

综合混练

人员

隋天翼、刘新睿、范家郡、牛晓晨、韩承煊 到课

上周作业检查

上周作业链接: <https://www.luogu.com.cn/contest/237555>

2025-0322 五队上课 (字符串)

报名

编辑比赛

题目数4 | 报名人数22

比赛说明 | 题目列表 | 排行榜

名次	参赛者	总分	A	B	C	D
#1	张皓宁	400 (2.31d)	100 (7.82h)	100 (8.62h)	100 (8.96h)	100 (1.25d)
#2	隋天翼	400 (5.71d)	100 (51.55min)	100 (1.62h)	100 (1.84h)	100 (5.53d)
#3	卢炫佑	400 (9.10d)	100 (7.85h)	100 (1.13d)	100 (1.15d)	100 (6.50d)
#4	刘锦轩	363 (2.25d)	100 (8.05min)	100 (1.18h)	100 (1.05d)	63 (1.14d)
#5	范家郡	363 (13.15d)	100 (1.37h)	100 (1.85h)	100 (6.49d)	63 (6.52d)
#6	黄诗琦	300 (23.83h)	100 (7.95h)	100 (8.68h)	100 (7.20h)	
#7	付丙霖	300 (1.04d)	100 (7.80h)	100 (8.54h)	100 (8.65h)	
#8	许岩	300 (1.04d)	100 (7.76h)	100 (8.57h)	100 (8.68h)	
#9	陈瀚霄	300 (1.05d)	100 (7.79h)	100 (8.49h)	100 (8.88h)	
#10	赵广宇	300 (4.20d)	100 (7.79h)	100 (8.60h)	100 (3.51d)	
#11	韩承煊	300 (6.63d)	100 (59.15min)	100 (1.75h)	100 (6.51d)	
#12	刘新睿	300 (6.68d)	100 (59.60min)	100 (1.81h)	100 (6.57d)	
#13	牛晓晨	263 (6.67d)	100 (59.37min)	100 (1.69h)		63 (6.55d)
#14	方冠霖	200 (16.30h)	100 (7.75h)	100 (8.55h)		
#15	韩鸿钜	200 (16.47h)	100 (7.84h)	100 (8.63h)		
#16	金一航	200 (16.95h)	100 (7.89h)	100 (9.07h)		
#17	李政毅	100 (8.00h)	100 (8.00h)			
#18	Elairin176	100 (1.14d)				100 (1.14d)

数据导出 (CSV 格式)

作业

<https://www.luogu.com.cn/contest/238522> (课上讲了 A ~ B 这些题, 课后作业是 C D 题)

课堂表现

同学们在今天的 B 题上遇到了比较大的困难, 这说明了之前的线段树写的不是很熟练, 同学们要课下多复习之前的线段树, 把线段树的题目一定要写熟。

课堂内容

P1637 三元上升子序列

```
#include <bits/stdc++.h>

using namespace std;

typedef long long LL;
const int N = 3e4 + 5, M = 1e5 + 5;
int w[N], tr1[M], tr2[M];

int lowbit(int x) { return x & (-x); }
void update(int x, int k, int tr[]) {
    while (x < M) { tr[x] += k, x += lowbit(x); }
}
int query(int x, int tr[]) {
    int res = 0;
    while (x) { res += tr[x], x -= lowbit(x); }
    return res;
}

int main()
{
    int n; cin >> n;
    for (int i = 1; i <= n; ++i) cin >> w[i];

    for (int i = 1; i <= n; ++i) update(w[i], 1, tr2);

    LL res = 0;
    for (int i = 1; i <= n; ++i) {
        update(w[i], -1, tr2);
        int lnums = query(w[i]-1, tr1), rnums = (n-i) - query(w[i], tr2);
        res += (LL)lnums * rnums;
        update(w[i], 1, tr1);
    }
    cout << res << endl;
    return 0;
}
```

P10471 最大异或对 The XOR Largest Pair

```
#include <bits/stdc++.h>
```

```

using namespace std;

const int N = 100000 + 5, M = 32;
int tr[N*M][2], idx = 0;

void tr_insert(int x) {
    int p = 0;
    for (int i = 31; i >= 0; --i) {
        int u = (x>>i)&1;
        if (!tr[p][u]) tr[p][u] = ++idx;
        p = tr[p][u];
    }
}

int tr_query(int x) {
    int p = 0, res = 0;
    for (int i = 31; i >= 0; --i) {
        int u = (x>>i)&1;
        if (tr[p][u^1]) p = tr[p][u^1], res += ((u^1)<<i);
        else p = tr[p][u], res += (u<<i);
    }
    return res;
}

int main()
{
    int n; cin >> n;
    int res = 0;
    for (int i = 1; i <= n; ++i) {
        int x; cin >> x; tr_insert(x);
        int t = tr_query(x);
        res = max(res, x^t);
    }
    cout << res << endl;
    return 0;
}

```

P3373 【模板】线段树 2

```

#include <bits/stdc++.h>

using namespace std;

typedef long long LL;
const int maxn = 1e5 + 5;
int mod;
struct node {
    int l, r, len;
    int mul, add;
    int sum;
} tr[maxn*4];

```

```

int w[maxn];

void pushup(int u) { tr[u].sum = (tr[u*2].sum + tr[u*2+1].sum) % mod; }

void pushdown(int u) {
    node &uu = tr[u], &ll = tr[u*2], &rr = tr[u*2+1];
    ll.sum = ((LL)ll.sum*uu.mul+(LL)uu.add*ll.len) % mod;
    ll.mul = (LL)ll.mul*uu.mul % mod, ll.add = ((LL)ll.add*uu.mul+uu.add) % mod;
    rr.sum = ((LL)rr.sum*uu.mul+(LL)uu.add*rr.len) % mod;
    rr.mul = (LL)rr.mul*uu.mul % mod, rr.add = ((LL)rr.add*uu.mul+uu.add) % mod;
    uu.mul = 1, uu.add = 0;
}

void build(int u, int l, int r) {
    tr[u] = {l, r, r-l+1, 1, 0, 0};
    if (l == r) { tr[u].sum = w[l]; return; }

    int mid = (l + r) / 2;
    build(u*2, l, mid), build(u*2+1, mid+1, r);
    pushup(u);
}

void modify(int u, int l, int r, int a, int b) {
    if (tr[u].l>=l && tr[u].r<=r) {
        tr[u].sum = ((LL)tr[u].sum*a+(LL)b*tr[u].len) % mod;
        tr[u].mul = (LL)tr[u].mul*a % mod;
        tr[u].add = ((LL)tr[u].add*a+b) % mod;
        return;
    }

    pushdown(u);
    int mid = (tr[u].l + tr[u].r) / 2;
    if (l <= mid) modify(u*2, l, r, a, b);
    if (r > mid) modify(u*2+1, l, r, a, b);
    pushup(u);
}

int query(int u, int l, int r) {
    if (tr[u].l>=l && tr[u].r<=r) return tr[u].sum;

    pushdown(u);
    int mid = (tr[u].l + tr[u].r) / 2;
    int res = 0;
    if (l <= mid) res += query(u*2, l, r);
    if (r > mid) res += query(u*2+1, l, r);
    return res%mod;
}

int main()
{
    int n, m; cin >> n >> m >> mod;
    for (int i = 1; i <= n; ++i) cin >> w[i];
    build(1, 1, n);
}

```

```
while (m -- ) {
    int op; cin >> op;
    if (op == 1) {
        int x, y, k; cin >> x >> y >> k; modify(1, x, y, k, 0);
    } else if (op == 2) {
        int x, y, k; cin >> x >> y >> k; modify(1, x, y, 1, k);
    } else {
        int x, y; cin >> x >> y;
        cout << query(1, x, y) << endl;
    }
}
return 0;
}
```