day29 turtle
Due:    6th period, Tuesday 10/19
        7th period, Wednesday 10/20

Today we're going to play with Turtle, a drawing library in Python.

Open IDLE on our new PCs.

Open a new document (File>New). Type in the following:

```
from turtle import *
circle(50)
```

Save your program to the folder you created on the Desktop with your name in it. Run it and it draws the following:

Add the following commands to your program (I put these into a text file attached to the assignment in the Google Classroom that you can copy from.)

```
forward(50)
right(90)
forward(50)
left(135)
forward(100)
```

Run it again, see that it draws the circle, then draws some straight lines. The above commands do just what you'd imagine: "forward" moves the turtle forward that many pixels, "right" turns 90 degrees towards the right and so on.
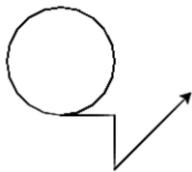
Replace your entire program with this program (copy and paste from Google Classroom text file):

```
from turtle import *
penup()
goto(-50,50)
pendown()
for x in range(24):
    forward(50+x)
    right(30)
    dot()
```

Run it. In this program we draw each line segment 4 pixels longer than the previous one using a variable and a for loop, to make the start of a cool spiral. The "dot()" command draws a dot where the turtle is once through each loop.

The turtle draws a line whenever it moves. You can control the line thickness, color, and direction using various turtle commands (see below). You can "lift the pen up" using the penup() command, then move the turtle without drawing, then put the pen back down and draw some more. The whole drawing board is an x-y coordinate plane (like you learned in math class) where ( 0, 0) is the center.

Replace your program with the following to draw something insane (copy and paste from Google Classroom text file):

```
from turtle import *
import random
speed(0)
for x in range(50):
   goto(random.randint(-300,300),random.randint(-200,200))
   dot()
   write(pos())
```

This program goes to 50 random locations and puts a dot at each one followed by the location of the dot. Add a penup() command and a pendown() command as follows to make it do the same thing without drawing lines between the dots (copy and paste from Google Classroom text file):

```
from turtle import *
import random
speed(0)
for x in range(50):
   penup()
   goto(random.randint(-300,300),random.randint(-200,200))
   pendown()
   dot()
   write(pos())
```

penup() and pendown() let you move the drawing pen around without drawing lines as you move.

There are tons of turtle commands, here are some I recommend you experiment with:

```
   forward(distance)
   backward(distance)
   right(degrees)
   left(degrees)
   dot(size)        you can make dots larger by including a number, or leave blank
   circle(r)        r is the radius of the circle
   penup()          lifts up the imaginary pen that the turtle draws with
   pendown()        puts the pen back down so you can draw
   home()           goes back to the center of the screen
   speed(n)         1 (slow) to 10 (faster), or 0, which makes turtle go the fastest
   pensize(n)       sets line thickness
   color("red")     or pink, or blue, yellow, etc.
   color("c1","c2") giving two colors makes one the line and one the fill
   begin_fill()     say begin_fill(), then do a bunch of stuff that includes overlapping lines,
                    then when you are done, do "end_fill()" to fill areas with color. See
                    example below.
   end_fill()
   goto(x,y)        Goes to the location (x,y) on the drawing board
   write("text")    writes text on the screen. You can set the font using a call like this:
                    write("Mr. Hays",font=("Times New Roman", 18, "normal"))
   bgcolor("color") Sets the whole window background color
```

Here's a fun example that uses begin_fill() and end_fill() (copy and paste from Google Class doc):

```
from turtle import *
color('red', 'yellow')
penup()
goto(-100,0)
pendown()
speed(0)
begin_fill()
for x in range(36):
   forward(200)
   left(170)
end_fill())
```

Here's a fun spiral program that uses the dot() command (copy and paste from Google Class):

```
from turtle import *
penup()
for x in range(50):
   dot(x)
   right(10)
   forward(5+x)
```

Try it. Try changing some of the numbers, see what happens.

Play around with the commands from the previous page. Try out lots of different things. Have some fun.

Today's actual assignment is on the next page.

```
from turtle import *
penup()
for x in range(50):
   dot(x)
```

# When you are ready, here is our assignment for today:

Create a program called "day29 turtle", put your name and the date in a comment at the top. Save it to your folder on the computer's Desktop.

Part 1: Use turtle to draw a plus sign like the image to the right with sides that are 50 in length. Set the pen width to be 4. Draw the plus sign using forward() and right() and left() commands. Write your name in the middle like I have done. Use a font size that is readable and fits nicely. Use penup() to move to an appropriate position then write your name so that it fits nicely in the plus sign. You'll just have to use trial and error to figure out where to use the write call.

Label your parts with comments in your program (#Part 1, #Part 2, etc.)

Part 2: In the same program go to (-50,100) and draw a plus sign again, this time with two different colors (one color for the lines and one for the fill). By the way, up towards the top of your entire program, add a "speed(0)" call to make all of your drawings go as fast as possible.

Part 3: In the same program go to (-150,0) or so and draw a shape of some kind that has fill, that uses dots or circles, and that uses a for loop. Use different colors than you used for the second plus sign. A sample thing is to the right. Yours should be something of your own creation. (You can take one of my sample programs and mess with it, but what you create should be your own.)