

# Lab Assignments

**Quick Links:** [Home \(https://q.utoronto.ca/courses/419441/assignments/syllabus\)](https://q.utoronto.ca/courses/419441/assignments/syllabus) [Lecture material \(https://q.utoronto.ca/courses/419441/pages/lecture-material\)](https://q.utoronto.ca/courses/419441/pages/lecture-material) [Lab assignments \(https://q.utoronto.ca/courses/419441/pages/lab-assignments\)](https://q.utoronto.ca/courses/419441/pages/lab-assignments) [Piazza Discussion !\[\]\(c8d96c8885d3000a912c2582004aed63\_img.jpg\) \(https://piazza.com/utoronto.ca/winter2026/ece353/home\)](https://piazza.com/utoronto.ca/winter2026/ece353/home)

---

## Lab Information

A major component of this class consists of five programming assignments. These assignments will be released on this page. All the labs are to be done individually. *This year, all lab assignments must be done and submitted remotely.*

*Please be aware that this course will require significant time for lab work.* However, the assignments have been designed so that there is enough time to completely finish each assignment. Many of these assignments require **significant** time for reading the lab documentation and the code provided for the assignment. Each assignment specifies the code that needs to be read and understood. Please read the handouts carefully before starting with the labs.

We will be using the [ECE Workstation Labs \(http://www.ece.utoronto.ca/lab-support/home-directories/\)](http://www.ece.utoronto.ca/lab-support/home-directories/), also called UG machines, for ECE353. Your assignment code is expected to run on these UG lab machines ([see below](#)). We will provide several unit test and an automated testing infrastructure that you can use to test each assignment. For more details, please see how you can [test your code \(https://q.utoronto.ca/courses/419441/pages/testing-your-lab-code\)](https://q.utoronto.ca/courses/419441/pages/testing-your-lab-code). We provide detailed instructions for submitting your lab work as part of the lab assignment instructions. For more details, please see how you can [submit your code \(https://q.utoronto.ca/courses/419441/pages/submitting-lab-code\)](https://q.utoronto.ca/courses/419441/pages/submitting-lab-code).

Assignments will not be accepted after their due dates, without prior arrangement with the course instructors. In general, no deadline extensions will be granted to individuals except in severe circumstances.

Marking will be performed by the TAs after the lab due dates. Marking will generally be performed using automated test scripts that check the output of your code to ensure that the implementation works correctly. The TAs will send you an email with your lab marks.

To get help with labs, you can attend the lab sessions, or you should use the Piazza website. Please do not post code on Piazza. Doing so will be treated as an academic offense. You can also post messages on Piazza that are only visible to TAs and the instructors.

# Lab Assignments

The lab assignments, their release dates, due dates and the time allotted for the assignment are shown in the following table. Labs are worth 25% of the total course mark. The breakup of the marks for each lab is shown in the last column of the table.

Assignments will generally be released on Friday evenings. *The previous assignment is typically due on the same Friday evenings at 11:59 pm.*

Lab	Topic	Release Date	Due Date	Time Allotted	Lab Marks
1	<a href="https://q.utoronto.ca/courses/419441/pages/lab-1-review-of-c-data-structures-using-system-calls">Lab 1: Review of C, Data Structures, Using System Calls</a> ( <a href="https://q.utoronto.ca/courses/419441/pages/lab-1-review-of-c-data-structures-using-system-calls">https://q.utoronto.ca/courses/419441/pages/lab-1-review-of-c-data-structures-using-system-calls</a> )	Jan 9	Jan 23	2 weeks	5

Please go over the lab assignments as they become available and the frequently asked questions below.

## Getting Started With Labs

All source code must be compiled and submitted on the lab computers. These computers are Linux machines with names that matches `ug*.eecg.utoronto.ca`, e.g., `ug149.eecg.utoronto.ca`. You can tell which machine you are using from the command prompt. For example, if you are using `ug149.eecg.utoronto.ca`, the command prompt will show `ug149`. The range of machines that are accessible is roughly `ug51-ug100`, `ug132-ug180` and `ug201-250`.

Each student has been assigned a *user number* for this course. You can find out your user number by typing the `groups` command after logging into the lab machines.

```
$ groups
```

This command provides information about all the Unix groups to which you belong. One of them should be in the range from `ss-001` to `ss-200`. The *3-digit* number after `ss-` is your user number for this course (e.g., with `031`, the user number is `031` and not `31`). If you do not have a user number, please send an email to [Tim Trant \(mailto:tim@eecg.utoronto.ca\)](mailto:tim@eecg.utoronto.ca), the lab administrator, and ask him to create a user number (starting with "ss-") for you. Make sure to include your UTORID in your email.

Below, we have provided instructions for accessing the [lab machines remotely](#). You can also setup your [SSH Keys \(https://q.utoronto.ca/courses/419441/pages/setting-up-ssh-keys\)](https://q.utoronto.ca/courses/419441/pages/setting-up-ssh-keys), which simplifies using ssh for remote access.

## Lab Documentation

These documents will be helpful for the ECE353 labs.

- [Testing your lab code \(https://q.utoronto.ca/courses/419441/pages/testing-your-lab-code\)](https://q.utoronto.ca/courses/419441/pages/testing-your-lab-code)
- [Submitting your lab code \(https://q.utoronto.ca/courses/419441/pages/submitting-lab-code\)](https://q.utoronto.ca/courses/419441/pages/submitting-lab-code)
- [Git Resources \(https://q.utoronto.ca/courses/419441/pages/git-resources\)](https://q.utoronto.ca/courses/419441/pages/git-resources)
- [Setting up SSH Keys \(https://q.utoronto.ca/courses/419441/pages/setting-up-ssh-keys\)](https://q.utoronto.ca/courses/419441/pages/setting-up-ssh-keys)
- [Debugging with GDB \(https://q.utoronto.ca/courses/419441/pages/debugging-with-gdb\)](https://q.utoronto.ca/courses/419441/pages/debugging-with-gdb)
- [Using Tags \(https://q.utoronto.ca/courses/419441/pages/ctags-and-stags-for-code-navigation\)](https://q.utoronto.ca/courses/419441/pages/ctags-and-stags-for-code-navigation) for code navigation
- [Using Cscope \(https://q.utoronto.ca/courses/419441/pages/cscope-for-code-navigation\)](https://q.utoronto.ca/courses/419441/pages/cscope-for-code-navigation) for code navigation
- [Using Screen \(https://q.utoronto.ca/courses/419441/pages/screen-for-managing-multiple-terminal-windows\)](https://q.utoronto.ca/courses/419441/pages/screen-for-managing-multiple-terminal-windows) for managing multiple terminal windows
- Introduction to [Make \(pdf file\) \(https://q.utoronto.ca/courses/419441/files/41484318?wrap=1\)](https://q.utoronto.ca/courses/419441/files/41484318?wrap=1)   
([https://q.utoronto.ca/courses/419441/files/41484318/download?download\\_frd=1](https://q.utoronto.ca/courses/419441/files/41484318/download?download_frd=1))

## Frequently Asked Questions

Please see the frequently asked questions about the ECE353 labs.

- [Tester FAQ \(https://q.utoronto.ca/courses/419441/pages/faq-tester\)](https://q.utoronto.ca/courses/419441/pages/faq-tester)
- [Labs FAQ \(https://q.utoronto.ca/courses/419441/pages/faq-labs\)](https://q.utoronto.ca/courses/419441/pages/faq-labs)
- [Lab 1 FAQ \(https://q.utoronto.ca/courses/419441/pages/faq-lab1\)](https://q.utoronto.ca/courses/419441/pages/faq-lab1)

## Remote Access to Lab Machines

To access these machines remotely, you need to be connected to the [U of T network \(http://help.ic.utoronto.ca/category/51/internet-\\_connectivity.html\)](#). After that, you can use SSH or Putty to access the Unix machines remotely. Here is a good guide for using setting up [SSH or Putty access \(https://undergrad.engineering.utoronto.ca/undergrad-resources/engineering-computing-facility-ecf/remote-access/\)](https://undergrad.engineering.utoronto.ca/undergrad-resources/engineering-computing-facility-ecf/remote-access/). *Note that these instructions are for ECF machines. However, you need to use the [UG machines \(http://www.ece.utoronto.ca/lab-support/home-directories/\)](http://www.ece.utoronto.ca/lab-support/home-directories/) for these labs.*

You can use the [screen \(https://q.utoronto.ca/courses/419441/pages/screen-for-managing-multiple-terminal-windows\)](https://q.utoronto.ca/courses/419441/pages/screen-for-managing-multiple-terminal-windows) program to multiplex multiple virtual terminals onto a single console. This is especially useful when debugging Lab 4 and Lab 5, which involve a client and a server program.

If you wish to use a graphical IDE to develop and debug your programs on the ECE lab machines from a remote site, you will need to use VNC. We recommend that you attempt this only if you are a "power user" and know how to get it to work on your own. Here is a good guide for connecting via [VNC on the UG machines \(https://www.eecg.utoronto.ca/~vaughn/ece297/ECE297/assignments/vnc/vnc.pdf\)](https://www.eecg.utoronto.ca/~vaughn/ece297/ECE297/assignments/vnc/vnc.pdf).

## Lab Machine Issues

Some students exceed their disk quota and then they are unable to access the lab machines. In this case, a remote ssh connection will still work (from ECF machines, or from a friend's account). After logging into the lab machines, run the command

```
du -sk * .??*
```

to find out exactly where the space is being used. Huge browser caches and/or unemptied Trash are the usual causes. You can also use the command


```
quota -s
```

to find out your disk quota on the machines.

If you have additional issues with the lab machines, including with forgotten passwords, please contact [Tim Trant \(mailto:tim@eecg.utoronto.ca\)](mailto:tim@eecg.utoronto.ca), the lab administrator, and make sure to include your UTORID in your email.

## The C Language

Here are some links to tutorials for C.

- [Programming in C: A Tutorial](http://www.lysator.liu.se/c/bwk-tutor.html)   [\(http://www.lysator.liu.se/c/bwk-tutor.html\)](http://www.lysator.liu.se/c/bwk-tutor.html) Brian W. Kernighan
- [C Programming](http://ergodic.ugr.es/cphys_pedro/c/c/ccourse.html)   [\(http://ergodic.ugr.es/cphys\\_pedro/c/c/ccourse.html\)](http://ergodic.ugr.es/cphys_pedro/c/c/ccourse.html) Steve Holmes
- A short [tutorial on pointers in C \(https://q.utoronto.ca/courses/419441/pages/pointer-tutorial\)](https://q.utoronto.ca/courses/419441/pages/pointer-tutorial) by Ashvin Goel