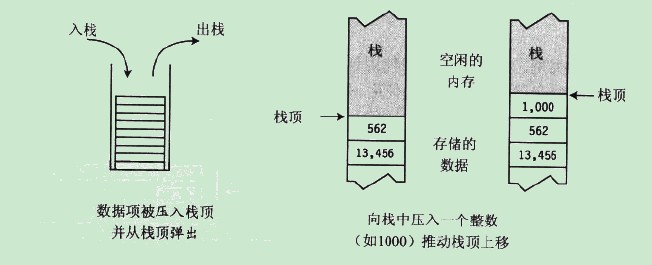
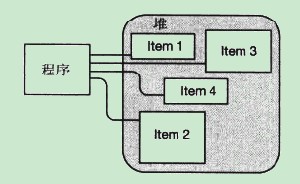
**C# 堆和栈**

    程序运行时，它的数据必须存储在内存中。一个数据项需要多大的内存、存储在什么地方、以及如何存储都依赖于该数据项的类型口。  
    运行中的程序使用两个内存区域来存储数据:栈和堆。  
**栈**   系统接管所有的栈操作。作为程序员,不需要显式地对它做任何事情。但了解栈的基本功能可以更好地了解程序在运行时正在做什么，并能更好地了解C#文档和图书。  
    栈是一个内存数组，是一个LIFO(last-in first-out,后进先出)的数据结构。栈存储几种类型的数据:  
    1）某些类型变量的值;  
    2）程序当前的执行环境:  
    3）传递给方法的参数。  
    栈的特征  
    栈有如下几个普遍特征〔见图1):  
    1）数据只能从栈的顶端插入和删除:  
    2）把数据放到栈顶称为入栈(push）;  
    3）从栈顶删除数据称为出找(pop）

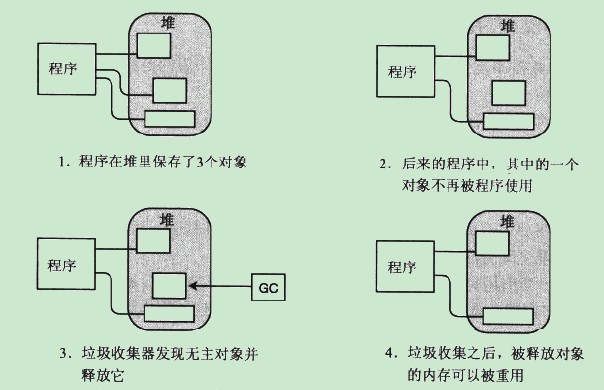
[](http://photo.blog.sina.com.cn/showpic.html%23blogid=5d2e69770101e8r5&url=http://album.sina.com.cn/pic/001HNxEXgy6EvmRUgf173)

**堆**

 堆是一块内存n:域，在堆里可以分配大块的内存用于存储某类型的数据。与栈不同，堆里的内存可以任意顺序存入和移除。图2展示了一个在堆里放了4项数据的程序。

[](http://photo.blog.sina.com.cn/showpic.html%23blogid=5d2e69770101e8r5&url=http://album.sina.com.cn/pic/001HNxEXgy6Evoz3QtYea)

  虽然程序可以在堆里保存数据，但并不能显式地删除它们。CLR的自动GC(Garbag Cal}ector，垃圾收集器)在判断出程序的代码将不会再访问某数据项时，自动清除无主的堆对象。  
  我们因此可以不再操心这项使用其他编程语言时非常容易出错的工作了。图3表明了垃圾收集过程。

[](http://photo.blog.sina.com.cn/showpic.html%23blogid=5d2e69770101e8r5&url=http://album.sina.com.cn/pic/001HNxEXgy6EvoNeW1za9)