LANGARA COLLEGE

DEPARTMENT OF COMPUTING SCIENCE AND INFORMATION SYSTEMS

ALGORITHMS AND DATA STRUCTURES' COURSES

by Gladys Monagan[1]

Technical Guide for Developing Code in C++

COMPUTER ACCOUNTS

At Langara, your home directory on most Langara labs is on the so-called "H" drive. There is only a limited amount of memory allocated to each account. If you have almost used up all the memory allocated to your account, you will not be able to work anymore. You will not be able to save files produced (e.g. by the compiler and linker) such as object files and executables. So, you need to free up memory by deleting files. If the files are your valuable assignments (e.g. source code), you probably want to save the files first before deleting them.

You can access the files in your "H" drive from home (or access the files from the lab C014) using https://myfiles.langara.bc.ca/NetStorage. You can get (download) files from your "H" drive and you can (upload) put files onto your "H" drive. You cannot run applications on the Langara machines, though. After accessing the files, do not forget to logout out of Storage.

COMPILERS, EDITORS, AND IDES

An Integrated Development Environment (IDE) incorporates programs for editing, compiling and running programs written in C++, Java, and other languages. The editor is for entering the source code and for revising it. The C++ compiler translates the source code into object code and the linker puts together the object code produced by the compiler from your source code with other object codes (like libraries) into an executable.

---

[1] parts written by Brian Green

COMPILERS

At Langara, most of our labs have Windows machines. We are using the compiler and linker g++ (from the GNU foundation). C++ is a high level language and although quite standardized, the C++ standards committee has been advancing the language. We are going to be using C++11 in this course. All the C++ source code that you submit for any assignment and for any lab, must compile without errors on the Langara Windows machines or on the server **csis-lab.j43.ca**. All the C++ source code that you submit must compile with the compiler and the linker called g++ and with the compiler flags `-pedantic -Wall -Wextra -std=c++11.`

EDITORS

SciTE

I will use a simple text editor called SciTE (which stands for Scintilla Text Editor) in the lab for the beginning of the course. You should find it quite easy to use and install at home too, especially if you took CPSC 1150 or CPSC 1155 with us. You might want to switch to a proper IDE later in the course. SciTE can be used with compilers of programs written in several other languages, so it may be useful to you in other ways later. Since it is free for Windows and Linux, you can download it at home from the Scintilla website http://scintilla.org/SciTE.html

SciTE is a simple text editor that will enable you to compile, link and later call the executable. You can even invoke the Makefile (though there are some quirks) The set up of SciTE in the labs at Langara consists of a text editor, a command to call the compiler, a command to call the Java 8 interpreter and a command for producing documentation with **javadoc**. Note that just having installed the SciTE editor in your computer is not enough to compile and run your Java programs.

Warning: SciTE has a most annoying "feature": the backspace key is not usually implemented in the input console but it is buffered. Thus, if you are typing input into your program and you use the backspace key, the character just entered is not removed from the buffer.

BIGGER Warning: SciTE running on a Linux machine has a super most annoying "feature" or more like a "huge problem" in my opinion: if you want to input into your program, you cannot do so from the "command window" in SciTE. You can have output displayed but you cannot read there – use a shell (like an xterm) to run your program so that you can get input from the shell.

If you are running a Windows machine, get the **SciTE executable full download** under http://www.scintilla.org/SciTEDownload.html and look under Windows to see listed the Windows Executables and get the **full download**. Download a zip file. A typical name for the zip file might be `wscite3400.zip` Once you have extracted all the files, you should have a directory such as `wscite` where you will find the file **SciTE.exe** which is the executable of the editor.

I suggest that you download the file zzz.properties to your home computer. I have put a copy on D2L, put it into the `wscite` directory.

There is a version of SciTE for OS X for the Mac but it needs to be purchased from the Mac App Store. (It's unlikely that you will be able to run C++ code from inside of SciTE). The author sends upgrades regularly. Yes, it costs, but then again, we hope that someone will pay for the software that **we** develop some day!

If you are on a Linux machine, download and install as explained in http://www.scintilla.org/SciTEDownload.html under **GTK+ / Linux**

If you are the system manager of your Linux PC, then you can install a file zzz.properties in the same directory as the other properties files, e.g. in /etc/scite. However, if you cannot write onto the directory that has all the properties files, then you can copy the contents of zzz.properties into `SciTEUser.properties` in the user's home directory so typically into `~/.SciTEUser.properties`

Read about the order of how the local and directory and user and global files are loaded in http://www.scintilla.org/SciTEDoc.html

If you want to convert the file that we have posted in D2L which was produced in Windows and has CR/LF to a file with single LF, try

```
tr -d '\15\32' < zzz.properties > ~/.SciTEUser.properties
```

THE EDITORS NOTEPAD++ SUBLIME TEXT TEXTMATE TEXTWRANGLER

Many students love Sublime Text: it is indeed a powerful editor but you will need to compile and run from the command shell, which is not a big deal. The official version does cost money but again, we do like supporting the software industry as we might be part of it one day – but then, there are free alternatives ….

Notepad++

Notepad++ is a viable option (one of my colleagues will laugh at you if you don't use it).

Editors for the Mac

TextMate which you can see under http://macromates.com is an editor for the Mac. They have various categories but at least one of the versions is free.

Another nice Mac editor is http://www.textwrangler.com/products/textwrangler and it does not cost.

IDES

You can develop your code using Eclipse or using Visual C++ or another IDE. Please set the flags of the compiler to `-pedantic -Wall -Wextra -std=c++11`

and do not have any extra #include files that are not standard. And provide a standalone makefile. We should be able to unzip your source code and Makefile from your assignment and simply say "make" at a Langara lab and we should be able to see the results of your assignment without having to edit any code.

Eclipse is used in industry so you might want to invest the time into learning how to use it.

Microsoft Visual C++

There is a freely available version of Microsoft Visual C++. It provides a good IDE with a text editor, compiler, linker and debugger. Do not use any of the features and extensions of the dialect Visual C++ but program in the standardized C++11 for all your assignments and labs. Make sure that there are no "#include"s in the source code that you submit that are not standard to C++.

DevC++

I have been told over the years by students that take CPSC 1160 that they find Dev-C++ (which also is based on the gcc compiler) easy to install and practical to use at home. Dev-C++ has forked to newer versions developed by Orwell http://orwelldevcpp.blogspot.ca/. If you go to the web bottom to **Download** you'll see how the **setup** includes the compiler!

Eclipse

Two other IDEs that you might consider are Eclipse CDT and Code::Blocks. The great advantage to Eclipse is that it has a debugger and Eclipse is used for many other programming languages. And it's free. Eclipse is installed at Langara. The disadvantage? You will have to spend time learning how to use it.

Please adjust the flags in Eclipse to be `-pedantic -Wall -Wextra -std=c++11`

C++ for the Apple Macintosh

You may develop your code on a Macintosh. You can use the IDE Xcode which is free for Mac OS X but often needs to be downloaded and installed (using the Mac App Store).

Note that if you do develop your code on a Mac, the submitted assignments and labs which are in C++ must compile in the labs at Langara using g++ with the already listed flags `-pedantic -Wall -Wextra -std=c++11.` There is also g++ for the Macintosh.

C++ COMPILERS AND LINKERS

TDM-GCC

We use the freely available (with a GNU license) "g++" as a compiler and linker. You can download and install a version easily for your Linux machine. If you are on a Windows machine, at Langara we have been using traditionally MinGW (see below). However, I highly recommend

http://tdm-gcc.tdragon.net/

which is a re-packaging of MinGW and related tools and it sets up the environment paths for you. There is a TDM64 bundle that seems to have tools to produce 64-bit executables even though the documentations says that the compiler itself is a 32bit application (thanks Brian Koehler for the discovery of TDM-GCC).

I highly recommend that if you install MinGW (be it from TDM-GCC or not), that you copy the command `mingw32-make.exe` into `make.exe` It is a good idea to use the standard "make" command so that you won't have to change your SciTE properties files and so that you can simply type "make" in your DOS command shell. The file `mingw32-make` is usually installed `in C:\MinGW\bin` so your (copied) file `make.exe` will also be in `C:\MinGW\bin`

MinGW which is described in http://www.mingw.org/wiki/MinGW as the "Minimalistic GNU for Windows which is a collection of freely available and freely distributable Windows specific header files and import libraries combined with GNU toolsets that allow one to produce native Windows programs that do not rely on any 3rd-party C runtime DLLs." (DLLs stands for dynamic linked libraries)

If you are on a Linux machine or on a Mac, you can install "g++" relatively easily.

The website that accompanies the CPSC 1155/CPSC 1160 textbook http://cs.armstrong.edu/liang/cpp3e/solution.html has under the section <>Download Software a list of compilers including Microsoft Visual C++ and the Borland C++ compilers and Dev-C++.