

青少年软件编程（C 语言）等级考试试卷（四级）

1、人以群分

社交网络中我们给每个人定义了一个“活跃度”，现希望根据这个指标把人群分为两大类，即外向型（outgoing，即活跃度高的）和内向型（introverted，即活跃度低的）。要求两类人群的规模尽可能接近，而他们的总活跃度差距尽可能拉开。

时间限制：5000

内存限制：65536

输入

输入第一行给出一个正整数 N ($2 \leq N \leq 10^5$)。随后一行给出 N 个正整数，分别是每个人的活跃度，其间以空格分隔。题目保证这些数字以及它们的和都不会超过 2^{31} 。

输出

按下列格式输出：Outgoing #: N_1 Introverted #: N_2 Diff = N_3 其中 N_1 是外向型人的个数； N_2 是内向型人的个数； N_3 是两群人总活跃度之差的绝对值。

样例输入

样例 1：

10

23 8 10 99 46 2333 46 1 666 555

样例 2：

13

110 79 218 69 3721 100 29 135 2 6 13 5188 85

样例输出

样例 1：

Outgoing #: 5

Introverted #: 5

Diff = 3611

样例 2：

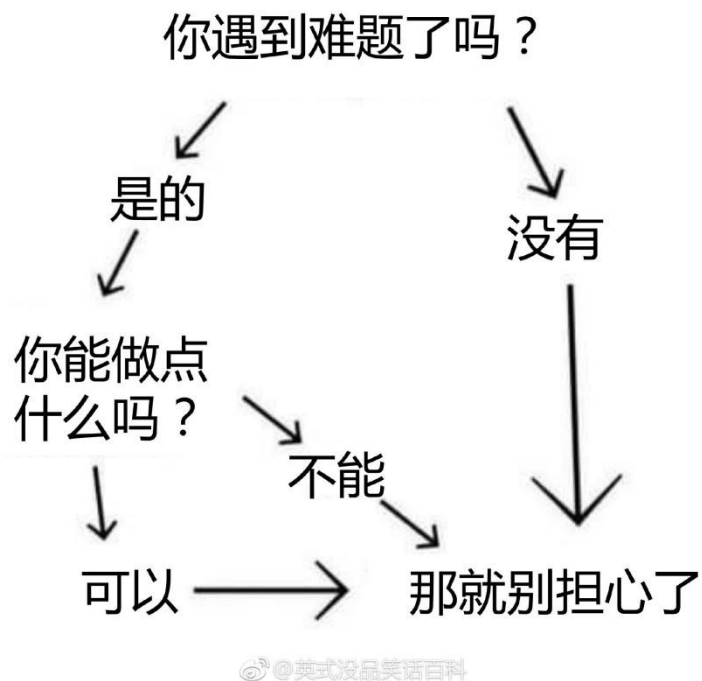
Outgoing #: 7

Introverted #: 6

Diff = 9359

2、那就别担心了

下图转自“英式没品笑话百科”的新浪微博——所以无论有没有遇到难题，其实都不用担心。



博主将这种逻辑推演称为“逻辑自洽”，即从某个命题出发的所有推理路径都会将结论引导到同一个最终命题（开玩笑的，千万别以为这是真正的逻辑自洽的定义……）。现给定一个更为复杂的逻辑推理图，本题就请你检查从一个给定命题到另一个命题的推理是否是“逻辑自洽”的，以及存在多少种不同的推理路径。例如上图，从“你遇到难题了吗？”到“那就别担心了”就是一种“逻辑自洽”的推理，一共有 3 条不同的推理路径。

时间限制：7000

内存限制：65536

输入

输入首先在一行中给出两个正整数 N ($1 < N \leq 500$) 和 M ，分别为命题个数和推理个数。这里我们假设命题从 1 到 N 编号。接下来 M 行，每行给出一对命题之间的推理关系，

即两个命题的编号 S1 S2，表示可以从 S1 推出 S2。题目保证任意两命题之间只存在最多一种推理关系，且任一命题不能循环自证（即从该命题出发推出该命题自己）。最后一行给出待检验的两个命题的编号 A B。

输出

在一行中首先输出从 A 到 B 有多少种不同的推理路径，然后输出 Yes 如果推理是“逻辑自治”的，或 No 如果不是。 题目保证输出数据不超过 10⁹。

样例输入

样例 1：

7 8

7 6

7 4

6 5

4 1

5 2

5 3

2 1

3 1

7 1

样例 2：

7 8

7 6

7 4

6 5

4 1

5 2

5 3

6 1

3 1

7 1

样例输出

样例 1：

3 Yes

样例 2：

3 No

3、凑零钱

韩梅梅喜欢满宇宙到处逛街。现在她逛到了一家火星店里，发现这家店有个特别的规矩：你可以用任何星球的硬币付钱，但是绝不找零，当然也不能欠债。韩梅梅手边有 10^4 枚来自各个星球的硬币，需要请你帮她盘算一下，是否可能精确凑出要付的款额。

时间限制：7000

内存限制：65535

输入

输入第一行给出两个正整数： N ($\leq 10^4$) 是硬币的总个数， M ($\leq 10^2$) 是韩梅梅要付的款额。第二行给出 N 枚硬币的正整数面值。数字间以空格分隔。

输出

在一行中输出硬币的面值 $V_1 \leq V_2 \leq \dots \leq V_k$ ，满足条件 $V_1 + V_2 + \dots + V_k = M$ 。数字间以 1 个空格分隔，行首尾不得有多余空格。若解不唯一，则输出最小序列。若无解，则输出 `No Solution`。注：我们说序列 $\{A[1], A[2], \dots\}$ 比 $\{B[1], B[2], \dots\}$ “小”，是指存在 $k \geq 1$ 使得 $A[i] = B[i]$ 对所有 $i < k$ 成立，并且 $A[k] < B[k]$ 。

样例输入

样例 1：

8 9

5 9 8 7 2 3 4 1

样例 2：

4 8

7 2 4 3

样例输出

样例 1：

1 3 5

样例 2：

No Solution

4、拼题 A 打卡奖励

拼题 A 的教超搞打卡活动，指定了 N 张打卡卷，第 i 张打卡卷需要 m_i 分钟做完，完成后可获得 c_i 枚奖励的金币。活动规定每张打卡卷最多只能做一次。活动总时长为 M 分钟。请你算出最多可以赢得多少枚金币？

时间限制：7000

内存限制：524288

输入

输入首先在第一行中给出两个正整数 N ($\leq 10^3$) 和 M ($\leq 365 \times 24 \times 60$)，分别对应打卡卷的数量和以“分钟”为单位的活动总时长（不超过一年）。随后一行给出 N 张打卡卷要花费的时间 m_i (≤ 600)，最后一行给出 N 张打卡卷对应的奖励金币数量 c_i (≤ 30)。上述均为正整数，一行内的数字以空格分隔。

输出

在一行中输出最多可以赢得的金币数量。

样例输入

5 110

70 10 20 50 60

28 1 6 18 22

样例输出

40

提示

样例解释：选择最后两张卷子，可以在 $50+60=110$ 分钟内获得 $18+22=40$ 枚金币。