1、哈特进来一直沉迷于玩GT游戏。游戏开始时，给玩家一个n\*n的方格矩阵。每个方格又被划分为4个三角形，每个三角形标有一个数字（范围从0到9），分别将这些三角形成为左三角形、右三角形、顶三角形和底三角形。图1就是一个2\*2方格初始状态。

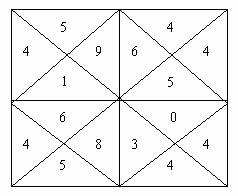


图1

要求玩家通过移动方格达到结局状态。在结局状态，任意俩个相邻方格中，相邻三角形都标有相同数字。图2就是一个结局状态。

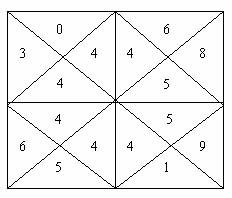


图2

游戏其实不难，但哈特不擅长玩，有时面对复杂问题，他花很长时间都没成功，最后发现有的局面根本无法完成。所以他希望对于每个局面开始时，就被告知是否可以完成转变。

输入：

输入包括多个游戏测试。每个例子第一行是一个整数n（0<=n<=5），给出游戏中方格矩阵的大小。若遇到0表示输入结束。

在接下来n\*n行中，描述这些三角形标记上的数字。每一行，有4个数字，顺序是顶三角形、右三角形、底三角形和左三角形。

输出：

对于每个例子输出是否可以完成的结论。输出Game+游戏编号，一个冒号和判断结果，若能完成输出“YES”，否则“NO”。

输入样例：

2

5       9       1       4

4       4       5       6

6       8       5       4

0       4       4       3

2

1       1       1       1

2       2       2       2

3       3       3       3

4       4       4       4

0

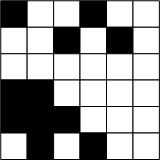
输出样例：

Game 1:YES

Game 2:NO

2、输入一个n\*n的黑白图像（1表示黑色，0表示白色），任务是统计其中八连块的个数。如果两个黑格子有公共边或者有公共顶点，就说它们属于同一个八连块。

如下图所示，八连块的个数为3。



Input

第一行输入一个n (1<=n<=1000)表示图的大小

接下来n行 用来表示图的组成

Output

输出八连块的个数

Sample Input

6

1  0  0  1  0  0

0  0  1  0  1  0

0  0  0  0  0  0

1  1  0  0  0  0

1  1  1  0  0  0

0  1  0  1  0  0

Sample Output

3

3、Description

乔治拿来一组等长的木棒，将它们随机地砍断，使得每一节木棍的长度都不超过50个长度单位。然后他又想把这些木棍恢复到为裁截前的状态，但忘记了初始时有多少木棒以及木棒的初始长度。请你设计一个程序，帮助乔治计算木棒的可能最小长度。每一节木棍的长度都用大于零的整数表示。

**Input**

输入包含多组数据，每组数据包括两行。第一行是一个不超过64的整数，表示砍断之后共有多少节木棍。第二行是截断以后，所得到的各节木棍的长度。在最后一组数据之后，是一个零。

**Output**

为每组数据，分别输出原始木棒的可能最小长度，每组数据占一行。

**Sample Input**

9

5 2 1 5 2 1 5 2 1

4

1 2 3 4

0

**Sample Output**

6

5