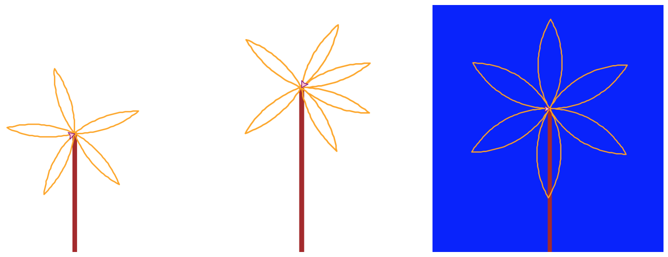
CSCI 2202 Computer Modelling for Scientists Due:2 Feb 2021
Lab 5: Functions

This lab explores modular program development using functions (See: Lec. 3B). You will write simple functions to draw line segments and arcs and then use these functions to draw a flower with a stem. See examples below.

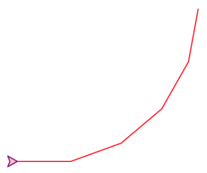
- Any effort is fine - you will have a chance to improve on your art another time.
- The purpose is to build a little drawing library and to have a bit of fun doing it.
- Make sure to test each function before proceeding.



(a) 5 Petals. (b) Lopsided (c) A good effort.

Figure 1: flowers.

Ex. 1 Write a function `polyLine(...)` that draws n line segments, each of the same fixed length, each at an angle θ to the previous. The function should accept the following parameters: a turtle t, the number of segments n, the length of each segment *length*, and the angle each segment is tilted to the previous one *angle*. If you are unsure how to pass values into a function, see the example programs in **lecture 3A**. The output for 5 segments, each of length 50, each tilted at 20^o to the previous: Test your function before proceeding.



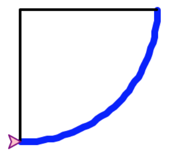
It is good practice to put a description of the function just after the name. *e.g.*

```
def polyLine(t, n, length, angle):
'''Draws 'n' line segments of 'length'; tilted at 'angle' to the prev. "'
```

note the triple quotes - they allows for multi-line comments. Triple quoted strings at the tops of functions are known as docstrings.

Ex. 2. Use `polyLine(...)` to a construct a function `polygon(t, numSides)`.

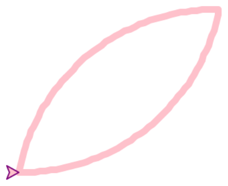
Ex 3 Write a function `drawArc` to draw an arc of circle of radius r, subtended by an angle α. The example below shows the arc of a circle of radius 100 in blue subtended by $\alpha = 90^o$



You must use `polyLine(...)` to write `drawArc`.
The arc length $s = r \cdot \alpha$ where α is in radians.
Break up the arc into (an integer) number of st. line segments n of a small fixed length ($\sim$ 5 pixels). Therefore each line segment of the arc will be of `length = s/n`
Call `polyLine(...)` with the above parameters to draw the arc

Ex. 4 Use `drawArc(...)` to define a function `drawCircle(t, r)` (r is the radius of the circle).

Ex. 5 Define a function `drawPetal(...)`. A petal is drawn by making two calls to `drawArc(...)`, **appropriately flipping the turtle heading between the calls.**



Ex. 6 Define `drawStem(t, length, heading)`. Which draws a stem, a straight line, coloured brown.

Ex. 7 Finally, define `drawFlower(t, r, angle, numPetals, p)` which makes `numPetal` calls to `drawPetal(...)`, each turned an angle `p` away from the previous).

Ex.8 Finally, using the function above, draw a flower. Your result may look like one of the flowers shown at the beginning of the lab. The individual parameter settings will result in different flowers.
Bonus for very nicely drawn flowers! Esp. those that incorporate more colours, new elements *etc.*.

- Zip into a folder and submit!