第14章 数据结构

14.1 数组

Lua语言中一般以1作为数组的起始索引。

14.2 矩阵及多维数组

Lua语言有两种方式表示矩阵，第一种方式是使用一个不规则数组，即数组的数组，也就是一个所有元素均是另一个表的表。

表示矩阵的第二种方式是将两个索引合并为一个。典型情况下，我们通过将第一个索引乘以一个合适的常量再加上第二个索引来实现这种效果。

应用程序经常会用到稀疏矩阵，这种矩阵中的大多数元素是0或nil。

14.3 链表

14.4 队列及双端队列

Lua语言中实现队列的一种简单方法是使用table标准库中的函数insert和remove。

不过这种 移动对于较大的结构来说开销很大。一种更高效的实现是使用两个索引，一个指向第一个元素，另一个指向最后一个元素。

编程：实现一个双端队列的数据结构

14.5 反向表

days = {"Sunday","Monday",...}的反向表为

revDays = {["Sunday"] = 1, ["Monday"] = 2,...}

14.6 集合与包

14.7 字符串缓冲区

Java提供了StringBuffer类解决这个问题。

而在Lua中，可以把一个表当作字符缓冲区，使用table.concat，将指定列表中的所有字符串连接起来并返回连接后的结果。

concat海鸥第2个可选参数，用于指定插在字符串间的分隔符。

14.8 图形

使用一个由两个字段组成的表来表示每个节点，即name（节点的名称）和adj（与此节点邻接的节点的集合）。

编程：实现图的数据结构

编程：从文件中加载图

编程：寻找两个节点之间的路径