

2026 年 6 月 SACCC 认证三级 题解

注：本题解中的所有参考代码均省略文件读写。

产量 (cpu)

计算所有 a_i 的总和即可。

```
#include <bits/stdc++.h>
using namespace std;

int main() {
    int n = 0, x = 0, s = 0;
    cin >> n;
    for (int i = 1; i <= n; ++i) {
        cin >> x;
        s += x;
    }
    cout << s << endl;
    return 0;
}
```

水费 (bill)

- 如果选择 A 公司，则自来水费用为 $a \times x$ ；
- 如果选择 B 公司，则自来水费用为 $b + d \times \max(x - c, 0)$ 。

取两者的最小值作为答案。

```
#include <bits/stdc++.h>
using namespace std;

int main() {
    int a = 0, b = 0, c = 0, d = 0, x = 0;
    cin >> a >> b >> c >> d >> x;
    int A = a * x;
    int B = b + max(x - c, 0) * d;
    cout << min(A, B) << endl;
    return 0;
}
```

排行 (rank)

用一个数组存储排行榜上的模型的得分，并输出第一个得分低于 x 分的编号。

```
#include <bits/stdc++.h>
using namespace std;
```

```

int x = 0, a[11] = {};

int main() {
    for (int i = 1; i <= 10; ++i) {
        cin >> a[i];
    }
    cin >> x;
    for (int i = 1; i <= 10; ++i) {
        if (x > a[i]) {
            cout << i << endl;
            return 0;
        }
    }
    cout << -1 << endl;
    return 0;
}

```

游戏 (game)

用一个数组存储所有同学的得分，根据题意模拟游戏流程，统计所有人的得分。

```

#include <bits/stdc++.h>
using namespace std;

int n = 0, m = 0;
int s[101] = {};

int main() {
    cin >> n >> m;
    for (int t = 1; t <= m; ++t) {
        int x = 0;
        cin >> x;
        for (int i = 1; i <= n; ++i) {
            int y = 0;
            cin >> y;
            ++s[x == y ? i : x];
        }
    }
    for (int i = 1; i <= n; ++i) {
        cout << s[i] << " \n"[i == n];
    }
    return 0;
}

```

时间 (time)

一种简单的做法是：将起始时间和结束时间全部转化为秒数，计算它们的差值，再转换回时间表示即可。

```
#include <bits/stdc++.h>
using namespace std;

int parse(const string& s) {
    int hh = (s[0] - '0') * 10 + (s[1] - '0');
    int mm = (s[3] - '0') * 10 + (s[4] - '0');
    int ss = (s[6] - '0') * 10 + (s[7] - '0');
    return hh * 3600 + mm * 60 + ss;
}

int main() {
    string a, b;
    cin >> a >> b;
    int x = parse(a);
    int y = parse(b);
    int t = y - x;
    int hh = t / 3600;
    int mm = (t - hh * 3600) / 60;
    int ss = t % 60;
    cout << setw(2) << setfill('0') << hh << ":";
    cout << setw(2) << setfill('0') << mm << ":";
    cout << setw(2) << setfill('0') << ss << endl;
    return 0;
}
```

道路 (road)

用一个二维数组存储每个位置的居民数量。横向道路总共有 n 条，纵向道路总共有 m 条，因此总共只有 $n \times m$ 种不同的改造方案。本题的数据范围较小，直接枚举所有的情况，计算距离之和并取最优解即可。时间复杂度 $O(n^2m^2)$ 。

```
#include <bits/stdc++.h>
using namespace std;
constexpr int INF = 0x3f3f3f3f;

int n = 0, m = 0, ans = INF;
int a[51][51] = {};

int work(int x, int y) {
    int ret = 0;
    for (int i = 1; i <= n; ++i) {
        for (int j = 1; j <= m; ++j) {
```

```

        ret += a[i][j] * min(abs(i - x), abs(j - y));
    }
}
return ret;
}

int main() {
    cin >> n >> m;
    for (int i = 1; i <= n; ++i) {
        for (int j = 1; j <= m; ++j) {
            cin >> a[i][j];
        }
    }
    for (int i = 1; i <= n; ++i) {
        for (int j = 1; j <= m; ++j) {
            ans = min(ans, work(i, j));
        }
    }
    cout << ans << endl;
    return 0;
}

```