

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

SampleInput

4

5 5 5

1 2 40

10 5 41

7 2 42

SampleOutput

125

80

Explanation

The first box is really low, only 5 feet tall, so it can pass through the tunnel and its volume is $5 \times 5 \times 5 = 125$.

The second box is sufficiently low, its volume is $1 \times 2 \times 40 = 80$.

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

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

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

Question2: SmallTriangles, LargeTriangles

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 the area 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 single integer n . n lines follow with a_i , b_i and c_i each separated by single spaces.

Constraints

$$1 \leq n \leq 100$$

$$1 \leq a_i, b_i, c_i \leq 70$$

$$a_i + b_i > c_i, a_i + c_i > b_i \text{ and } b_i + c_i > a_i$$
 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

3

72425

51213

3 4 5

SampleOutput

3 4 5

51213

72425

Explanation

Thesquareofthefirsttriangleis84.Thesquareofthesecondtriangleis30.Thesquareof
thethirdtriangleis6.So,thesortedorderisthereverseone.

Answer: (penalty regime: 0 %)

```
1 #include <stdio.h>
2 #include <math.h>
3 #include <stdlib.h>
4 typedef struct {
5     double area;
6     int a,b,c;
7 }Triangle;
8
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;
15     Triangle *t2=(Triangle *)y;
16     if (t1->area < t2->area) return -1;
17     if (t1->area > t2->area) return 1;
18     return 0;
19 }
20 int main(){
21     int n;
22     scanf("%d",&n);
23     Triangle triangles[n];
24
25     for (int i=0; i<n;i++){
26         int a,b,c;
27         scanf("%d %d %d",&a,&b,&c);
28
29         triangles[i].a = a;
30         triangles[i].b = b;
31         triangles[i].c = c;
32         triangles[i].area = calculate_area(a,b,c);
33     }
34
35     qsort(triangles, n, sizeof(Triangle),compare);
36
37     for(int i=0;i<n;i++){
38         printf("%d %d %d\n",triangles[i].a, triangles[i].b, triangles[i].c);
39     }
40     return 0;
41 }
```

	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! ✓