选择排序： a、每次把比较的第一个个数当做是最小数，如果找到后面有比自己小的数，就记录位置，比较完之后交换这两个数的位置。

b、选择排序的核心思想是每次比较时先假定参与比较的元素中最左边的为最小值，然后拿这个 不一定是最小值（假定）的最小值与后面的各个元素比较，找到最小值，并把最小值放到参与比较的元素的最左边。选择排序以左边的最小值作为排序比较的出发点

插入排序： a、(1)第一部分包含了这个数组的所有元素，但将最后一个元素除外，而第二部分就只包含这一个元素

(2)在第一部分排序后，再把这个最后元素插入到此刻已是有序的第一部分里的位置

b、插入排序的核心思想是局部有序，非局部有序的元素与局部有序的元素比较，并排序，直到全部有序，这个过程局部有序序列不断变大。

选择排序：每次从i的位置开始比到array.length-1，确定一个最小的数放到左边

1、确定循环次数：array.length-1次

for（int i=0;i<array.length-1;i++）

2、用index记录每次循环中最小数的下标，记录当前下标i

int index = i;

3、从当前位置(i)开始比到结尾，记录下最小的数的下标

for(int j=i+1;j<array.length;j++){

if(array[index]>array[j]){

index = j;

} }

4、更换array[i]与array[index]的值。每次循环确定下标i的值。

插入排序：假设数组部分有序，比如前3位有序。然后将无序的部分循环插入到已有序的序列中

1、确定循环次数，一个无序数组使用插入排序，每一位都要与前面有序的部分进行比较，

即第二位开始，与第一位比较（把第一位看成有序部分）

for(int i=0;i<array.length;i++)

2、记录当前数,即要插入的值temp 和 插入值的下标index；

int temp = array[i] int index = i；

3、循环遍历，当要插入的值小于前面的数array[j]时，需要将array[j]移动到array[j+1]的位置上。同时插入元素的位置会在前面。用index--表示

插入元素a[i]会移动到数组中的那个位置。

array[j + 1] = array[j];

index--;

4、遍历结束后将值a[i]插入空出来的位置index上

array[index] = temp;