

# ESS201: C++ Programming

Jaya Sreevalsan Nair \*

International Institute of Information Technology, Bangalore

Term I: 2017-18 (Lab on 2017-10-10)

Tasks:

1. Write a single C++ program, named `geometry3d.cpp` which uses struct to define various three-dimensional geometric primitives.

```
struct triangle;  
struct line;  
struct point;
```

`point` is the basic building block with `vector3d` as position coordinates; `line` consists of two `vector3d` members; and `triangle` consists of three `vector3d` members. Build with the `vector3d` struct you have already built in the previous lab.

Hint: You could rename your `vector3d.cpp` as `geometry3d.cpp` as a starting point, and work with the same file.

- The function definitions are:  
`vector3d normalize(vector3d a)`  
`float get_angle_between_tri_points(point a, point b, point c)`  
`float get_length(line a)`  
`float get_area(triangle a)`  
`vector3d get_normal(triangle a)`

Hint: normalize a `vector3d` using its norm, find angle using dot product between `ba` and `bc`; use difference between points on line to find length; find area using lengths of sides of triangle ( $A = \sqrt{s(s-a)(s-b)(s-c)}$  for semi-perimeter  $s$ , and lengths of sides  $a, b, c$ ), find normal using cross product between any two sides of the triangle.

Hint: overload `get_length` and `get_angle_between_tri_points` using `vector3d` input parameters.

---

\*(jnair@iiitb.ac.in)

- Input: should take in 3 lines of input at commandline prompt, which are points on a triangle. e.g. p1, p2, p3, are as follows:  

```

3 4 5
1.2 4.2 4.1
0.5 3 2

```
- Output: should be in 4 lines, i.e.  
the first line of output prints the lengths of the three sides of the triangle (i.e. length between p1 and p2, p2 and p3, and p3 and p1),  
the second line of output prints the three angles of the triangle (in radians between (a) p1, p2, and p3, (b) p2, p3, and p1, and (c) p3, p1, and p2),  
the third line of output prints the area of the triangle, and  
the fourth line prints the normalized vector (or unit vector) of the normal of the triangle. For the afore-mentioned triangle, the results are:  

```

2.02237 2.51794 4.03113
2.17908 0.42432 0.538191
2.08941
0.358953 0.753802 -0.550395

```