**Lab: Strings and Files**

You know the drill. Work with your group – GO THROUGH THE POWERPOINT – try the activities. Then work on these problems, paste your code here, and submit to the Lab Strings dropbox at the end of class, you should submit the work you have done, whether you have finished or not.

Try running some of your programs in www.pythontutor.com if they aren’t working.

**Problem 0**

What is a Method?

A function inside of an object.

In IDLE enter

>>> dir(str)

Then ask Python for help to learn about the method\_descriptors. Paste the information on the method\_descriptors: lower, count, capitalize, find, rfind, lfind, lstrip, split, startswith, endswith

>>> help(str.lower)

Help on method\_descriptor:

lower(...)

S.lower() -> str

Return a copy of the string S converted to lowercase.

>>> help(str.count)

Help on method\_descriptor:

count(...)

S.count(sub[, start[, end]]) -> int

Return the number of non-overlapping occurrences of substring sub in

string S[start:end]. Optional arguments start and end are

interpreted as in slice notation.

>>> help(str.capitalize)

Help on method\_descriptor:

capitalize(...)

S.capitalize() -> str

Return a capitalized version of S, i.e. make the first character

have upper case and the rest lower case.

**Problem 1**

s = “Call me maybe”

slice to get “Call me” in 4 ways

s1 = s[:7]

s2 = s[0:7]

s3 = s[-14:-6]

s4 = s[0:-6]

find the length of s

print(len(s))

Change from “Call me maybe” to “Call me perhaps”

Did you edit or change s? Explain.

Use a for loop to print each letter on a line

for index in s:

print (index)

**Problem 2**

Write a program which starts like this;

quote = “640K ought to be enough for anybody.”

Which Bill Gates said about RAM in 1981. Can you add code to your program to create and print the following substrings? Can you do it 2 ways? How about 3?

print(quote.split(' '))

“640K”

“ought”

“ought to be enough”

“anybody” [you might want to use negative indexes]

“for” [again, can you do this with negative indexes?]

print(quote[0:4])

print(quote[5:10])

print(quote[5:-11])

print(quote[-8:-1])

print(quote[-12:-9])

Take Bill Gates’ quote, and print it in uppercase.

print(quote.upper())

Determine the length of the string

print(len(quote))

Print all of the words, one on each line.

print(quote.replace(" ", "\n"))

Have your program decide if the quote begins with “64” or not.

Have your program decide if the quote begins with “65” or not.

Have your program decide if the quote ends with “anybody.” or not.

Have your program decide if the quote begins with “everybody” or not.

**Problem 3**

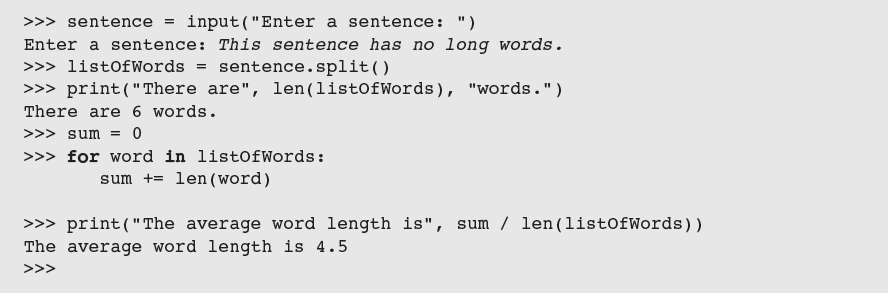
Write a program that turns a domain name, e.g. “www.google.com” or “minneapolis.ims.mnscu.edu” into the constituent labels and prints them – for example into

[“www”, “google”, “com”]

Can you add to your program so it also decides if the domain name is a commercial (.com), non profit (.org) or an educational (.edu) site?

**Problem 4**

From the PowerPoint, there’s a program which calculates the average length of words in a sentence. Type in this program, run and test it with some sentences.



Modify the program so it prints the total count of letters in the sentence – so ignore spaces.

Modify the program so it also prints the longest word in the sentence.