

Parmly Lab

- Hand Sanitizer
- Ergonomic information

Oct 23, 2007

Sprengle - CS111

1

Lab 5 Feedback

- Program organization
 - main functions at top of program
 - import statements at top of program (outside of any functions)
- Program/algorithm improvements
 - Iterate through the string as characters
 - Rather than iterating through positions in string
 - Only encode a letter if it's not a space

```
if ch != " ":  
    # convert to ASCII, etc. ...
```

Oct 23, 2007

Sprengle - CS111

2

Magic Numbers

- A lot of you used 97, 122, 26 explicitly in program
 - Sometimes not documented (commented) why
- What if ASCII values changed or used different representation?
- Better to use constants and lookup values:

```
LOWER_BOUND = ord('a')  
UPPER_BOUND = ord('z')  
RANGE = UPPER_BOUND-LOWER_BOUND + 1
```

Oct 23, 2007

Sprengle - CS111

3

Readability/Efficiency

- Added translateLetter
- Improved readability
- Slightly decreased efficiency because adds another function call
 - Payoff in readability dominates

Oct 23, 2007

Sprengle - CS111

4

Comments

- Comment functions, programs/modules at high level
- Comment inside functions at low-level (implementation details)
- Example (derived from students' submissions):

```
# This function calculates the speeding ticket fine, if speeding.  
# PRE: Input params are two positive numbers representing the  
# speed limit and the clocked speed  
# POST: Return the computed fine  
def calculateFine(speed_limit, clock_speed):  
    ...
```

Oct 23, 2007

Sprengle - CS111

5

Why is This a Good Comment?

```
# This function calculates the speeding ticket fine, if speeding.  
# PRE: Input params are two positive numbers representing the  
# speed limit and the clocked speed  
# POST: Return the computed fine  
def calculateFine(speed_limit, clock_speed):  
    ...
```

- Describes the input parameters' types and implies the type of the output
- Says that if you give it negative numbers, you could get a goofy answer back
- Does not include implementation details
- **Tells caller how to use the function**

Oct 23, 2007

Sprengle - CS111

6

Commenting Exercise

- Write a comment for this function:

```
def translateLetter(char, key):
```

Oct 23, 2007

Sprenkle - CS111

7

Commenting Exercise

- Write a comment for this function:

```
# Encodes a single character.  
# PRE: Input parameters are a single, lowercase character  
# string (char) and an integer key  
# POST: returns the encoded character  
def translateLetter(char, key):
```

Oct 23, 2007

Sprenkle - CS111

8

Commenting Exercise

- Write a comment for this function:

```
def encode(toEncode, key):
```

Oct 23, 2007

Sprenkle - CS111

9

Commenting Exercise

- Write a comment for this function:

```
# Encodes a lowercase string using a key, preserving spaces in  
# the string  
# PRE: a lowercase string to be encoded (toEncode) and the  
# integer key  
# POST: returns the encoded string  
def encode(toEncode, key):
```

Oct 23, 2007

Sprenkle - CS111

10

Commenting Notes

- Well-named parameters make documentation easier
- I'm not strict on the pre/post format.
- Just need to be clear on
 - what the function does (at high level)
 - types of parameters
 - type of the output
- ➔ The caller knows what to pass to the function and if they should assign the output to a variable

Oct 23, 2007

Sprenkle - CS111

11

Deal or No Deal Overview

- Have 26 cases with various amounts of money
 - Amounts are known
- Player selects a case (hope has the big jackpot)
- In each round, player opens up cases
 - Reveals amounts that are not in the case they chose
- Banker makes an offer to buy the case
- Player decides if want to take the deal
 - Is the offer more than what is in the case?
 - Make decision based on amounts that haven't been opened yet
- Games ends when only one more case to open or player takes the deal.

Oct 23, 2007

Sprenkle - CS111

12

Programming: Deal or No Deal

- Play the game, written in Python
- Given: partial solution in code
- Your job:
 - Read, understand given code
 - Fill in the functions for a complete solution

Oct 23, 2007

Sprenkle - CS111

13

Lab 6

- Warm up problems using lists
- Deal or No Deal

Oct 23, 2007

Sprenkle - CS111

14