1.把数组分为[已排序]和[未排序]两部分,第一个数为[已排序]，其余为[未排序]

2.从[未排序]抽出第一个数，和[已排序]部分比较，插入到合适的位置

function insertionSort(myArray) {

var len = myArray.length, // 数组的长度

value, // 当前比较的值

i, // 未排序部分的当前位置

j; // 已排序部分的当前位置

for (i=0; i < len; i++) {

// 储存当前位置的值

value = myArray[i];

/\*

\* 当已排序部分的当前元素大于value，

\* 就将当前元素向后移一位，再将前一位与value比较

\*/

for (j=i-1; j > -1 && myArray[j] > value; j--) {

myArray[j+1] = myArray[j];

}

myArray[j+1] = value;

}

return myArray;

}

方法改进，修改交换值和循环过程

function insertionSort(myArray) {

var len = myArray.length, // 数组的长度

value, // 当前比较的值

i, // 未排序部分的当前位置

j; // 已排序部分的当前位置

loop:for (i=0; i < len; i++) {

// 储存当前位置的值

value = myArray[i];

/\*

\* 当已排序部分的当前元素大于value，

\* 就将当前元素向后移一位，再将前一位与value比较

\*/

for(j=i-1 ;j>-1 ;j--){

if(myArray[j+1] < myArray[j]){

myArray[j+1] = myArray[j]

myArray[j] = value;

}else{

continue loop;

}

}

}

return myArray;

}

insertionSort([5,3,7,2,1,8,4,6,9])