Question 1

Correct

Flag question

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 - $length_i$, $width_i$ and $height_i$ which are length, width and height in feet of the ***i***-th box.

Constraints

$1 \le n \le 100$

$1 \le length_i, width_i, height_i \le 100$

Output Format

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

Question 1

Correct

Flag question

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 - $\boldsymbol{length_i}$, $\boldsymbol{width_i}$ and $\boldsymbol{height_i}$ which are length, width and height in feet of the ***i***-th box.

Constraints

$1 \leq n \leq 100$

$1 \leq length_i, width_i, height_i \leq 100$

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 0

```
4
5 5 5
1 2 40
10 5 41
7 2 42
```

Sample Output 0

```
125
80
```

Explanation 0

Answer: (penalty regime: 0 %)

```
#include<stdio.h>
int main(){
int n;
scanf("%d",&n);
for(int i=0;i<n;i++
int length,width,he
scanf("%d %d %d",&l
if(height<41){
int volume=length*w
printf("%d\n",volum
}
}
return 0;
}
```

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

Passed all tests! ✓

estion 2

rrect

Flag question

You are given n triangles, specifical
sides a_i, b_i and c_i. Print them in the
but sorted by their areas from the s
one to the largest one. It is guarant
the areas are different.

The best way to calculate a volume
triangle with sides ***a***, ***b*** and ***c*** is Her
formula:

$S = \sqrt{p * (p - a) * (p - b) * (p - c)}$ w
$+ b + c) / 2$.

Input Format

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

Constraints

$1 \le n \le 100$

Output Format

Print exactly ***n*** lines. On each line pr
integers separated by single spaces
a_i, b_i and c_i of the corresponding tri

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***
square of the second triangle is ***30***.
square of the third triangle is ***6***. So t
order is the reverse one.

Answer: (penalty regime: 0 %)

```
1  #include<stdio.h>
2  #include<math.h>
3  #include<stdlib.h>
4  typedef struct {
5      int a,b,c;
6      double area;
7  }triangle;
8  double calculate_area
9      double p=(a+b+c)
10     return sqrt(p*(p
11 }
12 int compare(const vo
13     triangle*tri1=(t
14     triangle*tri2=(t
```

```
15      if(tri1->area<tr
16      return-1;
17      if(tri1->area>tr
18      return 1;
19      return 0;
20  }
21  int main(){
22      int n;
23      scanf("%d",&n);
24      triangle triangl
25      for(int i=0;i<n;
26          int a,b,c;
27          scanf("%d %d
28          triangles[i]
29          triangles[i]
30          triangles[i]
31          triangles[i]
32      }
33      qsort(triangles,
34      for(int i=0;i<n;
35          printf("%d %
36      }
37      return 0;
38  }
```

	Input	E
✓	3 7 24 25 5 12 13 3 4 5	3 5 7

Passed all tests! ✓