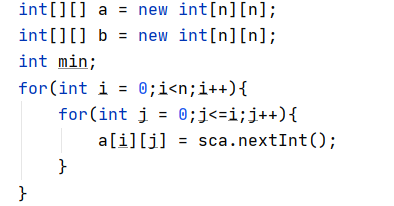
1. 需要有一个变量 n 来存储输入的行数

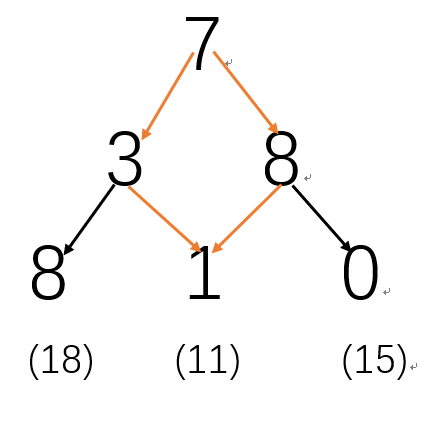


1. 需要一个二维数组 a 来存储输入的数字三角形



1. 需要另一个同样大小的二维数组 b，用来存储到每一层的每一个数的最短路径，

例如：

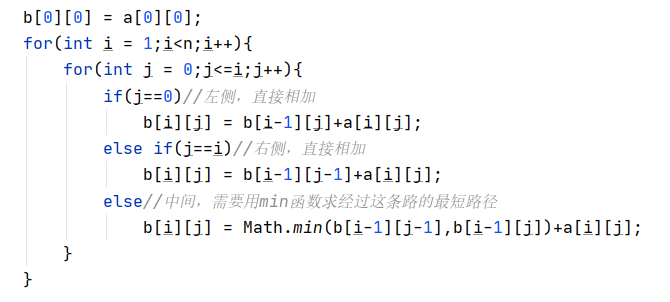


到三角形的第三层，有两条路会经过1，

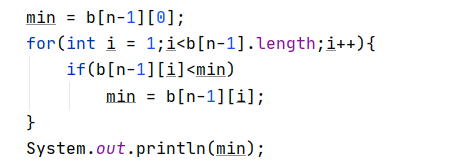
由于7+3=10<7+8=15，所以b数组的1的位置存储的是最短路径7—>3—>1等于11，

而在最两边的，就直接累加就ok了，7+3+8=18，7+8+0=15

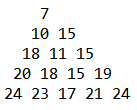
 这也是这个程序的核心部分，代码如下：



因为第一层跟a数组的第一层相同，所以i从1开始循环 然后遍历最后一层，求出最小值



b数组最后的值



复杂度分析

时间复杂度：O(n^2)，其中 n 是三角形的行数。

空间复杂度：O(n^2)。我们需要一个 n\*n 的二维数组存放所有的状态。