第十六届全国青少年信息学(计算机)奥林匹克竞赛

（NOI’99）

第二试试题

· 试题说明

试题名称	分值	可执行文件名	输入文件名	输出文件名	试题文本页数
棋盘分割	50	Chess.exe	Input.txt	Output.txt	2
最优连通子集	50	Subset.exe	Input.txt	Output.txt	2
内存分配	50	Memory.exe	Input.txt	Output.txt	2

· 竞赛时间：8:00—12:00

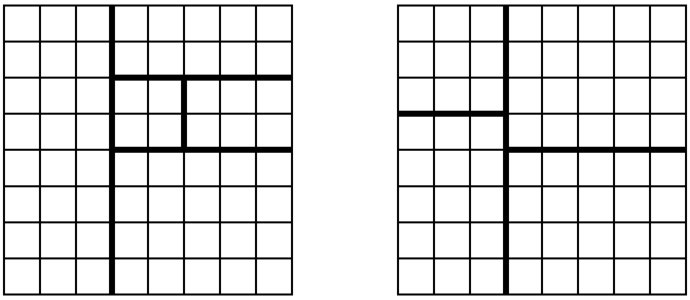
棋盘分割

Chessboard Division

Chess.{pas|bas|c}

Chess.exe

将一个 8 × 8 的棋盘进行如下分割：将原棋盘割下一块矩形棋盘并使剩下部分也是矩形，再将剩下的部分继续如此分割，这样割了(n-1)次后，连同最后剩下的矩形棋盘共有 n 块矩形棋盘。(每次切割都只能沿着棋盘格子的边进行)



允许的分割方案 不允许的分割方案

原棋盘上每一格有一个分值，一块矩形棋盘的总分为其所含各格分值之和。现在需要把棋盘按上述规则分割成 n 块矩形棋盘，并使各矩形棋盘总分的均方差最小。

均方差 $\sigma=\sqrt{\frac{\sum_{i=1}^{n}(x_i-\overline{x})^2}{n}}$ 平均值 $x\ \overline{x}=\frac{\sum_{i=1}^{n}x_i}{n}$ 形棋盘的分。

请编程对给出的棋盘及 n，求出 σ 的最小值。

输入

第 1 行为一个整数 n(1<n<15)。

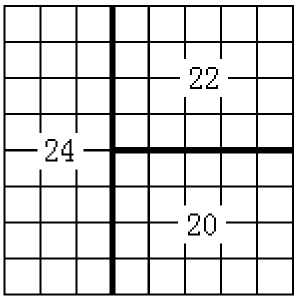
第 2 行至第 9 行每行为 8 个小于 100 的非负整数，表示棋盘上相应格子的分值。每行相邻两数之间用一个空格分隔。

输出

仅一个数，为 σ （四舍五入精确到小数点后三位）。

样例输入

3
1 1 1 1 1 1 1 3
1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 0
1 1 1 1 1 1 0 3



样例输出

1.633

最优连通子集

Optimal Connected Subset

Subset.{pas|bas|c}

Subset.exe

众所周知，我们可以通过直角坐标系把平面上的任何一个点P用一个有序数对(x,y)来唯一表示，如果x,y都是整数，我们就把点P称为整点，否则点P称为非整点。我们把平面上所有整点构成的集合记为W。

定义1 两个整点$P_1(x_1,y_1)$,$P_2(x_2,y_2)$，若$|x_1-x_2|+|y_1-y_2|=1$，则称**P_1,P_2相邻**，记作**P_1~P_2**，否则称P_1,P_2不相邻。

定义2 设点集S是W的一个有限子集，即$S=\{P_1,P_2,\ldots,P_n\}(n>=1)$，其中$P_i(1<=i<=n)$属于W，我们把S称为**整点集**。

定义3 设S是一个整点集，若点R,T属于S，且存在一个有限的点序列$Q_1,Q_2,\ldots,Q_k$满足:

1. Q_i属于S（1<=i<=k）;
2. $Q_1=R,Q_k= T$;
3. Q_i~$Q_{i+1}(1<=i<=k-1)$，即Q_i与Q_{i+1}相邻;
4. 对于任何1<=i<j<=k有$Q_i\neq Q_j$;

我们则称点R与点T在整点集S上**连通**，把点序列$Q_1,Q_2,\ldots,Q_k$称为整点集S中连接点R与点T的**一条道路**。

定义4 若整点集V满足：对于V中的任何两个整点，V中有且仅有一条连接这两点的道路，则V称为**单整点集**。

定义5 对于平面上的每一个整点，我们可以赋予它一个整数，作为该点的权，于是我们把一个整点集中所有点的权的总和称为该**整点集的权和**。

我们希望对于给定的一个单整点集V，求出一个V的最优连通子集B，满足：

1. B是V的子集
2. 对于B中的任何两个整点，在B中连通；

3. B 是满足条件(1)和(2)的所有整点集中权和最大的。

输入

第 1 行是一个整数 N，表示单整点集 V 中点的个数；

以下 N 行中，第 i 行(1<=i<=N)有三个整数，X_i,Y_i,C_i 依次表示第 i 个点的横坐标，纵坐标和权。同一行相邻两数之间用一个空格分隔。

输出

仅一个整数，表示所求最优连通集的权和。

样例输入

5
0 0 –2
0 1 1
1 0 1
0 –1 1
-1 0 1

样例输出

2

参数约定

2<=N<=1000
-10^6<=X_i,Y_i<=10^6
-100<=C_i<=100

内存分配

Memory Distribution

Memory.{pas|bas|c}

Memory.exe

内存是计算机重要的资源之一，程序运行的过程中必须对内存进行分配。

经典的内存分配过程是这样进行的：

1. 内存以内存单元为基本单位，每个内存单元用一个固定的整数作为标识，称为地址。地址从 0 开始连续排列，地址相邻的内存单元被认为是逻辑

上连续的。我们把从地址 i 开始的 s 个连续的内存单元称为首地址为 i 长度为 s 的地址片。
2. 运行过程中有若干进程需要占用内存，对于每个进程有一个申请时刻 T，需要内存单元数 M 及运行时间 P。在运行时间 P 内（即 T 时刻开始，T+P 时刻结束），这 M 个被占用的内存单元不能再被其他进程使用。

3、假设在 T 时刻有一个进程申请 M 个单元，且运行时间为 P，则：

1. 若 T 时刻内存中存在长度为 M 的空闲地址片，则系统将这 M 个空闲单元分配给该进程。若存在多个长度为 M 个空闲地址片，则系统将首地址最小的那个空闲地址片分配给该进程。
2. 如果 T 时刻不存在长度为 M 的空闲地址片，则该进程被放入一个等待队列。对于处于等待队列队头的进程，只要在任一时刻，存在长度为 M 的空闲地址片，系统**马上**将该进程取出队列，并为它分配内存单元。注意，在进行内存分配处理过程中，处于等待队列队头的进程的处理优先级最高，队列中的其它进程不能先于队头进程被处理。

现在给出一系列描述进程的数据，请编写一程序模拟系统分配内存的过程。

输入

第一行是一个数 N，表示总内存单元数（即地址范围从 0 到 N-1）
从第二行开始每行描述一个进程的三个整数 T、M、P（M<=N）。
数据已按 T 从小到大排序。
最后一行用三个 0 表示结束。
输入文件最多 10000 行，且所有数据都小于 10^9。
输入文件中同一行相邻两项之间用一个或多个空格隔开。

输出
包括 2 行。
第一行是全部进程都运行完毕的时刻。
第二行是被放入过等待队列的进程总数。

样例输入

10
1 3 10
2 4 3
3 4 4
4 1 4
5 3 4
0 0 0

样例输出

12
2

样例示例

时刻T	内存占用情况										进程事件
	0	1	2	3	4	5	6	7	8	9	
1	**A**										进程 A 申请空间（M=3,P=10）<成功>
2	**A**			**B**							进程 B 申请空间（M=4,P=3）<成功>
3	**A**			**B**							进程 C 申请空间（M=4,P=4）<失败进入等待队列>
4	**A**			**B**				D			进程 D 申请空间（M=1,P=4）<成功>
5	**A**			**C**				D			进程 B 结束，释放空间 进程 C 从等待队列取出，分配空间 进程 E 申请空间（M=3,P=4）<失败进入等待队列>
6	**A**			**C**				D			
7	**A**			**C**				D			
8	**A**			**C**				**E**			进程 D 结束，释放空间 进程 E 从等待队列取出，分配空间
9	**A**							**E**			进程 C 结束，释放空间
10	**A**							**E**			

11								**E**			进程 A 结束，释放空间
12											进程 E 结束，释放空间