# 题目

给定一个整型数组，在数组中找出由三个数组成的最大乘积，并输出这个乘积。

示例 1:

输入: [1,2,3]

输出: 6

示例 2:

输入: [1,2,3,4]

输出: 24

注意:

给定的整型数组长度范围是[3,104]，数组中所有的元素范围是[-1000, 1000]。

输入的数组中任意三个数的乘积不会超出32位有符号整数的范围。

# 分析

class Solution {

public:

int maximumProduct(vector<int>& nums) {

std::sort(nums.begin(),nums.end());

int size = nums.size();

int a = INT\_MIN,b=INT\_MIN;

if(nums[0]<0 && nums[1]<0)//如果是负数,去最大两个负数和最大整数

a = nums[0] \* nums[1] \* nums[size-1];

b = nums[size-1] \* nums[size-2] \* nums[size-3];

return max(a,b);

}

};