CIS 122

Homework 1 Review

Homework Review

- Almost all homework received
  - If you are planning on submitting late, let me know!
- Feedback tonight
  - Homework grades
- Feedback soon
  - Homework solutions

Homework Review - Part 1

```
def FtoC(Ftemp):
    """Converts temperature from Fahrenheit to Celsius"""
    Ctemp = (5.0/9.0) * (Ftemp - 32)
    return Ctemp

def CtoK(Ctemp):
    """Converts temperature from Celsius to Kelvin"""
    Ktemp = Ctemp + 273
    return Ktemp
```

Homework Review - Part 1

```
def FtoK(Ftemp):
    """Converts temperature from Fahrenheit to Kelvin"""
    Ctemp = (5.0/9.0) * (Ftemp - 32)
    Ktemp = Ctemp + 273
    return Ktemp

def FtoK(Ftemp):
    """Converts temperature from Fahrenheit to Kelvin"""
    Ctemp = FtoC(Ftemp)
    Ktemp = CtoK(Ctemp)
    return Ktemp
```

Homework Review - Part 2

```
def myMax(a,b):
    """Return the larger of a and b"""
    if a > b:
        return a
    else:
        return b
```

Homework Review - Part 2

```
def myMax3(a,b,c):
    """Return the largest of a, b, and c"""
    if a > b:
        if a > c:
            return a
    elif b > a
        if b > c:
            return b
    else:
        return c
```

Homework Review - Part 2

```
def myMax3(a,b,c):
    """Return the largest of a, b, and c"""
    if a > b and a > c:
        return a
    elif b > a and b > c:
        return b
    else:
        return c
```

Homework Review - Part 2

```
def myMax3(a,b,c):
    """Return the largest of a, b, and c"""
    if a >= b and a >= c:
        return a
    elif b >= a and b >= c:
        return b
    else:
        return c
```

Homework Review - Part 2

```
def myMax3(a,b,c):
    """Return the largest of a, b, and c"""
    if a >= b and a >= c:
        return a
    elif b >= a and b >= c:
        return b
    else:
        return c

def myMax3(a,b,c):
    """Return the largest of a, b, and c"""
    d = myMax(a,b)
    e = myMax(c,d)
    return e
```

Homework Review - Part 2

```
def myMax3(a,b,c):
    """Return the largest of a, b, and c"""
    if a >= b and a >= c:
        return a
    elif b >= a and b >= c:
        return b
    else:
        return c

def myMax3(a,b,c):
    """Return the largest of a, b, and c"""
    return myMax(myMax(a,b), c)
```

Homework Review - Part 3

- Shifting letters
  - Convert letter to number
  - Add shiftNum to number
  - Convert back to letter

```
def shiftUpperChar(char, shiftNum):
    """Shifts char forward by shiftNum letters"""
    charNum = ord(char)
    shiftedNum = charNum + shiftNum
    shiftedChar = chr(shiftedNum)
    return shiftedChar
```

Homework Review - Part 3

- Shifting letters
  - Convert letter to number
  - Add shiftNum to number
  - If we've gone too far, shift back
  - Convert back to letter

```
def shiftUpperChar(char, shiftNum):
    """Shifts char forward by shiftNum letters"""
    charNum = ord(char)
    shiftedNum = charNum + shiftNum
    if shiftedNum > ord('Z'):
        shiftedNum = shiftedNum - 26
    shiftedChar = chr(shiftedNum)
    return shiftedChar
```

Homework Review - Part 3

- Shifting characters - 3 cases
  - Upper case letter - use shiftUpperCase
  - Lower case letter - use shiftLowerCase
  - Anything else - return original char

```
def shiftChar(char, shiftNum):
    """Shifts char forward by shiftNum letters"""
    if ord('a') <= ord(char) <= ord('z'):
        return shiftUpperChar(char)
    etc...
```

Homework Review - Part 3

- Shifting characters - 3 cases
  - Upper case letter - use shiftUpperCase
  - Lower case letter - use shiftLowerCase
  - Anything else - return original char

```
def shiftChar(char, shiftNum):
    """Shifts char forward by shiftNum letters"""
    if 'a' <= char <= 'z':
        return shiftUpperChar(char)
    etc...
```

Ironing out Bugs

- We all write buggy code
  - Easy to fix if we know what they are
  - How do we catch them?
- Testing code
  - I give you some test cases
  - Run known input through functions
  - Check output
- My test cases don't catch everything
  - Run tests of your own

Ironing out Bugs

- We can't check everything
  - That's a lot of input
  - Can we never be sure our code works?
- Technically no...
  - But we can be pretty sure
- Test smart
  - If one input works, similar inputs probably also work
  - Test different kinds of inputs
  - Try to cover all cases

Ironing out Bugs - myMax

- Three important cases
  - a > b
  - a < b
  - a == b
- Three test cases (at least)
  - myMax(1,2)
  - myMax(2,1)
  - myMax(3,3)

Ironing out Bugs - ShiftChar

- What cases should we test?
  - Come up with a test for each

Ironing Out Bugs

- Always check your edge cases
- If your function deals with numbers
  - Make sure it works for 0
- If your function deals with strings
  - Make sure it works on the empty string
- If your function works up to a specific bound
  - Check that bound

Ironing out Bugs

- I don't need to see your test cases
  - Don't need to print out results
  - Don't need to submit records
- If you've tested your code, it will be evident
  - Your code will work