

## 循环的嵌套

我们早在第2章的“旋转的正方形”一节中就已见过循环的嵌套结构。一个循环（内层循环）负责绘制正方形，另一个循环（外层循环）控制旋转的次数。下面将把循环计数器和循环的嵌套结构结合起来，从而在两个甚至更多的循环中迭代。这种编程技术是非常重要的，它可以解决大量的编程问题。

某家餐厅提供四种披萨（P1、P2、P3 和 P4）以及三种沙拉（S1、S2 和 S3）。如果在这家餐厅吃饭，则共有 12 种组合，即 P1 搭配三种沙拉之一，或者 P2 搭配三种沙拉之一，以此类推。餐厅老板希望打印出一张菜单，其中罗列了各种组合的价格以及卡路里含量。下面我们使用嵌套循环生成各种组合的列表。（计算价格和卡路里含量留给你作为练习。）

仔细思考，你会发现该问题需要两个循环：一个循环（外层循环）迭代披萨类型，另一个循环（内层循环）迭代沙拉类型。当外层循环从 P1 开始时，内层循环分别迭代了 S1、S2 和 S3。当外层循环迭代 P2 时，内层循环再一次迭代了 S1、S2 和 S3。当外层循环迭代完所有的披萨类型后，整个过程就结束。图 7-15 的脚本实现了上述流程。

NestedLoops1.sb2

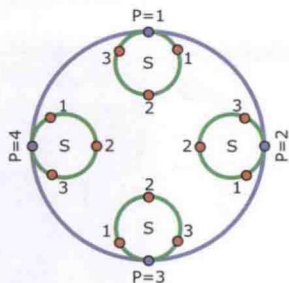


图 7-15：变量 P 控制外层循环，变量 S 控制内层循环

脚本使用了两个循环和两个循环计数器。外层循环的计数器为 P，内层循环的计数器为 S。在外层循环的第一轮迭代中（即 P=1），首先将 S 设定为 1，然后内层循环重复 3 次，每次执行说…用以显示当前的 P 和 S 的值，再将 S 的值增加 1。因此，在外层循环的第一轮迭代过程中，角色会依次说“P1,S1”“P1,S2”和“P1,S3”。

当内层的循环重复 3 次结束后，变量 P 会增加 1，这为下一次