# 53、最大子序和（Maximum Subarray）

## 题目：

给定一个整数数组 nums ，找到一个具有最大和的连续子数组（子数组最少包含一个元素），返回其最大和。

示例:

输入: [-2,1,-3,4,-1,2,1,-5,4]   
输出: 6   
解释: 连续子数组 [4,-1,2,1] 的和最大，为 6。

进阶:

如果你已经实现复杂度为 O(*n*) 的解法，尝试使用更为精妙的分治法求解。

## 解答：

法一：暴力搜索

|  |
| --- |
| public static int MaxSubArray(int[] nums)  {  int sum = 0;  int max = nums[0];  for (int i = 0; i < nums.Length; i++)  {  sum = 0;  for (int j = i; j < nums.Length; j++)  {  sum += nums[j];  if (sum > max)  {  max = sum;  }  }  }  return max;  } |

法二：

|  |
| --- |
| public static int MaxSubArray1(int[] nums)  {  // -1 2 3 -2 1  //maxTocur -1 1/2√ 2+3√/3 3√/-2 4√/1  //max -1 2 5 5 5  //maxTocur：Max(maxTocur+num[i],num[i])  //max：Max(max,maxTocur)  int maxToCurr = nums[0];  int sum = nums[0];  for (int i = 1; i < nums.Length; i++)  {  maxToCurr = Math.Max(maxToCurr + nums[i], nums[i]);  sum = Math.Max(sum, maxToCurr);  }  return sum;  } |