

# SY DE 121 – Digital Computation - Lab #1

**NOTE:** *You do not have to know how to program to complete this lab!*

The lab is designed for you to gain familiarity with the Bloodshed Dev-C++ programming environment, using the engineering undergraduate facilities. Although the lab is submitted, there will be no grade assigned to it. TAs will record that your submission was completed properly.

## OVERVIEW

- A. Complete the “Computing Background Questionnaire” and return it to your Lab TA.
- B. Log on to your Nexus account and follow the steps outlined in "Basic C++ Program".
- C. Have your TA sign your printed output.
- D. Email an attached copy of your modified C++ program to *sd121@engmail.uwaterloo.ca* and submit the signed hardcopy.

## **TASK B: Basic C++ Program**

This is your guide to entering and running a rudimentary C++ program on NEXUS. Before you launch Bloodshed Dev-C++, use the Windows Explorer Program to create a directory on your N:\ drive called SD121. This is where you should save your C++ files, documentation files, etc. Avoid putting the SD121 directory under your "desktop" path - otherwise you will be plagued with desktop icons whenever you run a C++ program. Do not save your files on the D drive as it is a temporary drive.

### **Steps:**

#### *1. Create directory*

Under your SD121 directory, create another directory called “Lab01”. It is recommended that you open a new directory for each of your labs.

#### *2. Start Bloodshed Dev-C++*

Go to the "Start" menu, then to “Programs”, then to "Programming", then to “Bloodshed Dev-C++” and finally to “Dev-C++”.

#### *3. Open Dev-C++ editor*

From the **File** menu select **New -> Project**. Select “Empty project”. Do not worry about the other project types on the list. Type in a name for the project – use Lab0101 to represent the first lab and the first part of the lab. Make sure that the path is set to your Lab01 directory.

You will see the Solution Explorer appear on the left hand side of the screen. This area allows you to navigate from one file to another. It should be empty for the moment.

#### *4. Adding a source file to the project.*

Right-click on “Lab0101” in the panel on the left hand side. Select **New File**. An “[\*] Untitled” window will appear in the right hand panel. Select **File -> Save As**, and name it “lab0101.cpp”.

#### *5. Edit your first C++ code*

Now, type the C++ program listed below in your “lab0101.cpp” file. Don’t worry about understanding the syntax – this will be covered in the course over the next couple of lectures. We are just trying to learn how to enter code, compile it and execute it for the time being.

English translation	C++ program
Include comments about program	//filename: lab0101.cpp // This program uses a loop // to calculate squares of // integers 3 to 9
Include standard Input/Output functions	#include <iostream> using namespace std;
The main ( ) function is contained by { ...}	int main( ) {
Define an integer i and set to 0	int i = 0;
FOR i starting at 3 and ending at 9, in steps of 1 print "i = " the integer value of i, and the square of i, then go to a new line.	for ( i = 3; i < 10; i++ ) { cout << "i = " << i << " squared = " << i * i << endl; }
When loop is done, comment displayed.	cout << "\nFun!\n";
The 'system' command instructs Windows to keep the console open. Without it, the console window will close as soon as the program finishes executing.	system("pause");
Return '0' to indicate that main() function is finished.	return 0; }

6. *Save your file*  
After typing in the above code, save your file (Ctrl-S is the shortcut for this).
7. *Compile and running your program*

Before running your program you have to compile it. For your information, and we will talk about this more in class on Thursday, compiling takes the C++ text that you entered and converts it to a format that the computer can understand (namely, machine code).

To compile the program select **Execute -> Compile**. This step will produce an executable file in your project folder. In order to run the .exe you may select **Execute -> Run**, or simply double click it.

Hint: you can ask the computer to compile and then run the compiled code all in one step by selecting **Execute -> Compile&Run**.

If there is any syntax error (not logical error) in the program you will be notified. If the compile and execution were both successful, you will see the output of your program in a separate black screen (a 'DOS' screen). In this case, you should see an output similar to the next figure.

There are ways to compile individual files and then build later – we will learn about these when we start to integrate multiple files into the same compile sequence (later!).

*lab0101.cpp program output*

```
i = 3 squared = 9
i = 4 squared = 16
i = 5 squared = 25
i = 6 squared = 36
i = 7 squared = 49
i = 8 squared = 64
i = 9 squared = 81

Fun!
Press any key continue...
```

8. *Make these changes*

- a) Change `< 10` to `<=10` and run the program again.
- b) Remove `<< endl` from the `cout` (output line) and re-run.
- c) Put back `<< endl`, and change `i++` to `i+=2` and rerun.
- d) In the FOR loop, change `i++` to `i+` and re-run. Note the error message, then fix.
- e) Remove the semi-colon from the `int i = 0` line. Note the error message, then fix.

9. *Explore Bloodshed Dev-C++*

You are welcome to spend a few minutes exploring the other menus, function keys, and "Help". Be careful not to destroy the Lab0101 project in the process.

10. *Go back to your original "lab0101.cpp"*

Change the statement "Fun!" to read "Printed by: *your first and last name student id*". Re-run the program.

11. *Create a header.cpp*

Open a new file by selecting **New** -> **Source File** from the **File** menu. Copy the following information in your header file. Replace 'XXX' with the correct information. Save your file as *header.cpp* for future use. Copy the text from header file and paste it at the top of your *lab0101.cpp* file.

*header.cpp*

```
//*****  
// Student Name: XXX  
// Student Number: XXX  
//  
// SD 121 Lab: XXX Assignment: XXX  
//  
// Filename:  
// Date submitted: XXX  
//  
// I hereby declare that this code, submitted  
// for credit for the course SD121, is a product  
// of my own efforts. This coded solution has  
// not been plagiarized from other sources and  
// has not been knowingly plagiarized by others.  
//  
// Signed,  
//  
//  
//*****
```

**IMPORTANT: Sign your disclaimer for all lab exercises submitted during SD121.**

## 12. Printing

Print your source code and its output.

**TASK C.** Remember to get your TA to sign your printed output and record that you have completed Tasks A and B.

**TASK D.** Email a copy of your source code to sd121@engmail using the format indicated below.

Use the following directions whenever you are submitting code via email for marking purposes to ensure that your source code is received and graded. Send an attached copy of the C++ source code (\*.cpp), *not* the executable (\*.exe) (you never have to submit any files other than source codes). You must follow this format – this will greatly assist the TAs when they are grading your papers (and you don't want to upset your TAs!!!!)

Ensure that your source code file is named using the following format:

- Filename begins with “lab” followed by
- the lab number (two digits) followed by
- the exercise number (two digits)

Example: lab0302.cpp (Homer Simpson's program for lab 3, question 02)

I. COMPOSE an email using the following information.

To: sd121@engmail.uwaterloo.ca

Subject: Lab ##, Exercise ##, StudentId, LastName, FirstName,

e.g. Lab 03, Exercise 02, 08123456, Simpson, Homer

Attach: your source code file

e.g. lab0102.cpp

The subject heading of the email MUST BE IN THE SPECIFIED ORDER and USE THE IDENTICAL SPACING to allow the TAs to sort the 100+ emails.

II. SEND the email to the course account by the required deadline. Keep in mind that the file must be included as an attachment.

Your signed copy of the lab should be submitted to the course submission box found in the hallway outside your classroom.

### Due Date

All material (email and hardcopy) is due by Friday, Sept. 16 by 5:00pm.

Note: Lab assignments are typically due the Friday after the lab session that week.