# **用excel解决八皇后问题**

作者：[恩皮西](http://gad.qq.com/user/index?id=34040)

链接：<http://gad.qq.com/article/detail/286628>

## **用excel解决八皇后问题**

-这可能是最蛋疼的excel问题之一

1. **1.八皇后介绍**

某天，好友毁童年发来一个题目，就是找到八皇后的所有解。我说这非常简单，我写段代码给你撸出来，网上也到处都是代码。他说，不能写代码用excel的本身功能。我觉得你这就为难我胖西了，但是不能就这么认怂，所以仔细思考了一下，经过好几个小时之后确实解决了这个问题。只用了excel的基础功能+函数，找到了八皇后的全部的解。

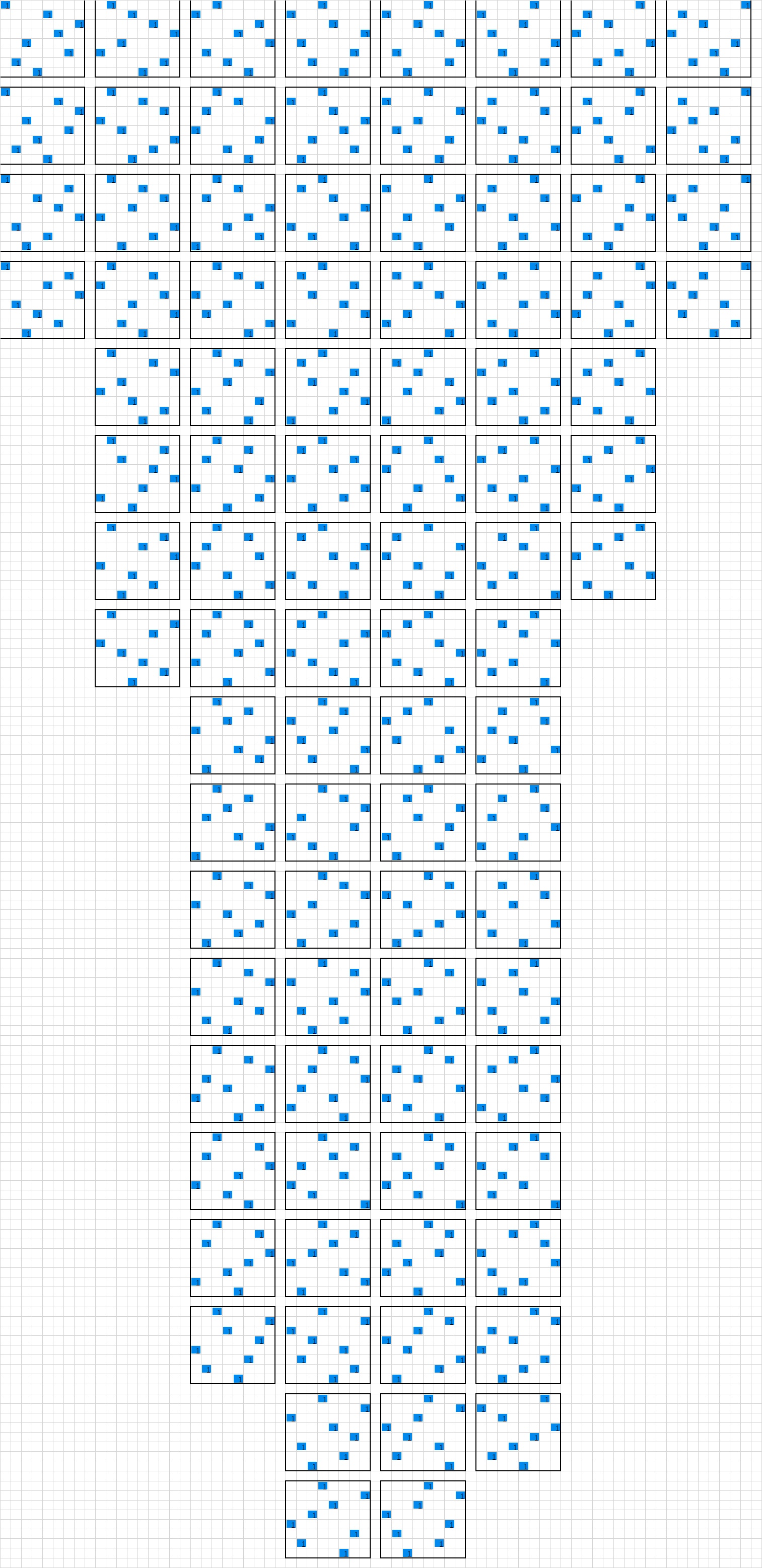
先简单介绍八皇后的题目，下方来自百度：

[八皇后](https://link.zhihu.com/?target=https://baike.baidu.com/item/%E5%85%AB%E7%9A%87%E5%90%8E" \t "http://gad.qq.com/article/edit/_blank)问题，是一个古老而著名的问题，是[回溯算法](https://link.zhihu.com/?target=https://baike.baidu.com/item/%E5%9B%9E%E6%BA%AF%E7%AE%97%E6%B3%95/9258495" \t "http://gad.qq.com/article/edit/_blank)的典型案例。该问题是国际西洋棋棋手马克斯·贝瑟尔于1848年提出：在8×8格的国际象棋上摆放八个皇后，使其不能互相攻击，即任意两个皇后都不能处于同一行、同一列或同一斜线上，问有多少种摆法。 [高斯](https://link.zhihu.com/?target=https://baike.baidu.com/item/%E9%AB%98%E6%96%AF/24098" \t "http://gad.qq.com/article/edit/_blank)认为有76种方案。1854年在柏林的象棋杂志上不同的作者发表了40种不同的解，后来有人用图论的方法解出92种结果。计算机发明后，有多种计算机语言可以解决此问题。

再继续看下面的内容时，读者可以尝试用自己的方法来解决这个问题。以免会有更好的方案被我的思路给干扰到。

实际上我最后也一次性的成功找到了全部的解。如下图：

每一个方块代表一个棋盘，标色是皇后。



1. **2.尝试思路**

首先我尝试的思路是找到一个解，找到一个解就能理解题目的意思，在这之前我从没有做过这个题，也没有用其他方法来解决这个问题。现在得从头开始。

第一个方案是，第一个皇后位置进行假设，后面的棋子位用if函数进行剔除，后来发现这明显是失败的，因为我无法让每个单元格的函数进行重复计算，这样会造成循环引用，第一个方案失败。

我紧接着的第二个想法就是，吧图形编码化，把图形问题转化为数学问题。

1. **3.图形编码化**

要把一个棋盘编码化意味着，我需要用一个数据来储存一个棋盘，很快我发现这个很容易做到，我们可以用一个八位数来储存一个棋盘，每一个数字代表一列，实际上代表一行也是没有问题的，好在八皇后的条件就是棋子不能同一列，这个省去了我们太多麻烦。只要哪个棋子再哪一行，这一列的数字就是多少即可。例如，11111111，这代表的是一个第一行被占满棋子的棋盘。

实际上棋子也不能同一行，这又给我们省去很多事情，棋盘就会变成123456578，因为同样的数字会代表同样的列，现在8个数字不能重复。接下来我们就要处理掉最后一个问题，那就是如何让他们不再统一斜排。不能在同以斜排意味着，这个数字的任意两位的差不能等于他们的间隔，这个问题如果采用代码解决就会非常简单，循环遍历即可，但是这里加上了自虐的条件，就是不能用代码来实现。我们要实现类似与循环的行为来进行这里的数据处理。

1. **4.用拖拉解决循环问题**

Excel里面有一个非常像循环的操作，就是拖拉，循环几次就拖拉几行，最后一行的结果，就是循环的结果，这种行为非常类似于for（i=1 to 行数）机制，但是这里又有另一个问题，这只是单层循环并且根据excle的基础功能，似乎没有实现多层循环的方案（是不是二维数组可以视为2层循环？那样就得用一个单元格储存所有信息，就得给处理的数据设计数据结构来储存。这样似乎更幸苦。如果有其他的方案请告知我。）

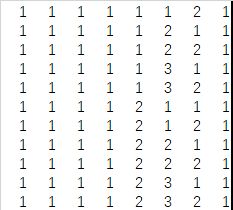
再确定了只有一层循环之后，要做的就是循环展开，就要精确的计算最内层的循环次数，并且确定每一次循环的外层的编号数据。最内层的数据实际上在本题中，就是12345678能够组成多少个不同的数？

1. **5.排列组合**

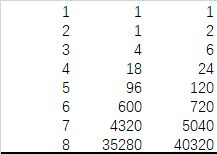
实际上有一定数学基础的人很容易解决这个答案，那就是8！=40320个。也就是说，做到最好的情况下，我们只要拖拉40320行，就可以便利所有符合条件的棋盘，excel一共支持104w行，在数据量超过10w行之后就基本会出现严重的卡顿情况，在这种复杂的计算的情况下，40320行的拖拉，还不是特别大的问题。

接下来的问题是，我没有要把这40320种可能性全部列出来。由于我们只有一层循环，我们的列出这些数字的时候一定需要按照某种严格的排序，才能确保不会遗漏。一个简单的方法就是，从小到大的方法去进行排列。最小的数字显然是12345678，其次就是12345687，按照这种方案，我们得写一些公式，就能吧他们全部列出来。这个函数是下一个数字 = f（上一个数字，1），然后你可以发现，这种函数极其难写，因为你得对这个数字进行内部排序，而且多次，一定会牵扯到循环问题。

实际上每一次的变化，都是2个数字的变动，都是一次局部排序，如果能把局部排序的需要调动的数字给列出来，就可以容易的列出这个数据组合。我们渴望获得一个这样的数组。



前面的没有变动的数字就让他不要变动，只要对后面的排序变动的数据进行操作。这里有一个问题，就是一定要知道我的数字调动到哪一层了。比如只变动最后2个，还是最后3个，还是最后4个。这样我需要知道，每一行，我变动到了哪个位置，我就需要知道每一层的行数，也是循环量。实际上这个内部的循环量每一个都是小的x皇后问题。用阶乘计算，然后逐行减去，就是本层的数据。使用mod和vlookup就可以定位泵行的变动位数，就能列出上体这种表，只需要再增加一个阶乘表的差就可以。实际上这个表也可以写在公式里，但是未来清晰，单列一个表，然后用vlookup引用即可。这个表shi这样的：



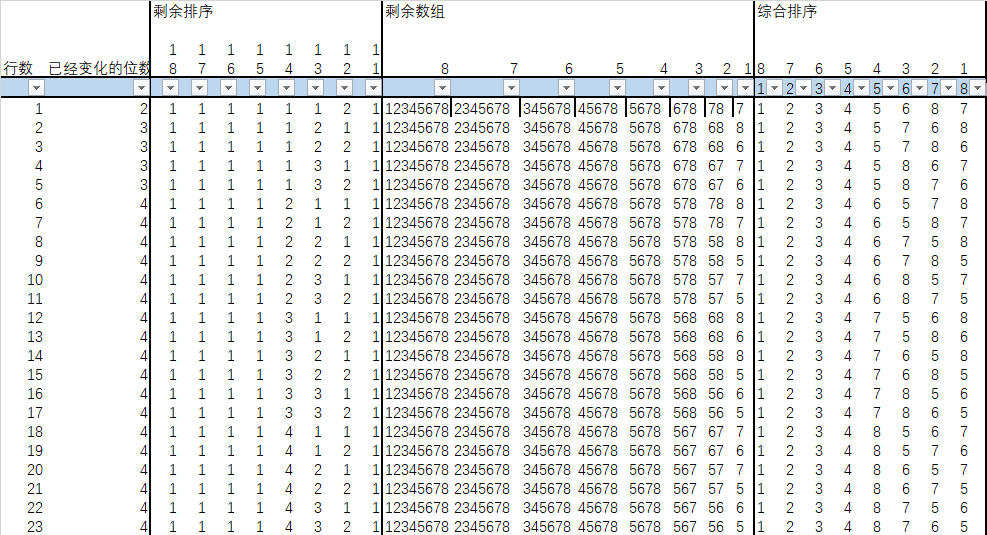
第一列是位数，第二列是阶乘差，第三列是阶乘。

你知道对自己的行数和位数定位取余就能确定自己的排序位置。然后就能得到yi张我上面说到的表。

1. **6.用字符串函数解决抽取问题**

现在我们要用上面的”字符变化表”来获得真正的数据对字符进行真正的排序。我如何对这个数据进行真正的排序呢。这里采用字符串来储存。一个完整的字符串初始状态是12345678，遇到了一个11111121，意味着最后2位需要颠倒才能拿到正确的数值。

这里你要理解我们上面做11111121的真正目的，就是1代表着这一位实际上是后续的数字种排序最小的数字。最后的21的2意味着这里应该是后面2为种排序第二小的数字，那就是78的8，这样就转化为了12345687.为了顺利的实现这个目的，加上了8行的辅助列。逐步的后拆解后续的数字，方便对后续进行排序。本质上，我们是从这个字符串种，抽出了它的每一位的数字。就拿12345678，遇到11111121来说，11111121的第一个1，需要抽取的是12345678的第一个数字，最小的，实际上哟个mid就可以完成，剩余2345678，第二个抽去2，依次，直到78，遇到了2，这次需要先抽取8，再抽取7，就实现了12345678转化为12345687.于是现在的表格变成这样



1. **7.展示所有可能方案**

你发现最后的综合排序已经是我们需要的解了，至此，只要拖拉40320行，我们就已经找出了全部可能的解，接下来我们只要对数据进行筛选，找出符合的数据。

1. **8.对可能的数据验证**

上面说到，要找到所有位数差=数字差的数，都不符合条件。如12345678，1在第一位，2在第二位，1-2=1的位数-2的位数，这是不符合的。这里用笨办法来处理，因为只有八位，我们把所有的位数差都列出来。那就是：

第一位和后面的位数差，2345678

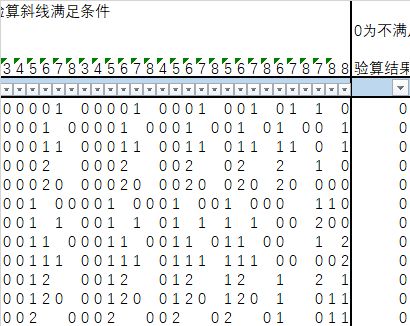
第二位和后面的位数差，345678

最后以为位数差，8

总体列出来就是，2345678345678456785678678788.

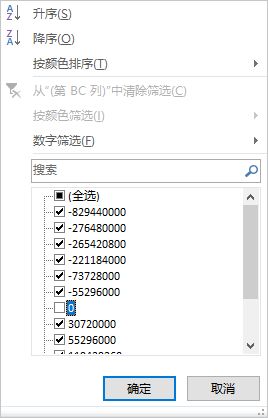
把他们每一个变成一行，然后真正的计算位数差，位数差=数字差，该位为0.

我们现在要剔除所有本行中包含0 的数字，很简单的方法，全部相乘，只要有一个0结果就是0，结果不是0那么意味着所有的都不是0.新起一列进行验算。结果是这样的。



1. **9.筛选出正确答案**

即将大公搞成，增加一个筛选，验算结果0的选项去掉。



选中这一列，excel惊喜的提示你

用excel解决八皇后问题

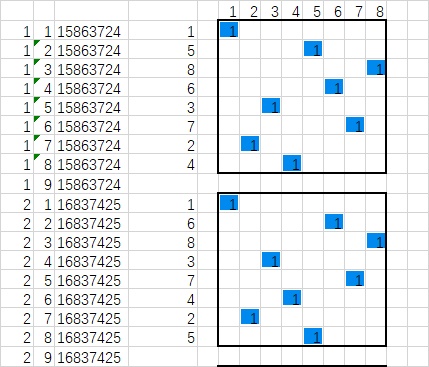
没错，我们吧所有的解找出来了，再加一列，吧前面一个一个挑出来的数字完整的数字吧。

下面就是全部的答案。

15863724,16837425,17468253,17582463,24683175,25713864,25741863,26174835,26831475,27368514,27581463,28613574,31758246,35281746,35286471,35714286,35841726,36258174,36271485,36275184,36418572,36428571,36814752,36815724,36824175,37285146,37286415,38471625,41582736,41586372,42586137,42736815,42736851,42751863,42857136,42861357,46152837,46827135,46831752,47185263,47382516,47526138,47531682,48136275,48157263,48531726,51468273,51842736,51863724,52468317,52473861,52617483,52814736,53168247,53172864,53847162,57138642,57142863,57248136,57263148,57263184,57413862,58413627,58417263,61528374,62713584,62714853,63175824,63184275,63185247,63571428,63581427,63724815,63728514,63741825,64158273,64285713,64713528,64718253,68241753,71386425,72418536,72631485,73168524,73825164,74258136,74286135,75316824,82417536,82531746,83162574,84136275

1. **10.答案图形化**

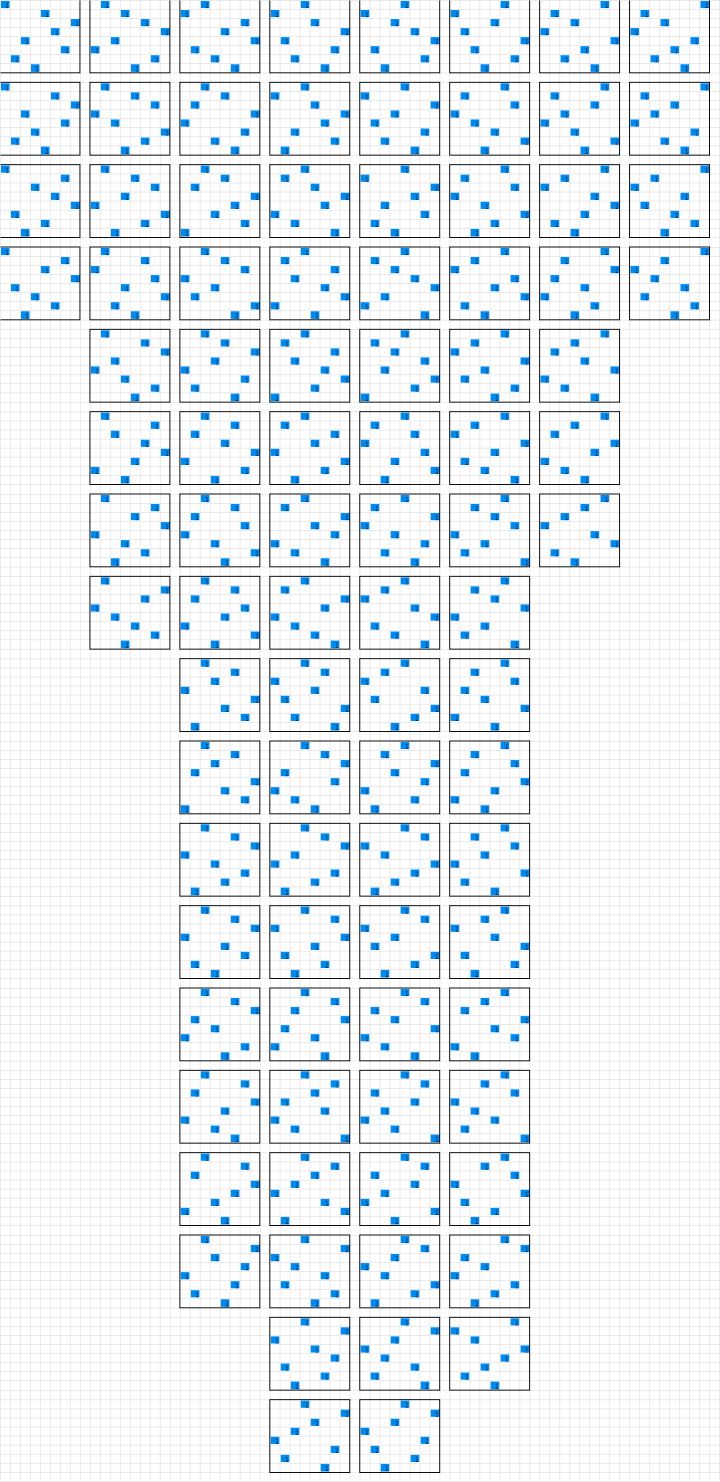
接下来把这些答案变成棋盘，为了方便观看，这就容易的多了



一个判断函数，数字位置判断行号，符合的为1，不符合为0.然后用条件格式，或者其他任何方法，把有数字的位置显示出来。

1. **11.整理发布**

最后一步，为了看的更清晰，吧所有的解根据列棋子的位置分列，就得到了最开始的图。



**12.给我点赞**

可以前往我的知乎，找我要原始表格

