Notes from Quiz 4
- Declare variables once
- Once declared, reference by name
- Do not keep redeclaring them!
- Arrays need loops
- Array type has [] following the basic item type
  - int[], float[], Square[]
- Loop indices and array indices should be integers
  - nums[i] = 2*nums[i];
  - nums[i] = nums[2*i];
- Constructor has no return type
- Constructor should take parameters
- Function parameters and return types!

Odds and Ends
- Load an image ONCE
  - loadImage() in setup() (it's sloooooow)
  - use image() to render the image obj in draw()
- Constructor overload
  - We are going to call your constructor(s)!
- Transformations are drawing-related commands!
- Drawing does not depend on global variables
  - If your fish needs anything, it should be stored in the object

Review
- Recursion (recursive function)
  - a function that calls itself
  - base case
  - reduction of the work to a smaller instance
- Rotation ccw in Processing – negative angle

Creating a maze, recursively
1. Start with a rectangular region defined by its upper left and lower right corners
2. Divide the region at a random location through its more narrow dimension
3. Add an opening at a random location
4. Repeat on two rectangular subregions

Examples
- recursive sum
- recursive sum with array
- recursive findMax
Lindenmayer System
- A formal grammar developed to model the development of biological systems
- Generates strings that represent movements
- When traced in the plane, produce remarkable lifelike plant systems
- Components
  - An alphabet (a set of symbols)
  - An axiom or start string
  - A rule set that defines substitutions

L-system Example
- Alphabet: (A, B)
- Axiom: A
- Rules
  1. (A → AB)
  2. (B → A)
- Generation:
  1. A
  2. AB
  3. ABA
  4. ABAAB
  5. ABAABABA

Turtle Graphics
- Imaginary turtle with a pen
- Moves in the plane
  - Forward
  - Turn left
  - Turn right
- Traces with the pen as it moves
- Can put the pen up or down
  - Pen up: no trace
  - Pen down: trace

L-systems Example
- Alphabet: (F)
- Axiom: F
- Rules
  1. (F → F+F−F−F+F)
- Interpretations:
  1. F Forward (pen down)
  2. + Turn left (pen up)
  3. − Turn right (pen up)

3 and 5 Iterations

Koch Snowflake
- Alphabet: (F)
- Axiom: F++F++F
- Rules
  1. (F → F−F−F+F)
**Quadratic Flake**
- Alphabet: \( \{F\} \)
- Axiom: \( F+F+F+F \)
- Rules:
  1. \( F \rightarrow F+F-F-F+F+F+F-F \)

**Heighway Dragon (8 and 11 iterations)**

**Plants**
- Alphabet: \( \{F\} \)
- Axiom: \( F \)
- Rules:
  1. \( F \rightarrow F[-F][+F][F] \)
Recursive Subdivision
- Decide on a split ratio
- Find the sum of all n values
- Sort the values
- Select the first k values that sum up to the split ratio (\(\leq 0.6 \times \text{sum}\))
- Allocate these k values to the corresponding split and the remaining n-k to the other
- Recurse when any split contains more than one value

Obama speach