

Chapter 5

Local system particulars

The information in this chapter is believed to be mostly valid as of December 5, 2008. As the computer systems undergo updates and upgrades, some of it will become obsolete.

5.1 Electronic submissions

The command `submit` is to be used to submit your assignments electronically. Run it from your local shell window prompt in Linux.

You should place all the files you wish to submit for your assignment in a directory with a certain name determined by your instructor: `math2130-a1`, `math2130-a2`, etc. We recommend that this directory be different from your working directory for the current assignment (it is convenient to make it a subdirectory). After you have completed your work, the working directory will contain files with extensions `.log`, `.aux`, `.bak`, which should not be included in the submission. The working directory can also contain drafts that you do not want to submit. Copy only those files to the submission directory that are needed to reproduce your work in a form identical to the printed copy you are submitting. Include the `.tex` file (or files, if there are several) and all graphics files (`.eps`, `.pdf`, `.eepic`, etc.) that your master file refers to. Also include a copy of your computer program (`.f`, `.c`, etc.) or Maple worksheet (`.mw`). If there is a certain data file with “hidden” (not told to the reader) data which your report depends on, it too should be included.

Before issuing the `submit` command, make sure your **current directory is the parent directory of the directory containing the files to be submitted**. For example, if your Assignment 1 files are located in `~/A1/math2130-a1`, use the command `cd ~/A1` to change into the `A1` directory before running the `submit` command.

Actual submission for Assignment 1 for a course called `math2130-2` (`math2130`, section number 2) is done by typing the following in the command line (from the parent directory of the directory `math2130-a1` containing your submission:

```
submit submit math2130-2 a1
```

Assignment 2 will have a submission ID of a2 and so on. Students of section 3 should replace `math2130-2` into `math2130-3`.

You will be asked to enter your password.

(Students who wonder why the word *submit* is repeated twice can compare the above command with the one that an instructor would use to retrieve your submissions:

```
submit retrieve math2130-2 a1.)
```

If you are not sure of the course or assignment ID to use with the `submit` command, you may type at the shell window prompt

```
submit list
```

to get a list of courses and dates on which there are assignments due.

If you mistakenly submit incorrect file(s) or you make changes to your file(s) after you have submitted them, you may submit again (before the due time). Your most recent submission must exactly match the submitted hard copy of the report.

Be aware that the size of your total submission is limited by 10 MB and the size of each single file being submitted should not exceed 2MB.

Why are the electronic submissions required? They are archived year over year and can be automatically checked to detect any occurrences of textual overlap. An instructor can use electronic submissions to verify whether your reported results are actually obtained through your program. If such a verification is impossible due to insufficient information provided, the instructor has a right to doubt the integrity of your research (see Sect. 1.3.2).

5.2 Laboratory computers on campus

5.2.1 Where

There are two computer laboratories available to mathematics and statistics majors, HH-3030/3056 and EN-2036. As well students may use computers in various General Access Labs around campus.

For example, students may access their accounts through one of the computers located in Chemistry/Physics General Access Lab (C-2004) or at the Commons located in the QEII Library. They also may access computers in C-2003 and CS-1019 when there are no classes taking place in these labs. Please note that those machines are maintained by the Department of Computing and Communication. Should you have difficulty logging into a computer, please advise the person who is working in the lab.

5.2.2 Your computer account

Everyone has been assigned a computer account, which provides access to the desktop workstations, which comprise **MUN LABNeT**. Each student is provided with his/her *home directory*

on the student disk, where your personal files reside. You are not able to read or write files in someone else's home directory and no one else can read or write to yours.

Each workstation has a name. Newfoundland communities are used in HH-3030/3056; the workstations in EN-2036 are named after Transformers characters. The machines in HH-3030/3056 will boot to either Linux or Microsoft Windows. They boot to Windows by default. To work with L^AT_EX and program compilers, you must click on Linux.

To log in, type your user name and hit *Enter*, then type your password and hit *Enter* in the appropriate field in the middle window. Assuming no typing errors, the screen will clear and you will see a desktop with a few icons to the left and a toolbar along the bottom.

The home directory, where all your files reside, does not depend on the particular workstation you are using, nor on the operating system you are booted to. Thus if one day you log onto **lumsden** in HH-3030/3056 under Windows and the next day to **bumblebee** in EN-2036 under Linux, your files will be the same. Whenever you create a file and save it, it does not get saved on the local or remote workstation; rather it is written in your home directory on the student disk and it is then available to you from any other workstation. This is all done in a seamless fashion so you really need not concern yourself.

Organize your work. Create a separate subdirectory in your home directory for each course. Within it, create a separate subdirectory for each project. In your Math-2130 directory, you may also create a subdirectory for L^AT_EX and Maple samples provided by the instructor or for your own experiments with these software systems.

We suggest that you manage your home directory carefully, regularly deleting unnecessary files. Though disk space limitations may not be a big concern these days, regular clean up will keep your directory well organized; otherwise you will feel an increasing discomfort due to accumulating junk files.

Rules of conduct. You are strongly encouraged to read and heed the university's policy concerning the appropriate use of campus computers. It can be found at

<http://www.mun.ca/scac>

5.2.3 Printing

A laser printer is available in each lab. It is important to replenish your paper account in advance of the submission deadline. Inability to get a printed copy of your work due to insufficient funds is not an excuse for missing a deadline. There are several stations across Campus where you can deposit funds onto your Mun One Card. There is one in the library Commons area and one in the Chemistry/Physics (CP2003-4). This money is not locked into a particular lab and can be used on all LABNeT printers. A noted peculiarity is that, once started, a printing job will apparently be finished even if it results in a negative account balance — but the card reader will be showing zero balance until you cover the debt.

5.3 Software

5.3.1 Processing L^AT_EX files in the command line

You process the file `mylab1.tex` with the command

```
latex mylab1
```

Note that there is no need for the extension `.tex`.

If there are errors, L^AT_EX will stop at the first one and leave you hanging at a question mark on the screen. At this point, you may answer `x` to stop processing and to fix the reported error. Or if you answer `r` (for *run*), L^AT_EX will finish processing to the best of its ability, writing all errors to `mylab.log`, which you can then review in one window while correcting `mylab.tex` in another.

To view `mylab.dvi`, use a UNIX program called `xdvi`:

```
xdvi mylab1
```

Again, there is no need for the extension `.dvi`.

As of now, it is not possible to print your `dvi` document from the viewer's window. For printing, you need to convert it to either to a Postscript file:

```
dvips mylab1.dvi mylab1.ps
```

or to a PDF file:

```
dvipdf mylab1.dvi mylab1.pdf
```

To open a Postscript picture, invoke the GhostView viewer by typing `gv` or `ghostview` on the command line:

```
gv mylab1.ps
```

PDF files are viewed with familiar Acrobat Reader:

```
acroread mylab1.pdf
```

Both GhostView and Acrobat allow you to print the document you are viewing.

L^AT_EX files that do not contain `eps` graphics can be conveniently processed in one step by the `pdflatex` command:

```
pdflatex mylab1
```

Remember to run the L^AT_EX compiler (whether `latex` or `pdflatex`) **two times** if your file contains either of the following: the table of contents, the bibliography section, automatically numbered equations or figures.

5.3.2 Kile — integrated L^AT_EX environment

Kile (launched by the command `kile`) is a UNIX text editor specifically designed to assist with preparation of L^AT_EX documents. Besides the many functions common to text editors (including spell checking and text highlighting), Kile allows you to compile L^AT_EX documents and view them without switching to a shell window.

The icon depicting a little blue wheel invokes `latex` command. The icon where the wheel overlaps with a red curve invokes `pdflatex`. There are buttons that launch `dvi`, `ps`, and `pdf` viewers, and buttons that convert `dvi` to `ps` or `pdf`.

5.3.3 Compilers

There are compilers available on the Linux machines for the FORTRAN, C, C++, and Java languages. Some students may use other languages of their choice, like Python, for example, unless the instructor raises objections.

There are naming conventions for source code filenames which must be adhered to with each of the various compilers.

FORTRAN	There are two compilers, <code>f95</code> and <code>ifort</code> , and program source filenames must end in <code>.f</code> .
C	The compiler is <code>gcc</code> , and program source filenames must end in <code>.c</code> .
C++	The compiler is <code>g++</code> , and program source filenames end in <code>.c</code> , <code>.cc</code> , or <code>.cxx</code> .
java	The compiler is <code>javac</code> , and the program source file names end in <code>.java</code> .

For example, suppose we have a FORTRAN program named `assign1.f`.

1. Compile the program with the command

```
lumsden $ f95 assign1.f
```

which creates the executable file `a.out`. If you want to have the executable file named differently, use the `-o` option as in

```
lumsden $ f95 assign1.f -o assign1
```

which will name the executable file `assign1` instead of the default `a.out`.

If you just want to get an error report, use

```
lumsden $ f95 assign1.f -c assign1
```

Any syntactical errors in your program will be reported by the compiler. If any are found you will have to go back to the first step and re-edit your source file.

2. Run your program, which is done simply by typing the name of the executable file. For example, if you created the executable file `assign1` as in Step 1, it is run with the command

```
lumsden $ assign1
otherwise, the command would be
lumsden $ a.out
```

5.3.4 Maple

To enable access to Maple in your account, a licence information must be set. This needs to be done only once, by issuing the command

```
export LM_LICENSE_FILE=28002@noether
```

After that, you should be able to use Maple.

The command `xmaple` invokes Maple 11 (the most recent version currently available in the lab) in the standard mode.

The command `xmaple -cw` invokes Maple 11 in the Classic Worksheet mode.

The command `xmaple 9` invokes Maple 9.

Maple can be invoked in a non-graphical command line mode by the `maple` command. The session opens quickly. This mode is particularly advantageous if you connect to Maple from your home computer and do not need graphics. To end the session, type **quit**.

5.3.5 Miscellaneous

1. Gnuplot is invoked by the `gnuplot` command.
2. XFig is invoked by the `xfig` command.
3. To convert a `ps` file into `eps` file, many UNIX systems have a command `pstoeps` or `ps2eps`. Windows-based versions of GhostView have an EPS convertor under *File* menu. Unfortunately, these options are not available on the LabNET machines. Here is a conversion method, which is a little ugly, but it works. Suppose `fig1.ps` is the name of the original Postscript file.

- (a) Type the command

```
gs -sDEVICE=bbox fig1.ps
```

- (b) Among the output that appears on the screen, find and select by mouse a line that looks like this:

```
%%BoundingBox: 20 118 575 673
```

- (c) Open the file `fig1.eps` in a text editor (say, in Kile). The first line will look like this:

```
%!PS-Adobe-2.0
```

Change it into

```
%!PS-Adobe-3.0 EPSF-3.0
```

Paste the selected BoundingBox specification immediately after, so the second line in your updated file must be similar to this:

```
%%BoundingBox: 20 118 575 673
```

- (d) Finally, save the file as `fig1.eps`.
4. Do not use command `pdflatex` or the corresponding icon in Kile if your \LaTeX file contains references to `eps` graphics. Compile your `tex` file into a `dvi` file and then convert `dvi` into `pdf` if you wish. You can successfully use `pdflatex` if the only type of graphics in your document is that provided by the \LaTeX 's *picture* environment or its extensions as described in Sect. 4.4.
 5. The command `lgrind` (Sect. 3.1.9) may not work on your home computer (although you may have successfully installed a \LaTeX distribution).