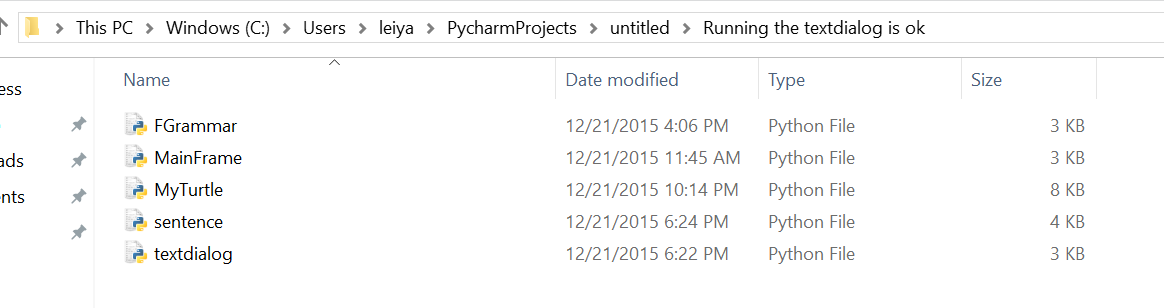
**Final project report –** **Basic graphics painter**

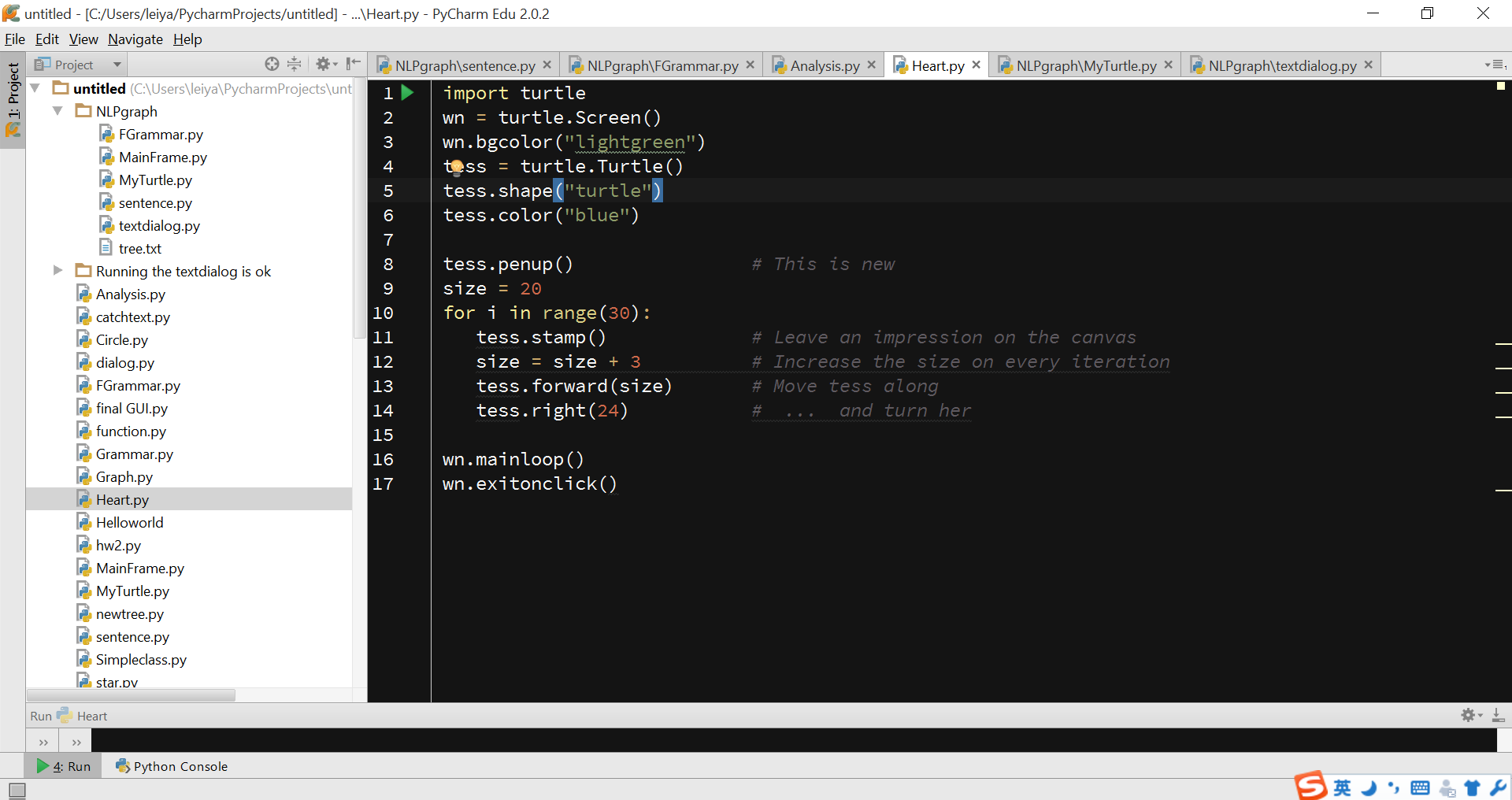
**Group member:**

Lei Yao

1. **Introduction about how to use my software:**



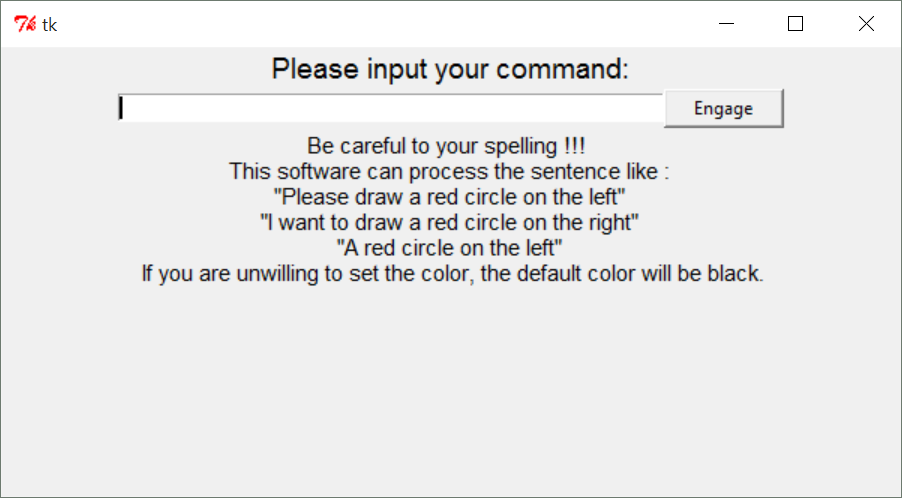
When open up my package, there are 5 files, import them in the python IDE named PyCharm (can be download on the URL: [**https://www.jetbrains.com/pycharm/download/#tabs\_1=windows**](https://www.jetbrains.com/pycharm/download/#tabs_1=windows)**)**



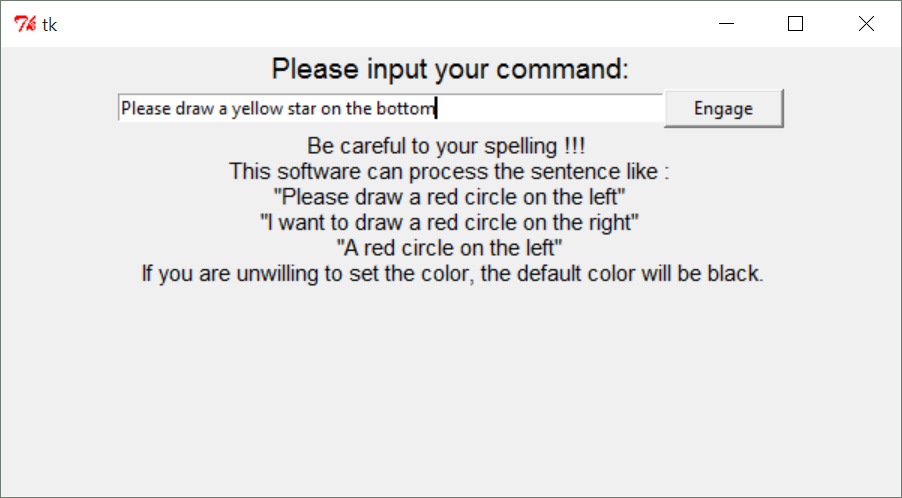
Then see the right top of the screen, all five files are there, and choose the file named “textdialog.py”, open it.



Click the green button to run the file “textdialog.py”. It can invoke the whole project.



This is the initial user interface, it give three example sentence structures to let the user input their command in a suitable way.



Users can imitate sentences given from the bottom of the UI and input them to the text box.

Graphs the software support this time are : circle, triangle, rectangle, square, star, OlympicSymbol, ladder, snowflake.

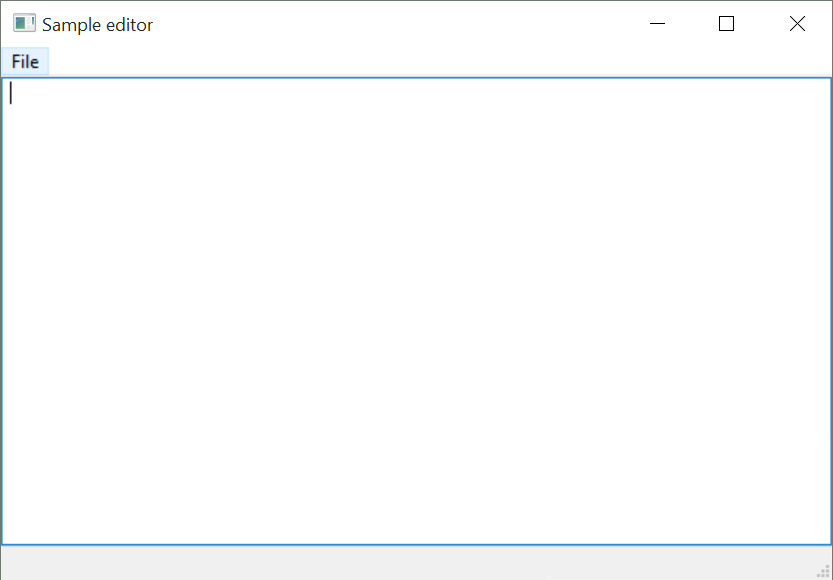
Colors the software support this time are : blue, green, red, yellow, black.

Direction the software support this time are: right, left, top, bottom, middle.

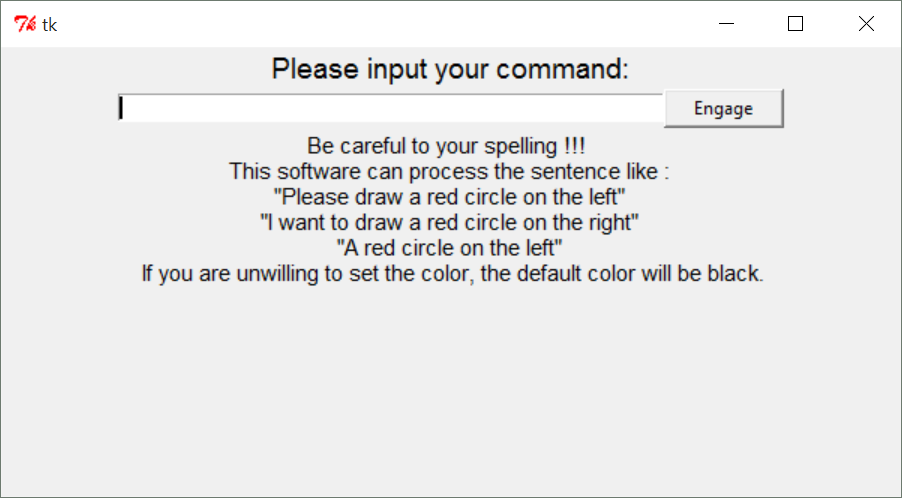
1. **Introduction about how I construct my software:**
2. **The user interface design:**

At the beginning, I prefer to import the package named ‘wx’ to construct a frame for this software, while I did not make the UI look good in that way.





So, finally, I use the package named Tkinter to make the user interface**.**



Here are some code about the User Interface:

def acceptInput(self):  
 r = self.frame  
 k = Label(r, text='Please input your command:', font=("Helvetica", 14))  
 k.pack(side='top')  
 w = Label(r, text='Be careful to your spelling !!! \n This software can process '  
 'the sentence like : \n''"Please draw a red circle on the '  
 'left"''\n''"I want to draw a red circle on the right"''\n'  
 '"A red circle on the left"''\n If you are unwilling to set '  
 'the color, the default color will be black.''',  
 font=("Helvetica", 11))  
 w.pack(side='bottom')  
 self.e = Entry(r, text='Name', width=60)  
 self.e.pack(side='left')  
 self.e.focus\_set()  
 b = Button(r, text='Engage', width=10, height=1, command=self.gettext)  
 b.pack(side='right')

1. **The data extraction:**

The most important things is how to let the software make sure about the sentences input by the user. From NLP class, we use python to construct grammar trees to analysis the sentence. Here is the grammar I construct for this program.

S -> IMPERATIVE | BACKGROUND | REQUEST | COMMAND  
IMPERATIVE -> NNP ACTION PP  
BACKGROUND -> ENTITY VBZ COLOR  
REQUEST -> HEAD ACTION PP  
COMMAND -> ENTITY PP  
NNP -> "Please" | "please"  
ACTION -> VB ENTITY  
VB -> "draw" | "want"  
ENTITY -> DT COLOR GRA | DT GRA  
DT -> "a" | "the" | "an" | "An" | "A" | "The"  
DIR -> "right" | "left" | "top" | "bottom" | "middle"  
GRA -> "circle" | "triangle" | "rectangle" | "square" | "star" | "OlympicSymbol" | "ladder" | "snowflake" | "smileface" | "dart"  
PP -> IN DIRECTION  
IN -> "on"  
DIRECTION -> DT DIR  
NN -> "background"  
VBZ -> "is"  
COLOR -> "blue" | "green" | "red" | "yellow" | "black" | "pink" | "skyblue"  
HEAD -> PRP VB TO  
ACTION -> VB ENTITY  
PP -> IN DIRECTION  
PRP -> "I"  
TO -> "to"

Here is when the user input the sentence “I want to draw a red circle on the left”, the grammar tree construct.

(S  
 (REQUEST  
 (HEAD (PRP I) (VB want) (TO to))  
 (ACTION (VB draw) (ENTITY (DT a) (COLOR red) (GRA circle)))  
 (PP (IN on) (DIRECTION (DT the) (DIR left)))))

Based on the tree, we can extract the useful keyword like “circle” “red” “left” to go to next step.

def FiltGra(self, x):  
 return x.label() == 'GRA'  
  
def FiltDir(self, x):  
 return x.label() == 'DIR'  
  
def FiltColor(self, x):  
 return x.label() == 'COLOR'

so, the software extract the word which has the label ‘GRA’ ‘DIR’ ‘COLOR’, and here is the results:

**circle left red**

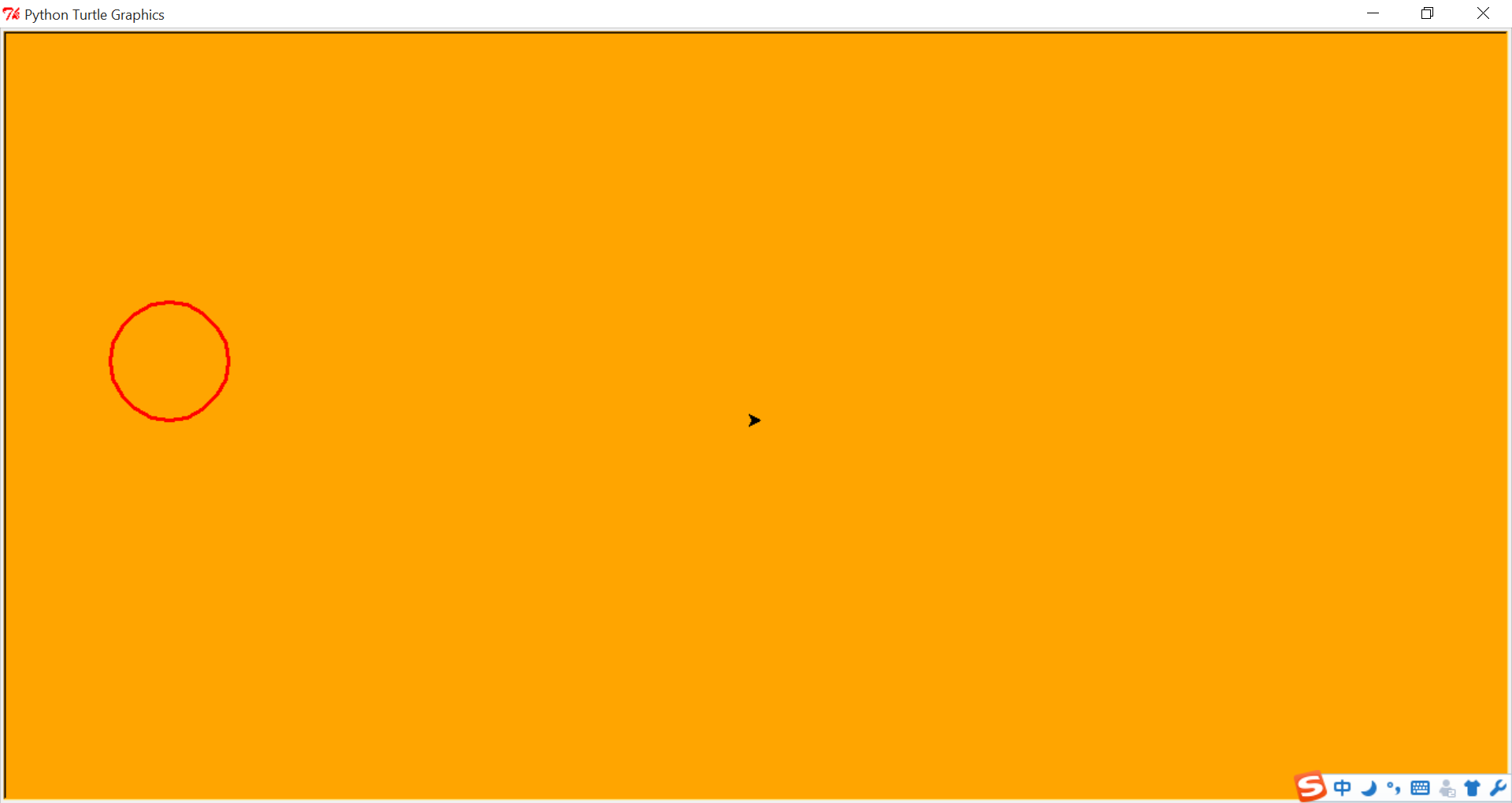
When finish this step, the keyword of the sentence have been extracted. And the software is going to next step.

1. **Drawing the graph:**

After the software get the keywords , it going to draw the graph. The example we discuss above is to draw a red circle on the left. So the software will running the code have relationship with keywords “circle” “red” “left”. Here are the code:

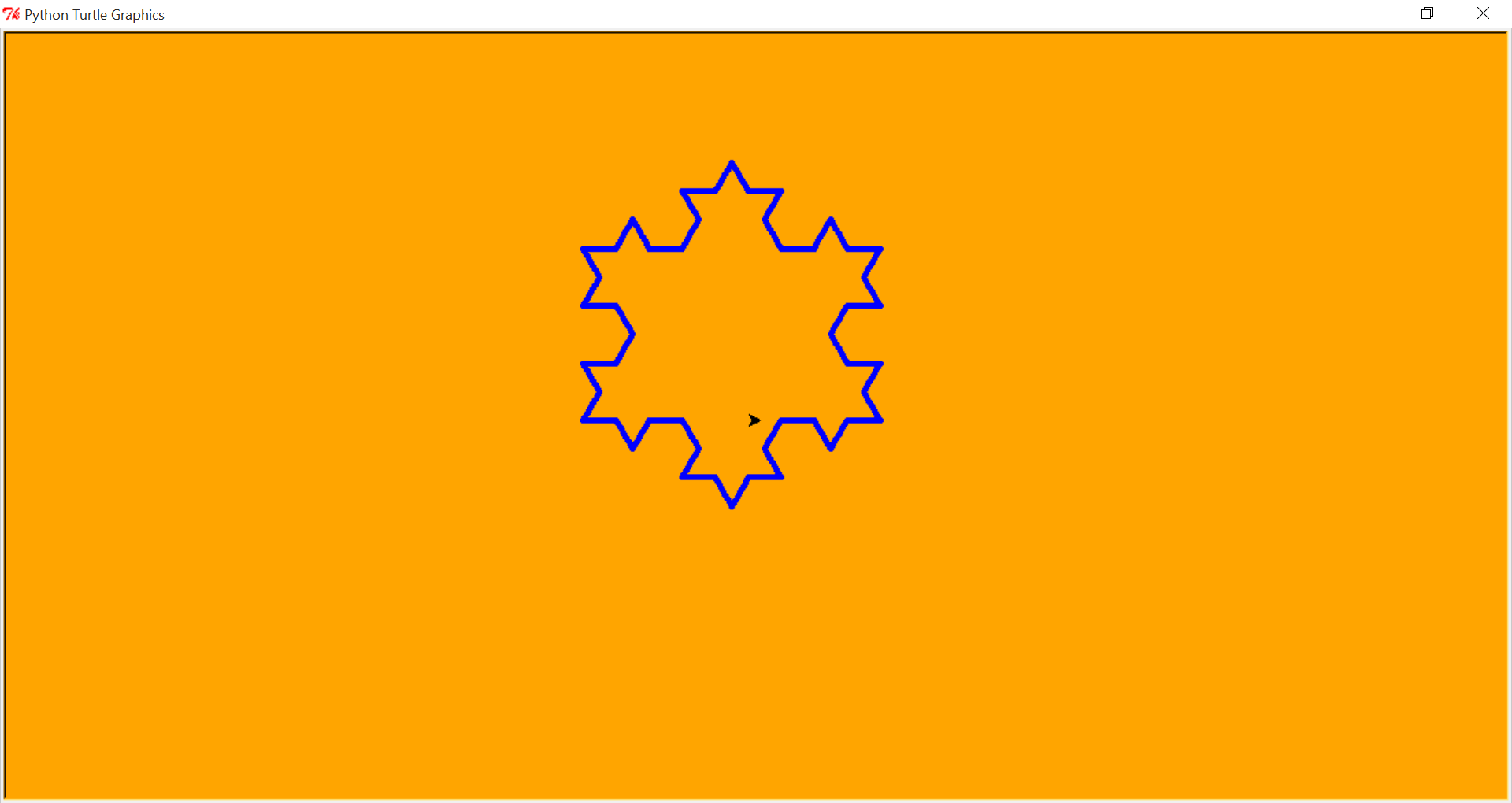
def drawCircle(self, x, y, color, radius=50):  
 *"""  
 Moves the turtle to the correct position and draws a circle  
 """* self.penup()  
 self.setposition(x, y)  
 self.width(2.5)  
 self.speed(10)  
 self.pendown()  
 *# self.begin\_fill()* self.color(color)  
 self.circle(radius)  
 self.hideturtle()  
 *# self.end\_fill()*

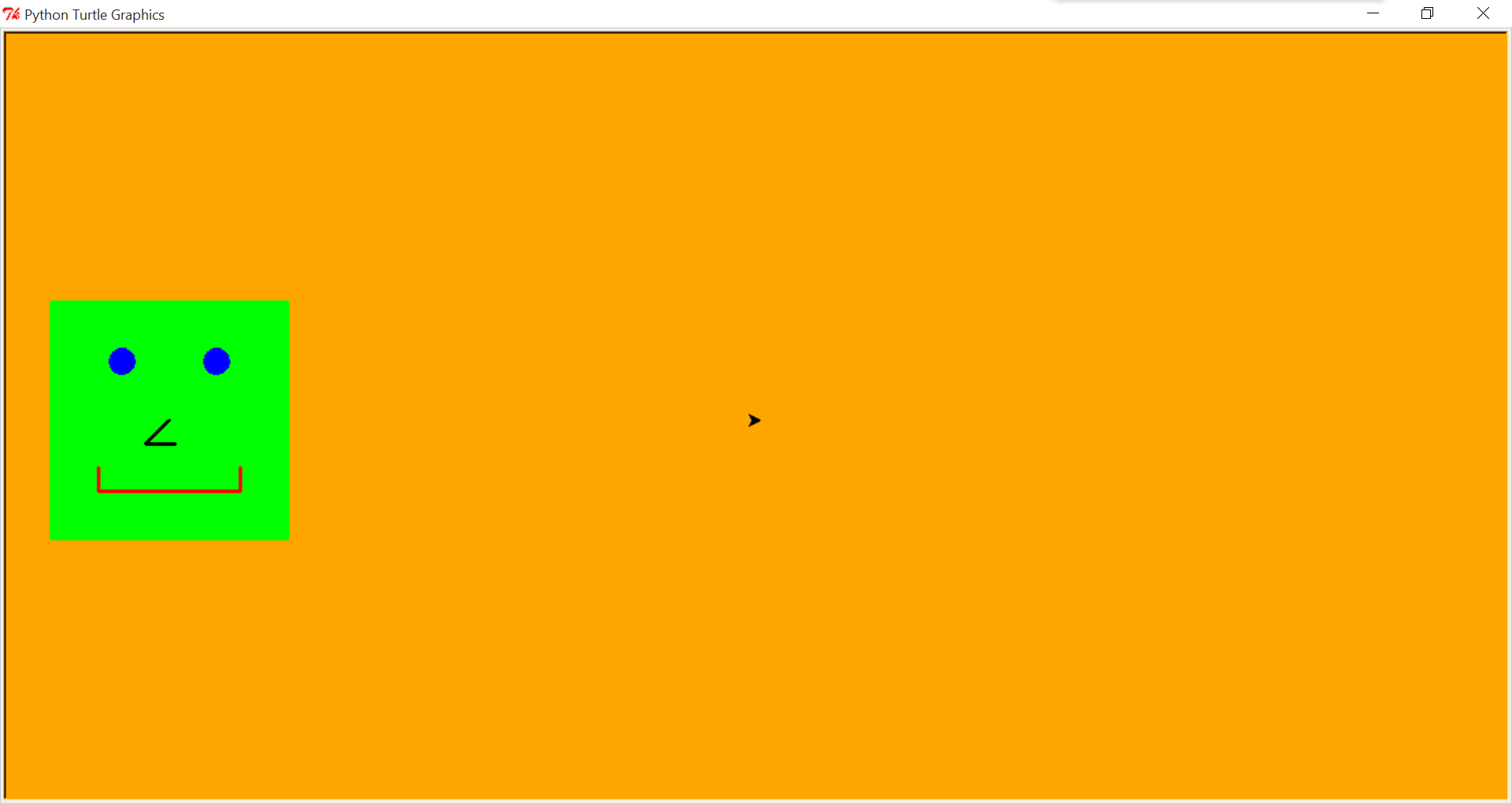
Here is the final output. The turtle draws a red circle on the left of the screen.



Besides, we can also draw some other graphs:







1. **Future work and reference:**

(1)For the future work, I hope this software can be use in the class for the teacher when they want to show some graph information to students. However, this need the software to be update many functions. For example, if the teacher can use the voice message to let the software to draw a suitable picture like the map, the function image and some other things. Not only save time in class, but also make the knowledge easily for students to comprehend.

Maybe it can also to be used in presentation, if the moderator want to show some graph data they can use this software to help audience to realize what the moderator want to express.

(2)Reference about this software:

* 1.NLTK Book:
* <http://www.nltk.org/book/>
* 2.Website:
* <http://openbookproject.net/thinkcs/python/english3e/hello_little_turtles.html>
* https://michael0x2a.com/blog/turtle-examples