

```

int pref_max (int p) {
    int nes = 0;
    for (i = p; p > 0; p = p & p) {
        nes = max (nes, fenw_max [p]);
    }
    return nes;
}

```

```

void update_max (int p, int x) {
    for (i = p; i <= m; p = p & p) {
        fenw_max [p] = max (fenw_max [p], x);
    }
}

```

```

void add (int x) {
    int prv = 0;
    for (int pos: where [x]) {
        update_max (prv + 1, pos);
        prv = pos;
    }
}

```

```

index_lim = min (index_lim, prv);
}

```

```

int main () {
    ios::sync_with_stdio (0);
    cin.tie (0);
}

```

```

cin >> n;
for (int i = 1; i <= n; i++) {
    cin >> a [i];
    where [a [i]].pb (i);
}

```

```

index_lim = n;
for (int x = 0; x <= n - 1; x++) {
    add (x);
    int i = 1, ans = 0;
    while (i <= n) {
        int j;
        if (i > index_lim) {
            j = n + 1;
        }
        else {
            j = pref_max (i);
        }
    }
}

```

```

} else {
    j = pref_max (i);
}
}

```

```

} ans++; // 2-бөлем
if (j == i) {
    ans--;
    break;
}
i = j;
}
cout << ans << "\n" [x == n];
}
// cerr << (double) clock () / CLOCKS_PER_SEC << endl;
return 0;
}

```

```

#pragma GCC optimize ("O3")
#include <bits/stdc++.h>
#define pb push_back
#define all(x) (x).begin(), (x).end()
#define sz(x) (int)(x).size()

```

```

using namespace std;
typedef long long ll;
const int MAXN = (int) 1e5 + 5;
const int k = 300;

```

```

const int l = MAXN / k + 2;
int p [MAXN], a [l] [l];
ll s [MAXN], t [MAXN];
ll a [MAXN], b [l];

```

```

int p [MAXN];
int n, m, q;
void sqrtUpd (int p, ll x) {
    s [p] += x;
    t [p / k] += x;
}

```

```

void sqrtUpd (int id, int l, int r, int x) {
    int ci = 1 / k, cn = n / k;
}

```

50 ұнау