

综合混练

人员

程梓豪、隋钰涵、赵熙羽、高健桓、杨瑾硕、董浩桢、牛同泽、秦显森、武敬哲、谢亚锴、牟茗、刘闯速 到课，郭骐嘉 线上

上周作业检查

https://www.luogu.com.cn/contest/237832

2025-0323周日10:30

报名

编辑比赛

题目数

7

报名人数

13

比赛说明

题目列表

排行榜

名次	参赛者	总分	A	B	C	D	E	F	G
#1	谢亚锴	620 (12.96d)	100 (19.97min)	100 (46.62min)	100 (1.63h)	100 (1.99h)	100 (2.01h)	100 (6.34d)	20 (6.34d)
#2	杨瑾硕	600 (9.60h)	100 (1.06h)	100 (1.07h)	100 (1.63h)	100 (1.84h)	100 (1.99h)	100 (2.01h)	
#3	程梓豪	600 (6.74d)	100 (26.53min)	100 (2.05h)	100 (1.67h)	100 (1.82h)	100 (4.26h)	100 (6.32d)	
#4	赵熙羽	600 (7.26d)	100 (33.18min)	100 (49.92min)	100 (1.59h)	100 (1.74h)	100 (1.76h)	100 (6.99d)	
#5	郭骐嘉	500 (6.69d)	100 (22.23min)	100 (44.10min)	100 (1.62h)	100 (2.07h)	100 (6.49d)		
#6	隋钰涵	500 (13.06d)	100 (23.13min)	100 (1.01h)	100 (4.33d)	100 (4.33d)	100 (4.33d)		
#7	牛同泽	500 (13.94d)	100 (28.47min)	100 (51.53min)	100 (1.62h)	100 (6.91d)	100 (6.91d)		
#8	高健桓	470 (1.57d)	100 (21.37min)	100 (51.62min)	100 (1.51h)	80 (1.84h)	90 (1.38d)		
#9	董浩桢	400 (4.54h)	100 (23.77min)	100 (48.97min)	100 (1.33h)	100 (2.00h)			
#10	夏志赫	400 (6.69d)	100 (1.04h)	100 (1.04h)		100 (6.53d)	100 (1.74h)		
#11	秦显森	380 (4.42h)	100 (21.67min)	100 (42.85min)	100 (1.64h)	80 (1.71h)			
#12	刘闯速	300 (3.57h)	100 (59.80min)	100 (59.62min)	100 (1.58h)				
#13	武敬哲	250 (6.29d)	100 (1.19h)	50 (1.18h)	100 (6.19d)				

作业

https://www.luogu.com.cn/contest/238526 (课上讲了 A ~ C 题, 课后作业是 D 题)

课堂表现

课上题目细节比较多, 题目也比较难一些, 同学们课上做题不是特别好, 课下要再多好好研究研究。

课堂内容

P1843 奶牛晒衣服

二分, 判断 mid 秒能否把所有衣服都烘干即可

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

using namespace std;

typedef long long LL;
const int maxn = 5e5 + 5;
int w[maxn];
int n, a, b;

int get_up(int x, int y) { return (x+y-1)/y; }

bool check(int mid) {
    int res = 0;
    for (int i = 1; i <= n; ++i) {
        if (a*mid >= w[i]) continue;
        res += get_up(w[i]-a*mid, b);
        if (res > mid) return false;
    }
    return true;
}

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

    int l = 1, r = 5e5 + 5;
    while (l <= r) {
        int mid = (l + r) / 2;
        if (check(mid)) r = mid-1;
        else l = mid+1;
    }
    cout << l << endl;
    return 0;
}
```

P2360 地下城主

bfs 板子题, 只是变成了 三维 问题

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

using namespace std;

const int maxn = 30 + 5;
char s[maxn][maxn][maxn];
int f[maxn][maxn][maxn];
struct node {
```

```

    int x, y, z;
};
int dx[] = {-1, 1, 0, 0, 0, 0};
int dy[] = {0, 0, -1, 1, 0, 0};
int dz[] = {0, 0, 0, 0, -1, 1};

int main()
{
    int n, m, c; cin >> n >> m >> c;
    int sx, sy, sz, ex, ey, ez;
    for (int i = 1; i <= n; ++i) {
        for (int j = 1; j <= m; ++j) {
            cin >> (s[i][j]+1);
            for (int k = 1; k <= c; ++k) {
                if (s[i][j][k] == 'S') sx=i,sy=j,sz=k;
                if (s[i][j][k] == 'E') ex=i,ey=j,ez=k;
            }
        }
    }

    memset(f, -1, sizeof(f));
    queue<node> q; q.push({sx,sy,sz}); f[sx][sy][sz] = 0;
    while (!q.empty()) {
        node u = q.front(); q.pop();
        int x = u.x, y = u.y, z = u.z;
        for (int i = 0; i < 6; ++i) {
            int nx = x+dx[i], ny = y+dy[i], nz = z+dz[i];
            if (nx>=1&&nx<=n&&ny>=1&&ny<=m&&nz>=1&&nz<=c&&s[nx][ny][nz]!='#'&&f[nx][ny][nz]==-1) {
                q.push({nx,ny,nz}); f[nx][ny][nz] = f[x][y][z]+1;
            }
        }
    }

    if (f[ex][ey][ez] == -1) cout << "Trapped!" << endl;
    else cout << "Escaped in " << f[ex][ey][ez] << " minute(s)." << endl;
    return 0;
}

```

P10416 [蓝桥杯 2023 国 A] XYZ

当 $Z == 2*L$ 时, X/Y 的方案只有 1 种

当 $Z == 2*L+1$ 时, X/Y 的方案有 2 种

当 $Z == 2*L+2$ 时, X/Y 的方案有 3 种

...

当 $Z == R$ 时, X/Y 的方案有 $R-2*L+1$ 种

所以, 总答案应该是 $1 + 2 + 3 + \dots + R-2*L+1$, 可以用等差数列 $O(1)$ 求和

需要注意的问题: $R-2*L+1$ 有可能 小于1, 此时说明无解, 应该输出 0

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

using namespace std;

typedef long long LL;

LL get_sum(int l, int r) {
    if (l > r) return 0;
    return ((LL)l+r)*(r-l+1)/2;
}

int main()
{
    int T; cin >> T;
    while (T -- ) {
        int l, r; cin >> l >> r;
        cout << get_sum(1, r-2*l+1) << endl;
    }
    return 0;
}
```

P10417 [蓝桥杯 2023 国 A] 第 K 小的和

二分, 判断所有 a_i+b_j 的组合中, 是否有 $\geq k$ 个数满足 $\leq \text{mid}$ 的条件

check 的方法: 对每个 a_i 来说, 看 b 数组中, 有多少满足 $\leq \text{mid}-a_i$ 的数即可, 这里可以用 二分查找 加速

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

using namespace std;

typedef long long LL;
const int maxn = 1e5 + 5;
int a[maxn], b[maxn];
int n, m; LL k;

bool check(int mid) {
    LL res = 0;
    for (int i = 1; i <= n; ++i) {
        int x = mid - a[i];
        int pos = upper_bound(b+1, b+m+1, x) - b - 1;
        res += pos;
    }
    return res >= k;
}

int main()
{
}
```

```

cin >> n >> m >> k;
for (int i = 1; i <= n; ++i) cin >> a[i];
for (int i = 1; i <= m; ++i) cin >> b[i];
sort(b+1, b+m+1);

LL l = 1, r = 2e9;
while (l <= r) {
    int mid = (l + r) / 2;
    if (check(mid)) r = mid-1;
    else l = mid+1;
}

cout << l << endl;
return 0;
}

```

P1404 平均数

二分, 判断能否有一个长度 $\geq m$ 的区间的平均值满足 $\geq \text{mid}$ 的条件

check 方法: 每个数 $- \text{mid}$, 此时, 原问题转化为了 判断能否有一个长度 $\geq m$ 的区间的总和满足 ≥ 0 的条件

此时, 可以用之前 最大子段和 的方法, 找到最大子段和, 看是否 ≥ 0 即可

唯一区别是, 要满足长度 $\geq m$ 的要求, 所以求最大子段和时, 每次要用 $p[i] - p_{\min}[i-m]$

```

#include <bits/stdc++.h>

using namespace std;

typedef long long LL;
const int maxn = 1e5 + 5;
int w[maxn];
int n, m;
int a[maxn];
LL p[maxn], p_min[maxn];

bool check(int mid) {
    for (int i = 1; i <= n; ++i) {
        a[i] = w[i] - mid, p[i] = p[i-1] + a[i];
        p_min[i] = min(p_min[i-1], p[i]);
    }

    for (int i = m; i <= n; ++i) {
        if (p[i] - p_min[i-m] >= 0) return true;
    }
    return false;
}

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

```

```
for (int i = 1; i <= n; ++i) cin >> w[i], w[i] *= 1000;

int l = 0, r = 2000000;
while (l <= r) {
    int mid = (l + r) / 2;
    if (check(mid)) l = mid+1;
    else r = mid-1;
}
cout << r << endl;
return 0;
}
```