

# 综合练习

## 人员

温郝冬、柳力玮、谢梓轩、韩昱辰、赵牧之、田心一、李瑞涵、初锦阳、苑钊 到课, 刘宸熙、姜皓轩、刘子轩 线上

## 上周作业检查

上周作业链接: <https://www.luogu.com.cn/contest/252014>

2025-0615周日15:30

报名

编辑比赛

题目数6 | 报名人数16

比赛说明

题目列表

排行榜

名次	参赛者	总分	A	B	C	D	E	F
#1	韩昱辰	600 (11.74h)	100 (2.13h)	100 (9.21s)	100 (2.24h)	100 (2.39h)	100 (2.39h)	100 (2.58h)
#2	柳力玮	600 (20.06h)	100 (1.93s)	100 (8.33s)	100 (5.85s)	100 (18ms)	100 (2.13h)	100 (17.92h)
#3	田心一	600 (1.90d)	100 (530ms)	100 (8.20s)	100 (5.48s)	100 (19ms)	100 (22.59h)	100 (23.02h)
#4	赵牧之	600 (16.19d)	100 (623ms)	100 (15.44s)	100 (1.26d)	100 (1.27d)	100 (6.81d)	100 (6.84d)
#5	初锦阳	600 (20.51d)	100 (506ms)	100 (16.35s)	100 (5.55s)	100 (6.83d)	100 (6.85d)	100 (6.83d)
#6	李瑞涵	500 (6.14d)	100 (581ms)	100 (8.29s)	100 (5.50s)	100 (21ms)		100 (6.14d)
#7	谢梓轩	500 (13.84d)	100 (515ms)	100 (9.23s)	100 (5.34s)	100 (6.92d)	100 (6.92d)	0
#8	苑钊	500 (14.81d)	100 (583ms)	100 (11.19s)	100 (6.87d)	100 (1.07d)	100 (6.87d)	0
#9	温郝冬	500 (21.19d)	100 (7.06d)	100 (16.19s)	100 (5.79s)	100 (7.06d)	100 (7.07d)	
#10	王馨琪	400 (14.91s)	100 (515ms)	100 (8.69s)	100 (5.69s)	100 (21ms)	0	
#11	liuyang_123	400 (27.00d)	100 (6.73d)	100 (6.75d)	100 (6.75d)	100 (6.77d)		
#12	刘派	400 (27.05d)	100 (6.74d)	100 (6.75d)	100 (6.77d)	100 (6.79d)		
#13	刘子轩	200 (21.37s)		100 (15.77s)	100 (5.61s)			
#14	李知朔	200 (2.11h)	100 (2.11h)	100 (8.77s)	0 (0ms)			
#15	刘宸熙	30 (0ms)	30 (0ms)					

## 作业

<https://cppoj.kids123code.com/contest/108> (课上讲了 A ~ C 题, 课后作业是 D 题)

## 课堂表现

今天的 A 题 和 C 题属于比较相似的题目, 都是把原数组, 转换到前缀和数组上, 然后利用前缀和数组来解决问题。

同学们课下可以好好复习一下这两道题, 比较一下这两道题的相似处跟不同处。

## 课堂内容

### B4006 [GESP202406 四级] 宝箱

对数组排序, 然后  $O(n^2)$  枚举左右端点, 找所有 右端点-左端点 $\leq k$  条件下的最大的区间和

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

using namespace std;

const int maxn = 1000 + 5;
int w[maxn], p[maxn];

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

int main()
{
    int n, k; cin >> n >> k;
    for (int i = 1; i <= n; ++i) cin >> w[i];
    sort(w+1, w+n+1);
    for (int i = 1; i <= n; ++i) p[i] = p[i-1] + w[i];

    int res = 0;
    for (int i = 1; i <= n; ++i) {
        for (int j = i; j <= n; ++j) {
            if (w[j] - w[i] <= k) res = max(res, get_sum(i, j));
        }
    }
    cout << res << endl;
    return 0;
}
```

### [蓝桥杯 2017 省 B] k 倍区间

之前做过的题, 这节课复习一下, 维护前缀和, 然后把 前缀和数组模k, 看有多少相同的即可

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

using namespace std;

typedef long long LL;
const int maxn = 1e5 + 5;
int w[maxn];
LL p[maxn];
int f[maxn];
```

```

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

    for (int i = 0; i <= n; ++i) f[p[i]%k]++;

    LL res = 0;
    for (int i = 0; i < k; ++i) res += (LL)f[i]*(f[i]-1)/2;
    cout << res << endl;
    return 0;
}

```

## 珍珠链

枚举左端点  $i$ , 枚举右端点  $j$ , 然后 check 这一段区间即可。

```

#include <bits/stdc++.h>

using namespace std;

const int maxn = 450 + 5;
char s[maxn];

int main()
{
    cin >> (s+1);
    int n = strlen(s+1);
    int res = 0;
    for (int i = 1; i <= n; ++i) {
        for (int j = i; j <= n; j++) {
            int cnt0 = 0, cnt1 = 0;
            for (int k = i; k <= j; k++) {
                if (s[k] == 'R') cnt0++;
                else cnt1++;
            }
            if (cnt0 == cnt1) res = max(res, j-i+1);
        }
    }
    cout << res << endl;
    return 0;
}

```

## 珍珠链2

把字母G当做1, 把字母R当做-1, 其实就是找最长的区间, 满足 区间和==0 的条件

区间和==0, 其实就是2个点的前缀和相同, 针对每个前缀和的值, 找这个前缀和第一次出现的位置和最后一次出现的位置即可, 可以用两个 map 进行维护

```

#include <bits/stdc++.h>

using namespace std;

const int maxn = 1e6 + 5;
char str[maxn];
int f[maxn];

int main()
{
    cin >> (str+1);
    int n = strlen(str+1);
    for (int i = 1; i <= n; ++i) {
        if (str[i] == 'G') f[i] = f[i-1] + 1;
        else f[i] = f[i-1] - 1;
    }

    map<int, int> mp1, mp2;
    mp1[0] = 0, mp2[0] = 0;
    for (int i = 1; i <= n; ++i) {
        int t = f[i];
        if (!mp1.count(t)) mp1[t] = i;
        mp2[t] = i;
    }

    int res = 0;
    for (int i = 0; i <= n; i++) res = max(res, mp2[f[i]] - mp1[f[i]]);
    cout << res << endl;
    return 0;
}

```

## 「Wdoi-4」兔已着陆

栈里面维护结构体 {l,r}, 代表里面存放了 l~r 之间的数, 这样可以加快插入和删除的效率

```

#include <bits/stdc++.h>

using namespace std;

typedef long long LL;
const int maxn = 5e5 + 5;
struct node {
    int l, r;
};
stack<node> stk;

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

LL calc(LL k) {
    LL res = 0;

```

```
while (!stk.empty()) {
    node p = stk.top(); stk.pop();
    int l = p.l, r = p.r, len = p.r-p.l+1;
    if (len <= k) k -= len, res += get_sum(l, r);
    else {
        int ll = r - k + 1;
        res += get_sum(ll, r);
        stk.push({l, ll-1});
        break;
    }
}
return res;
}

int main()
{
    int n; cin >> n;
    while (n -- ) {
        int op; cin >> op;
        if (op == 1) {
            int l, r; cin >> l >> r;
            stk.push({l, r});
        } else {
            LL k; cin >> k;
            cout << calc(k) << endl;
        }
    }
    return 0;
}
```