

## PFTW: Functions and Lists and Loops

- **In Python every function returns a value**
  - When return statement executes, function terminates
  - Return value replaces statement/expression for call
  - If no value explicitly returned None is returned
- **Properties of functions in Python**
  - Return value: int, float, string, list, file, boolean, ...
  - Functions always called with (..), arguments or none
  - Know built in/standard functions and methods
    - .split(), .strip(), .find(), .count(), len(..), max(..), sum(..), ...

## Piglatin, age-stay one-way

```
def convert(s):  
    if s[0] == 'q':  
        return s[2:]+"-quay"  
    if is_vowel(s[0]):  
        return s+"-way"  
    else:  
        return s
```

- **Why is this a reasonable place to start?**
  - Why not write all the code for all cases at once?

## Functions, lists, strings, memory

```
x = [1,2,3,4]  
y = x.append(5)  
# what are values of x and y? Why?  
  
x = [1,2,3,4]  
y = x  
x.append(5)  
print y[-1]  
s = "hello"  
s.replace('h','j') # return value ignored
```

## Good ideas that don't work

- **Find location of first 'a', first 'e', first 'o', ...**
  - Take the minimal of these, that's the first vowel
- **This might-could work, but it won't**
  - What if there are no a's? No vowels at all?
- **Is this how you do it? Often a bad or good idea**
  - Write code to do what you'd do, often a terrible idea
- **What do you have experience doing with loops?**
  - Examining each character in a string starting from first

## Piglatin, age-stay o-tway

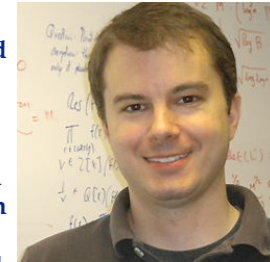
```
def convert(s):  
    if s[0] == 'q':  
        return s[2:]+"-quay"  
    if is_vowel(s[0]):  
        return s+"-way"  
  
    if is_vowel(s[1]):  
        return s[1:]+"-"+s[0]+"ay"  
    if is_vowel(s[2]):  
        return s[2:]+"-"+s[:2]+"ay"  
    if is_vowel(s[3]):  
        return s[3:]+"-"+s[:3]+"ay"  
    if is_vowel(s[4]):  
        return s[4:]+"-"+s[:4]+"ay"
```

Compsci 101, Fall 2012

9.5

## Craig Gentry

Duke '95, Harvard Law, Stanford  
Compsci PhD  
ACM 2010 Hopper Award for...



"Fully homomorphic encryption  
is a bit like enabling a layperson  
to perform flawless  
neurosurgery while blindfolded,  
and without later remembering the episode. We  
believe this breakthrough will enable businesses to  
make more informed decisions, based on more  
studied analysis, without compromising privacy."

IBM VP, Software Research

Compsci 101, Fall 2012

9.6

## Piglatin, age-stay ee-threay

```
def convert(s):  
    if s[0] == 'q':  
        return s[2:]+"-quay"  
    if is_vowel(s[0]):  
        return s + "-way"  
  
    for index in range(1,len(s)):  
        if is_vowel(s[index]):  
            return s[index:]+"-"+s[:index]+"ay"
```

- Generalize/parameterize by what varies
  - What does a loop do? it repeats! Is anything missing?

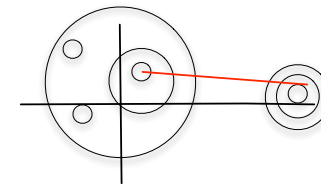
Compsci 101, Fall 2012

9.7

## Debugging APTs: Going green

- CirclesCountry APT: from ideas to code to green
  - How do we solve the problem? May not be apparent
  - How do we loop over circles? What is a circle?
    - When is a point inside a circle?

```
x = leastBorder([-3,2,2,0,-4,12,12,12],  
                [-1,2,3,1,5,1,1,1], [1,3,1,7,1,1,2,3], 2,3,13,2)
```



Compsci 101, Fall 2012

9.8

## Loops: definite and otherwise

- Range is useful for indexing over collections
  - Why is `range(len(s))` acceptable?
  - What does `range(2,10,4)` do?
- Accumulation idiom very important
  - Sum numbers, concatenate strings, append to lists
  - Initialize, accumulate (selectively), return
- Similar to min/max idiom
  - Find maximal value satisfying ..., e.g., max GPA

## Loops, Comprehensions, Idioms

Max GPA: ["owen:3.8", "bob:3.7", "brook:3.9"]

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

## Test Practice: finding perfect numbers

- 6, 28, 496 are perfect:  $28 = 1 + 2 + 4 + 7 + 14$ 
  - Add up divisors
  - if `n % d == 0`: #what does this mean?

```
s = 0
for x in range(1,n):
    if n % x == 0: s = s + x
if s == n: return True
```

- Alternative:

```
total = sum([x for x in range(1,n) if n % x == 0])
return total == n
```

## Find all links in a web page

- advantages: printing v returning a list
  - What is signature (aka header) for function
  - What will function return
  - What is parameter
- How do we start writing code, helper functions?
  - As with APTs, avoid writing too much code, call it
    - We'll need to write it later, but concentrate on one task ...

