## #TBT

- Finish some Python concepts and questions from earlier
  - Review for midterm exam
- Strategies for success in 101 assignments
  - Reading, writing, understanding, … success!
  - Knowing when to ask for help when you're feeling …



## Counting Questions

http://bit.ly/101fall15-0922-2



## Extreme Python, Extreme Values

- If I start reading a list of numbers …..
  - How do you remember the largest?
  - What do you think or do when I say "572" …
- Keep a variable storing extreme/max/min
  - Update when new/next value processed
  - What do you initialize max/min to?
  - What if you want the index as well as the value?



## Find largest value in a list of …

Max value: `[1,2,3]`, `["zebra", "armadillo"]`

- Does code below work for strings?

```
maxval = 0
for val in lst:       # type of data
   if val > maxavl:   # initial maxval?
        maxval = val
return maxval
```

What about using `max(lst)`, similar to `sum(lst)`

## Find largest value in a list of …

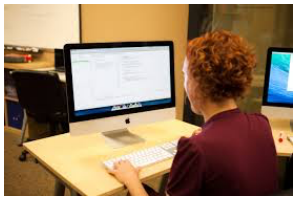
grades: `["owen:3.8", "bob:3.7", "susan:3.9"]`

```
mname = ""
mgpa = 0.0
for data in grades:                # type of data
    parts = data.split(':')        # type of parts
    name = parts[0]
    gpa = float(parts[1])          # type of gpa
    if gpa > mgpa:                 # initial mgpa?
        mgpa = gpa
        mname = name
return mname
```



## How to approach a 101 Assignment

- Programming compared to Cooking
  - Follow a recipe to create {food or masterpiece}?
  - Understand the whole project before coding
  - Know at least a few steps before coding





## What do we learn from assignment?

- We will snarf to get started
  - We will modify Pigify.py
  - We will create Caesar.py
- The grading tells us:
  - Caesar counts the same as Pigify
  - The chi-squared test will be difficult
  - The README will be worth more than normal



## What does HowTo say about Pigify.py

- Lots of details on how to pigify a word
  - Ignore at first, make the structure of the program work
- We have to write four functions
  - Details on function headers/prototypes given
  - Details on function functionality given
- Types and values in main program
  - Work to understand the flow
  - Run the program, where do you start?

## Making pigall work

- Make sure you understand this
  - What do you need to do so this works?
  - What is header, signature, prototype: `pigword`

```
def pigall(st):
    all = []
    for word in st.split():
        all.append(pigword(word))
    return ' '.join(all)
```



## Making pigword work

- Once you know what pigword does, how do you implement it?
  - Review rules for piglatin
  - Review code for APT you hopefully did ☺
- Don't try to make every case work at once!
  - Start small and grow a working program.
  - How about first word is a vowel to begin …
  - Then add another case, …



## If `pigword` is done … else …

- Get to `unpigall` and `unpigword`
  - Which will be easy? Why?
  - Can you do one easy case in `unpigword`?
- Why does it help to do one case at a time?
  - Builds confidence in reaching completion
  - Decreases time-to-completion: code works! Bugs easier to find.



## In class Questions

http://bit.ly/101fall15-0924-1

## Cracking the Caesar Cipher

- First create Caesar.py
  - Where do you start?
  - What's in the main program?
  - What's copied from Pigify.py
- What functions will you write first?
  - Where do you find this information?
  - What's not clear about it?



## Lots of details in making this work

- How do you loop over characters in word?
  - Is there anything familiar here?
- How do you know if a character is
  - Alphabetic?
  - Uppercase or lowercase?
  - A vowel or a consonant?
- Once again: start simple, make something work, add functionality incrementally



## How do you know encryption works?

- Is this a chicken and egg question?
  - Could you write decrypt first?
  - Isn't decrypting by eyeball decryption just encrypting 26 times?

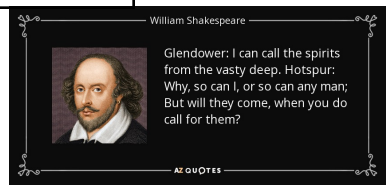
14 Pljbqfjbp fq'p bxpv ql zlrkq colj 1-10, yrq klq xitxvp
15 Qmkcrgkcq gr'q cyqw rm amslr dpmk 1-10, zsr lmr yjuywq
16 Rnldshldr hs'r dzrx sn bntms eqnl 1-10, ats mns zkvzxr
17 Sometimes it's easy to count from 1-10, but not always
18 Tpnfujnft ju't fbtz up dpvou gspn 1-10, cvu opu bmxbzt
19 Uqogvkogu kv'u gcua vq eqwpv htqo 1-10, dwv pqv cnycau



## Can you call a function 26 times?

- Encrypt using 26 shift keys and ... eyeball!

```
em = #encrypted message
for n in range(26):
    sem = encrypt(em,n)
    print n,sem
```



## What is chi-square about?

- If you expect [5, 9, 6, 11] then how close is?
  - [1, 9, 4, 8]
  - [4, 8, 9, 4]
  - [5, 5, 5, 5]
- What does $\Sigma (C_i - E_i)^2 / E_i$ mean?
  - $4^2/5 + 0^2/9 + 2^2/6 + 3^2/11 = 4.684$
  - $1^2/5 + 1^2/9 + 3^2/6 + 7^2/11 = 6.265$
  - $0^2/5 + 4^2/9 + 1^2/6 + 6^2/11 = 5.215$
- And the answer is ...