**Lab 3**

**Information structures with python**

**Purpose :**

To define triangle class and run the code in test triangle class to check the triangle class.

**Deliverables:**

**Triangle.py**

**class** Triangle:  
 **def** \_\_init\_\_(self, three\_points, two\_angles):  
 self.points = three\_points  
 self.angles = two\_angles  
  
 **def** has\_point(self, a\_point):  
 *# Returns whether or not a\_point is a vertex of self* **for** i **in** range(0,len(self.points)):  
 **if** self.points[i]==a\_point: *# check every element with a\_point* **return** True  
  
 **def** point\_following(self, a\_point):  
 **if** a\_point== self.points[-1]: *# checking the last element if it is equal to a\_point return the first element* **return** self.points[0]  
 **else**:  
 **for** i **in** range(0, len(self.points)): *# else checking other points and return the self following a\_point* **if** a\_point== self.points[i]:  
 **return** self.points[i+1]  
  
  
 **def** point\_preceding(self, a\_point):  
 **if** a\_point == self.points[0]: *#checking the first element if it is equal to a\_point return the last element* **return** self.points[-1]  
 **else**:  
 **for** i **in** range(0, len(self.points)): *# else checking other points and return the self following a\_point* **if** a\_point == self.points[i]:  
 **return** self.points[i - 1]  
  
  
 **def** get\_angles(self):  
 third=self.third\_angle()  
 self.angles.append(third)  
 **return** self.angles *#simply return 3 angles of the triangle* **def** get\_points(self):  
  
 **return** self.points *#simply return all the points* **def** third\_angle(self):  
 third\_angle= 180-(self.angles[0]+self.angles[1]) *# calculating the third angle by property* **return** third\_angle *# return the third angle* **def** to\_string(self):  
 **return "\nTriangle: Vertices {0}, {1}, {2}; Angles {3}, {4}, {5}"** \  
 .format(self.points[0], self.points[1], self.points[2],  
 self.angles[0], self.angles[1], self.third\_angle())

**also attached .py file**

**Output :**

