Extra Conditional Practice

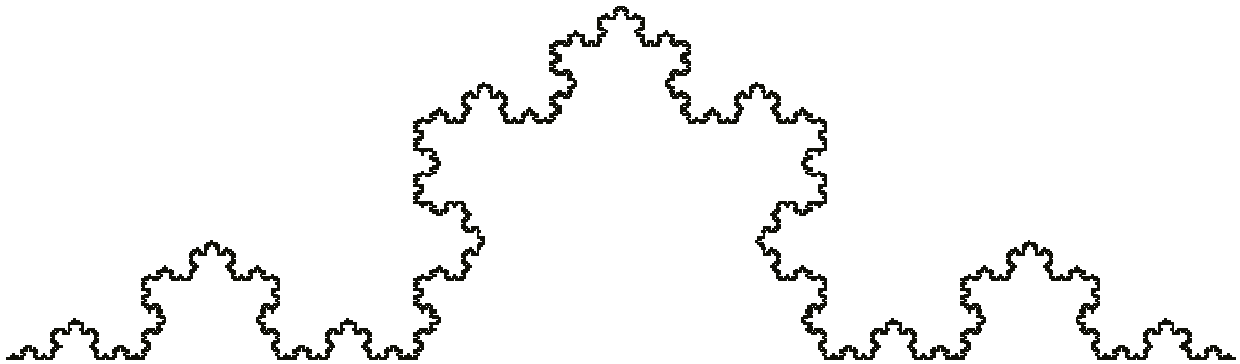
1. Write a function `can_add` that takes three integers as parameters and prints out “Yes” if it is possible to add two of the numbers to get the third and “No” otherwise.

   ```
   can_add(1, 2, 3) → “Yes”
   can_add(3, 1, 2) → “Yes”
   can_add(3, 2, 2) → “No”
   ```

2. Write a function called `tri_type` that takes three integer parameters representing the lengths of the sides of a triangle and prints what type of triangle it is. The three types are equilateral, isosceles, and scalene. An equilateral triangle has all three sides the same length, an isosceles triangle has two sides the same length, and a scalene triangle has three sides of different lengths. Here are some example calls to `tri_type`:

   ```
   tri_type(5, 7, 7) → isosceles
   tri_type(6, 6, 6) → equilateral
   tri_type(5, 7, 8) → scalene
   tri_type(2, 18, 2) → isosceles
   ```

3. The Koch curve is a fractal that looks something like this:



To draw a Koch curve with length x, all you have to do is

1. Draw a Koch curve with length x/3.
2. Turn left 60 degrees.
3. Draw a Koch curve with length x/3.
4. Turn right 120 degrees.
5. Draw a Koch curve with length x/3.
6. Turn left 60 degrees.
7. Draw a Koch curve with length x/3.

The only exception is if x is less than 3. In that case, you can just draw a straight line with length x. NOTE: to draw a Koch curve, you call your function that draws a Koch curve…

1. Write a function called koch that takes a length as a parameter, and that uses the turtle to draw a Koch curve with the given length.

2. Write a function called snowflake that draws three Koch curves to make the outline of a snowflake.

3. The Koch curve can be generalized in several ways. See wikipedia.org/wiki/Koch_snowflake for examples and implement your favorite.