1.不断将数组对半分，直到每个数组只有一个  
2.将分出来的部分重新合并  
3.合并的时候按顺序排列

// 被拆分的数组重新合并

function merge(left, right) {

var result = [],

left\_index = 0,

right\_index = 0;

// 将两个数组合并

// 合并的时候按从小到大的顺序

while (left\_index<left.length&&right\_index<right.length) {

if (left[left\_index] < right[right\_index]) {

result.push(left[left\_index++]);

} else {

result.push(right[right\_index++]);

}

}

// 和其他数组拼接，这里用concat是因为left和right循环时，数组里的数不一定全部都取出来了，需要把数组中余下的数合并，用slice取余下的数组。

Returnresult.concat(left.slice(left\_index)).concat(right.slice(right\_index));

}

functionmergeSort(myArray) {

// 只有一个数的时候退出递归

if (myArray.length< 2) {

returnmyArray;

}

var middle = Math.floor(myArray.length / 2),

left = myArray.slice(0, middle),

right = myArray.slice(middle);

// 递归

// 不断拆分只到一个数组只有一个数

return merge(mergeSort(left), mergeSort(right));

}