**Three Great Candidates**

[array](http://www.practice.geeksforgeeks.org/tag-page.php?tag=array&isCmp=0)

The hiring team of Google aims to find 3 candidates who are great collectively. Each candidate has his or her ability expressed as an integer. 3 candidate are great collectively if product of their abilities is maximum. Find the maximum collective ability from the given pool of candidates.

**Input:**  
The first line of input contains an integer T denoting the number of test cases. Then T test cases follow. First line of each test case conatins an interger N  denoting the number of candidates.  
The second line of each test case contains N space separated elements denoting the ablities of candidates.

**Output:**  
Corresponding to each test case, print the desired output(maximum collective ability of three candidates) in a newline.

**Constraints:**  
1 ≤ T ≤ 100  
3 ≤ N ≤ 1000  
-1000 ≤ ability ≤ 1000

**Example:**  
**Input**  
1  
6  
0 -1 3 100 70 50

**Output:**  
350000

**Explanation**  
70\*50\*100 = 350000 which is the maximum possible.

\*\*For More Examples Use Expected Output\*\*

<http://www.practice.geeksforgeeks.org/problem-page.php?pid=765>

#include <iostream>

#include <stdio.h>

#include <vector>

using namespace std;

int main(){

    int t;

    scanf("%d", &t);

    while(t-- ) {

       int n;

       scanf("%d", &n);

       std::vector<int> v;

       for(int i =0; i<n; i++) {

          int elem;

          scanf("%d", &elem);

          v.push\_back(elem);

        }

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

        int len = v.size();

        //printf("%d\n", v[len-1] \* v[len-2] \* v[len-3]);

        //cout << v[len-1] \* v[len-2] \* v[len-3] << endl;

        int max =0;

         //el maximo pueden ser 2 negativos y uno positivo ej.

         //-489 -478 496 is the triplet.

        max =  std::max( v[0] \* v[1] \* v[len-1] , v[len-1] \* v[len-2] \* v[len-3]);

        printf("%d**\n**",max);

    }

 return 0;

}