How to install NetBeans in Windows

(Updated Mar25, 2024)

Note: Don't run the NetBeans installer before both Java and Cygwin are installed on your system.

1. **Install Java SE 13 (JDK)**
   Although newer versions of the JDK are available, NetBeans requires a file included in versions 13 *and earlier* for the installation of the C/C++ plugin[1].

   a. Follow this link to download Java SE 13:
      https://www.oracle.com/java/technologies/javase/jdk13-archive-downloads.html

      You can also search for '**java jdk 13**' in a web browser, if the link above does not work.

   b. Select the **Windows x64 Installer** option for JDK 13.0.2 (scroll down the page to reach this spot)[2]. Click the link on the right side of this option to download it.
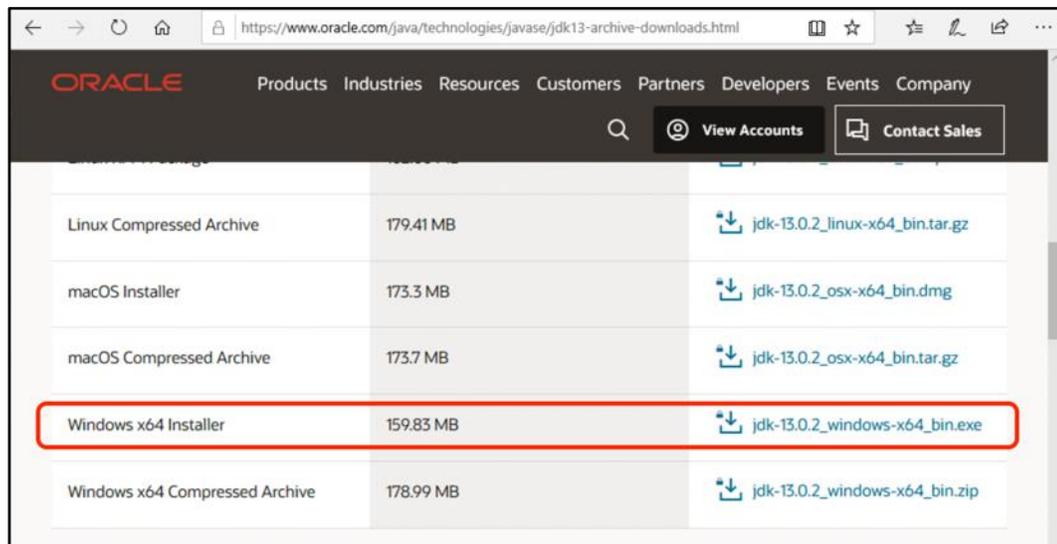
      You may need to create an Oracle User account to download this software. If so, you can use your school email account and address when setting up your account:

      Riverside City College
      4800 Magnolia Avenue
      Riverside, CA 92506
      (951) 222-8000

---

[1] It *is* possible to get C/C++ working in NetBeans using a JDK version greater than 13, but extra work is required, and no features of these newer versions are needed for the C/C++ plugin to work.
If you are interested in getting JDK version 14 or greater working with NetBeans, a separate document called "*How to fix NetBeans and Java incompatibility*" shows a way to do this.
[2] JDK versions 13.0.1 or lower should also work.

c. After downloading, double-click the downloaded file (likely in your **Downloads** folder) and follow the installation instructions[3]. Leave default settings from the installer as they are.

2. **Install Cygwin**

a. Open the web page: https://www.cygwin.com/ .

b. Scroll down the page to find the link to '**setup-x86_64.exe**'. Click it to download the installer.
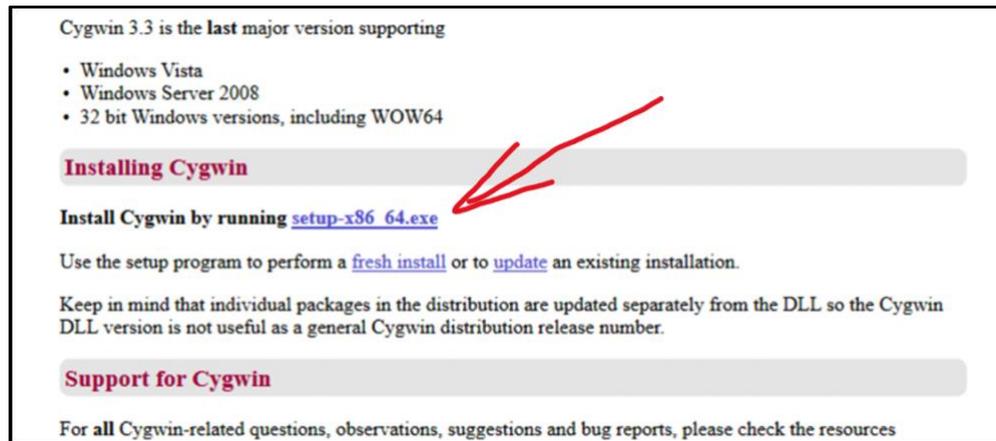
c. Run '**setup-x86_64.exe**' to begin installing Cygwin[4].
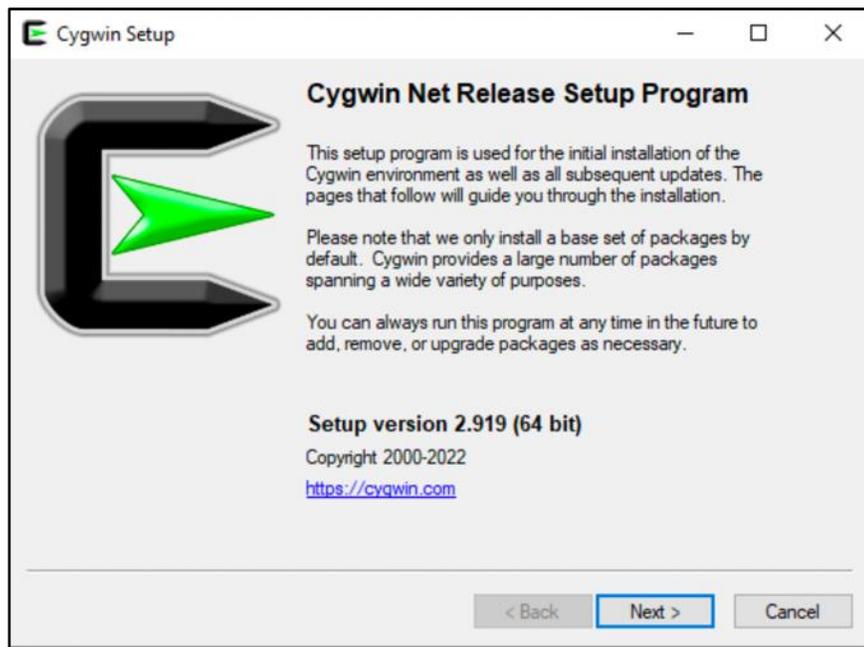
Press **Next** for the next few steps until you reach list of packages. Leave all default settings as they are:

---

[3] If a window pops up asking if you want to allow Java to make changes to the device (computer), click the **Yes** button.
[4] Again, there may be a popup window asking to allow Cygwin to make changes on your system, click **Yes**.

Cygwin Setup - Choose Installation Type

Choose A Download Source
Choose whether to install or download from the internet, or install from files in a local directory.

- ◉ Install from Internet
  (downloaded files will be kept for future re-use)

- ○ Download Without Installing

- ○ Install from Local Directory

< Back   Next >   Cancel

Cygwin Setup - Choose Installation Directory

Select Root Install Directory
Select the directory where you want to install Cygwin. Also choose a few installation parameters.

Root Directory

`C:\cygwin64`   Browse...

Install For

◉ All Users (RECOMMENDED)
Cygwin will be available to all users of the system.

○ Just Me
Cygwin will still be available to all users, but Desktop Icons, Cygwin Menu Entries, and important Installer information are only available to the current user. Only select this if you lack Administrator privileges or if you have specific needs.

< Back   Next >   Cancel

d. You may change the local package directory to another location to save the installation files to. If you change to a nonexistent directory, you will be asked if you want to create it.
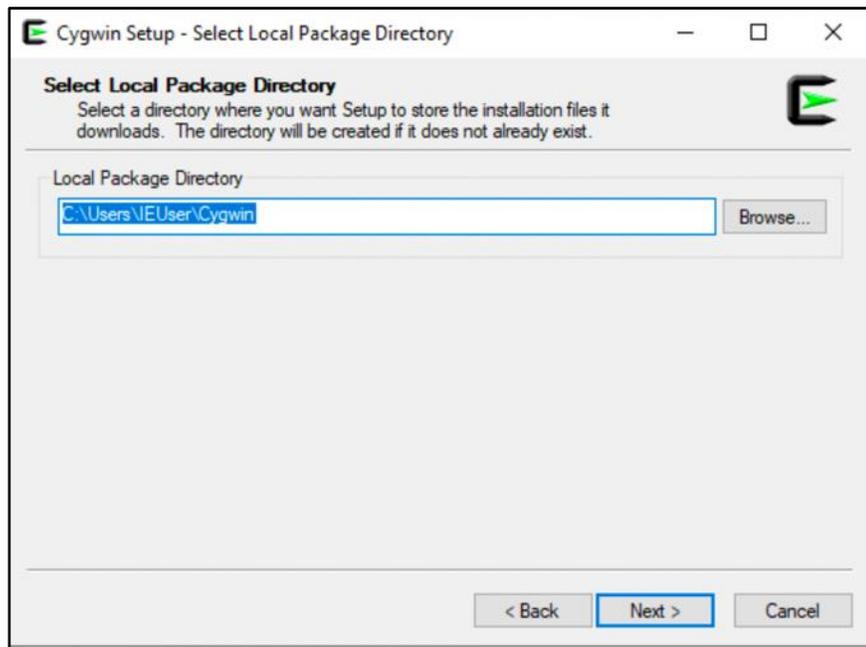
e. Keep your internet connection as '**Use System Proxy Settings**':
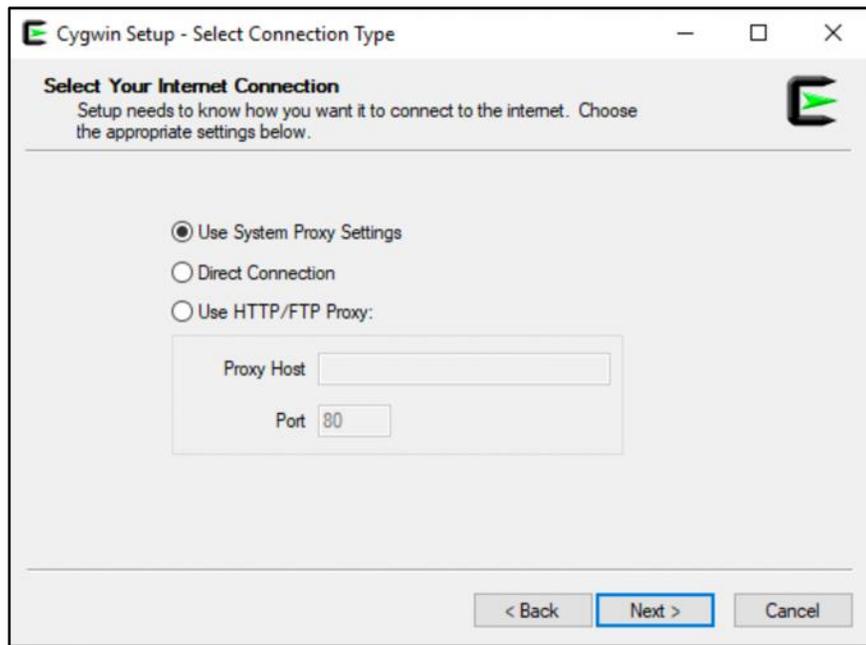
f. Choose a download site. It does not matter which one.
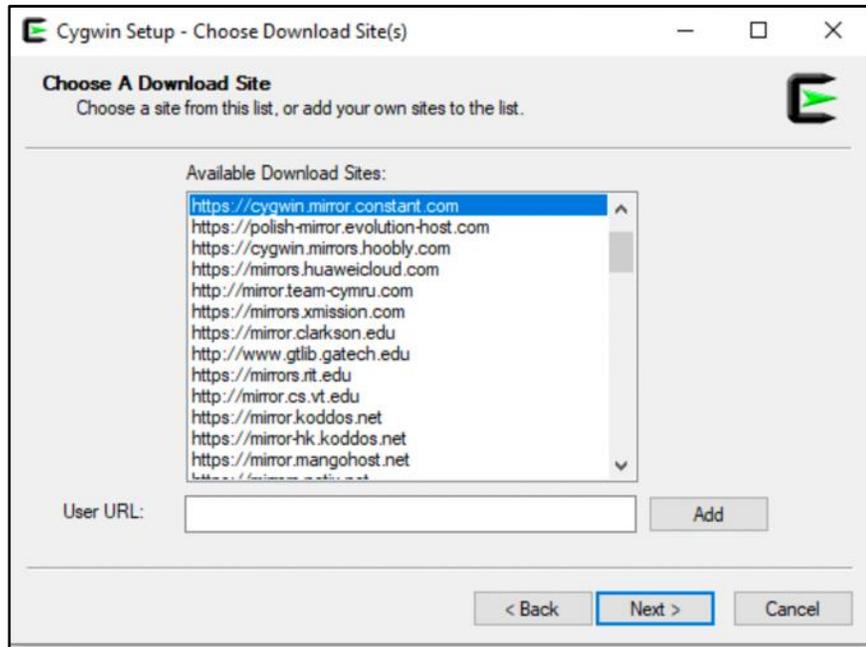
g. In the **Select Packages** window, change the **View** dropdown list in the upper left corner to **Full**.
Then in the **Search** textbox, enter **gcc**.
The list of packages will be filtered to only show items with 'gcc' in their names.
Find the packages called **gcc-core** and **gcc-g++.** For each of these, click the dropdown arrow button under the **New** column and select the highest *non-test* version in the list.
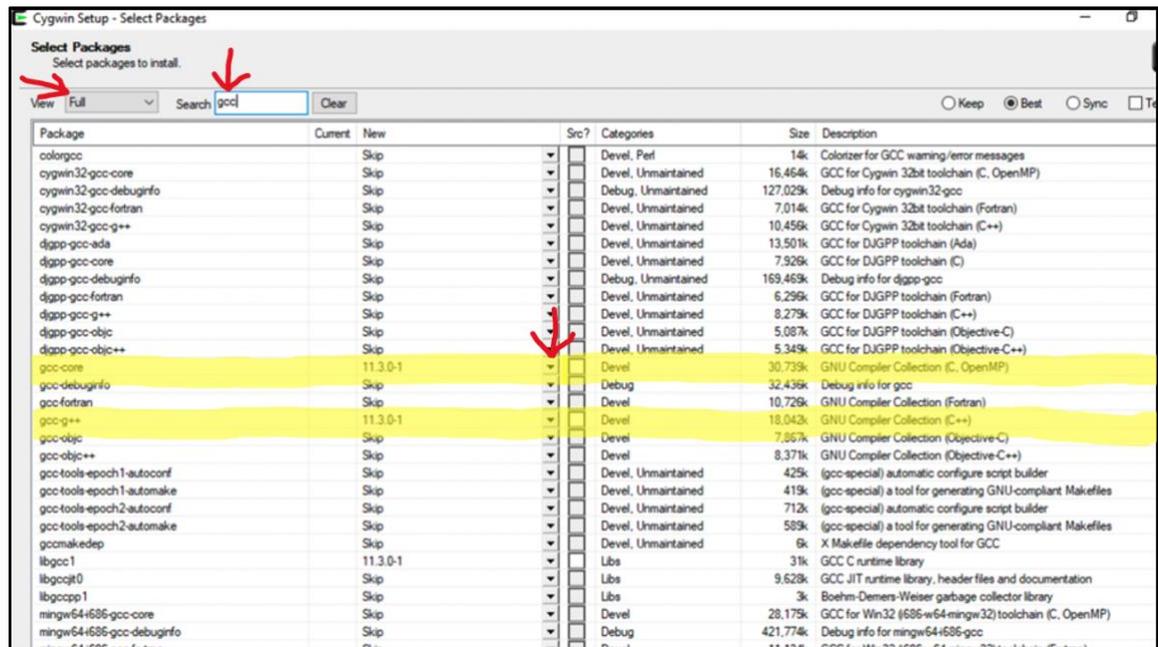
h. Now in the same window, enter **gdb** in the **Search** textbox and find package name **gdb**.
Click the dropdown arrow button in the **New** column for **gdb** and _select version 9.2-1_. **Do not select a higher version[5]**.
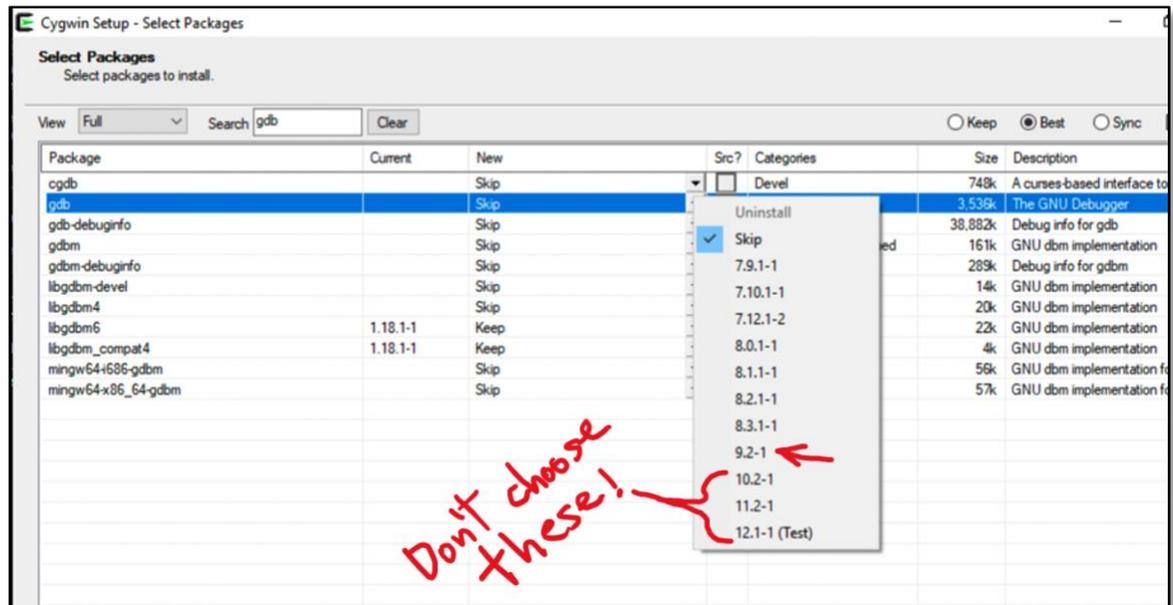
i. Still in the same window (**Select Packages**), enter **make** in the **Search** textbox. Find the **make** entry in the package list. Click the dropdown arrow button in the **New** column for **make** and select the highest non-test version.

---
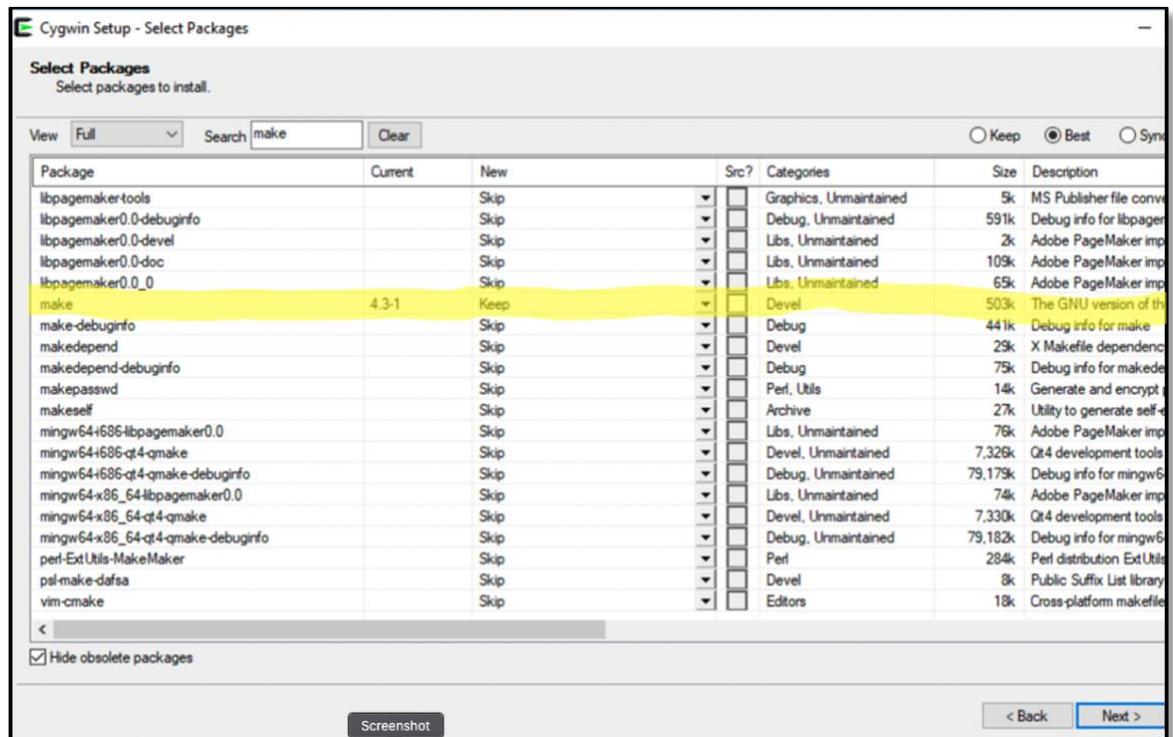
[5] Versions above 9.2-1 do not currently work with the NetBeans C/C++ plugin. **gdb** is the debugger for C/C++.

j. Click **Next**, now that all four packages are selected (**gcc-core**, **gcc-g++**, **gdb**, and **make**). The next window should list all packages that will be installed[6]. After viewing the list, click **Next** and download should start. This may take a few minutes.
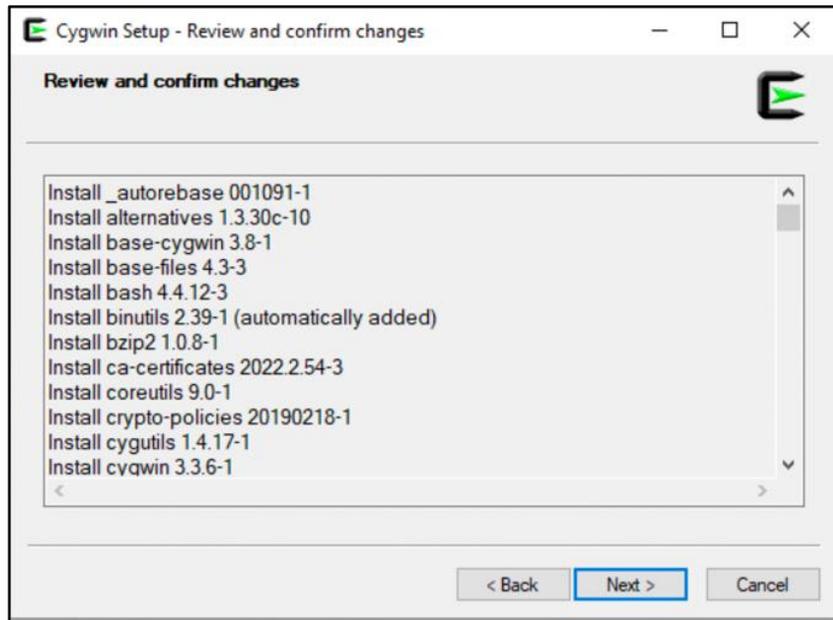
k. After installation is complete, click **Finish** to exit the installer.

3. **Install NetBeans**
    a. Open the web page https://netbeans.apache.org/download/ .
        Go to the **NetBeans** download page by clicking one of the **Download** buttons[7]:

---
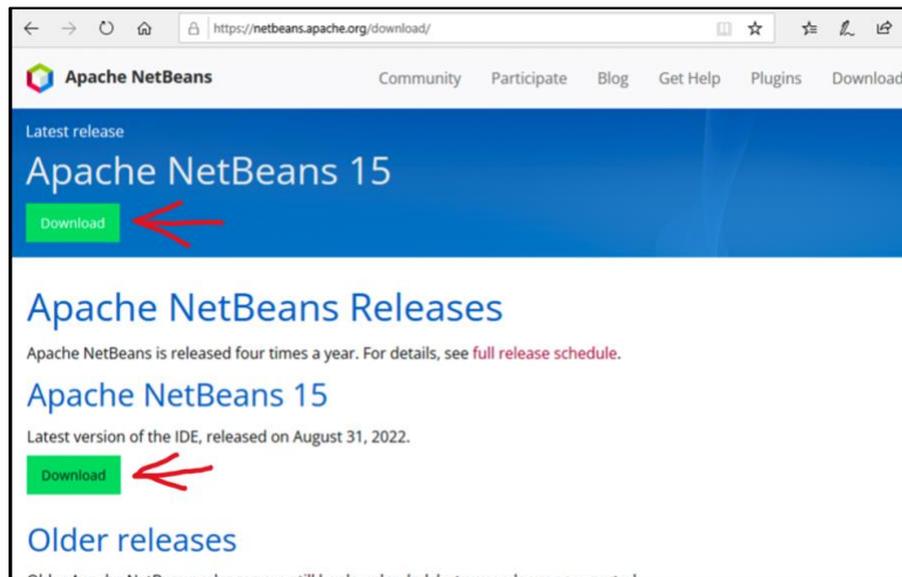
[6] If this is the first time Cygwin is being installed and configured on the system, several more packages will be installed besides the four specifically selected. These include the base Cygwin system.

[7] As of this writing, NetBeans 21 is available. A recent NetBeans version should work fine – but NetBeans 15 has been more extensively tested with the setup described in this document. If needed, you can download past versions by clicking the button under "Older releases" on the page.

b. In the next page, make sure to download the Windows 64-bit version of NetBeans:
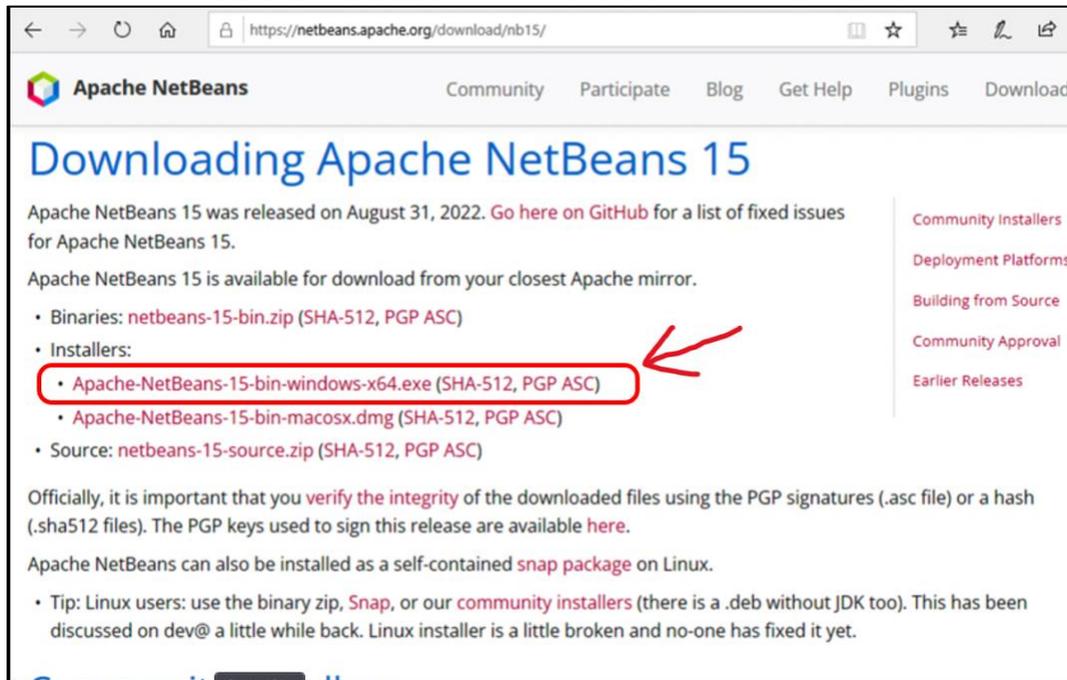
c. Now go to your **Downloads** folder (or wherever you had NetBeans downloaded to) and double-click the NetBeans installer file to run it (e.g. **Apache-NetBeans-15-bin-windows-x64.exe**)[8]. Click the **Next** button on the NetBeans installer window.

---
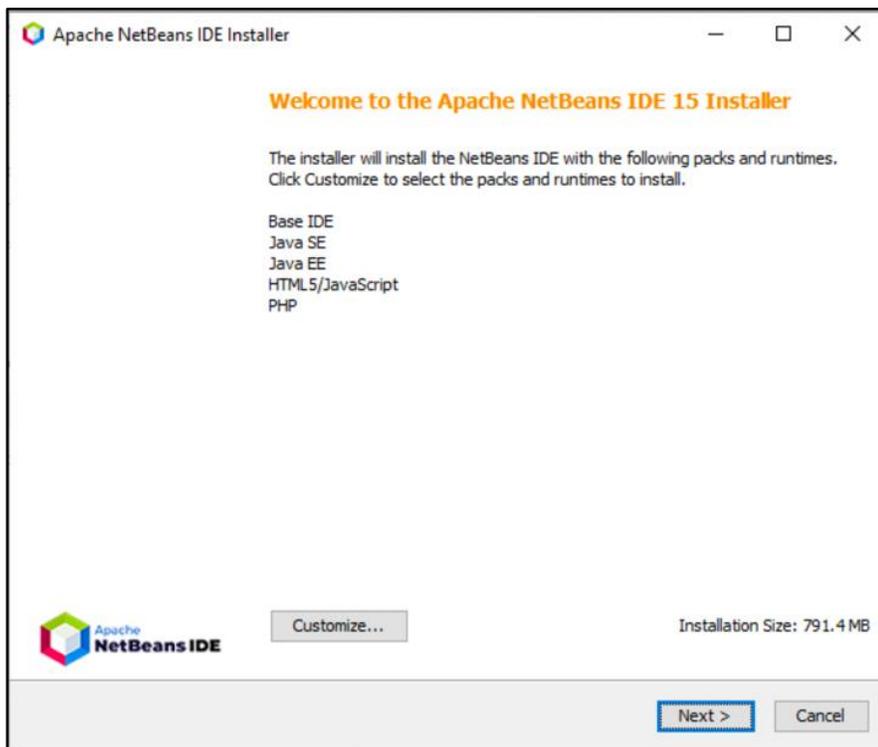
[8] Once again, if a window pops up saying NetBeans wants to make changes to your computer, click the **Yes** button to allow it.

d. In the License Agreement window, click the checkbox to accept the terms. Then click **Next**.
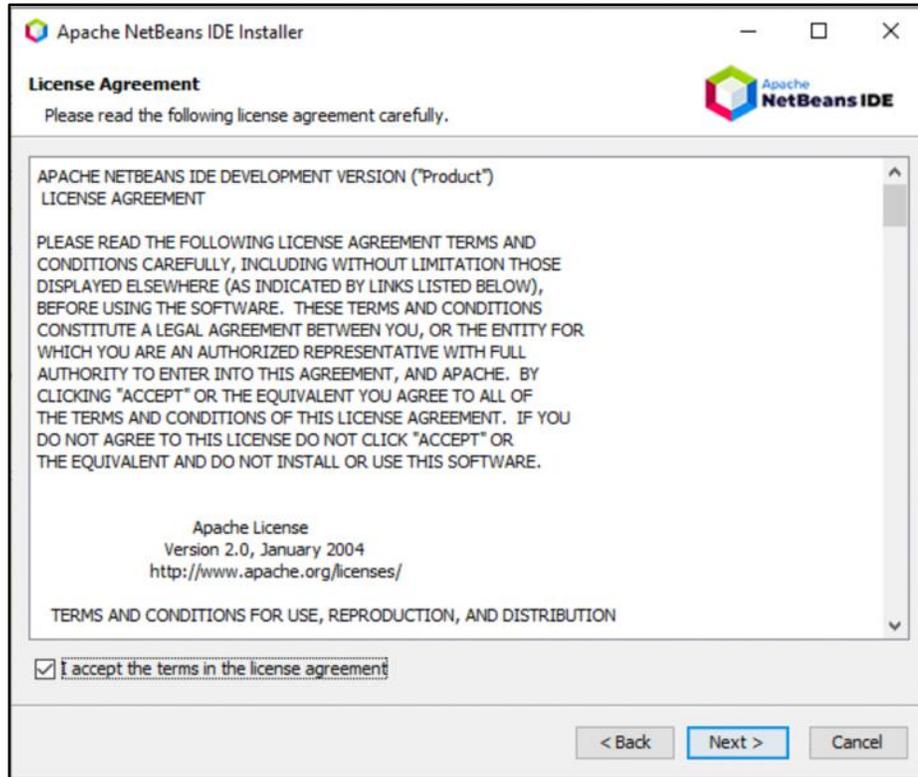
e. In the next window, under **JDK for the Apache NetBeans IDE**, make sure that *the location of the correct JDK has been chosen*. You may have multiple versions of JDK on your computer. The version you installed in Step 1 should be specified here (change to the right one, if it says different).
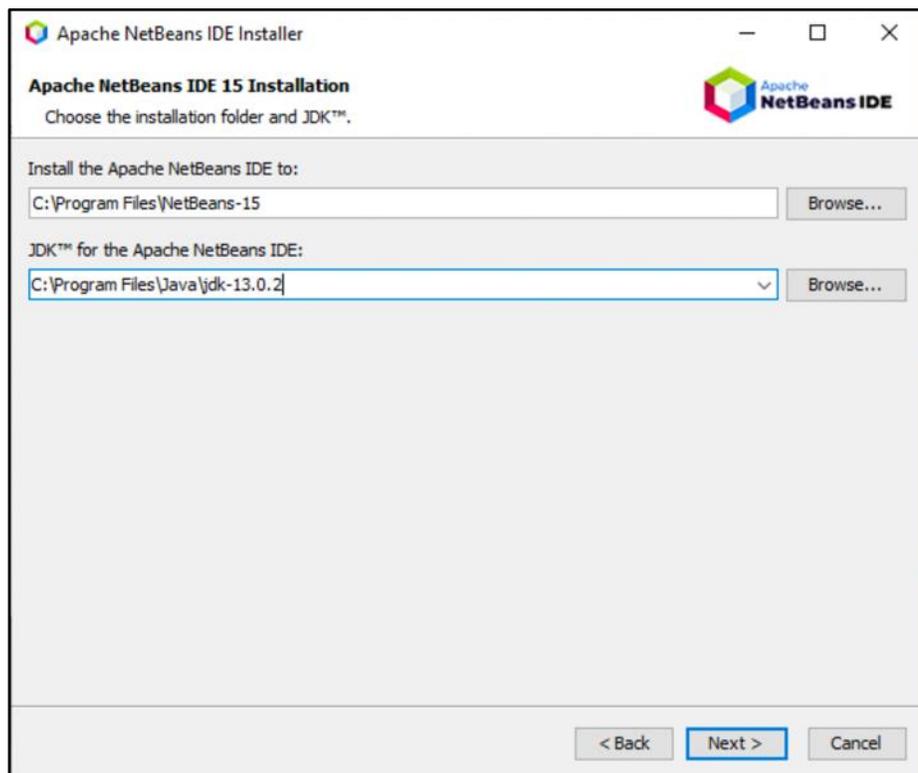
f. Click **Install** in the next window:
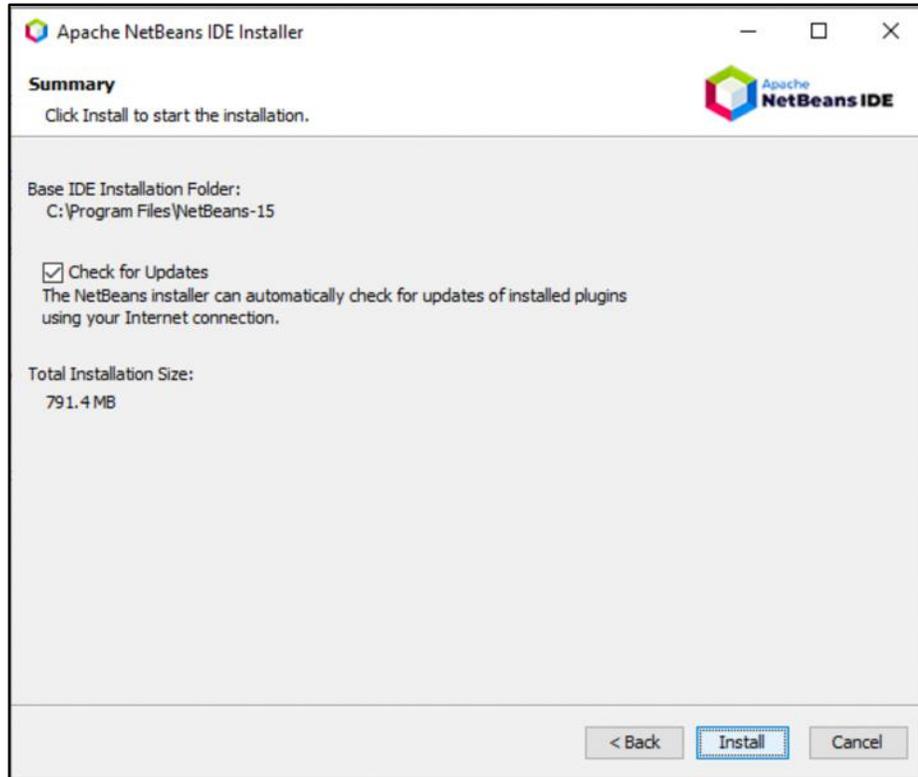
Installation may take a few minutes. After it's done, click the **Finish** button**.**

4. **Modify NetBeans to build C/C++ projects**
    The NetBeans IDE[9] is not set up to use with C/C++ projects by default. Follow the next steps to add this capability.

    a. Start NetBeans (e.g., go to the Windows Start Button, look for "**Apache NetBeans IDE…**", and click on it).
        In the NetBeans main window, go to the **Tools** menu, then **Plugins**.

---
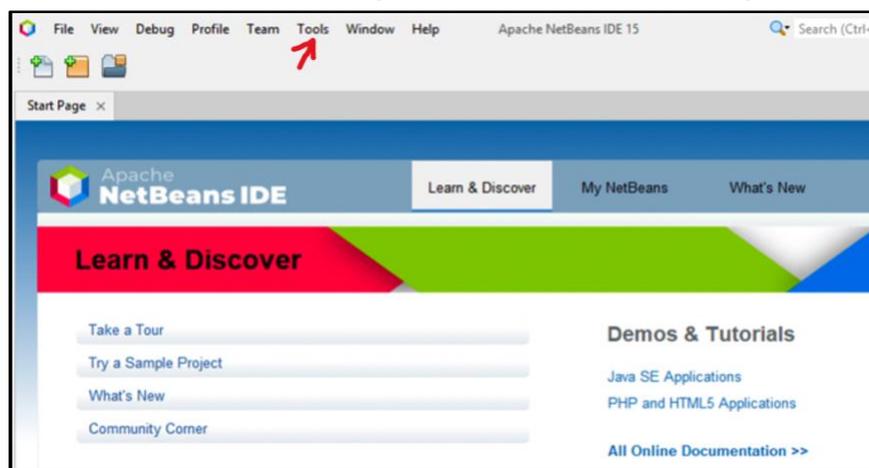
[9] IDE = Integrated Development Environment.

b. In the Plugins window, click on the **Settings** tab and make sure that **NetBeans 8.2 Plugin Portal** is *Active* (it's checkbox should be checked).
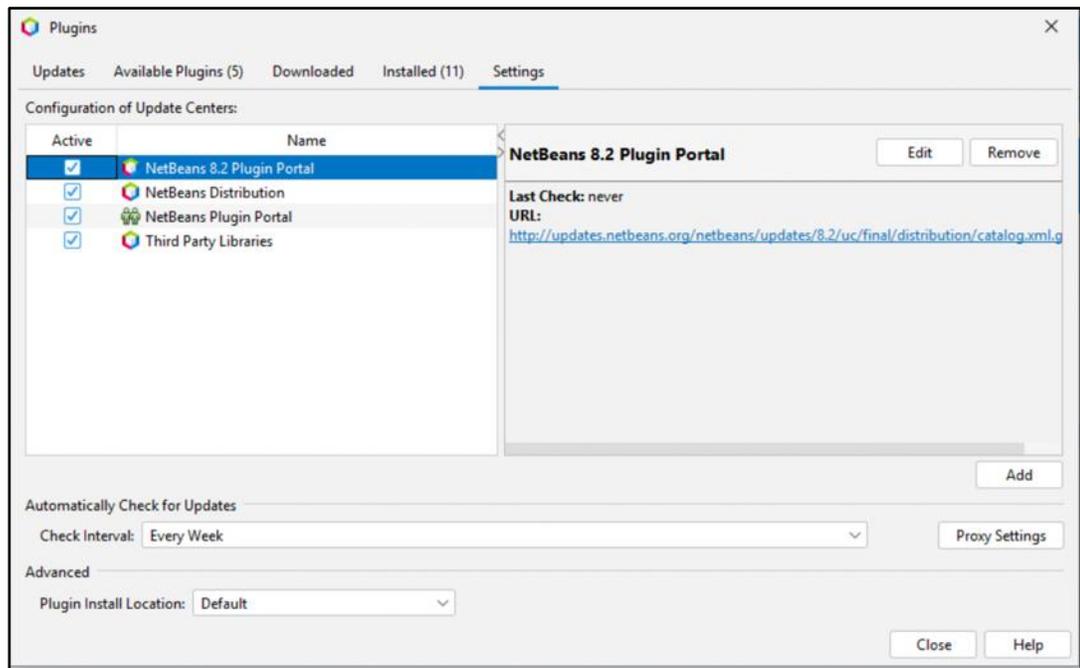
If the **NetBeans 8.2 Plugin Portal** is not available, click the **Add** button and the Update Center Customizer window will appear.
Enter "NetBeans 8.2 Plugin Portal" for the Name field, and use the URL:
    http://updates.netbeans.org/netbeans/updates/8.2/uc/final/distribution/catalog.xml.gz
for the repository.
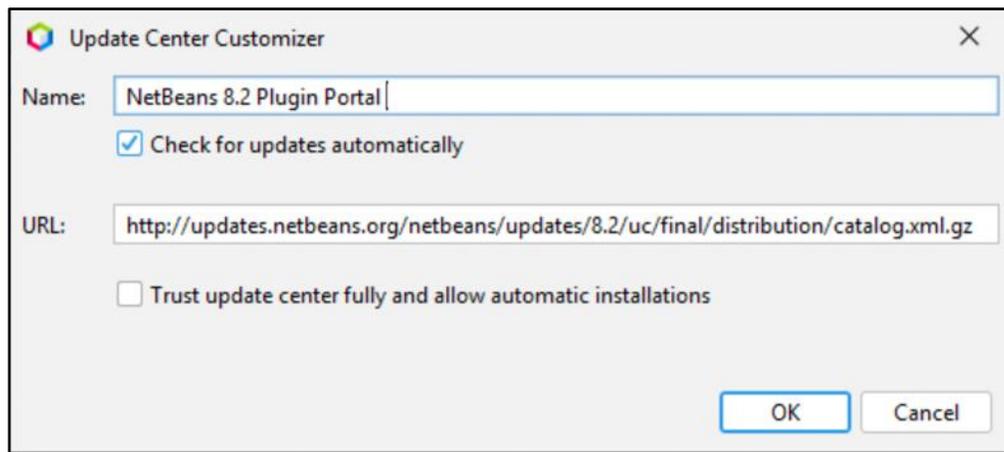
c. Now click on the **Available Plugins** tab. If you don't see "**C/C++**" in the list of options,
    click the **Check for Newest** button.
    Click the Install checkbox next to "**C/C++**", then click the **Install** button.
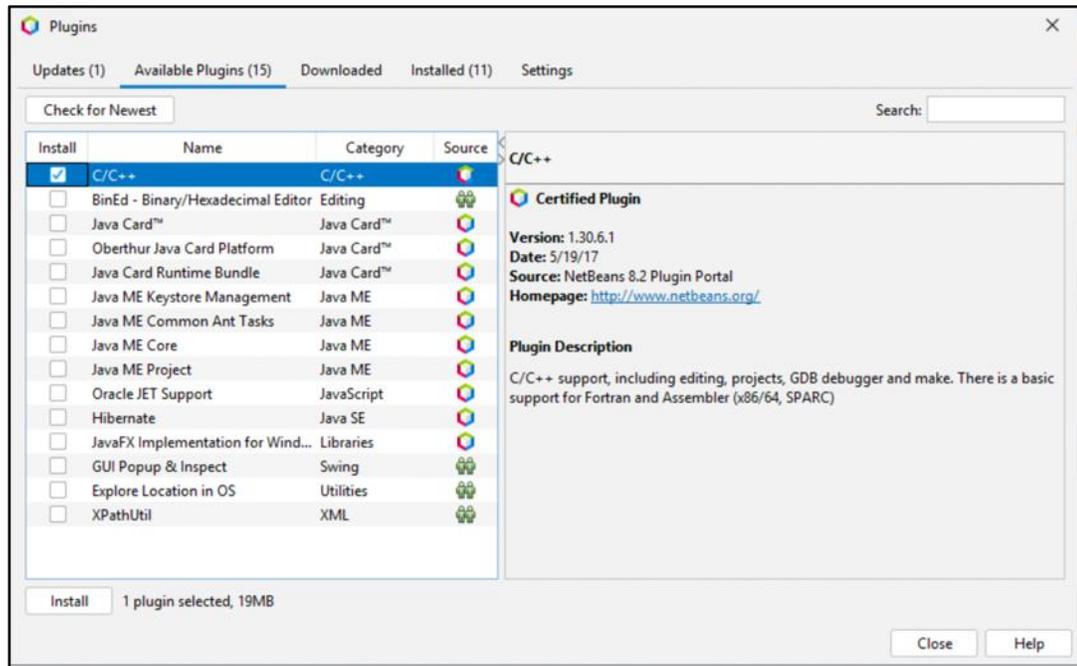
d. Click **Next** in the next window.
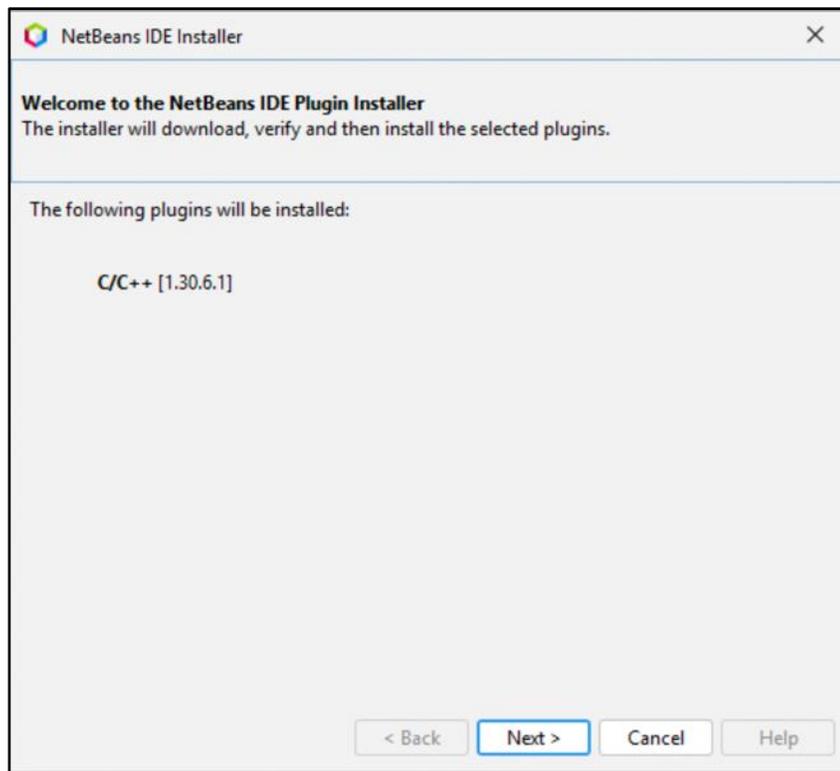
e. In the next window, click the checkbox to accept the license agreement, then click **Install**.
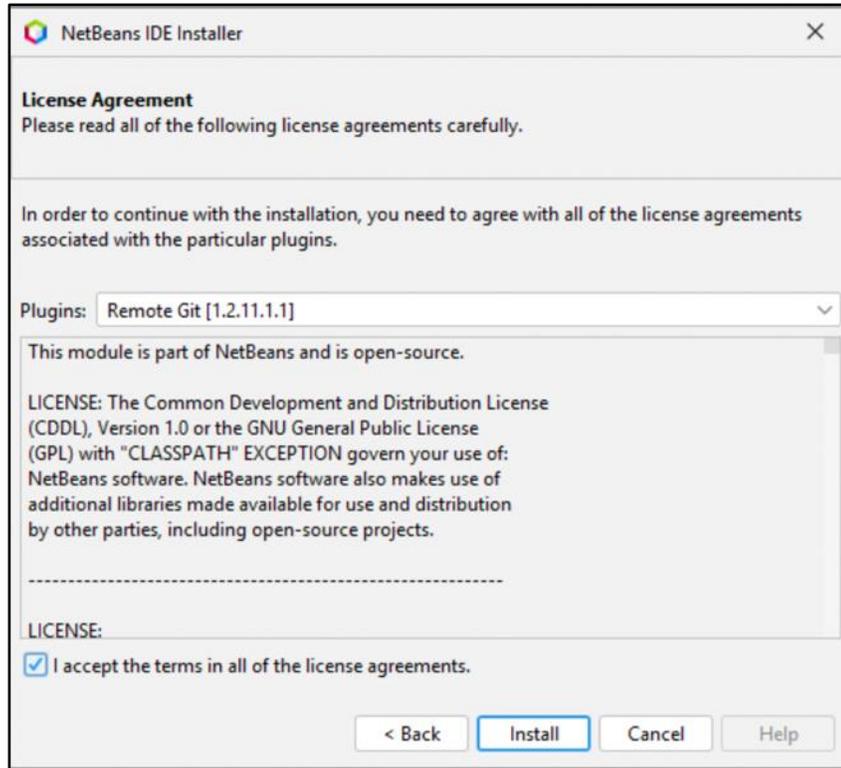
f. In the Verify Certificate window, click **Continue**.
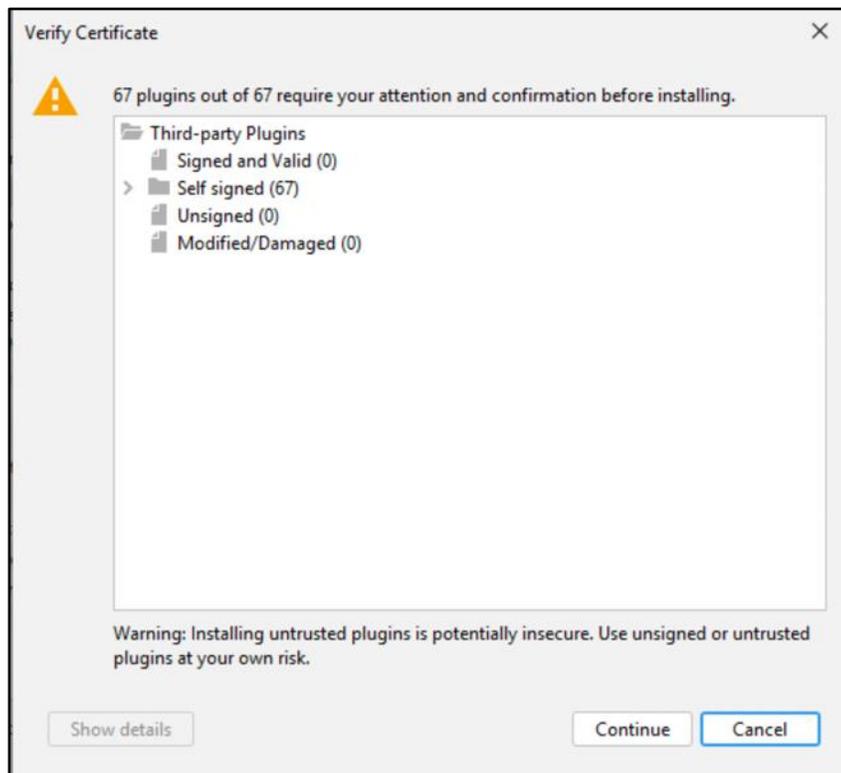
g. On the final window of the plugin installer, click **Finish**[10].

Then restart NetBeans.
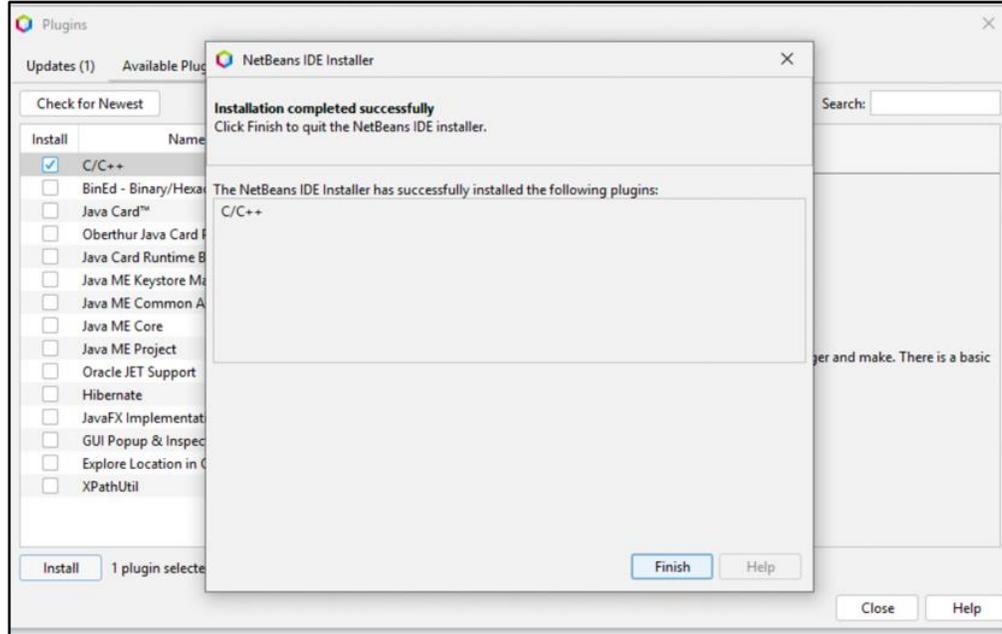
5. **Test C/C++ in NetBeans**
   Finally, after restarting NetBeans, try creating a new project.
   The method described here is likely the one you will be using for creating a new NetBeans C/C++ project throughout the course.

   a. Do *one* of the following to start a new project: a) click the gold/yellow "**New Project**" icon in the upper left of the NetBeans window; b) go to the **File** menu and then **New Project…**; or c) use the **<Ctrl>-<Shift>-N** key combination.

---
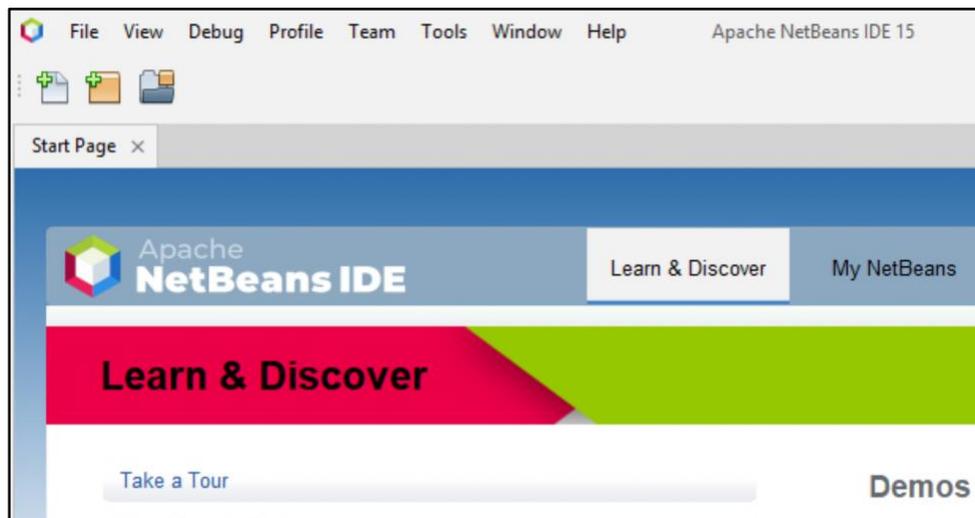
[10] In this final window, if the message says "Installation completed _un_successfully", proceed with the final steps anyways. This message occasionally occurs but often the C/C++ plugin still gets installed. If it looks like the plugin did not get installed correctly, get help from a lab aide or instructor.

b. The New Project window should open. You should see a **C/C++** option under Categories in this window. Click on it.
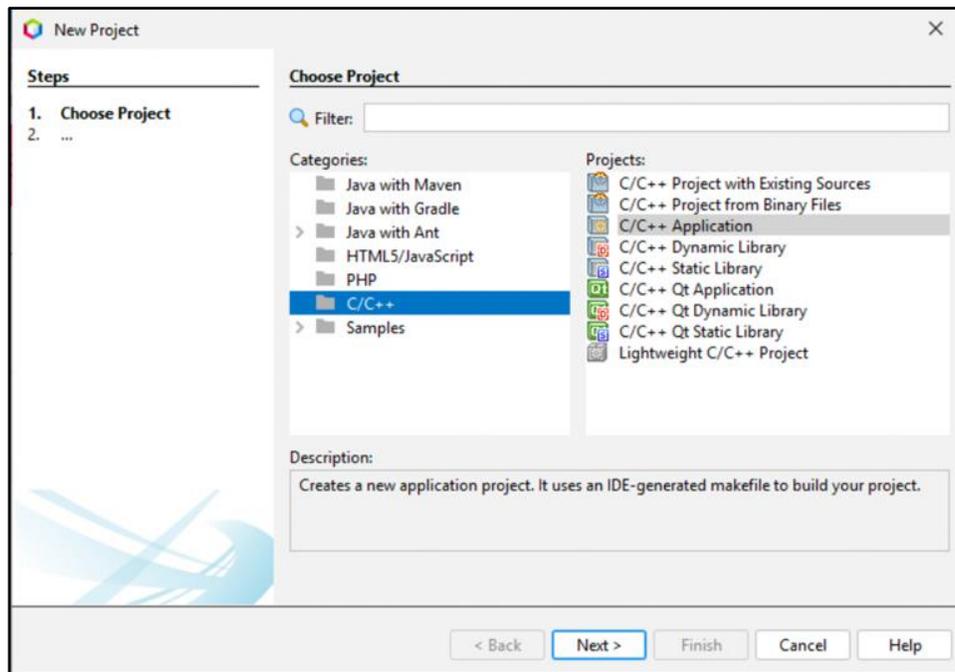
Under **Projects**, there should be several C/C++ entries.

If you only see "**Lightweight C/C++ Project**" in this section, the C/C++ configuration was *not* set up properly. In this case, go back to Step 4: **Modify NetBeans to build C/C++ projects**, and check those steps to see if something was missed.

If instead, you see all the C/C++ choices under **Projects**, select **C/C++ Application** and click **Next**.

c. The New C/C++ Application window appears.
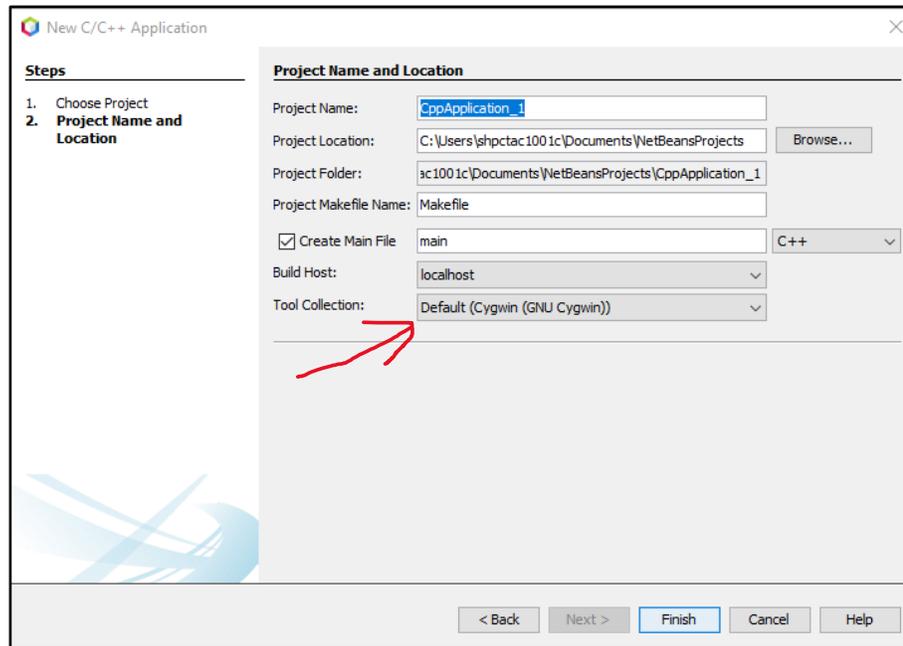
You can leave the Project Name as the default ("CppApplication_...") or change it to a more meaningful name.

IMPORTANT!
Make sure that the **Tool Collection** dropdown field is set to "**Cygwin (GNU Cygwin)**" or **"Default (Cygwin (GNU Cygwin))"**. If it is set to something else (e.g. "**MinGW (GNU MinGW)**"), change it to a Cygwin entry. *Some of the code you will be writing may not work properly under a tool collection other than Cygwin*.

If you want to set Cygwin as the default tool collection (so the **Tool Collection** dropdown would not need to be changed for every new project), see the **Appendix**.

d. Click the **Finish** button in the New C/C++ Application window. The window will close.

In the main NetBeans window, a new C/C++ application project folder 'tree' will appear in the **Projects** tab.
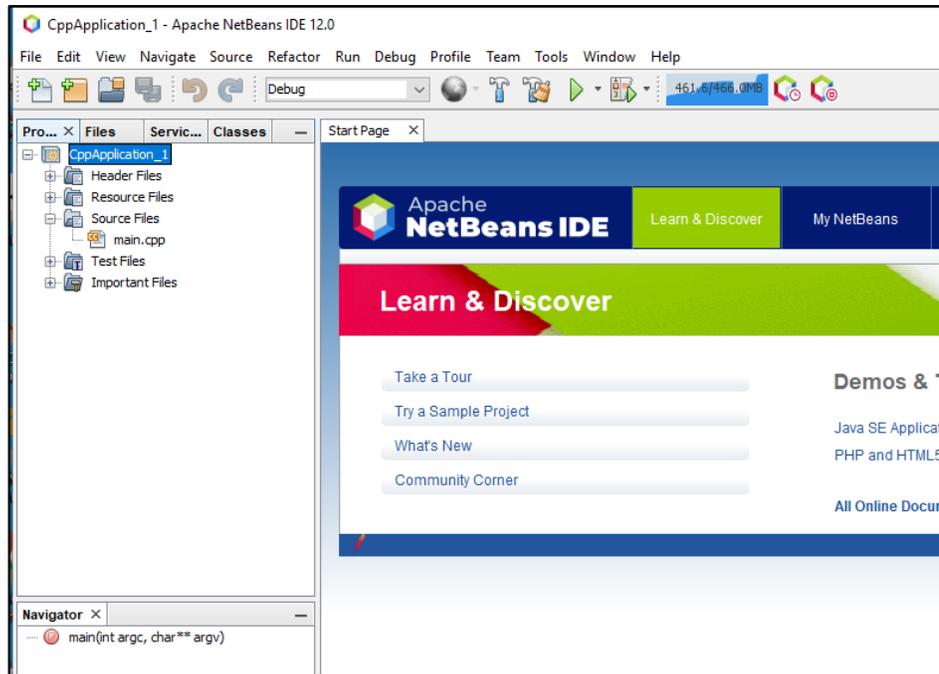
In the new project tree, click the "+" next to "Source Files" to open that section up. Then *double-click* on "**main.cpp**"[11].

e. The main.cpp file will open with some default text in it:

---
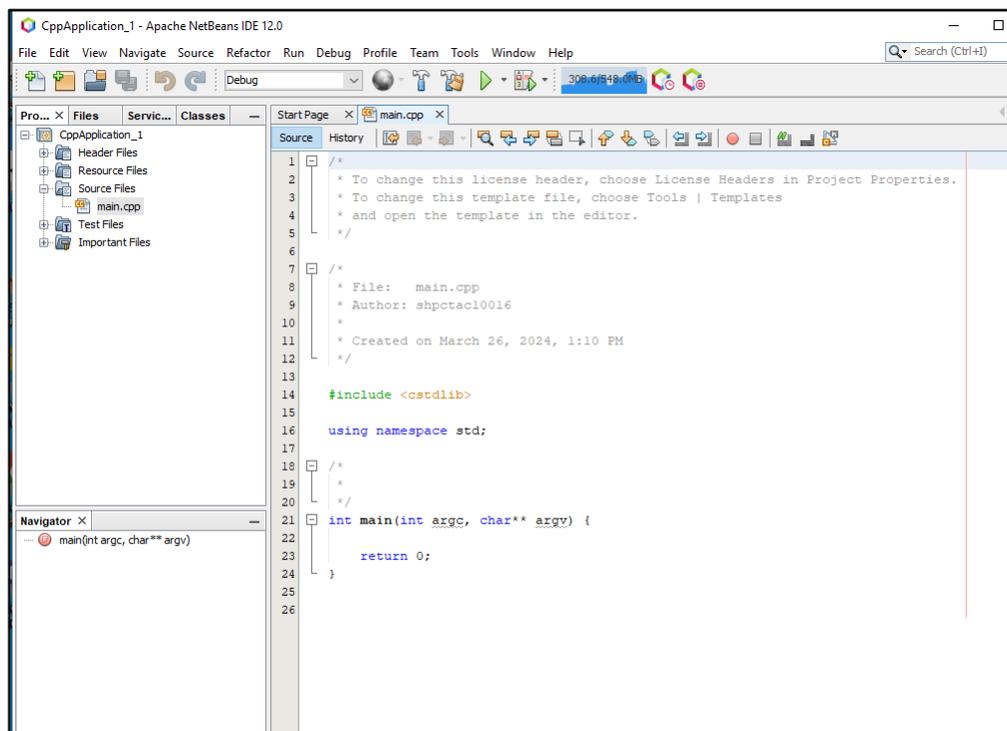
[11] You may have renamed this file from "main" in the New C/C++ Application window (**Step 5c**). If so, double-click on whatever the new name of the file is here.

f. Replace this text with the following:

```
#include <iostream>

using namespace std;

int main(int argc, char** argv) {
    cout << "Hello world" << endl;

    return 0;
}
```

This very simple "Hello World" program will help determine if NetBeans is able to run your code.

g. After entering the program, click the triangular green "Run Project" button in the toolbar:
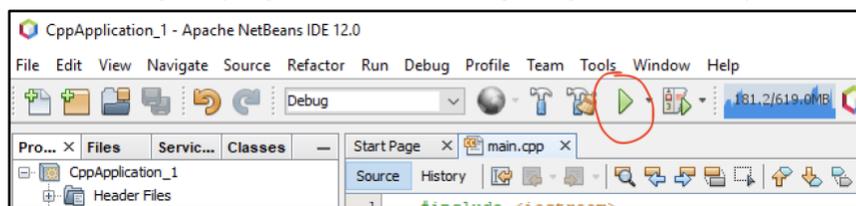
This button will attempt to compile, link and run your program.

h. If everything works right, you should see the text "Hello world" in the output panel in the lower part of the window:
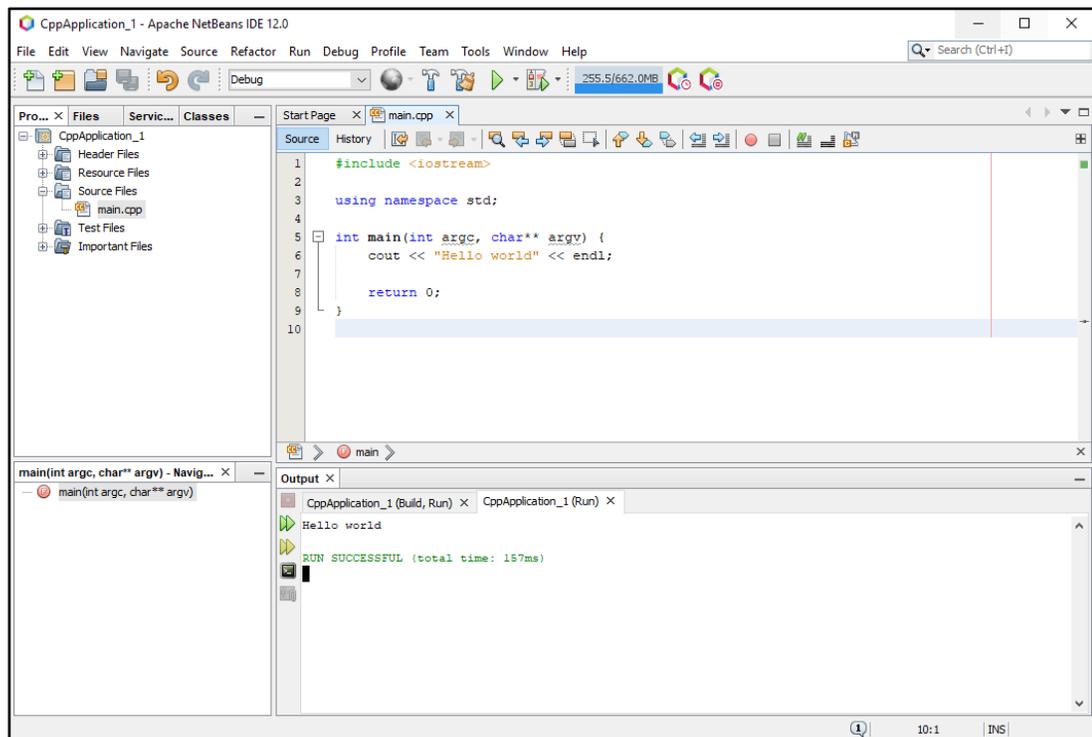
If you got this working, it's time to take a break and celebrate!

Appendix –

How to set Cygwin as your default tool collection

I. In **the** main NetBeans window, go to **Tools->Options**. The Options window opens.
Then click on the **C/C++** tab.

If you do **not** see the **C/C++** tab in the Options window, the C/C++ plugin was not installed. In this case, go back to Step 4: **Modify NetBeans to build C/C++ projects** above and repeat it.
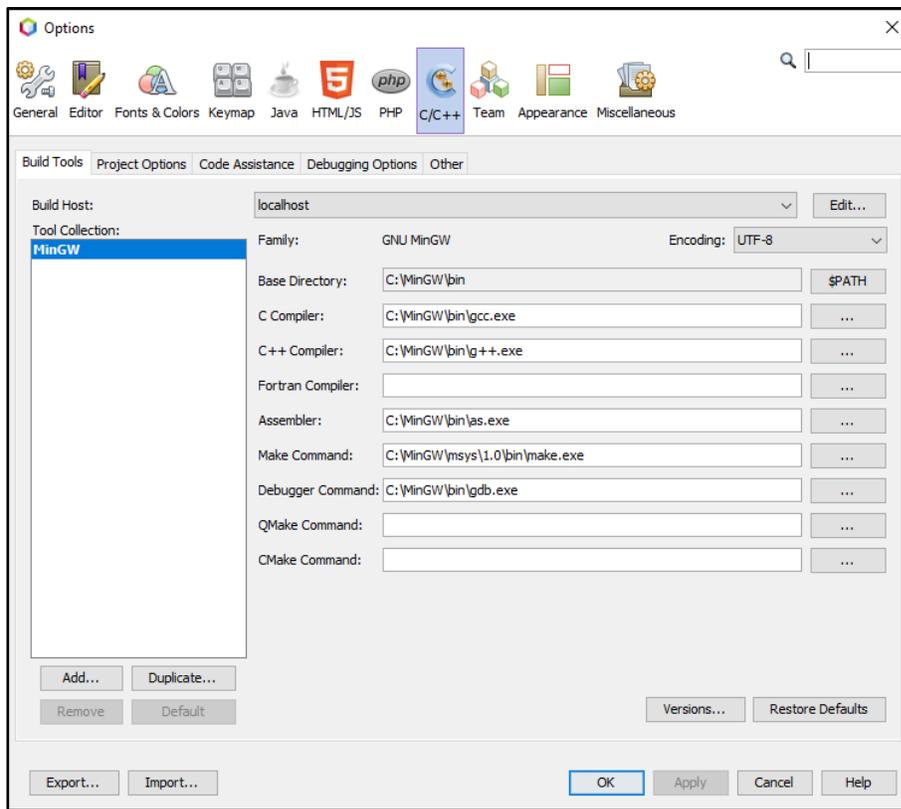
II. If "**Cygwin**" *is* an option under **Tool Collection**, skip over this step.

Otherwise, if you do not see "**Cygwin**" here, click the **Add** button below the **Tool Collection** section.

The Add New Tool Collection window appears.

Use the **Browse** button, or type "**C:\cygwin64\bin**" in the **Base Directory** text box[12].
Then set the **Tool Collection Family** to "**GNU Cygwin**".
The **Tool Collection Name** field should automatically populate with "**Cygwin**".

---

[12] Cygwin likely has been installed in "**C:\cygwin64\bin**" or a similar directory on your computer. If you cannot find Cygwin on the computer, it may not have been installed. Go back and repeat Step 2: **Install Cygwin** in this case.
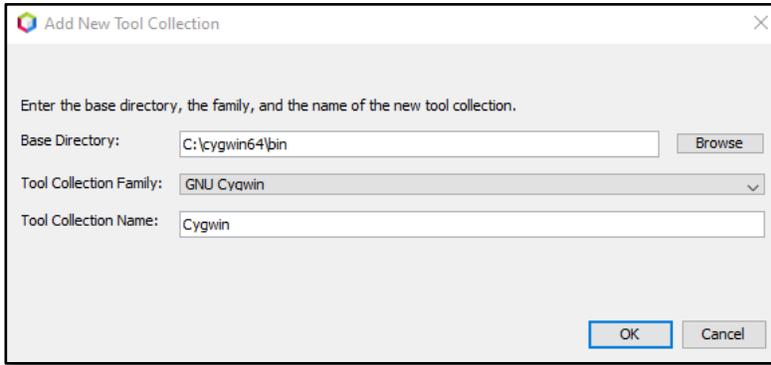
If everything looks okay in the Add New Tool Collection window, click the **OK** button.

III.   In the Options window, in the **C/C++** tab, select "**Cygwin**" under **Tool Collection**, if it is not already selected.
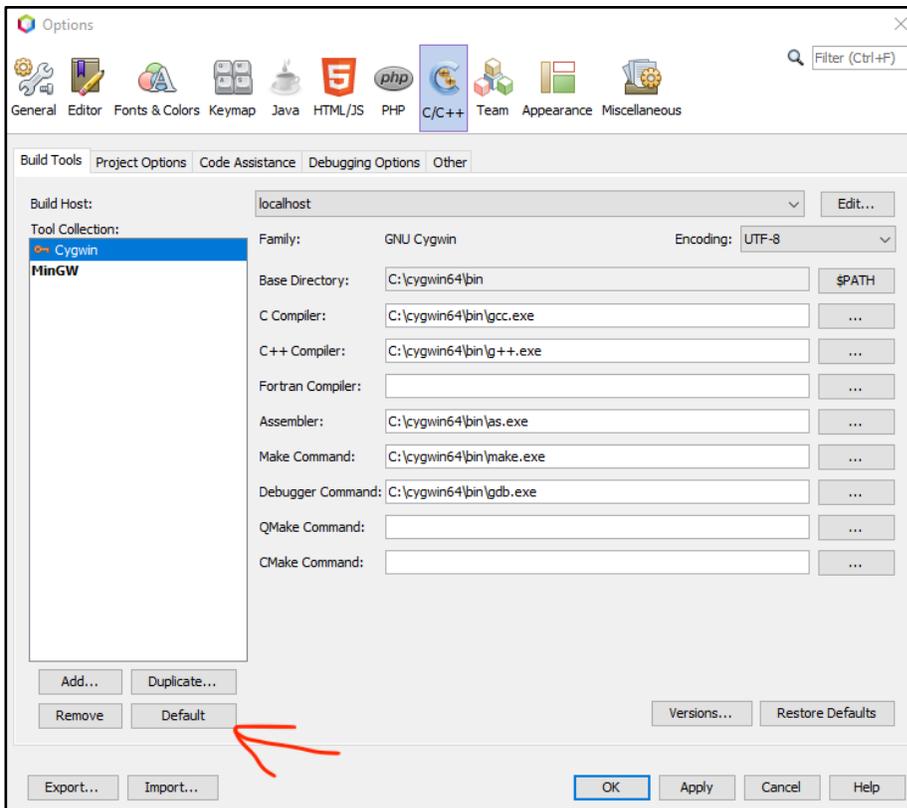
Now click the **Default** button. *You must do this do this to set Cygwin as the default tool collection – even if it is selected under **Tool Collection** in this window.*

Click the **OK** button to close the Options window.

Your tool collection should now default to Cygwin when you create new C/C++ projects.