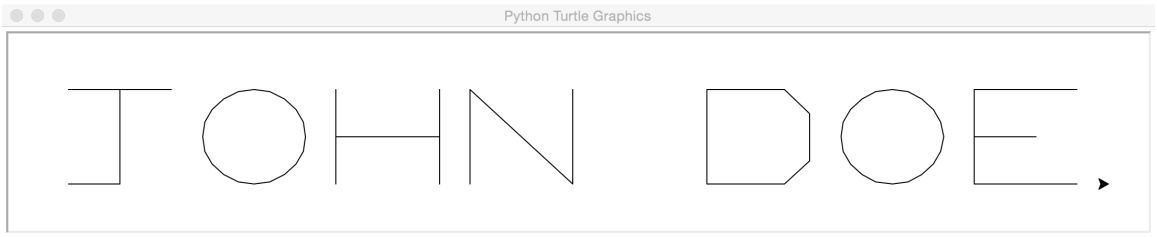
Computational Problem Solving CSCI-603
Write My Name With Turtle Lab 1

8/25/2017

1 Problem

Using Python's turtle graphics module, design the typography for, and implement a program that uses it to draw your family name, as shown below.



You may choose to follow the above design, or you may design your own font.

1.1 Problem-solving Session (20%)

You will work in a team of three or four students as determined by the instructor. Each team will work together to complete the following activities. Each activity is worth 3% (plus 5% for attendance and participation).

1. Choose a word that has at least 4 letters in it, and spells something that is longer than 4 characters. Draw on a piece of graph paper how the word will appear. Label enough points with coordinates so that the exact size of the letters and the spacing between them is clear.
2. Develop a design for writing messages where the turtle always starts at the same orientation and position relative to where the letter will appear. Write down a description for the initial spatial state of the turtle, and the location of the turtle when the drawing of the letter is finished, ready to draw the next letter.
3. Each teammate should write turtle code for a function that draws a single letter. The code that does the work should be the body of a Python function that follows your design decisions above. The function declaration/header should have real Python syntax.
4. Write a main function that uses the single letter functions and draws out the whole word.

At the end of problem-solving, put all team members' names on the sheet, number each item and hand in your work, one copy per team.

1.2 Implementation (80%)

Each student will *individually implement* and submit their own solution to the problem as a Python program named `myname.py`.

You will render your own family name, or a prefix or superset of it. The restriction is that the name string you choose must have the following properties:

- It uses at least 6 different characters (not including a space).
- It repeats at least 1 character.

If your original name uses non-Latin characters and can still follow the above properties, you are encouraged to program it that way!

1.2.1 Formula Calculations

If the letters you need include lines written on a diagonal, you will require mathematical functions, e.g. roots, trigonometry. Include "`import math`" at the top of your program file. To find the name of the math function you need, search the library documentation for the Python math module.

1.2.2 Program Operation

When run, the program should execute `turtle.mainloop()` after drawing the figure so that the user can see the drawing before it terminates.

1.2.3 Grading

The assignment grade is based on these factors:

- 20%: Attendance at problem-solving and results of problem-solving teamwork.
- 50%: The program draws an image containing your name. The rules about the name properties and the turtle behavior mentioned above is followed.
- 20%: The design promotes code reuse by defining and reusing functions.
  For example, the code should be commented so that it would be fairly easy for someone to reuse and extend the code to (a) spell a different word with the letters provided, and (b) add a function for a new letter.
- 10%: The code follows the style guidelines on the course web site.

Note: The submitted file must be a **Python 3 program**.

1.3 Submission

Transfer your program to the CS machines (using sftp or scp). If you have not already done so, you will need to register with the try system by running:

```
try grd-603 register /dev/null
```

Make sure you enter the correct section number so that you can be graded! Submit your program before the deadline using try:

```
try grd-603 lab1-1 myname.py
```