Functions as Object

```
def recurse_until(ending, the_function, n):
    yield n
    while not ending(n):
        n = the_function(n)
        yield n
```

- General-purpose high-order function for lots of fractal kinds of processing
- Requires two plug-in functions:
  - `ending` is a function to test to see if we're done, e.g.

```
lambda n: n==1
```

  - `the_function` is some form of the Syracuse function

Using the recurs_until() function

```
>>> recurse_until(lambda n: n==1, syracuse2, 9)
<generator object recurse_until at 0x1021278c0>
```

- What's that?
- That's how a lazy generator looks
  - It didn't return any values because we didn't demand any values