**难度: 简单**

输入一个整型数组，数组中的一个或连续多个整数组成一个子数组。求所有子数组的和的最大值。

要求时间复杂度为O(n)。

示例1:

输入: nums = [-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。

提示：

1 <= arr.length <= 10^5

-100 <= arr[i] <= 100

来源：力扣（LeetCode）

链接：https://leetcode-cn.com/problems/lian-xu-zi-shu-zu-de-zui-da-he-lcof

著作权归领扣网络所有。商业转载请联系官方授权，非商业转载请注明出处。

**思路:**

1). 当最大值小于0时,就替换最大值否则就最大值+下一个节点的值

2). 如果比较结果集与最大值进行比较,把大的重新赋给最大值

3). 当循环结束后,直接返回结果

|  |
| --- |
| public int maxSubArray(int[] nums) {  //假设第一个是最大值  int max = nums[0];  //假设第一个是结果  int res = nums[0];  //循环  for(int i = 1; i < nums.length; i++){  //最大值,小于0  if(max < 0) {  max = nums[i];  } else {  //最大值相加  max += nums[i];  }  //比较加过的最大值是否大于结果集  res = Math.max(res, max);  }  return res;  } |