PL-2020-C-Small Triangles, Large Triangles

locked

You are given n triangles, specifically, their sides ai, bi and ci. Print them in the same style but sorted by their areas from the smallest one to the largest one. It is guaranteed that all the areas are different.

The best way to calculate a volume of the triangle with sides a, b and c is Heron's formula:

S = p * (p – a) * (p - b) * (p – c) where p = (a + b + c) / 2.

Input Format

First line of each test file contains a single integer n. n lines follow with ai, bi and ci on each separated by single spaces.

Constraints

1 ≤ n ≤ 100 1 ≤ ai, bi, ci ≤ 70 ai + bi > ci, ai + ci > bi and bi + ci > ai

Output Format

Print exactly n lines. On each line print 3 integers separated by single spaces, which are ai, bi and ci of the corresponding triangle.

Sample Input 0

3 7 24 25 5 12 13 3 4 5

Sample Output 0

3 4 5 5 12 13 7 24 25

Explanation 0

The square of the first triangle is 84. The square of the second triangle is 30. The square of the third triangle is 6. So the sorted order is the reverse one.

Input Format

-

Constraints

-

Output Format

-

Sample Input 0

```
3
7 24 25
5 12 13
3 4 5
```

Sample Output 0

```
3 4 5
5 12 13
7 24 25
```

Submissions: 1008
Max Score: 100
Difficulty: Medium

Rate This Challenge:

More

C

```c
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>

struct triangle_volume{
    int a;
    int b;
    int c;
    int result1;
    int result2;
};
struct triangle_volume t[100];
int main() {


    int n,i,j,p,temp;
    double x,s;
    scanf("%d",&n);
    for (i=0;i<n;i++)
    {
        scanf("%d %d %d",&t[i].a,&t[i].b,&t[i].c);
    }
    for (i=0;i<n;i++)
    {
        p=(t[i].a+t[i].b+t[i].c)/2;
        x=(p*(p-t[i].a)*(p-t[i].b)*(p-t[i].c));
        s=sqrt(x);
        t[i].result1=s;

        t[i].result2=t[i].result1;
    }
    for (i=0;i<n;i++)
    {
        for (j=i+1;j<n;j++)
        {
            if(t[i].result1>t[j].result1)
            {
                temp=t[i].result1;
                t[i].result1=t[j+1].result1;
                t[j+1].result1=temp;
            }
        }
```

```
    }
   for (i=n-1;i>=0;i--)
    {
        for (j=n-1;j>=0;j--)
        {
            if(t[j].result2==t[j].result1)
            {
                printf("%d %d %d\n",t[i].a,t[i].b,t[i].c);
            }
        }
    }

    return 0;
}

```

Line: 1 Col: 1

Upload Code as File Test against custom input Run Code Submit Code