open.michigan

Unless otherwise noted, the content of this course material is licensed under a Creative Commons Attribution 3.0 License.
http://creativecommons.org/licenses/by/3.0/.

Copyright © 2009, Charles Severance.

You assume all responsibility for use and potential liability associated with any use of the material. Material contains copyrighted content, used in accordance with U.S. law. Copyright holders of content included in this material should contact open.michigan@umich.edu with any questions, corrections, or clarifications regarding the use of content. The Regents of the University of Michigan do not license the use of third party content posted to this site unless such a license is specifically granted in connection with particular content. Users of content are responsible for their compliance with applicable law. Mention of specific products in this material solely represents the opinion of the speaker and does not represent an endorsement by the University of Michigan. For more information about how to cite these materials visit http://michigan.educommons.net/about/terms-of-use.

Any medical information in this material is intended to inform and educate and is not a tool for self-diagnosis or a replacement for medical evaluation, advice, diagnosis or treatment by a healthcare professional. You should speak to your physician or make an appointment to be seen if you have questions or concerns about this information or your medical condition. Viewer discretion is advised: Material may contain medical images that may be disturbing to some viewers.

UNIVERSITY OF MICHIGAN

CC BY

UNIVERSITY OF MICHIGAN SCHOOL OF INFORMATION
SI 502: Networked Computing: Storage, Communication, and Processing

Assignment 5 - Figuring out who has the most commits

Due Date: Friday February 13, 2009 at 11:50PM

This program reads through a mail log, and figures out who had the most messages in the file.

The program looks for From lines and takes the second parameter on those lines as the person who sent the mail.

The program creates a Python dictionary that maps E-Mail address to the total number of messages for that person.

After all the data has been read - the program looks through the dictionary to find who has the most Mail messages and how many messages the person has.

```
$ python whomost.py
Enter a file name: mbox-short.txt
cwen@iupui.edu 5
$ python whomost.py
Enter a file name: mbox.txt
zqian@umich.edu 195
```

First, make your program so that it recovers when the user types a bad file name – and when the user simply presses enter at the file name – assume mbox-short.txt – here is some sample code:

```
fname = raw_input("Enter file name:")
if len(fname) == 0 :
    print "Assuming mbox-short.txt"
    fname = "mbox-short.txt"
try:
   infile = open(fname, "r")
….
```

You are to turn in three programs source code and a screen shot of each of the programs running:

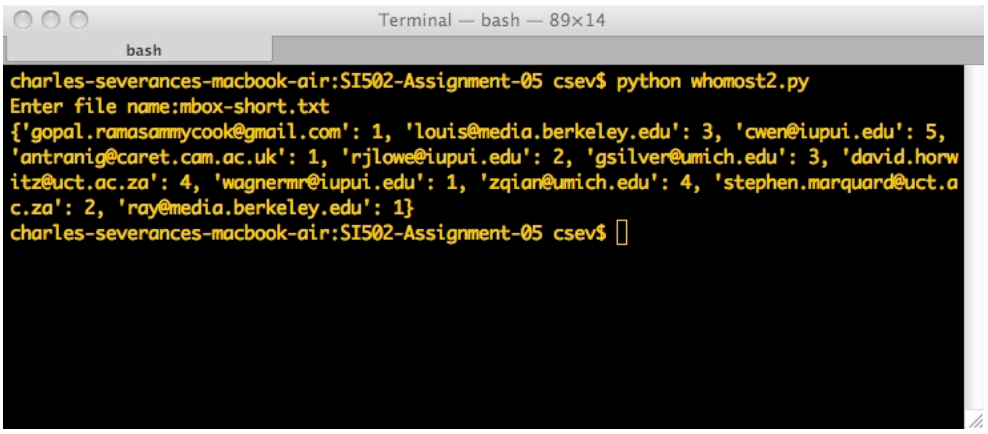
- The first program simply prints out the dictionary contents at the end of the program. Don't worry about making it pretty – just print out the dictionary. To do this you will need to make a dictionary and each time you see a new e-Mail address add an entry to the dictionary – and when you already have an entry for the address in the dictionary – you add one to the count.

- The second program simply prints out the E-Mail addresses as they are encountered in the file and prints out the current total of the messages seen so far for each person.

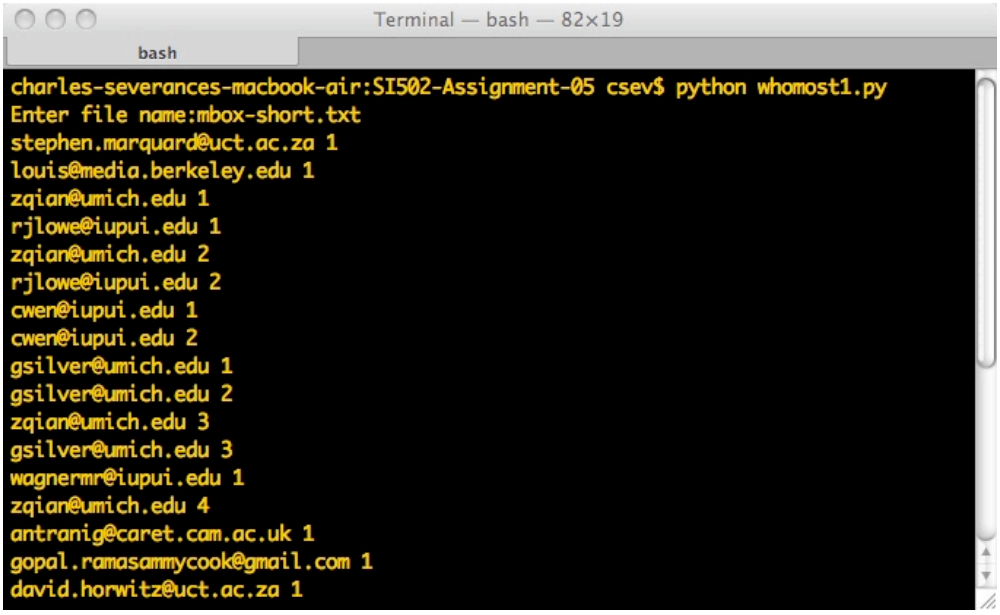
- The third program simply prints the name and number of commits of the person who has the most commits. The program does not print anything else out.

Turn in

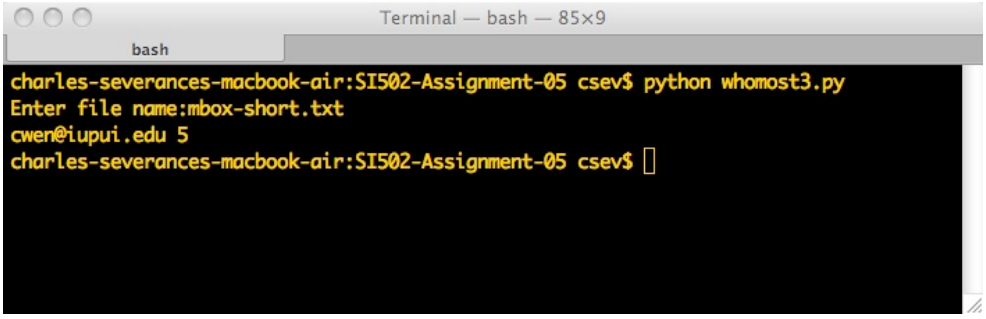
For this assignment turn in all three Python programs and three screen shots of the programs running. Sample screen shots below:



Source: Mac OS Terminal



Source: Mac OS Terminal



Source: Mac OS Terminal