

综合混练

人员

赵广宇、金一航、曹承贤、黄诗琦、卢炫佑、张皓宁、李政毅、付丙霖、韩鸿钜、王彦臻 到课, 陈瀚霄 线上

上周作业检查

上周作业链接: <https://vjudge.net/contest/712438>

Begin: 2025-04-26 08:30 CST

☆ 2025-0426 五队上课 (综合混练)

End: 2025-11-20 16:30 CST

Elapsed: 9:09:20:24

Running

Remaining: 198:22:39:35

OverviewProblemStatusRank (9:09:20:17)DiscussSettingCloneUpdateDelete

Rank	Team	Score	Penalty	A 19 / 39	B 19 / 25	C 17 / 21	D 9 / 14	E 2 / 3
1	☆ Hacker_Cracker sty0948 (隋...)	5	14985	1:30:35	1:14:23	2:31:53	4:14:13:26	5:13:55:17 (-1)
2	☆ fbl123bc (付丙霖)	5	41211	7:37:53 (-3)	8:54:02 (-5)	9:05:51:11	9:06:07:36	9:07:41:12
3	☆ zhn123bc (张皓宁)	4	11725	7:37:47 (-3)	8:03:02	9:04:18	7:01:40:00	
4	☆ two_tiger (卢炫佑)	4	12197	7:40:44 (-3)	8:03:27	8:52:29 (-1)	7:09:21:14	
5	☆ longlong_int (刘锦轩)	4	14967	0:41:15	12:52:19	12:35:03	9:07:19:04	
6	☆ qp_an (赵广宇)	4	15046	7:24:10	8:06:02	13:27:13	9:05:48:59	
7	☆ ikunTLE (方冠霖)	4	21852	9:37:24 (-2)	8:06:44	6:06:33:55	8:03:14:07	
8	☆ dana230513 (金一航)	4	25686	7:35:00	8:12:18	8:13:51:52 (-1)	8:14:07:05	
9	☆ misaka16384 (黄诗琦)	4	27702	7:38:34 (-1)	8:06:05	9:06:47:01	9:06:50:31	
10	☆ chx123bc (陈瀚霄)	3	1462	7:24:13	8:02:05	8:55:44		
11	☆ niuxiaochen (牛晓晨)	3	12028	0:36:26	1:09:52	8:06:22:24 (-1)	(-1)	
12	☆ lxr123bc (刘新睿)	3	12102	0:35:33	1:10:42	8:07:56:00		
13	☆ fj123bc (范家郡)	3	13039	7:54:26	8:38:32	8:08:46:40	(-1)	
14	☆ cck123bc (曹承贤)	3	14277	7:33:12	8:40:39	9:05:44:08	(-3)	
15	☆ WangYanzhen (王彦臻)	3	30978	7:02:31:10	7:03:36:32 (-1)	7:05:50:32		
16	☆ HT008 (XY)	3	39396	9:02:30:03 (-1)	9:02:45:30	9:03:00:34		
17	☆ Hanhj (韩鸿钜)	3	40026	9:05:28:27 (-3)	9:05:50:38	9:06:27:12 (-1)		
18	☆ FeatherCrow (许岩)	2	965	7:36:47 (-1)	8:08:38			
19	☆ lzy123bc (刘智予)	2	1018	7:40:39 (-3)	8:18:12			

作业

<https://vjudge.net/contest/714711> (课上讲了 A ~ C 这些题, 课后作业是 D E 题)

课堂表现

同学们之前的题目补题情况不太理想, 一定要注意补题的问题, 把之前的题目都补完。

课堂内容

CF1856C To Become Max

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

using namespace std;

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

bool dfs(int n, int pos, int k, int value) { // 是否能 <=k 次把 w[pos] 变成 value
    if (k < 0) return false;
    if (w[pos] >= value) return true;
    if (pos == n) return false;

    if (dfs(n, pos+1, k-(value-w[pos]), value-1)) return true;
    return false;
}

bool check(int n, int k, int mid) {
    for (int i = 1; i <= n; ++i) {
        if (dfs(n, i, k, mid)) return true;
    }
    return false;
}

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

    int l = 1, r = 2e8;
    while (l <= r) {
        int mid = (l + r) / 2;
        if (check(n, k, mid)) l = mid+1;
        else r = mid-1;
    }

    // cout << "----- ";
    cout << r << endl;
}

int main()
{
```

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

CF1286B Numbers on Tree

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

using namespace std;

const int maxn = 2000 + 5;
vector<int> vec[maxn];
int w[maxn], sz[maxn];

bool dfs(int u, int fa) {
    sz[u] = 1;
    for (int i : vec[u]) {
        if (i == fa) continue;
        if (!dfs(i, u)) return false;
        sz[u] += sz[i];
    }

    if (w[u] > sz[u]-1) return false;
    return true;
}

int f[maxn];
set<int> s;

void dfs_2(int u, int fa) {
    int cnt = 0, t;
    for (int i : s) {
        if (cnt == w[u]) { t = i; break; }
        ++cnt;
    }
    f[u] = t, s.erase(t);

    for (int i : vec[u]) {
        if (i == fa) continue;
        dfs_2(i, u);
    }
}

int main()
{
    int n; cin >> n;
    int root;
    for (int i = 1; i <= n; ++i) {
        int a; cin >> a >> w[i];
        if (!a) root = i;
    }
}
```

```

        else vec[a].push_back(i);
    }

    if (!dfs(root, -1)) { cout << "NO" << endl; return 0; }

    cout << "YES" << endl;

    for (int i = 1; i <= n; ++i) s.insert(i);
    dfs_2(root, -1);
    for (int i = 1; i <= n; ++i) cout << f[i] << " "; cout << endl;
    return 0;
}

```

CF30C Shooting Gallery

```

#include <bits/stdc++.h>

using namespace std;

typedef long long LL;
const int maxn = 1000 + 5;
struct node {
    int x, y, t;
    double p;
    bool operator < (const node& p) const { return t < p.t; }
} w[maxn];
double f[maxn];

bool check(node a, node b) {
    int x1 = a.x, y1 = a.y, t1 = a.t, x2 = b.x, y2 = b.y, t2 = b.t;
    int dx = x1 - x2, dy = y1 - y2, dt = t1 - t2;
    return (LL)dx*dx + (LL)dy*dy <= (LL)dt*dt;
}

int main()
{
    int n; cin >> n;
    for (int i = 1; i <= n; ++i) {
        cin >> w[i].x >> w[i].y >> w[i].t >> w[i].p;
    }
    sort(w+1, w+n+1);

    double res = 0;
    for (int i = 1; i <= n; ++i) {
        for (int j = 1; j <= i-1; ++j) {
            if (check(w[j], w[i])) f[i] = max(f[i], f[j]);
        }
        f[i] += w[i].p;
        res = max(res, f[i]);
    }
}

```

```
printf("%.8f\n", res);  
return 0;  
}
```

CF1884B Haunted House

```
#include <bits/stdc++.h>  
  
using namespace std;  
  
typedef long long LL;  
const int maxn = 1e5 + 5;  
char s[maxn];  
  
void solve() {  
    int n; cin >> n >> (s+1);  
  
    LL res = 0;  
    for (int i = n, j = n; i >= 1; --i) {  
        while (j >= 1 && s[j] != '0') --j;  
        if (j >= 1) {  
            res += i-j;  
            cout << res << " ";  
            --j;  
        }  
        else {  
            for (int k = i; k >= 1; --k) cout << -1 << " ";  
            break;  
        }  
    }  
    cout << endl;  
}  
  
int main()  
{  
    int T; cin >> T;  
    while (T -- ) solve();  
    return 0;  
}
```

CF1903D1 Maximum And Queries (easy version)

```
#include <bits/stdc++.h>  
  
using namespace std;  
  
typedef long long LL;  
const int maxn = 1e5 + 5;
```

```
LL w[maxn], a[maxn];

LL solve(int n, LL k) {
    for (int i = 1; i <= n; ++i) a[i] = w[i];

    LL res = 0;
    for (int i = 61; i >= 0; --i) {
        LL sum = 0;
        for (int j = 1; j <= n; ++j) {
            if ((a[j]>>i)&1) continue;
            sum += (1LL<<i) - a[j]%(1LL<<i);
            if (sum > k) break;
        }
        if (sum > k) continue;

        k -= sum; res += (1LL<<i);
        for (int j = 1; j <= n; ++j) {
            if ((a[j]>>i)&1) continue;
            a[j] += (1LL<<i) - a[j]%(1LL<<i);
        }
    }
    return res;
}

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

    while (T -- ) {
        LL k; cin >> k;
        // cout << "----- ";
        cout << solve(n, k) << endl;
    }
    return 0;
}
```