

下面将学习链表中更常用的搜索和排序操作。

试一试 9-5

将三个过程 `FindAverage`、`FindMax` 和 `FindMin` 整合到一个过程 `ProcessList` 中，使其同时显示链表 `score` 的平均数、最大值和最小值。

搜索和排序

如果手机的通信录未按字母表排序，那么找出某位好友的信息将会费时费力。为了解决该问题，你可能会将通信录按照姓名排序，然后进行搜索，从而得到好友的电话号码等信息。本节将介绍编程中常见的两种操作：搜索和排序。

线性搜索

SearchList.sb2

Scratch 的包含积木可以检测出链表是否含有特定的变量。因此，若要获得被搜索的变量在链表中的位置，我们需要亲自实现该过程。

下面介绍一种被称为线性搜索（或顺序搜索）的算法。该算法思想简单、易于实现，可作用于任何链表。然而线性搜索需要进行大量的比较，若链表中的变量较多，花费时间则较长。

假设要搜索链表 `fruit`，若其中包含要搜索的项，我们希望找出该项的确切位置。图 9-19 的过程 `SearchList` 对链表 `fruit` 进行线性搜索，并返回搜索项的位置。

过程 `SearchList` 从链表的第一个变量开始，依次与我们想搜索的变量，即参数 `target` 进行比较。过程仅在找到 `target` 或到达链表末尾时结束：如果链表找到了希望搜索的值，变量 `pos` 就等于该变量的索引，否则过程设置 `pos` 为无效值（本例为 `-1`），表示链表中不存在 `target`。图 9-20 的脚本展示了该过程的调用和输出。