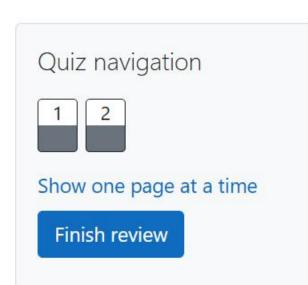
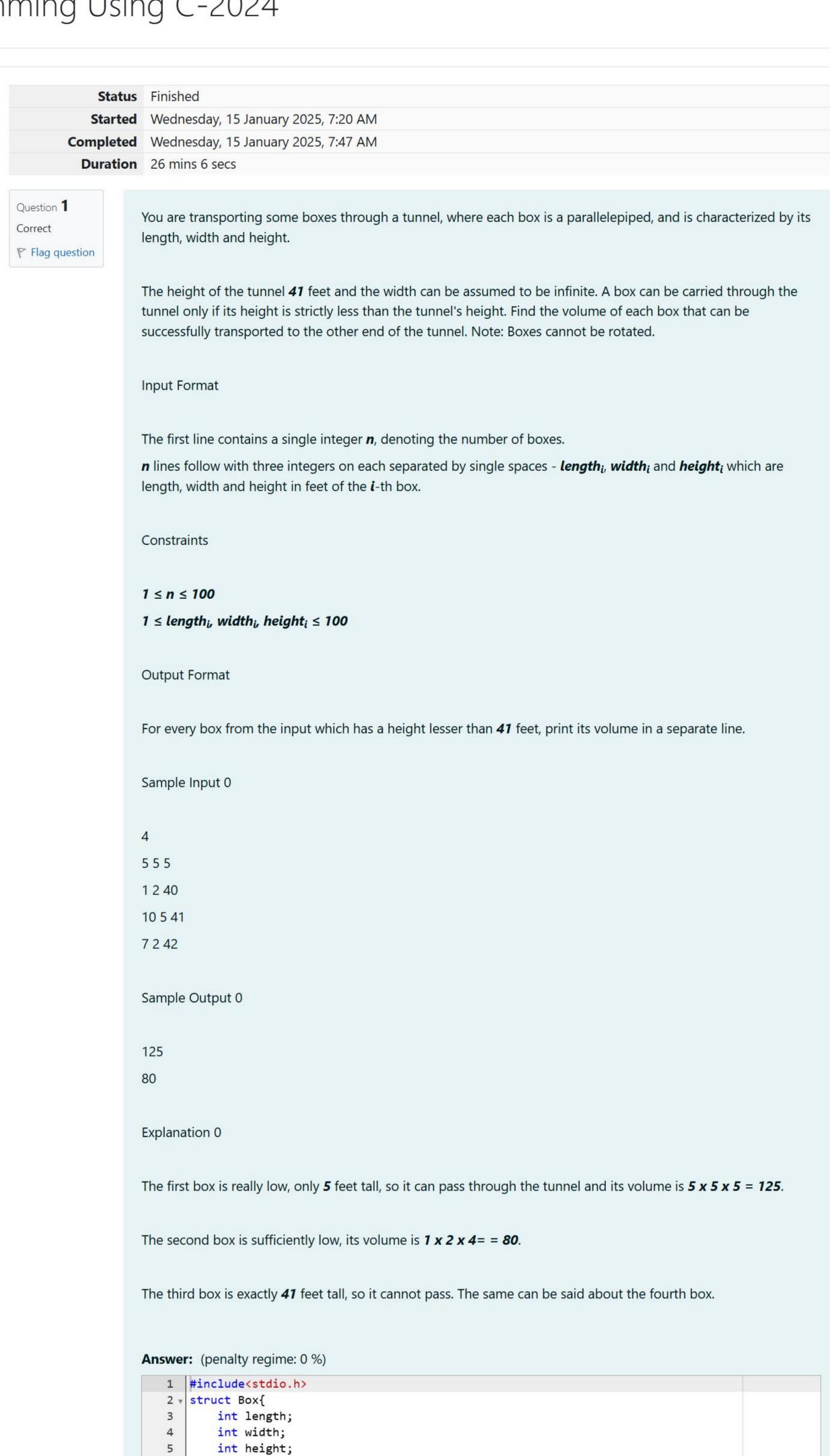
REC-CIS

GE23131-Programming Using C-2024







scanf("%d%d%d",&boxes[i].length,&boxes[i].width,&boxes[i].height);

printf("%d\n",boxes[i]. length * boxes[i]. width * boxes[i]. height);

Question 2

The best way to calculate a volume of the triangle with sides \boldsymbol{a} , \boldsymbol{b} and \boldsymbol{c} is Heron's formula:

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

```
S = \ddot{O} p * (p - a) * (p - b) * (p - c) where <math>p = (a + b + c) / 2.
```

from the smallest one to the largest one. It is guaranteed that all the areas are different.

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

Input Format

6 7

8 v 9

10

11

12 13 v

14

15 16 v

17 18 19

20 21 int main()

int n;

return 0;

scanf("%d",&n);

struct Box boxes[n];

for (int i=0;i<n;i++)</pre>

if(boxes[i].height<41)</pre>

Spaces.

Constraints

 $1 \le n \le 100$ $1 \le a_i, b_i, c_i \le 70$

Output Format

 $a_i + b_i > c_i$, $a_i + c_i > b_i$ and $b_i + c_i > a_i$

Output Format

Corresponding triangle.

Sample Input 0

Print exactly n lines. On each line print a integers separated by single spaces, which are a_i , b_i and c_i of the

7 24 25 5 12 13

3

3 4 5

Sample Output 0
3 4 5

Answer: (penalty regime: 0 %)

#include<stdio.h>

7 24 25

5 12 13

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.

```
#include<math.h>
    struct Triangle
4 v {
 5
        int a,b,c;
 6
        double area;
    int main()
 8
9
        int n;
10
11
        scanf("%d",&n);
        struct Triangle triangles[n];
12
13
        for (int i=0;i<n;i++)</pre>
14 *
15
             scanf("%d%d%d",&triangles[i].a,&triangles[i].b,&triangles[i].c);
            double p=(triangles[i].a + triangles[i].b + triangles[i].c)/2.0;
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++)</pre>
19
20
            for(int j=i+1; j<n; j++)</pre>
21
22 1
                 if(triangles[i].area>triangles[j].area)
23
24 1
25
                     struct Triangle temp = triangles[i] ;
                     triangles[i] = triangles[j] ;
26
                     triangles[j] = temp ;
27
28
29
30
        for(int i=0;i<n;i++)</pre>
31
32 1
             printf("%d %d %d\n",triangles[i].a,triangles[i].b,triangles[i].c);
33
34
35
        return 0;
36
37
38
39
40
41
42
43
44
45
46
47
48
```