

## Contents

Visual Studio via MyApps on Seneca lab PCs.....	1
Run Visual Studio.....	2
Hello World .....	5
Troubleshooting .....	7
Locating VS repository and source file.....	8
Transfer, Compile, and Test on matrix .....	11
Connection Requirements.....	11
Methods of Connection .....	11
Transfer source via SFTP .....	12
Configure WinSCP .....	12
Transfer source via WinSCP .....	17
Compile and Run on matrix – GUI tools .....	19
Compile and Run on matrix – GUI tools .....	19

A good place to start for most things at Seneca is at <https://students.senecacollege.ca/>. The Wiki Index to Pages and the search feature are useful in accessing or finding many Seneca resources. Students must set up Microsoft Authenticator and enroll in MFA (Multi-Factor Authentication) – it is a bother, but it is secure. Do you bother to lock your car or front door?

## Visual Studio via MyApps on Seneca lab PCs

Seneca's computer lab and other PCs on campus launch Visual Studio from **myApps**. Use the desktop link or <https://myapps.senecacollege.ca/>. Sign on may be required:

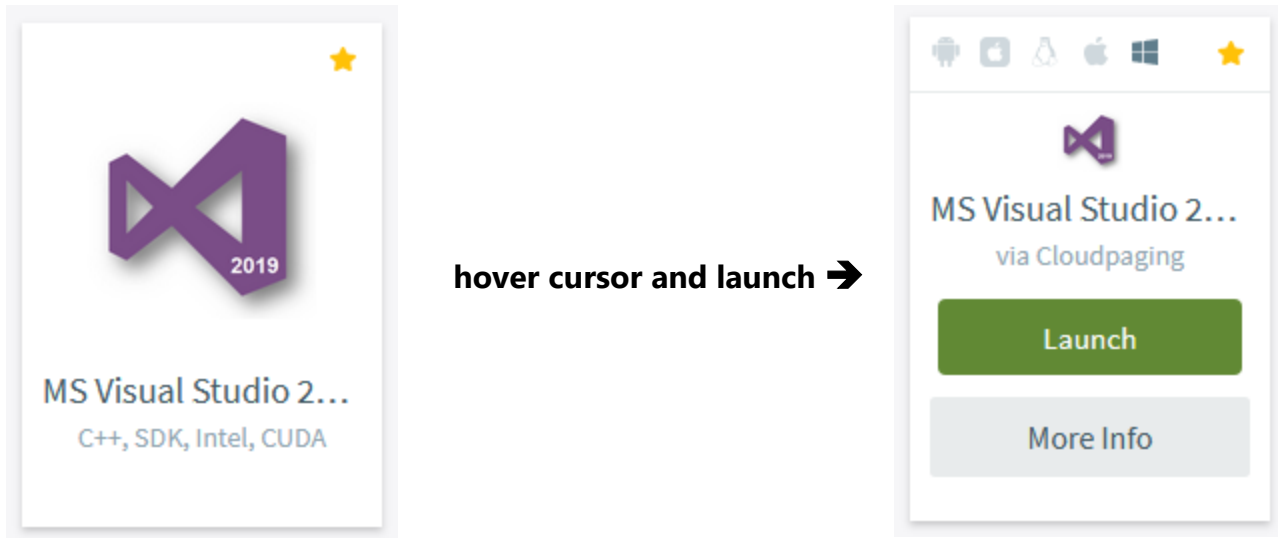


**Seneca MyApps**

Sign in via...



**Search for "visual" or scroll way down the list...**



#### *Troubleshooting myApps:*

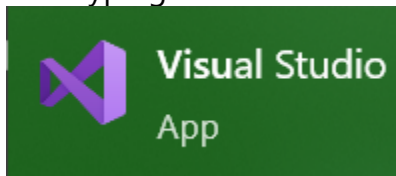
If you see "Validation Required" instead of "Launch", backtrack and sign on.

Be patient, the app...is...being...downloaded...from the cloud. After running out of patience, try reloading the page. If your UserID will not validate or the Visual Studio item itself needs validation, visit the ITS Service Desk in the Learning Commons: there may be a problem with your user account's permissions.


#### Run Visual Studio

**On a lab PC,** select the Visual Studio shortcut you see in File Explorer and press Enter.

**On your own PC:** Press the Windows key  or click the Windows icon in the lower left and start typing "Visual Studio" until you see



(click on this or press Enter to launch)

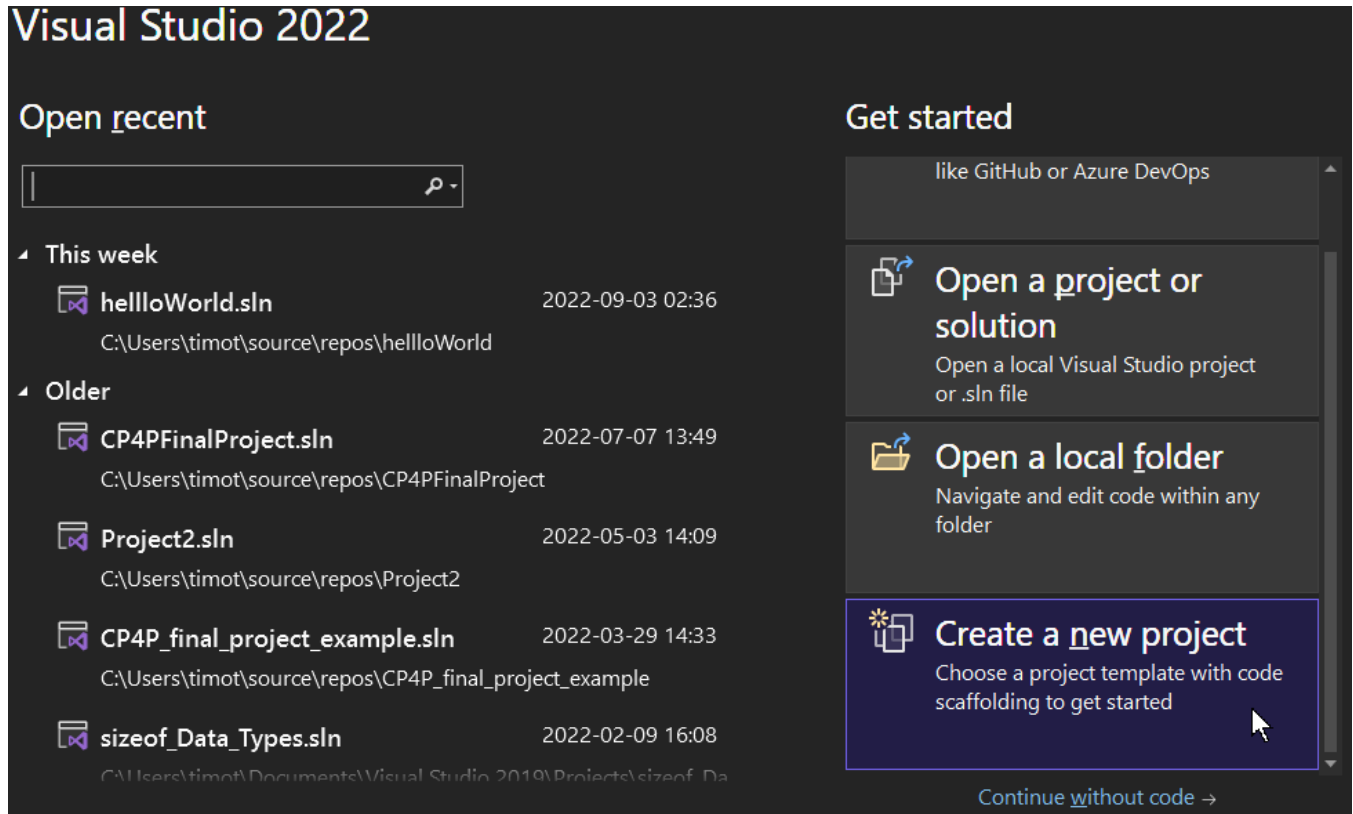
**Show these notes on one side of your screen: Windows key  + → [right arrow] then select Visual Studio from among the other open windows to display it on the other side.**

**Windows key  + up | down | left | right arrows** will help to organize one or more open windows.

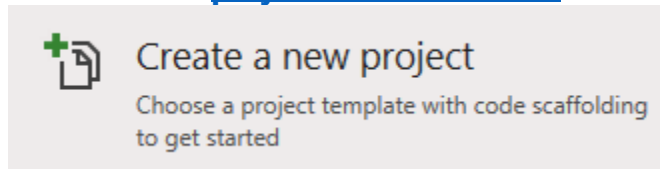
Because Visual Studio (VS) can manage various types of projects, it is one of the industry standards for systems development. As such, it is far more than just a programming code

editor. Thus, the next few steps require your careful attention to set up the VS project for the type of program we will be creating.

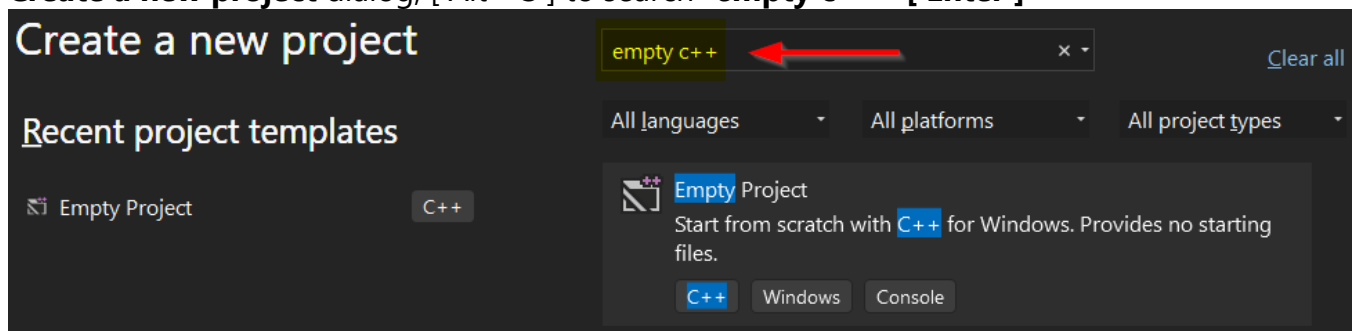
From the **Visual Studio, Get started window ...**



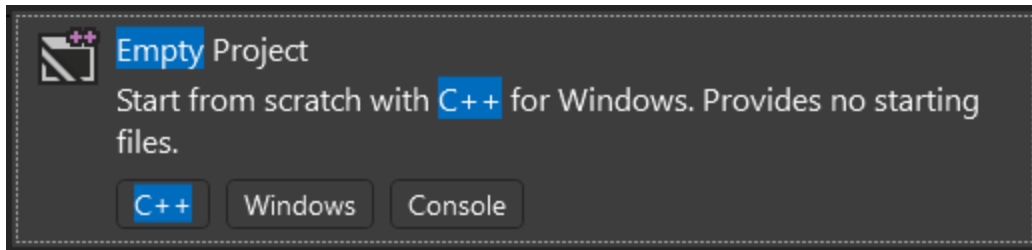
### Create a new [project in Visual Studio](#)



Create a new **project** dialog, [ Alt – S ] to search "empty c++" [ Enter ]

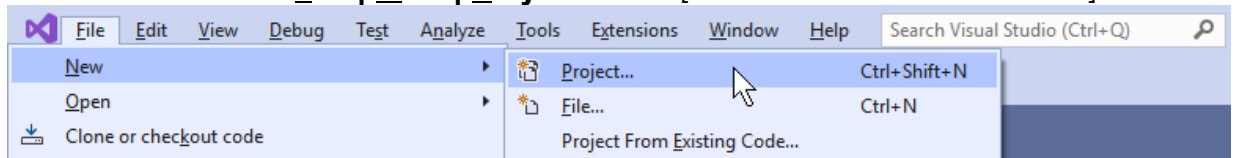


[ TAB ] to select



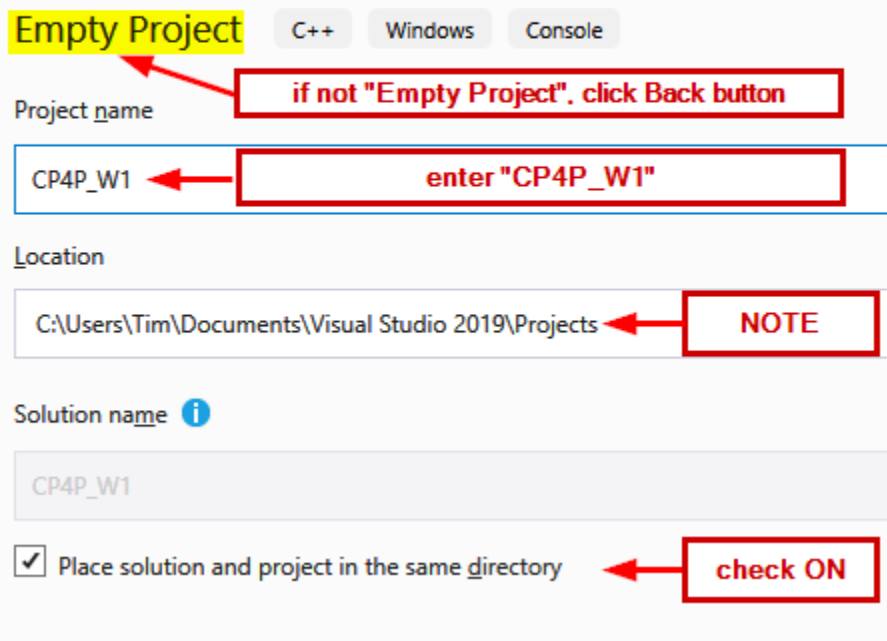
Or click on [Continue without code](#) → to get to the application menu

- use VS menu: select **File | New | Project** [ Alt + FNP or Ctrl+Shift+N ]



- Enter **CP4P\_W1** as the Project Name  
(Computer Principles *for* Programmers in Week 1)  
Note the Location (or override it).

## Configure your new project

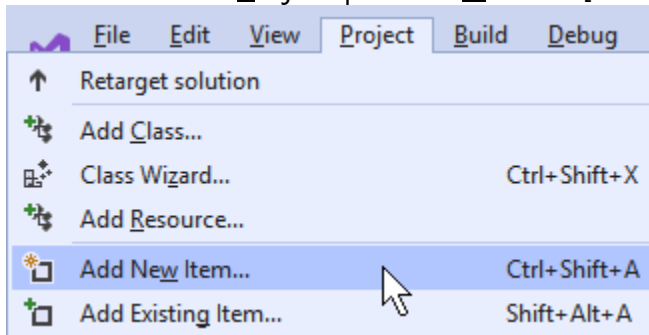
A screenshot of the 'Configure your new project' dialog box. It has a light gray background. At the top, it says 'Empty Project' in a yellow box, with 'C++', 'Windows', and 'Console' buttons to its right. Below this is the 'Project name' field, which contains 'CP4P\_W1'. A red arrow points from a red box containing the text 'if not "Empty Project". click Back button' to the 'Project name' field. Below the 'Project name' field is the 'Location' field, which contains 'C:\Users\Tim\Documents\Visual Studio 2019\Projects'. A red arrow points from a red box containing the text 'NOTE' to the 'Location' field. Below the 'Location' field is the 'Solution name' field, which contains 'CP4P\_W1'. At the bottom, there is a checkbox labeled 'Place solution and project in the same directory' which is checked. A red arrow points from a red box containing the text 'check ON' to the checkbox.

Create using [ Alt + C ] or [ TAB ] to the Create button and press Enter | space-bar

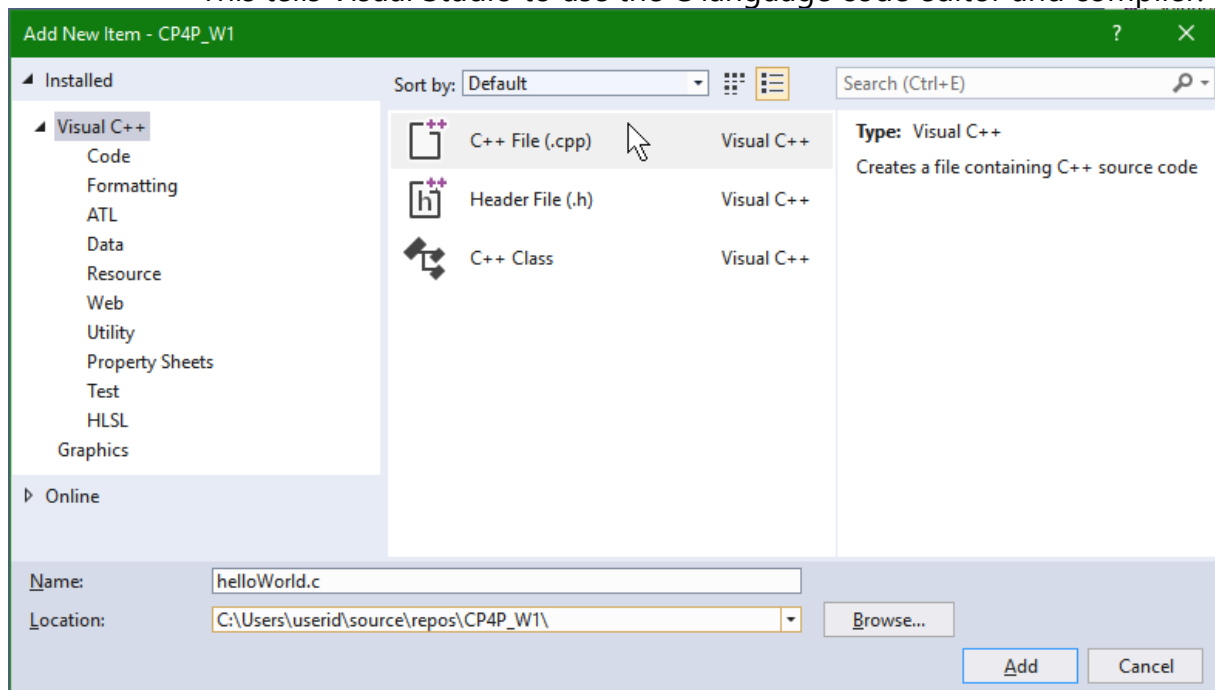
## Hello World

### Create a C language source code file...

- VS menu: select **Project | Add new Item** [ Ctrl+Shift+A ]



- Check that Visual C++ is selected on the left and **C++ File (.cpp)** in the center pane is also selected.  
[ Alt + N ] and enter **helloWorld.c** as the file Name, [ Alt + A ] to Add
  - Make sure the file extension is ".c", not the default .cpp** ( C++ )  
This tells Visual Studio to use the C language code editor and compiler.



- Copy & paste the source code for the classic first program, "Hello World".
  - ➔ Change **yourNameHere** and other unique identifiers to yours.
    - See [Hello World](#), [The History of 'Hello, World'](#), [The Hello World Collection](#)
- If the first line of comments is not wrapping,
  - VS menu: **E**dit > **A**dvanced > **W**ord **W**rap [ Alt + E V R ] or [ Ctrl + EW ]

```
/* Thanks to Brian Kernighan, 'Hello World' is the traditional first C program. It became
legendary with the publication of "The C Programming Language" by Kernighan and Ritchie (1978).
Now, Hello World is the canonical test of any programming language. */
/*
Full Name :      yourNameHere
Student ID#  :    123456789
UserID / Email :  UserID|username@mySeneca.ca
*/
#include <stdio.h> // C language module providing Input/Output facilities

int main(void)    // main() is automatically called to start a C program
{
    printf("Hello, World!\n");    // output greeting

    // Why I came to Seneca ...
    /* insert printf statements here using code completion */

    return 0; // return to operating system
}
```

Add code using IntelliSense, Microsoft's name for its code completion feature.

- Position the cursor on a new line above the return statement
- Type pr
  - Notice the code completion suggestions appear
  - Pressing [ TAB ] will complete printf
- Type ( then "
  - Notice how the matching paired characters ) " are inserted
- Type I am yourNameHere.\n
  - printf("I am yourNameHere.\n")
- press [ End ] key
- Type ; // output your name
  - printf("I am yourNameHere.\n"); // output your name

**➔ Use the above steps to add one or more printf statement(s) to answer, Why did you come to Seneca?**

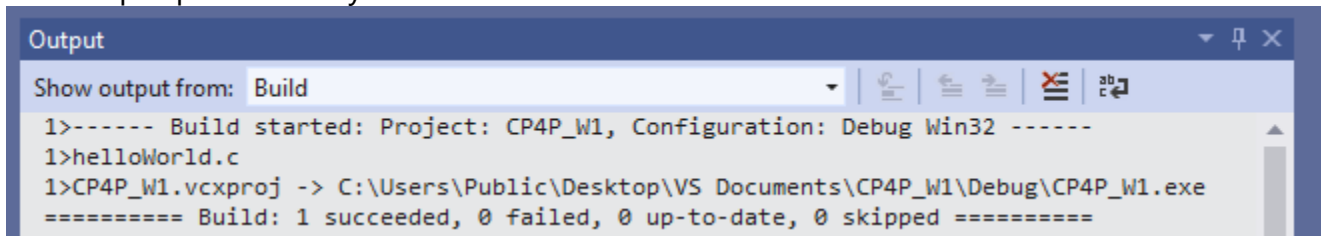
**Type as much text as you like between the double quote marks.**

- Save the source file [ Ctrl+S ]

Compile your C program...

- VS menu: select **B**uild | **B**uild **S**olution [ Alt + B Enter ], or [ F7 ], or [ Ctrl+Shift+B ]

The Output pane below your code should show **Build: 1 succeeded**

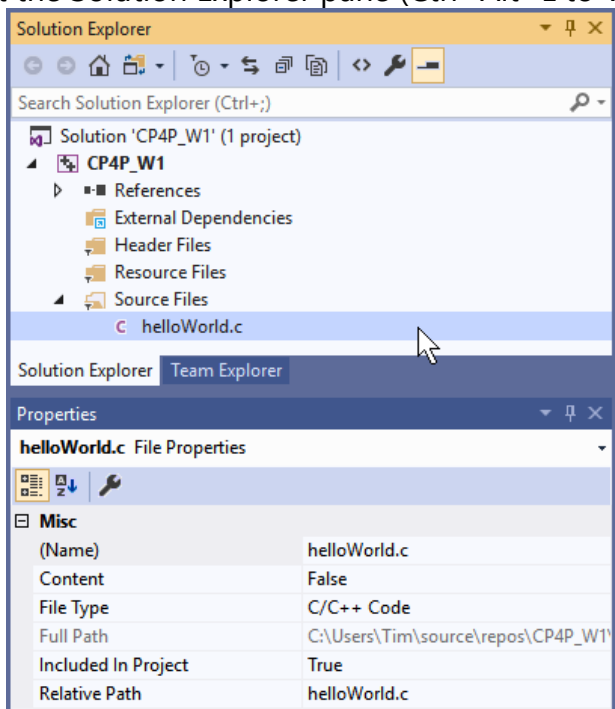


The screenshot shows the 'Output' window in Visual Studio. The 'Show output from:' dropdown is set to 'Build'. The output text is as follows:

```
1>----- Build started: Project: CP4P_W1, Configuration: Debug Win32 -----
1>helloWorld.c
1>CP4P_W1.vcxproj -> C:\Users\Public\Desktop\VS Documents\CP4P_W1\Debug\CP4P_W1.exe
===== Build: 1 succeeded, 0 failed, 0 up-to-date, 0 skipped =====
```

## Troubleshooting

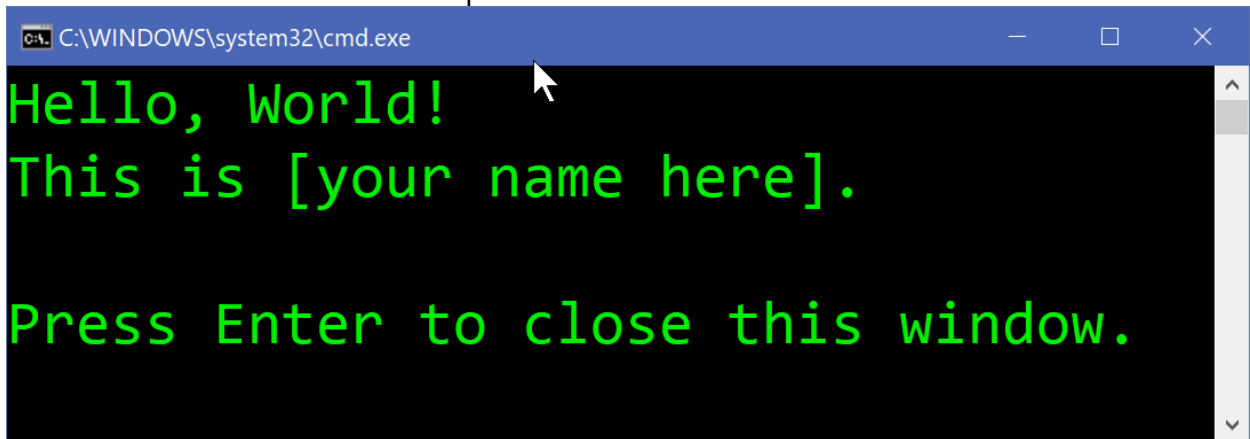
**If it failed** and your source code exactly matches the above, look at the Solution Explorer pane (Ctrl+Alt+L to View it)



- Only one source file should exist in the project
  - delete all except **helloWorld.c**
- Check that **helloWorld** ends with the extension **.c** and not **.cpp**
  - Use the Solution Explorer / Properties pane to rename the file to end in **.c**
- ...and run Build Solution again.
- If all else fails, it is best to start again.
  - Exit Visual Studio
  - Use File Explorer to find the CP4P\_W1 folder under ...\\repos\\
  - Delete the CP4P\_W1 folder
  - Then recreate the project as per the above notes.

**When the compile was successful**, run your program.

- VS menu: select **Debug | Start without Debugging** (Ctrl+F5)
- a terminal console window will open with



The appearance of the window varies.

Customize by clicking the icon in the top left / Properties.

- Close the window when you've sufficiently admired your work.

Microsoft has notes for [Getting Started with Visual Studio for C and C++ Development](#) which you can explore.

### Locating VS repository and source file


**Now, where is that helloWorld.c source file?**

You may have seen the full pathname when your program ran in the terminal window.

Source files are located under the VS Project name folder. And where is that?

There are several ways to find it.

You can start from the file's location and work your way up the folder structure:

- Press the Windows key  and type the filename **helloWorld.c**
- Right clicking on the filename, depending on your PC's configuration, may give you a list of options such as Open file location, Copy full path

You can use Visual Studio:

- Hover the mouse pointer over the **helloWorld.c** tab in the editing pane and the full path will appear.
- Right click on the **helloWorld.c** tab in the editing pane for more options.
- In the Solution Explorer pane, right click on the project name and select Open Folder in File Explorer and search from there.

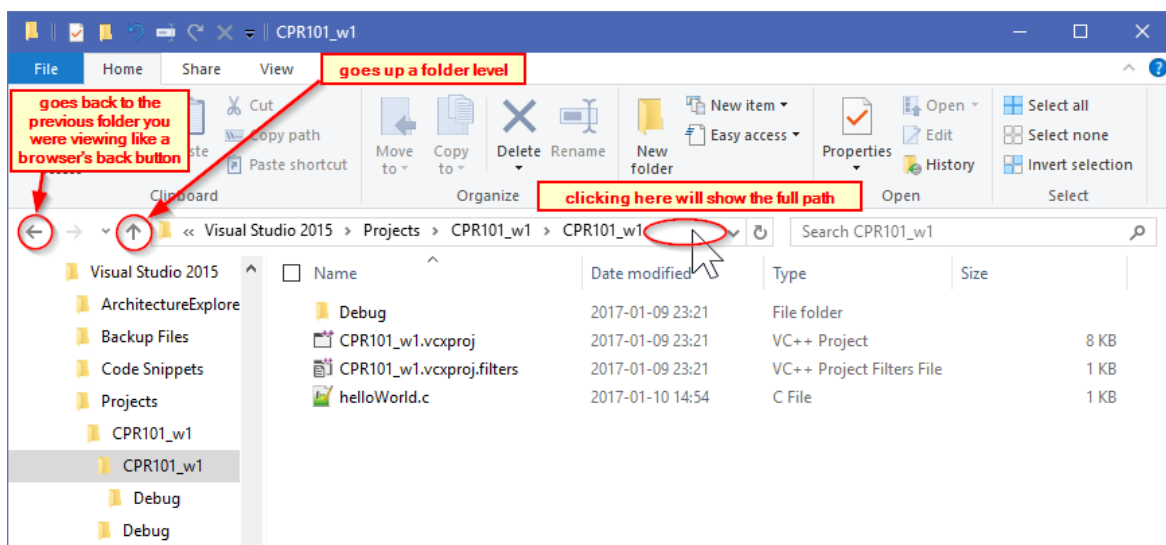
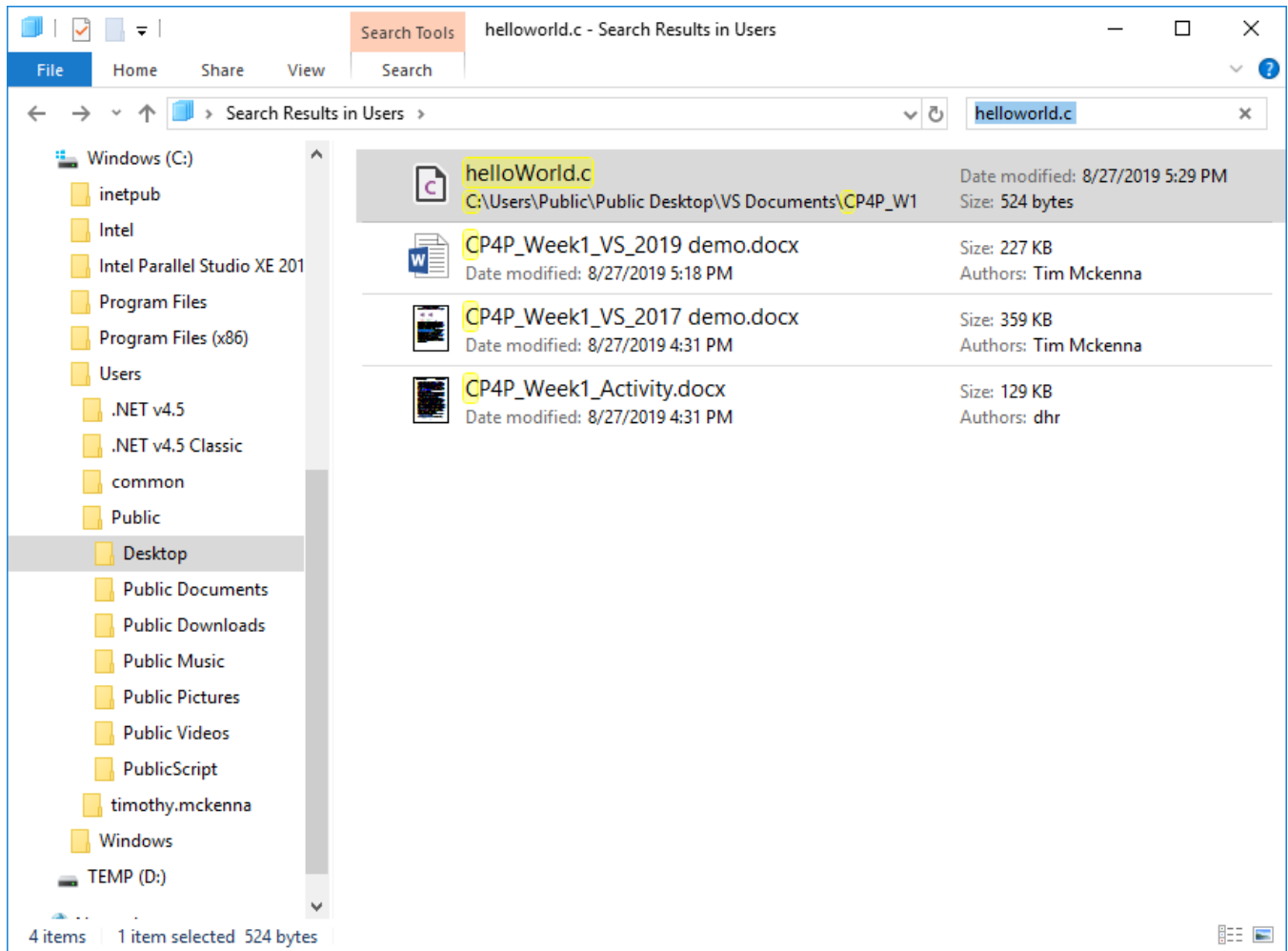
You can use the Windows File Explorer (  + E )



## Visual Studio Community — introductory demonstration

- File Explorer has a Search feature but if you search “This PC” for **file:helloWorld.c**, your patience will be tested. Before searching...
- Use the left pane to navigate to Windows (C:) and click on the Users folder
  - Then search for the helloWorld.c file

## Visual Studio Community — introductory demonstration



## Transfer, Compile, and Test on matrix

matrix.senecacollege.ca is a Linux [cluster](#) which is the host server for your completed C programs in IPC144.

### Connection Requirements

- install [GlobalProtect VPN](#) (Virtual Private Network).  
Being signed on to the VPN and using [multi-factor authentication](#) is necessary for access to many systems at Seneca including matrix.  
(Note the link for the VPN client to use if located in China.)
- *You must reconnect to the VPN every time you restart your computer and after periods of inactivity.* This will require Second-Factor Authentication (2FA, aka MFA).
- macOS users might save time by reading this first:  
<https://tech.wayne.edu/kb/security/wsu-virtual-private-network/500214>
- [Troubleshooting](#)

### Methods of Connection

**See the "Visual Studio Terminal cmd line" document for steps with command line utilities probably already on your computer. Be brave. Life is better without a mouse.**

- [SSH](#) / [SFTP](#) / [SCP](#) to matrix

Other methods follow below.

- install a Secure SHell Telnet (SSH) client which is a terminal app to sign on and use a Unix/Linux server like Matrix.
  - Windows – use [Windows Terminal](#) or install [PuTTY](#) using the .MSI Package
  - macOS -- use the built-in SSH Terminal (Finder, Go -> Utilities), or [other options](#)
  - Linux -- use the preinstalled [OpenSSHspan>](#) terminal, or [other options](#) including [PuTTY](#)
- install a Secure File Transfer Protocol (SFTP) GUI app for transferring files from your system to a server
  - Windows -- install [WinSCP](#). Its UI can launch PuTTY if it was previously installed.
  - macOS / Linux / Windows -- install [FileZilla](#) for a cross-platform app similar to WinSCP

Upload your C source code file to matrix via an SFTP application (Secure File Transfer Protocol) such as WinSCP or Filezilla. Next, compile and run the program on matrix using a terminal program with SSH (Secure SHell). Steps to do this follow below.

Host name is matrix.senecacollege.ca on port 22 for SFTP and SSH terminal.

Use your Seneca UserID and password as for other Seneca systems.

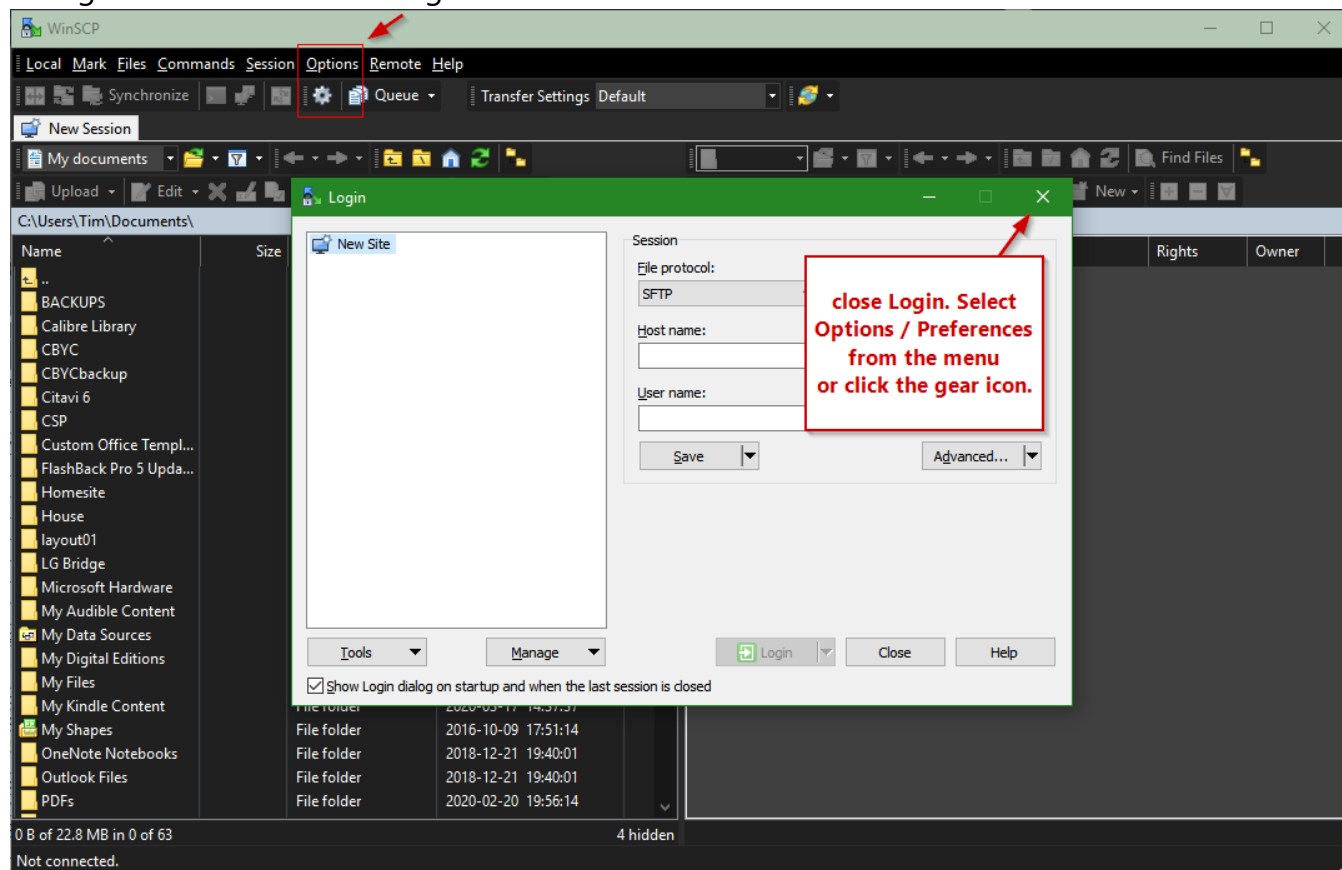
**N.B. ensure you input your UserID in lower case.** Unix/Linux/\*nix regards 'a' and 'A' as different characters in UserIDs, directory/file names, and on the command line unlike many other Operating Systems (OS) which are case independent in those instances. Passwords are *always* case sensitive in any OS.

### Transfer source via SFTP

Regardless of the SFTP client you use, it is CRITICAL that C source files are **Transferred in TEXT mode**. **Automatic** mode transfer works only if \*.c and \*.cpp file extensions are included in the FTP client's list of text file masks. There are small but important differences in the way different operating systems encode plain text files such as your .c source code files; use TEXT mode and FTP will take care of [End of Line](#) & [line termination](#) differences. Transferring files in Binary mode *ignores* all differences and will cause you grief.

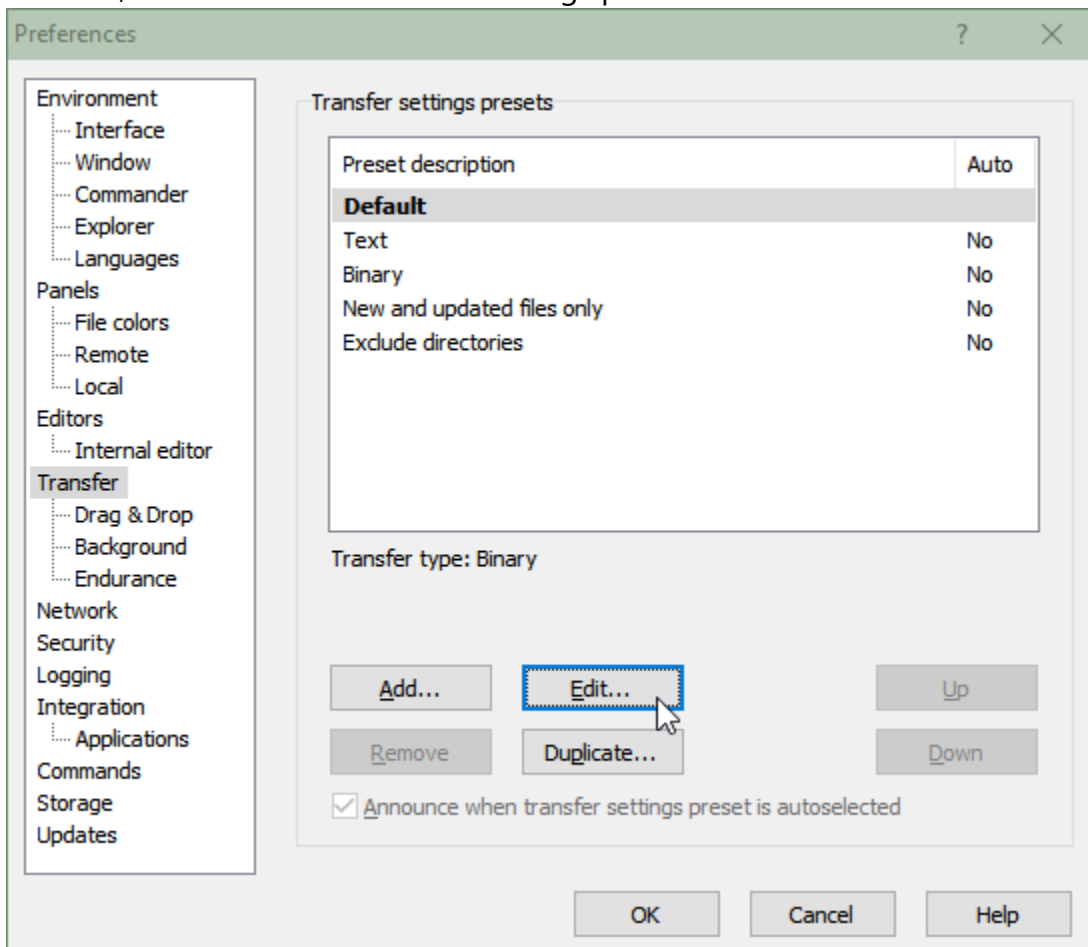
### Configure WinSCP

Starting WinSCP for the first time will show this. Close the Login dialog and make the following changes to the WinSCP settings...



### WinSCP Options / Preferences

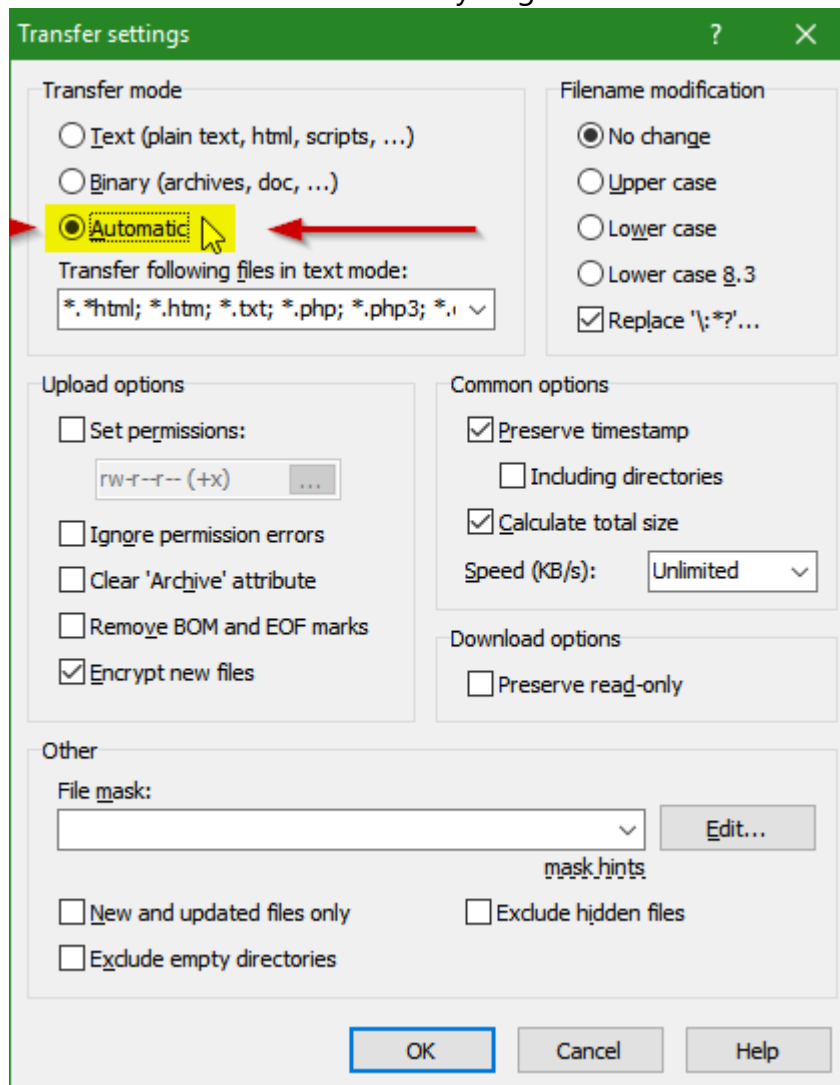
- Transfer, Edit **Default** in 'Transfer settings presets'



It is CRITICAL that you change the **Transfer** preset for **Default** to **Automatic**.

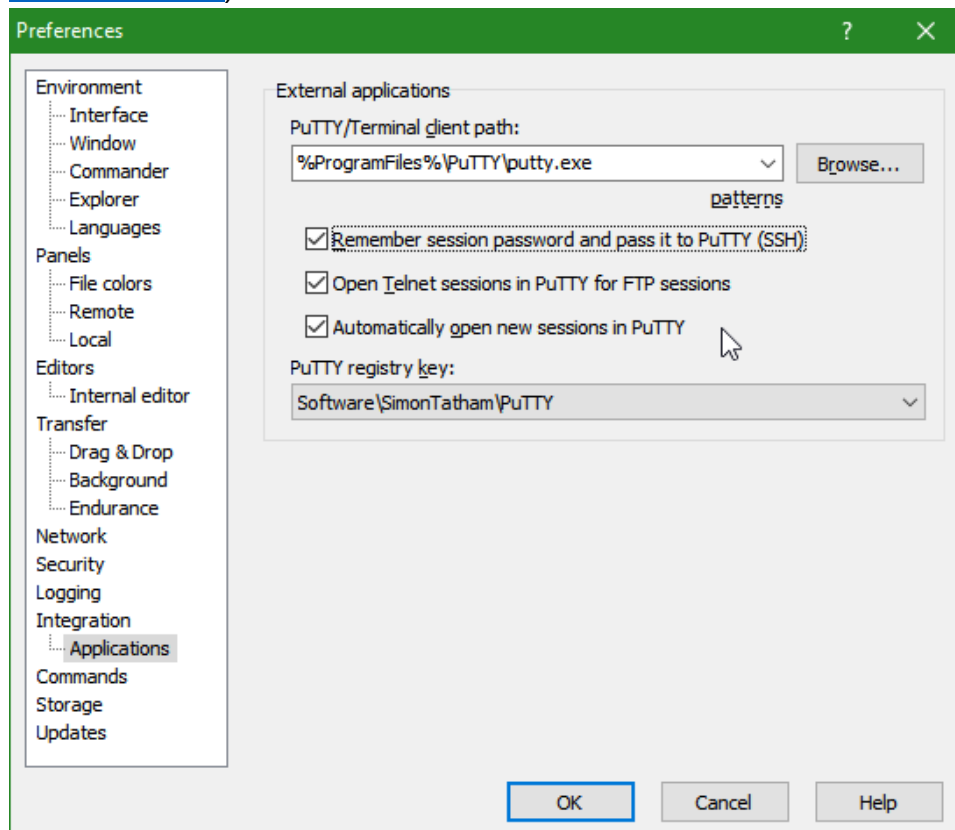
There are small but important differences in the way different operating systems encode plain text files such as your .c source code files. Transferring files in Binary mode *ignores*

those differences and will cause you grief.

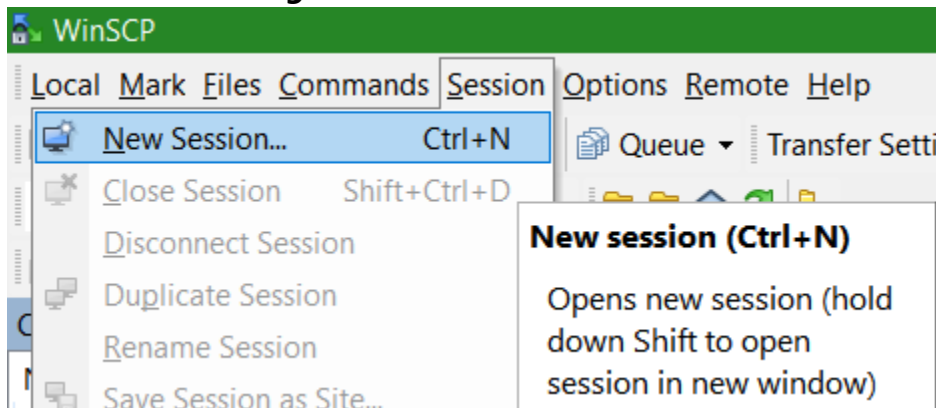


- WinSCP can open a PuTTY terminal window automatically when you open an SFTP session to transfer a source file. This will save you a step later. Change the following settings in Preferences / Integration / Applications. ([WinSCP Integration with PuTTY](#))

[documentation](#))

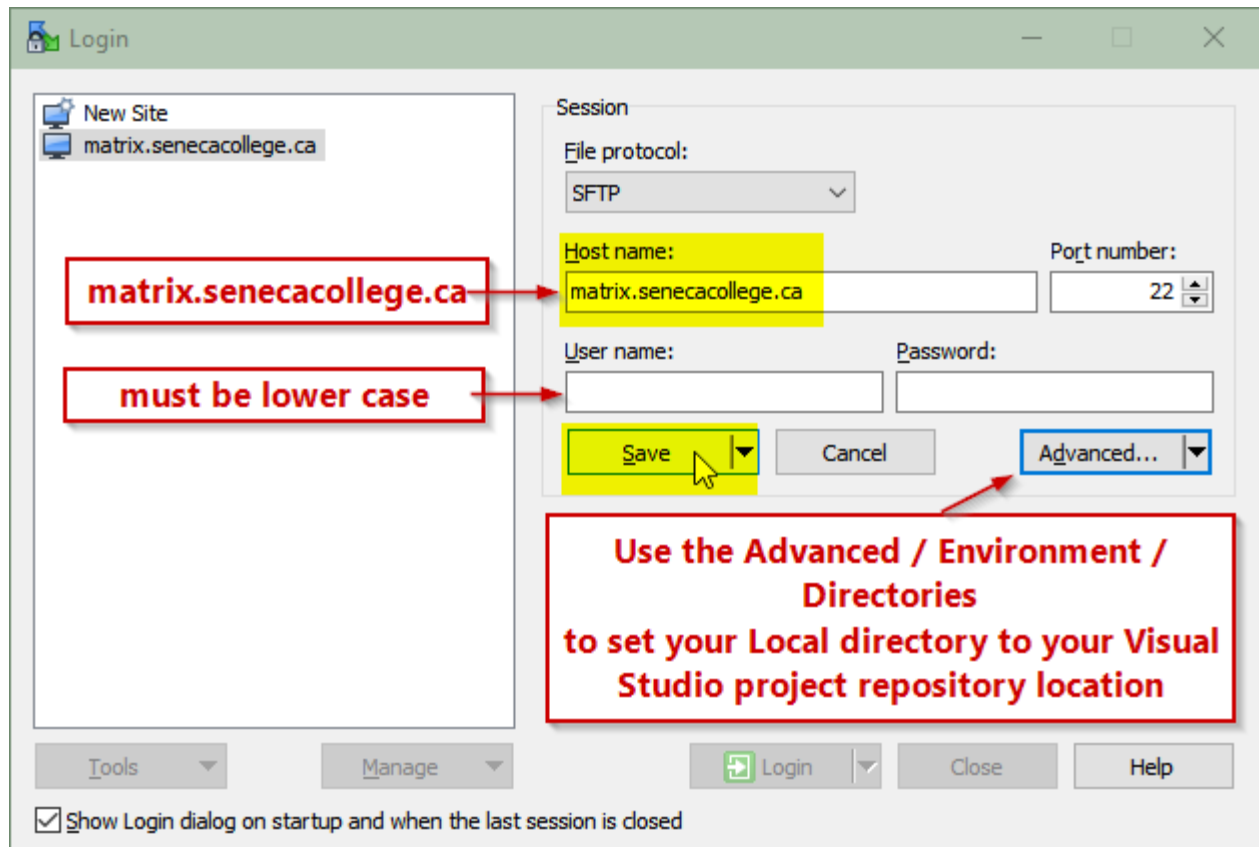


- Set up a New Session in the Site Manager for the Seneca server at **matrix.senecacollege.ca**



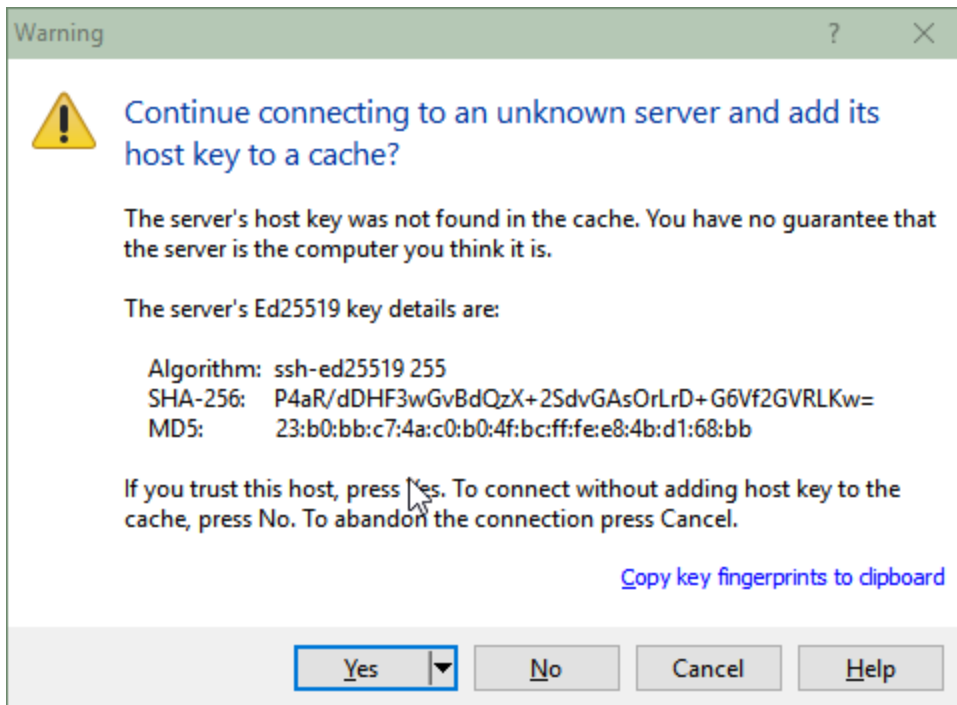
Ensure your Username is input in lower case.

## Visual Studio Community — introductory demonstration



- Save the Session
- Ensure your [Student VPN](#) is installed and you are signed in.
- Select **matrix.senecacollege.ca** then click the Login button. You will see the following the first time you contact matrix. Accept the host key.

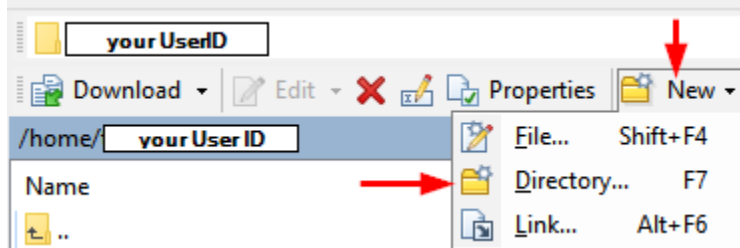




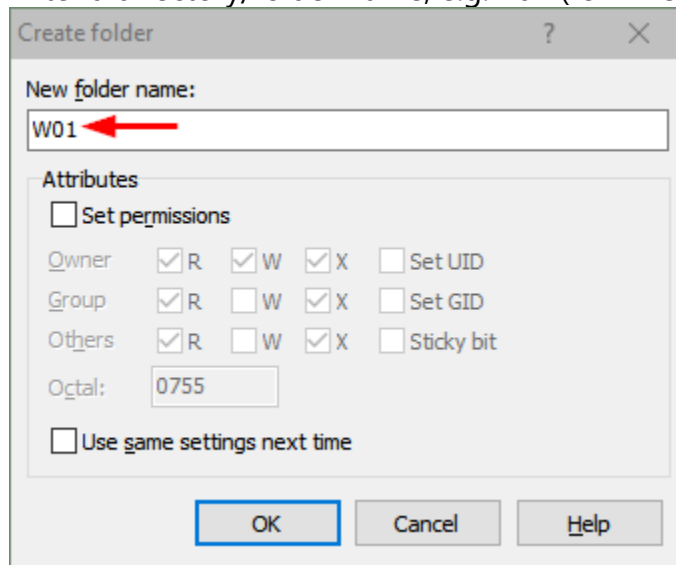
### Transfer source via WinSCP

- From this point onward, you can click and drag your .c source file from the left to the right side. Consider first creating a directory on matrix to contain your .c files.

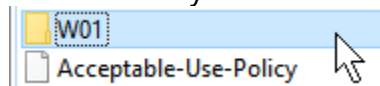
In the right-hand panel, click New, then Directory, and



Enter a directory/folder name, e.g. w01 (for "Week 1")

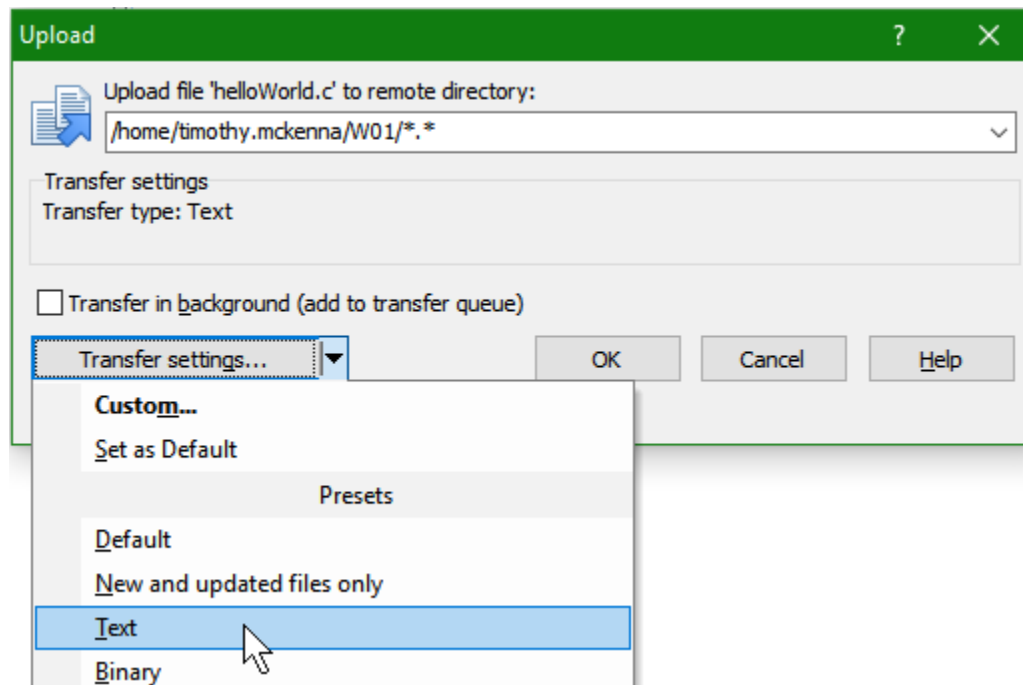


Double click your new directory/folder name to make it your current directory/folder



Find your helloWorld.c source file using the left panel, click on your source file and drag/drop it to the right panel. This will transfer your source file from your local computer to the directory named w01 on the matrix server.

*Make sure the files are transferred in Text mode, **not** binary. See above notes on this.*



## Visual Studio Community — introductory demonstration

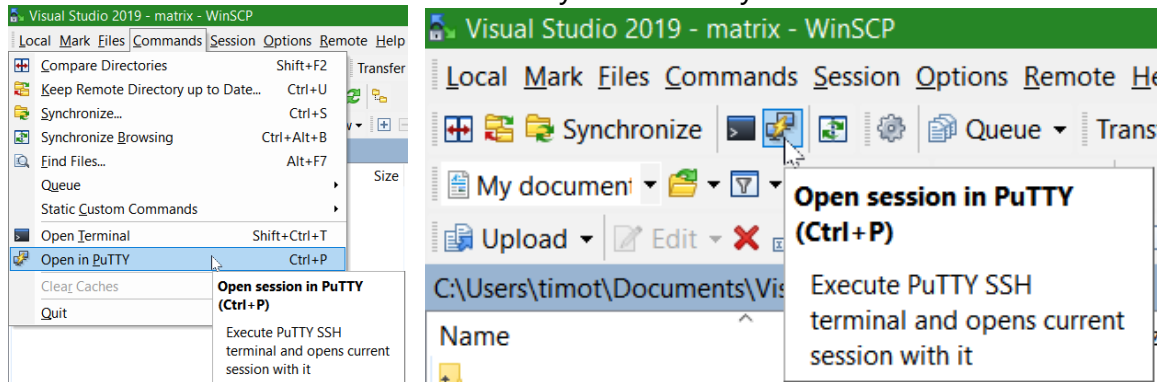
Double click on the file in the right-hand panel to see the file's contents. Close that window.

### Compile and Run on matrix – GUI tools

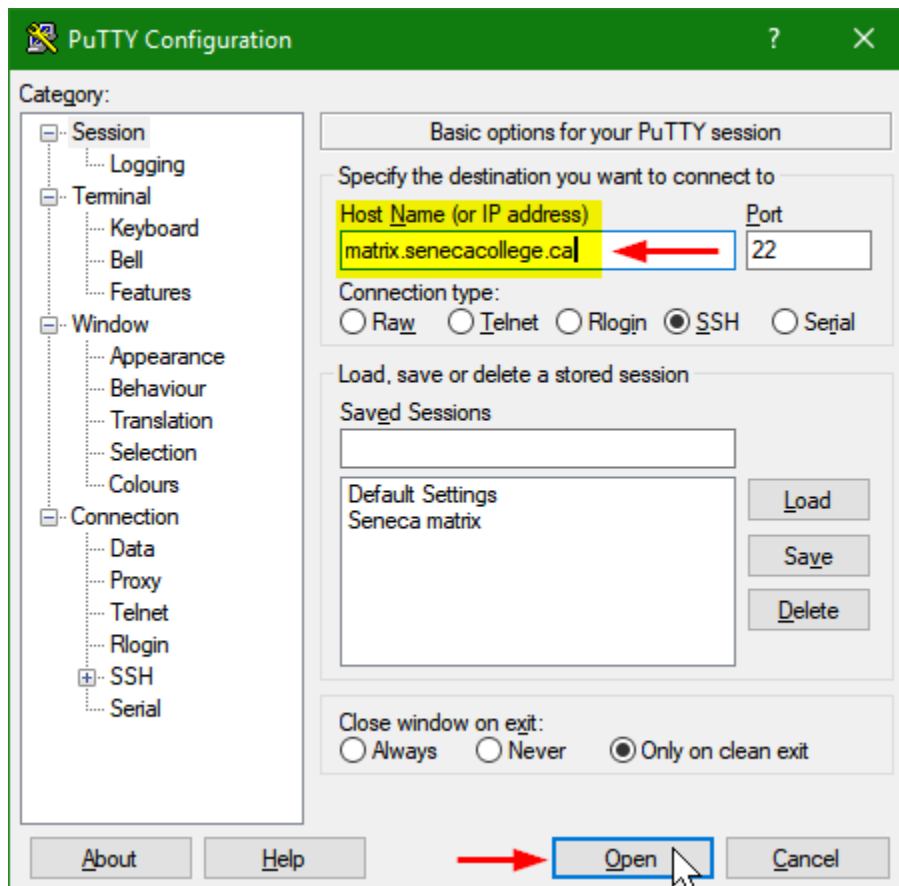
Compile and run your program on matrix using a traditional telnet terminal and command line interface.

### Compile and Run on matrix – GUI tools

Launch PuTTY if WinSCP has not already done it for you.



See the [PuTTY documentation](#) for details.



## Visual Studio Community — introductory demonstration

To customize PuTTY for the " WinSCP temporary session", launch PuTTY, open that Session, change settings in the Category tree (Window / Appearance / Font is a favourite to be changed), and remember to return to Session and Save.

```
login as: yourUserID
#####
# Welcome to Matrix
<snip>
# All activities on this system are governed by
# Seneca Information Technology Acceptable Use Policy
# For complete ITAU policy visit http://www.senecacollege.ca/policies/itau.html
#####
yourUserID@matrix.senecacollege.ca's password: yourPassword
Last login: Thu Sep 13 17:42:46 2018 from 174.93.86.94
[yourUserID@mtrx-node02pd ~]$ cd w01 change to the directory containing your source file
[yourUserID@mtrx-node02pd w01]$ ls List the current directory-is your source file there?
helloWorld.c
[yourUserID@mtrx-node02pd w01]$ gcc helloWorld.c -o helloWorld compile your source file
with the gnu c compiler
[yourUserID@mtrx-node02pd w01]$ helloWorld run your program
Hello, World!
This is yourNameHere.
[yourUserID@mtrx-node02pd w01]$ logout sign off
```