

## Explanation 0

The first box is really low, only **5** feet tall, so it can pass through the tunnel and its volume is  **$5 \times 5 \times 5 = 125$** .

The second box is sufficiently low, its volume is  **$1 \times 2 \times 4 = 8$** .

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 %)

```
1 #include<stdio.h>
2 int main()
3 {
4     int n;
5     scanf("%d",&n);
6     for(int i=0;i<n;i++)
7     {
8         int length,width,height;
9         scanf("%d %d %d",&length,&width,&
10            height);
11        if(height<41)
12        {
13            int volume=length*width*height;
14            printf("%d\n",volume);
15        }
16    }
17 }
```

	Input	Expected	Got	
✓	4	125	125	✓
	5 5 5	80	80	
	1 2 40			
	10 5 41			
	7 2 42			

Passed all tests! ✓

**Answer:** (penalty regime: 0 %)

```
1 #include<stdio.h>
2 #include<math.h>
3 #include<stdlib.h>
4 typedef struct
5 {
6     int a,b,c;
7     double area;
8 }
9 Triangle;
10 double calculate_area(int a,int b,int c){
11     double p=(a+b+c)/2.0;
12     return sqrt(p*(p-a)*(p-b)*(p-c));
13 }
14 int compare (const void *t1,const void *t2)
15 {
16     Triangle *tri1=(Triangle*)t1;
17     Triangle *tri2=(Triangle*)t2;
18     if(tri1->area < tri2->area)
19         return -1;
20     if(tri1->area > tri2->area)
21         return 1;
22     return 0;
23 }
24 int main()
25 {
26     int n;
27     scanf("%d",&n);
28     Triangle triangles[n];
29     for(int i=0;i<n;i++)
30     {
31         int a,b,c;
32         scanf("%d %d %d",&a,&b,&c);
33         triangles[i].a=a;
34         triangles[i].b=b;
35         triangles[i].c=c;
36         triangles[i].area=calculate_area(a,b,c);
37     }
38     qsort(triangles,n,sizeof(Triangle),compare);
39     for(int i=0;i<n;i++)
40     {
41         printf("%d %d %d\n",triangles[i].a,triangles[i].b,triangles[i].c);
42     }
43     return 0;
44 }
```

	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! ✓