REC-CIS

QUIZ HAVIGATION Show one page at a time Finish review

Status Finished

Started Monday, 13 January 2025, 11:03 AM Completed Monday, 13 January 2025, 11:38 AM

Duration 34 mins 52 secs

Ouestion 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 other end of the tunnel. Note: Boxes cannot be rotated.

Input Format

and height in feet of the i-th box. Constraints

1 ≤ n ≤ 100

1 ≤ length;, width;, height; ≤ 100

The first line contains a single integer n, denoting the number of boxes.

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

n lines follow with three integers on each separated by single spaces - length, width and height, which are length, width

REC-CIS 1 ≤ n ≤ 100 $1 \le length_i$, width_i, height_i ≤ 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 0 555 1 2 40 10 5 41 7 2 42 Sample Output 0 125 80 Explanation 0 The first 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. Explanation 0

The first 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 sufficiently low, its volume is $1 \times 2 \times 4 = 80$.

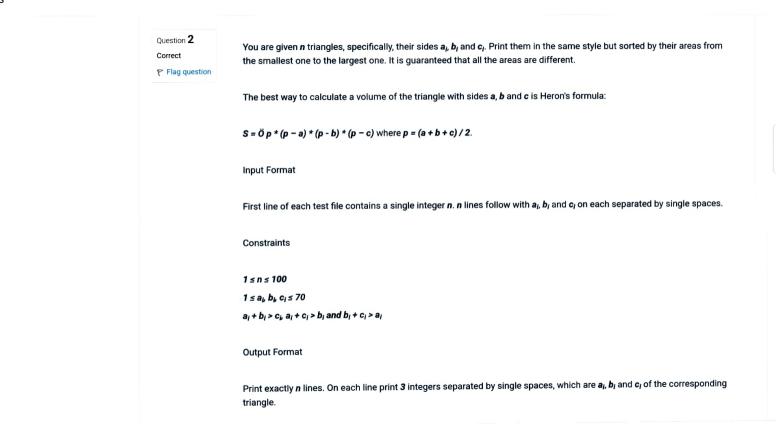
The third box is exactly 41 feet tall, so it cannot pass. The same can be said about the fourth box.

Answer: (penalty regime: 0 %)

```
|#include<stdio.h>
    int main()
        int n:
        scanf("%d",&n);
        for(int i=0;i<n;i++)</pre>
            int length, width, height;
            scanf("%d %d %d",&length,&width,&height);
10
11
            if(height<41)
12 ▼
                int volume=length*width*height;
13
14
                printf("%d\n",volume);
15
16
17 }
```

```
Answer: (penalty regime: 0 %)
      #include<stdio.h>
      int main()
          int n;
          scanf("%d",&n);
          for(int i=0;i<n;i++)
   8
              int length, width, height;
   9
              scanf("%d %d %d",&length,&width,&height);
  10
  11
               if(height<41)
  12 ₩
  13
                   int volume=length*width*height;
  14
                   printf("%d\n",volume);
  15
  16
  17 }
```



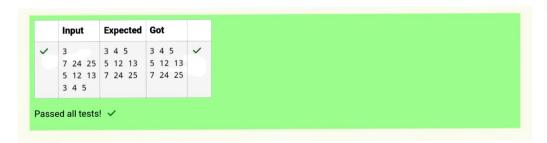


REC-CIS	
	Sample Input 0
	3
	7 24 25
	5 12 13
	345
	Sample Output 0
	345
	5 12 13
	7 24 25
	Explanation 0
	The square of the first triangle is 84 . The square of the second triangle is 30 . The square of the third triangle is 6 . So the sorted order is the reverse one.
	sorted order is the reverse one.
	Answer: (penalty regime: 0 %)
	1 #include <stdio.h> 2 #include<math.h></math.h></stdio.h>
	3 #include <stdlib.h></stdlib.h>
	4 typedef struct 5 ▼ {

```
Answer: (penalty regime: 0 %)
      #include<stdio.h>
      #include<math.h>
      #include<stdlib.h>
      typedef struct
   5 ▼ {
          double area;
          int a,b,c;
      }Triangle;
      double calculate_area(int a,int b,int c)
  11 ▼ {
  12
          double p=(a+b+c)/2.0;
  13
          return sqrt(p*(p-a)*(p-c));
  14
  15
      int compare(const void*x,const void*y)
  16 ▼ {
  17
          Triangle *t1=(Triangle *)x;
  18
          Triangle *t2=(Triangle *)y;
          if(t1->area<t2->area) return -1;
  20
          if(t1->area>t2->area) return 1;
  21
          return 0;
  22
      int main()
  24 ▼ {
  25
          int n;
  26
          scanf("%d",&n);
  27
          Triangle triangles[n];
          for(int i=0;i<n;i++)
  28
  29 ₹
  30
               int a,b,c;
               scanf("%d %d %d",&a,&b,&c);
  31
  32
               triangles[i].a=a;
  33
               triangles[i].b=b;
  34
               triangles[i].c=c;
  35
               triangles[i].area=calculate area(a,b,c);
```

```
12
        double p=(a+b+c)/2.0;
13
        return sqrt(p*(p-a)*(p-c));
14
    int compare(const void*x,const void*y)
15
16 ▼ {
17
        Triangle *t1=(Triangle *)x;
18
        Triangle *t2=(Triangle *)y;
        if(t1->area<t2->area) return -1;
19
        if(t1->area>t2->area) return 1;
20
21
        return 0:
22
23
    int main()
24 ▼ {
25
        int n:
        scanf("%d",&n);
26
        Triangle triangles[n];
27
28
        for(int i=0;i<n;i++)
29 ₹
            int a,b,c;
30
            scanf("%d %d %d",&a,&b,&c);
31
32
            triangles[i].a=a;
            triangles[i].b=b;
33
34
            triangles[i].c=c;
            triangles[i].area=calculate_area(a,b,c);
35
36
37
        qsort(triangles,n,sizeof(Triangle),compare);
38
39
        for(int i=0;i<n;i++)
40 ▼
            printf("%d %d %d\n",triangles[i].a,triangles[i].b,triangles[i].c);
41
42
43
44
        return 0;
45 }
```

```
int a,b,c;
30
            scanf("%d %d %d",&a,&b,&c);
31
            triangles[i].a=a;
32
            triangles[i].b=b;
33
34
            triangles[i].c=c;
            triangles[i].area=calculate_area(a,b,c);
36
37
        gsort(triangles,n,sizeof(Triangle),compare);
38
39
        for(int i=0;i<n;i++)</pre>
40 ₩
            printf("%d %d %d\n",triangles[i].a,triangles[i].b,triangles[i].c);
41
42
        return 0;
45 }
```



Finish review