

Lab 5: Recursion

due October 19th/20th, 2017

This lab assignment asks you to trace, modify and write recursive functions.

Learning Objectives

- Trace recursive functions.
- Modify an existing program.
- Rewrite loops as recursive functions.
- Use assertions for testings.

Deliverables

- Your programs should start with a comment that includes your name, email, date, assignment description & collaboration statement.
- Bring a hardcopy of your programs (the source code, not the graphics) to your next lab period.
- Be prepared to run the Processing sketches and demonstrate your “[Theory of the Program.](#)”

WARM-UP: Trace the $\text{factorial}(5)$ ¹ function with a partner.

Add assertions to test the function, and see if there are any bugs, if so, fix them.

$$4! = 4 \times 3 \times 2 \times 1 = 4 \times 3! = 24$$

$$3! = 3 \times 2 \times 1 = 6$$

$$0! = 1$$

$$n! = n \times (n-1)!$$

```
int factorial(int n) {
  if (n == 1) {
    return 1;
  } else {
    return n + factorial(n-1);
  }
}
```

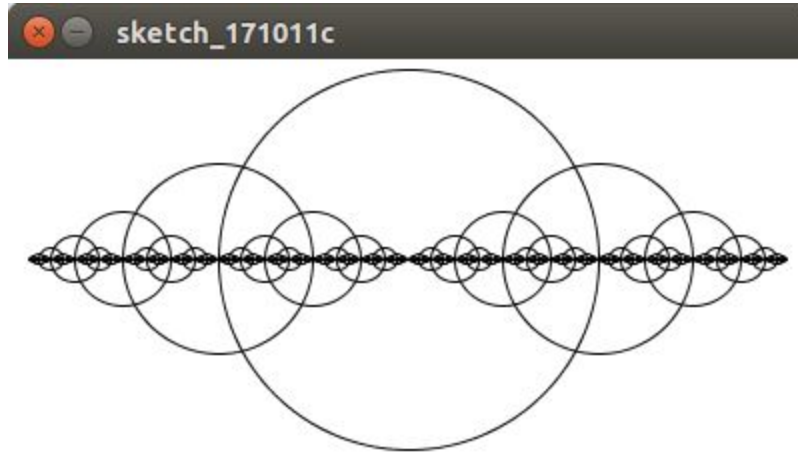
Sketch 1: Rewrite the drawCircle function using recursion rather than a loop.

```
void drawCircle(float x, float y, float radius) {
  while(radius > 2) {
    ellipse(x, y, radius, radius);
    radius *= 0.75f;
  }
}
```



¹ <https://en.wikipedia.org/wiki/Factorial>

Sketch 2: Write a `drawCircles` function, similar to `drawCircle` from earlier, but it draws circles recursively to the left and right, in addition to the central circle.



Sketch 3 (CHALLENGE): Fill in the blanks to create the recursive ruler below.

```
void setup(){
  size(400, 200);
  background(255);
  stroke(0);
  rectMode(CENTER);
  fill(255,196,0);
  stroke(0);
  strokeWeight(2);
  rect(width/2, height/2,
        width-10, height*0.8);
  ruler(width/2, height/2,
        width-10, height*0.8);
}

void ruler(float x, float y,
           float w, float h){
  line (_____,_____,_____,_____);

  if (w > _____){
    ruler(_____,_____,_____,_____);
    ruler(_____,_____,_____,_____);
  }
}
```

