快速排序与冒泡排序的不同

快速排序：**在每一轮挑选一个基准元素，并让其他比它大的元素移动到数列一边，比它小的元素移动到数列的另一边，**

冒泡排序在每一轮只把一个元素冒泡到数列的一端

思路：递归+分治法

**两种实现方法**

1. **挖坑法：**

实现过程与思路:

1. 选定基准元素 pivot 该位置为index---为坑的位置
2. 设置两个指针i,j分别指向最左边和最右边
3. 第一轮从右边开始如果arr[j]大于等于基准元素，j前移即j--
4. 如果arr[j]小于基准元素，arr[i]=arr[j]

i指针后移即i++, index指向j,下一轮从左侧开始

1. arr[j]小于等于基准元素，i后移即i++
2. 如果arr[i]大于基准元素， arr[j]= a[i]

j指针前移即j--,index指向i,下一轮从右侧开始

7.i和j重复的时候arr[index]=基准元素

i,j重复时跳出大循环

递归思路

Quicksort（）

*//递归结束条件***if**($first>=$end){  
 **return** ;  
}**else**{  
 *//获取基准元素* $pivot=$this->parttition($arr,$first,$end);  
 $this->quicksort($arr,$first,$pivot-1);  
 $this->quicksort($arr,$pivot+1,$end);

}

}

*//寻找基准元素***public function** parttition($arr,$first,$end){  
 $pivot=$arr[$first];  
 $i=$first;  
 $j=$end;  
 *//坑的位置* $index=$first;  
 *//外层循环在左右指针重合或交错时结束* **while**($i<$j){  
 *//右侧扫描* **while**($i<$j){  
 *//右侧指针值大于等于基准元素，右侧指针前移* **if**($arr[$j]>=$pivot){  
 $j--;  
 }  
 **else**{*//右侧指针值小于基准元素，* $arr[$i]=$arr[$j];  
 $index=$j;  
 $i++;*//左侧指针后移* **break**;  
 }  
 }  
 *//左侧扫描* **while**($i<$j){  
 *//左侧指针值小于等于基准元素，左侧指针后移* **if**($arr[$i]<=$pivot){  
 $i++;  
 }**else**{*//左侧指针值小于基准元素，* $arr[$j]=$arr[$i];  
 $index=$i;  
 $j--;  
 **break**;  
 }  
 }  
 }  
 $arr[$index]=$pivot;  
 **return** $index;  
}

1. 指针交换法
2. 选定基准元素 pivot 该位置为index-
3. 设置两个指针i,j分别指向最左边和最右边
4. 每一轮从右侧开始，如果arr[j]>=基准元素j前移即j--，如果arr[j]<基准元素，就跳到左指针，如果arr[i]<=基准元素i后移即i++，如果arr[i]<基准元素则交换a[i],a[j]的值------简单地说即使每一轮从又开始右边找到第一个小于基准元素的值，左边找到第一个大于基准元素的值然后交换
5. 当i==j的时候将基准元素和arr[start]（arr的第一个元素）交换
6. *//指针交换法***public function** quicksort2(&$arr,$first,$end){  
    *//递归结束条件* **if**($first>=$end){  
    **return** ;  
    }**else**{  
    $pivot=$this->parttition2($arr,$first,$end);  
    $this->quicksort2($arr,$first,$pivot-1);  
    $this->quicksort2($arr,$pivot+1,$end);  
    }  
   }  
   **public function** parttition2($arr,$first,$end){  
    $pivot=$arr[$first];  
    $i=$first;  
    $j=$end;  
    *//外层循环在左右指针重复的时候结束* **while**($i!=$j){  
    *//右侧扫描* **while**($i<$j&&$arr[$j]>=$pivot){  
    $j--;  
    }  
    *//左侧扫描* **while**($i<$j&&$arr[$i]<=$pivot){  
    $i++;  
    }  
    *//交换left和right指向的元素* **if**($i<$j){  
    $temp=$arr[$i];  
    $arr[$i]=$arr[$j];  
    $arr[$j]=$temp;  
    }  
    }  
    *//左右指针重合 替换基准元素* $temp=$arr[$i];  
    $arr[$i]=$arr[$first];  
    $arr[$first]=$temp;  
    **return** $i;  
   }

参考 ：<http://www.sohu.com/a/246785807_684445>

<http://www.xinyueseo.com/algorithm/209.html>