

# ECLIPSE C++ TUTORIAL

## Table of Contents

<b>A. INTRODUCTION .....</b>	<b>2</b>
<b>B. SECTION 1: Creating C++ Project.....</b>	<b>2</b>
1. Step 1: Create a C++ project and Run .....	2
2. Step 2: Create Source Files.....	4
3. Step 3: Build a Project .....	5
4. Step 4: Run the application.....	6
<b>C. SECTION 2: Printing and Transferring Files .....</b>	<b>8</b>
1. Part 1: Print Source Codes.....	8
2. Part 2: Print screen output.....	9
3. Part 3: Save and store files using SSH .....	11
<b>D. SECTION 3: Saving Files via SSH and Adding Files into Project.....</b>	<b>13</b>
1. Part 1: Locate source files in the project folder and save them via SSH.....	13
2. Part 2: Download source files via SSH and add them into an empty project .....	15

## A. INTRODUCTION

**Section 1** guides you through creating a simple C++ application using the Eclipse C/C++ Development Toolkit (CDT) using the following steps:

- ❖ Create a C++ project
- ❖ Create source files
- ❖ Build a project
- ❖ Run the application

**Section 2** shows you how to:

- ❖ Print source codes
- ❖ Print screen output
- ❖ Save files using SSH

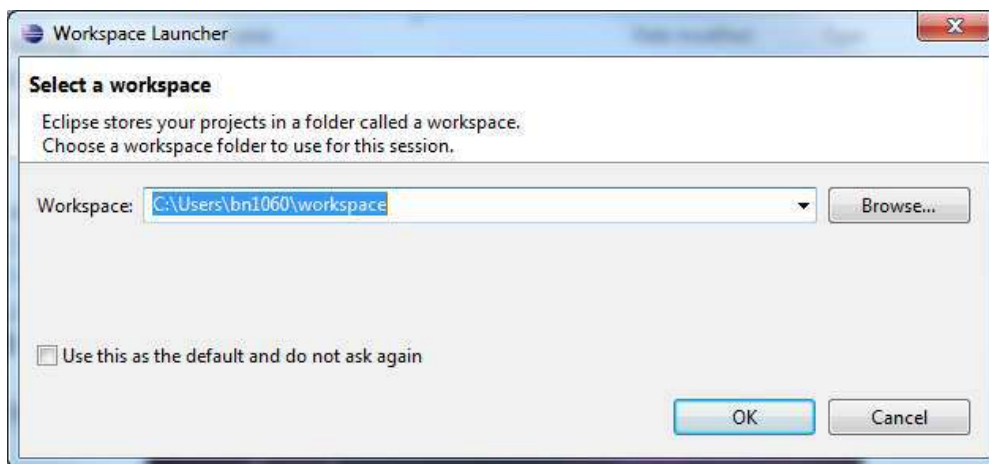
**Section 3** shows you how to:

- ❖ Upload or download source files via SSH
- ❖ Importing existing source files to a project

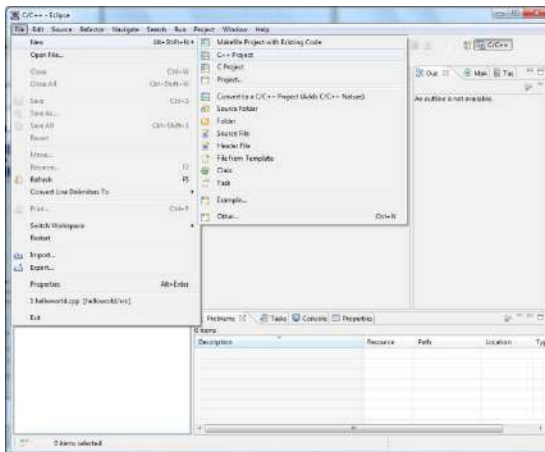
## B. SECTION 1: Creating C++ Project

### 1. Step 1: Create a C++ project and Run

- 1.1. Run Eclipse C++ by double clicking on **eclipse.exe**, the **Workspace Launcher** window will pop up. You need to specify the workspace you want to set up. The **default** workspace of lab machine is :  
`C:\Users\<username>\workspace`

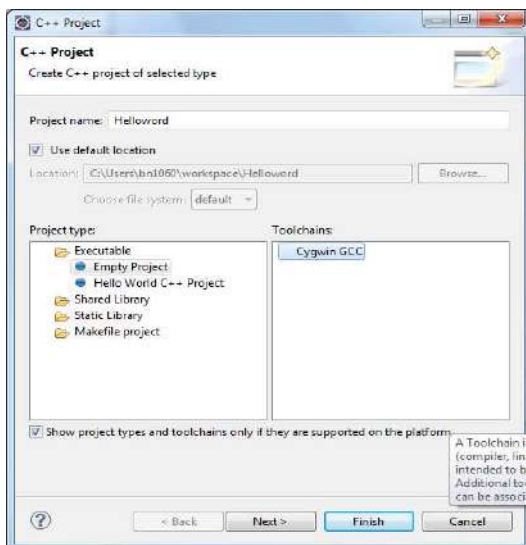


- 1.2. Select **File** → **New** → **C++ Project**

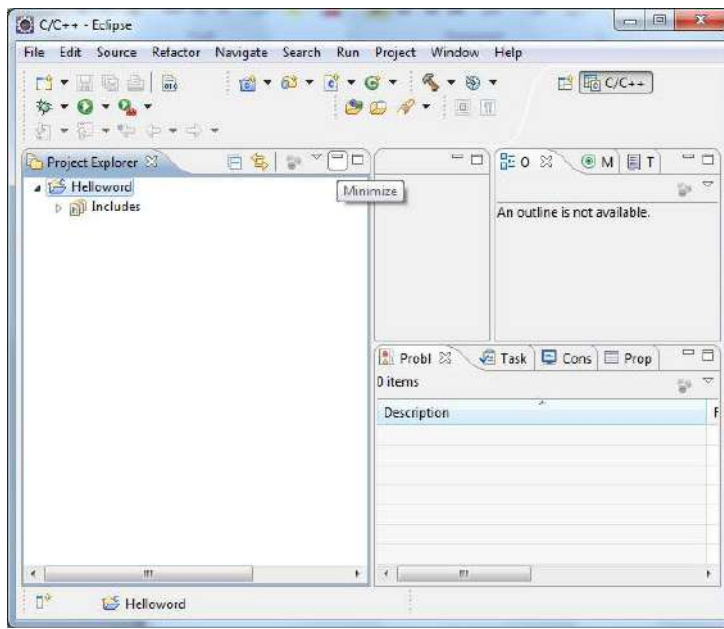


1.3. The C++ Project Wizard opens.

- ❖ In the **Project name** field, type in a name for the project, for example *Helloworld*.
- ❖ In the **Project type** field, select **Empty Project** under **Executable** folder.
- ❖ In the **Toolchains** field, select a set of tools (compiler, linker, assembler or debugger) to build the project. You may have more than one toolchain depending on what is installed on your system. However, in the “**Installing Eclipse C++ for Window and Linux**”, we installed **Cygwin**; therefore, it only shows **Cygwin GCC**.



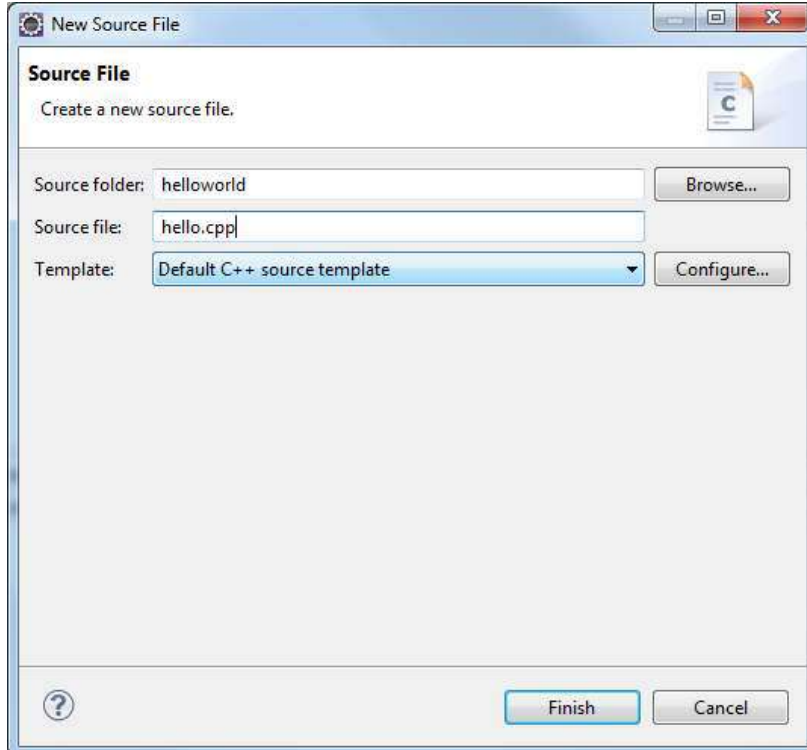
1.4. Click **Finish**. A project is created with default settings and a full set of configurations based on the project type and the toolchain you selected.



## 2. Step 2: Create Source Files

### 2.1. Select **File** → **New** → **Source File**

- ❖ A pop up window open. In the **Source file** field, type the name of your new source file, for example **hello.cpp**.
- ❖ **Click Finish.**



### 2.2. Type the following code into the blank editor

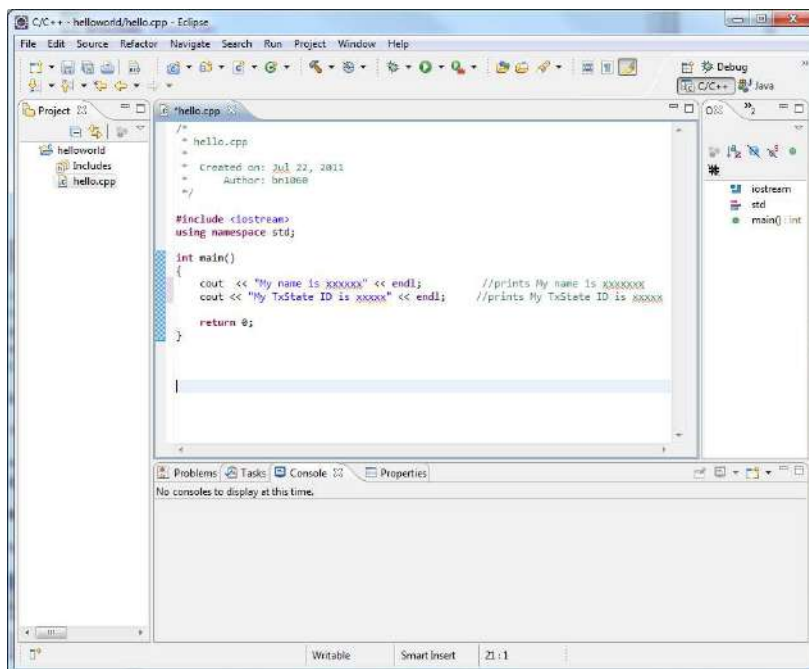
```
#include <iostream>
```

using namespace std;

```
int main()
{
    cout << "My name is xxxxxx" << endl;    //prints My name is xxxxxxxx
    cout << "My TxState ID is xxxxx" << endl; //prints My TxState ID is xxxxx

    return 0;
}
```

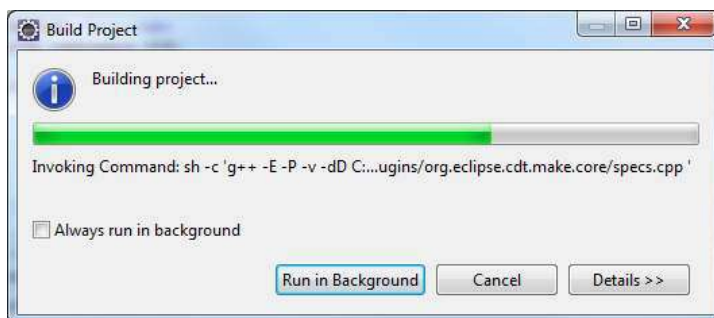
**Note:** Replace xxxxxx with your name and your ID



2.3. Select **File** → **Save** (or **Ctrl + S**)

### 3. Step 3: Build a Project

3.1. Select **Project** → **Build All**



3.2. If the project builds successfully, the following message will be displayed in the **Console** view.

\*\*\*\* Build of configuration Debug for project helloworld \*\*\*\*

```
make all
Building file: ../hello.cpp
```

```

Invoking: Cygwin C++ Compiler
g++ -O0 -g3 -Wall -c -fmessage-length=0 -MMD -MP -MF"hello.d" -MT"hello.d" -o
"hello.o" "../hello.cpp"
Finished building: ../hello.cpp

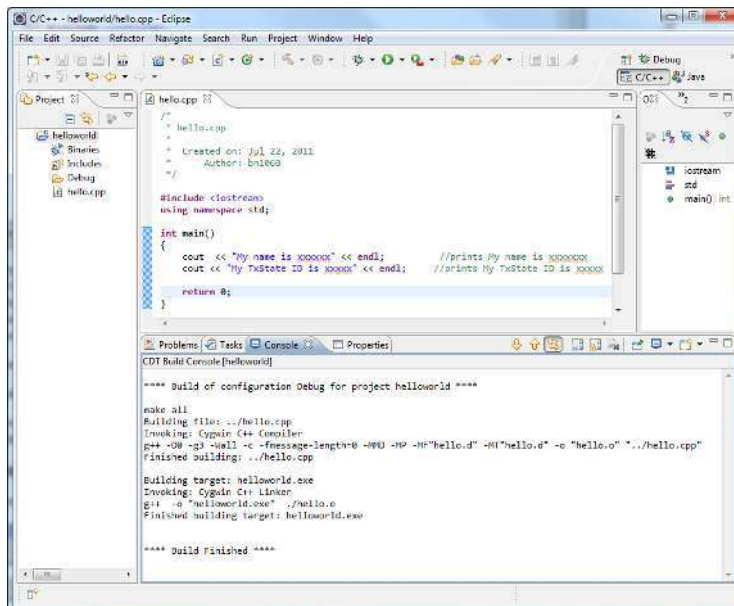
```

```

Building target: helloworld.exe
Invoking: Cygwin C++ Linker
g++ -o "helloworld.exe" ./hello.o
Finished building target: helloworld.exe

```

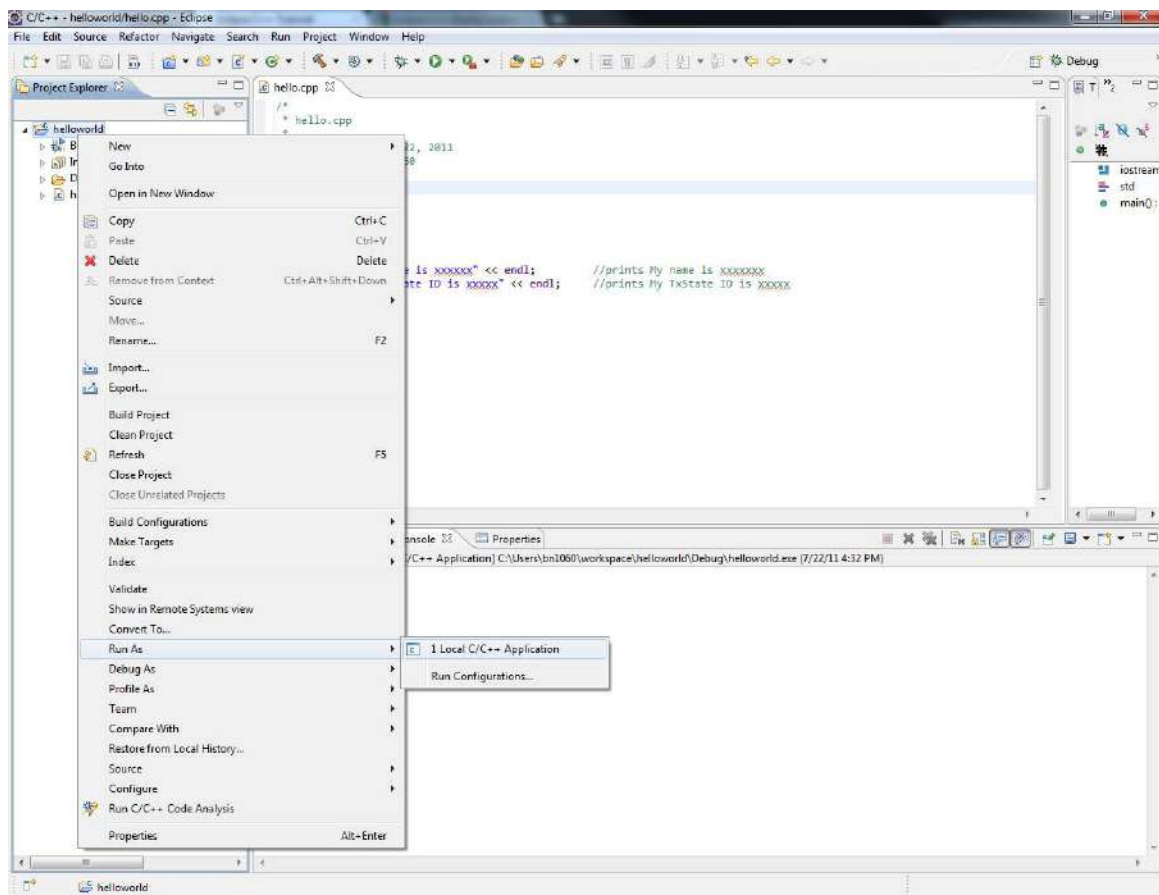
\*\*\*\* Build Finished \*\*\*\*



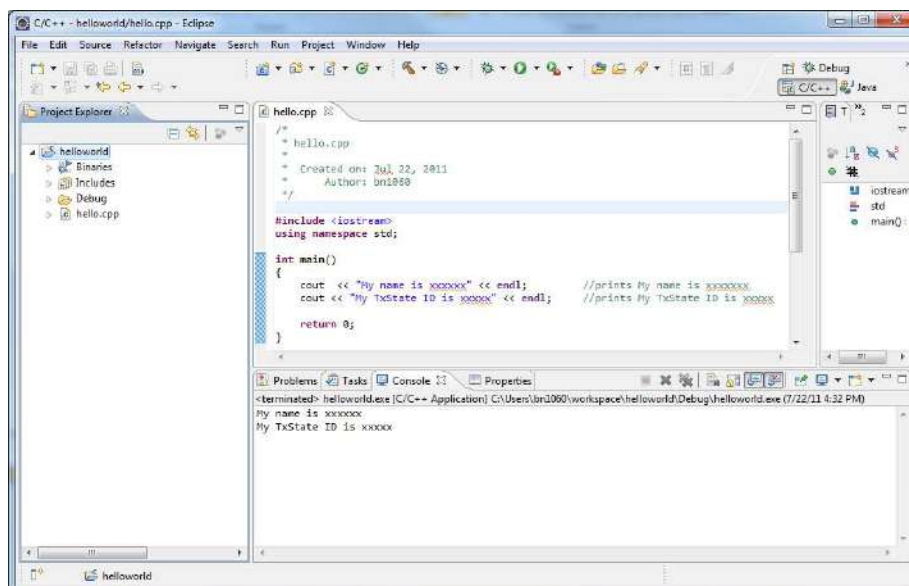
**Note:** If the project builds unsuccessfully, error or warning messages will be displayed in the **Console** view.

#### 4. Step 4: Run the application

- 4.1. In the **Project Explorer** tab, right click on the title of project name and select **Run As** → **Local C/C++ Application**.

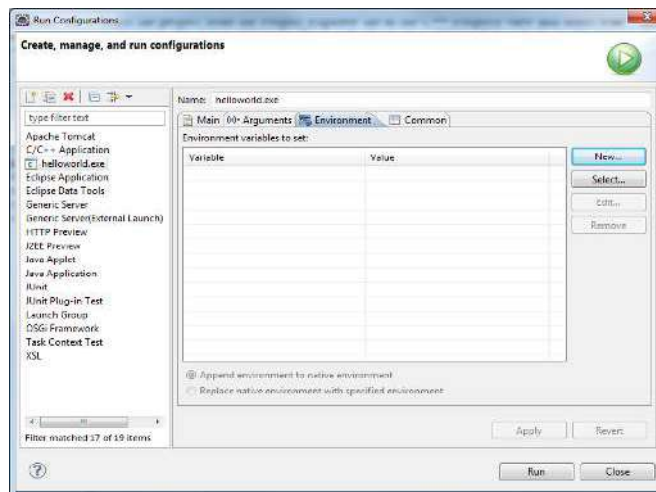


4.2. The output of the application will be displayed in the **Console** view.

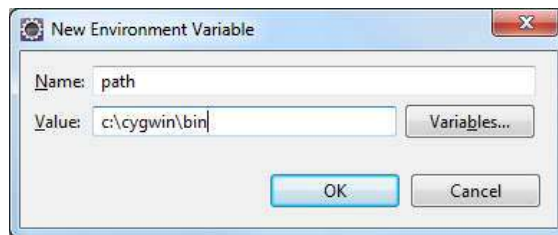


**Note:** If you installed Java **64 bit** and Eclipse **64 bit** (this is not recommended), in order to see the output of your application in the **Console** view, you need to follow the steps below:

- ❖ Select **Run** → **Run Configurations...**



- ❖ Click on the **Environment** tab and click on **New**.
- ❖ In **Name** field, type **PATH**.
- ❖ In **Value** field, type **c:\cygwin\bin**.
- ❖ Click **OK**.

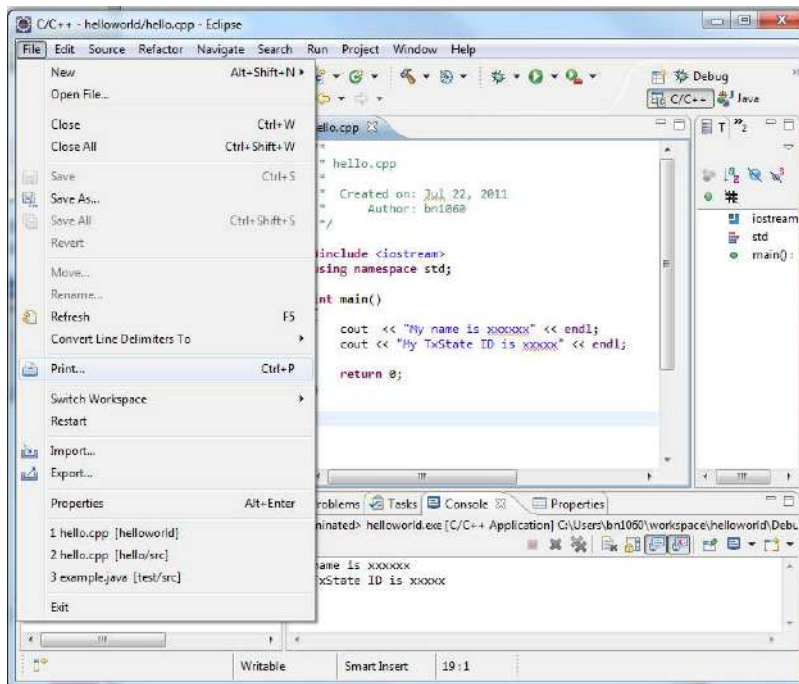


## C. SECTION 2: Printing and Transferring Files

### 1. Part 1: Print Source Codes

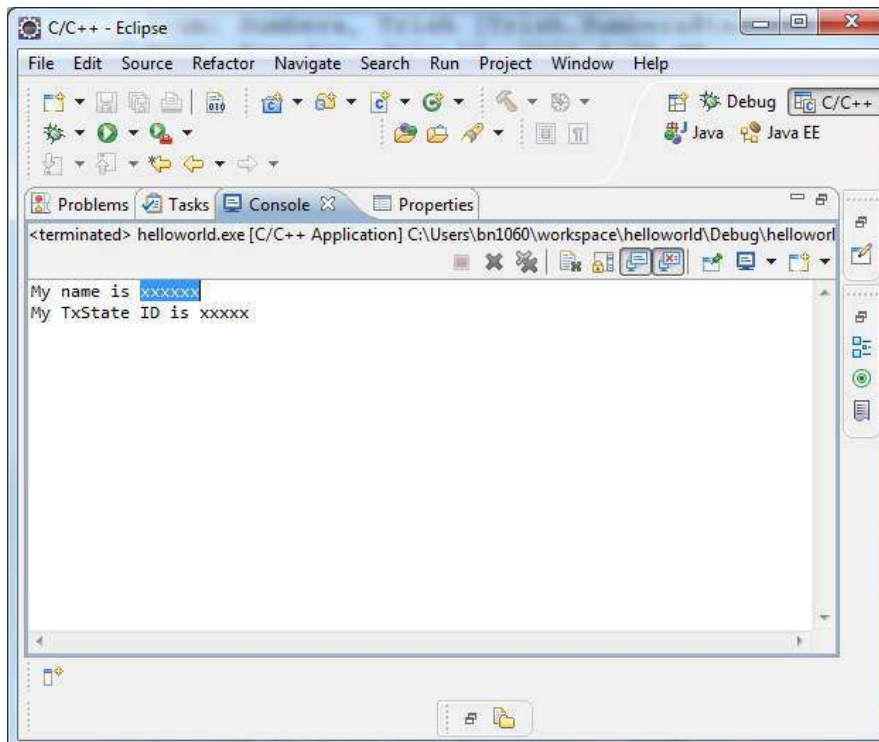
- 1.1. Click on the **Editor** (hello.cpp)
- 1.2. Select **File** → **Print** (or **Ctrl + P**)



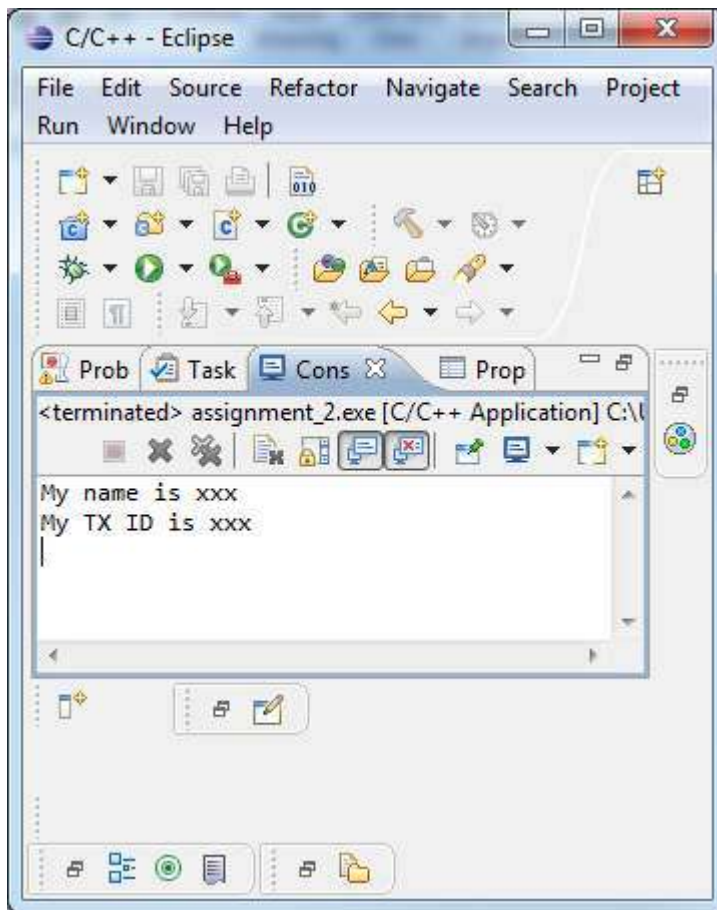


## 2. Part 2: Print screen output

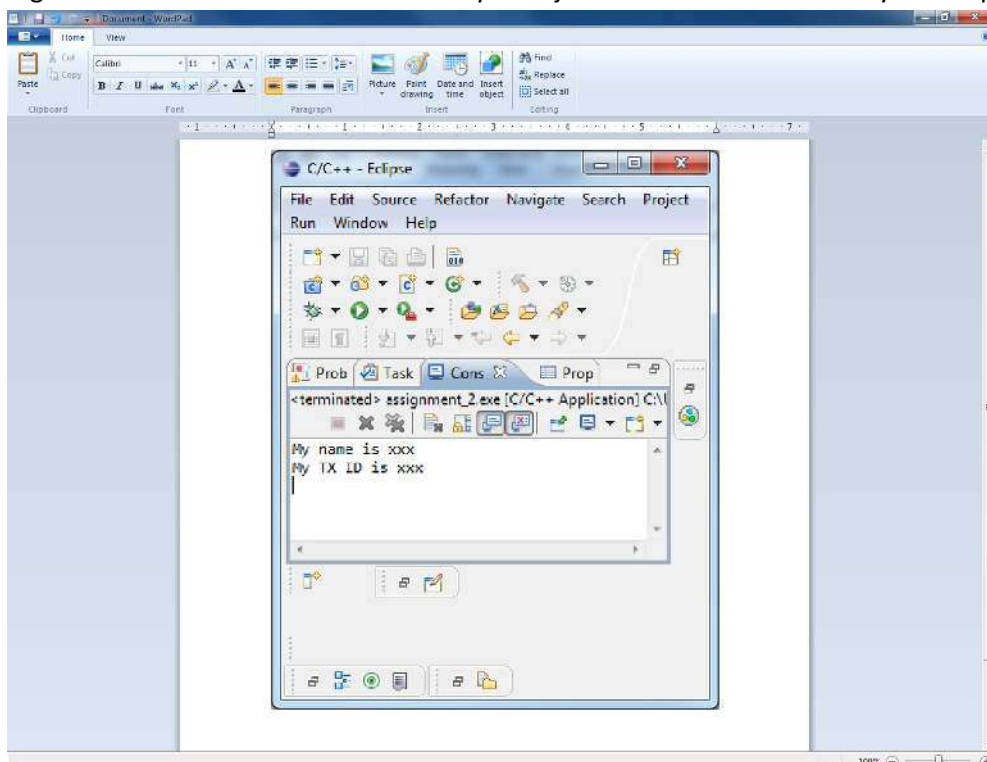
- 2.1. Click on the **Console** view
- 2.2. Click on the **Maximize** button



- 2.3. Resize Eclipse to prevent unnecessary white space.



- 2.4. Take a screen shot of the Console view by pressing **Ctrl + Alt + PrintScreen/SysRq**
- 2.5. Open **WorldPad**
- 2.6. Right click and select **Paste**. Make sure you adjust the screenshot so that your output is **readable**.



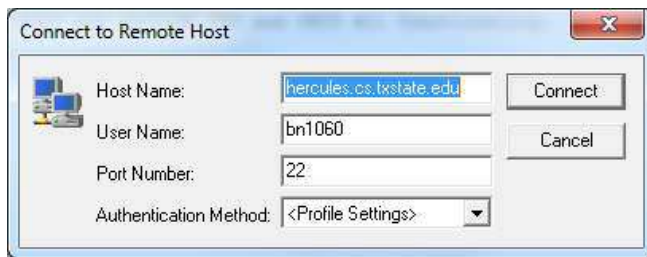
### 3. Part 3: Save and store files using SSH

#### 3.1. Click on the SSH Secure Shell Client icon

If you don't have SSH Secure Shell Client, you can download it free at

[http://downloads.cs.txstate.edu/os/windows/remote\\_access/SSHSecureShellClient-3.2.9.exe](http://downloads.cs.txstate.edu/os/windows/remote_access/SSHSecureShellClient-3.2.9.exe)

#### 3.2. Click on **Quick Connect** to connect to the remote host.



#### 3.3. In the **Host Name** field, type any one of these host names:

- ❖ athena.cs.txstate.edu
- ❖ zeus.cs.txstate.edu
- ❖ eros.cs.txstate.edu
- ❖ hercules.cs.txstate.edu

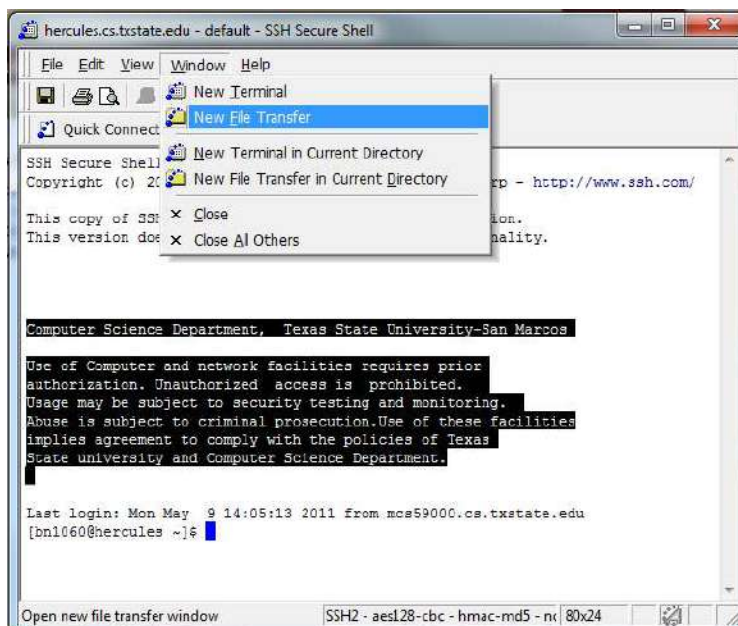
#### 3.4. In the **User Name** field, type your CS Linux username.

#### 3.5. Click **Connect**.

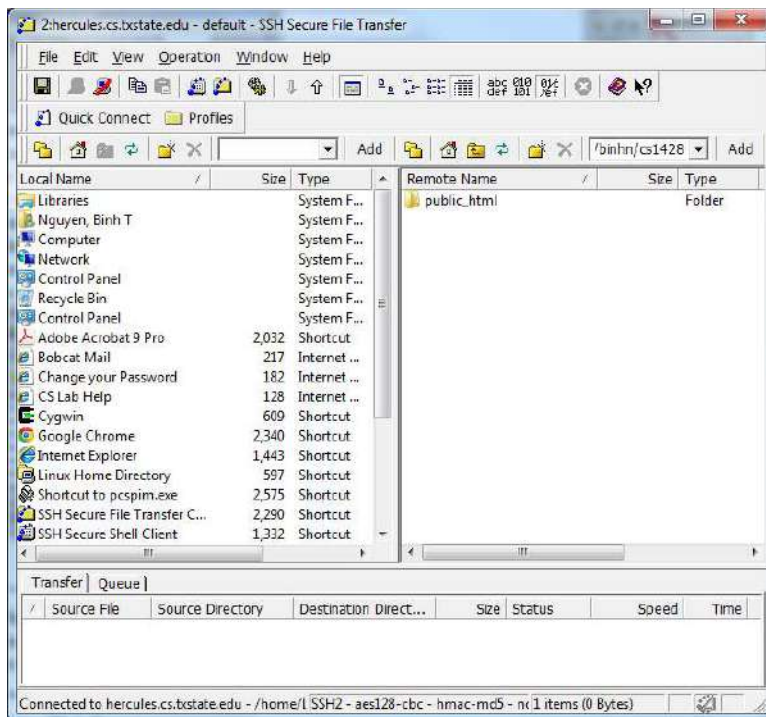
#### 3.6. Type in your CS Linux password.



#### 3.7. Select **Window** → **New File Transfer**



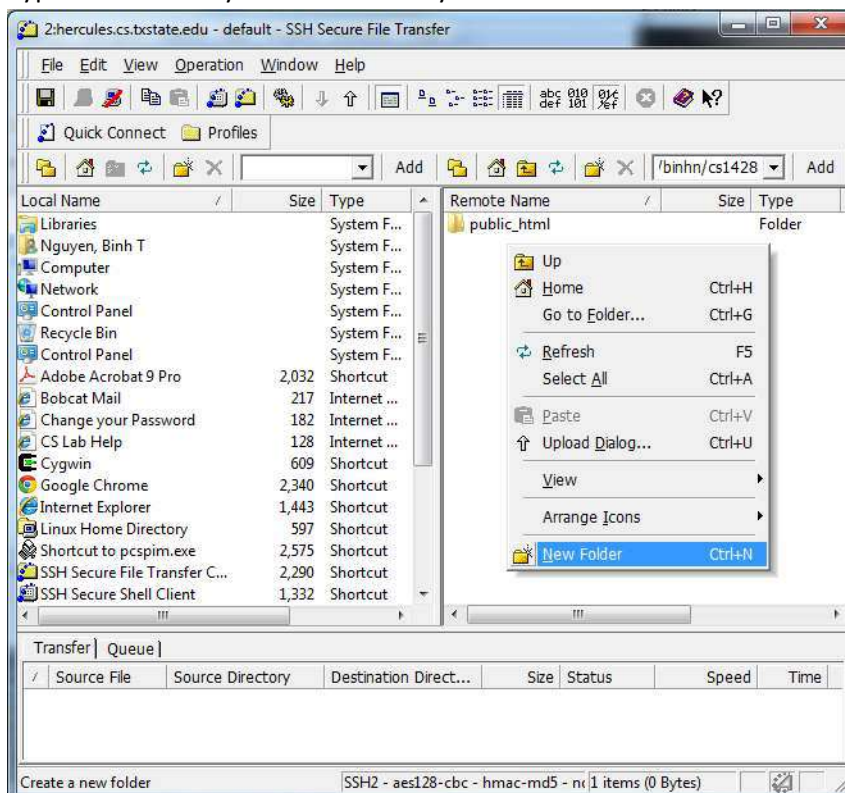
3.8. A SSH Secure File Transfer window will appear.



3.9. The left pane is your local machine. The right pane is the remote host. If you want, you can create a new directory to save your file.

3.10. Right click on the remote host pane and select **New Folder**.

3.11. Type the name of your new directory.



- 3.12. To upload files from your local machine to a remote host or download files from a remote host to your local machine, follow these steps:
- ❖ Navigate to the files you want to download or upload.
  - ❖ Once you have located file(s) you want to download or upload, select them by clicking once on the file name. To select multiple files, hold down the Ctrl key and using your mouse click on each of the files.
  - ❖ Once you have selected the files you want to download or upload, you may simply drag them:
    - from right hand pane to left hand pane and drop them into the desired directory to **download**
    - from left hand pane to right hand pane and drop them into the desired directory to **upload**
  - ❖ When you have finished, select **File** → **Disconnect** and click **OK** to confirm.

## D. SECTION 3: Saving Files via SSH and Adding Files into Project

### Scenario:

- ❖ Assume that you already created a C++ project and source files on a lab computer. You want to keep working on these source files at home or on a different computer. (Because profiles stored on the lab computers will be deleted over the weekend, you should ALWAYS save your source files either via SSH to your Linux home directory or to a flash drive before leaving the lab.)
- ❖ This section will demonstrate:
  - How to locate source files in the project folders and save them via SSH (**part 1**)
  - How to download source files via SSH and import them into an empty project (**part 2**)

### 1. Part 1: Locate source files in the project folder and save them via SSH

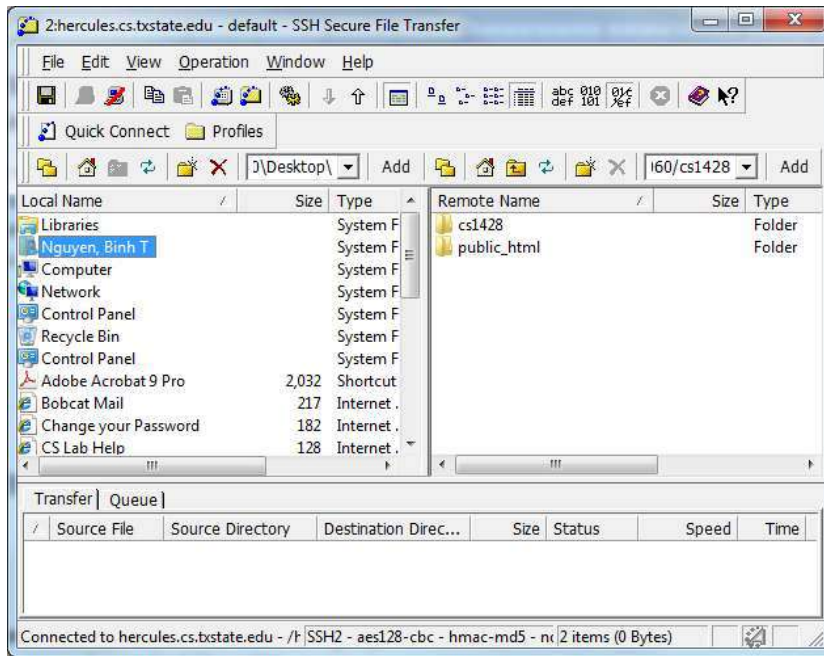
**Note:** If you want to save source files via a **flash drive**, follow the path below to locate them:

```
C:\Users\<your-TxState-ID>\workspace\<project-name>
```

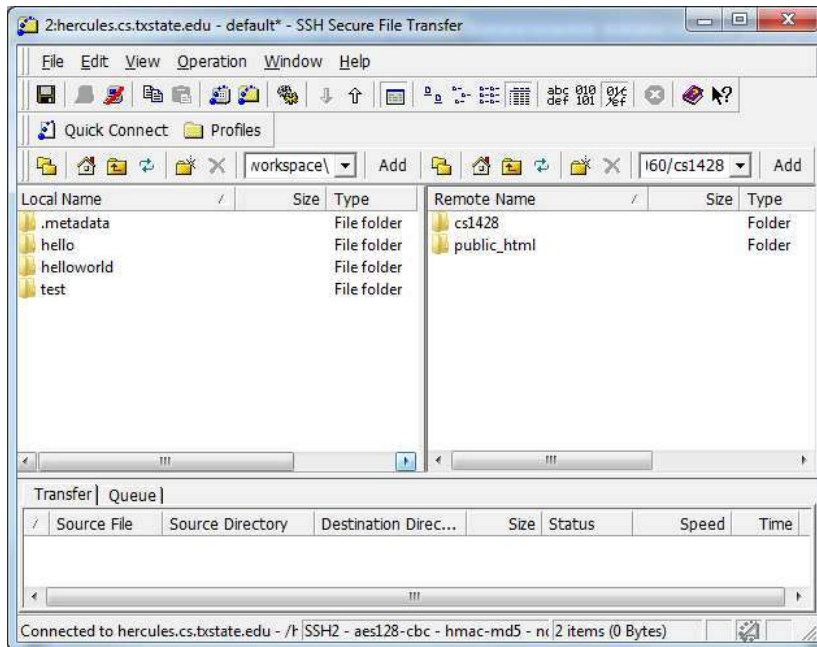
The source files are stored inside the *<project-name>* folder

Log in to your Linux home directory via Secure Shell Client SSH

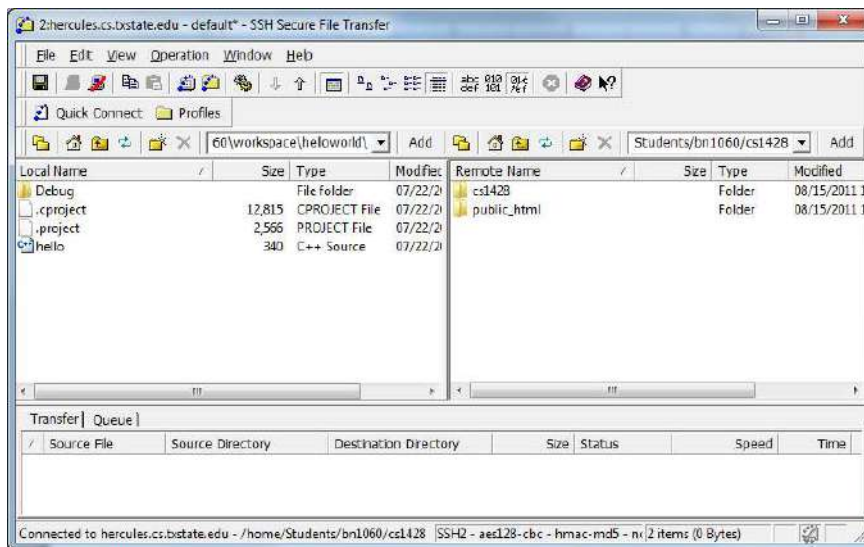
- 1.1. Select **Window** → **New File Transfer**
- 1.2. Select the folder under your name.



1.3. Select **workspace** folder. A list of projects created on the local machine will display.



1.4. Select the project folder which stores your desired source files. For example, **helloworld** folder



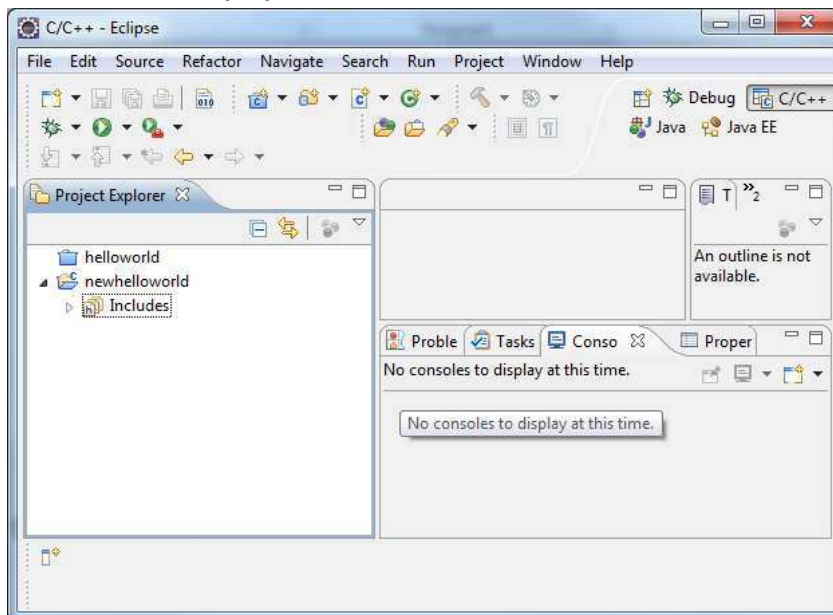
- 1.5. Select the desired source files. Drag and drop them to the desired folder on the right pane (remote pane)

**Note:**

The executable file (if you have one) is stored in the **Debug** folder.

## 2. Part 2: Download source files via SSH and add them into an empty project

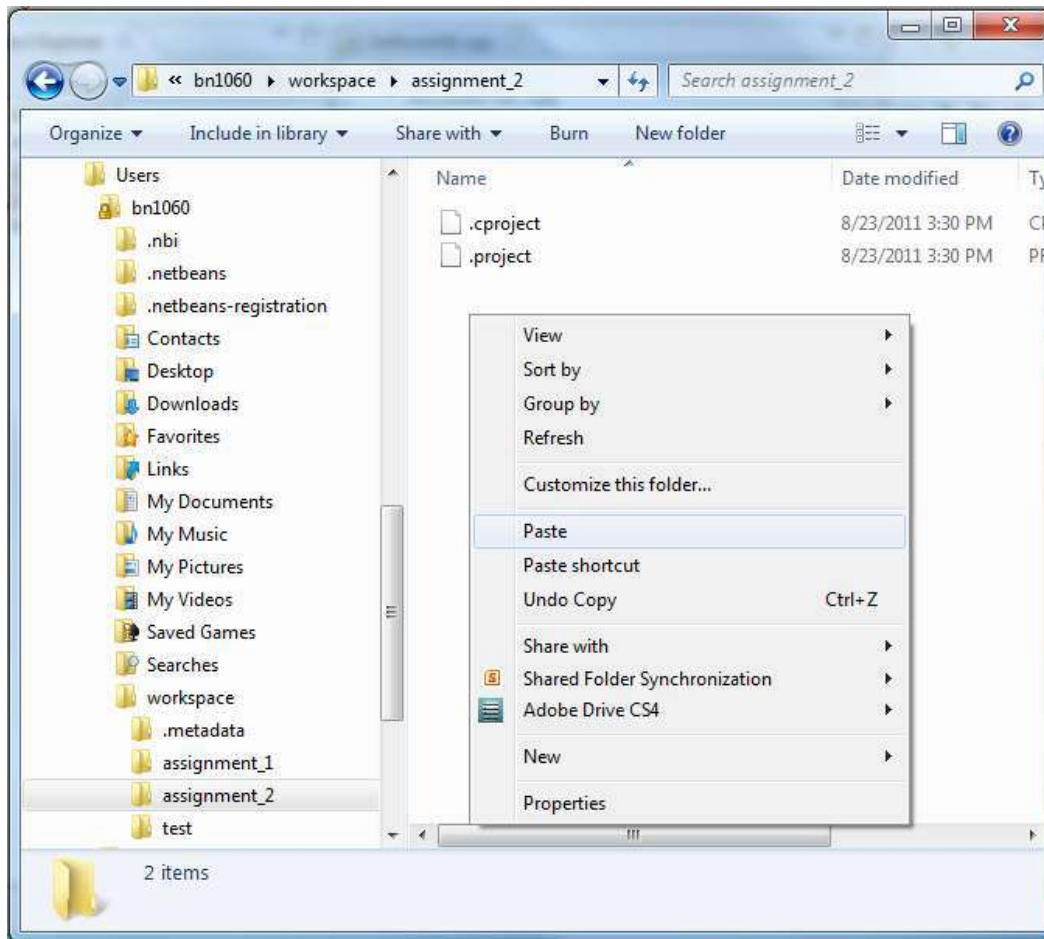
- 2.1. Log in Secure Shell Client SSH
- 2.2. Select **Window** → **New File Transfer**
- 2.3. Locate **source files** saved on the remote access server. Drag and drop them to the **desired folder** on the left pane (local machine). For example, in this tutorial, source files will be saved on the **desktop** of local machine.
- 2.4. Run **Eclipse.exe**
- 2.5. Create a **new C++ project** on local machine.



*For example:* **newhelloworld** project is created.

2.6. To add source files to an empty project in the workspace, copy these source files from Desktop or flash drive to the empty project folder in `C:\Users\<your-TX-ID>\workspace\<project-name>` by following below steps:

- ❖ Open **Windows Explorer**, navigate to source files
- ❖ Select source files, right click and select **Copy** (To select multiple files, hold down the Ctrl key and using your mouse click on each of the files)
- ❖ Navigate to project folder by following the path `C:\Users\<your-TX-ID>\workspace\<projectname>`
- ❖ Right click and select **Paste**



2.7. Go back to **Eclipse**, click on **File** → **Refresh**.



