# 连续子数组的最大和

## 题目

[牛客网](https://www.nowcoder.com/practice/459bd355da1549fa8a49e350bf3df484?tpId=13&tqId=11183&rp=1&ru=%2Fta%2Fcoding-interviews&qru=%2Fta%2Fcoding-interviews%2Fquestion-ranking&tPage=2)

例如:{6,-3,-2,7,-15,1,2,2},连续子向量的最大和为8(从第0个开始,到第3个为止)。给一个数组，返回它的最大连续子序列的和，你会不会被他忽悠住？(子向量的长度至少是1)

## 解题思路

通过动态规划计算最大和，

$$f(i)$$

定义为以第

$$i$$

个数字结尾的子数组的最大和，那么

$$max(f(i))$$

就有以下公式：

$$max(f(i))=\left\{\begin{matrix}num[i]&i=0orf(i)<0\\num[i]+f(i)&i\ne 0andf(i)>0\end{matrix}\right.$$

public int FindGreatestSumOfSubArray(int[] array) {
 if (array == null || array.length == 0) {
 return 0;
 }

 int max = array[0];
 int sum = 0;
 for (int a : array) {
 if (sum + a > a) {
 sum += a;
 } else {
 sum = a;
 }

 if (sum > max) {
 max = sum;
 }
 }

 return max;
}