WEEK - 14

Week-14-Structures and Unions

Week-14-Structures and Unions ✓ Done

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

```c
#include<stdio.h>
#include<math.h>
#include<stdlib.h>
typedef struct{
    int a, b, c;
    double area;
} triangle;
double c_area(int a, int b, int c){
    double p = (a + b + c) / 2.0;
    return sqrt(p * (p - a) * (p - b) * (p - c));
}
int compare(const void * t1, const void * t2){
    triangle * tri1 = (triangle *)t1;
    triangle * tri2 = (triangle *)t2;
    if(tri1 -> area < tri2 -> area)
    return -1;
    if(tri1 -> area > tri2 -> area)
    return 1;
    return 0;
}
int main(){
    int n;
    scanf("%d", &n);
    triangle triangles[n];
    for(int i = 0; i < n; i++){
        int a, b, c;
        scanf("%d %d %d", &a, &b, &c);
        triangles[i].a = a;
        triangles[i].b = b;
        triangles[i].c = c;
        triangles[i].area = c_area(a, b, c);
    }
    qsort(triangles, n, sizeof(triangle), compare);
    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! ✓