

# 位运算

## 人员

杨咏丞、李雨谦、陈欣妙、杨俊彦、龙沛轩、李锦澍、袁晨峻、徐思远、王陆文龙、刘奕辰、曹塬、阮文璋、周治润、刘锦轩 到课

## 上周作业检查

https://www.luogu.com.cn/contest/247001

2025-0518六队上课(综合混练)

报名

编辑比赛

题目数5 | 报名人数20

比赛说明 | 题目列表 | 排行榜

名次	参赛者	总分	A	B	C	D	E
#1	袁晨峻	500 (2.29d)	100 (831ms)	100 (245ms)	100 (80ms)	100 (1.14d)	100 (1.15d)
#2	徐思远	500 (3.70d)	100 (698ms)	100 (4.47h)	100 (180ms)	100 (70ms)	100 (3.52d)
#3	杨俊彦	400 (7.28d)	100 (13.38h)		100 (13.65h)	100 (3.07d)	100 (3.08d)
#4	阮文璋	400 (8.24d)	100 (1.14s)	100 (3.51d)	100 (7.12h)	100 (4.43d)	
#5	刘奕辰	400 (15.67d)	100 (4.53d)	100 (4.57d)	100 (290ms)	100 (6.57d)	
#6	杨咏丞	400 (20.75d)	100 (4.42d)	100 (4.42d)	100 (5.55d)	100 (6.36d)	
#7	王承周	300 (7.81d)	100 (3.38d)		100 (636ms)	100 (4.43d)	
#8	褚锦轩	200 (6.99h)	100 (1.70s)	0	100 (6.99h)		
#9	李锦澍	200 (12.45d)				100 (6.19d)	100 (6.26d)
#10	白芸琿	100 (110ms)			100 (110ms)		
#11	王陆文龙	100 (2.51s)	100 (2.51s)				
#12	周治润	82 (0ms)	82 (0ms)		0 (0ms)		
#13	SSJ司云心	4 (0ms)	4 (0ms)				
#14	王毅博	0 (0ms)	0 (0ms)				
#15	许睿谦	0 (0ms)	0 (0ms)				

## 作业

https://www.luogu.com.cn/contest/248672 (课上讲了 A ~ E 题, 课后作业是 F 题)

## 课堂表现

同学们这节课课上做题表现整体不错, 只有第三题写的相对差一点, 课上没写出来的同学课下要再写一写。

## 课堂内容

### B4033 [语言月赛 202409] 考试

先记录  $a[i] > b[i]$  的有  $cnt1$  个,  $a[i] == b[i]$  的有  $cnt2$  个,  $a[i] < b[i]$  的有  $cnt3$  个

1.  $cnt1 > cnt3$  时, 直接输出 0
2.  $cnt1 + cnt2 > cnt3$  时, 直接输出  $cnt3 + 1 - cnt1$
3. 把所有  $a[i] < b[i]$  的, 按照  $b[i] - a[i]$  的值从小到大排序, 然后从小往大处理

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

using namespace std;

const int maxn = 1000 + 5;
int a[maxn], b[maxn];

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

    vector<int> vec;
    int cnt1 = 0, cnt2 = 0, cnt3 = 0;
    for (int i = 1; i <= n; ++i) {
        if (a[i] > b[i]) ++cnt1;
        else if (a[i] == b[i]) cnt2++;
        else vec.push_back(b[i] - a[i]), cnt3++;
    }

    if (cnt1 > cnt3) { cout << 0 << endl; return 0; }
    if (cnt1 + cnt2 > cnt3) { cout << cnt3 + 1 - cnt1 << endl; return 0; }

    int res = cnt2;
    cnt1 += cnt2;
    sort(vec.begin(), vec.end());
    for (int i : vec) {
        res += i;
        cnt3--;
        if (cnt1 > cnt3) break;
        res++;
        cnt1++;
        if (cnt1 > cnt3) break;
    }
    cout << res << endl;
    return 0;
}
```

## P2853 [USACO06DEC] Cow Picnic S

以每个奶牛为起点做一遍 dfs或bfs, 把搜到的点进行标记, 最后看哪些点走过 k 次即可

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

using namespace std;

const int maxn = 1000 + 5;
vector<int> vec[maxn];
int k, n, m;
bool st[maxn];
int f[maxn];

void dfs(int u) {
    st[u] = true; f[u]++;
    for (int i : vec[u]) {
        if (!st[i]) dfs(i);
    }
}

void solve(int x) {
    memset(st, false, sizeof(st));
    dfs(x);
}

int main()
{
    cin >> k >> n >> m;

    vector<int> cows;
    for (int i = 1; i <= k; ++i) {
        int x; cin >> x; cows.push_back(x);
    }

    while (m -- ) {
        int u, v; cin >> u >> v; vec[u].push_back(v);
    }

    for (int i : cows) solve(i);

    int res = 0;
    for (int i = 1; i <= n; ++i) {
        if (f[i] == k) ++res;
    }
    cout << res << endl;
    return 0;
}
```

## P1469 找筷子

把所有整数异或一遍

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

using namespace std;

int main()
{
    int n; cin >> n;
    int res = 0;
    for (int i = 1; i <= n; ++i) {
        int x; scanf("%d", &x);
        res ^= x;
    }
    cout << res << endl;
    return 0;
}
```

### B3622 枚举子集（递归实现指数型枚举）

二进制枚举

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

using namespace std;

int main()
{
    int n; cin >> n;
    for (int i = 0; i < (1<<n); ++i) {
        for (int j = n-1; j >= 0; --j) {
            if ((i>>j)%2 == 1) cout << "Y";
            else cout << "N";
        }
        cout << endl;
    }
    return 0;
}
```

### P11233 [CSP-S 2024] 染色

这个题同学们利用二进制枚举, 拿 20 分即可

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

using namespace std;

const int maxn = 15 + 5;
```

```

int w[maxn];
bool st[maxn];

int calc(int n) {
    int res = 0;
    for (int i = 1; i <= n; ++i) {
        int pos = -1;
        for (int j = i-1; j >= 1; --j) {
            if (st[i] == st[j]) {
                pos = j; break;
            }
        }
        if (pos != -1 && w[pos] == w[i]) res += w[i];
    }
    return res;
}

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

    int res = 0;
    for (int i = 0; i < (1<<n); ++i) {
        for (int j = 0; j < n; ++j) {
            if ((i>>j) % 2 == 1) st[j+1] = true;
            else st[j+1] = false;
        }
        res = max(res, calc(n));
    }
    // cout << "----- ";
    cout << res << endl;
}

int main()
{
    int T; cin >> T;
    while (T -- ) solve();
    return 0;
}

```

## T214660 众数

开一个大小为 70 的数组代表每个二进制位出现过多少次, 考虑每个二进制位出现过多少次即可

```

#include <bits/stdc++.h>

using namespace std;

typedef long long LL;
const int n = 999999;
int st[70];

```

```

int main()
{
    for (int i = 1; i <= n; ++i) {
        LL x; cin >> x;
        for (int j = 0; j <= 62; ++j) {
            if ((x>>j) % 2 == 1) st[j]++;
        }
    }

    LL res = 0;
    for (int i = 0; i <= 62; ++i) {
        if (st[i] > n/2) res += (1LL<<i);
    }
    cout << res << endl;
    return 0;
}

```

### P9426 [蓝桥杯 2023 国 B] 抓娃娃

**关键点:** 每次询问的区间长度, 都会  $\geq$  每一个线段的长度

看一个区间是否覆盖了一半的线段, 就看这个区间是否覆盖了一个线段的中点即可

```

#include <bits/stdc++.h>

using namespace std;

const int maxn = 2e6 + 5;
int w[maxn], p[maxn];

int get_sum(int l, int r) { return p[r] - p[l-1]; }

int main()
{
    int n, T; cin >> n >> T;
    while (n -- ) {
        int l, r; cin >> l >> r; w[l+r]++;
    }
    for (int i = 1; i < maxn; ++i) p[i] = p[i-1] + w[i];

    while (T -- ) {
        int l, r; cin >> l >> r;
        cout << get_sum(2*l, 2*r) << endl;
    }
    return 0;
}

```