MOWNISWARAN.R

ECE-D

240801209

Question 1:Boxes through a Tunnel

Problem Statement:

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 is 41 feet and the width can be assumed to be infnite. 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 frst line contains a single integer n, denoting the number of boxes.

n lines follow with three integers on each separated by single spaces - lengthi, widthi and

heighti which are length, width and height in feet of the i-th box.

Constraints

$1 \leq n \leq 100$

$1 \leq$ lengthi, widthi, heighti ≤ 100

Output Format

For every box from the input which has a height lesser than 41 feet, print its volume in a

separate line.

Sample Input

4

5 5 5

1240

10541

7242

Sample Output

125

80

Explanation

The frst box is really low, only 5 feet tall, so it can pass through the tunnel and its volume is 5 x 5 x 5 = 125.

The second box is sufciently low, its volume is 1 x 2 x 4= = 80.

The third box is exactly 41 feet tall, so it cannot pass. The same can be said about the fourth box.

```c
#include <stdio.h>
int main(){
    int n;
    scanf("%d",&n);
    for (int i=0;i<n;i++){
        int length,width,height;
        scanf("%d %d %d",&length,&width,&height);

        if(height < 41){
            int volume=length*width*height;
            printf("%d\n",volume);
        }
    }
}
```

	Input	Expected	Got	
✓	4 5 5 5 1 2 40 10 5 41 7 2 42	125 80	125 80	✓

Question 2:Small Triangles, Large Triangles

You are given n triangles, specifcally, 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 diferent.

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 fle contains single integer n. n lines follow with ai, bi and ci on each

separated by single spaces.

Constraints

$1 \le n \le 100$

$1 \le ai, bi, ci \le 70$

$ai + bi > ci$, $ai + ci > bi$ 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

3

3

72425

51213

3 4 5

Sample Output

3 4 5

51213

72425

Explanation

The square of the frst 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.

4

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
typedef struct {
    double area;
    int a,b,c;
}Triangle;

double calculate_area(int a,int b,int c){
    double p=(a+b+c)/2.0;
    return sqrt(p*(p-a)*(p-b)*(p-c));
}
int compare(const void*x,const void*y){
    Triangle *t1=(Triangle *)x;
    Triangle *t2=(Triangle *)y;
    if (t1->area < t2->area) return -1;
    if (t1->area > t2->area) return 1;
    return 0;
}
int main(){
    int n;
    scanf("%d",&n);
    Triangle triangles[n];

    for (int i=0; i<n;i++){
        int a,b,c;
        scanf("%d %d %d",&a,&b,&c);

        triangles[i].a = a;
        triangles[i].b = b;
        triangles[i].c = c;
        triangles[i].area = calculate_area(a,b,c);
    }

    qsort(triangles, n, sizeof(Triangle),compare);

    for(int i=0;i<n;i++){
        printf("%d %d %d\n",triangles[i].a, triangles[i].b, triangles[i].c);
    }
    return 0;
}
```

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

Passed all tests! ✓

5