Sessions and C Programming
Lab D

A mentoring section is provided at the end of this lab.

The labs, for this course, are designed to be completed on your own at home or in the 3rd floor Trottier labs. These labs are not graded. You do not hand in these labs. If you prefer to work on a lab with your TA tutorial group, then check the schedule for your TA's tutorial session. You will find this schedule in our MyCourses page under Content/Course Information/TA Information. Since the university has limited lab space, your TA might ask you to bring your laptop and work in a classroom instead of a lab.

This lab is about programming in C.

Some labs will have a question zero. These questions will not be covered by the TA during the tutorial. It is extra content meant for you to do on your own.

QUESTION ZERO: Optional problem

Type `man gcc` to see what it tells you.

Use `echo $SHELL` and our class notes to see which shell you are using to determine your login script.

QUESTION ONE: Edit your login script

Your login script is probably `.bashrc` but you should make sure. You can make sure by editing .bashrc and then logging out and then back in again to see if anything happened. If the file does not exist, then create it.

Do the following at the command-line:

```
vi .bashrc
```

Then edit the file in the following ways:

- Change your prompt
- Echo Welcome to Bash

You will need to logout and then log back in to make sure it is running. You can test it temporarily by typing at the command-line:

```
source .bashrc
```

But to make sure it actually runs when you login, the best test is to logout and then back in.

QUESTION TWO: Compiling your first C program

Using vi type in the following C program, and call the file `labd.c`:

```
#include<stdio.h>
int main() {
```

```
    puts("Hello World\n\n");
    return 0;
}
```

Now compile that file by typing the following command at the command-line:

```
gcc labd.c
```

If you see no errors, then run the program by typing at the command-line:

```
./a.out
```

If there were errors, then look at the errors and fix them and try to compile and run again.

QUESTION THREE: Edit your first C program

Modify the file `labd.c` in the following way:

- Create an integer variable named count and initialize it to 5.
- Using a for loop, print out the Hello World, count times.

Compile and run the program.

You have completed your lab.

TA MENTORING SECTION

During this mentoring session, the TA will do the following:

- Help the student discover their mimi login script
- Show the student how to edit and run their login script from the command-line
- Show the student how to manage C language compile time errors from the command-line
    - Redirect errors to a text file
    - Use the vi colon command to jump directly to an error

- The TA will also guide the student through this lab