# 数字三角形

## 1题目描述

7

3 8

8 1 0

2 7 4 4

4 5 2 6 5

如上图所示，从一个数字三角形的顶部走到底部有很多条不同的路径，规则是只能从当前节点走到下一层相邻的节点，即下一层的左边或右边。例如第三行第二个数字“1”只能走到第四行的第二个数字“7”与第三个数字“4”。请寻找最佳一条路径，使得这条路径上节点的数字总和最大。

### 1.1　输入描述:

输入包含多组。每组数据的第一行包含一个正整数n（1≤n≤100），代表三角形的层数。紧接着有n行数字，第i（1≤i≤n）行包含i个自然数。

### 1.2　输出描述:

对应每组数据，输出最大的和。

### 1.3　输入例子:

5

7

3 8

8 1 0

2 7 4 4

4 5 2 6 5

### 1.4　输出例子:

30

## 2　解题思路

将示例的输入转换成二维数组。如图2-1所示。

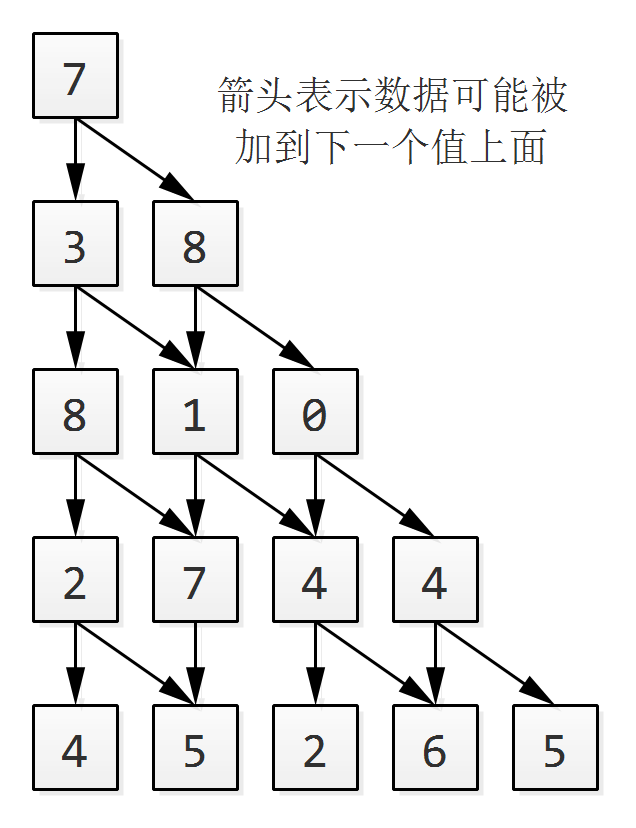


图2-1　数据流动图

根据意思和图2-1分分析，本题可以使用动态规划解决。设有待求最大值的矩阵matrix，它有n行，第i行最多i列，**每个元素的值都是正数**。第i（从1开始计算）行第j（从1开始计算）列对应到matrix的位置为matrix[i-1][j-1]。

第一列，它的第一行只有一个元素，它的最大值就是其自身，对于其它行，第一列的最大值等于它上一个元素的最大值加上当前值。对象线上的元素，除第一个元素外，其它元素的最大值等于它的左上角元素的最大值和当前值的和。最后对矩阵中的其它元素，其最大值等于左上角元素的最大值和上方元素的最大两都取较大的，再和当前值相加。对于matrix[i][j]有递推方程

最后，要求出matrix中的最大值只要遍历最后一列，它们中的最大值就是所求的matrix的最大值。