# Section 2 Programming Challenge Problems

We used a variable, `sides`, in the *ColorSpiral.py* program at the end of this chapter, but we didn't vary it much, or change its value except for editing the program, saving, and running again. Try changing the value of `sides` to another number, say 5. Save and run the program to see how this affects your drawing. Now try 4, 3, 2, and even 1! (Note: you won't be able to use bigger numbers unless you add more colors to the list of colors in the fourth line of the program.)

That's fun, but every time you want to change the number of sides, you have to edit the program, save it, and run it. What if you wanted to let a user decide the number of sides when the program's running, without having to save and rerun the program? Using a bit of skill from Section 1, you can ask the user for a number of sides and store that input in the variable `sides`. Our only extra step is to *evaluate* the number the user entered from the keyboard. We can find out what number the user typed with the `eval()` function, like this:

```
sides=eval(input('Enter a number of sides between 2 and 6: '))
```

Replace the line `sides=6` in *ColorSpiral.py* with the preceding line. Your new program will ask how many sides the user wants to see. Then, the program will draw the shape the user asks for. Give it a try!

As an additional challenge, you can change the *ColorSpiral.py* program into a more tangled and abstract shape just by adding an extra turn inside the end of the drawing loop. Add a line like `t.left(90)` to the bottom of the `for` loop to make the angles sharper (remember to indent, or space over, to keep the statement *in the loop*). The result, shown below, looks like a geometric toy or perhaps a ball made of colored rubber bands.
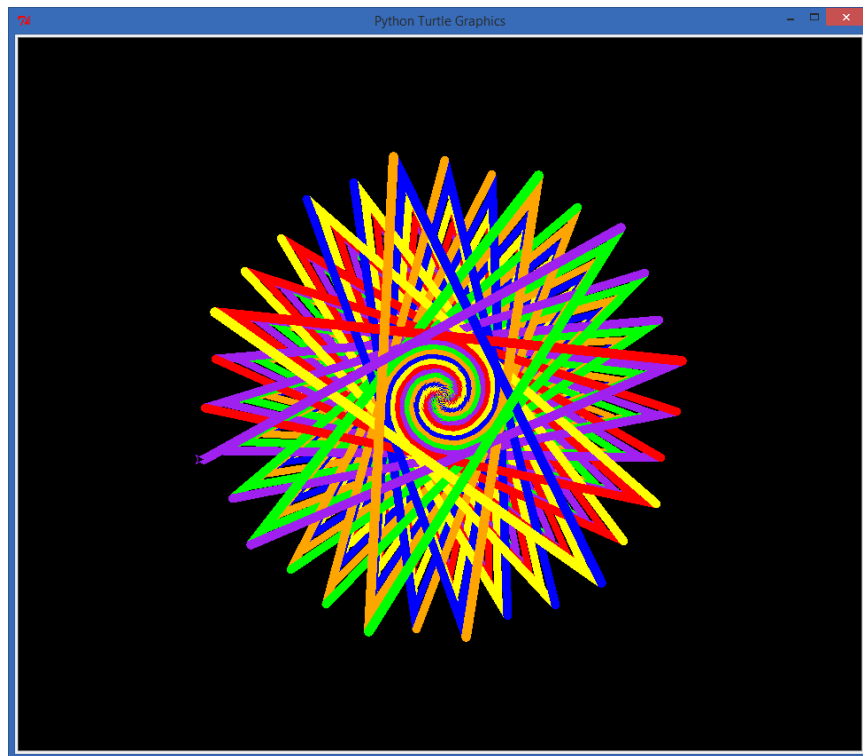
Adding an extra 90 degrees to each turn in ColorSpiral.py turns it into RubberBandBall.py.

Save this new version of the program as *RubberBandBall.py* ☺!