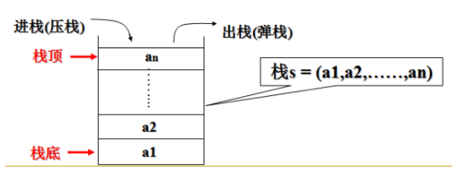
如果你学过数据结构，就一定会遇到“堆”,"栈","堆栈","队列"，而最关键的是这些到底是什么意思？最关键的是即使你去面试，这些都还会问到，所以如果你不懂对你是损失很大的。

**堆栈都是一种**[**数据项**](http://baike.sogou.com/v452927.htm)**按序排列的数据结构，只能在一端(称为栈顶(top))对数据项进行插入和删除。**

**要点：堆：顺序随意      栈：后进先出(Last-In/First-Out)**

****

**堆**

堆：什么是堆？又该怎么理解呢？

①堆通常是一个可以被看做一棵树的数组对象。堆总是满足下列性质：

   ·堆中某个节点的值总是不大于或不小于其父节点的值；

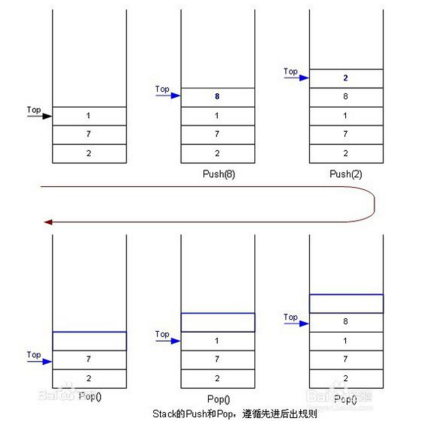
   ·堆总是一棵完全二叉树。

将根节点最大的堆叫做最大堆或大根堆，根节点最小的堆叫做最小堆或小根堆。常见的堆有二叉堆、斐波那契堆等。

②堆是在程序运行时，而不是在程序编译时，申请某个大小的内存空间。即动态分配内存，对其访问和对一般内存的访问没有区别。

③堆是应用程序在运行的时候请求操作系统分配给自己内存，一般是申请/给予的过程。

④堆是指程序运行时申请的动态内存，而栈只是指一种使用堆的方法(即先进后出)。



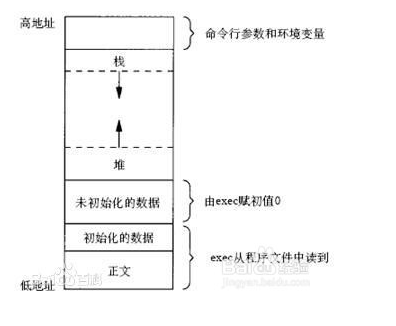
**栈**

栈：什么是栈？又该怎么理解呢？

①栈（stack）又名堆栈，它是一种运算受限的线性表。其限制是仅允许在表的一端进行插入和删除运算。这一端被称为栈顶，相对地，把另一端称为栈底。

②栈就是一个桶，后放进去的先拿出来，它下面本来有的东西要等它出来之后才能出来（先进后出）

③栈(Stack)是操作系统在建立某个进程时或者线程（在支持多线程的操作系统中是线程）为这个线程建立的存储区域，该区域具有LIFO的特性，在编译的时候可以指定需要的Stack的大小。

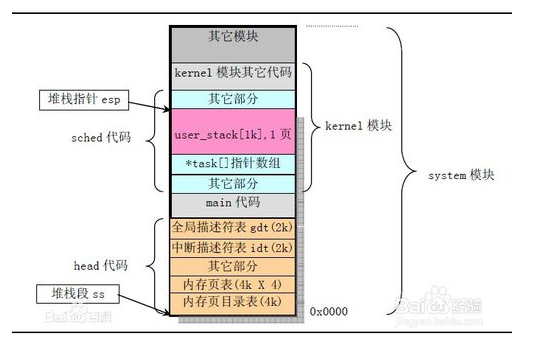


**堆栈**

堆栈：什么是堆栈？又该怎么理解呢？

注意：其实堆栈本身就是栈，只是换了个抽象的名字。

堆栈的特性： 最后一个放入堆栈中的物体总是被最先拿出来， 这个特性通常称为后进先出(LIFO)队列。 堆栈中定义了一些操作。 两个最重要的是PUSH和POP。 PUSH操作在堆栈的顶部加入一 个元素。POP操作相反， 在堆栈顶部移去一个元素， 并将堆栈的大小减一。



**总结：**

堆、栈区别总结：

1.堆栈空间分配

 ①栈（操作系统）：由操作系统自动分配释放 ，存放函数的参数值，局部变量的值等。其操作方式类似于数据结构中的栈。

 ②堆（操作系统）： 一般由程序员分配释放， 若程序员不释放，程序结束时可能由OS回收，分配方式倒是类似于链表。

2.堆栈缓存方式

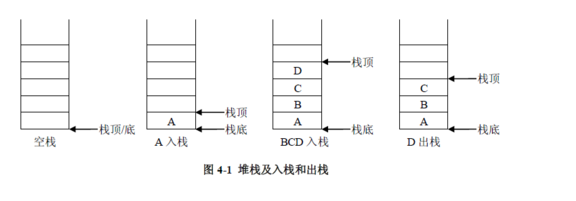
①栈使用的是一级缓存， 他们通常都是被调用时处于存储空间中，调用完毕立即释放。

②堆则是存放在二级缓存中，生命周期由虚拟机的垃圾回收算法来决定（并不是一旦成为孤儿对象就能被回收）。所以调用这些对象的速度要相对来得低一些。

3.堆栈数据结构区别

①堆（数据结构）：堆可以被看成是一棵树，如：堆排序。

②栈（数据结构）：一种先进后出的数据结构。



**队列：**

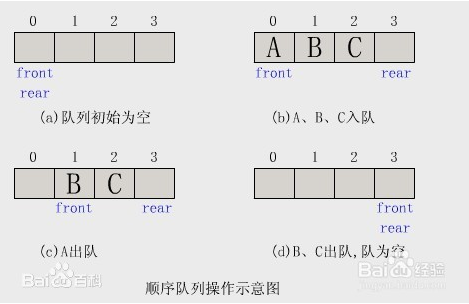
队列：什么是队列？又该怎么理解呢？

①队列是一种特殊的线性表，特殊之处在于它只允许在表的前端（front）进行删除操作，而在表的后端（rear）进行插入操作，和栈一样，队列是一种操作受限制的线性表。进行插入操作的端称为队尾，进行删除操作的端称为队头。

②队列中没有元素时，称为空队列。

③建立顺序队列结构必须为其静态分配或动态申请一片连续的存储空间，并设置两个指针进行管理。一个是队头指针front，它指向队头元素；另一个是队尾指针rear，它指向下一个入队元素的存储位置。

④队列采用的FIFO(first in first out)，新元素（等待进入队列的元素）总是被插入到链表的尾部，而读取的时候总是从链表的头部开始读取。每次读取一个元素，释放一个元素。所谓的动态创建，动态释放。因而也不存在溢出等问题。由于链表由结构体间接而成，遍历也方便。（先进先出）



**区别:**

堆、栈、队列之间的区别是？

①堆是在程序运行时，而不是在程序编译时，申请某个大小的内存空间。即动态分配内存，对其访问和对一般内存的访问没有区别。

②栈就是一个桶，后放进去的先拿出来，它下面本来有的东西要等它出来之后才能出来。（后进先出）

③队列只能在队头做删除操作,在队尾做插入操作.而栈只能在栈顶做插入和删除操作。（先进先出）

