SCSJ1023 Programming Technique II
Semester 1, 2018/2019

Tutorial 9(a)
Polymorphisms

Problem

Given an incomplete program in the attached file. The program is meant to be a simple geometrical editor whereby the user can create a list of geometrical objects consisting of circles and rectangles. The user will interact with the program using a menu-driven interface. It provides the following feature:

a. To add a circle into the list of objects
b. To add a rectangle into the list.
c. To display all the objects.
d. To move the position of all objects
e. End the program

Anlyze the given code mainly to identify the relationship between the classes, `Geometry`, `Circle` and `Rect`. Then complete the program according to the tasks specified in the program. Note that, you need **to create the objects dynamically**. Note also that, you may want to add other methods into the classes if necessary. For example, you may want to add the method that allows the user to enter the attributes for each object. The program should achieve the result as shown in Figure 1. Note that the **bold** texts indicate the input entered by the user in the text mode window. All the interactions shown in the figure are continuous.

Interaction 1: the user chooses to enter a rectangle

```
========== Operations =========
1. Add a circle to the list
2. Add a rectangle to the list
3. Display the list
4. Move the positions of the objects
5. Exit

Enter the operation to perform => 2


Enter the rectangle's information:
 Top Left Corner, x1 y1 => 0 200
 Bottom Right Corner, x2 y2 => 400 250

 A rectangle has been added to the list.
```

Interaction 2: the user chooses to enter a circle

```
========== Operations =========
1. Add a circle to the list
2. Add a rectangle to the list
3. Display the list
4. Move the positions of the objects
5. Exit

Enter the operation to perform => 1

Enter the circle's information:
 Center, x y => 50 250
 Radius => 25

 A circle has been added to the list.
```

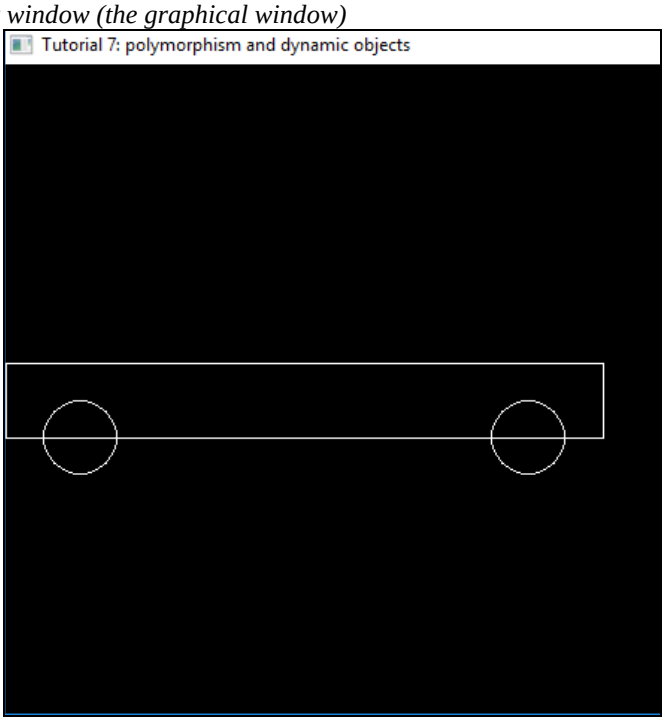
Interaction 3: the user chooses to enter another circle

```
========== Operations =========
1. Add a circle to the list
2. Add a rectangle to the list
3. Display the list
4. Move the positions of the objects
5. Exit

Enter the operation to perform => 1

Enter the circle's information:
 Center, x y => 350 250
 Radius => 25

 A circle has been added to the list.
```

Interaction 4: the user chooses to display all objects

```
========== Operations =========
1. Add a circle to the list
2. Add a rectangle to the list
3. Display the list
4. Move the positions of the objects
5. Exit

Enter the operation to perform => 3
```

The objects are drawn in another window (the graphical window)



Interaction 5: the user chooses to move the position of all objects

```
========== Operations =========
1. Add a circle to the list
```

2. Add a rectangle to the list
3. Display the list
4. Move the positions of the objects
5. Exit

Enter the operation to perform => **4**

Enter the distance dx and dy => **100 0**

The objects are drawn in another window (the graphical window)

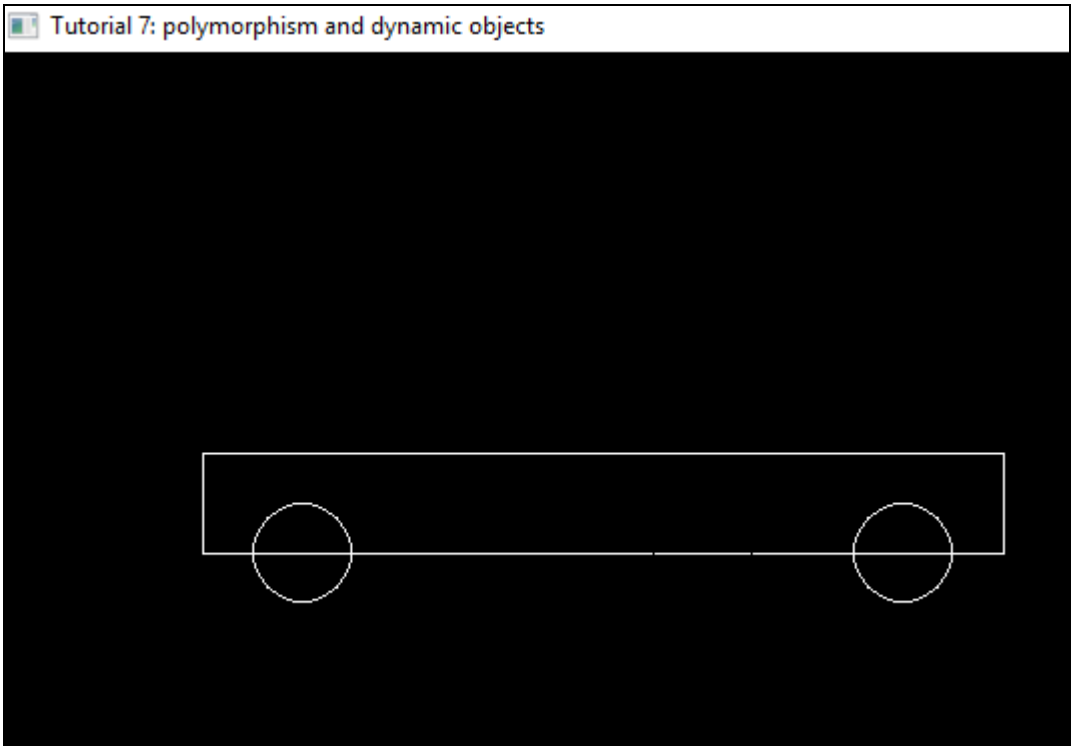


Figure 1: Expected result of the program