

Alternate Styles

```
def syracuse2(n):  
    return n//2 if n%2 == 0 else 3*n+1  
  
syracuse3 = lambda n: n//2 if n%2 == 0 else 3*n+1
```


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