

并修改它的值。假设有一个名为 side 的存储了数字 50 的盒子（即变量），我们可以使用它构建积木，例如移动（3*side）步。当 scratch 执行这块积木时，它首先在计算机内存中查找名为 side 的盒子，取出盒子内的数据（本例中即数值 50），用它替换移动（3*side）步中的 side。最后角色会移动 150（即 3×50）步。

打地鼠游戏需要记录玩家的分数，为此需要在计算机中分配内存（就像一个盒子）存储分数。我们给盒子一个标签 score，注意标签是不能重复的。有了标签就能随时找到它并修改其中的值。

游戏开始时，我们需要告诉 Scratch “把 score 设定为 0”。随后 Scratch 开始查找标签为 score 的盒子，然后将数字 0 放入其中。当玩家打到地鼠角色后，我们告诉 Scratch “把 score 的值增加 1”，Scratch 便会再次查找 score 盒子，取出盒子内的数字 0 并加 1，再把结果 1 重新放回盒子中。若玩家又打到了地鼠，Scratch 会再次执行上述操作，最后把数字 2 放入盒子中。

上述操作其实都是与 Scratch 的积木相对应的。注意 score 的值在不停地变化，这也就是变量中变的含义。

变量的一个重要用法是存储算术表达式的结果，从而方便脚本随后使用。这个过程类似于做心算。例如，要计算 2+4+5+7 的结果，你可能先计算 2+4 得到 6，并记住这个结果。然后可能用 5 加上之前的结果（你脑海中记忆的数字 6），并记住新的结果 11。最后用 7 加上之前记忆的结果 11，得到最后的结果 18。

下面演示使用变量存储表达式的结果。假设你想编写程序计算表达式：

$$\frac{(1/5) + (5/7)}{(7/8) - (2/3)}$$

你可能会将多块积木放在同一条积木中，但是这种做法的可读性非常差，不易理解：



另外一种计算方式是分别求出分子和分母的值，然后用说…积木显示两者相除的结果。我们可以创建两个变量，分别命名为 num（分子）和 den（分母），并设置它们的值，如图 5-4 所示。