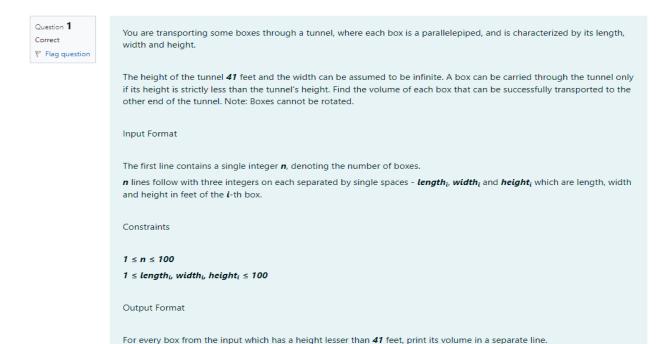
Name: Kirithick Raja

Roll no.: 240701627

Week-14

Question 1:



Source code:

```
Answer: (penalty regime: 0 %)
      #include<stdio.h>
       struct Box
   3 +
            int length;
            int width;
            int height;
       };
union Volume
  10
            int value;
  11 };
12 int main()
13 v {
  14
            const int tunnel_height=41;
scanf("%d",&n);
  15
  16
   17
            struct Box boxes[n];
  18
            union Volume volume;
            for(int i=0;i<n;i++)
  19
   20 +
   21
                scanf("%d %d %d",&boxes[i].length,&boxes[i].width,&boxes[i].height);
   22
   23
            for(int i=0;i<n;i++)
   25
                if(boxes[i].height<tunnel_height)</pre>
  26 4
   27
                     volume.value=boxes[i].length*boxes[i].width*boxes[i].height;
   28
                    printf("%d\n",volume.value);
   29
   30
            return 0;
   31
   32
```

Result:

	Input	Expected		
/	4	125	125	~
	5 5 5	80	80	
	1 2 40			
	10 5 41			
	7 2 42			

Question 2:

Question **2**Correct

Flag question

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 ${\it a}, {\it b}$ 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.

Source code:

```
#include<stdio.h>
     #include<math.h>
     typedef struct
 3
 4 + {
 5
         int a,b,c;
         double area;
 8
    Triangle;
    double calculate_area(int a,int b,int c)
10 + {
11
         double p=(a+b+c)/2.0;
12
         return sqrt(p*(p-a)*(p-b)*(p-c));
13
14
    void sort_triangles(Triangle triangles[],int n)
15 1
         for(int i=0;i<n;i++)</pre>
16
17
18
              for(int j=0;j<n-i-1;j++)
19 1
                  if(triangles[j].area>triangles[j+1].area)
20
21 1
                      Triangle temp=triangles[j];
22
23
                      triangles[j]=triangles[j+1];
                      triangles[j+1]=temp;
24
25
26
27
28
    int main()
30
31
32
         scanf("%d",&n);
         Triangle triangles[n];
33
34
         for(int i=0;i<n;i++)
35
             scanf("%d %d %d",&triangles[i].a,&triangles[i].b,&triangles[i].c);
triangles[i].area=calculate_area(triangles[i].a,triangles[i].b,triangles[i].c);
36
37
38
         sort_triangles(triangles,n);
39
40
         for(int i=0;i<n;i++)</pre>
41
             printf("%d %d %d\n",triangles[i].a,triangles[i].b,triangles[i].c);\\
42
43
44
         return 0;
45
```

Result:

<u>.</u>		Expected		
✓ 3		3 4 5	3 4 5	~
7 2	4 25	5 12 13	5 12 13	
5 1	2 13	7 24 25	7 24 25	
3 4	5			