

Lab 6 Feedback

- 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

Mar 9, 2010

Sprenkle - CSCI111

1

Palindrome

- Keep it simple
 - Remove spaces and lower case original word before reversing
 - Don't have to do operations twice
- More efficient solution

```
def isPalindrome(phrase):
    for i in xrange(len(phrase)/2):
        if phrase[i] != phrase[-i-1]:
            return False
    return True
```

Mar 9, 2010

Sprenkle - CSCI111

2

Which Loop to Use?

- Prob 5: Reversing String

```
for index in xrange(len(userString)-1,-1,-1):
    backwardsString += userString[index]
```

vs.

```
for character in stringStart:
    stringEnd = character + stringEnd
```

If you have a different solution, it's probably too complicated.

Mar 9, 2010

Sprenkle - CSCI111

3

Generating a Random Password

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

Define outside
of for loop

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

Good variable names

```
for char_type in xrange(len_password):
    #determines if character is number, uppercase, or lowercase
    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:
        lletter= randint(97,122)
        passwordch=chr(lletter)
    elif char_type == UPPER:
        uletter= randint(65,90)
        passwordch=chr(uletter)
    password+=passwordch
```

Even better to use
constants for ASCII
values.
(I'm short on space)

Mar 9, 2010

Sprenkle - CSCI111

4

Caesar Cipher

- Needs comments, explaining wrapping

```
HIGHEST_ASCII = ord('z')
LOWEST_ASCII = ord('a')
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 9, 2010

Sprenkle - CSCI111

5

Fines

- Make sure give correct output
- Make sure you add a comment

```
# precondition: ???
# postcondition: ???
def fine(speedlimit, clockspeed):
```

Mar 9, 2010

Sprenkle - CSCI111

6

Lab 7 Overview

- Focus: program organization
 - Defining and Using Functions
 - Will need to correct Caesar cipher solution, if lab 6 wrong
 - Creating and using your own module
- Basic problems:
 - Reading data from file
 - Processing the data
 - Handling numeric data