

**MA1008 - INTRODUCTION TO COMPUTATIONAL THINKING**

**MINI PROJECT: WORKING WITH POLYGONS**

**TUT GROUP: MA17**

**INSTRUCTOR: DR MOHAMED ARIF BIN MOHAMED**

**DATE OF SUBMISSION: 15 APRIL 2022**

**SEAH YONG LE STANLEY (U2121954F)**

Contents

[**INPUT** 3](#_Toc100941872)

[**POLYGON MANIPULATION** 4](#_Toc100941873)

[**POLYGON ANALYSIS** 6](#_Toc100941874)

[**STRENGTHS OF PROJECT** 7](#_Toc100941875)

[**LIMITATIONS OF PROJECT** 7](#_Toc100941876)

[**REFERENCES** 8](#_Toc100941877)

# **INPUT**

1. **All the documents attached, especially the miniproject.py, inputPoints.py and values.txt need to be in the same folder.**
2. To open the program, open “Miniproject.py”. Upon running the code, the terminal will ask: *“Would you like to input the coordinates manually (Y/N)?”*
   1. If user keys in “Y” or “y”, it goes to manual input where user needs to key in coordinates
      1. Question will ask: *"Please input the coordinates in the format eg. (x,y) and when done, type 'Done':* *"*
      2. User has to input coordinates in the form of x,y (eg. 100,200). If user does not input proper coordinates as mentioned, terminal will reply: *"Please input a proper coordinate"*, and prompt for another coordinate. Else, terminal will keep asking for an additional coordinate until a prompt is given.
      3. Once user is done with inputting all the coordinates, user has to type in *“Done”, “DONE” or “done”* to break the while loop
   2. If user keys in “N” or “n”, turtle screen opens up with a dot grid on the screen, where user can use mouse clicks on the screen to form the polygon
      1. User can keep clicking the number of points on the screen
      2. Once done, user can press space bar on the keyboard to end the polygon. A line will be drawn from the last coordinate to the first original coordinate.
   3. If user keys in “file” or “FILE” or “File”, user will be prompted to key in name of text file, such as polygon1.txt.

# **POLYGON MANIPULATION**

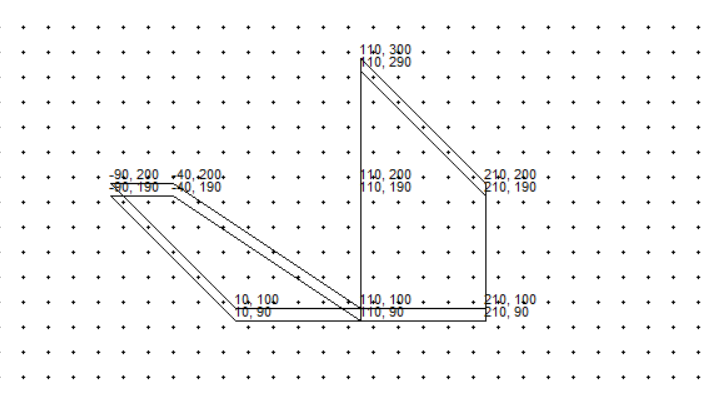
Below is the figure of the list of functions for polygon modification of polygons (graphic user interface at the right side of the screen):

1. Shift Right by 10 units
2. Shift Left by 10 units
3. Shift Up by 10 units
4. Shift Down by 10 units
5. Rotation by 45˚ about origin
6. Scaling up by 2 times its size
7. Scaling down by ½ its size

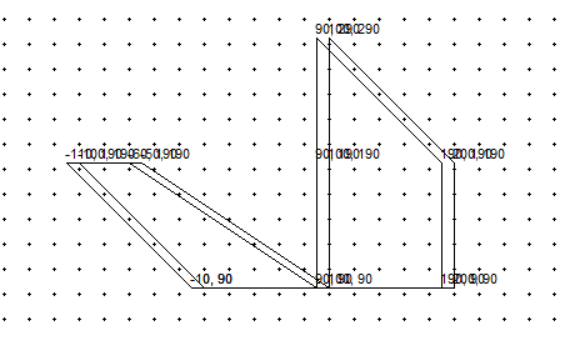
All the functions have a choice of duplicating it while shifting or not.

To access the functions, user just has to click the box in which the function is at to call out the function. Changes to the screen will occur.

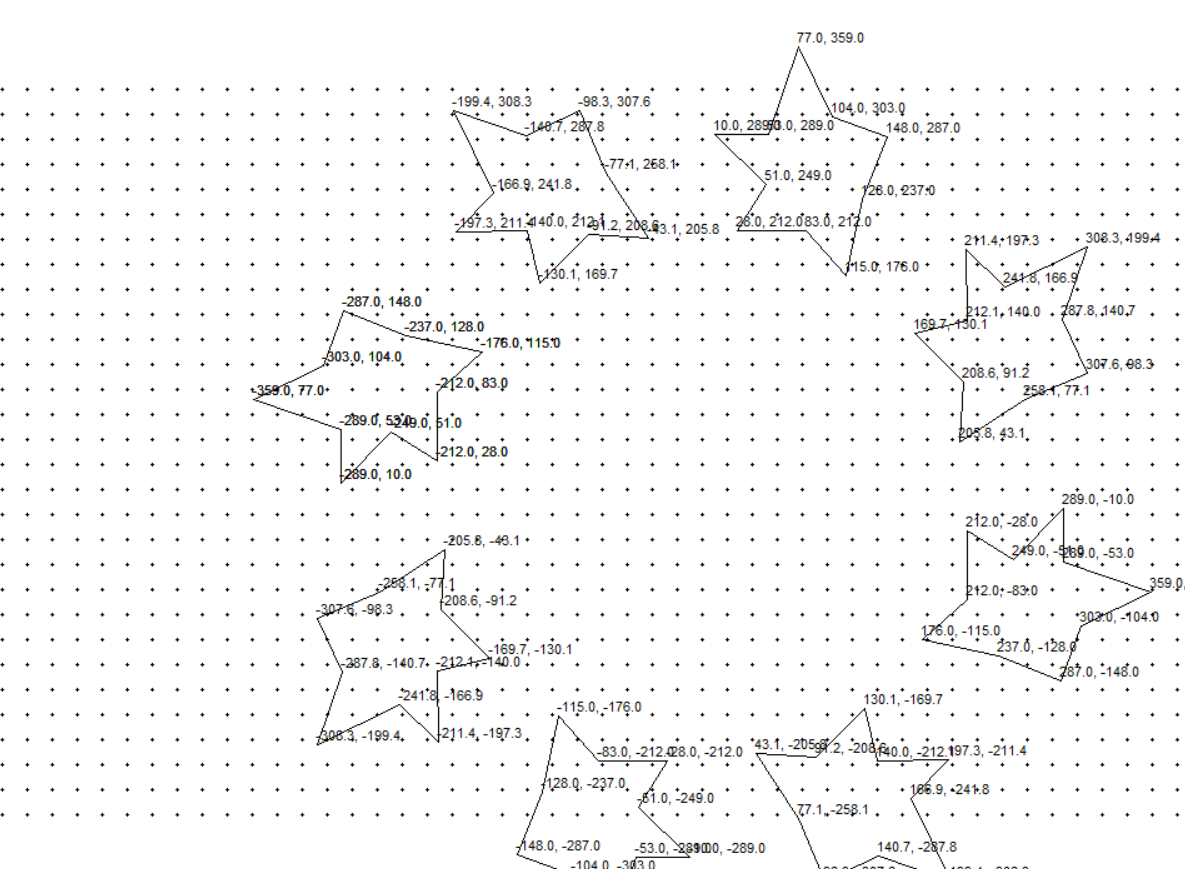
**Shifting down with duplicates:**



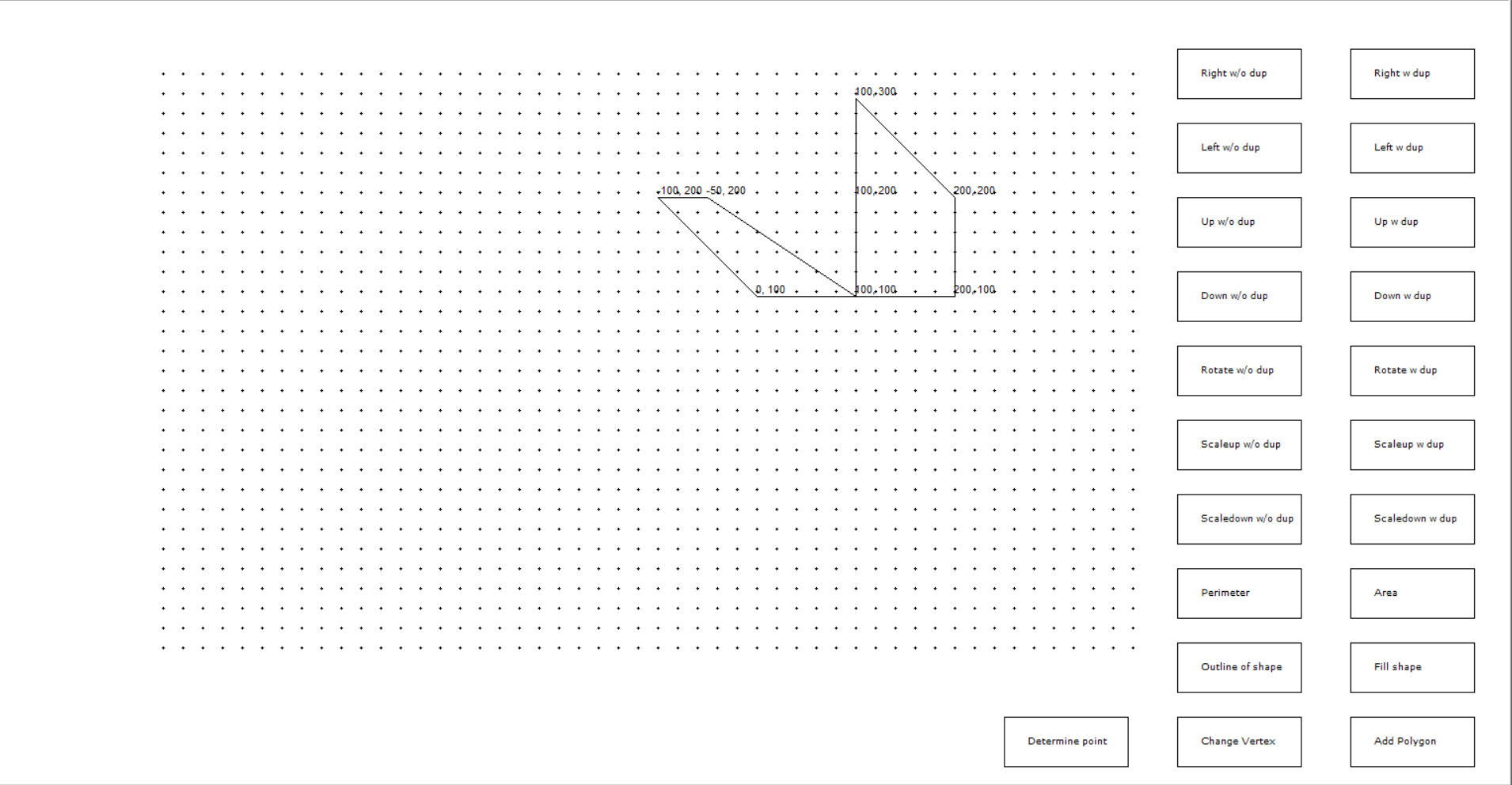
**Shifting Left with duplicates**

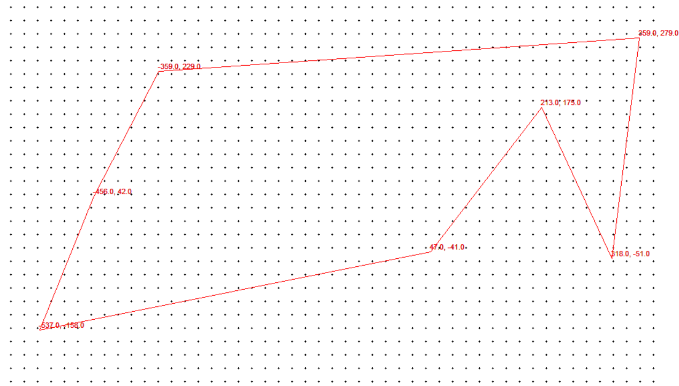


**Rotation with Duplicates**

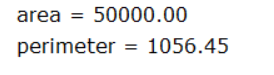


# **POLYGON ANALYSIS**

Below are the functions for polygon analysis:

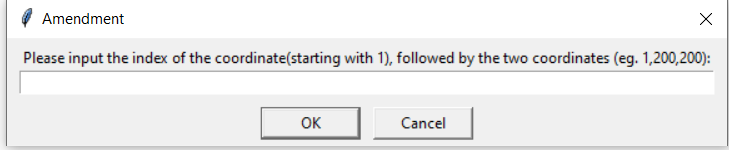
1. Perimeter
2. Area
3. Changing colour of the outline of the polygon
4. Filling in the polygon
5. Changing vertex of the polygon
6. Adding another polygon
7. Determining if point is inside a polygon or not

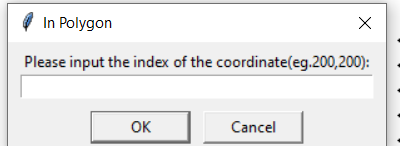
To access the functions, user just has to click the box in which the function is at to call out the function. Changes to the screen will occur.

For perimeter and area, the total calculation of perimeter and area will be written at the bottom left of the turtle screen in 2 decimal places.

Filling and changing colour of the outline can be accessed through the button. User can click the button multiple times to cycle colour between 4 colours (black, blue, green, red).

Adding another polygon allows user to create another polygon saved to a different list.

Amendment of the polygon works where user will be asked to type their amendment in the format as shown:

Similarly, to determine if a point is inside a polygon, another text input box will be shown for user to type in the coordinate

# **STRENGTHS OF PROJECT**

1. Allows flexibility in manipulation of the diagram such as being able to choose whether to duplicate the diagram or not, since drawing the background using a different turtle means that if user wants to remove all the duplicates, he does not need to reload the whole program but just click the function without the duplicates.
2. Relatively fool proof, user just needs to click the buttons on the screen
3. Try and excepts for user inputs that are not correct, so program will not terminate if there is incorrect input
4. User’s mouse inputs snap grid to the nearest 10 which makes it more accurate if user wants to click on the grid on the screen.

# **LIMITATIONS OF PROJECT**

1. Polygon is not limited within the boundary, meaning if the polygon were to shift towards the right across the grid, it will not be cut off but instead overlap with the GUI. However, the functions will still be able to run since program is working with coordinates of the vertices.
2. Drawing of the dot grid and GUI for the screen takes some time as Turtle needs to draw every single dot and move even though turtle speed is at the maximum, reducing efficiency of the program
3. Coordinates of duplicated polygons overlap each other, which may make reading the coordinates harder.

# **REFERENCES**

GeeksforGeeks. (n.d.). *Www.geeksforgeeks.org*. Geeks for Geeks. Retrieved April 15, 2022, from https://www.geeksforgeeks.org/how-to-check-if-a-given-point-lies-inside-a-polygon/-rightbar/