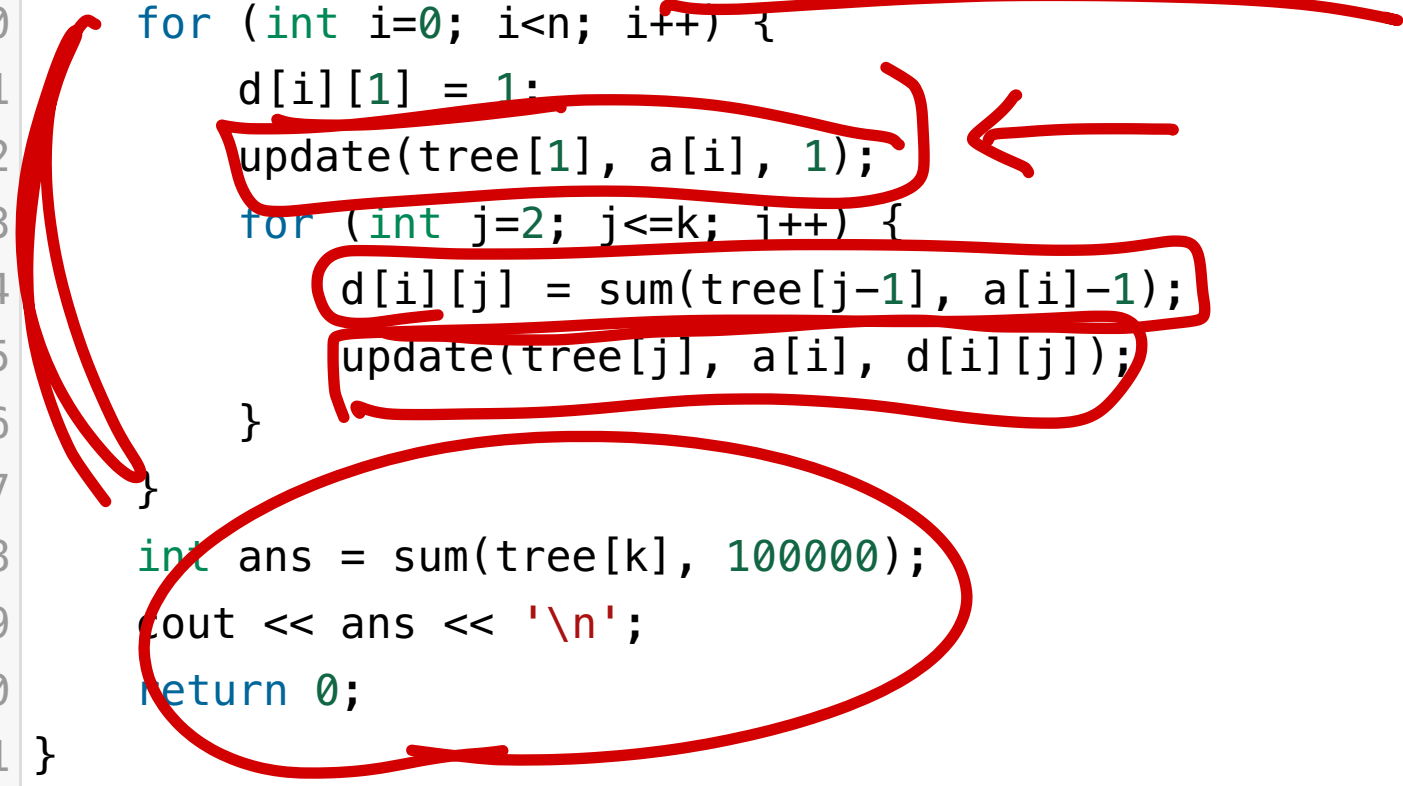
C

$O(N \log N)$

C++14

```
1 #include <iostream>
2 #include <vector>
3 using namespace std;
4 const int mod = 5000000;
5 void update(vector<int> &tree, int i, int diff) {
6     while (i < tree.size()) {
7         tree[i] += diff;
8         tree[i] %= mod;
9         i += (i & -i);
10    }
11 }
12 int sum(vector<int> &tree, int i) {
13     int ans = 0;
14     while (i > 0) {
15         ans += tree[i];
16         ans %= mod;
17         i -= (i & -i);
18     }
19     return ans;
20 }
21 int main() {
22     int n, k;
23     cin >> n >> k;
24     vector<int> a(n);
25     for (int i=0; i<n; i++) {
26         cin >> a[i];
27     }
28     vector<vector<int>> d(n, vector<int>(k+1));
29     vector<vector<int>> tree(k+1, vector<int>(100001));
30     for (int i=0; i<n; i++) {
31         d[i][1] = 1;
32         update(tree[1], a[i], 1);
33         for (int j=2; j<=k; j++) {
34             d[i][j] = sum(tree[j-1], a[i]-1);
35             update(tree[j], a[i], d[i][j]);
36         }
37     }
38     int ans = sum(tree[k], 100000);
39     cout << ans << '\n';
40     return 0;
41 }
42
```

Op + 값을 채워
 $\Sigma \leftarrow \frac{n+1}{2}$



결과	메모리	시간	코드 길이
맞았습니다!!	46884 KB	804 ms	940 B

끝

코드 플러스

<https://code.plus>

- 슬라이드에 포함된 소스 코드를 보려면 "정보 수정 > 백준 온라인 저지 연동"을 통해 연동한 다음, "백준 온라인 저지"에 로그인해야 합니다.
- 강의 내용에 대한 질문은 코드 플러스의 "질문 게시판"에서 할 수 있습니다.
- 문제와 소스 코드는 슬라이드에 첨부된 링크를 통해서 볼 수 있으며, "백준 온라인 저지"에서 서비스됩니다.
- 슬라이드와 동영상 강의는 코드 플러스 사이트를 통해서만 볼 수 있으며, 동영상 강의의 녹화와 다운로드, 배포와 유통은 저작권법에 의해서 금지되어 있습니다.
- 다른 경로로 이 슬라이드나 동영상 강의를 본 경우에는 codeplus@startlink.io 로 이메일 보내주세요.
- 강의 내용, 동영상 강의, 슬라이드, 첨부되어 있는 소스 코드의 저작권은 스타트링크와 최백준에게 있습니다.