

L1532：赌徒 ☆☆☆

题目描述

有 n 个赌徒打算赌一局。规则是：每人下一个赌注，赌注为非负整数，且任意两个赌注都不相同。胜者为赌注恰好是其余任意三个人的赌注之和的那个人。如果有多个胜者，我们取赌注最大的那个为最终胜者。

例如， A, B, C, D, E 分别下赌注为 2、3、5、7、12，最终胜者是 E ，因为 $12 = 2 + 3 + 7$ 。

输入输出格式

输入：输入包含多组测试数据。每组首先输入一个整数 n ($1 \leq n \leq 1000$)，表示赌徒的个数。接下来 n 行每行输入一个非负整数 b ($0 \leq b < 32768$)，表示每个赌徒下的赌注。当 $n = 0$ 时，输入结束。

输出：对于每组输入，输出最终胜者的赌注，如果没有胜者，则输出 `no solution`。

输入示例	输出示例
5	12
2	no solution
3	
5	
7	
12	
5	
2	
16	
64	
256	
1024	
0	

样例解释

- 第一组数据：**赌注为 2、3、5、7、12
 $12 = 2 + 3 + 7$ ，满足条件，所以输出 12。
- 第二组数据：**赌注为 2、16、64、256、1024
没有任何一个赌注等于其他三个赌注之和，所以输出 `no solution`。

算法分析

1. 问题分析

我们需要找最大的 x ，使得存在三个数字 y, z, w ，使得 $x = y + z + w$ 。如果使用暴力做法，分别枚举 x, y, z, w ，时间复杂度会达到 $O(n^4)$ ，超时。

2. 优化思路

- **预处理：** 枚举 y, z 通过二重循环得到所有的 $y + z$ 和，并记录每个和出现的次数为 $cnt[y + z]$ 。
- **枚举优化：** 再枚举 x, w ，通过公式 $x = y + z + w$ ，得到 $y + z = x - w$ ，在预处理的和中查找是否存在 $x - w$ 。
- **避免重复：** 使用您的判断逻辑来避免重复使用同一个元素：

– 避免重复元素的逻辑分析：

对于等式 $x - w = y + z$ ，我们需要确保 x, y, z, w 是四个不同的赌注。

当 w 被重复使用时，会出现以下情况：

$$x - w = y + z$$

$$\text{如果 } y = w \Rightarrow x - w = w + z \Rightarrow x - 2w = z$$

- * 如果 $x - 2w > 0$ 且 $cnt[x - 2w] = 1$ ，则必然会出现 w 被重复使用的情况
- * 如果 $x - 2w > 0$ 且 $cnt[x - 2w] > 1$ ，则存在多组 $\{a, b\}$ 满足条件，可以避免重复
- * 如果 $x - 2w \leq 0$ ，则不会出现重复问题

代码实现

预处理 $y + z$

```
for (int i = 0; i < n; i++) {
    for (int j = i; j < n; j++) {
        if (i == j) continue;
        s[a[i] + a[j]]++;
    }
}
```

枚举 x, w

```
for (int i = 0; i < n; i++) {
    for (int j = 0; j < n; j++) {
        if (i == j) continue;
        int ns = a[i] - a[j]; // x - w
        int t = a[i] - 2 * a[j]; // z = x - 2w
        if (ns <= 0) continue;
        if (s[ns] <= 0) continue;
        if (t > 0 && s[ns] <= cnt[t]) continue;
        res = max(res, a[i]);
    }
}
```

拓展思考

- 如果改成七个人改成寻找满足的 $A = B + C + D + E + F + G$ 该如何实现？