

## UVA11059 最大乘积

## 题目描述

给一串整数S，你需要找到其连续子序列中乘积最大的那个，输出其乘积。如果找到的答案不是正数，那么输出0，表示无解。

另外注意：每组输入后面都有一行空格，每组输出后面也应该有一个空行。输入以 EOF 结束。

## 输入样例

```
3
2 4 -3
5
2 5 -1 2 -1
```

## 输出样例

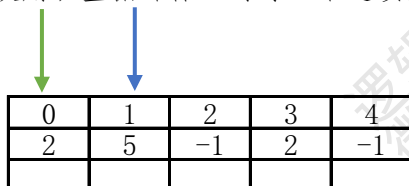
Case #1: The maximum product is 8.

Case #2: The maximum product is 20.

题目地址: <https://www.luogu.com.cn/problem/UVA11059>

## 解析

使用双重循环保证对每一个连续的子集进行计算。



0	1	2	3	4
2	5	-1	2	-1

## 编码

```
#include<bits/stdc++.h>

using namespace std;

int main() {
    //使用长整型记录最大值
    long long n, m = 0, maxValue;
    //记录输入的数字
    long long a[20] = {0};
    //输入以EOF结束
    while (cin >> n) {
        //清空数据
        memset(a, 0, sizeof(a));
        maxValue = 0;
        //记录组号
        m++;
        for (int i = 0; i < n; i++) {
```

```

        cin >> a[i];
    }

    //不断的枚举其被乘数
    for (int i = 0; i < n; i++) {
        //以i作为起始的乘积结果
        long long temp = a[i]; // -1;
        //用数组中的每一个作比较，防止有遗漏
        //例如特殊数据： -1 2
        if (temp > maxValue) {
            maxValue = temp;
        }
        //不断的枚举乘数
        for (int j = i + 1; j < n; j++) {
            temp *= a[j];
            //计算结果并比较
            if (temp > maxValue) {
                maxValue = temp;
            }
        }
    }
    cout << "Case #" << m << ": The maximum product is" << maxVal
}
return 0;
}

```

逻辑航线培优教育，信息学奥赛培训专家。

扫码添加作者获取更多内容。





```
lue << "." << endl << endl;
```