

CSC108H/A08H Week 2 Lab

Welcome to the first CSC108H/A08H lab! To earn your lab marks, you must actively participate in the lab. *You do not need to finish in the time allotted, you just need to arrive on time and work hard.*

1 Objectives

1. Introductions
2. Log in
3. Set up email forwarding
4. Define *driver* and *navigator*
5. Image manipulation

2 Introductions

Your TA will lead you in an activity that will help you get to know your TA and labmates better.

3 Log in and change your password

Sit down with your partner. Complete the following task twice, once for each student in your group.

- Log in: enter your user ID and password. Wait while the computer starts up. If you don't know your CDF username, then you can look it up: <http://www.cdf.toronto.edu/cgi-bin/webfinger>

The first time you log in you will be asked to change your password from your student ID to something more secure. Change your password and then let your partner sign in and change theirs.

4 Set up email forwarding

We will sometimes send important email to everyone's CDF account. You need to set up "email forwarding", so that the email sent to your CDF address will be forwarded to a different address that you check regularly. We suggest that you use your utoronto account (and make sure that you check *it* regularly, because the university sends very important messages there).

Execute the following set of instructions twice, once for each student in your group.

- (a) Log in using your new password.

Please note that you may not be able to login again immediately, because the network takes a few minutes to update the password changes. Please be patient and use the time to read the lab handout and to get to know your partner better.

- (b) From the menu, launch Wing. Wait while Wing starts up; please be patient, and don't close the little window that pops up.

- (c) In Wing, click "New". In the upper pane, type an email address that you check regularly.

(d) Select **File|Save**. A dialog box will appear. In the text field labelled

File name:

type exactly this:

`.forward`

Change the “**Save as type:**” to “**Plain text**”. Then select your home directory (`/../../c8_____`) in the “**Save In:**” drop-down menu, and click the **Save** button.

(e) Quit Wing and log out. Your partner should now complete steps (a) to (e).

5 Driver and Navigator

Throughout the term, we will use the terms *driver* and *navigator*. Here are the definitions of the two roles:

driver: The person typing at the keyboard.

navigator: The person watching for mistakes, and thinking ahead.

Here is the most important rule for this and all future labs:

The navigator must not touch the keyboard. If the navigator does type when they are not supposed to, the navigator will get a zero for this lab. (The navigator may use the mouse, but only occasionally: the driver should be in control of it most of the time.)

The rest of these instructions call you two `s1` and `s2`. Pick which one is which. `s1` will be the first driver.

6 Image Manipulation

Now that the administrative details are out of the way, we can get started with the fun part of the lab! For many of you, this will be the first time that you have programmed, which is exciting, but it may also be a bit overwhelming. Don't hesitate to ask your TA or your labmates for help. You may also refer to your lecture notes and textbook.

To begin, visit the Labs page of the course website (<http://www.cdf.toronto.edu/~csc108h/summer>) and download all of the lab's associated files to your home directory (`/../../c8_____`).

You will recall from lecture that one can trace the execution of a program, step by step, and examine the values of the variables as the program executes. We will start this week's lab by having you do this on the program `w2.trace.py`.

Double-click the Wing icon. It should open several panes; one of these is the editor/tracer on the top, and another, at the bottom, is the Python shell. It looks something like:

```
Python 2.5 (r25:51908, Sep 19 2008, 09:52:17) [MSC v.1310 32 bit (Intel)]
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

Complete the following tasks:

- (a) Open file `w2.trace.py`, which you just downloaded. Opening the file puts the text of the file in the editor window.
 - Click the **Step Into** button. This will highlight the first line of the file in the editor window and it will start the step-by-step tracing or debugging process.

- In the lower left-hand area of Wing, click on the **Stack Data** tab. This is where the values of the variables are displayed. For now you can ignore names that are surrounded by underscores (`--`). You will notice that the variables `x` and `y` are not currently listed. Why not? (If you don't know, ask your partner: if your partner doesn't know, ask the TA.) What about the other variables in the program?
 - The PDF file `w2.trace.pdf` has code on the left-hand side of the page and a list of variables on the right-hand side. Discuss with your partner what values the variables will have as the program executes. In a text file, write down these lines of code and variable values. Some variables may be unpredictable. Why? Show your text file to your TA.
- Switch roles: s2 drives and s1 navigates.**
- For the rest of this exercise, you will use the **Step Over** button (instead of **Step Into**, as above). Each time after pressing this button, check the contents of the **Stack Data** window.
 - As you keep stepping over, write down the values for the variables on your list, and see if they match your predictions. If they don't, discuss with your partner (and/or your TA) to determine why not.

Switch roles: s1 drives and s2 navigates.

- Open file `w2.py`, which you just downloaded.
- Click the Run button.
- In the shell, call function `get_picture()` and store the value returned in a variable called `pic`. To display the picture, call `media.show(pic)`.
- Call function `maximize_red`, passing `pic` as an argument. You should probably choose a small picture so that it won't take too long to run; We have provided the Bahen picture for you to work with.

Now call `media.show(pic)` to see the modified picture.

Switch roles: s2 drives and s1 navigates.

- Add to `w2.py`: define a function called `remove_blue` that takes a picture as a parameter. It should set each pixel's blue value to 0.
- In the Python shell, call `remove_blue` with `pic` as an argument. If you have not clicked the Run button since defining `remove_blue`, then you will not be able to execute it: the shell will not know that it exists. If that is the case, click Run. You'll notice that the shell is reset, which means you'll have to start over and reopen the picture before calling `remove_blue`. After you call `remove_blue`, call `media.show(pic)` to display the picture.
- Add to `w2.py`: define a function called `halve_green` that takes a picture as a parameter. This function should set each pixel's green value to half of the old green value.

- In the shell, call the function on your picture and then call `show` to see the result.

Switch roles: s1 drives, and s2 navigates.

- Add to `w2.py`: define a function called `swap_red_blue` that takes a picture as a parameter. This function should swap each pixel's red and blue values. For example, if a pixel's red and blue values are 117 and 38, respectively, then the red and blue values should be set to 38 and 117, respectively.
- In the shell, call the function on your picture and then call `media.show` to see the result.
- If time permits, try calling the functions that you have written on your picture in different combinations to see how you can modify the image.