



读完本文档后需要达到的基本要求：

1、掌握在VS下一个解决方案建立多项目的方法

- 能按要求指定解决方案的起始目录，而不是仅使用系统默认的目录
- 能按要求指定每个项目的名称，而不是仅使用系统默认的名称

2、掌握在VS下每个项目含多个源程序及头文件的方法

- 能按要求指定每个源程序/头文件的名称，而不是仅使用系统默认的名称
- 能做到源程序/头文件的正确包含

3、掌握在VS下切换“启动项目”的方法

提示：

1、认真阅读，按步操作

2、带着思考去阅读，任何文档的示例名称都是固定的（起始目录、解决方案名、项目名、源程序文件名、头文件名），要掌握按自己的要求进行变化的能力

在VS2022中建立含多源程序文件的项目(无头文件)



- ★ 以main()调用max()为例，要求将main()和max()放在两个不同的源程序文件中

解决方案: example(新建)

项目名称: ex_without_h

源程序名: ex1.cpp (main函数+max函数的声明)

ex2.cpp (max函数的实现)



1. “文件”-“新建”-“项目”，出现的对话框选择“空项目”，按“下一步”
(注：本文档是在新的解决方案下建立新项目)

The screenshot illustrates the process of creating a new project in Visual Studio. On the left, the 'File' menu is open, with 'New' selected. A red arrow points to 'New'. Within the 'New' submenu, 'Project...' is highlighted, also indicated by a red arrow. On the right, the 'Create New Project' dialog is displayed. It shows a list of project templates under 'Recent Items'. The 'Empty Project' template is selected, indicated by a red arrow. At the bottom right of the dialog, a blue-bordered button labeled 'Next Step (N)' is highlighted with a red arrow.

创建新项目

最近使用的项目模板(R)

所有语言(L) 所有平台(P) 所有项目类型(D)

空项目 C++ 空项目
使用适用于 Windows 的 C++ 从头开始操作。不提供基础文件。
C++ Windows 控制台

控制台应用 C++ 在 Windows 终端运行代码。默认打印 "Hello World"。
C++ Windows 控制台

CMake 项目 C++ Windows Linux 控制台
生成不依赖于 .sln 或 .vcxproj 文件的新式跨平台 C++ 应用。

Windows 桌面向导 C++ Windows 桌面 控制台 库
使用向导自行创建 Windows 应用。

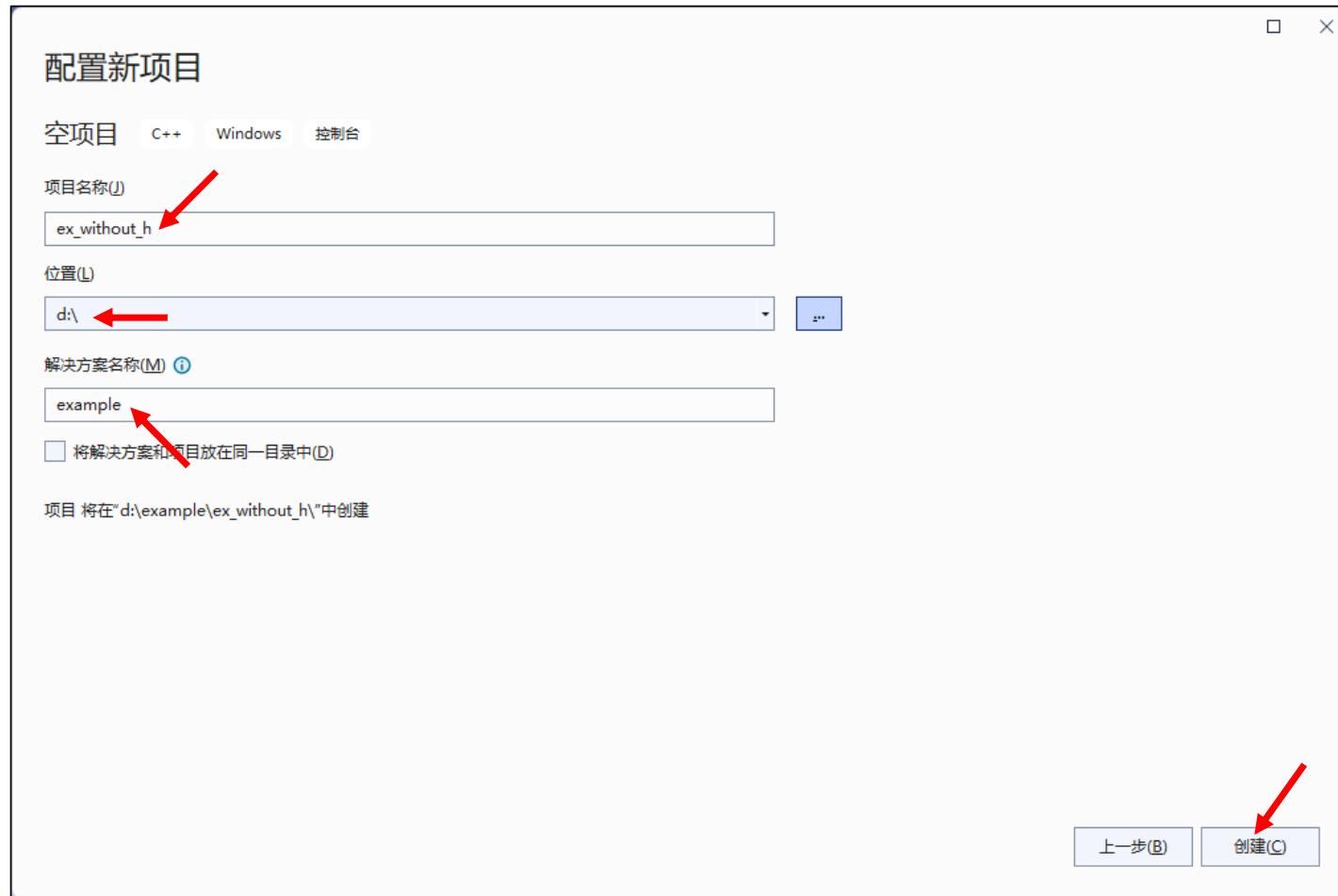
Windows 桌面应用程序 C++ Windows 桌面
具有在 Windows 上运行的图形用户界面的应用程序的项目。

空白解决方案

下一步(N)



2. 在出现的对话框中，依次输入解决方案名称(example)/
位置(自行决定，图示为d:\)/项目名称(ex_without_h)，按“创建”





3. 建立第一个源文件名ex1.cpp，并在ex1.cpp中输入main函数及max函数的声明(注：记得切换到x86)

The screenshot shows the Microsoft Visual Studio IDE interface. The top menu bar includes File, 编辑(E), 视图(V), Git(G), 项目(P), 生成(B), 调试(D), 测试(S), 分析(N), 工具(T), 扩展(X), 窗口(W), 帮助(H), and 搜索 (Ctrl+Q). The title bar says "example". The toolbar has various icons for file operations. The dropdown menu "Debug" is circled in red, and the sub-menu "x86" is selected.

The Solution Explorer on the left shows a project named "example" with one item: "ex1.cpp". A red arrow points to "ex1.cpp". The Project Explorer shows "ex_without_h" and its sub-items: 头引用, 外部依赖项, 头文件, 源文件, and ex1.cpp.

The main code editor window displays the following C++ code:

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

int max(int x, int y);

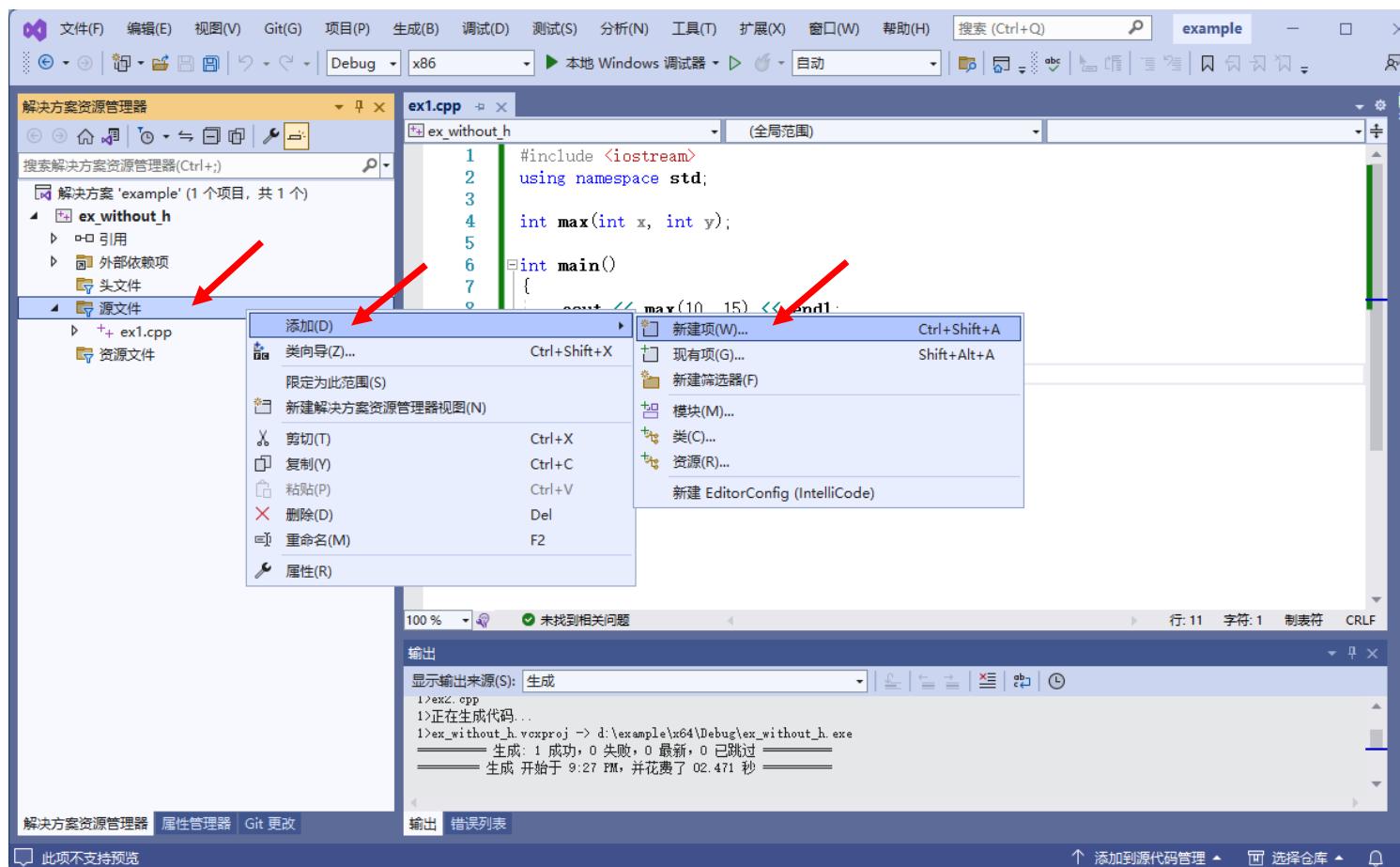
int main()
{
    cout << max(10, 15) << endl;
    return 0;
}
```

The Output window at the bottom shows the build logs:

```
显示输出来源(S): 生成
1>ex1.cpp
1>正在生成代码...
1>ex_without_h.vcxproj -> d:\example\x64\Debug\ex_without_h.exe
===== 生成: 1 成功, 0 失败, 0 最新, 0 已跳过 =====
===== 生成: 开始于 9:27 PM, 并花费了 02.471 秒 =====
```



4. “源文件” – 右键菜单 – “添加” – “新建项”



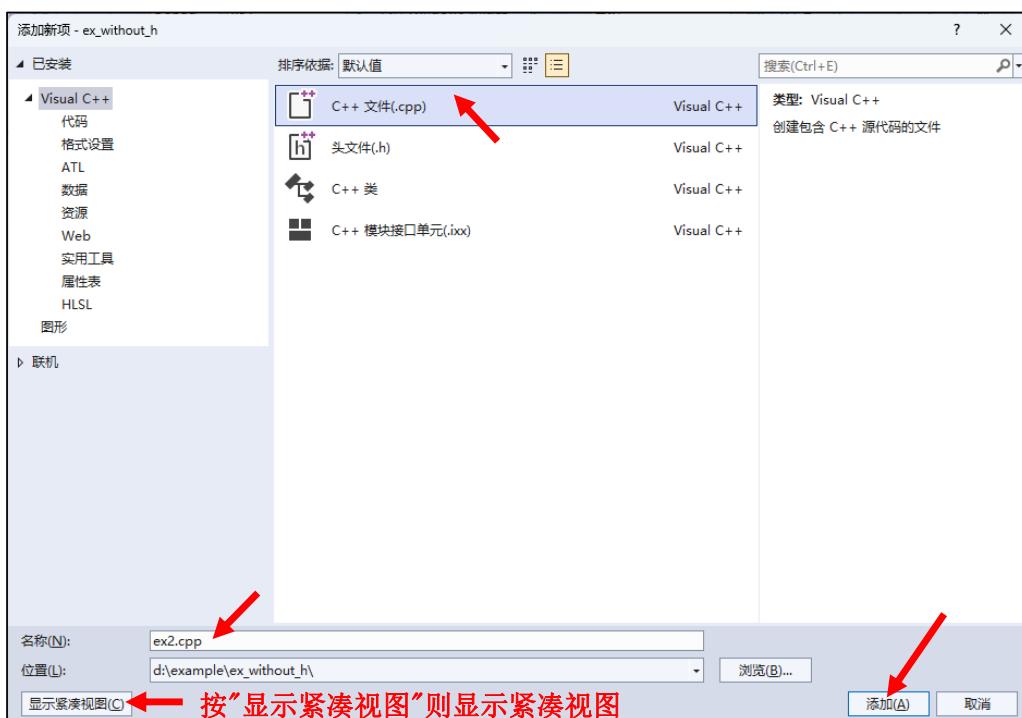


4. 出现的对话框有两种：紧凑视图和所有模板视图

★ 紧凑视图中直接输入ex2.cpp，按“添加”



★ 所有模板视图中选择“C++文件(.cpp)”，输入名称ex2.cpp，按“添加”





5. 看到ex2.cpp已在项目中，在其中输入max函数的实现

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

int max(int x, int y)
{
    return x > y ? x : y;
}
```

输出

```
显示输出来源(S): 生成
1>ex2.cpp
1>正在生成代码...
1>ex_without_h.vcxproj -> d:\example\x64\Debug\ex_without_h.exe
===== 生成: 1 成功, 0 失败, 0 最新, 0 已跳过 =====
===== 生成 开始于 9:27 PM, 并花费了 02.471 秒 =====
```

★ 单击文件标签栏/左侧cpp文件名，可切换当前显示的文件



6. “调试”-“开始执行(不调试)” / 按CTRL+F5，可以编译并运行成功

The screenshot shows the Microsoft Visual Studio interface during the compilation and execution of a C++ project named 'example'.

Solution Explorer: Shows the project structure with files `ex1.cpp` and `ex2.cpp` under the `源文件` (Source Files) node.

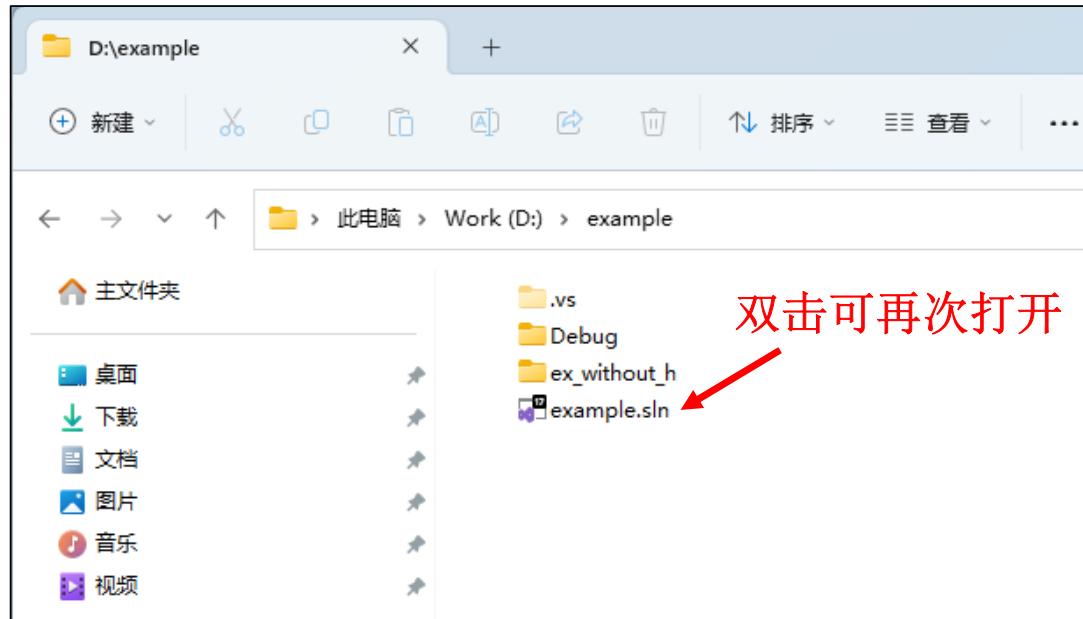
Code Editor: Displays the content of `ex2.cpp`:1 #include <iostream>
2 using namespace std;
3
4 int max(int x, int y)
5 {
6 return x > y ? x : y;
7 }

Output Window: Shows the build log:100 % 未找到相关问题
输出 显示输出来源(S): 生成
1> ex2.cpp
1> 正在生成代码...
1> ex_without_h.vcxproj -> d:\example\x64\Debug\ex_without_h.exe
===== 生成: 1 成功, 0 失败, 0 最新, 0 已跳过 =====
===== 生成 开始于 9:27 PM, 并花费了 02.471 秒 =====

Debug Console: Shows the command prompt output:15
d:\example\Debug\ex_without_h.exe (进程 11224)已退出, 代码为 0。
按任意键关闭此窗口. . .



7. 退出VS2022后，再次双击 *.sln 图标，即可再次打开该项目



★ 切记：不要在VS里面直接打开cpp文件名!!!

在VS2022中建立含多源程序文件的项目(有头文件)



- ★ 以main()调用max()为例，要求将main()和max()放在两个不同的源程序文件中

解决方案: example(已有)

项目名称: ex_with_h

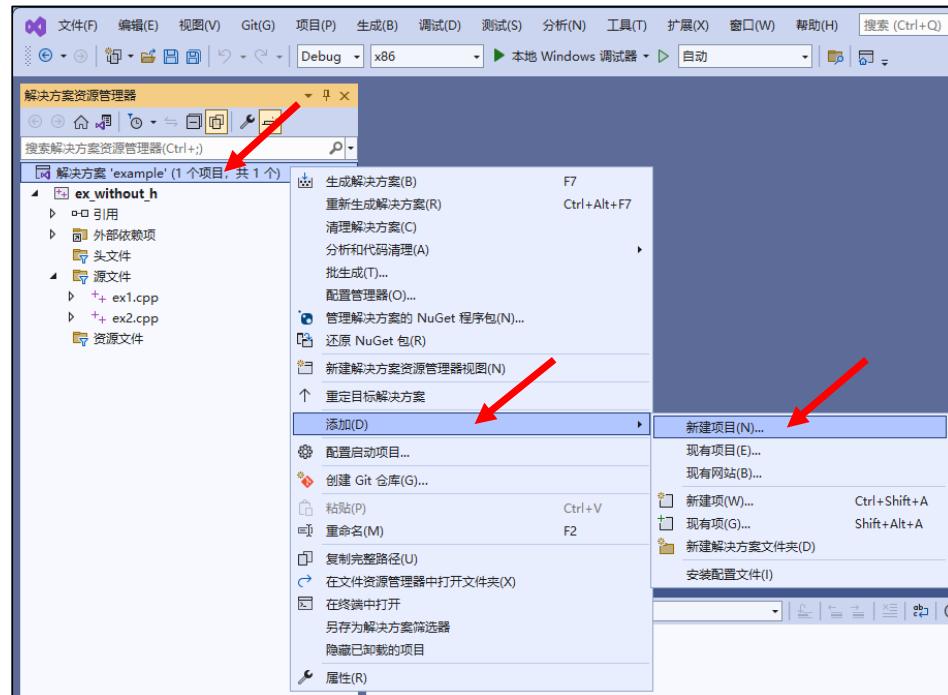
源程序名: ex3.cpp (main函数)

 ex4.cpp (max函数的实现)

头文件名: ex.h (max函数的声明)

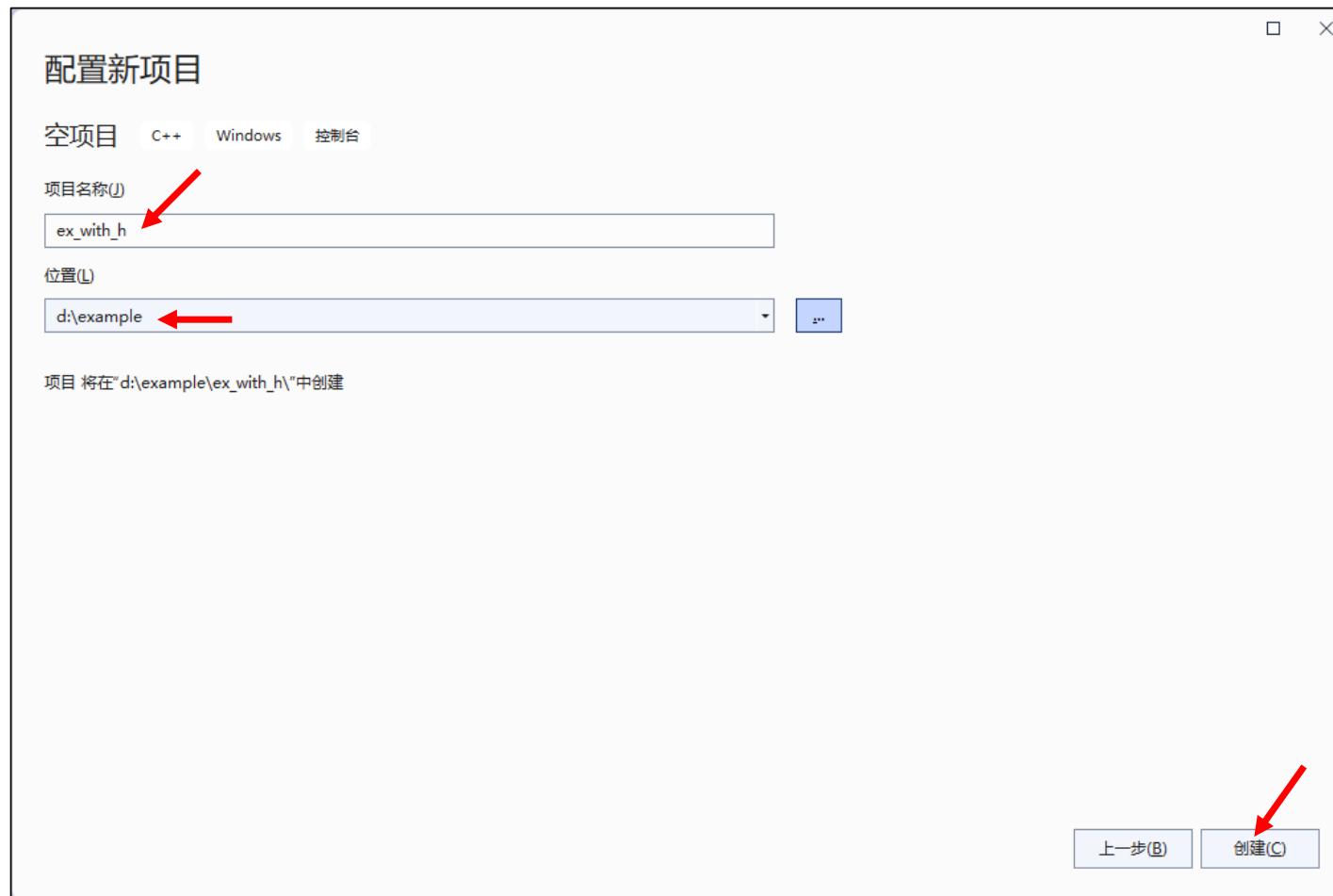


1. “example解决方案” - “右键菜单” - “添加” - “新建项目”，
出现的对话框选择“空项目”，按“下一步”
(注：本文档是在已有解决方案下建立新项目)





2. 在出现的对话框中，输入项目名称(ex_with_h)/位置保持不变，按“创建”





3. 将建立的新项目ex_with_h设为启动项目

The screenshot shows the Microsoft Visual Studio interface with the following details:

- Solution Explorer:** Displays the solution named 'example' containing two projects:
 - 'ex_with_h': Contains files ex1.cpp and ex2.cpp.
 - 'ex_without_h': Contains files ex1.cpp and ex2.cpp.
- Status Bar:** Shows the message "正在创建项目“ex_with_h”... 项目创建成功。" (Creating project "ex_with_h"... Project creation successful.)
- Output Window:** Empty window labeled "输出" (Output).
- Toolbar:** Standard Visual Studio toolbar with icons for file operations, search, and help.
- Menu Bar:** Includes File, Edit, View, Git, Project, Build, Debug, Test, Analyze, Tools, Extensions, Windows, Help, and Search.
- Search Bar:** "搜索 (Ctrl+Q)" (Search (Ctrl+Q))
- Build Configuration:** Set to "Debug" and "x86".
- Local Debugger:** Set to "本地 Windows 调试器" (Local Windows Debugger) and "自动" (Automatic).



4. 建立第一个源文件名ex3. cpp， 并在ex3. cpp中输入main函数并包含头文件ex. h

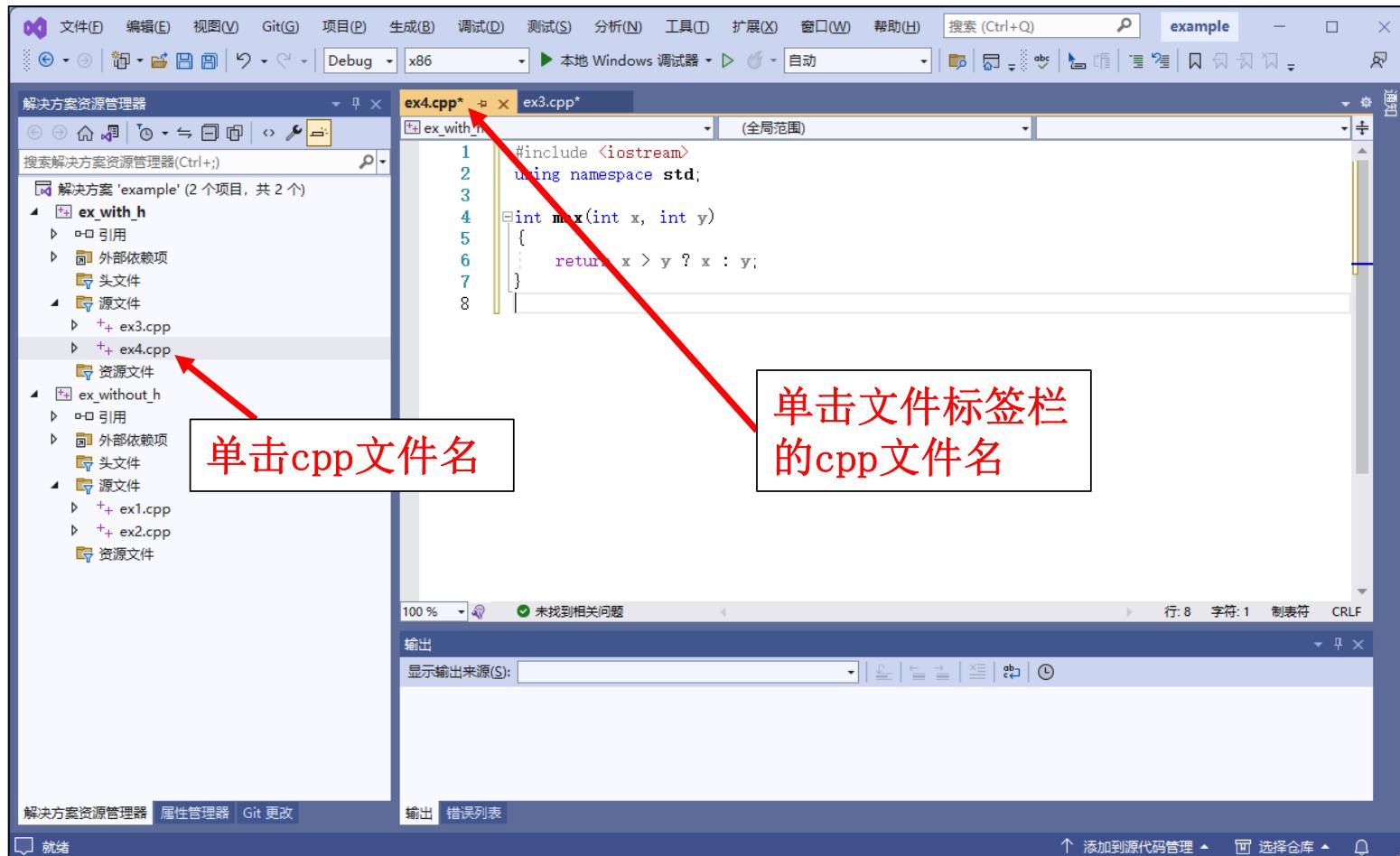
The screenshot shows the Microsoft Visual Studio IDE interface. The title bar indicates the project is named 'example'. The '解决方案资源管理器' (Solution Explorer) on the left lists two projects: 'ex_with_h' and 'ex_without_h'. Under 'ex_with_h', there are '引用' (References), '外部依赖项' (External Dependencies), '头文件' (Headers), '源文件' (Source Files), and '资源文件' (Resource Files). A red arrow points to the '源文件' section, where 'ex3.cpp' is listed. The '输出' (Output) window at the bottom is empty.

```
#include <iostream>
#include "ex.h"
using namespace std;

int main()
{
    cout << max(10, 15) << endl;
    return 0;
}
```



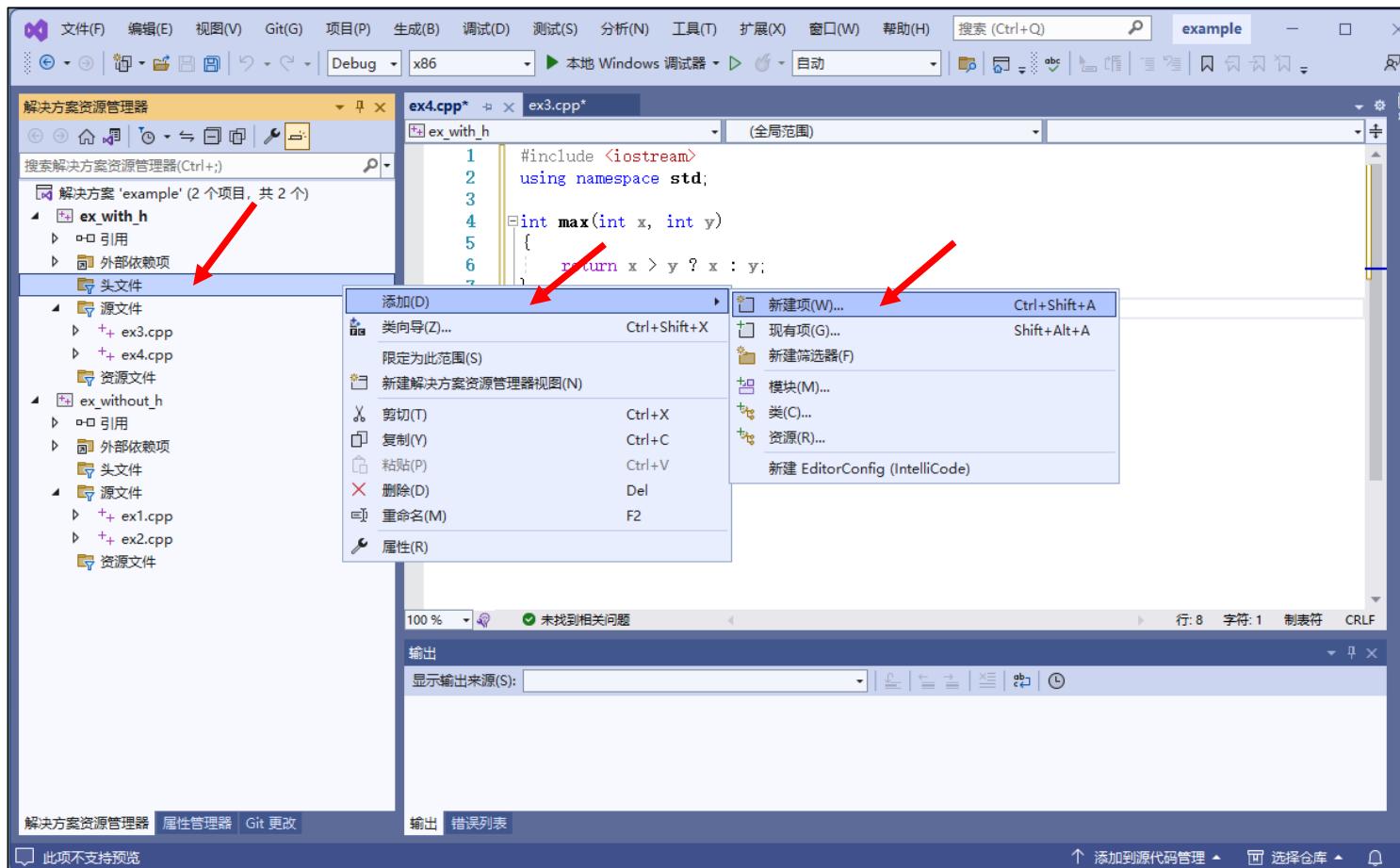
5. 建立第二个源文件名ex4.cpp，并输入max函数的实现



★ 单击文件标签栏/左侧cpp文件名，可切换当前显示的文件

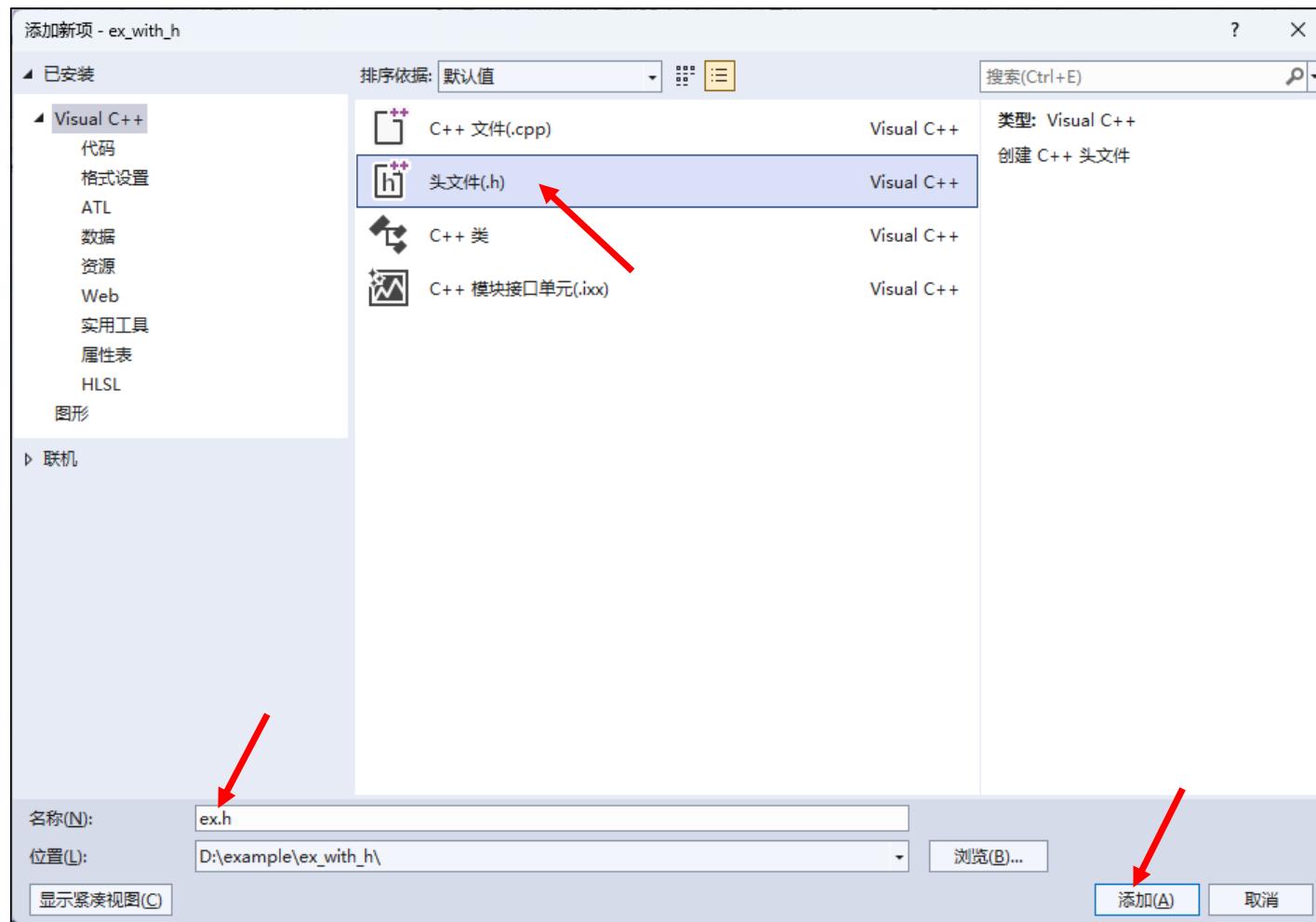


6. “头文件” – 右键菜单 – “添加” – “新建项”





7. 出现的对话框中选择“头文件(.h)”，名称ex.h，按“添加” (注：选择全部模板视图)





8. 看到ex.h已在项目中，在其中输入max函数的声明

该行内容是系统自动添加，不要动
.h文件的首行信息加在此行的前面

单击.h文件名

单击文件标签栏的.h文件名

```
#pragma once
int max(int x, int y);
```

★ 单击文件标签栏/左侧.h文件名，可切换当前显示的文件



9. “调试”-“开始执行(不调试)” / 按CTRL+F5，可以编译并运行成功

The screenshot shows the Microsoft Visual Studio interface during a build process. The top menu bar includes File, Edit, View, Git, Project, Build, Debug, Test, Analyze, Tools, Extensions, Windows, Help, and a search bar. The toolbar below has icons for file operations like Open, Save, and Print, along with build-related icons.

The Solution Explorer on the left lists two projects: 'example' (2 items) and 'ex_with_h'. The 'ex_with_h' project contains files ex.h, ex4.cpp, and ex3.cpp. The 'ex_without_h' project contains files ex1.cpp and ex2.cpp.

The main code editor window displays ex4.cpp with the following content:

```
1 #pragma once
2
3 int max(int x, int y);
4
```

To the right of the code editor is the 'Microsoft Visual Studio 调试控制台' (Debug Console) window, which shows the output of the application's execution:

```
15
d:\example\Debug\ex_with_h.exe (进程 19168)已退出, 代码为 0。
按任意键关闭此窗口. . .■
```

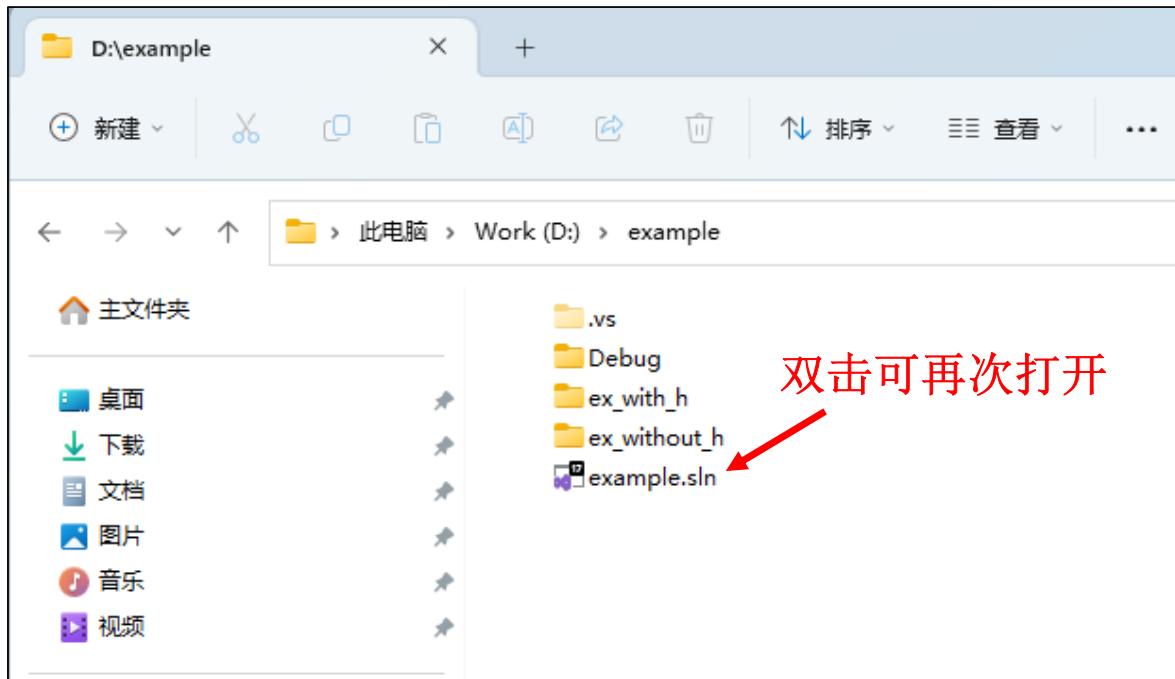
At the bottom, the Output window shows the build logs:

```
显示输出来源(S): 生成
1>ex4.cpp
1>正在生成代码...
1>ex_with_h.vcxproj -> d:\example\Debug\ex_with_h.exe
===== 生成: 1 成功, 0 失败, 0 最新, 0 已跳过 =====
===== 生成 开始于 10:00 PM, 并花费了 02.894 秒 =====
```

The status bar at the bottom indicates "生成成功" (Build Success).



10. 退出VS2022后，再次双击 *.sln 图标，即可再次打开该项目



★ 切记：不要在VS里面直接打开cpp文件名!!!