# Objective

You will use a recursive algorithm to generate a simple fractal tree. A tree is generated by simply drawing a line, then splitting the line (like a letter “Y”), and repeating the algorithm for each new line, but reducing the length. The base case is when you reach a minimum line length.

## Topics: recursion, fractal, binary

# Instructions

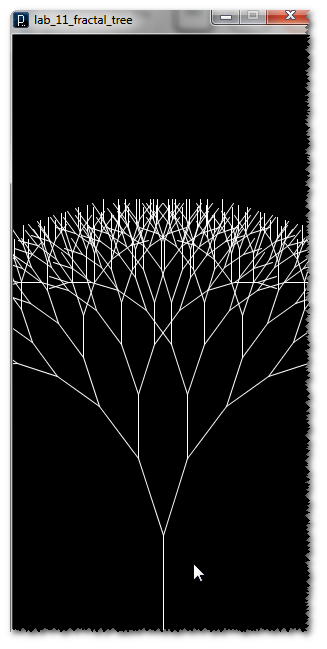
You will write a method with the following signature:

void **tree**(float x, float y, float len, float ang)

This method will draw a line, from the specified point (x, y), extending len distance at angle ang. Then, if the line length is greater than a specified minimum length (your *base case* for this recursive algorithm), call tree again. Use the endpoint of the line as the starting point of a new line, and vary the angle and line length.

# Hints

# Examples



You will need to use a little trigonometry to calculate the end points. Use:

# Challenge

Try splitting the line into more than two parts (be careful, the number of calculations grows exponentially). Alternately, tie the angle to random numbers, or mouse position. You can even try coloring the branches based on distance.