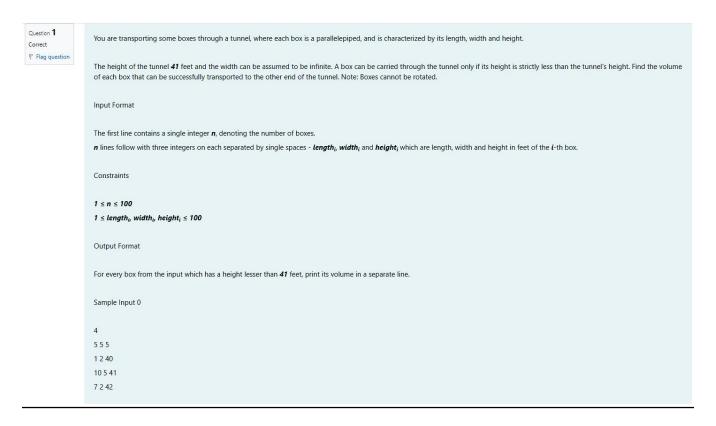
Week-14-Structures and Unions

Week-14-01-Practice Session-Coding



Source code

Answer: (penalty regime: 0 %)

```
#include<stdio.h>
 2
 3 +
    int main(){
        int n;
4
        scanf("%d",&n);
 5
 6
 7 +
        struct boxes{
8
            int 1,b,h,v;
9
        } box[n];
10
11 +
        for(int i=0;i<n;i++){</pre>
            scanf("%d %d %d",&box[i].1,&box[i].b,&box[i].h);
12
            if(box[i].h<41){
13 +
                 box[i].v=box[i].l*box[i].b*box[i].h;
14
15
                 printf("%d\n",box[i].v);
16
            }
17
18
        return 0;
19
```

Result

~	4	125	125	~
•	5 5 5 1 2 40 10 5 41 7 2 42	80	80	

Question **2**Correct

Flag question

You are given n triangles, specifically, their sides a_l , b_l and c_l . 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 ${\it a}$, ${\it b}$ and ${\it c}$ 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 ≤ n ≤ 100

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

 $a_i + b_i > c_i$, $a_i + c_i > b_i$ 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 3 4 5

Activate Windows

Source code

```
Answer: (penalty regime: 0 %)
       #include<stdio.h>
       #include<math.h>
   2
   3
   4
       int main(){
   5
           int n;
   6
           scanf("%d",&n);
   7
   8
           struct triangle{
   9
               int a,b,c;
  10
               int area;
  11
           } triangles[n];
  12
           for(int i=0;i<n;i++){</pre>
  13
               scanf("%d %d %d",&triangles[i].a,&triangles[i].b,&triangles[i].c);
  14
  15
               int p=(triangles[i].a+triangles[i].b+triangles[i].c)/2;
  16
               triangles[i].area=sqrt(p*(p-triangles[i].a)*(p-triangles[i].b)*(p-triangles[i].c));\\
  17
  18
           for(int i=0;i<n-1;i++){
  19
               for(int j=0;j<n-i-1;j++){
  20 4
                    if(triangles[j].area>triangles[j+1].area){
  21
  22
                        struct triangle temp=triangles[j];
  23
                        triangles[j]=triangles[j+1];
  24
                        triangles[j+1]=temp;
  25
  26
   27
  28
  29 ,
           for(int i=0;i<n;i++){</pre>
  30
               printf("%d %d %d\n",triangles[i].a,triangles[i].b,triangles[i].c);
  31
  32
           return 0;
  33
```

Result

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