

# 端午欢乐赛题目讨论

# ➤ 问题1：采访

## A 采访

时间限制：- MS 空间限制：- KB

评测说明：1s 256MB

### 问题描述

有 $n$ 个人站成一排，xxx250要采访其中一些人“你幸福吗？”。但是相邻两个人不能都被采访，否则这两个人就会因为相互影响而说不真实的回答。xxx250想知道一共有多少种满足条件的采访方法呢？(可以不选)

### 输入格式

一行一个 $n$ 。

对于30%的数据， $1 \leq n \leq 10$

对于50%的数据， $1 \leq n \leq 100$

对于70%的数据， $1 \leq n \leq 1000$

对于100%的数据， $1 \leq n \leq 30000$

### 输出格式

一行表示答案。

### 样例输入 1

1

### 样例输入 2

3

### 样例输出 1

2

### 样例输出 2

5

## ➤ 问题1：采访

---

题目解法本来就是一个斐布那切数列，但是需要写高精度版本。

## ➤ 问题1：采访

参考代码：

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

using namespace std;

int a[10000] = {1, 1};
int b[10000] = {1, 1};
int c[10000];
int main() {
    int n;
    cin >> n;
    for (int i = 1; i <= n; ++i) {
        int len = max(a[0], b[0]) + 1;
        c[1] = 0;
        for (int j = 1; j <= len; ++j) {
            c[j + 1] = (c[j] + a[j] + b[j]) / 10;
            c[j] = (c[j] + a[j] + b[j]) % 10;
        }
        while (len > 1 && c[len] == 0) len--;
        for (int j = 1; j <= b[0]; ++j) {
            a[j] = b[j];
        }
        a[0] = b[0];
        for (int j = 1; j <= len; ++j) {
            b[j] = c[j];
        }
        b[0] = len;
    }
    for (int i = b[0]; i >= 1; --i) {
        cout << b[i];
    }
    return 0;
}
```

## 问题2：密码锁

### B 密码锁

时间限制：- MS 空间限制：- KB

评测说明：1s 256mb

#### 问题描述

果老师听说何老板家里有矿之后，就偷偷来到何老板家门口，但他在何老板家的门前发现了一个密码锁，他必须解开这个锁才能进入何老板家。于是他就向你请教。请快速解决这个问题，何老板还有1秒就会回家了。

锁上有两行，第一行一个数字 $N$ 。第二行是一串字符串 $S$ ，字符串由许多子串构成，每个子串的格式均是 $XA$ 其中 $X$ 是一个运算符， $A$ 是一个数字。 $X$ 可能是 $*$ ， $+$ ， $-$ ， $\%$ ， $^$ （ $^$ 表示次方）。

现在需要把数字 $N$ 代入字符串 $S$ ，从左到右进行运算。密码就是运算结果的绝对值。

题目保证运算过程中 $N$ 在 $(-2147483648-2147483647)$ 范围内， $^$ 后面的数字只能为2。运算过程从左至右，不满足运算的优先级（详见样例）

#### 输入格式

两行，第一行一个正整数 $N$

第二行是一个字符串 $S(|S| \leq 6 * 10^5)$

#### 输出格式

一个数，表示运算结果的绝对值

#### 样例输入

```
5
-7*3
```

#### 样例输出

```
6
```

#### 提示

5-7=-2

-2\*3=-6

|-6|=6

## ➤ 问题2：密码锁

---

就按题意模拟过去，注意是有可能爆int的。

## ➤ 问题2：密码锁

参考代码：

```
#include <bits/stdc++.h>
using namespace std;
typedef long long ll;
const int N = 1e6 + 5;
char st[N];
ll n;
int main() {
    scanf("%lld", &n);
    scanf("%s", st+1);
    int len = strlen(st + 1), id = 1;
    while(id <= len) {
        char now = st[id]; ll num = 0; id++;
        while(1) {
            if(st[id] > '9' || st[id] < '0') break;
            num = num * 10 + st[id] - '0', id++;
        }
        if(now == '+') n = n + num;
        if(now == '-') n = n - num;
        if(now == '*') n = n * num;
        if(now == '%') n = n % num;
        if(now == '^') n = n * n;
    }
    printf("%lld", n < 0 ? -n : |n);
    return 0;
}
```

# 问题3: 大吉大利，今晚吃鸡

## 大吉大利，今晚吃鸡

时间限制：- MS 空间限制：- KB

评测说明：1s 256mb

### 问题描述

有  $n$  个oier，编号为  $1 \sim n$ ，第  $i$  个oier有  $a[i]$  枚金币，若第一个oier金币数大于 0，则可以选择一个  $i (2 \leq i \leq n)$ ，然后弃置 1 枚金币，让第  $i$  个oier弃置  $b[i]$  枚金币，若第  $i$  个oier金币数少于  $b[i]$  则弃置所有金币。现需要让第 1 个oier弃置最少的金币，成为唯一的金币数最多的oier。

### 输入格式

第一行一个正整数  $n (n \leq 100000)$ ，第二行  $n$  个正整数  $a_i (1 \leq a_i \leq 100000)$ ，第三行， $n - 1$  个正整数，第  $i$  个表示  $b_{i+1} (1 \leq b_{i+1} \leq 100000)$ 。

### 输出格式

输入需要弃置的最少金币，如果无解输出 -1。

### 样例输入 1

```
3
3 3 3
3 3
```

### 样例输入 2

```
2
3 10
5
```

### 样例输入 3

```
3
3 2 1
1 1
```

### 样例输出 1

```
2
```

### 样例输出 2

```
2
```

### 样例输出 3

```
0
```



## ➤ 问题3：大吉大利，今晚吃鸡

---

思路：

二分答案，然后判断可行性。

## ➤ 问题3：大吉大利，今晚吃鸡

参考代码：

```
#include <bits/stdc++.h>
using namespace std;
const int maxn = 1e5 + 10;
int n, v[maxn], b[maxn];
int isok(int times) {
    int cur = v[1] - times;
    int num = 0;
    for(int i = 2; i <= n; ++i)
        if(cur <= v[i]) num += (v[i] - cur) / b[i] + 1;
    return num <= times;
}
int main() {
    scanf("%d", &n);
    for(int i = 1; i <= n; ++i) scanf("%d", &v[i]);
    for(int i = 2; i <= n; ++i) scanf("%d", &b[i]);
    int l = 0, r = v[1];
    while(l < r) {
        int mid = (l + r) >> 1;
        if(!isok(mid)) l = mid + 1;
        else r = mid;
    }
    cout << (l == v[1] ? -1 : l) << endl;
    return 0;
}
```

## ➤ 问题4：石子游戏

### D 石子游戏

时间限制：- MS 空间限制：- KB

评测说明：1s 256mb

#### 问题描述

有  $n$  堆石头，第  $i$  堆石头有  $a_i$  个石子，两人轮流操作，每次操作先选择一堆石头，再从这堆石头中取走一个石子，如果此次操作取完了被选择的这堆石头的最后一个石子，操作者得一分。当所有石子被取走时，游戏结束。输出先手最大得分。

#### 输入格式

第一行一个整数  $n$  ( $1 \leq n \leq 10^5$ )，第二行  $n$  个整数，第  $i$  个数字表示  $a_i$  ( $2 \leq a_i \leq 10^9$ )。

#### 输出格式

输出一个整数表示先手最大得分。

#### 样例输入 1

1  
3

#### 样例输入 2

2  
2 2

#### 样例输出 1

1

#### 样例输出 2

0

## ➤ 问题4：石子游戏

---

偶数堆先取必输，如果我当前取偶数堆，那对手再取一次该堆就能使局势保持不变，因此只能先取奇数堆，判断一下谁会先将奇数堆取完谁就输，并且要么全胜要么全败。

## ➤ 问题4：石子游戏

参考代码：

```
#include <bits/stdc++.h>
using namespace std;
int n, m;
int main() {
    cin >> n;
    int cnt = 0;
    int x;
    for(int i = 1; i <= n; i++) {
        scanf("%d", &x);
        if(x % 2) cnt++;
    }
    if(cnt % 2) cout << n;
    else cout << 0;
    return 0;
}
```

# 问题5：数字转换

## A 数字转换

时间限制：- MS 空间限制：- KB

评测说明：1s 256MB

### 问题描述

果老师有  $N$  个数字  $a_1, a_2, \dots, a_N$ 。果老师现在想把所有数字变成相等的整数。

每个数字最多只能变化一次。把一个整数  $x$  变为  $y$  的花费是  $(x - y)^2$  软妹币。

果老师已经非常穷困潦倒了，希望你能帮他计算一下最少的花费是多少。

### 输入格式

输入包括两行，第一行一个正整数  $N$  ( $1 \leq N \leq 100$ )。第二行包括  $N$  个整数  $a_i$  ( $-100 \leq a_i \leq 100$ )。

### 输出格式

输出一个整数，表示最少的花费使得所有数都相等。

### 样例输入 1

```
3
1 1 3
```

### 样例输入 2

```
3
4 2 5
```

### 样例输出 1

```
3
```

### 样例输出 2

```
5
```

### 提示

对于样例1：

把所有数字都变为2，所需的花费是  $(1 - 2)^2 + (1 - 2)^2 + (3 - 2)^2 = 3$ ，这是最少花费的方案。

## ➤ 问题5：数字转换

---

枚举最后所有数变成的数，然后计算对应的代价，维护最小值即可。

## ➤ 问题5：数字转换

参考代码：

```
#include <bits/stdc++.h>
using namespace std;
const int mx = 100 + 5, inf = 1e9;
int a[mx];
int main() {
    int n, s, ans = inf;
    scanf("%d", &n);
    for(int i = 1; i <= n; i++) scanf("%d", a + i);
    for(int x = -100; x <= 100; x++) {
        s = 0;
        for(int i = 1; i <= n; i++)
            s += (x - a[i]) * (x - a[i]);
        ans = min(ans, s);
    }
    printf("%d\n", ans);
    return 0;
}
```



# ➤ 问题6：插入括号

## B 插入等号

时间限制：- MS 空间限制：- KB

评测说明：1s 256MB

### 问题描述

果老师给你一个只包含字符'1'到'9'的字符串 $S$ 。现在你可以在这个字符串当中去插入若干(可以不插)个'+'号, 每种插入之后你都可以得到一个算式, 每个算式都有一个结果, 果老师希望你现在能计算所有结果之和。

### 输入格式

输入包括一个字符串 $S(1 \leq |S| \leq 10)$ ,  $S$ 当中只包含'1'到'9'。

### 输出格式

输出所有可能的算式结果之和。

### 样例输入 1

125

### 样例输出 1

176

### 样例输入 2

9999999999

### 样例输出 2

12656242944

### 提示

对于样例1:

这里有4种可能的算式"125", "1 + 25", "12 + 5", "1 + 2 + 5",

所有算式结果之和为 $125 + 26 + 17 + 8 = 176$ .

## ➤ 问题6：插入括号

---

Dfs暴力搜索所有插入的情况，然后对所有情况的结果求和。

## ➤ 问题6：插入括号

参考代码：

```
#include <bits/stdc++.h>
#define ll long long
using namespace std;
ll a[15], n, ans;
void dfs(ll x, ll s, ll tot) {
    if(x == n + 1) {
        ans += tot + s;
        return;
    }
    s = s * 10 + a[x];
    dfs(x + 1, 0, tot + s);
    dfs(x + 1, s, tot);
}
int main() {
    char x;
    while(scanf("%c",&x) != EOF) {
        if(x <= '9' && x >= '0') a[++n] = x - '0';
        else break;
    }
    dfs(1, 0, 0);
    printf("%lld", ans / 2);
    return 0;
}
```

# 问题7：吃雪糕

## C 吃雪糕

时间限制：- MS 空间限制：- KB

评测说明：1s 256MB

### 问题描述

夏天来了，当然是要吃雪糕呀！

现在有 $N$ 台冰箱在南渝小卖部排列为一行。最开始第 $i$ 台冰箱内有 $a_i$ 支雪糕。

果老师现在可以执行若干次以下操作：

选择一台有雪糕的冰箱，从中拿出一支吃掉。

果老师在他执行完若干次操作之后，使得任意两台相邻的冰箱中的雪糕总数不超过 $x$ 。

果老师想问你最少他需要做多少次吃雪糕的操作。

### 输入格式

输入的第一行包括两个整数 $N$ 和 $x$  ( $2 \leq N \leq 10^5, 0 \leq x \leq 10^9$ )

第二行包含 $N$ 个整数 $a_i$  ( $0 \leq a_i \leq 10^9$ )。

### 输出格式

输出最少的操作次数，使其满足果老师的要求。

### 样例输入 1

```
3 3
2 2 2
```

### 样例输入 2

```
6 1
1 6 1 2 0 4
```

### 样例输出 1

```
1
```

### 样例输出 2

```
11
```

## ➤ 问题7：吃雪糕

因为我们不关心最后每个冰箱里分别有多少雪糕，只是要让相邻两个位置之和都满足要求就可以了。

所以我们从前往后扫描，相邻位置之和如果大于 $x$ 了，我们就把多余的部分在较后的一个位置上减掉，如果该位置减掉之后变为 $<0$ ，就把该位置置为0即可。

## ➤ 问题7：吃雪糕

参考代码：

```
#include <bits/stdc++.h>
using namespace std;
long long a[100005], ans, n, x;

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

# ➤ 问题8：构造矩形

## D 构造矩形

时间限制：- MS 空间限制：- KB

评测说明：1s,256m

### 问题描述

何老板有N根直木棍，第i根木棍的长度为 $A_i$   
何老板要你从中选四根木棍，并使得他们能构成一个矩形(可以是正方形)。请你计算出最大可能得到的矩形的面积。

### 输入格式

第一行，一个整数N  
第二行，N个空格间隔的整数，表示木棍的长度

### 输出格式

一行，一个整数，表示矩形的面积

### 样例输入 1

```
6
3 1 2 4 2 1
```

### 样例输入 2

```
4
1 2 3 4
```

### 样例输入 3

```
10
3 3 3 3 4 4 4 5 5 5
```

### 提示

$4 \leq N \leq 10^5$   $1 \leq A_i \leq 10^9$   $A_i$ 是整数

### 样例输出 1

```
2
样例解释：构造1x2的矩形
```

### 样例输出 2

```
0
```

### 样例输出 3

```
20
```

## ➤ 问题8：构造矩形

---

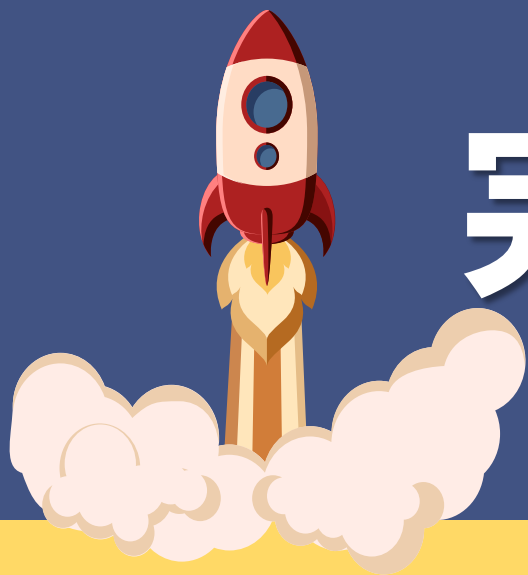
排序之后，选取最大的两组两两相同的木棍构成矩形即可。



## ➤ 问题8：构造矩形

参考代码：

```
#include <bits/stdc++.h>
using namespace std;
int n, a[100005], ans, k = 1;
long long s[3];
bool cmp(int x, int y) { return x > y; }
int main() {
    cin >> n;
    for (int i = 1; i <= n; i++) {
        cin >> a[i];
    }
    sort(a + 1, a + 1 + n, cmp);
    for (int i = 1; i <= n; i++) {
        if (a[i + 1] == a[i]) {
            s[k++] = a[i];
            i++;
        }
        if (k > 2) {
            break;
        }
    }
    cout << s[1] * s[2];
    return 0;
}
```



# 完！

以梦为码 心之所往

