Git, Looping and File Processing

June 04, 2018

Class Details

- Office Hours: Tuesdays at 3pm, NSH 186
- Today's material is at
  - `nd.edu/~brose3/2018reu-cmp/Day03.zip`

Goals

1. ~~Be able to use~~ ~~`git`~~ ~~'s~~ ~~`pull`~~~~,~~ ~~`add`~~~~,~~ ~~`commit`~~~~, and~~ ~~`push`~~ ~~commands~~

2. Be able to use `while` and `for` loops

3. Be able to import and manipulate text or data from a file

From last time...

- `if`, `elif`, `else` blocks
- What are lists, tuples, sets, and dictionaries?
    - How to use sets
    - How to call a dictionary by key
- How would you check if a variable is a list?
    - is a int, or float or string
- Imutable

~~Git~~

- `pull` from Classroom
- change `change_this_file.txt`
- `add` and `commit` changes
- `push` to your own remote repository

Loops

`while`

Basic structure:

```
while condition is True:
    do_something()
    update_condition()
```

- condition can be anything that returns a *boolean* (usually math-related)
- without update, infinite loop

`for`

Basic structure:

```
for item in container:
    do_something_with(item)
```

- container is an *iterable* (list, tuple, string, or *generator*)
- if you just want numbers, use `range`

Estimating π

Srinivasa Ramanujan discovered an infinite series for π:

$$\frac{1}{\pi} = \frac{2\sqrt{2}}{9801} \sum_{k=0}^{\infty} \frac{(4k)!(1103 + 26390k)}{(k!)^4 396^{4k}}.$$

File Input/Output (I/O)

We will only get to file input today.

String detour

- `s.strip()`
- `s.split()`
- `int(s)`
- `float(s)`
- `\n`

Filenames -- UNIX

- `.` is current directory or folder
- `..` is the directory above
- `/` to separate directories

Basic file reading

```python
f = open(filename, 'r')
f.read()
f.close()
```

Better file reading

```python
with open(filename, 'r') as f:
    f.read()
```

There is also a very useful `.readline()` method that reads one line at a time instead of the whole file at once.

"Best" file reading

```python
import csv

with open(filename, 'r') as f:
    reader = csv.reader(
        f, delimiter=',')
    next(reader)
```

Note: `numpy` and other packages will be even better, but lets walk before we run.

Practice Time!

- Practice with the handout

- Import `nothwind.txt` then separate by word
  - Try to count how many times each word appears

- Calculate the average sunspot form `sunspots.txt`
  - Can you count the days who's number of sunspots fell with an arbitrary range?

Readings – nothing new, just catch up if you need to.