

三分

人员

洪晨栋、洪晨棋、崔宸赫、于家瑞、郭栩睿、陶汇笙 到课

上周作业检查

上周作业链接: <https://cppoj.kids123code.com/contest/952>

2025-1007周六10:30 (综合练习)

刷新

#	用户名	姓名	编程分	时间	A	B	C	D	E
1	hongchendong	洪晨栋	400	7284	100	100	100	100	
2	hongchenqi	洪晨棋	400	11198	100	100	100	100	
3	yujiarui	于家瑞	300	2892	100	100	100		
4	guoxurui	郭栩睿	300	2908	100	100	100		
5	cuichenhe	崔宸赫	200	2852	100		100		

本周作业

<https://cppoj.kids123code.com/contest/1107> (课上讲了 A ~ C 这些题, 课后作业是 D 题)

课堂表现

今天上课讲了 三分 这个知识点, 同学们课上做题整体做的都不是很好, 建议同学们课下把这几道题再做一遍。

课堂内容

ABA (上周作业)

枚举第 i 项作为中间的第二项, 分别看 i 前面和后面有多少 A/B/C/.../Z, 利用前缀和可以 $O(1)$ 看

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

using namespace std;

typedef long long LL;
const int N = 2e5 + 5, M = 26 + 5;
char s[N];
int pre[N][M], suf[N][M];

int main()
{
    cin >> (s+1);
    int n = strlen(s+1);
    for (int i = 1; i <= n; ++i) {
        for (int j = 'A'; j <= 'Z'; ++j) pre[i][j-'A'] = pre[i-1][j-'A'] + (s[i]==j);
```

```

}
for (int i = n; i >= 1; --i) {
    for (int j = 'A'; j <= 'Z'; ++j) suf[i][j-'A'] = suf[i+1][j-'A'] + (s[i]==j);
}

LL res = 0;
for (int i = 2; i <= n-1; ++i) {
    for (int j = 0; j < 26; ++j) res += (LL)pre[i-1][j]*suf[i+1][j];
}
cout << res << endl;
return 0;
}

```

【模板】三分 | 函数

三分 模板题

三分: 一般是针对单峰题用的

```

#include <bits/stdc++.h>

using namespace std;

const int maxn = 1e4 + 5;
const double eps = 1e-9;
int a[maxn], b[maxn], c[maxn];
int n;

double calc(double mid) {
    double res = -1e18;
    for (int i = 1; i <= n; ++i) {
        res = max(res, a[i]*mid*mid + b[i]*mid + c[i]);
    }
    return res;
}

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

    double l = 0, r = 1000;
    while (r-l > eps) {
        double lmid = l + (r-l)/3, rmid = r - (r-l)/3;
        if (calc(lmid) < calc(rmid)) r = rmid;
        else l = lmid;
    }
    printf("%.4f\n", calc(l));
}

int main()
{

```

```

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

```

[CSP-J2022 山东] 宴会

把所有值全部乘 2, 方便后面避免小数运算, 最后输出结果的时候除 2 即可

很明显, 位置在中间某个位置的时候是最好的, 再最左最右都不好, 因此可以三分做

```

#include <bits/stdc++.h>

using namespace std;

const int maxn = 2e5 + 5;
struct node {
    int x, t;
    bool operator < (const node& p) const { return x < p.x; }
} w[maxn];
int n;

int calc(int mid) {
    int res = 0;
    for (int i = 1; i <= n; ++i) res = max(res, abs(w[i].x-mid)+w[i].t);
    return res;
}

void solve() {
    cin >> n;
    for (int i = 1; i <= n; ++i) cin >> w[i].x, w[i].x *= 2;
    for (int i = 1; i <= n; ++i) cin >> w[i].t, w[i].t *= 2;
    sort(w+1, w+n+1);

    int l = w[1].x, r = w[n].x;
    while (r - l >= 10) {
        int lmid = l + (r-l)/3, rmid = r - (r-l)/3;
        if (calc(lmid) < calc(rmid)) r = rmid;
        else l = lmid;
    }

    int id = l;
    for (int i = l+1; i <= r; ++i) {
        if (calc(i) < calc(id)) id = i;
    }

    if (id&1) cout << id/2 << ".5" << endl;
    else cout << id/2 << endl;
}

int main()

```

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