# 栈的概念

就像子弹壳装弹，一粒一粒压进去，但是打出来的时候是从上面打出来的，最先压进去的最后弹出来，如果进去顺序是123，打出来顺序是321，这就是后进先出。

# 队列的概念

就是我们平时排队，按次序来，你排在第1个，那你就第一个轮到，就是先进先出，先到先来。

# 3、栈（Stack）和队列（Queue）是两种操作受限的线性表。

（线性表：线性表是一种线性结构，它是一个含有n≥0个结点的有限序列，同一个线性表中的数据元素数据类型相同并且满足“一对一”的逻辑关系。

“一对一”的逻辑关系指的是对于其中的结点，有且仅有一个开始结点没有前驱但有一个后继结点，有且仅有一个终端结点没有后继但有一个前驱结点，其它的结点都有且仅有一个前驱和一个后继结点。）

**这种受限表现在：**

**栈的插入和删除操作只允许在表的尾端进行（在栈中成为“栈顶”），满足“FIFO：First In Last Out”；**

**队列只允许在表尾插入数据元素，在表头删除数据元素,满足“First In First Out”。**

栈与队列的相同点：

1）都是线性结构。

2）插入操作都是限定在表尾进行。

3）都可以通过顺序结构和链式结构实现。

4）插入与删除的时间复杂度都是O（1），在空间复杂度上两者也一样。

5）多链栈和多链队列的管理模式可以相同。

栈与队列的不同点：

1）删除数据元素的位置不同，栈的删除操作在表尾进行，队列的删除操作在表头进行。

2）应用场景不同；

常见栈的应用场景包括括号问题的求解，表达式的转换和求值，函数调用和递归实现，深度优先搜索遍历等；

常见的队列的应用场景包括计算机系统中各种资源的管理，消息缓冲器的管理和广度优先搜索遍历等。

3）顺序栈能够实现多栈空间共享，而顺序队列不能。

# **4、堆和栈的区别**

（1）内存中的堆和栈

Java把内存划分成两种：一种是栈内存，一种是堆内存。

在函数中定义的一些基本类型的变量和对象的引用变量都在函数的栈内存中分配。 当在一段代码块定义一个变量时，Java就在栈中为这个变量分配内存空间，当超过变量的作用域后，Java会自动释放掉为该变量所分配的内存空间，该内存空间可以立即被另作他用。

堆内存用来存放由new创建的对象和数组。 在堆中分配的内存，由Java虚拟机的自动垃圾回收器来管理。

（2）数据结构中的堆Heap和栈Stack  
 数据结构方面的堆和栈，这些都是不同的概念。这里的堆实际上指的就是（满足堆性质的）优先队列的一种数据结构，第1个元素有最高的优先权；栈实际上就是满足先进后出的性质的数据结构。

栈就像装数据的桶或箱子。我们先从大家比较熟悉的栈说起吧，它是一种具有后进先出性质的数据结构，也就是说后存放的先取，先存放的后取。

堆像一棵倒过来的树。而堆就不同了，堆是一种经过排序的树形数据结构，每个结点都有一个值。通常我们所说的堆的数据结构，是指二叉堆。

堆的特点是根结点的值最小（或最大），且根结点的两个子树也是一个堆。

二叉堆是一种特殊的堆，二叉堆是完全二元树（二叉树）或者是近似完全二元树（二叉树）。二叉堆有两种：最大堆和最小堆。最大堆：父结点的键值总是大于或等于任何一个子节点的键值；最小堆：父结点的键值总是小于或等于任何一个子节点的键值。

|  |
| --- |
| 堆满足下列性质：  堆中某个结点的值总是不大于或不小于其父结点的值；  堆总是一棵完全二叉树，每个节点最多可以有两个节点；  除了根节点没有兄弟节点，最后一个左子节点可以没有兄弟节点，其他节点必须有兄弟节点。  将根结点最大的堆叫做最大堆或大根堆，根结点最小的堆叫做最小堆或小根堆。  堆就是可以迅速找到一堆数中的最大值或者最小值的数据结构，所以堆可以用来实现优先队列。  使用数组实现：  我们可以通过自己定义数组以及调整方法来实现堆。  对于完全二叉树，用数组存储时，对应的子节点的的索引：  如果第一个元素在数组中索引为0  索引为i的左子节点索引为2i+1  索引为i的右子节点索引为2i+2  索引为i的父节点索引为(i-1)/2  数组：arr = [12, 8, 4, 5, 1, -1, 3, 0]  对应的二叉树  12(0)  / \  8(1) 4(2)  /\ /\  5(3) 1(4) -1 3  /  0 |