Homework 0: Getting Started

Due Sunday, 29 September, 8p
Note: you won't be penalized if you are still having software install issues. We just need to know by 1 October if you are so that we can help.

Overview

This assignment consists of nine parts. None of them are hard, but some of them will take a while to download and configure software, so leave enough time. We are doing things to get started for the course, motivate the problems we will be solving in the class, and make sure that you have the proper software tools installed for the remainder of the course.

We would like you to complete this assignment by the second lecture, and that is the due date. If you have barriers in your critical path that are keeping you from doing part or all of this assignment – e.g., waiting for an external hard drive to arrive – please talk with Christina, who will work with you to figure out an alternative deadline. Note, however, that the sooner you get this setup done, the less behind you will get.

What to hand in via Canvas

1. Attach a single document (PDF preferred), containing:
    a. two screenshots of running your edited first_program.py (part 6)
    b. screenshot of Eclipse showing lecture 2 code download (part 7)
    c. text you write for parts 8 and 9
2. Also attach a copy of your code file first_program.py (part 6). Note that you can drag and drop the file from a Mint folder view to a folder on your computer outside of Mint, and can then upload it to Canvas from there.
3. Complete the survey on Canvas (see link in part 2)

Part 1: Join the 310 Facebook Group (optional)

Students are asked to contribute questions, comments, and thoughts relating to the class on the HCDE310 Facebook group. You will need a Facebook account to become a member of the group. To join the group, go to

https://www.facebook.com/groups/1539631376270810/

and click 'Ask to Join Group'. Students enrolled in the class (or on the waitlist) will be added within 24 hours. Posts made to the group are only visible to members of the group - students enrolled in the class and the instructors. In future assignments, we will be writing software to process records of activity on the group. We hope that the Facebook group will allow for more interactive and informal collaboration, but please keep in mind

that your posts will be visible to both students and instructors. This group is the place to post questions and requests for help. See the syllabus for details about the collaboration policy, which is quite liberal.

Some may prefer to keep their personal lives separate from their academic lives; you are not required or expected to "friend" anyone else in the class. You may also modify how you are notified about new posts to the group by clicking 'Edit Settings' in the upper right hand corner of the screen.

Part 2: Complete a background survey (if you haven't already)

One of the great things about HCDE is that students come from a variety of backgrounds and bring a mix of expertise and interests. This can, however, make pacing the course a bit of a challenge. To help us pace the course this quarter, please complete a survey at: https://canvas.uw.edu/courses/916351/quizzes/799006/take

Part 3: Get your tools set up

The easiest way to do this is to download and install a Linux Mint Virtual Machine. To do this, follow the installation instructions on Canvas, at https://canvas.uw.edu/courses/916351/pages/Software%20setup

Alternatively, you are welcome to install your own environment on your computer. You'll need Python 2.7 and a development environment (we will be teaching using Eclipse, but you can do something else), and the ability to access SVN repositories. *If you choose to go this route, rather than use the virtual machine, we will not be able to help you troubleshoot technical issues – there are many, many configurations possible and we cannot support them all. Also, the remainder of the instructions are written for the Virtual Machine. Adapt as appropriate for your environment.*

Part 4: Learn to use a terminal window

In Mint, open a terminal window. If you're not familiar with Unix commands, consult the How To document "Unix commands." Do something in the terminal window to try out all of the following commands and actions:

- `ls`
  try `ls -l` and `ls -a` and try to figure out what's different about the output; if you're stuck, discuss on the Facebook group
- `cd`
- `pwd`
- `cat` to view the contents of a file
- `<tab>` for auto-completion
- Up and down arrows for rerunning commands

Part 5: Learn to execute Python files from a terminal window

In the terminal window, navigate to /home/me/Homeworks/hw0:

```
cd ~/Homeworks/hw0
```

The file first_program.py contains a python program. Execute it by entering the command: `python2.7 first_program.py`.

Part 6: Learn to use the Eclipse editor to modify a simple program.

Now edit the program. Double-click on the "Eclipse" application on the Mint desktop.

In Eclipse, on the left side, you'll see the project "Homeworks". Inside folder "hw0," click on "first_program.py". The file will open in the center pane.

Follow the instructions inside the file to edit. Don't forget to save the file before you try to run it again— otherwise you won't see the effects of your edits!

When you get to the end of the instructions in the program, take two screenshots, one of the program executing in a terminal window and one of it executing inside Eclipse. If you're not sure how to take screenshots on your computer, you can use:

- in Mint: the built-in screenshot capability. For some hints, see http://community.linuxmint.com/tutorial/view/85.
- in Windows: alt+PrintScreen (while running Player)
- in OS X: the Grab utility or Command-Shift-4 (see http://guides.macrumors.com/Taking_Screenshots_in_Mac_OS_X). Or you can take a screenshot of the entire screen using Command-Shift-3.

Part 7: Use the download tools to get code snippets for lecture

In Eclipse, on the left side, right-click on the "lectures" project, select Team > Update.

To show that you've figured out how to do this, take a screenshot showing the code file for lecture 2 that shows up on the left side in the project listings area.

You should do the Team->Update action at the beginning of each lecture. Often, this will cause some code snippets to download. You'll learn more about the version control system (svn) we're using for code distribution as the quarter goes on.

Part 8: Design a program

The purpose of this exercise is to get you familiar with issues in data manipulation and to get you thinking about how a computer might do it. For this exercise, write an English description (not a computer program!) of instructions for counting the number of words in a document (text file). Try to be as precise as possible and remove all ambiguity. It may help you to eliminate ambiguity if you anticipate that the person to whom you are giving the instructions will purposely try to misinterpret your instructions so as to get the wrong result.

You may be asked to team up and compare your approach with a classmate's in the next section, so please be prepared to share.

Part 9: Reflection

Was the previous exercise easy or hard for you? What was satisfying or unsatisfying about trying to describe the instructions in English?

Also, it would help us if you would take a few sentences tell us what you want to get out of the course. Thanks!