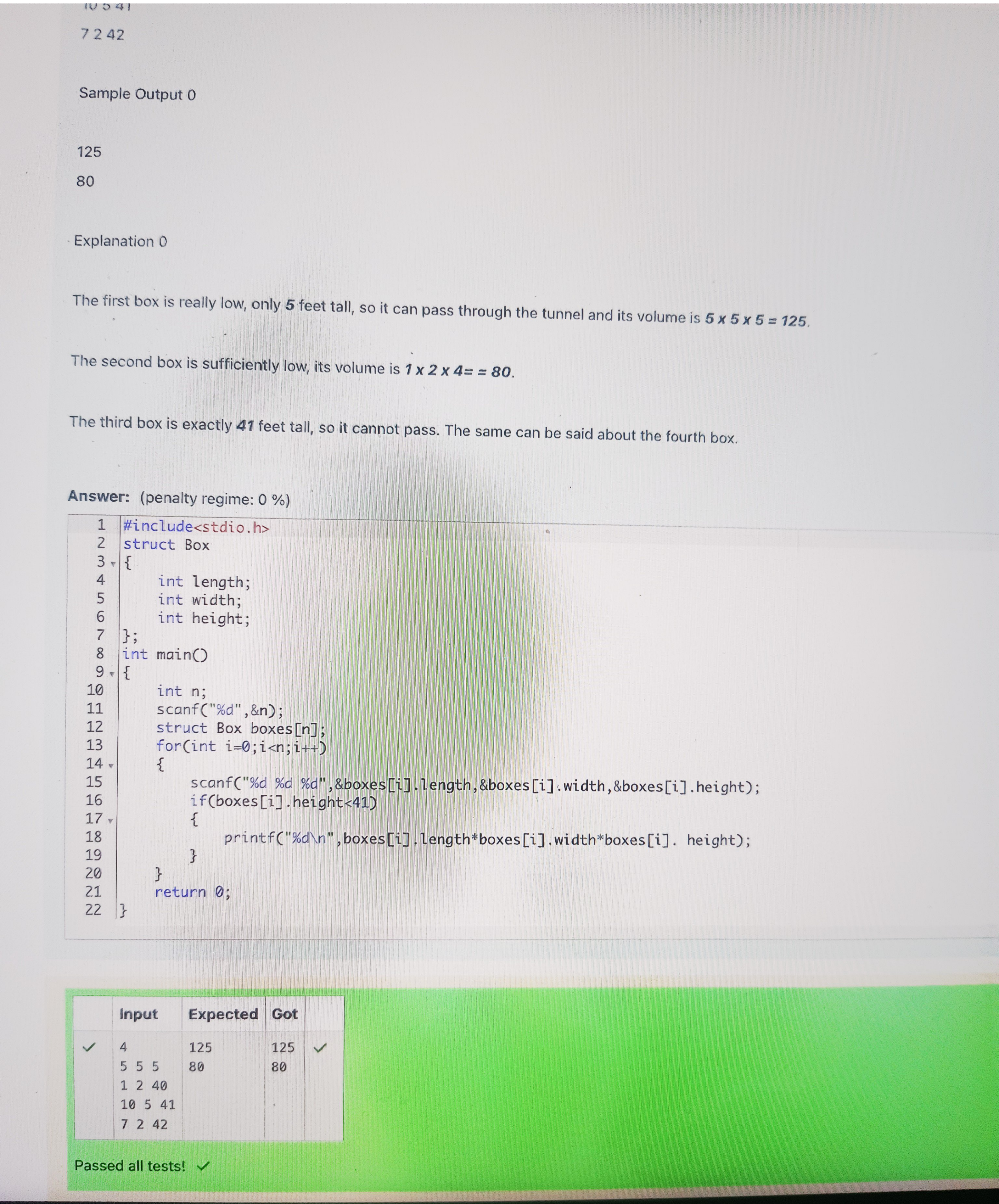


10 5 41

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***.

The second box is sufficiently low, its volume is ***1 x 2 x 4= = 80***.

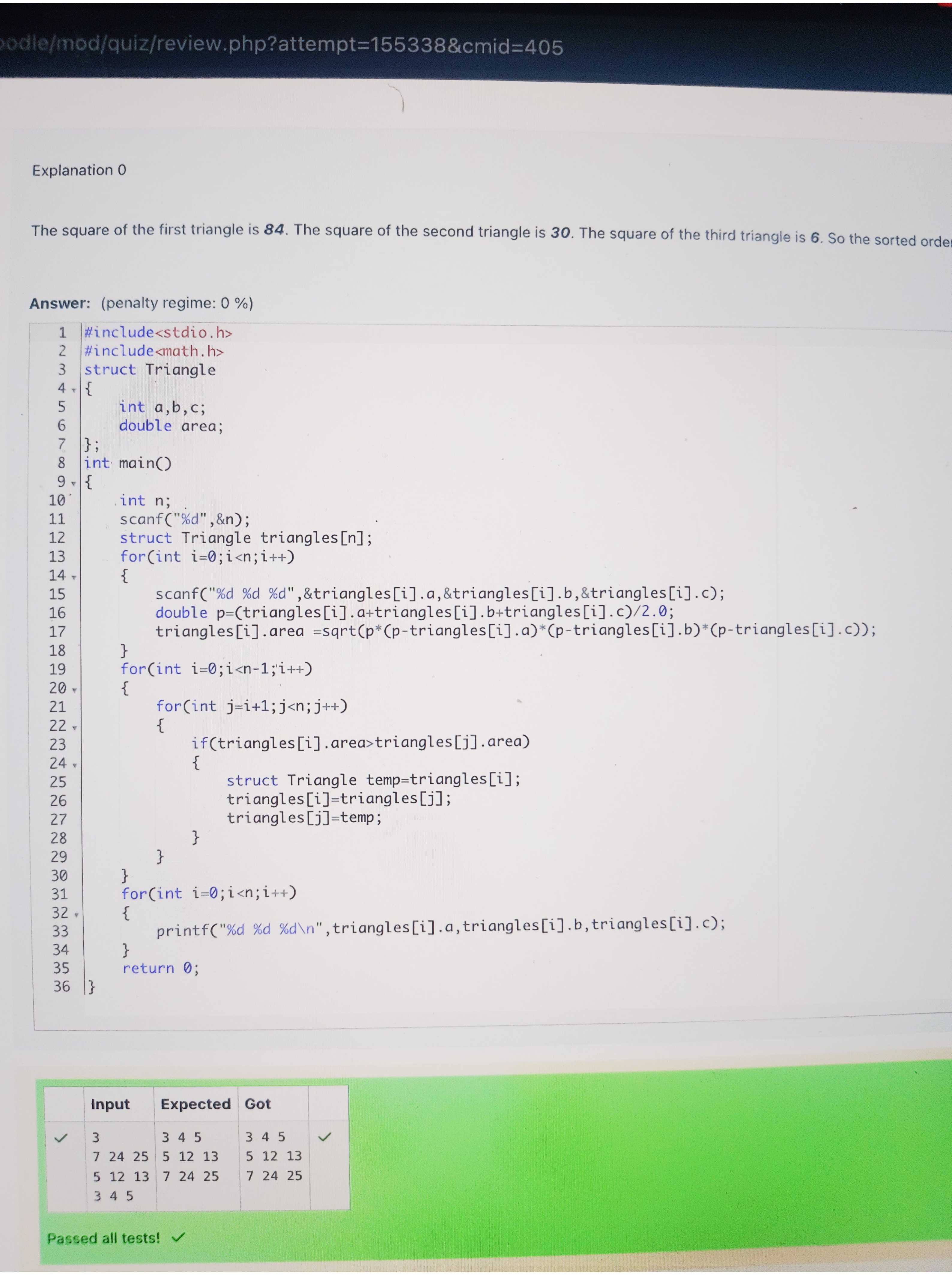
The third box is exactly ***41*** feet tall, so it cannot pass. The same can be said about the fourth box.

Answer: (penalty regime: 0 %)

```
#include<stdio.h>
struct Box
{
    int length;
    int width;
    int height;
};
int main()
{
    int n;
    scanf("%d",&n);
    struct Box boxes[n];
    for(int i=0;i<n;i++)
    {
        scanf("%d %d %d",&boxes[i].length,&boxes[i].width,&boxes[i].height);
        if(boxes[i].height<41)
        {
            printf("%d\n",boxes[i].length*boxes[i].width*boxes[i]. height);
        }
    }
    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! ✓



oodle/mod/quiz/review.php?attempt=155338&cmid=405

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 orde

Answer: (penalty regime: 0 %)

```
#include<stdio.h>
#include<math.h>
struct Triangle
{
    int a,b,c;
    double area;
};
int main()
{
    int n;
    scanf("%d",&n);
    struct Triangle triangles[n];
    for(int i=0;i<n;i++)
    {
        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;
        triangles[i].area =sqrt(p*(p-triangles[i].a)*(p-triangles[i].b)*(p-triangles[i].c));
    }
    for(int i=0;i<n-1;i++)
    {
        for(int j=i+1;j<n;j++)
        {
            if(triangles[i].area>triangles[j].area)
            {
                struct Triangle temp=triangles[i];
                triangles[i]=triangles[j];
                triangles[j]=temp;
            }
        }
    }
    for(int i=0;i<n;i++)
    {
        printf("%d %d %d\n",triangles[i].a,triangles[i].b,triangles[i].c);
    }
    return 0;
}
```

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

Passed all tests! ✓