

试一试 8-2

打开 *PigLatin.sb2* 并运行。程序首先询问用户输入单词，然后输出其 pig latin。尝试修改程序让其加密句子，例如，“Would you like some juice?”。(提示：对每一个单词调用过程 *PigLatin* 最后重新组装成句子。)你还可以尝试编写解密 pig latin 的过程，即把 pig latin 作为参数显示原先的单词。

单词修正

第二个案例是修正单词的小游戏。程序在单词的随机位置插入任意一个字母，创建错误的单词，而玩家需要输入错误单词的正确拼写。当然，错误单词的正确拼写可能不止一个，例如，正确单词是 *wall*，而游戏生成的错误单词是 *mwall*，那么 *mall* 和 *wall* 都是正确的。为了让游戏简单，我们将忽略这种可能性，仅认为原单词是正确的。

首先编写过程 *Insert*，它可以将某个字符插入到字符串的指定位置，因此含有三个参数：输入的单词 *strIn*、插入的字符（或字符串）*strAdd* 和插入的位置 *charPos*。该过程会生成新的字符串 *strOut*，如图 8-7 所示。

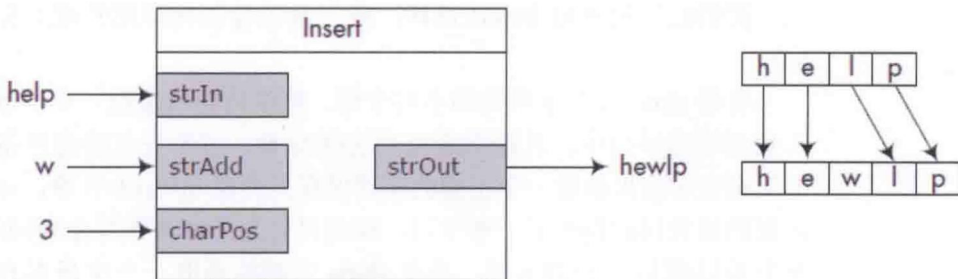


图 8-7：过程 *Insert* 的输入 / 输出

过程将 *strIn* 中的字符依次加入 *strOut*，直到达到位置 *charPos*。然后将 *strAdd* 加入到 *strOut* 之后，最后插入 *strIn* 剩余的字符。完整的过程如图 8-8 所示。