

# 插入排序 (Insertion sort)

1. 假设第1个数字是最小的（升序），从第2个数字开始，依次将数字插入到他们应该放置的位置，类似于整理手上的扑克牌。暂未参与排序的数字，可以视为放在桌上还没抓的扑克牌。
2. 若位置是"3 7 8 1"，可以先把1存在临时变量中，并依次把8 7 3向右移位，最后再把临时变量的1存在最左边，没有必要每次都把1和左边的数交换顺序。
3. 随着排序的进行，需要对比和换位的操作均可能明显减少，所以，插入排序的效率明显高于冒泡排序和选择排序。
4. 插入排序（升序和降序）：

```
import java.util.Arrays;

public class InsertionSort {
    public static void main(String[] args) {
        int[] array = {8, 1, 4, 9, 0, 3, 5, 2, 7, 6};
        int temp;

        for (int i = 1; i < array.length; i++) { //升序，从第2个元素开始排
            for (int j = i; j > 0; j--) { //每个元素从右往左比较
                if (array[j] < array[j - 1]) {
                    temp = array[j];
                    array[j] = array[j - 1];
                    array[j - 1] = temp;
                } else {
                    break;
                }
            }
        }

        for (int i = 1; i < array.length; i++) { //降序，从第2个元素开始排
            for (int j = i; j > 0; j--) { //每个元素从右往左比较
                if (array[j] > array[j - 1]) {
                    temp = array[j];
                    array[j] = array[j - 1];
                    array[j - 1] = temp;
                } else {
                    break;
                }
            }
        }

        System.out.println(Arrays.toString(array));
    }
}
```