

# Using C++ with NetBeans

## For Introduction to Programming With C++

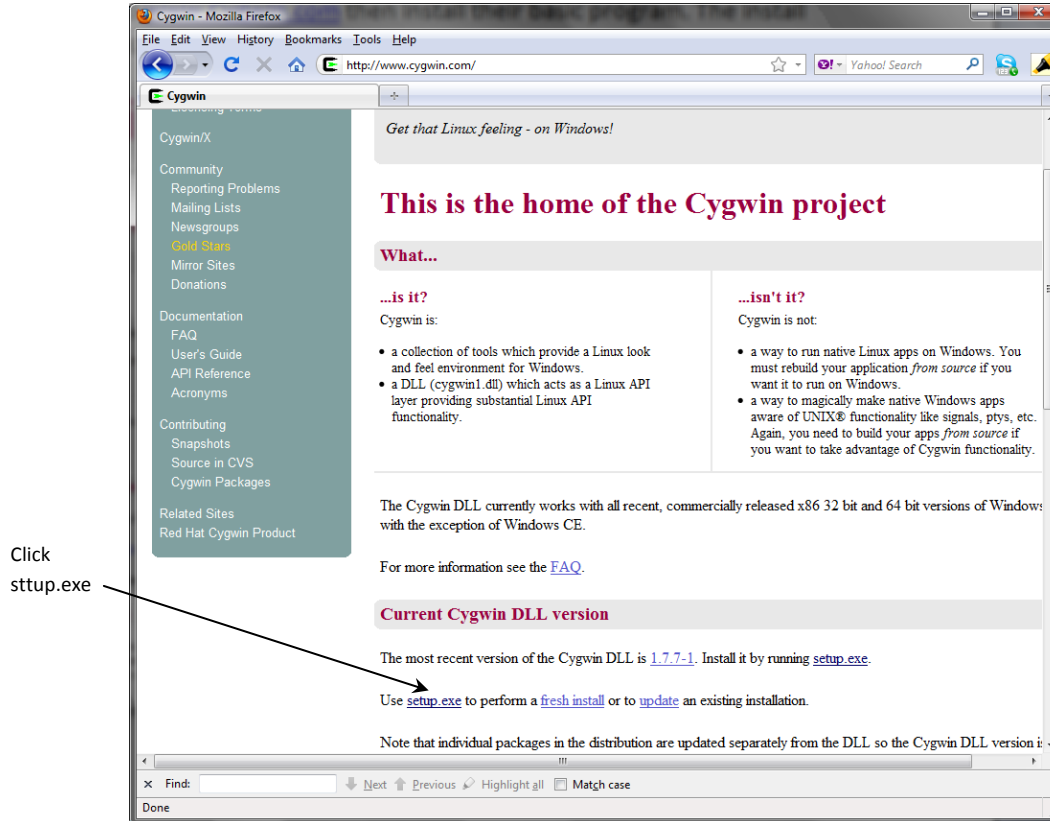
NOTE: This tutorial is for experience users to develop C++ programs using NetBeans.

To use C++ with NetBeans, you have to download and install a C++ compiler and NetBeans.

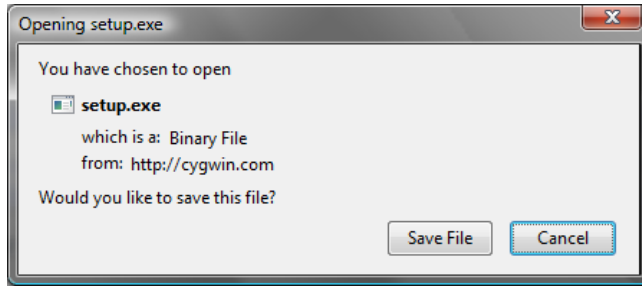
### Installing a C++ Compiler

NetBeans can work with any C++ compiler. In this tutorial, we will use the Cygwin compiler.

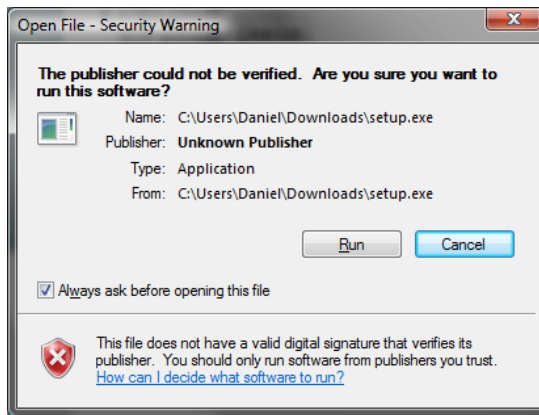
1. To download the Cygwin compiler, go to [www.Cygwin.com](http://www.Cygwin.com), as shown in Figure 1. Click the hyperlink on `setup.exe` to download `setup.exe`, as shown in Figure 2. After it is saved, open and run the file, as shown in Figure 3. Click the Run button to start the setup, as shown in Figure 4.



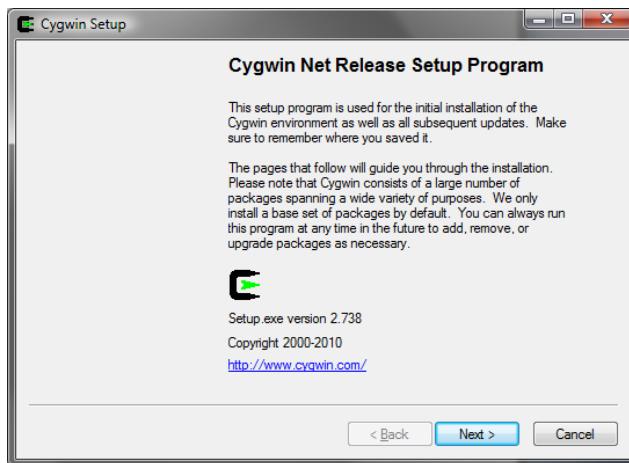
**Figure 1**  
*Download setup.exe to install a C++ compiler.*



**Figure 2**  
*You need to save setup.exe and then run it.*

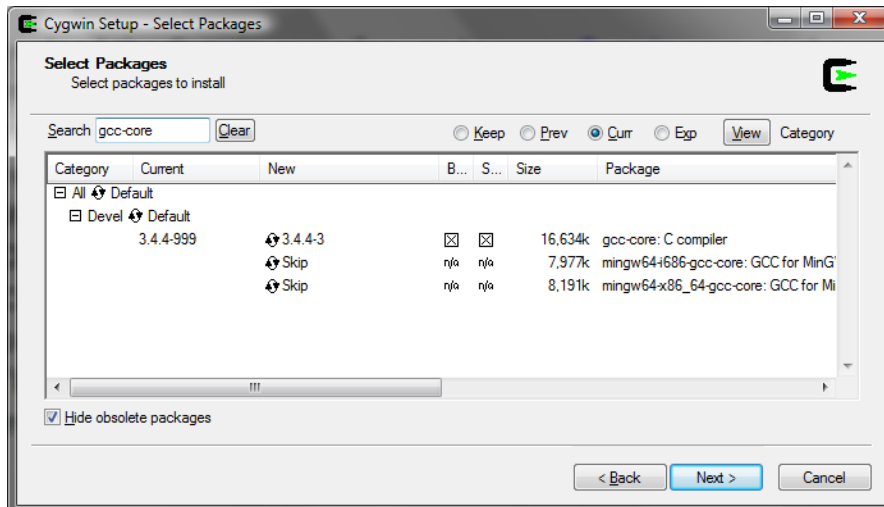


**Figure 3**  
*Run setup.exe to start setup.*



**Figure 4**  
*The setup program will install a C++ compiler.*

2. Continue to click Next on each subsequent dialog box until you see the Select Packages dialog box, as shown in Figure 5. In the Search field, enter gcc-core, you will see gcc-core compiler appearing Package column. Select it.



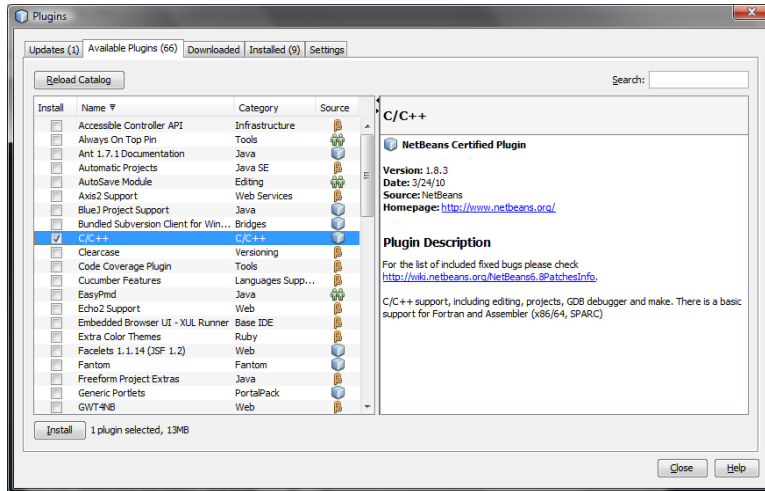
**Figure 5**

*You need to select appropriate packages to install.*

3. Continue to search and select the following packages
  - gcc-g++: C++ compiler
  - gdb: The GNU Debugger
  - make: the GNU version of the 'make' utility.
4. Click *Next* to install the selected packages.
5. After installation is completed, go to `c:\cygwin\bin` to verify if `gcc.exe`, `g++.exe`, `gdb.exe`, and `make.exe` are present. If not, go back to re-run setup and select these packages to install.

### Configuring C++ with NetBeans

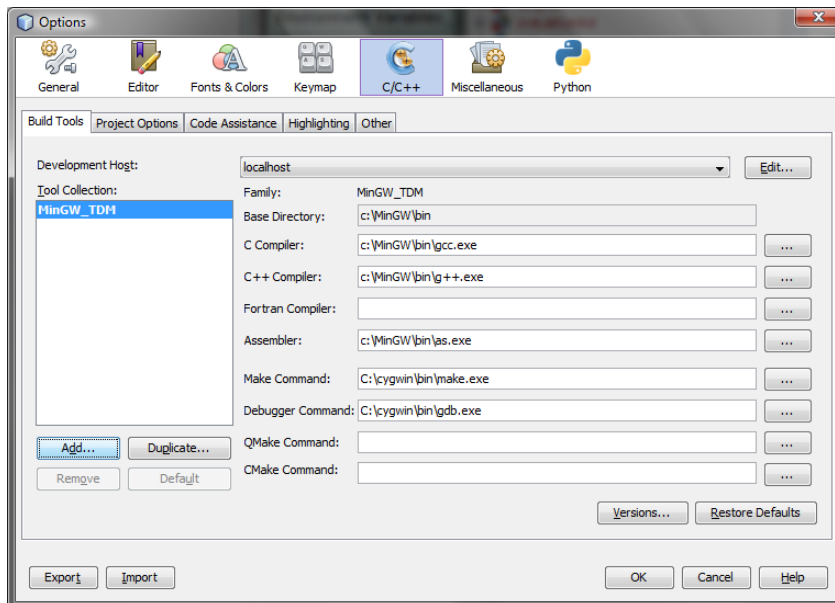
1. If you are not familiar with NetBeans, go to <http://www.cs.armstrong.edu/liang/intro8e/idesupplement.html> to obtain the information on installing and using NetBeans.
2. Launch NetBeans. Go to the **Tools** menu and choose Plugins to display the Plugins window, as shown in Figure 6.
3. In the **Available Plugins** tab, choose C/C++ and click the Install button. After the installation is completed, restart NetBeans.



**Figure 6**

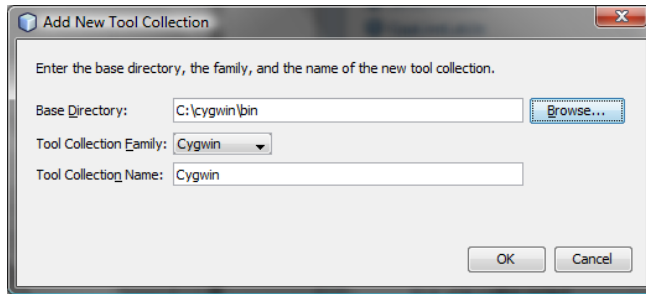
*Choose C/C++ to install C/C++ in NetBeans.*

4. From the **Tools** menu, choose Options to display the Options dialog box, as shown in Figure 7. Click C/C++. In the C/C++ page, click the Add button to display the Add New Tool Collection dialog box, as shown in Figure 8.



**Figure 7**

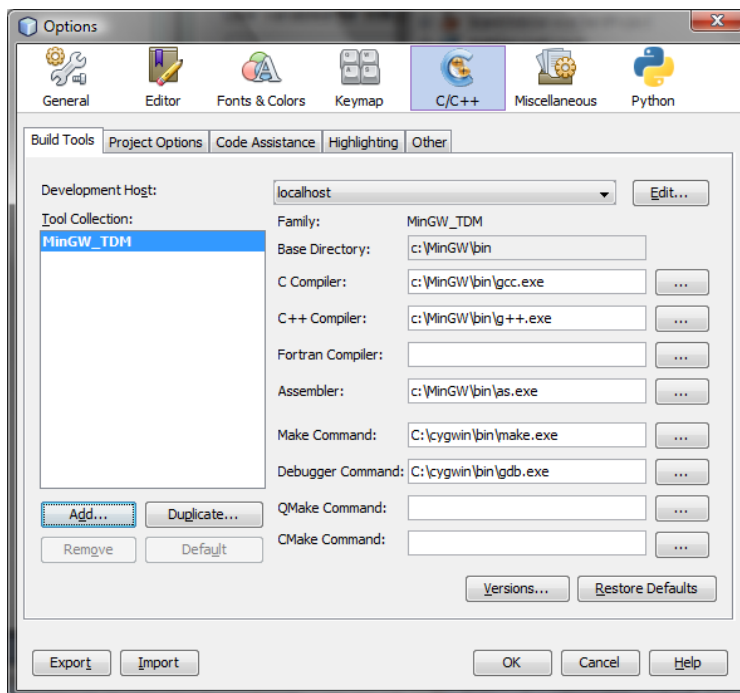
*Click the Add button to add a new tool.*



**Figure 8**

*Browse for c:\cygwin\bin to set it as the Base Directory.*

5. Browse to set c:\cygwin\bin as the Base Directory and click OK. You will see the compiler, assembler, make, and debugging commands set for the tool, as shown in Figure 9.

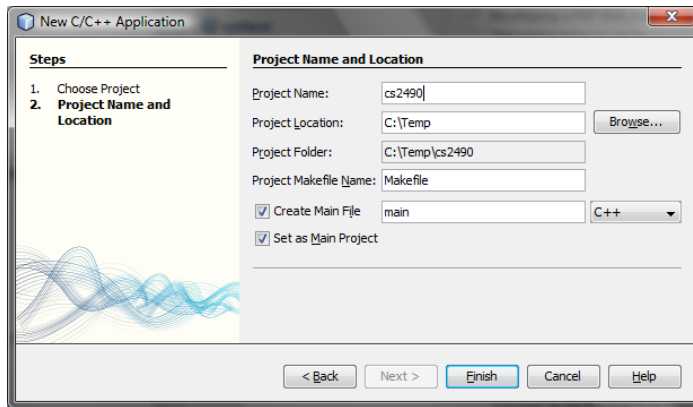


**Figure 9**

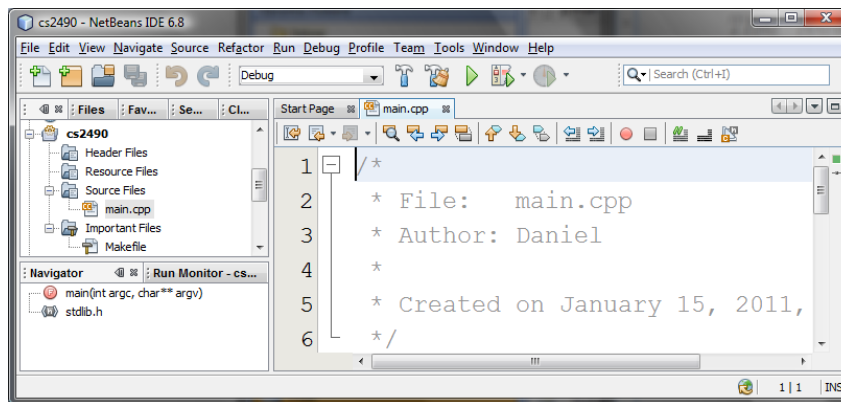
*The compiler and other supporting commands are set in the option.*

### Creating a C++ Project

1. Now you can create a C++ project to hold C++ programs. Choose File, New Project to display the New Project dialog box. Choose C/C++ in the Category section and choose C/C++ Application in the Projects section and click Next to display the New C/C++ Application dialog box, as shown in Figure 10.



**Figure 10**  
*The New C/C++ Application dialog box creates a new project for C/C++ applications.*

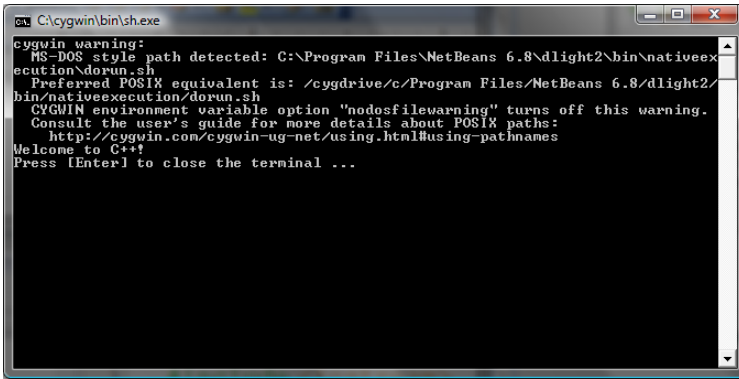


**Figure 11**  
*A new project for C/C++ Applications is created.*

2. Enter a project name (e.g., cs2490) in the Project Name field and click Finish to create a project, as shown in Figure 11.

### **Creating and Running a C++ Program**

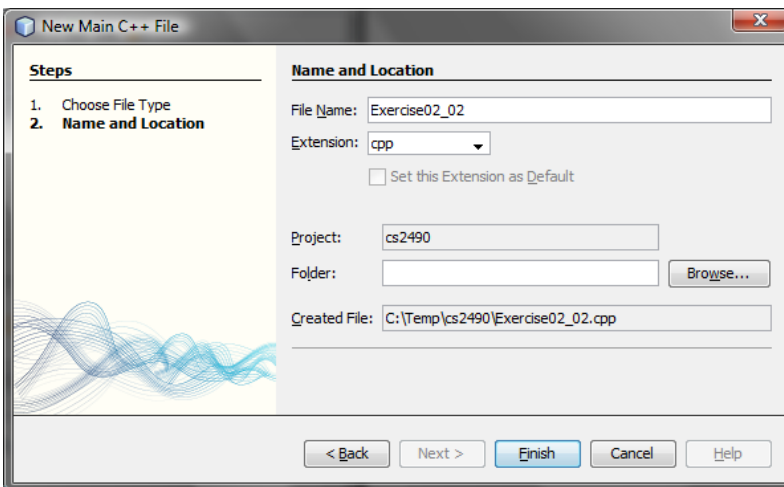
1. By default, main.cpp is automatically created. Replace main.cpp with Welcome.cpp in Listing 1.1 in the text.
2. Right-click the project node cs2490 and choose Run to run the program. The output is shown in a window, as shown in Figure 12.



**Figure 12**

*The output for the program is displayed in a command window.*

3. Now let us create a new program. To do so, right-click the Source Files node in the project and choose *New, Main C++ File* to display the New Main C++ File dialog, as shown in Figure 13. Enter Exercise02\_02 as the file name and click *Finish* to create the file.



**Figure 13**

*Enter a file name to create a .cpp file.*

4. Enter the following code for Exercise02\_02

```
#include <iostream>
using namespace std;

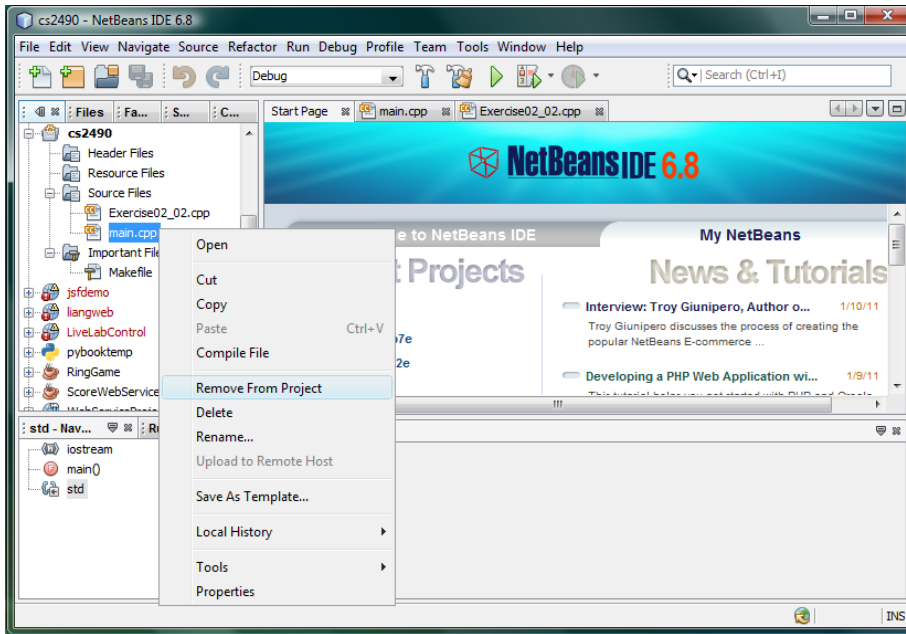
int main()
{
    cout << "Enter the radius and length of a cylinder: ";
    double radius, length;
    cin >> radius >> length;

    double area = radius * radius * 3.14159;
    double volume = area * length;

    cout << "The area is " << area << endl;
```

```
cout << "The volume is " << volume << endl;  
  
return 0;  
}
```

5. Before you can run this program, you need to remove main.cpp from the project, because the project can have only one file with a main function. Right-click on main.cpp and choose *Remove From Project* to remove it, as shown in Figure 14.
6. Right-click on the project node cs2490 to run the project.



**Figure 14**

*You can remove a file from the project.*