HY-150 Προγραμματισμός
HY-150 Programming

Tutorial 0.1:
C++ Programming tools
for Windows, Linux, Mac

G. Papagiannakis

Programming tools overview

- Windows
  - Microsoft Visual Studio 2010 express (free)
  - MinGW - Minimalist GNU for Windows (free)
- Linux
  - g++, gdb and a text editor
- Mac
  - g++
  - XCODE

Linux-MacOSX g++ compiler/tools

- Mac OSX
  - Here’s what you need to do to install g++ and gcc on Mac OS X 10.8.
    - Install XCode 4.4 from App Store.
    - Start XCode.
    - Run XCode > Preferences > Downloads > Components > Command Line Tools > Install.
  - Once done you should be able to run g++ and gcc from the command line as follows.
    - tron:~ dhruba$ g++
    - i686-apple-darwin11-llvm-g++-4.2: no input files
  - Otherwise download cmd line tools from apple: https://developer.apple.com/downloads/index.action?=command%20line%20tools

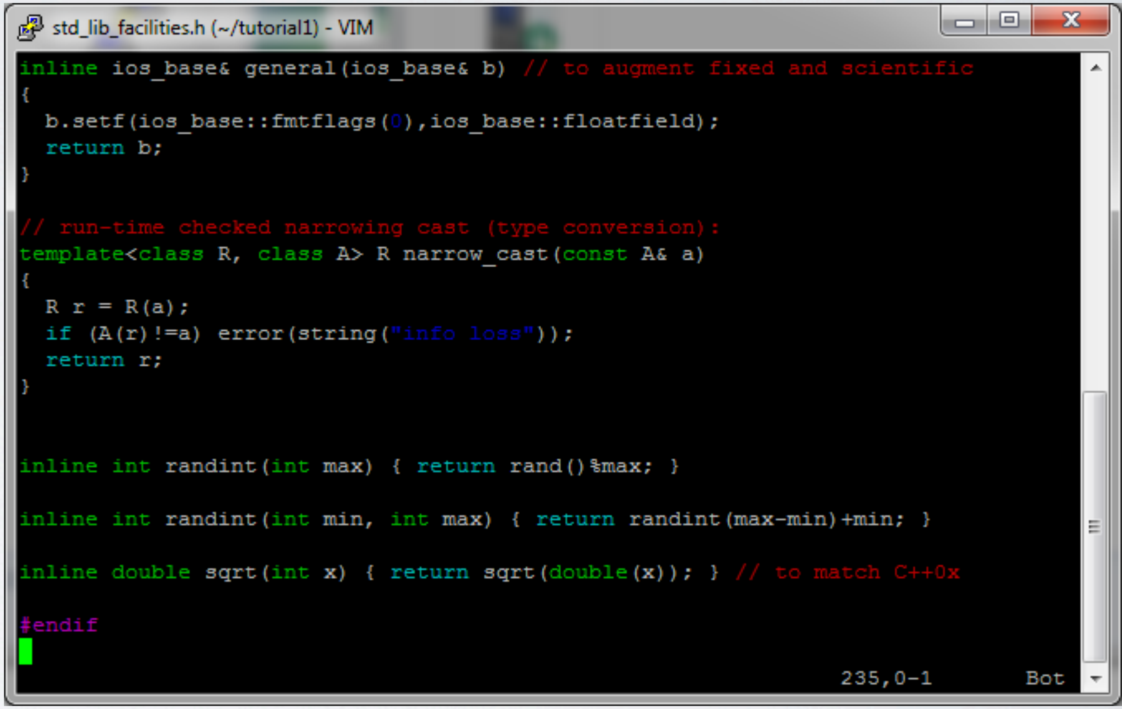
- Linux
- On CSD terminals all C++ tools are installed
- Otherwise (e.g. Ubuntu)
  - sudo apt-get install build-essential checkinstall
  - Or check directly https://help.ubuntu.com/community/CompilingEasyHowTo

Linux, Mac OS (1/4)

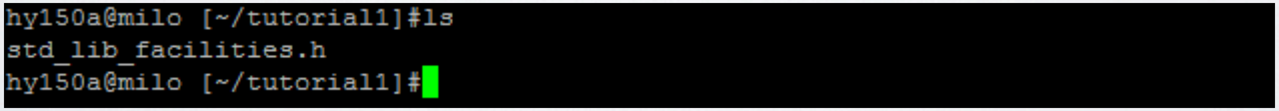
- Open Standard library access header from http://www.stroustrup.com/Programming/
- Select all (Ctrl+A) and copy (Ctrl+C) to clipboard
- Open a text editor and Paste
  - Mac: XCODE, pico, vi, textedit, etc.
  - Linux: vi, nano, pico, Kate, etc.
- Save with name std_lib_facilities.h
- Or otherwise download them from our site, Lecture 1

Linux, Mac OS (2/4)

- Press Escape, then : and write w <filename> Enter
- To quit: Escape :q Enter



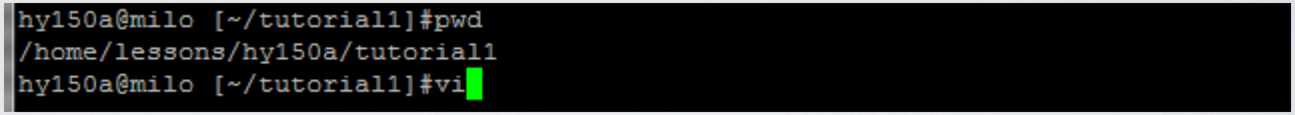
```
inline ios_base& general(ios_base& b) // to augment fixed and scientific
{
  b.setf(ios_base::fmtflags(0),ios_base::floatfield);
  return b;
}

// run-time checked narrowing cast (type conversion):
template<class R, class A> R narrow_cast(const A& a)
{
  R r = R(a);
  if (A(r)!=a) error(string("info loss"));
  return r;
}


inline int randint(int max) { return rand()%max; }

inline int randint(int min, int max) { return randint(max-min)+min; }

inline double sqrt(int x) { return sqrt(double(x)); } // to match C++0x

#endif

                                                         235,0-1      Bot
```



```
hy150a@milo [~/tutorial1]#ls
std_lib_facilities.h
hy150a@milo [~/tutorial1]#
```

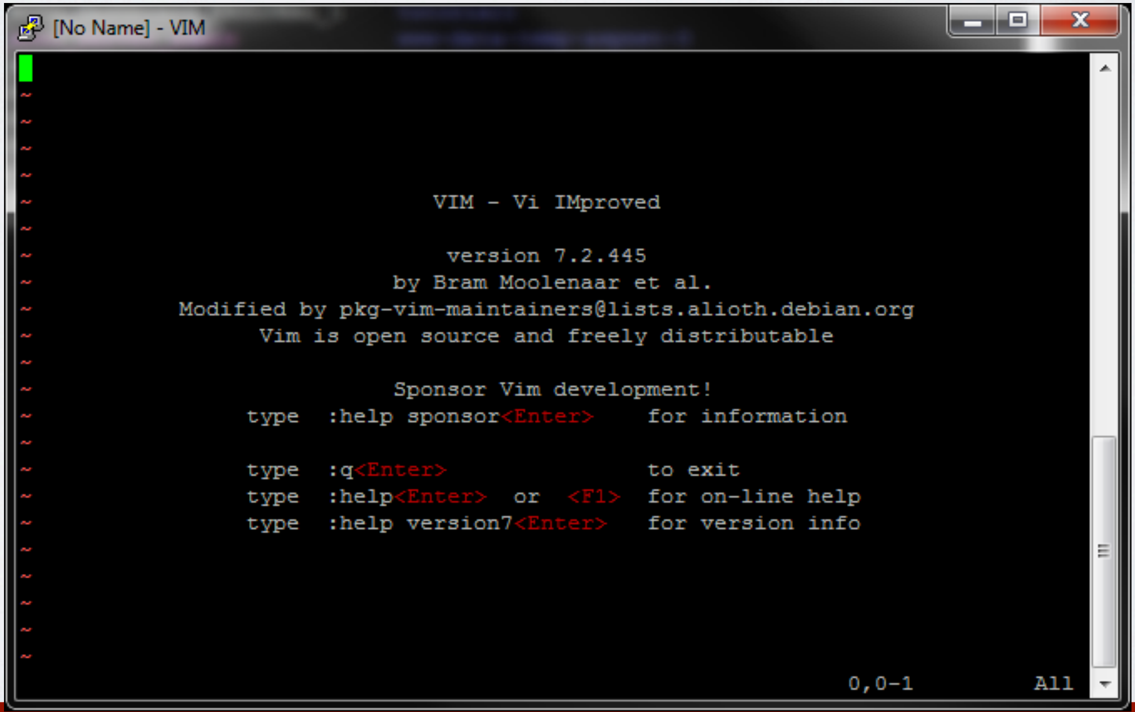
Linux, Mac OS (3/4)

- Start vi



```
hy150a@milo [~/tutorial1]#pwd
/home/lessons/hy150a/tutorial1
hy150a@milo [~/tutorial1]#vi
```

- Press i to enable the input mode



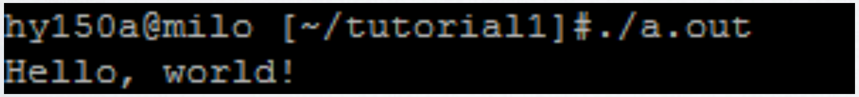
```
~
~
~
~
~                      VIM - Vi IMproved
~
~                       version 7.2.445
~                   by Bram Moolenaar et al.
~       Modified by pkg-vim-maintainers@lists.alioth.debian.org
~             Vim is open source and freely distributable
~
~                   Sponsor Vim development!
~         type  :help sponsor<Enter>    for information
~
~         type  :q<Enter>               to exit
~         type  :help<Enter>  or  <F1>  for on-line help
~         type  :help version7<Enter>   for version info
~
~
~
~
~
                                                 0,0-1         All
```

Linux, Mac OS (4/4)

- Follow same instructions to create the Hello World.cpp
- Compile and run
  - g++ HelloWorld.cpp
    - The default executable name is a.out.

    Try g++ –o MyFirstProgram HelloWorld.cpp

    To change executable name to “MyFirstProgram”
  - ./a.out
  - ./MyFirstProgram



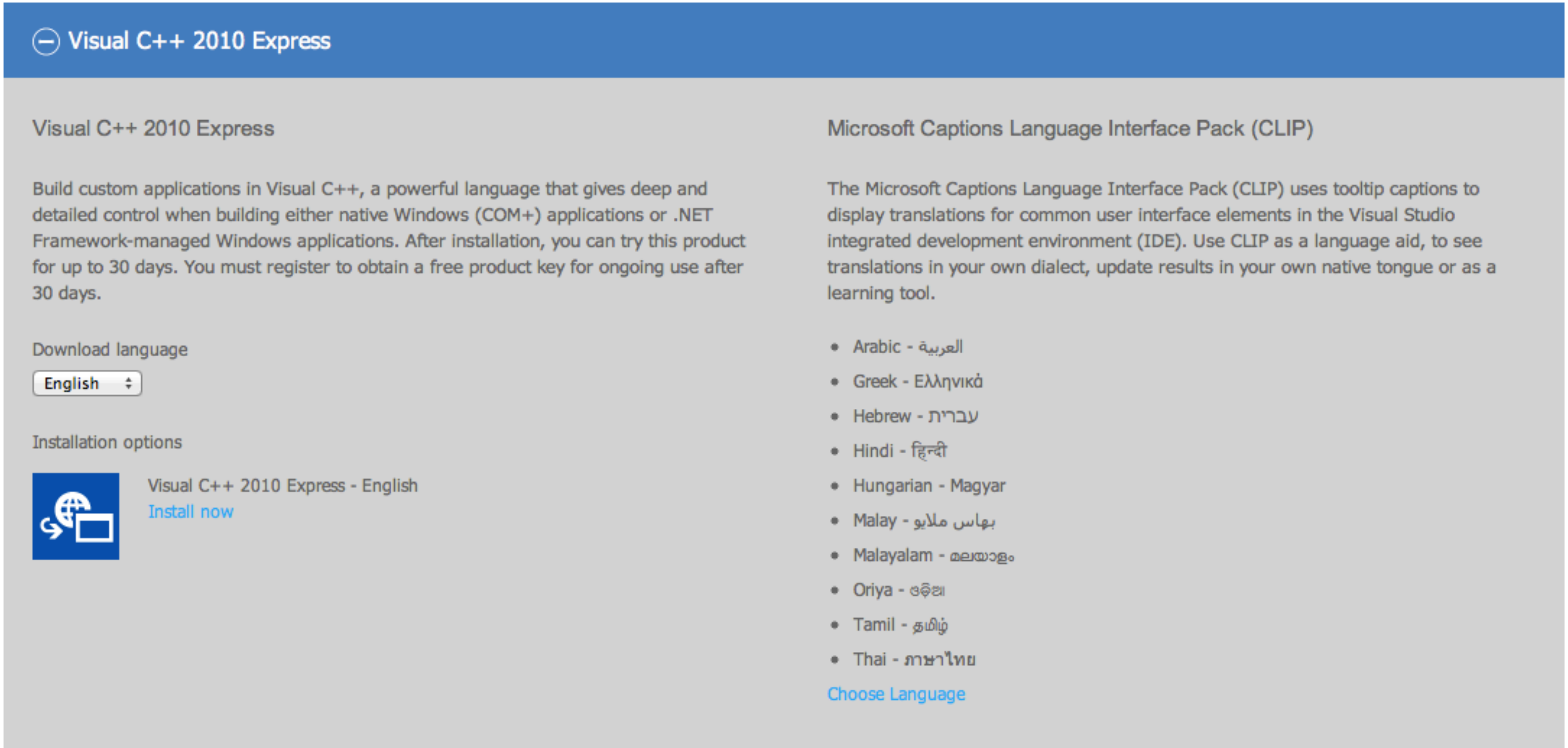
```
hy150a@milo [~/tutorial1]#./a.out
Hello, world!
```

Install Visual Studio Express 2010 (1/4)

- http://www.microsoft.com/visualstudio/eng/downloads#d-2010-express
- You can **install for free** Visual Studio 2010 express. **Do not *install the professional trial version or Visual Studio 2012.***

Install Visual Studio Express 2010 (2/4)

- Click INSTALL NOW - ENGLISH



⊖ Visual C++ 2010 Express

Visual C++ 2010 Express

Build custom applications in Visual C++, a powerful language that gives deep and detailed control when building either native Windows (COM+) applications or .NET Framework-managed Windows applications. After installation, you can try this product for up to 30 days. You must register to obtain a free product key for ongoing use after 30 days.

Download language

English ⬍

Installation options

Visual C++ 2010 Express - English
Install now

Microsoft Captions Language Interface Pack (CLIP)

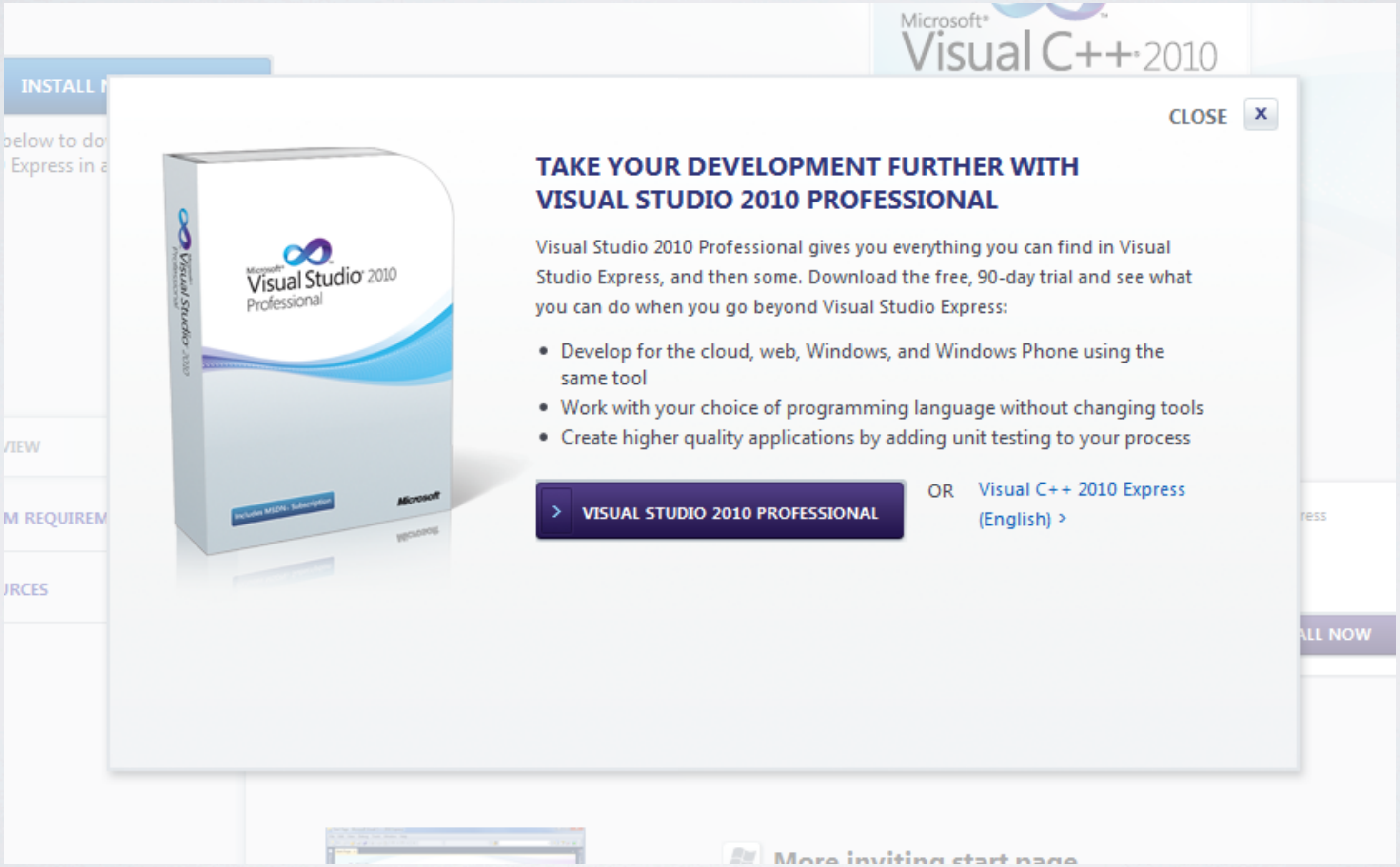
The Microsoft Captions Language Interface Pack (CLIP) uses tooltip captions to display translations for common user interface elements in the Visual Studio integrated development environment (IDE). Use CLIP as a language aid, to see translations in your own dialect, update results in your own native tongue or as a learning tool.

- Arabic - العربية
- Greek - Ελληνικά
- Hebrew - עברית
- Hindi - हिन्दी
- Hungarian - Magyar
- Malay - بهاس ملايو
- Malayalam - മലയാളം
- Oriya - ଓଡ଼ିଆ
- Tamil - தமிழ்
- Thai - ภาษาไทย

Choose Language

Install Visual Studio Express 2010 (3/4)

- Click the Visual C++ 2010 Express (English) button on the popup window that appears.



TAKE YOUR DEVELOPMENT FURTHER WITH VISUAL STUDIO 2010 PROFESSIONAL

Visual Studio 2010 Professional gives you everything you can find in Visual Studio Express, and then some. Download the free, 90-day trial and see what you can do when you go beyond Visual Studio Express:

- Develop for the cloud, web, Windows, and Windows Phone using the same tool
- Work with your choice of programming language without changing tools
- Create higher quality applications by adding unit testing to your process

Microsoft® Visual C++ 2010
INSTALL N
below to do
Express in a
CLOSE ×
VISUAL STUDIO 2010 PROFESSIONAL
OR
Visual C++ 2010 Express (English) >
Visual Studio 2010 Professional
VIEW
M REQUIREM
URCES
ALL NOW
More inviting start page

Install Visual Studio Express 2010 (4/4)

- Wait for downloading process to be completed.
- Find exe file and click to start
  - follow directions

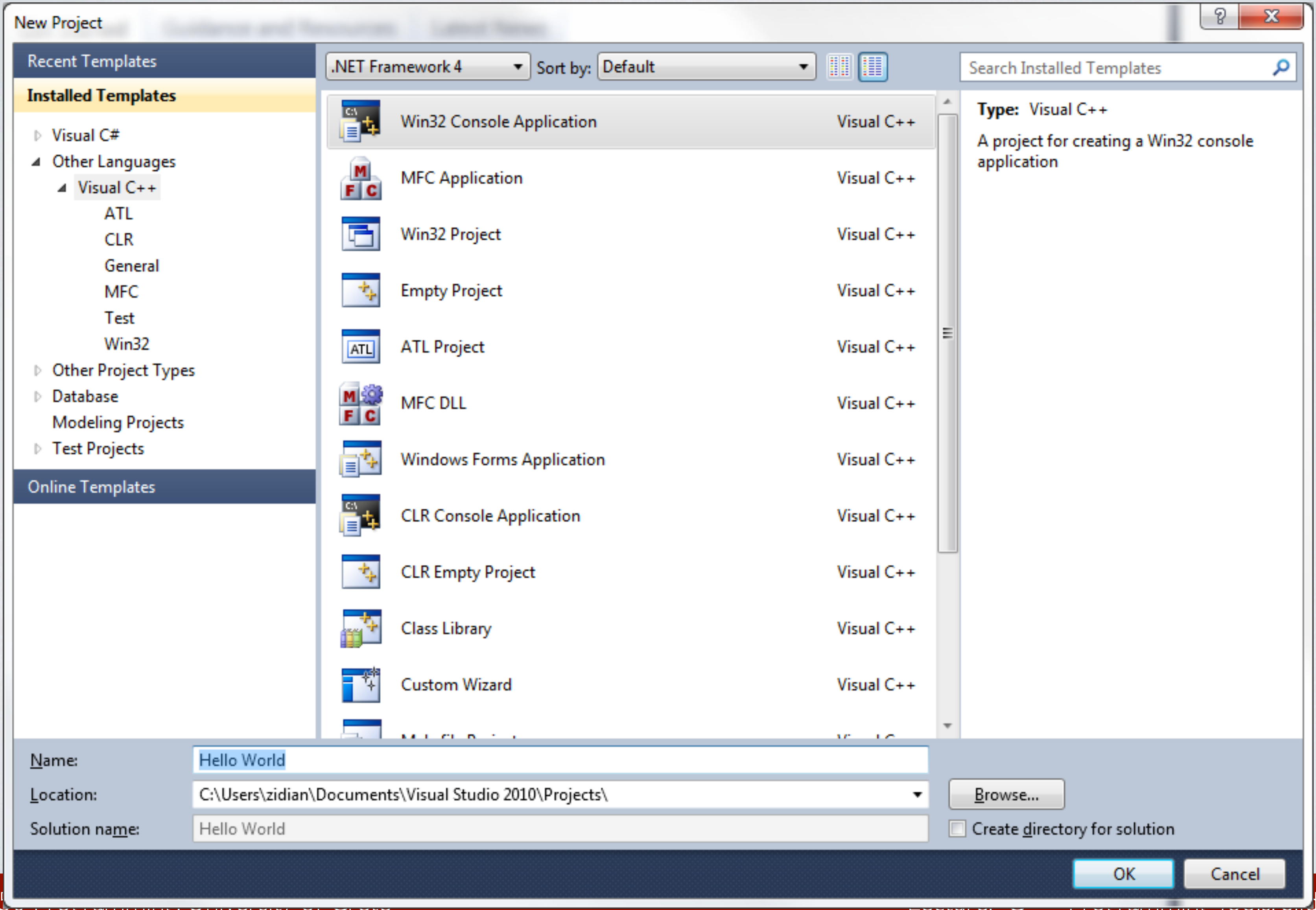
Creating and running a program

- The steps are:
  - 1. **Create a new project.**
  - 2. Add a C++ source file to the project.
  - 3. Enter your source code.
  - 4. Include “std_lib_facilities.h” to the project
  - 5. Build an executable file.
  - 6. Execute the program.

Create a new project (1/5)

- Open the Visual C++ IDE by clicking the Microsoft Visual Studio 2010 icon
- Open the File menu, point to New, and click Project.
- Under Project Types, select Visual C++.
- In the Templates section, select Win32 Console Application
- In the Name text box type the name of your project, for example, Hello World.
- Choose a directory for your project. The default, C:\Documents and Settings\Your Name\My Documents\Visual Studio 2010 Projects, is usually a good choice.

Create a new project (2/5)



Type: Visual C++

A project for creating a Win32 console application

Name: Hello World

Location: C:\Users\zidian\Documents\Visual Studio 2010\Projects\

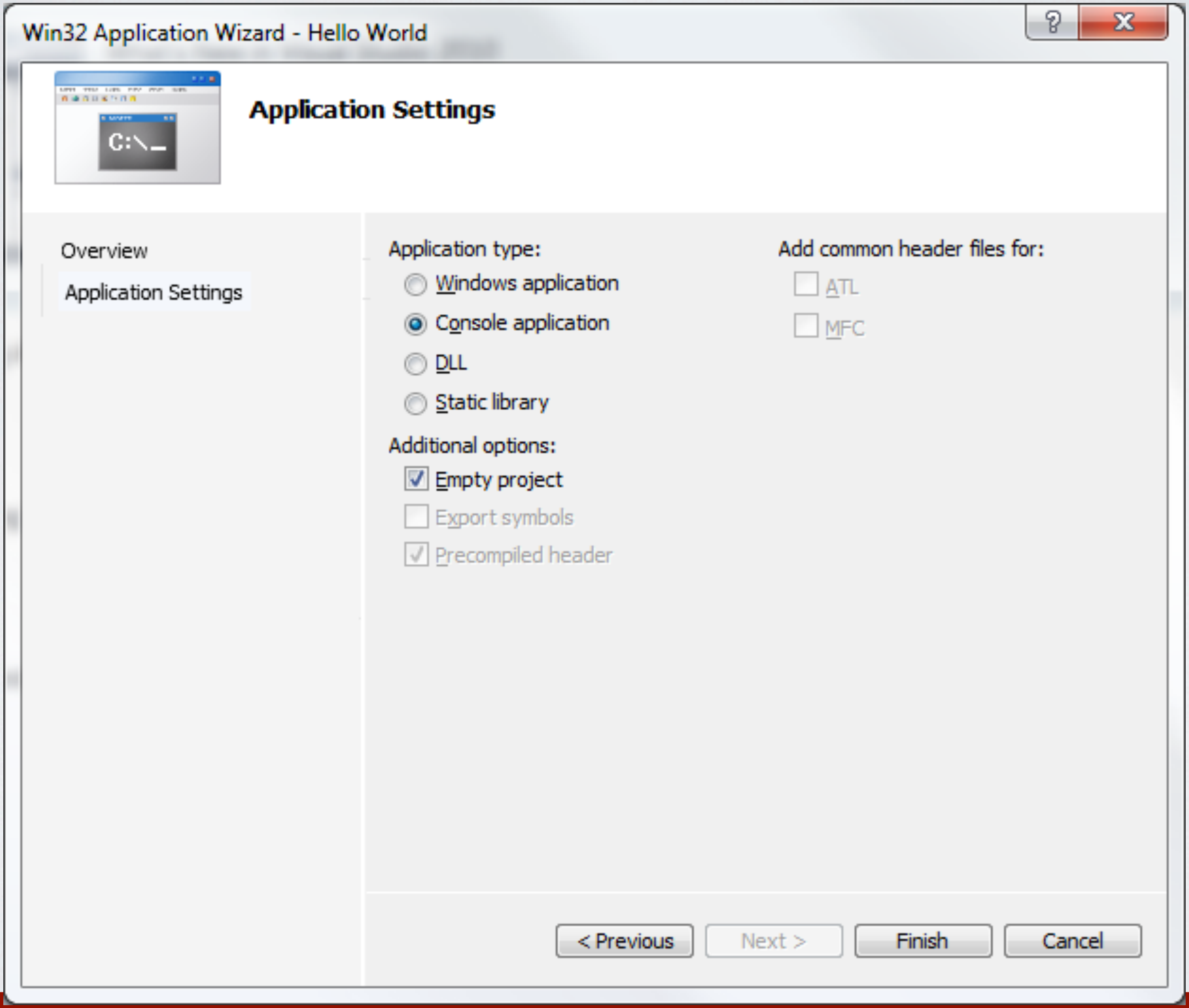
Solution name: Hello World

□ Create directory for solution

Create a new project (3/5)

- Click OK.
- The WIN32 Application Wizard should appear.
- Select Application Settings on the left side of the dialog box.
- Under Additional Options select Empty Project

Create a new project (4/5)



Win32 Application Wizard - Hello World

Application Settings

Application type:

- ○ Windows application
- ● Console application
- ○ DLL
- ○ Static library

Additional options:

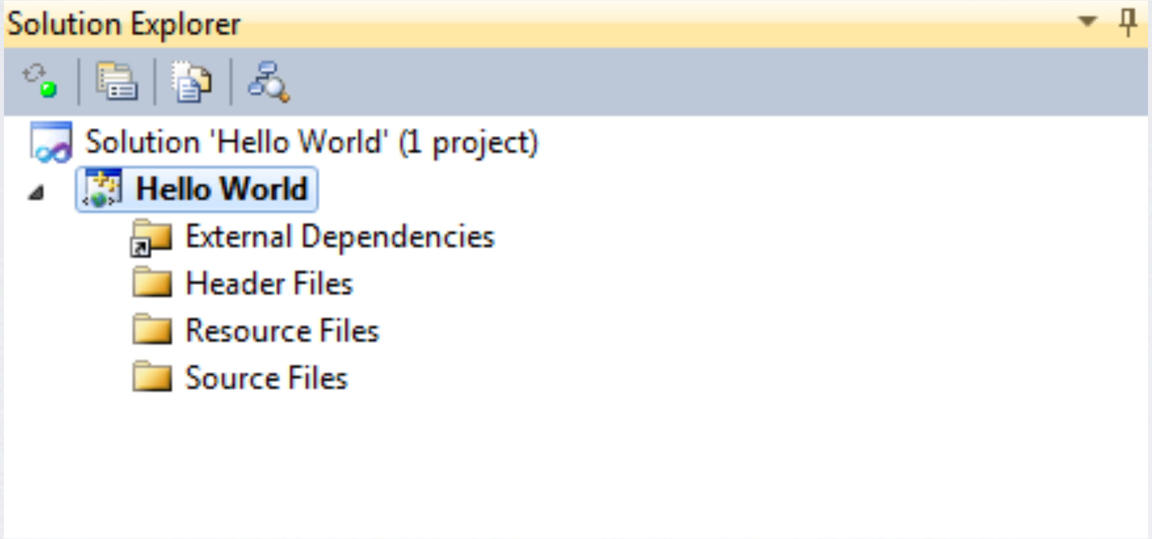
- ☑ Empty project
- □ Export symbols
- ☑ Precompiled header

Add common header files for:

- □ ATL
- □ MFC

Create a new project (5/5)

- Click Finish. All compiler settings should now be initialized for your console project.



Solution Explorer

Solution 'Hello World' (1 project)

Hello World

External Dependencies

Header Files

Resource Files

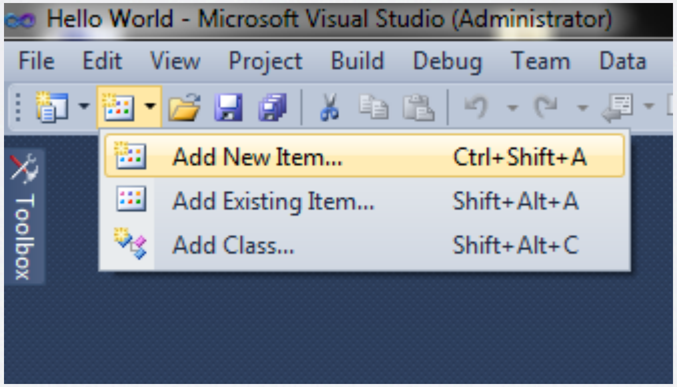
Source Files

Creating and running a program

- The steps are:
  - 1. Create a new project.
  - 2. **Add a C++ source file to the project.**
  - 3. Enter your source code.
  - 4. Include “std_lib_facilities.h” to the project
  - 5. Build an executable file.
  - 6. Execute the program.

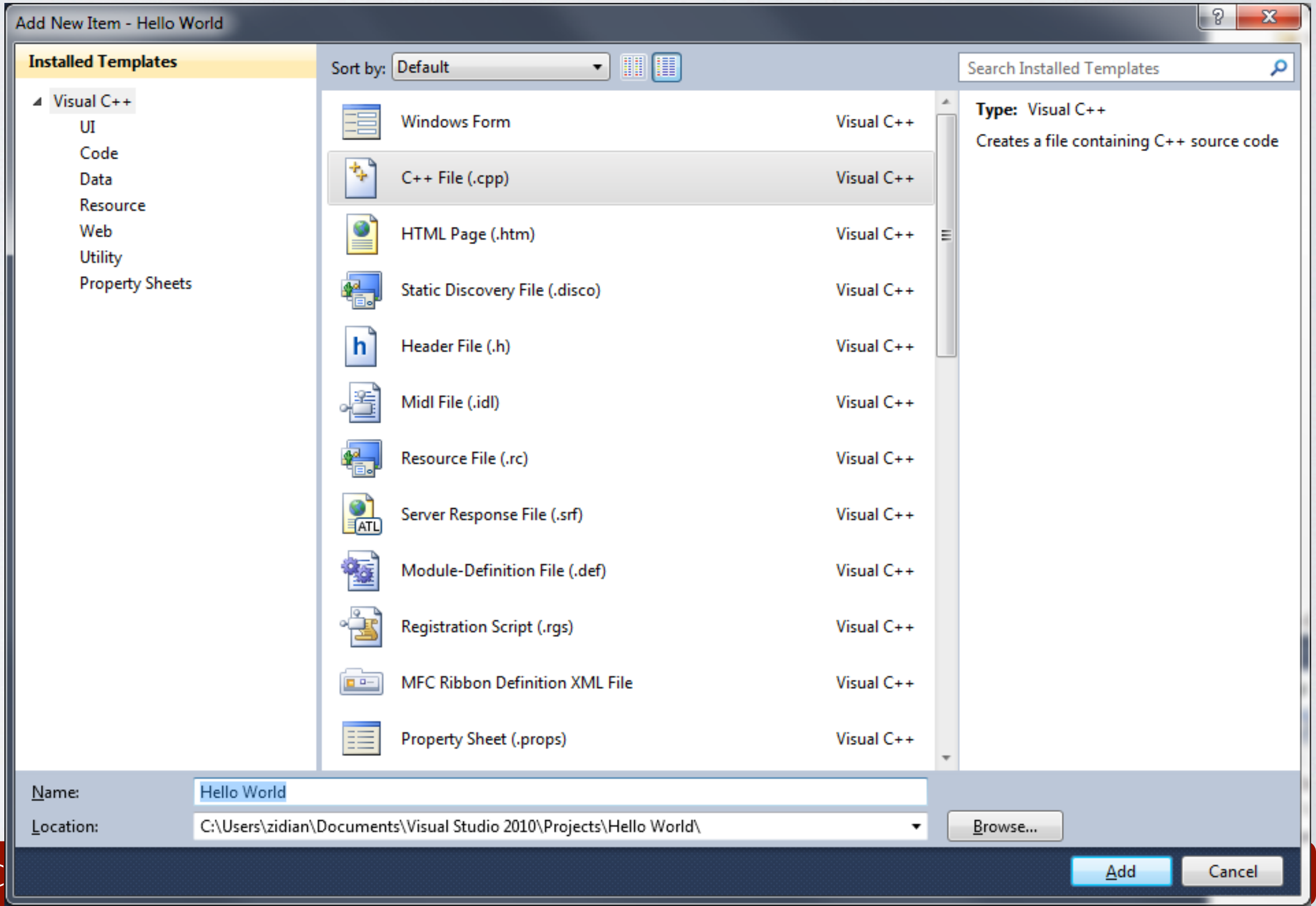
Add a c++ source file to the project (1/3)

- Click the Add New Item icon on the menu bar (usually the second icon from the left). That will open the Add New Item dialog box. Select Code under the Visual C++ category.


Hello World - Microsoft Visual Studio (Administrator)
File
Edit
View
Project
Build
Debug
Team
Data
Add New Item... Ctrl+Shift+A
Add Existing Item... Shift+Alt+A
Add Class... Shift+Alt+C
Toolbox

Add a c++ source file to the project (2/3)

- Select the C++ File icon in the template window. Type the name of your program file (Hello World) in the Name text box and click Add.



Add New Item - Hello World

Installed Templates

- Visual C++
  - UI
  - Code
  - Data
  - Resource
  - Web
  - Utility
  - Property Sheets

Sort by: Default

Windows Form	Visual C++
C++ File (.cpp)	Visual C++
HTML Page (.htm)	Visual C++
Static Discovery File (.disco)	Visual C++
Header File (.h)	Visual C++
Midl File (.idl)	Visual C++
Resource File (.rc)	Visual C++
Server Response File (.srf)	Visual C++
Module-Definition File (.def)	Visual C++
Registration Script (.rgs)	Visual C++
MFC Ribbon Definition XML File	Visual C++
Property Sheet (.props)	Visual C++

Type: Visual C++

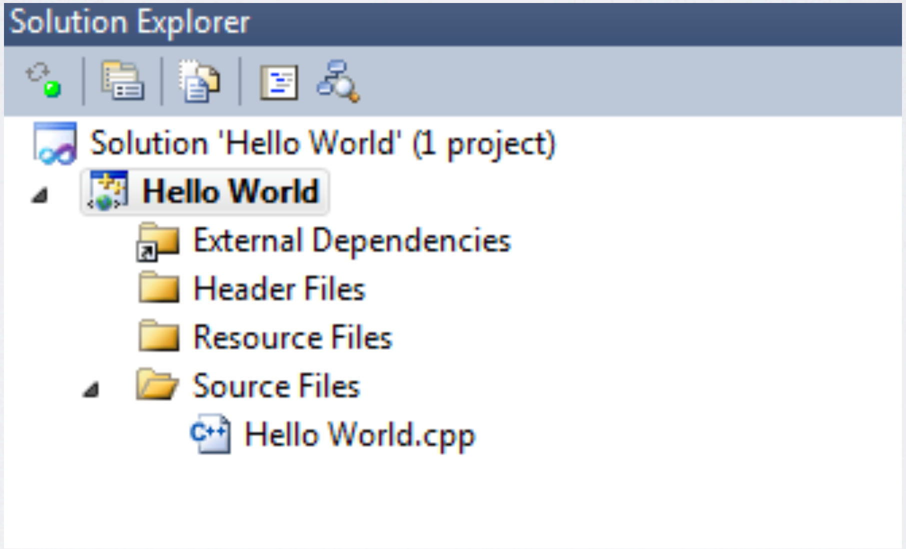
Creates a file containing C++ source code

Name: Hello World

Location: C:\Users\zidian\Documents\Visual Studio 2010\Projects\Hello World\

Add a c++ source file to the project (3/3)

- You have created an empty source code file. You are now ready to type your source code program (later on this tutorial)



Solution Explorer

Solution 'Hello World' (1 project)

Hello World

External Dependencies

Header Files

Resource Files

Source Files

Hello World.cpp

Creating and running a program

- The steps are:
  - 1. Create a new project.
  - 2. Add a C++ source file to the project.
  - 3. **Enter your source code.**
  - 4. Include “std_lib_facilities.h” to the project
  - 5. Build an executable file.
  - 6. Execute the program.

Enter Hello World source code

Double click on Hello World.cpp and inside the editor write the following:

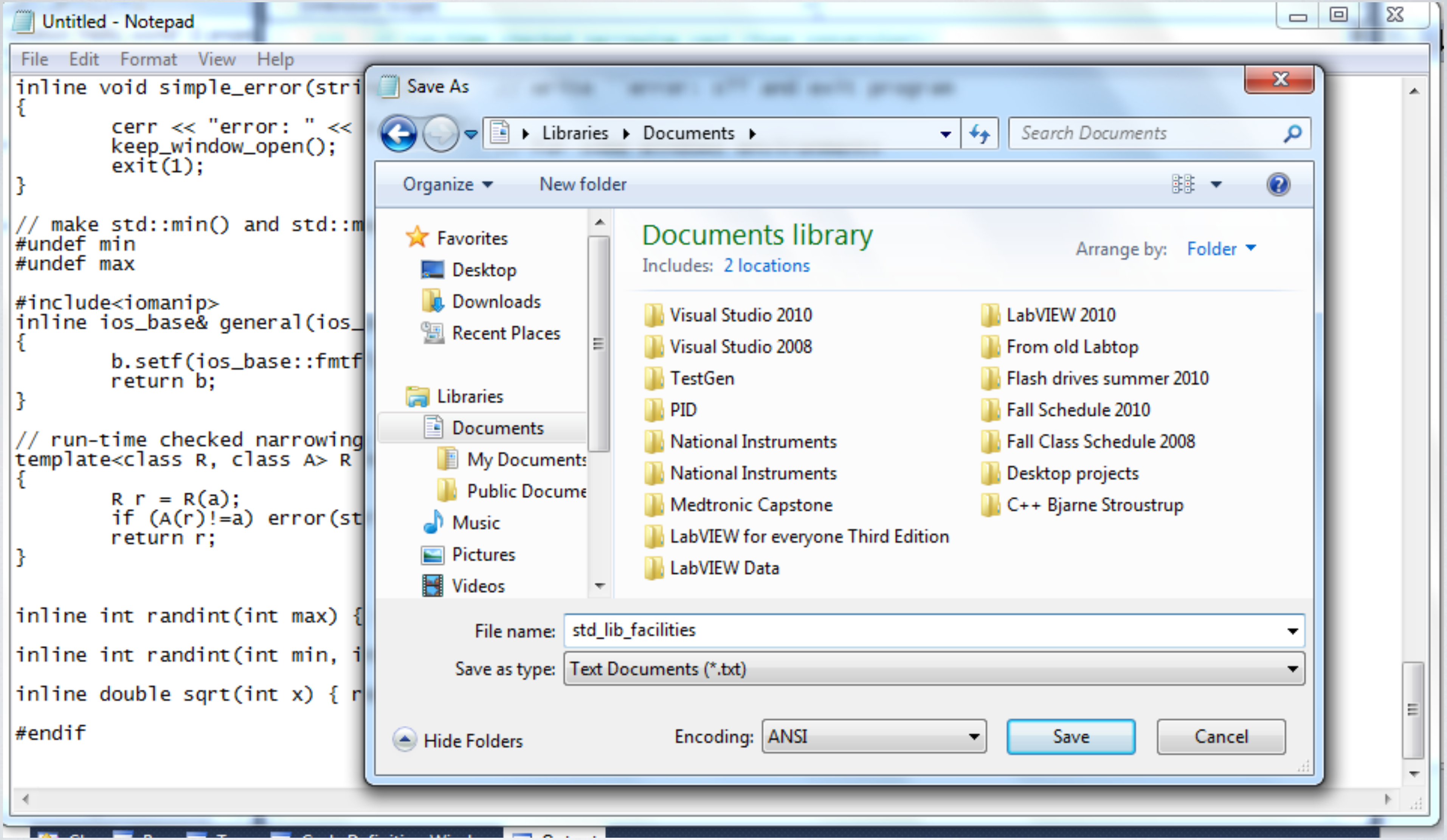
```
//  HY-150 Programming
//  University of Crete
//
//  based on
//  "Programming -- Principles and Practice Using C++" by Bjarne Stroustrup
// This program outputs the message "Hello, World!" to the monitor

#include "../../std_lib_facilities.h"
//--------------------------------------------------------------------------
int main() // C++ programs start by executing the function main
{
    cout << "Hello, world!\n "; // output "Hello, World!"
    return 0;
}
//--------------------------------------------------------------------------
```

Creating and running a program

- The steps are:
  - 1. Create a new project.
  - 2. Add a C++ source file to the project.
  - 3. Enter your source code.
  - 4. **Include "std_lib_facilities.h" to the project**
  - 5. Build an executable file.
  - 6. Execute the program.

How to get your #include "std_lib_facilities.h" working if you are using Visual Studio 2010 express

- Open Standard library access header from http://www.stroustrup.com/Programming/
- and Copy to Notepad.
- Save as **std_lib_facilities.txt**



```
inline void simple_error(stri
{
        cerr << "error: " <<
        keep_window_open();
        exit(1);
}

// make std::min() and std::m
#undef min
#undef max

#include<iomanip>
inline ios_base& general(ios_
{
        b.setf(ios_base::fmtf
        return b;
}

// run-time checked narrowing
template<class R, class A> R
{
        R r = R(a);
        if (A(r)!=a) error(st
        return r;
}


inline int randint(int max) {

inline int randint(int min, i

inline double sqrt(int x) { r

#endif
```

File name: std_lib_facilities

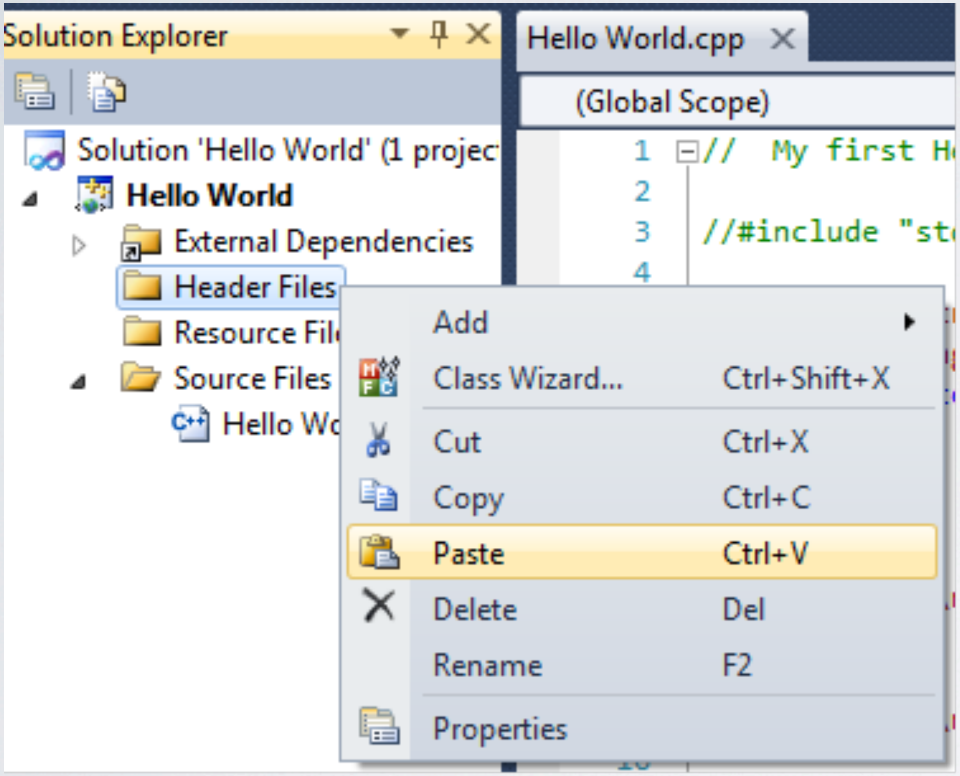
Save as type: Text Documents (*.txt)

Encoding: ANSI

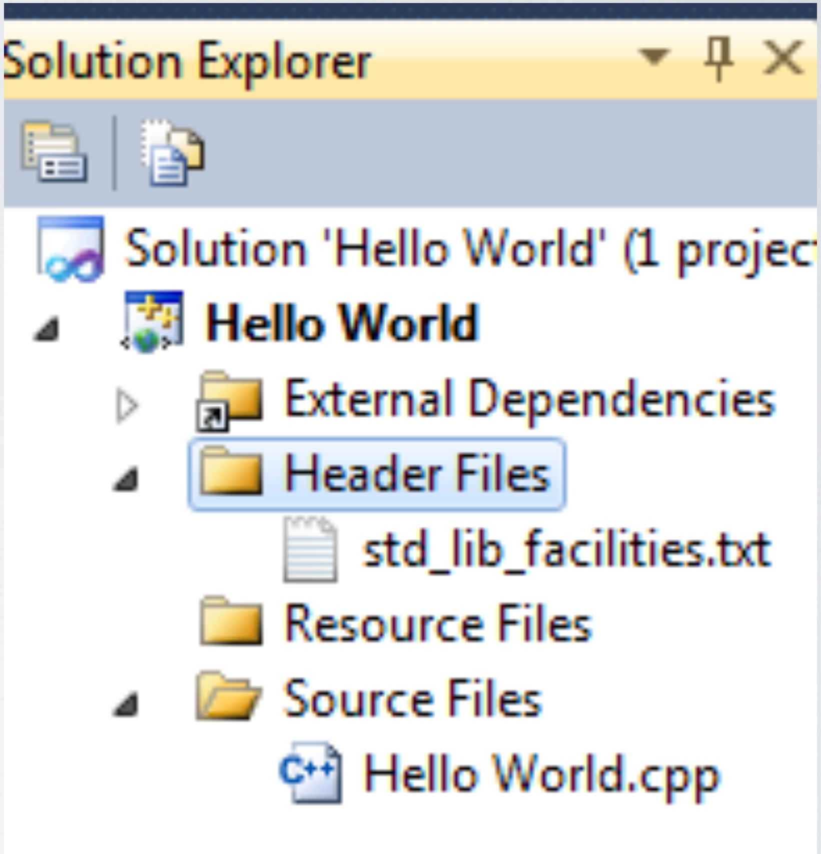
Paste a copy to header folder (1/2)

- For your program to work you must create a header file from your saved **std_lib_facilities** text file.
- With your first Hello World program opened, you must paste a copy of your **std_lib_facilities** text file to the **header folder**.
- **NOTE:** there are better ways but I found this easy for people who are creating their first program.

Paste a copy to header folder (2/2)



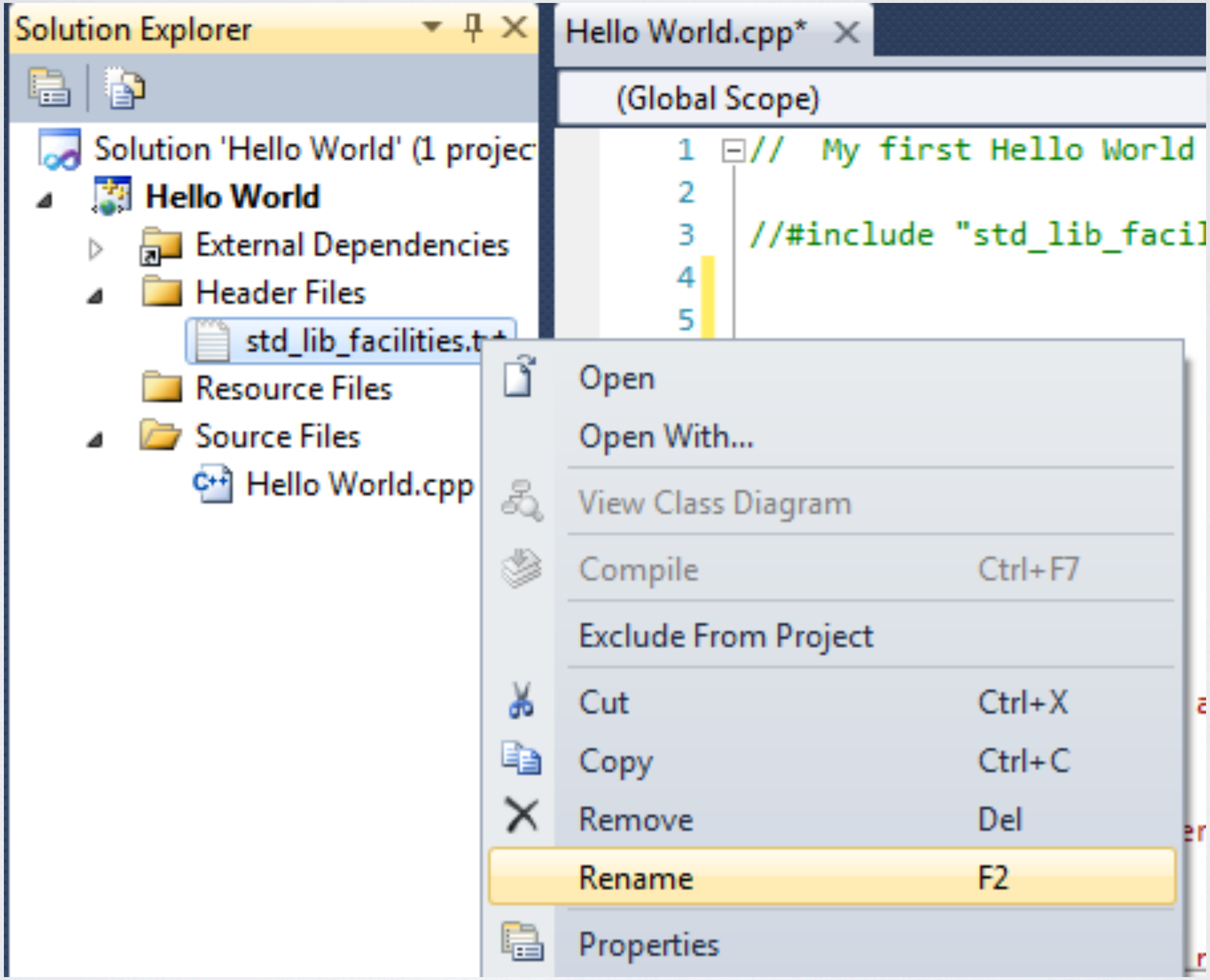
```
// My first H
//#include "st
```



Rename the filename (1/3)

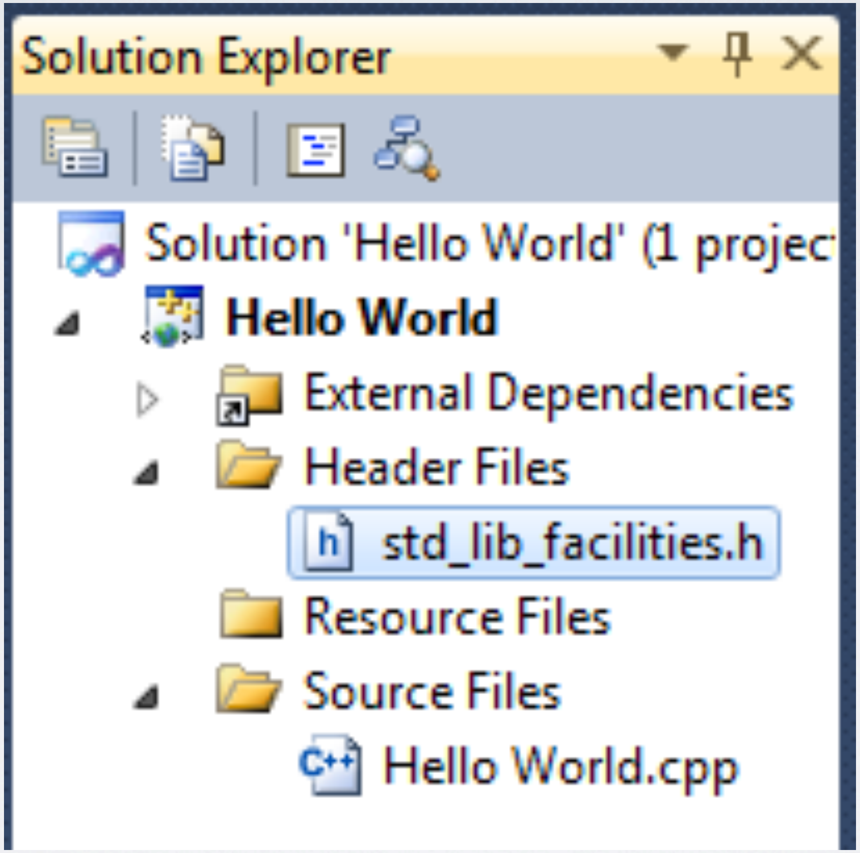
- You must now rename your file extension, instead of .txt change the extension to .h
- Right click on **std_lib_facilities** and click rename; remove the txt and insert h, you will see it changes to a .h header file

Rename the filename (2/3)



```
// My first Hello World

//#include "std_lib_facil
```

Rename the filename (3/3)



Solution Explorer

Solution 'Hello World' (1 projec

Hello World

External Dependencies

Header Files

std_lib_facilities.h

Resource Files

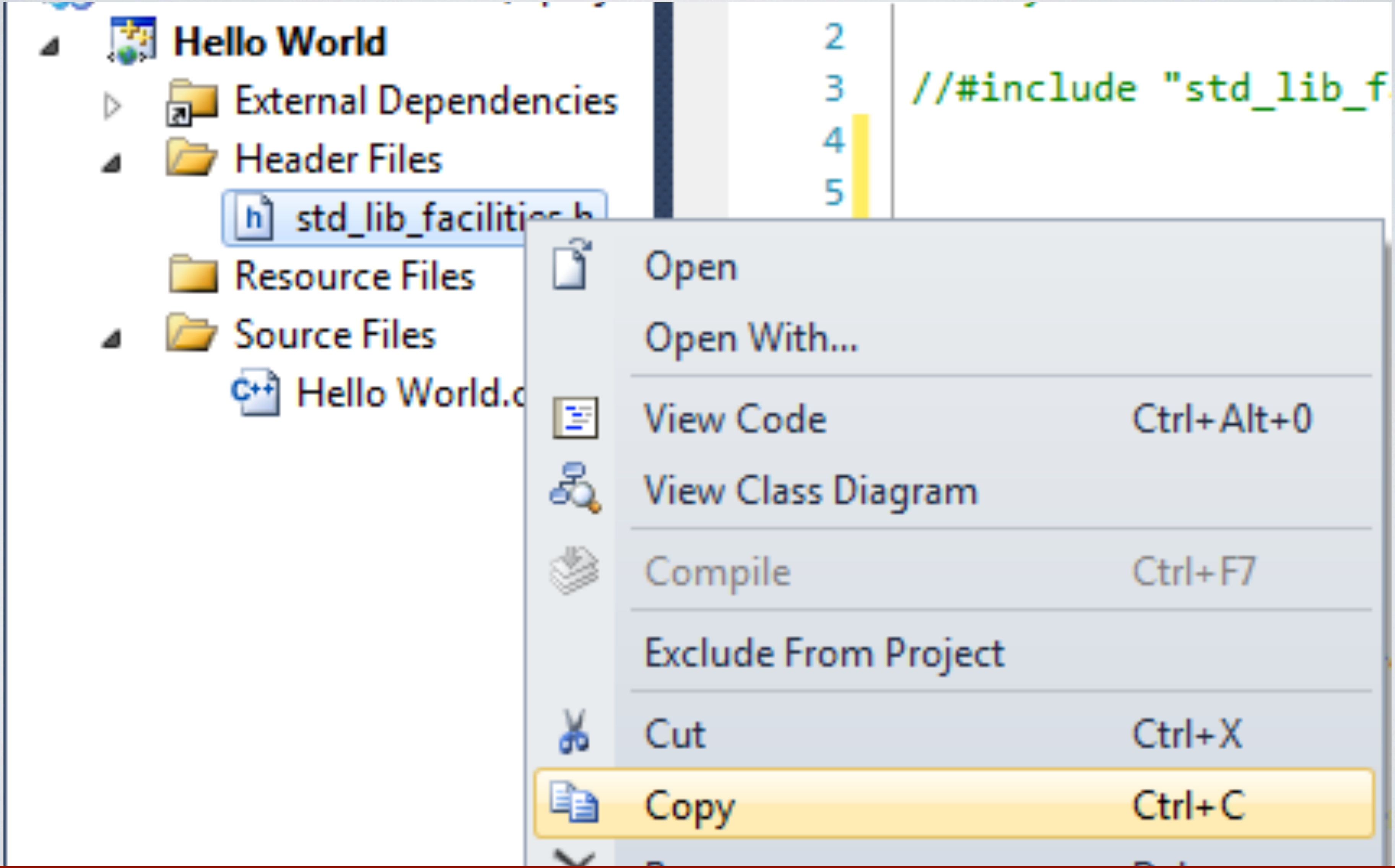
Source Files

Hello World.cpp

Copy to Visual Studio folder (1/3)

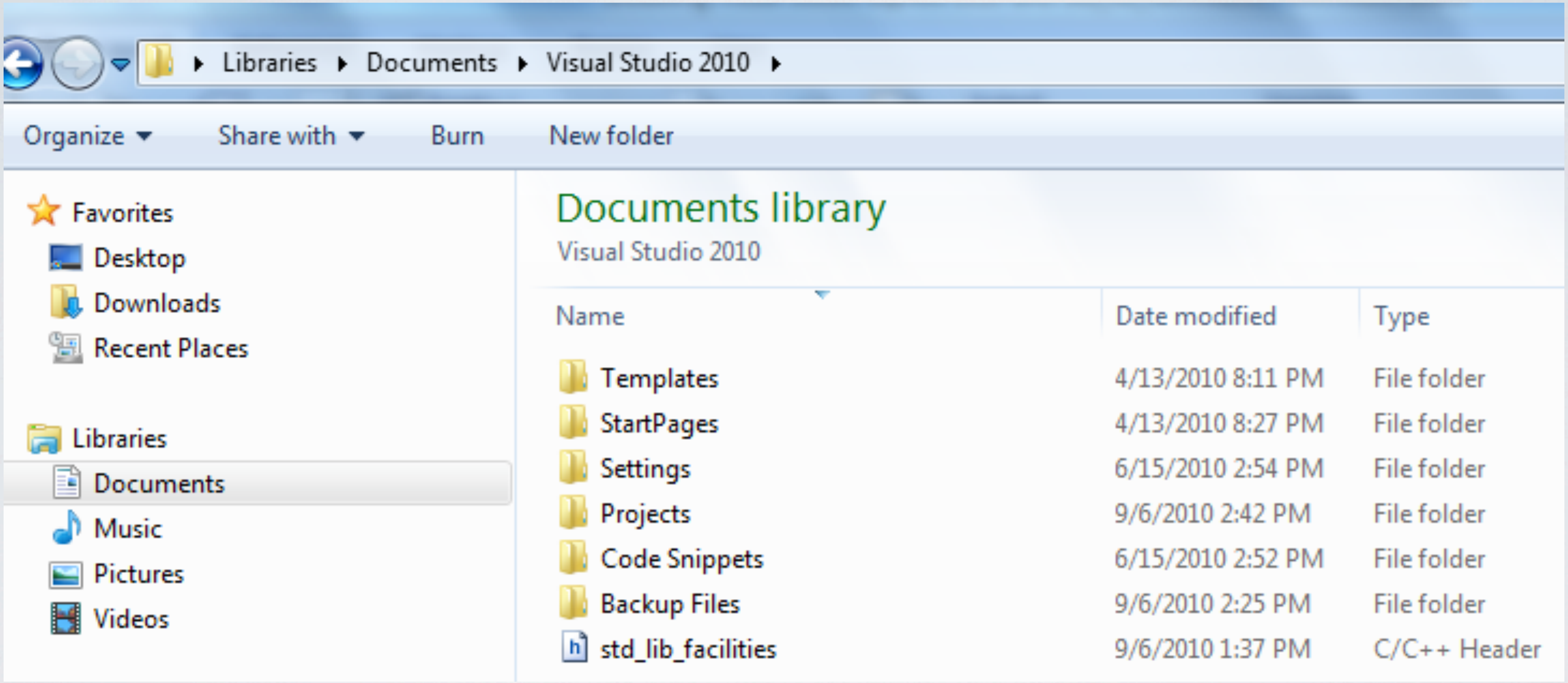
- You will need to copy this new std_lib_facilities.h and place it in Visual Studio 2010 folder
- you will see the file type is now a C/C++ header

Copy to Visual Studio folder (2/3)



```
//#include "std_lib_f
```

Copy to Visual Studio folder (3/3)



Documents library

Visual Studio 2010

Name	Date modified	Type
Templates	4/13/2010 8:11 PM	File folder
StartPages	4/13/2010 8:27 PM	File folder
Settings	6/15/2010 2:54 PM	File folder
Projects	9/6/2010 2:42 PM	File folder
Code Snippets	6/15/2010 2:52 PM	File folder
Backup Files	9/6/2010 2:25 PM	File folder
std_lib_facilities	9/6/2010 1:37 PM	C/C++ Header

- If paste does not work, locate std_lib_facilities.h with Explorer, copy (Ctrl+C) and paste (Ctrl+V) to that folder (Visual Studio 2010 folder)

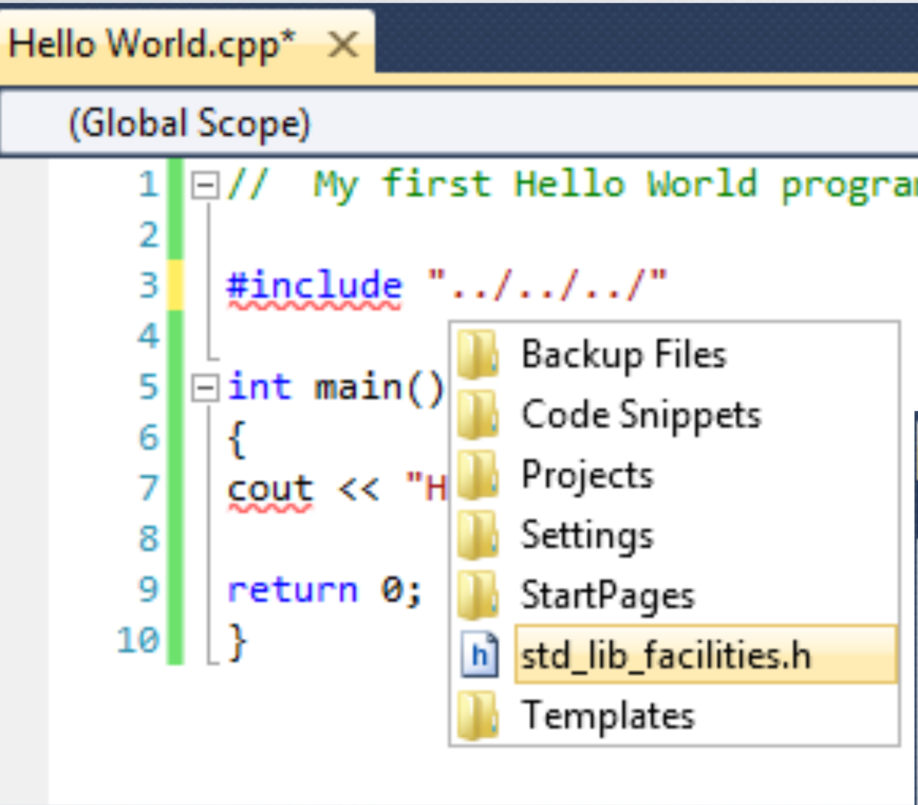
Include the header file (1/3)

- Depending on where you placed your **std_lib_facilities** .h file, you will need to add one two or three " ../". The file was put in the Visual Studio 2010 folder, so we will need to add three "../ ".
- Your program #include should look like this #include "../../../ std_lib_facilities.h"
- Make sure there are **no spaces** between " " . "../../../ std_lib_facilities.h"

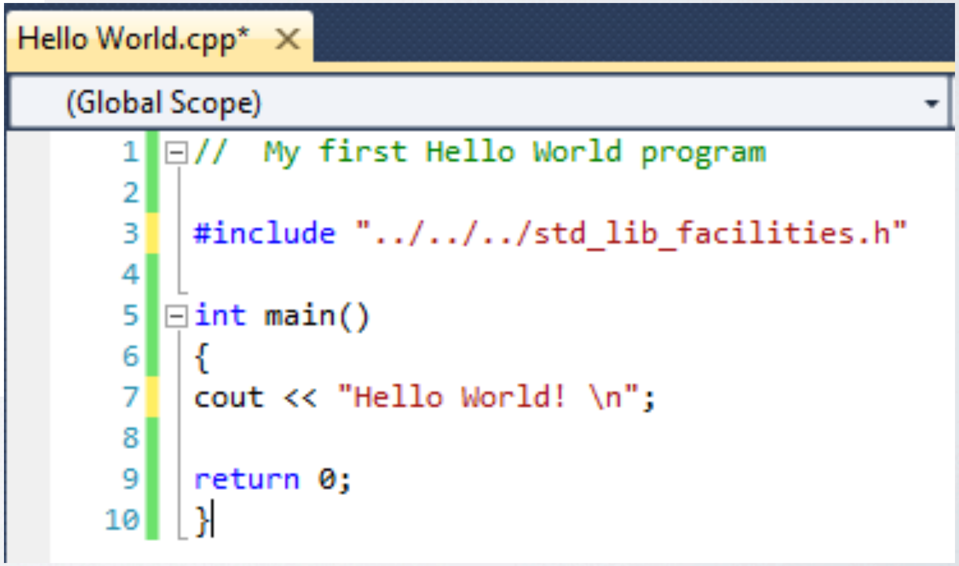
Include the header file (2/3)

- If you put a copy in your projects folder it will look like this
  - #include "../../std_lib_facilities.h"
- If you put a copy in your Hello world file folder (folders you create each time to start a new project) it will look like this
  - #include "../std_lib_facilities.h"
- If you put a copy in your Hello world file folder (folders where your .cpp file is stored), it will look like this
  - #include "std_lib_facilities.h"

Include the header file (3/3)



```
// My first Hello World program

#include "../../../"

int main()
{
cout << "H

return 0;
}
```



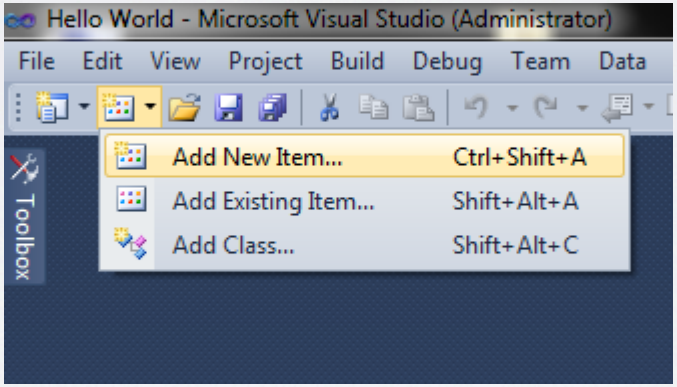
```
// My first Hello World program

#include "../../../std_lib_facilities.h"

int main()
{
cout << "Hello World! \n";

return 0;
}
```

How to get your #include “std_lib_facilities.h” working if you are using Visual Studio 2010 express (an alternative approach)

- Open Standard library access header from http://www.stroustrup.com/Programming/
- and Select All (Ctrl+A) and Copy to clipboard (Ctrl+C)

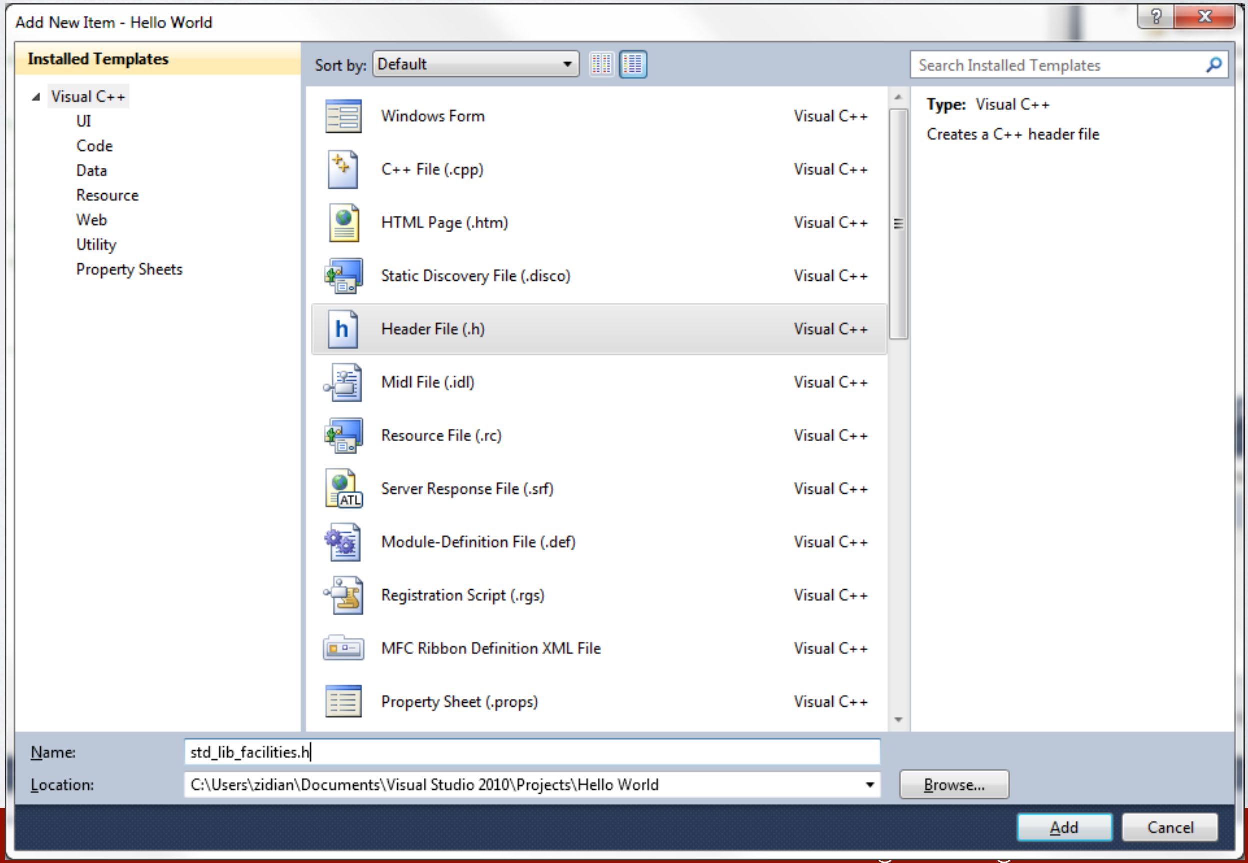
Add a c++ header file to the project (1/3)

- Click the Add New Item icon on the menu bar (usually the second icon from the left). That will open the Add New Item dialog box. Select Code under the Visual C++ category.


Hello World - Microsoft Visual Studio (Administrator)
File
Edit
View
Project
Build
Debug
Team
Data
Add New Item... Ctrl+Shift+A
Add Existing Item... Shift+Alt+A
Add Class... Shift+Alt+C
Toolbox

Add a c++ header file to the project (2/3)

- Select the Header File (.h) icon in the template window. Type the name of your header file (std_lib_facilities.h) in the Name text box and click Add.



Add New Item - Hello World
Installed Templates
Visual C++
UI
Code
Data
Resource
Web
Utility
Property Sheets
Search Installed Templates
Windows Form — Visual C++
C++ File (.cpp) — Visual C++
HTML Page (.htm) — Visual C++
Static Discovery File (.disco) — Visual C++
Header File (.h) — Visual C++
Midl File (.idl) — Visual C++
Resource File (.rc) — Visual C++
Server Response File (.srf) — Visual C++
Module-Definition File (.def) — Visual C++
Registration Script (.rgs) — Visual C++
MFC Ribbon Definition XML File — Visual C++
Property Sheet (.props) — Visual C++
Browse...
Add
Cancel

Sort by: Default

Type: Visual C++

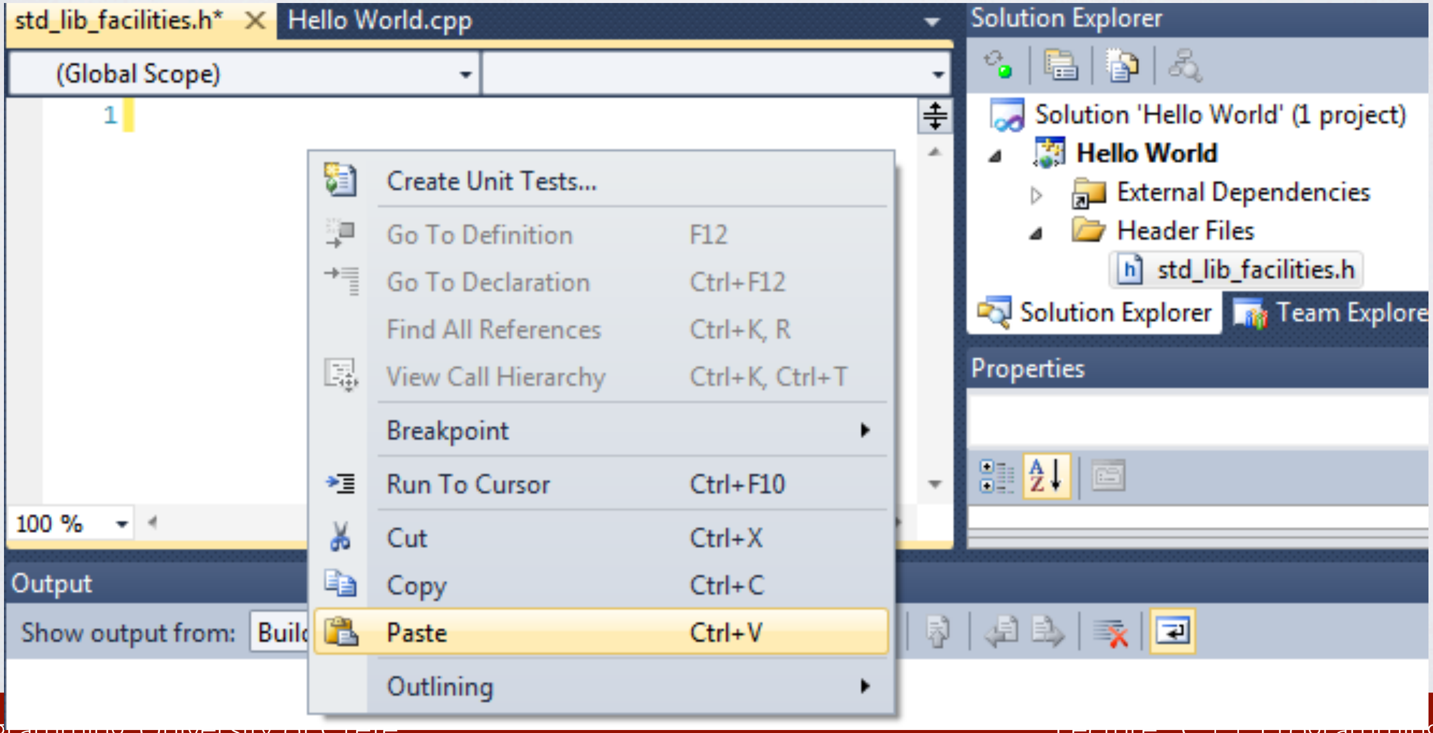
Creates a C++ header file

Name: std_lib_facilities.h

Location: C:\Users\zidian\Documents\Visual Studio 2010\Projects\Hello World

Add a c++ header file to the project (3/3)

- You have created an empty header file. Double click on it and inside the editor paste (Ctrl+V) the data from the clipboard.

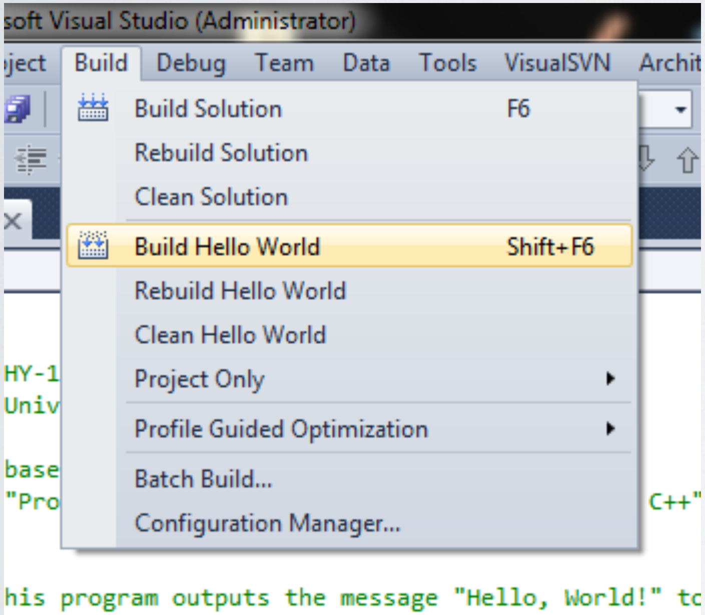


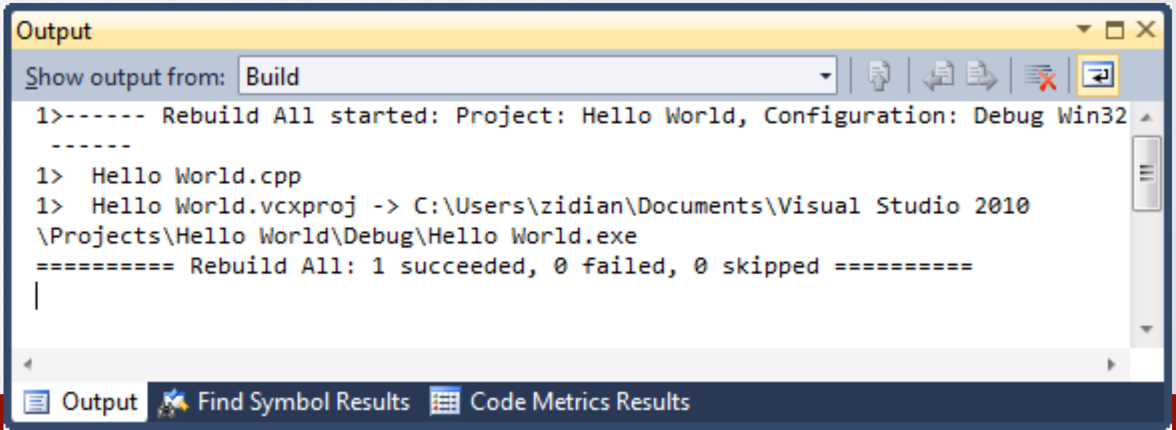
Creating and running a program

- The steps are:
  - 1. Create a new project.
  - 2. Add a C++ source file to the project.
  - 3. Enter your source code.
  - 4. Include “std_lib_facilities.h” to the project
  - 5. **Build an executable file.**
  - 6. Execute the program.

Build an executable file

- Click Build Hello World (Shift+F6, may be different), and check build results!


soft Visual Studio (Administrator)
ject Build Debug Team Data Tools VisualSVN Archit
Build Solution F6
Rebuild Solution
Clean Solution
Build Hello World Shift+F6
Rebuild Hello World
Clean Hello World
Project Only
Profile Guided Optimization
Batch Build...
Configuration Manager...
HY-1
Univ
base
"Pro
C++"
his program outputs the message "Hello, World!" to



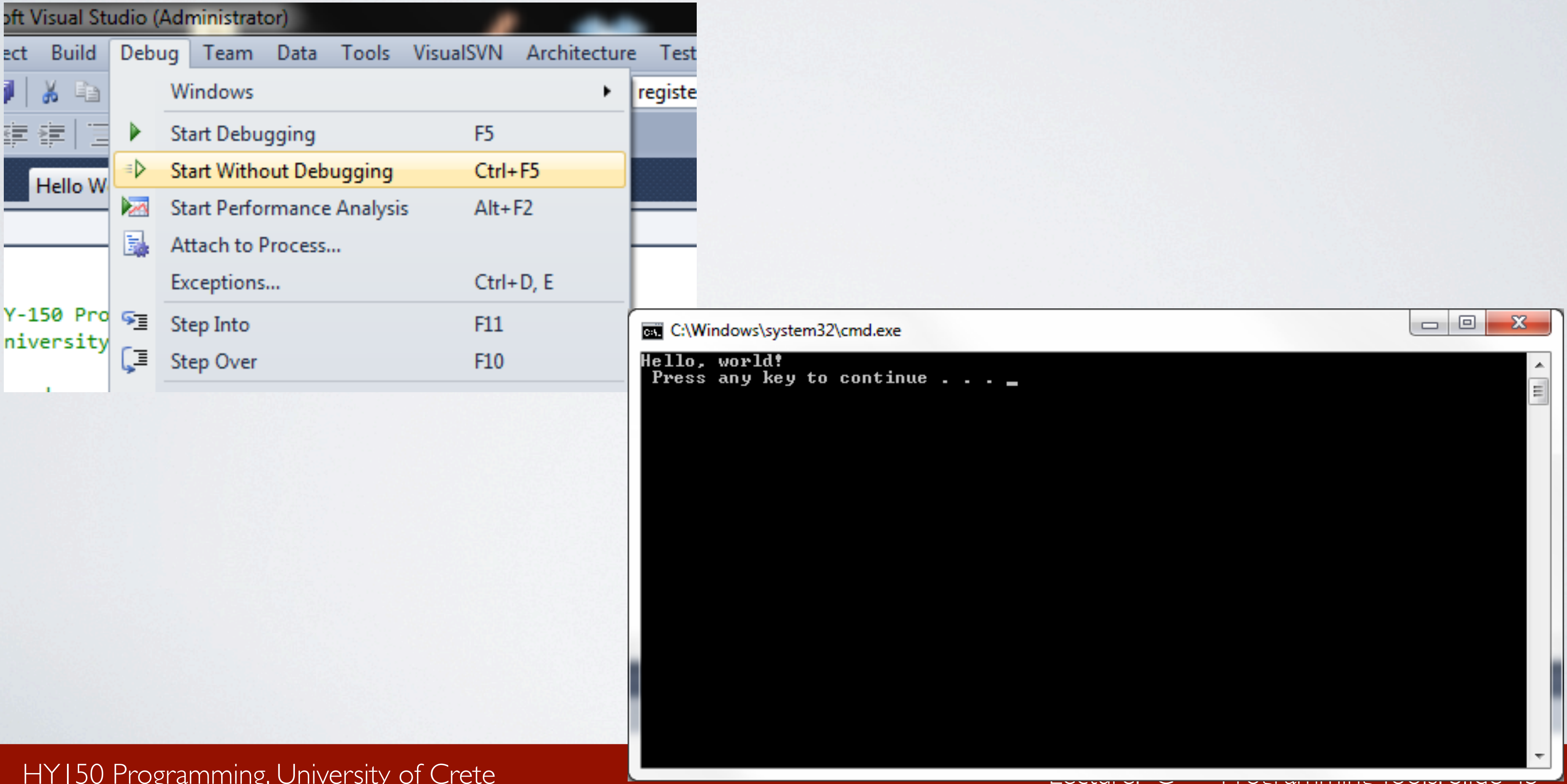
```
1>------ Rebuild All started: Project: Hello World, Configuration: Debug Win32
 ------
1>  Hello World.cpp
1>  Hello World.vcxproj -> C:\Users\zidian\Documents\Visual Studio 2010
\Projects\Hello World\Debug\Hello World.exe
========== Rebuild All: 1 succeeded, 0 failed, 0 skipped ==========
```

Creating and running a program

- The steps are:
  - 1. Create a new project.
  - 2. Add a C++ source file to the project.
  - 3. Enter your source code.
  - 4. Include “std_lib_facilities.h” to the project
  - 5. Build an executable file.
  - 6. **Execute the program.**

Execute the program

- Click the Start Without Debugging button



```
Hello, world!
 Press any key to continue . . . _
```

Experiment and prove the aforementioned to yourself

GOOD LUCK!