CSE 231 Fall 2008

Programming Project #2

Assignment Overview

This project focuses on some mathematical geometry equations and using a simple graphics program. It is worth 20 points (2% of your overall grade). It is due Monday, September 8th before midnight.

The Problem

You will use Turtle graphics to draw 2 lines and geometry to calculate the angle between them. Turtle graphics (http://en.wikipedia.org/wiki/Turtle_graphics) is one of the oldest graphics programs, originally written as a part of the *logo* programming language. It is 2d graphics that uses a Cartesian coordinate system and a "turtle" that you can imagine has a pen attached to it's body that can move around the plane, drawing as it goes. Python has a module that implements the behavior of the original turtle graphics program, this module is simply called "turtle". Your program will read in 2 points from the user, and then use the methods in the turtle module to draw a line from 0,0 to the 1st point, and from the 1st point to the second point. You will also use geometry to calculate the acute (smaller) angle between the 2 lines you draw.

Program Specifications

Your program will function as follows:

1. output a brief, descriptive message when it first starts, indicating the purpose of the program and the user-required input and output that will be provided.
2. prompt the user for the x coordinate of the first point, then for the y coordinate. Repeat for the 2nd point.
3. print the acute angle between the 2 lines (1st line from 0,0 to 1st point, 2nd line from 1st point to 2nd point).
4. using the turtle module, draw the 2 lines on the screen. At the end of the 2nd line, write the angle previously calculated.

Deliverables

proj02.py -- your source code solution (remember to include your section, the date, project number and comments).

1. Please be sure to use the specified file name, i.e. "proj02.py"
2. Save a copy of your file in your CS account disk space (H drive on CS computers).
3. Electronically submit a copy of the file .

Assignment Notes:

Finding the angle between 2 lines:

You are required to calculate the acute angle between 2 lines. In order to do that you need to determine the slope of each line, and the slope of a line is "rise/run" or

$$\frac{y_2 - y_1}{x_2 - x_1}$$

where (x_1, y_1) and (x_2, y_2) are 2 points on the line. Then the formula to calculate the angle between 2 lines is:

$$\tan\phi = \frac{m_2 - m_1}{1 + m_1 m_2}$$

where m_1 and m_2 are the slopes of the 2 lines, and m_2 is **larger than** m_1.

This presents a new problem. We should always subtract the smaller of the two slopes from the larger. Another way to do this is to take the absolute value of any subtraction. The absolute value function is called `abs` in Python.
Thus `abs(-27)` yields `27` and `abs(45)` yields `45.`

We also still need to determine the actual angle Φ, our equation gives us the tangent of Φ. For this we will need to use the *atan* method which can be found in the math module. To use a method from a module you must import the module at the beginning of each program. The command to import a module is

```
import modulename
```

or in this case

```
import math
```

To use a method from the module you must preceed it with the module name. So to use the *atan* method from the math module the command is

```
math.atan(x)
```

This method returns the arctangent of **x**.

One last thing, the *atan* method returns the angle in **radians**, but we would like to output it in the more human friendly **degrees**. To convert from radians to degrees you use the following equation:

$$\text{(angle in degrees)} = \text{(angle in radians)}*180/\pi$$

π can also be found in the math module (`math.pi`)**.**

Using turtle graphics:
In order to use turtle graphics in python you must first import the turtle module, just like you did the math module. You can then use the help function in idle to find out what methods this module includes and what they do, just type “import turtle” in the idle command window, hit enter, and then type help(turtle) and scroll up through the list and information. For this project 3 methods will probably be the most useful, *goto*, *write*, and *done*.

The *goto* method moves the turtle to a specified point, drawing a line along the way. Note: The turtle always starts at the point (0,0). The *goto* method has the form

```
goto(x,y)
```

where x and y are the coordinates to which the turtle will move. Don't forget to use a method that is part of a module you must preceed it with the module name and a ".", so to use this method you must type turtle.goto(x,y).

The *write* method simply writes something to the screen at the point the turtle is currently at. It has the form

```
write(x)
```

where x can be a number or a string.

The *done* method tells turtle that is has finished drawing, **this is very important**, if you don't use the *done* method the program will keep waiting for input and you won't be able to do anything else in idle until you close the turtle window. Your last command therefore should be the done method. It has the form

```
done()
```

Getting Started

1. Using IDLE create a new program.
2. If you are in a CSE lab, select the H: drive as the location to store your file
3. Save the name of the project: proj02.py
4. Create the preface print information and prompt for the user input (4 separate prompts)
5. Run the program and fix any errors
6. Use the web site to hand in the program (incomplete as this point but you should continually hand things in)
7. Now calculate the slopes of the 2 lines, print them out just for now and make sure they are correct.
8. Now try to calculate the angle between your 2 lines. Test that your answer is correct by using an angle you know (like 90 degrees).
9. Finally use the turtle methods to draw the 2 lines and write the angle on the turtle screen.
10. Now you enter a cycle of edit-run to incrementally develop your program.
11. Hand in your final version.

Questions for you to consider (not hand in)

1. What happens if you use the same x value for the 1st and 2nd points? Why is this?
2. Try drawing a square using turtle, you could use the *goto* method but try using only the *forward* method and the *right* method (you can use help(turtle) in the idle command window to get more information on what these methods do, or just try them out and see!