

Lab 6 Feedback

- Wider range of scores
 - Difference is mostly who is asking questions
- Need Comments!
 - Harder problems
 - Code for solutions aren't obvious
 - Reminders for yourself when you're reviewing the solutions for exam
- Use constants as appropriate
 - Type of character in password
 - ASCII codes
- Taking off for inefficient solutions

Mar 4, 2008

Sprenkle - CS111

1

Generating a Random Password

```
NUM=0
LOWER=1
UPPER=2

password=""
len_password= randint(6,8)

for char_type in xrange(len_password):
    #determines if the character is a number, upper-case, or lower-case
    char_type= randint(0,2)
    #for each case, randomly assigns character from ASCII
    if char_type == NUM:
        number= randint(48,57)
        passwordch=chr(number)
    elif char_type == LOWER:
        letter= randint(97,122)
        passwordch=chr(letter)
    elif char_type == UPPER:
        uletter= randint(65,90)
        passwordch=chr(uletter)
    password+=passwordch
```

Define outside of for loop

Good variable names

Even better to use constants for ASCII values.

Mar 4, 2008

Sprenkle - CS111

2

Caesar Cipher

- Needs comments, explaining wrapping

```
HIGHEST_ASCII = 122
LOWEST_ASCII = 97
NUM_LETTERS = 26

coded_text = ""
for char in text:
    if char != ' ':
        ascii = ord(char) + key
        if ascii > HIGHEST_ASCII:
            new_char = chr(ascii - NUM_LETTERS)
        elif ascii < LOWEST_ASCII:
            new_char = chr(ascii + NUM_LETTERS)
        else:
            new_char = chr(ascii)
        else:
            new_char = char
        coded_text += new_char
```

No unexplained numbers in code.

Mar 4, 2008

Sprenkle - CS111

3

Function comments

- Wheel of Fortune

```
# displays a pretty, centered header
# for the Wheel of Fortune game
def printHeader():
    ...
```

Good. Describes function at high level

```
# defines the printHeader function
def printHeader():
    ...
```

Not descriptive. Says what *you're* doing, not what *function* does

Mar 4, 2008

Sprenkle - CS111

4

Speeding Tickets

- Problems understanding specification

```
# Given the clocked speed of the car and the posted speed limit
# (as numbers), returns the calculated numeric fine.
# Fine is 0 if the person was not speeding.
def calculateFine(clockspped, speedlimit):
    if clockspped <= speedlimit:
        return 0
    else:
        diff = clockspped - speedlimit
        fine = 50 + 5 * diff
        if clockspped > 90:
            fine += 200
        return fine
```

Comment says nothing about where parameters came from

Mar 4, 2008

Sprenkle - CS111

5

Speeding Tickets

- Problems understanding specification

➢ "driver" --> main()

```
def main():
    clockspped = input("Enter the clocked speed: ")
    speedlimit = input("Enter the speed limit: ")
    fine = calculateFine(clockspped, speedlimit)
    if fine > 0:
        print "Slow Down! You have been fined $" +str(fine) + "."
    else:
        print "Continue driving safely."

# ...
def calculateFine(clockspped, speedlimit):
    ...
main()
```

Mar 4, 2008

Sprenkle - CS111

6

Fibonacci Sequence

- Create a list of the 1st 15 Fibonacci numbers

➤ $F_0 = F_1 = 1; F_n = F_{n-1} + F_{n-2}$

```
fibs = []      # create an empty list
fibs.append(1) # append the first two Fib numbers
fibs.append(1)
for x in xrange(2,15): # compute the next 13 nums
    newfib = fibs[x-1]+fibs[x-2]
    fibs.append(newfib)
```

```
print fibs      # print out the list
```

Mar 4, 2008

Sprenkle - CS111

fibs.py

7

Fibonacci Sequence

- Create a list of the 1st 15 Fibonacci numbers

➤ $F_0 = F_1 = 1; F_n = F_{n-1} + F_{n-2}$

Similar to xrange,
Call similarly

```
fibs = range(15) # creates a list of size 15,
                  # containing nums 0 to 14
```

```
fibs[0] = 1
fibs[1] = 1
```

```
for x in xrange(2,15):
    newfib = fibs[x-1]+fibs[x-2]
    fibs[x] = newfib
```

```
for num in fibs: # print each num on sep line
    print num
```

Mar 4, 2008

Sprenkle - CS111

fibs2.py

8

Lab 7 Overview

- Focus: program organization
 - Defining and Using Functions
 - Will need to correct palindrome and Caesar cipher solution, if lab 6 wrong
 - Creating and using your own module
- Basic problems:
 - lists, reading data from file

Mar 4, 2008

Sprenkle - CS111

9