Week 14

Roll no: 240801177

Name: Logaranjan R J

Status	Finished			
Started	Sunday, 12 January 2025, 7:08 PM			
Completed	Sunday, 12 January 2025, 7:25 PM			
Duration	16 mins 39 secs			

You are transporting some boxes through a tunnel, where each box is a parallelepiped, and is characterized by its length, width and height.

The height of the tunnel **41** feet and the width can be assumed to be infinite. A box can be carried through the tunnel only if its height is strictly less than the tunnel's height. Find the volume of each box that can be successfully transported to the other end of the tunnel. Note: Boxes cannot be rotated.

Input Format

The first line contains a single integer *n*, denoting the number of boxes.

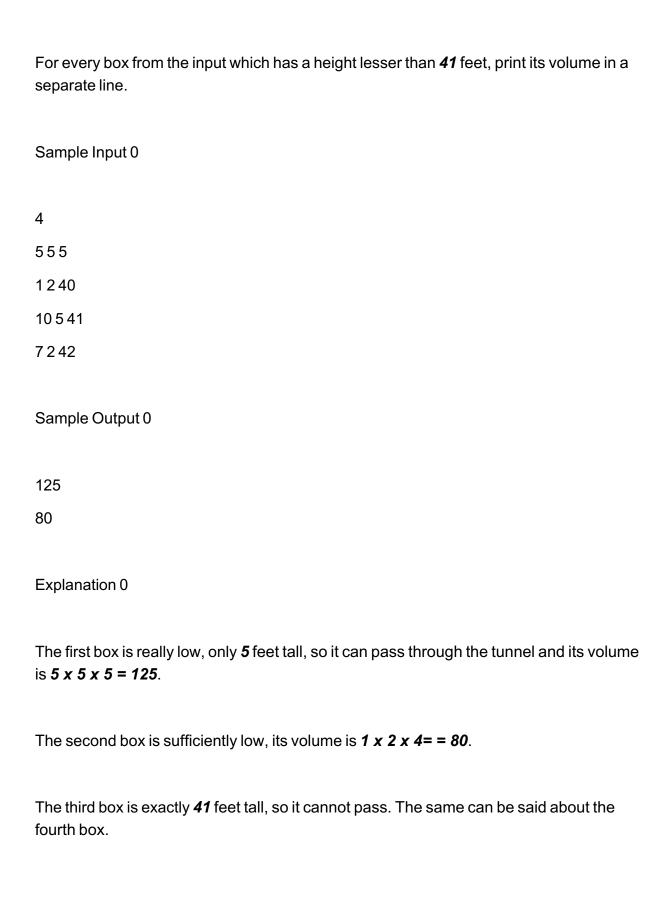
n lines follow with three integers on each separated by single spaces -length<sub>i</sub>, width<sub>i</sub> and height<sub>i</sub> which are length, width and height in feet of the i-th box.

Constraints

 $1 \le n \le 100$ 

 $1 \le length_i$ , width<sub>i</sub>, height<sub>i</sub>  $\le 100$ 

Output Format



```
#include <stdio.h>
2 v int main(){
 3
        int n;
        scanf("%d",&n);
 4
        for(int i=0;i<n;i++){</pre>
5 ,
 6
             int length,width,h;
            scanf("%d %d %d",&length,&width,&h);
7
             if(h<41){
8 ,
                 int volume = length*width*h;
9
                 printf("%d\n",volume);
10
11
12
            }
13
14
```

	Input	Expected	Got	
~	4 5 5 5 1 2 40 10 5 41 7 2 42	125 80	125 80	~
asse)	7 2 42 d all tests!	~		

You are given n triangles, specifically, their sides  $a_i$ ,  $b_i$  and  $c_i$ . 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:

$$S = \ddot{O} p * (p - a) * (p - b) * (p - c)$$
 where  $p = (a + b + c) / 2$ .

Input Format

First line of each test file contains a single integer n. n lines follow with  $a_i$ ,  $b_i$  and  $c_i$  on each separated by single spaces.

Constraints

$$1 \le n \le 100$$
  
 $1 \le a_i, b_i, c_i \le 70$   
 $a_i + b_i > c_i, a_i + c_i > b_i \text{ and } b_i + c_i > a_i$ 

**Output Format** 

Print exactly n lines. On each line print 3 integers separated by single spaces, which are  $a_i$ ,  $b_i$  and  $c_i$  of the corresponding triangle.

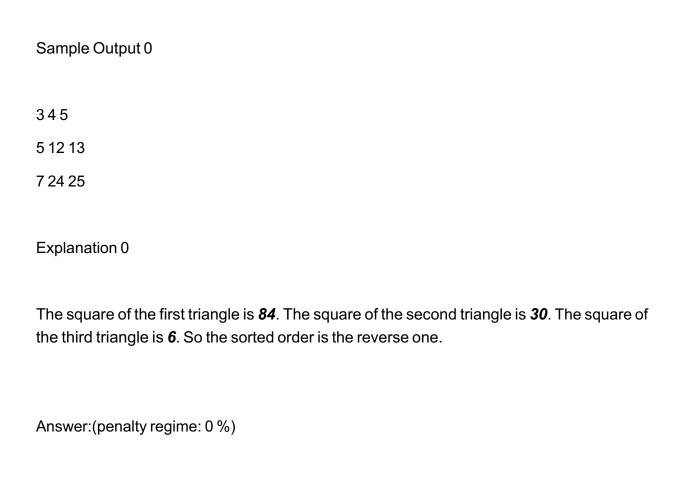
Sample Input 0

3

7 24 25

5 12 13

345



```
#include <stdio.h>
    #include <math.h>
 2
 3
    #include <stdlib.h>
4 v typedef struct{
 5
        double area;
        int a,b,c;
 6
 7
 8
   Triangle;
9 | double calculate_area(int a,int b,int c){
10
        double p=(a+b+c)/2.0;
11
        return sqrt(p*(p-a)*(p-b)*(p-c));
12
13 | int compare (const void*x,const void*y){
14
        Triangle *t1 = (Triangle *)x;
        Triangle *t2 = (Triangle *)y;
15
        if(t1->area <t2->area) return -1;
16
17
        if(t1->area >t2->area) return 1;
18
        return 0;
19
20 v int main(){
21
        int n;
        scanf("%d",&n);
22
        Triangle triangles[n];
23
24 •
        for(int i=0;i<n;i++){</pre>
25
            int a,b,c;
            scanf("%d %d %d",&a,&b,&c);
26
            triangles[i].a=a;
27
28
            triangles[i].b=b;
29
            triangles[i].c=c;
30
            triangles[i].area=calculate_area(a,b,c);
31
32
        qsort(triangles,n,sizeof(Triangle),compare);
33
        for(int i=0;i<n;i++){</pre>
34 •
35
            printf("%d %d %d\n",triangles[i].a,triangles[i].b,triangles[i].c);
36
37
38
        return 0;
39
        }
40
```

	Input	Expected	Got	
~	3 7 24 25 5 12 13 3 4 5		3 4 5 5 12 13 7 24 25	~