

CPSC 231 Tutorial #21

michael-hung.ca/teaching

Reminders

TODAY

Quiz 11

THURSDAY

Quiz 11 Review

FRIDAY

Assignment 6 Individual Component

Recursion

A function or method that calls itself.

Recursion

A function or method that calls itself

BWAHAHAHAHAHAHAHAHAHAHAHA

Base Condition

When recursing, you always need a **base condition**.

This prevents *infinite* recursion.

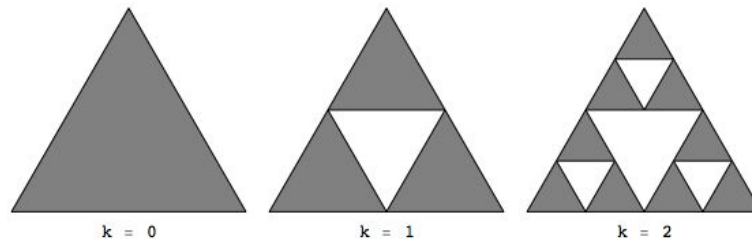
Base Condition

```
def recursive(arg1):  
    # if arg1 is blah  
    # do something to arg1  
    recursive(arg1)
```

Base Condition

```
def recursive(arg1):  
    # if arg1 is blah ← Base condition!  
    # do something to arg1  
    recursive(arg1)
```

Sierpinski Triangle



The Sierpinski Triangle is a fractal that demonstrates a three-way recursive algorithm. Draw a triangle, then divide it into four triangles by connecting the midpoints of each line. Repeat for all triangles *except the middle*.

INPUT

points	an array of the three vertices' coordinates
degree	a positive integer, how many times to recur

OUTPUT

Draw a Sierpinski Triangle... duh.

Assignment 6 Problem 1

```
def fill_brownian(a, i0, i1, variance, scale):  
    # Calculate midpoint index between a[i0] and a[i1]  
    # Add random number from normal distribution to midpoint value  
    # Set new variance  
    # Process array slices
```

Midpoint

Given points i_0 and i_1 , the midpoint is:

$$i_0 + i_1 // 2$$

Normal Distribution

```
sigma = random.gauss(mean, standard_deviation)
```

1. Normalize using the i_0 and i_1 of this segment, i.e. add σ to the average of $a[i_0]$ and $a[i_1]$

Setting the New Variance

Given the Hurst exponent, H , on the command line:

Divide the original variance by 2^{2H}

H should never change.

Process Both Array Slices

Call `fill_brownian` on the array slice:

1. before the midpoint (inclusive)
2. after the midpoint (exclusive)

Array slicing will come in handy here.