NIQO ROBOTICS
NO. 1074-A, 8TH CROSS, 11TH MAIN, HAL 2ND STAGE, INDIRANAGAR, BANGALORE, KA 560038
info@niqorobotics.com
www.niqorobotics.com
+91 7483302011
CIN: U74900KA2015PTC134399

Geometric Calculator

Robotics Engineer Evaluation Assignment

Allotted time: 48 hrs

Overview

The goal is to develop a geometric calculator to measure the shapes in 2D space. This application will allow the users to define certain shapes in a 2D space, and then measure the distance, area, circumference, cross-section etc.

Similar to Python's REPL (Read-Eval-Print-Loop), create a command line interface on which the user can define a set of shapes and then query the measurements. For example,

```
> p1 = Point(10, 10)
p1(10, 10)
> p2 = Point(20, 20)
p2(20, 20)
> p1.distance(p2)
14.1421356237
```

Key Features

1. Shapes
    - Point
    - Line
    - Circle
    - Rectangle
    - Union of two shapes (Optional)
    - Intersection of two shapes (Optional)
2. Queries
    - Area of a shape
    - Length / Circumference / Perimeter of a shape
    - Distance between two shapes
3. Supported data types
    - Shapes
    - Floating point numbers

Guidelines

- Seamless user experience with clear responses and consistent error handling
- For requirements which are unclear, make a fair assumption and document them in readme
- Use Python or C++ as the programming language
- Do not use libraries for core calculator requirements. Feel free to use them for non-core parts of the implementation if required.
- Do not look for a complete implementation in Google Search / ChatGPT

- Use of Google Search / ChatGPT is allowed otherwise, but understand the decisions, choices, details, and be ready to explain them
- Create a local git repo and commit at logical intervals with clear commit messages. Include the repo in your deliverable to depict your code evolution.
- Start with simple use cases, and include more complicated features as you go along. A product-with-half-the-features is better than a half-done-but-unusable-product.
- No bugs please!

Deliverables

1. Zip of the source
    - Zip your git repo and attach. Include a Readme.md.
2. Readme
    - Instructions for building, setting up and running the application
    - High level design of your application
    - Assumptions and known issues with your implementation
    - Any challenges you faced on this project
    - Anything else you would like to share

Success Criteria

- Ability to build and open the application
- Users can define the shapes according to the requirements
- Users can measure the shape parameters

For extra credits

- Optional shapes - Union / Intersection of shapes
- Any other interesting shapes or queries you can think about
- Include automated unit tests / follow Test–Driven-Development
- Support for 3D shapes and queries
- Along with geometric calculations, support for numeric calculations as well on the prompt. For example,

```
> p1 = Point(10, 10)
p1(10, 10)
> p2 = Point(20, 20)
p2(20, 20)
> p3 = Point(30, 30)
p2(20, 20)
> p1.distance(p2) + p1.distance(p3)
42.4264068712
```