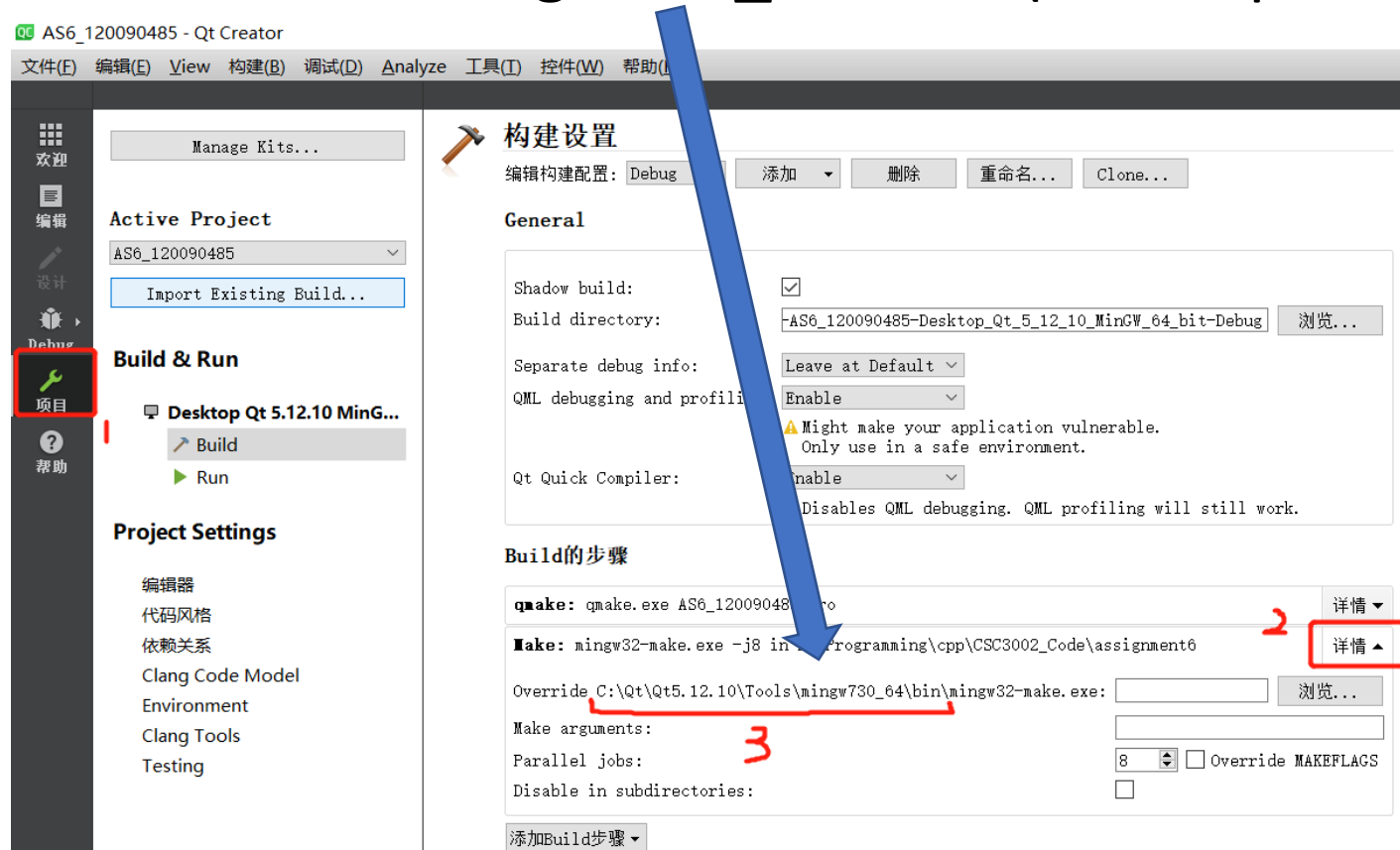


Build the link

For Windows User:

- 1. (Based on the fact that your Qt can run code now) open a qt project, remember this path at 3. In my computer, it is “C:\Qt\Qt5.12.10\Tools\mingw730_64\bin” (not all path, end to \bin)



For Windows User:

- 2. Set the environment variable. Double click the “Path” at 2, Add “C:\Qt\Qt5.12.10\Tools\mingw730_64\bin” (your path in the last step) to the “Path”.

The image shows a sequence of three overlapping windows from a Windows operating system, illustrating the steps to edit system environment variables. A blue arrow points from the text in the first block to the 'Path' variable in the third window.

Window 1: Windows Settings
The 'Settings' application is open to 'System' > 'Advanced system settings'. The 'Edit system environment variables' link is highlighted in the left sidebar. A red box highlights the 'Environment variables(N)...' button at the bottom right.

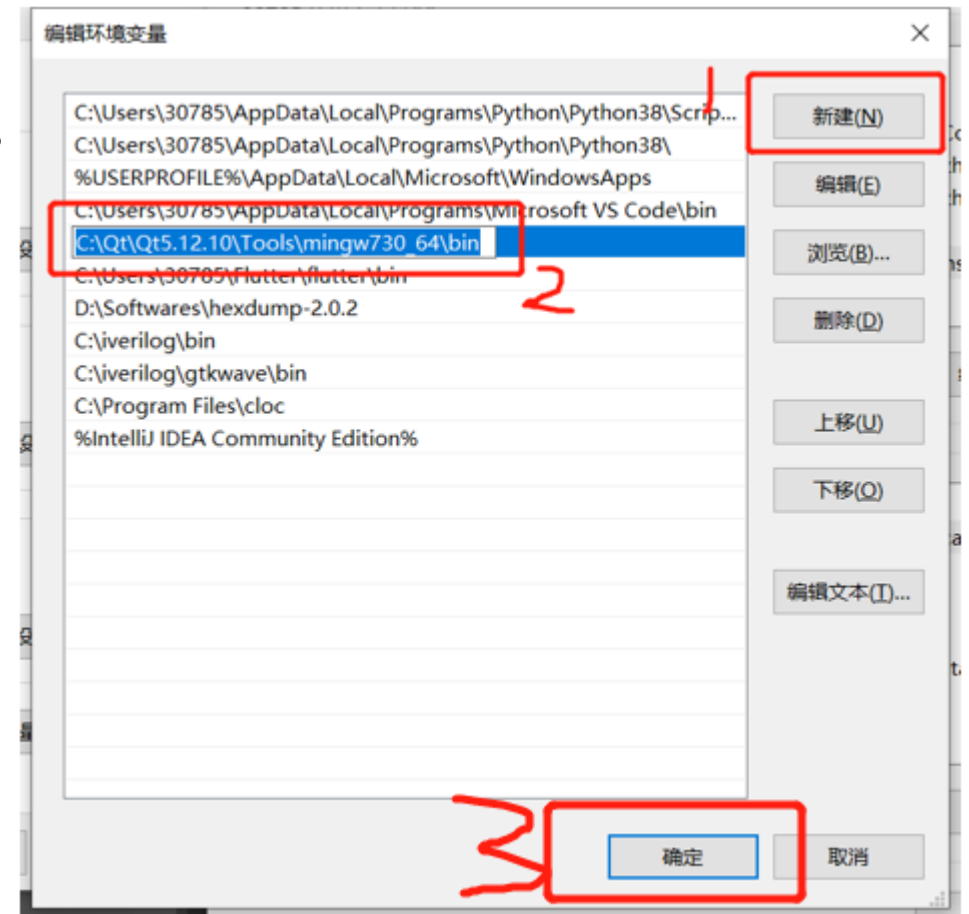
Window 2: System Properties
The 'System Properties' dialog box is open to the 'Advanced' tab. The 'Environment variables(N)...' button is highlighted with a red box.

Window 3: Environment Variables
The 'Environment Variables' dialog box is open, showing the '30785 的用户变量(U)' (User variables) section. The 'Path' variable is selected and highlighted with a red box. A red number '2' is placed next to it. The 'Edit(E)...' button is visible below the list.

| 变量 | 值 |
|-------------------------------|--|
| GUROBI_HOME | C:\gurobi952\win64 |
| IntelliJ IDEA Community Ed... | C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2022... |
| OneDrive | C:\Users\30785\OneDrive - CUHK-Shenzhen |
| OneDriveCommercial | C:\Users\30785\OneDrive - CUHK-Shenzhen |
| OneDriveConsumer | C:\Users\30785\OneDrive |
| Path | C:\Users\30785\AppData\Local\Programs\Python\Python38\Scr... |
| TEMP | C:\Users\30785\AppData\Local\Temp |
| TMP | C:\Users\30785\AppData\Local\Temp |

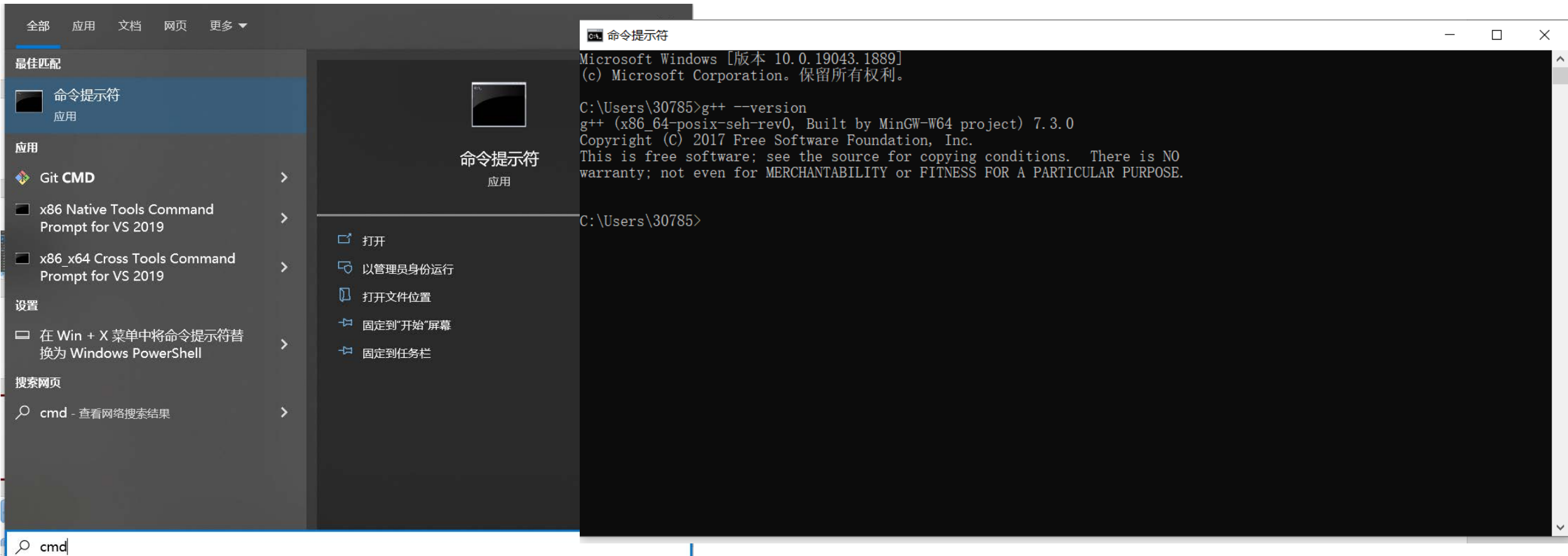
For Windows User:

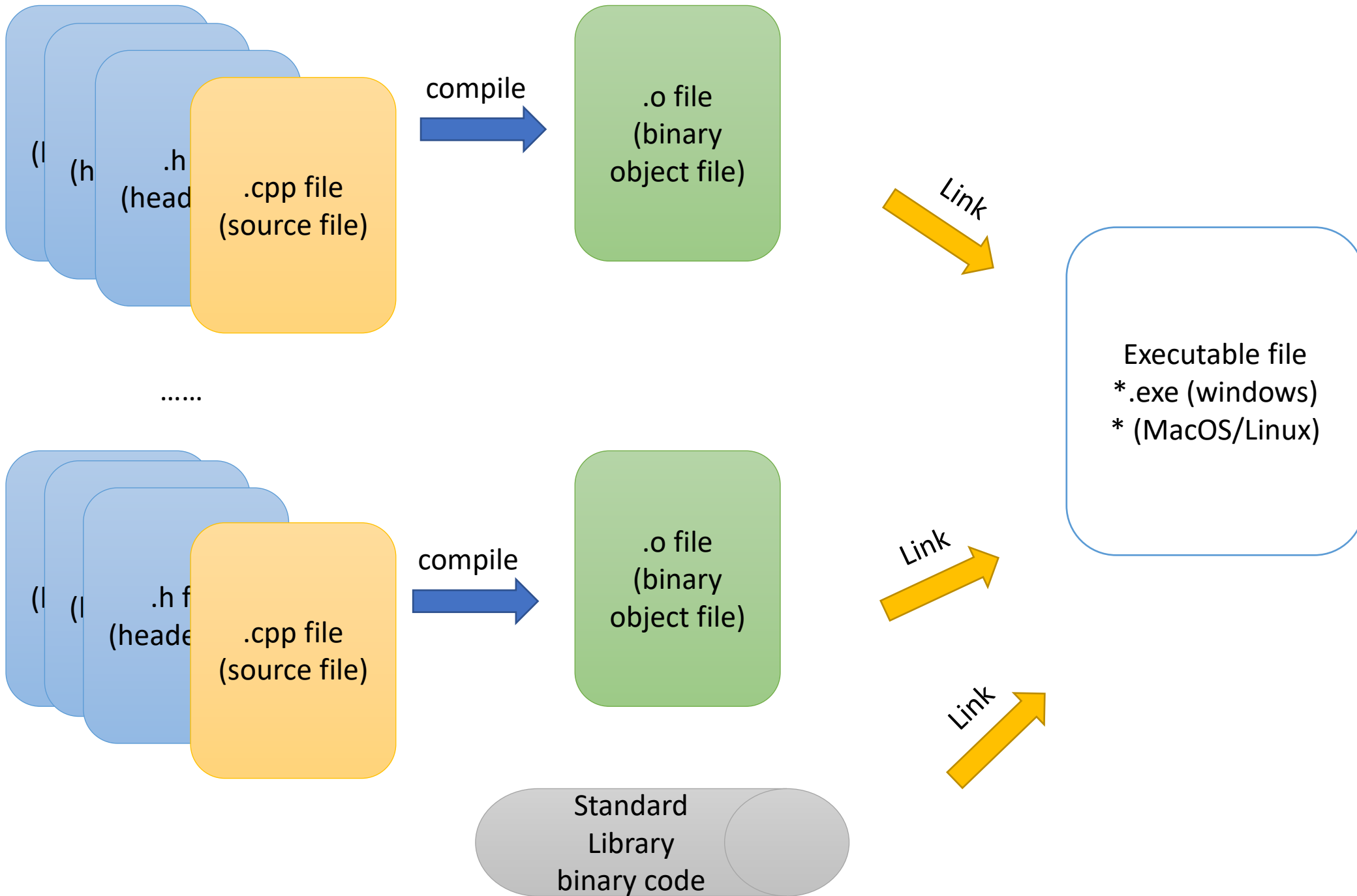
- 2. Set the environment variable. Double click the “Path” at 2, Add “C:\Qt\Qt5.12.10\Tools\mingw730_64\bin” (your path in the last step) to the “Path”.
- Click “confirm” at bottom right and exit.



For Windows User:

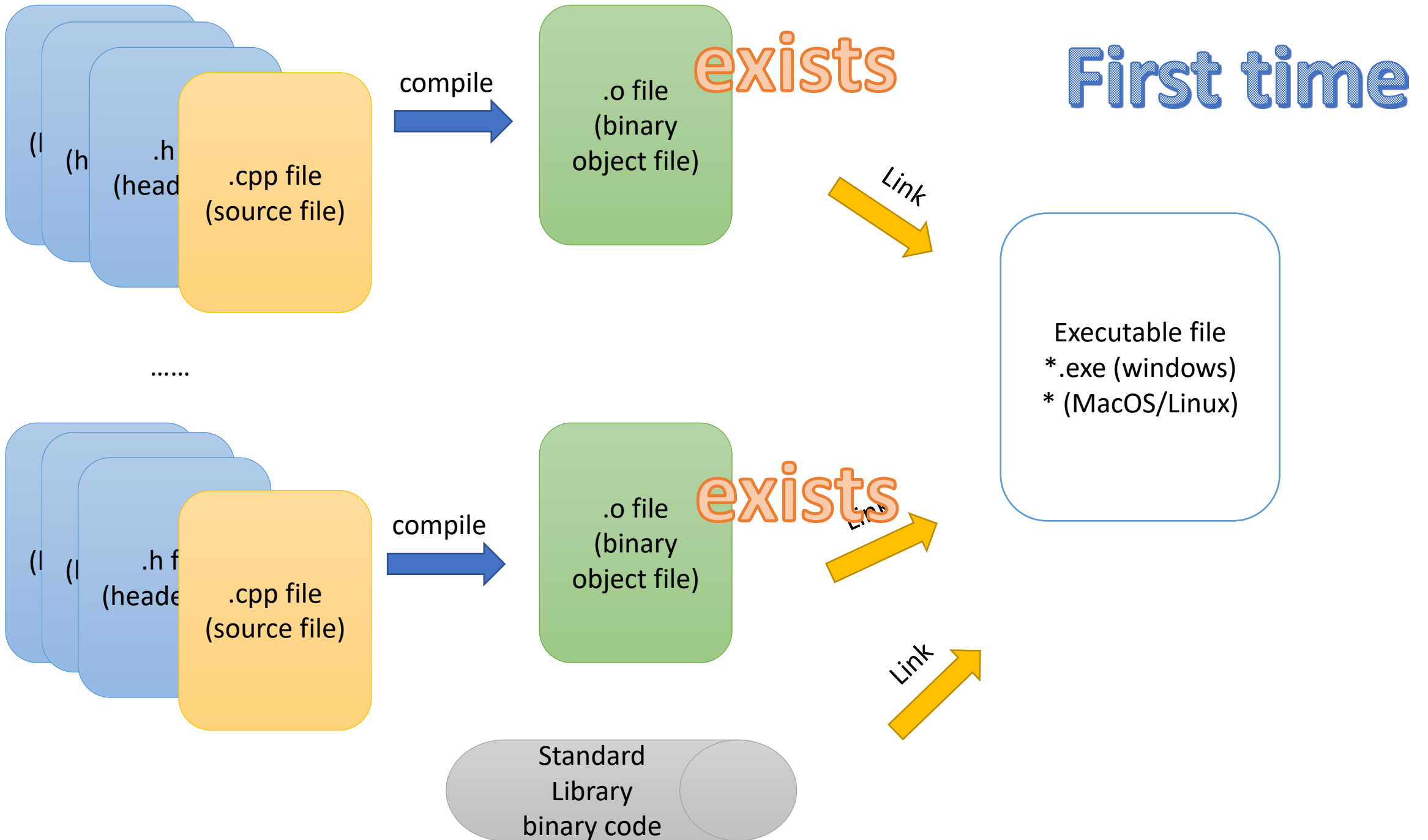
- 3. Test in the command line if the config is all right:
- Type in “g++ --version”

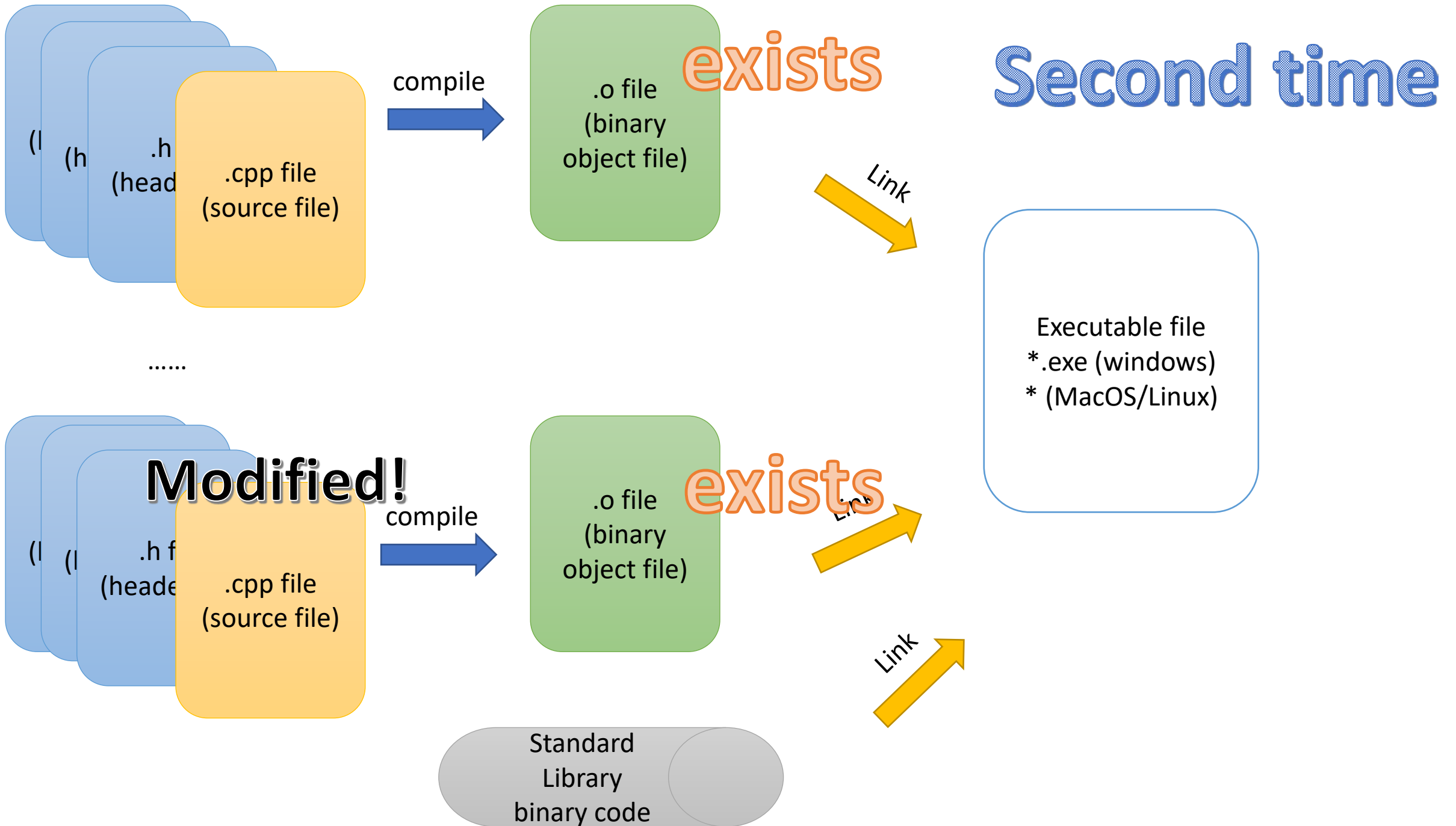


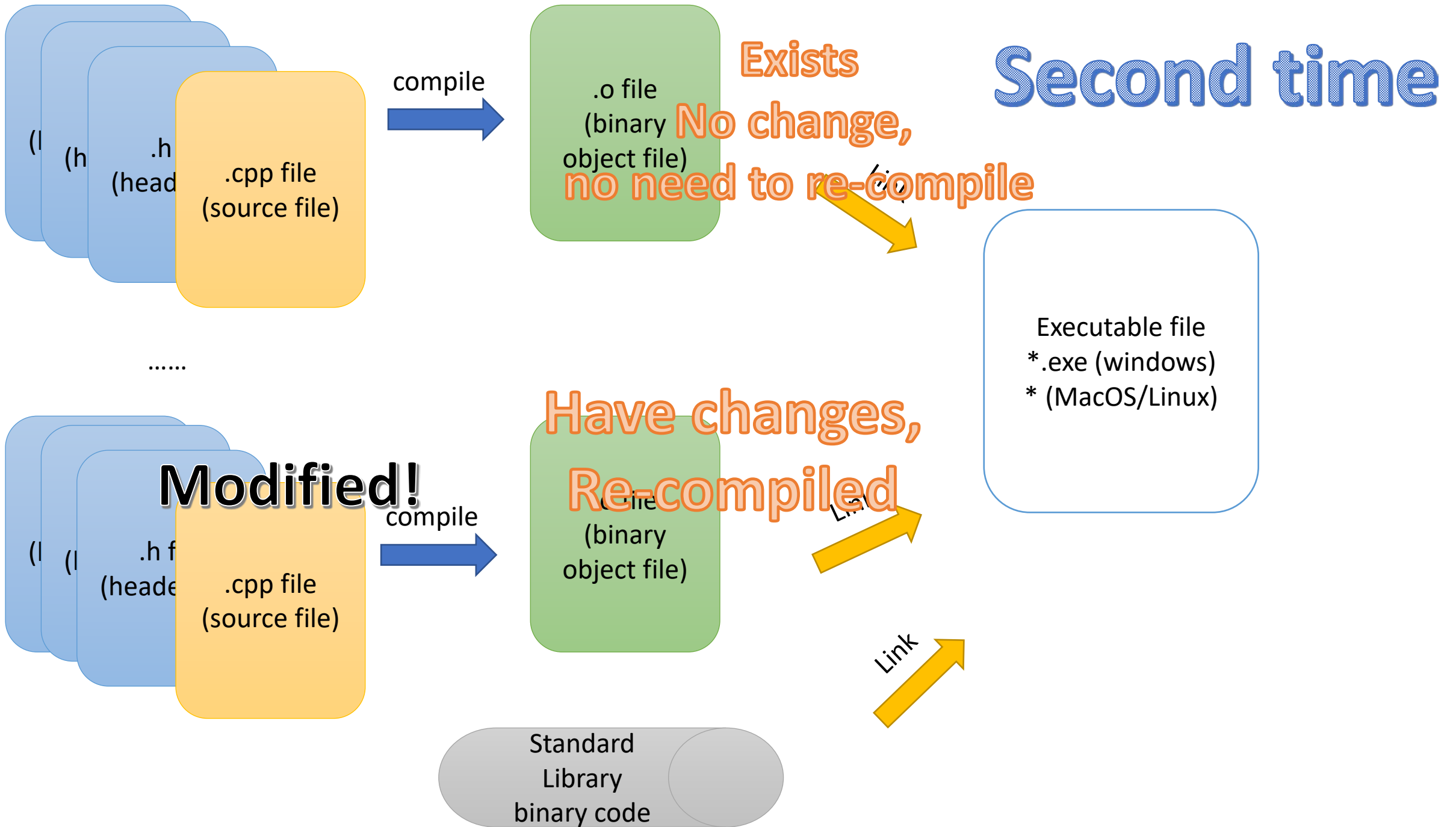


So why makefile?

- To tell a rule that how to compile C++ files and link to a C++ project.
- Cross platform (Windows, MacOS... All OK)
- **An important feature**: compile only changed files -> Save a lot of time!
-





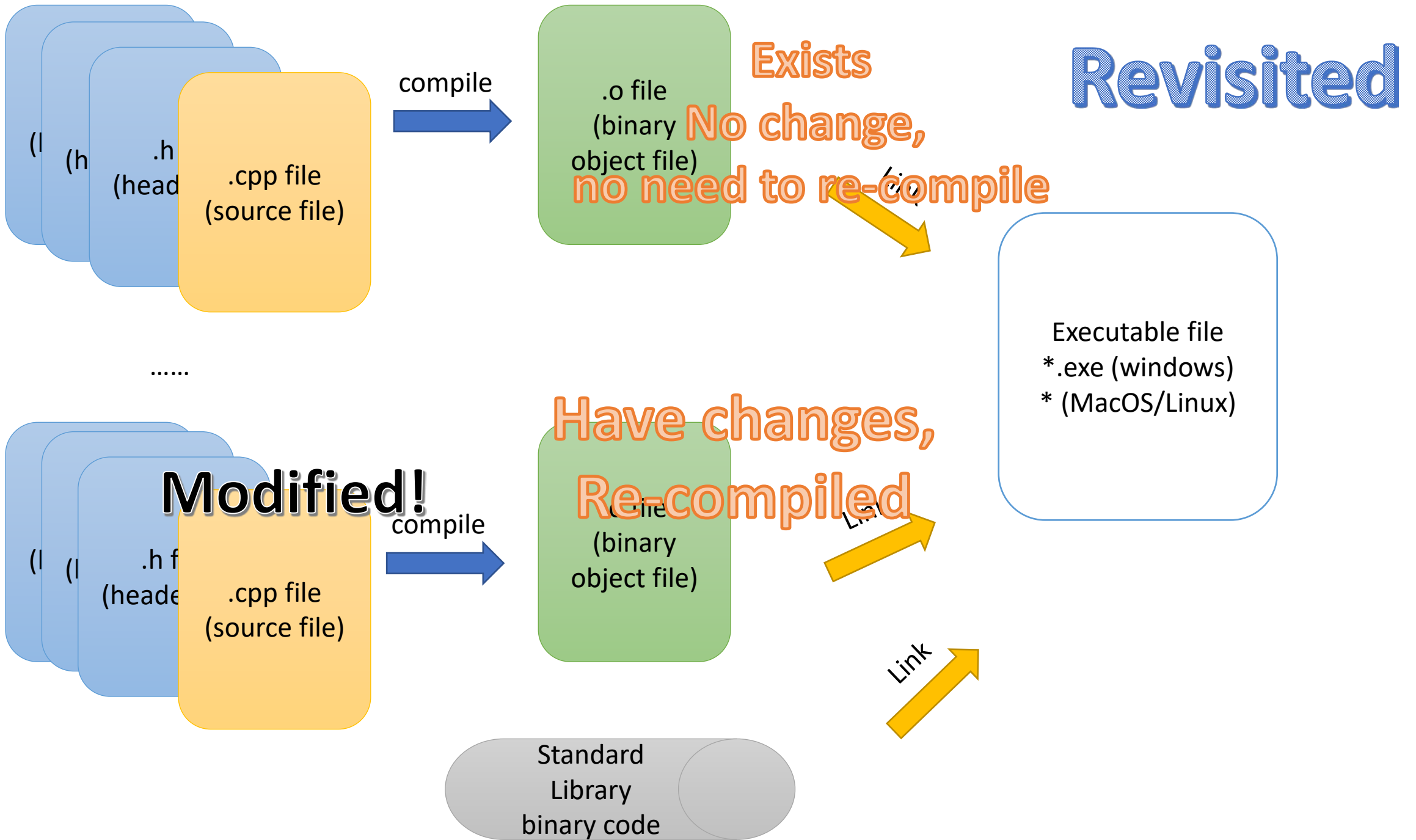


So why makefile?

- To tell a rule that how to compile C++ files and link to a C++ project.
- Cross platform (Windows, MacOS... All OK)
- **Advanced feature: compile only changed files -> Save a lot of time!**
 - If your project contains 1,000 cpp files. Every time, you are only working (modifying) on **one or two files**...
 - Therefore, only these **one or two files** needs to be re-compiled.

Extra: why separate into **different** header and source files?

- <https://stackoverflow.com/questions/1305947/why-does-c-need-a-separate-header-file>
- **One of the reason:** compile only changed files -> Save a lot of time!



If you don't know how to write makefile...

- Just use the Prof. Kinley's template! The template is **universal**.
- Remember to change the filenames to yours here

```
45 PROGRAM = \  
46     helloworld  
47  
48 OBJECTS = \  
49     helloworld.o \  
50     foo.o  
51
```

- You have x .cpp source files then you also have x .h header files with the same name. E.g.,
 - **hello.cpp**, util.cpp, math.cpp
 - **hello.h**, util.h, math.h