

下面用过程 **Squares** 创建一些有趣的图形吧！如图 4-20 所示，我们继续创建一个名为 **RotatedSquares** 的过程。它多次调用过程 **Squares**，每次调用都旋转特定的角度。

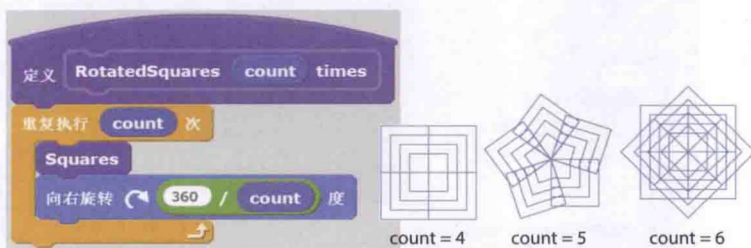


图 4-20：过程 **RotatedSquares** 及其可能的绘制结果

该过程两次使用参数 **count**：第一次用来确定重复执行的次数，第二次则在调用过程 **Squares** 之后计算转动的角度。例如，**count** 等于 5，那么一共重复 5 次，每次重复时向右旋转  $72^\circ$ （即  $360^\circ/5$ ）。尝试不同的 **count** 值，看看还能创建出什么图形。

Checkers.sb2

过程的嵌套是不是非常强大呢？我们再看一个案例。这次仍从图 4-16 的 **Square** 过程开始最终绘制一个棋盘。

创建绘制一排正方形的过程 **Row**，如图 4-21 所示。注意，正方形的个数使用参数指定。为了让程序简单，我们规定正方形的边长为 20。当然也可以将其作为过程 **Row** 的第二个参数。

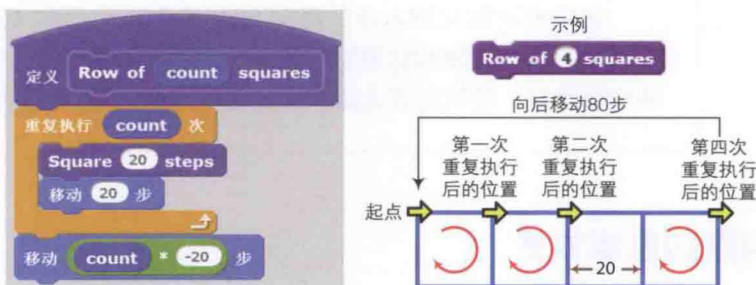


图 4-21：过程 **Row**

图 4-21 展示了当过程 **Row** 的参数为 4 的绘制结果，其中调用了四次 **Square 20 steps**。每绘制一个正方形，角色的位置会向前移动 20 步，从而为下一次绘制做好准备。在四个正方形全部绘制完后，最后一块积木命令角色回到最初的位置。

若要绘制如图 4-21 所示的下一排正方形，只需要将角色向下移