/\*

Find the contiguous subarray within an array (containing at least one number) which has the largest sum.

For example, given the array [-2,1,-3,4,-1,2,1,-5,4],

the contiguous subarray [4,-1,2,1] has the largest sum = 6.

方法二：动态规划，找转移状态，

如果前面的sum<0,则sum=nums[i]

否则，sum+=nums[i]

每一步找一次最大

\*/

class Solution {

public:

int maxSubArray(vector<int>& nums)

{

//way-2 DP

int sum=0;

int maxa=INT\_MIN;

for(int i=0;i<nums.size();i++)

{

if(sum>=0)

sum+=nums[i];

else

sum=nums[i];

maxa=max(maxa,sum);

}

return maxa;

}

};