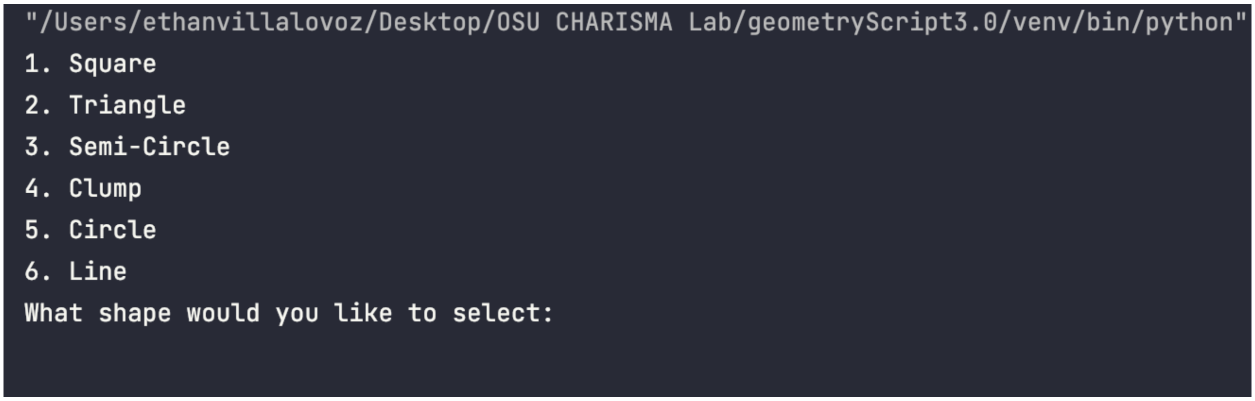
Documentation Tutorial

Hello, my name is Ethan Villalovoz and I am an REU student at Oregon State University: Robots in the Real World. I am a part of the CHARISMA Lab led by Dr. Heather Knight and under the mentorship of Alexandra Bacula. In this documentation, these are the steps of properly using the geometry_script python program. If there are any questions or inquiries about the program, please do not hesitate to contact me at my email: ethan.villalovoz@gmail.com. With that out of the way let's begin.

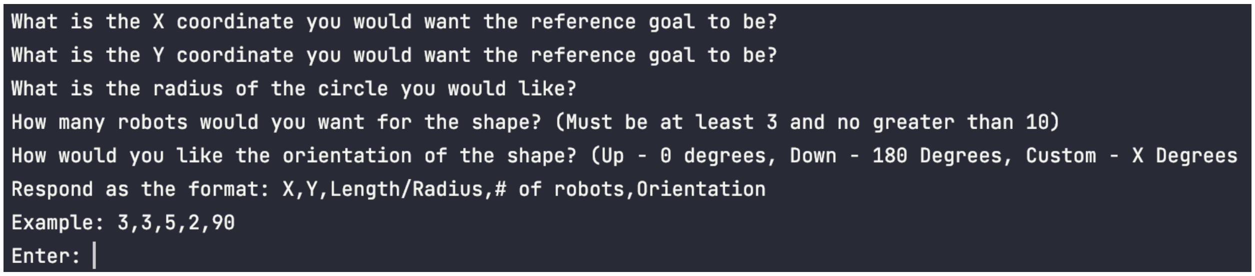
Once you start running the program, you will immediately be prompted to select from the various shapes.



```
"/Users/ethanvillalovoz/Desktop/OSU CHARISMA Lab/geometryScript3.0/venv/bin/python"
1. Square
2. Triangle
3. Semi-Circle
4. Clump
5. Circle
6. Line
What shape would you like to select:
```

To properly enter a valid input, you must type the name of the shape you want. You do not need to worry about capitalization of any of the letters; however, having a space at the end of input or not including the dash (-) for “semi-circle”, it will continue the program as if you chose line as your input option.

After you choose your wanted shape, you will see this on the console.



```
What is the X coordinate you would want the reference goal to be?
What is the Y coordinate you would want the reference goal to be?
What is the radius of the circle you would like?
How many robots would you want for the shape? (Must be at least 3 and no greater than 10)
How would you like the orientation of the shape? (Up - 0 degrees, Down - 180 Degrees, Custom - X Degrees
Respond as the format: X,Y,Length/Radius,# of robots,Orientation
Example: 3,3,5,2,90
Enter: |
```

You will be asked to enter information in this order: reference X goal, reference Y goal, radius/length, number of robots, and orientation of shape. Below I will describe each of the inputs:

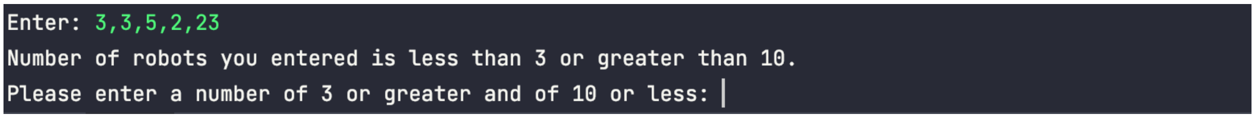
1. Reference X goal: The robot will be seen on a 2D plane. This is the “origin” which the robot will be placed upon based on the other inputs. This is specifically the X coordinate.
2. Reference Y goal: The robot will be seen on a 2D plane. This is the “origin” which the robot will be placed upon based on the other inputs. This is specifically the Y coordinate.
3. Radius/length: Self explanatory, just wants the size of the shape.
4. Number of robots: Self explanatory; however, must keep in mind that some shapes have different constraints for the minimum required robots.

5. Orientation of shape: Must enter in degrees, will rotate the shape based on that.

With knowing the information now, you must enter them as you see in the example above, you must have a comma in between for the program to parse through and store them into an array.

Ex: 333,34,23,345

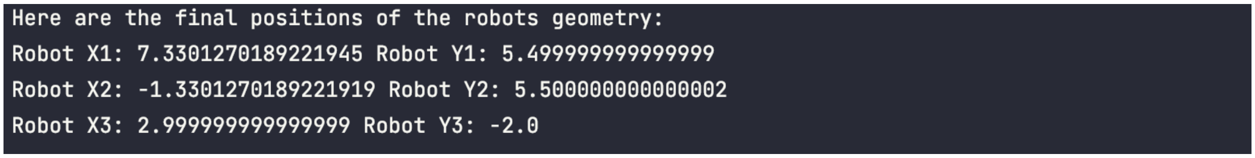
Do not have a space in between the commas. The only flag that will occur is if you enter the improper amount of robots. The maximum number of robots you can have is 10, but the minimum will vary.



```
Enter: 3,3,5,2,23
Number of robots you entered is less than 3 or greater than 10.
Please enter a number of 3 or greater and of 10 or less:
```

As you can see, when I selected a circle, but I entered 2 robots, the program caught me for it and will ask me for a valid number. Enter this and the program will continue.

After this what you will see is all of the final robot positions which they will be around the reference goal.



```
Here are the final positions of the robots geometry:
Robot X1: 7.3301270189221945 Robot Y1: 5.499999999999999
Robot X2: -1.3301270189221919 Robot Y2: 5.500000000000002
Robot X3: 2.999999999999999 Robot Y3: -2.0
```

End of program. If you see in the code any improvements please let me know and I will fix it!