|  |  |  |
| --- | --- | --- |
| American University of SharjahCollege of Engineering Department of Computer Science & Engineering  P. O. Box 26666, Sharjah, UAE |  | **Lab Instructor:** Praveena Kolli  **Office:** EB2126  **Phone**: 971-6-5152352  **e-mail**: pkolli@aus.edu  **Semester**: Spring 2020 |

**CMP 321 - Programming languages Laboratory**

**Lab 5 – Python Classes**

**Objectives**

* Understand and implement Classes in Python

**Please explore and make use of Python features where possible. Code that does not follow this note will be penalized.**

**Exercise 1**

Write a Polygon class that is a **sequence** of 2D points represented by **named tuples**, so that each point is given a name e.g., point ‘A’ is (4, 5). Thus: Point = namedtuple('Point', ['name', 'x', 'y'])

The class defines the following:

1. constructors, setter,getter and delete functions:
   1. A constructor that takes 3 or more points as arguments e.g., Polygon( ('A',5,0), ('B',10,5), ('C',5,10), ('D',-2,8) ), and initializes the polygon accordingly.
   2. A setter function that appends to the polygon a new point from given name and x, y coordinates. It should throw a user defined exception ExistingPointError if the point exists. Check the name of the point as well as x and y coordinates.
   3. A getter function that allows retrieving a point given its name. The function should throw a user defined exception PointNotFoundError if such point does not exist.
   4. A function that deletes a point by name. Throws PointNotFoundError.
2. Length, print and comparison functions:
   1. A function that returns the number of points in the polygon, so that len(poly) works.
   2. A function that implements comparison operator. E.g. polygon1 == polygon2.
   3. A function that allows using print(poly) to print a polygon’s points, in the following format: A: (5,6) -> B: (6,7) -> C: (12,15)
3. A function that draws a polygon on the screen, using Turtle graphics. Below is an example that draws a single line; adapt as necessary. For more info about Turtle graphics, see <https://docs.python.org/3/library/turtle.html>

import turtle

def draw\_line(p1, p2, speed=2, color='blue'):

turtle.speed(speed)

turtle.hideturtle()

turtle.penup()

turtle.goto(p1)

turtle.pendown()

turtle.color(color)

turtle.write('A')

turtle.goto(p2)

turtle.write('B')

turtle.exitonclick()

draw\_line( (0,50), (300,150) )