

Mini Project 3

Gomoku AI

Environment Settings Tutorial

Introduction

- This tutorial shows how to install required software to compile and execute this project
- There are also some troubleshooting steps
- Tutorial for Windows and Mac are both included

For All Students

- Please **do not use IDEs** to compile and run this project
 - Code Blocks, Dev-C++, Visual Studio, Xcode
- Please **use command line interfaces to compile and run** this project
 - Cmd, PowerShell, Terminal
- You can **use anything to write** your code
 - We recommend [VSCode](#)

For Windows Users

For Windows Users

1. Install mingw-w64
2. Compile your code
3. Execute your code
4. Troubleshooting

Go to MinGW-W64 Download Page

The screenshot shows the SourceForge project page for MinGW-w64. The browser address bar displays 'sourceforge.net/projects/mingw-w64/files/'. The page header includes the SourceForge logo and navigation links like 'Open Source Software', 'Business Software', and 'Resources'. A search bar is located on the right. The main heading is 'MinGW-w64 - for 32 and 64 bit Windows', with a subtitle 'A complete runtime environment for gcc' and credits 'Brought to you by: jon_y, ktietz70, nightstrike'. Below the heading is a tabbed interface with 'Files' selected. A green button 'Download Latest Version' (mingw-w64-v6.0.2.zip (15.3 MB)) and a blue 'Get Updates' button are visible. A table lists the project files with columns for Name, Modified, Size, and Downloads / Week. The 'mingw-w64' file has the highest download count at 33,705. A 'Recommended Projects' sidebar on the right lists related projects like 'MinGW - Minimalist GNU for Windows', 'MinGW-builds', 'TDM-GCC MinGW Compiler', and 'Windows Terminal'.

MinGW-w64 - for 32 and 64 bit Windows

A complete runtime environment for gcc
Brought to you by: jon_y, ktietz70, nightstrike

Summary Files Reviews Support Wiki Mailing Lists Tickets News Discussion Donate Code Git

Download Latest Version
mingw-w64-v6.0.2.zip (15.3 MB)

Get Updates

Home

Name	Modified	Size	Downloads / Week
Junk	2018-05-10		1
Multilib Toolchains(Targetting Win32 and Win64)	2016-05-19		45
mingw-w64	2015-07-25		33,705
Cygwin Snapshots	2013-07-07		0
3rd party development tools	2012-08-07		5
External binary packages (Win64 hosted)	2011-11-15		120
Toolchains targeting Win64	2011-05-06		59,614
Toolchains targeting Win32	2011-05-06		107,355
Toolchain sources	2011-05-06		6
Toolchains targeting NonWin	2010-12-17		0
README.rst	2018-05-24	7.0 kB	54

Recommended Projects

- MinGW - Minimalist GNU for Windows
A native Windows port of the GNU Compiler Collection...
- MinGW-builds
Dual-target(32 & 64-bit) MinGW-W64 compilers for 3...
- TDM-GCC MinGW Compiler
GCC for 32-bit and 64-bit Windows with a real installer ...
- Windows Terminal
The New Windows Terminal

Download MinGW-W64 Online Installer

The screenshot shows the SourceForge project page for MinGW-w64. The browser address bar displays `sourceforge.net/projects/mingw-w64/files/`. The page features a navigation bar with the SourceForge logo and links for 'Open Source Software', 'Business Software', and 'Resources'. A green button labeled 'Download Latest Version' (mingw-w64-v8.0.2.zip (15.3 MB)) and a blue 'Get Updates' button are visible. Below the navigation bar, a table lists various files and folders. The 'minGW-w64' folder is highlighted. A red box highlights the 'MinGW-W64 Online Installer' link, which points to 'MinGW-W64-install.exe'. Below this, the 'MinGW-W64 GCC-8.1.0' section is visible, listing several compiler packages.

Name	Modified	Size	Downloads / Week
Junk	2018-05-10		1
Multilib Toolchains(Targetting Win32 and Win64)	2016-05-19		45
mingw-w64	2015-07-25		33,705
Cygwin Snapshots	2013-07-07		0
3rd party development tools	2012-08-07		5
External binary packages (Win64 hosted)	2011-11-15		120
Toolchains targeting Win64	2011-05-06		59,614
Toolchains targeting Win32	2011-05-06		107,355
Toolchain sources	2011-05-06		6
Toolchains targeting NonWin	2010-12-17		0
README.rst	2018-05-24	7.0 kB	54
MinGW-W64 Online Installer		7.0 kB	200,905

MinGW-W64 Online Installer

- MinGW-W64-install.exe

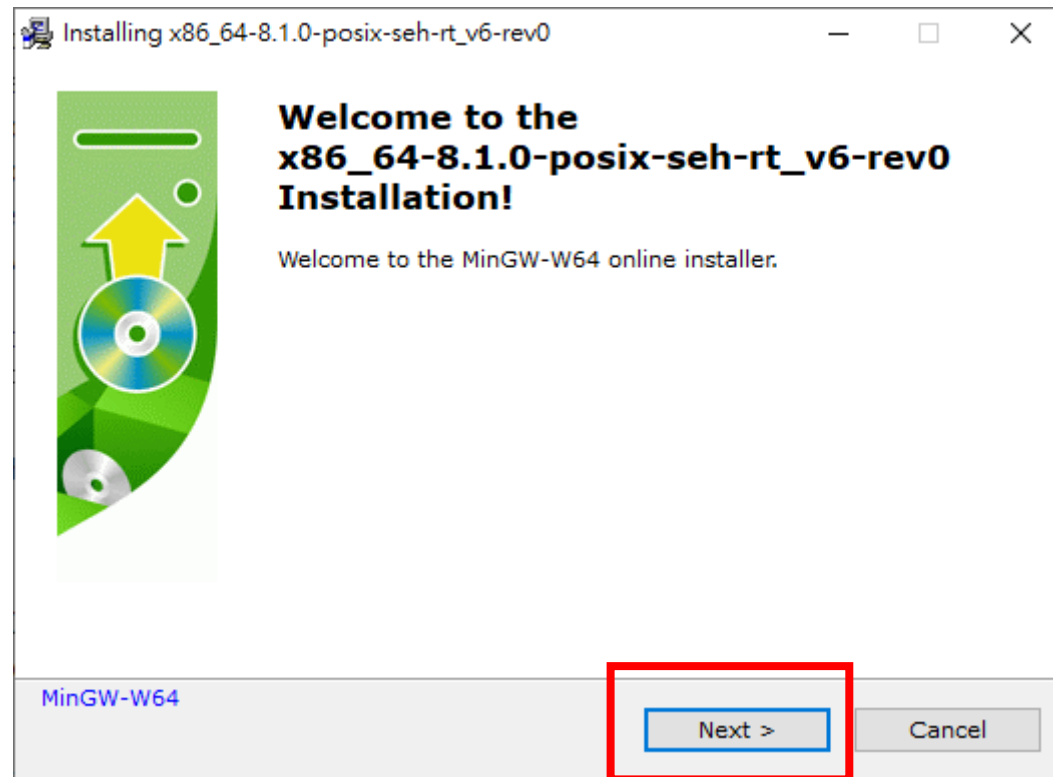
MinGW-W64 GCC-8.1.0

- x86_64-posix-sjlj
- x86_64-posix-seh
- x86_64-win32-sjlj
- x86_64-win32-seh

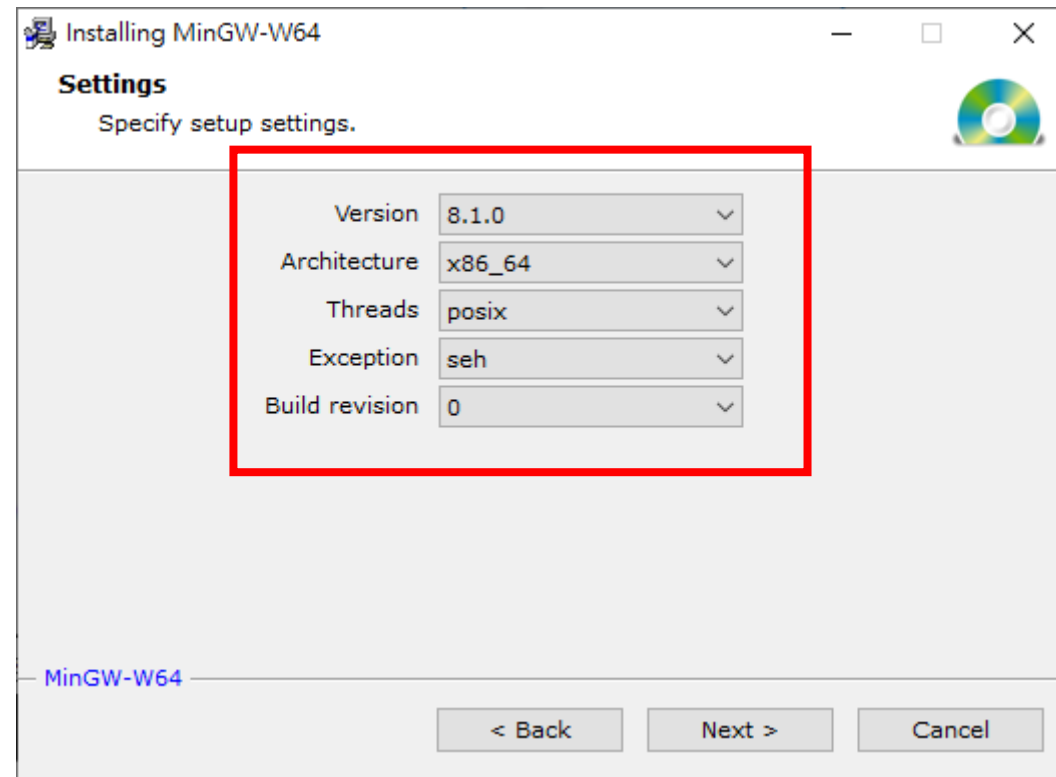
Recommended Projects

- MinGW - Minimalist GNU for Windows
A native Windows port of the GNU Compiler Collection...
- MinGW-builds
Dual-target(32 & 64-bit) MinGW-W64 compilers for 3...
- TDM-GCC MinGW Compiler
GCC for 32-bit and 64-bit Windows with a real installer ...
- Windows Terminal
The New Windows Terminal

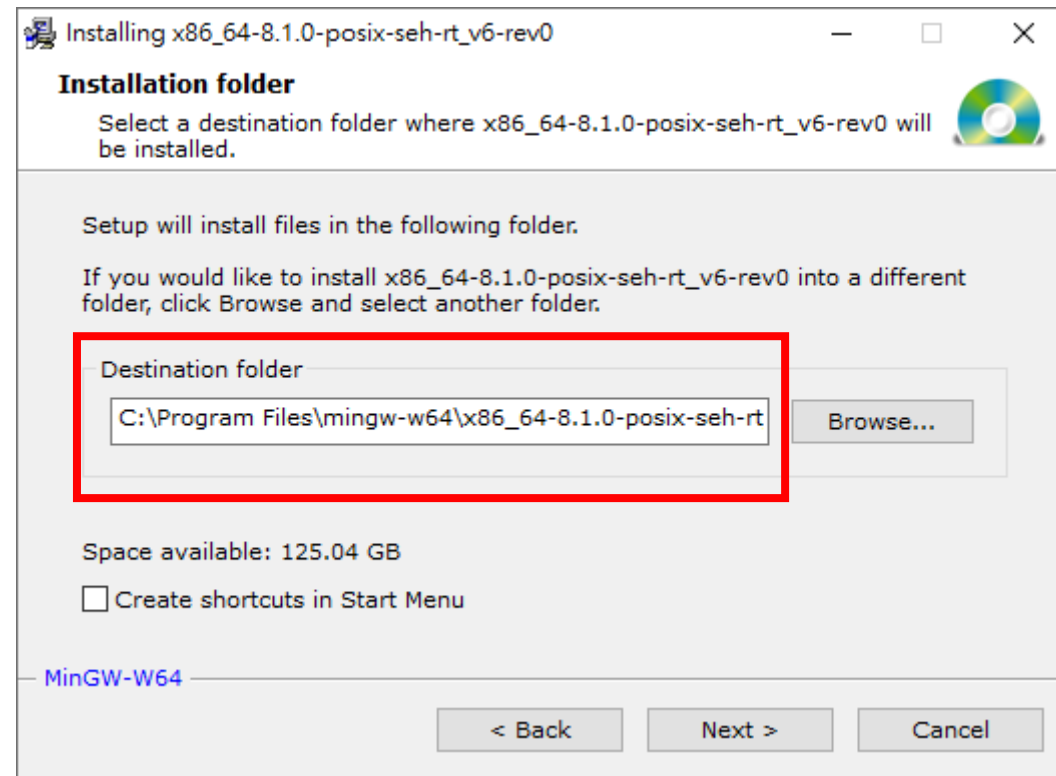
Start the installer, click next



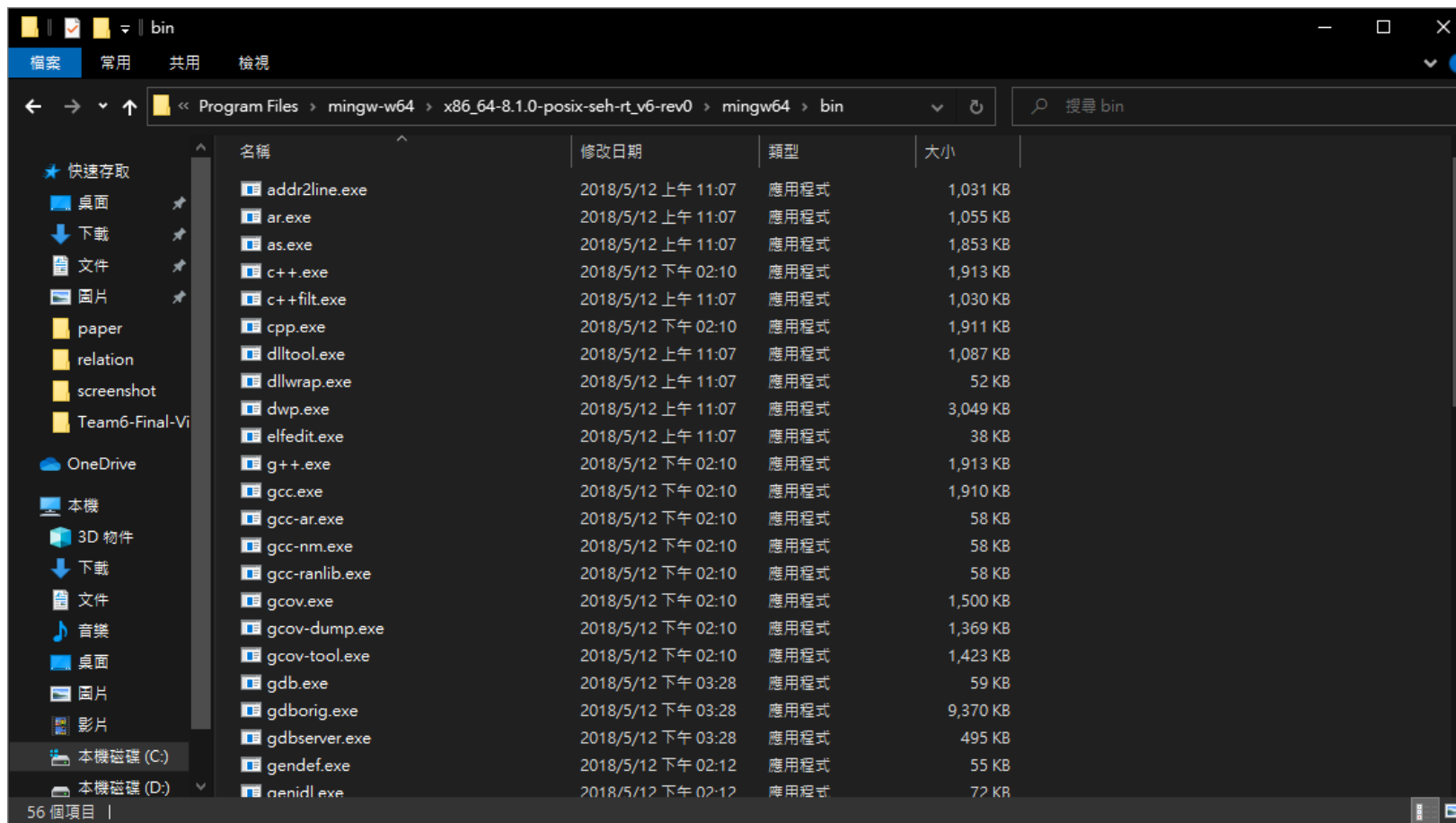
Pick right options according to the screenshot



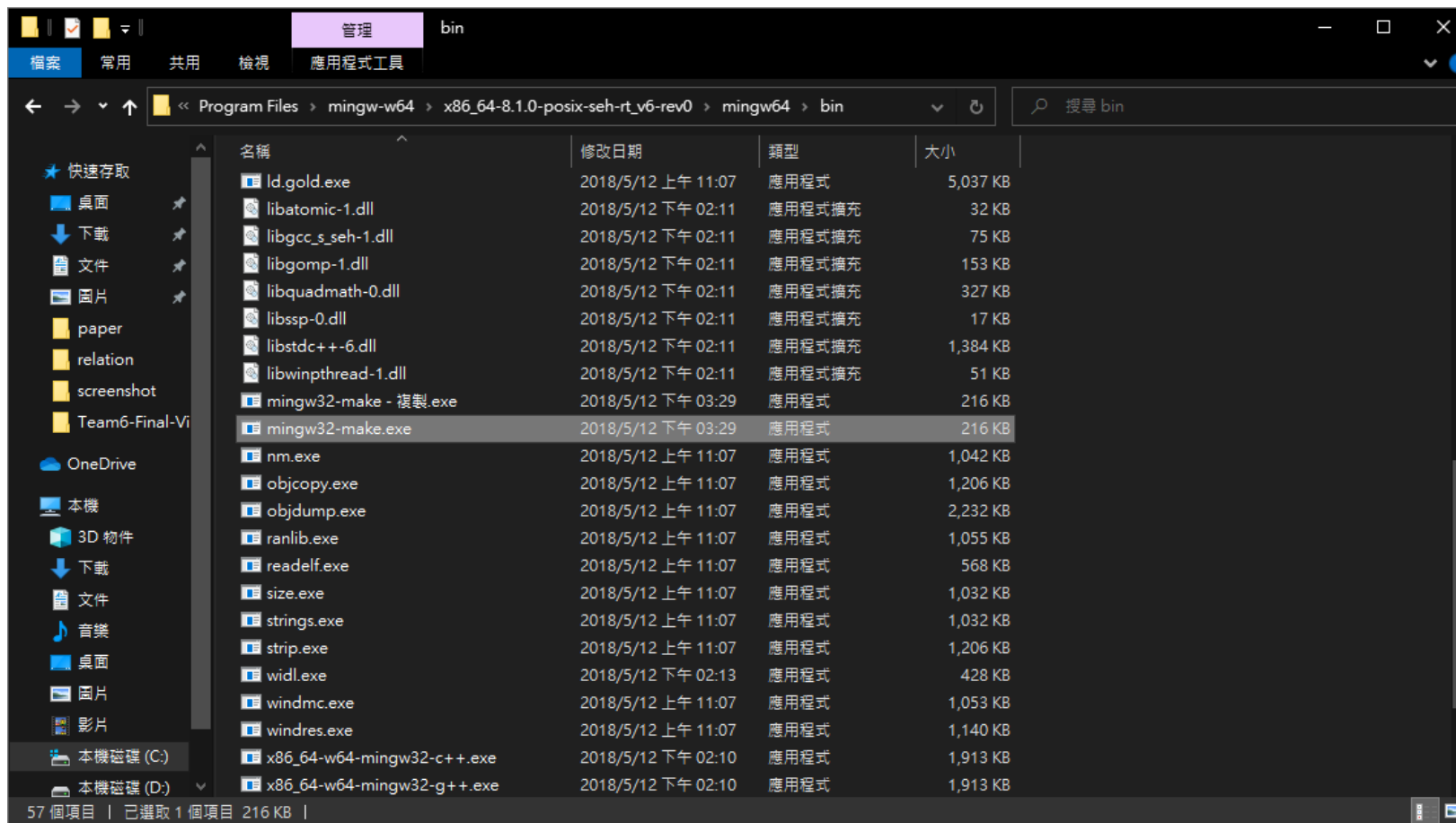
Make sure you remember the install path



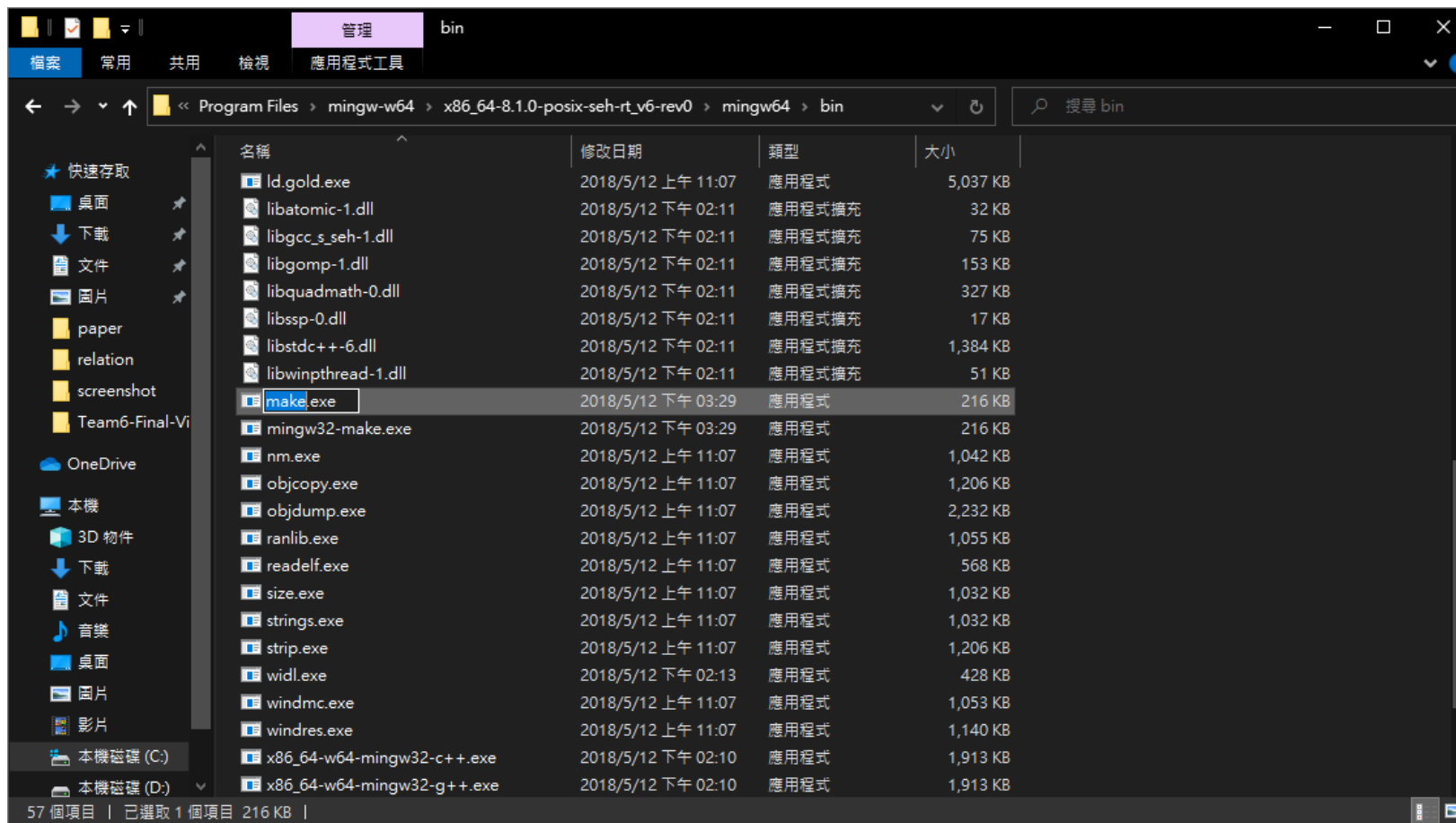
Go to “[install_path]/mingw64/bin”



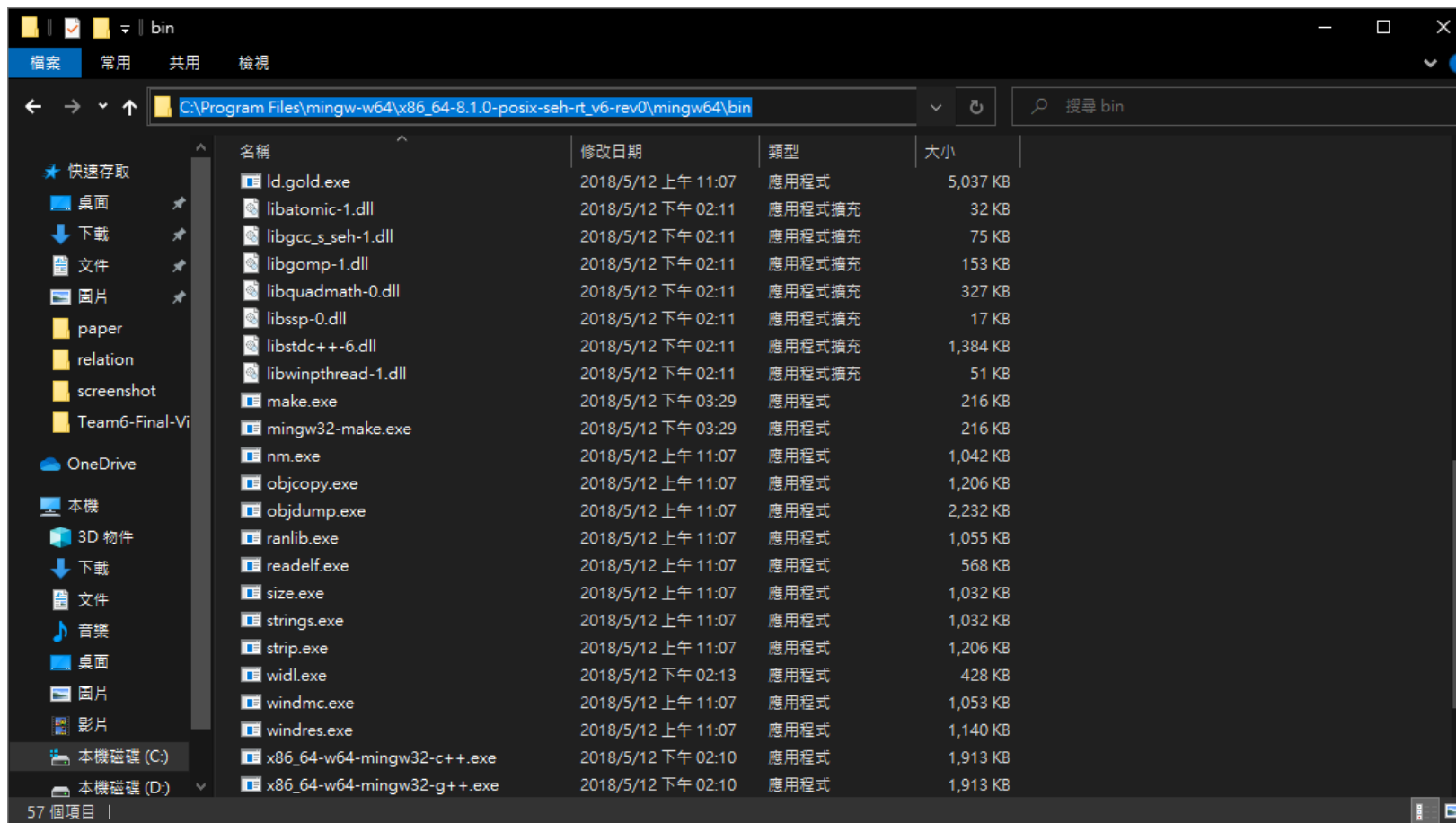
Find “mingw32-make.exe” and copy it



Rename the copied file to “make.exe”



Copy the path of this folder



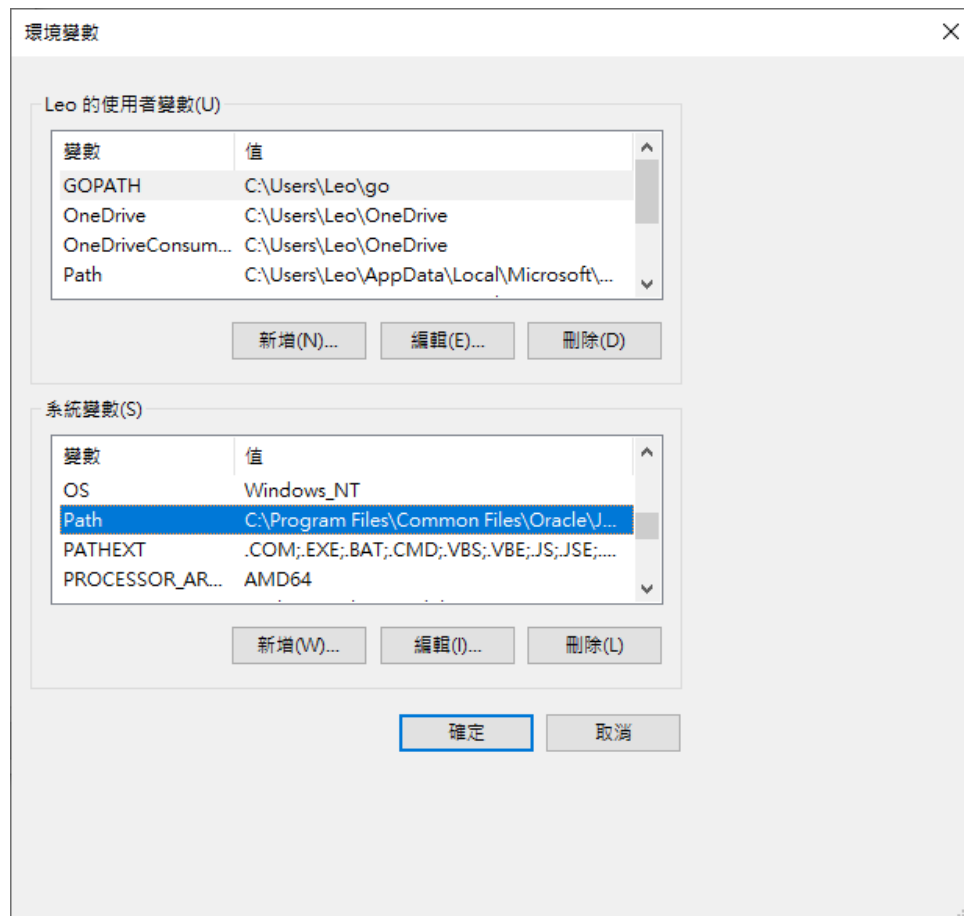
Search for environment variable



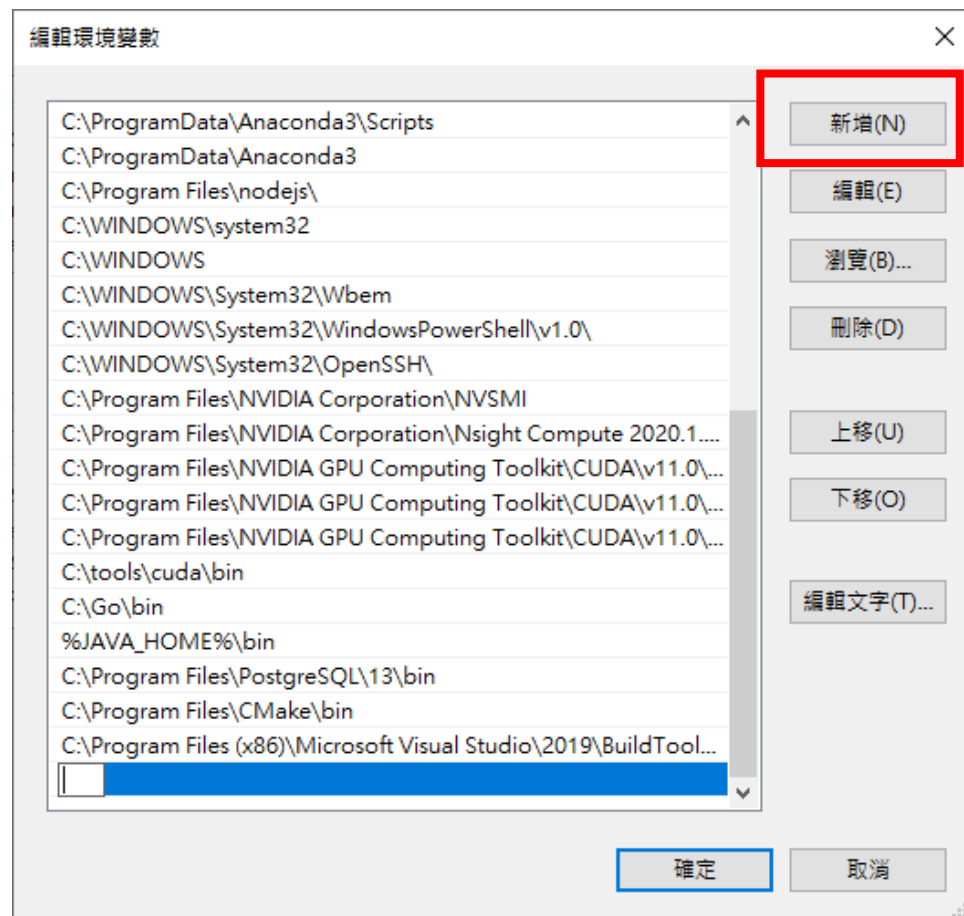
Click the “environment variable” button



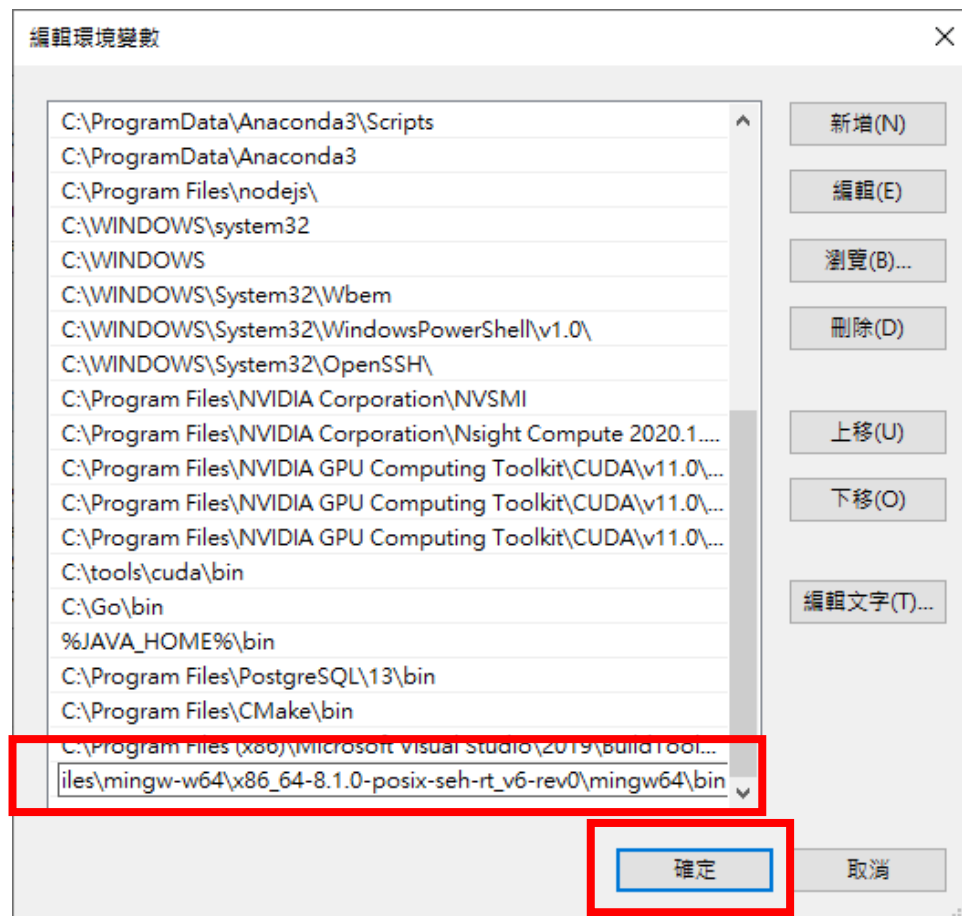
Double click “Path”



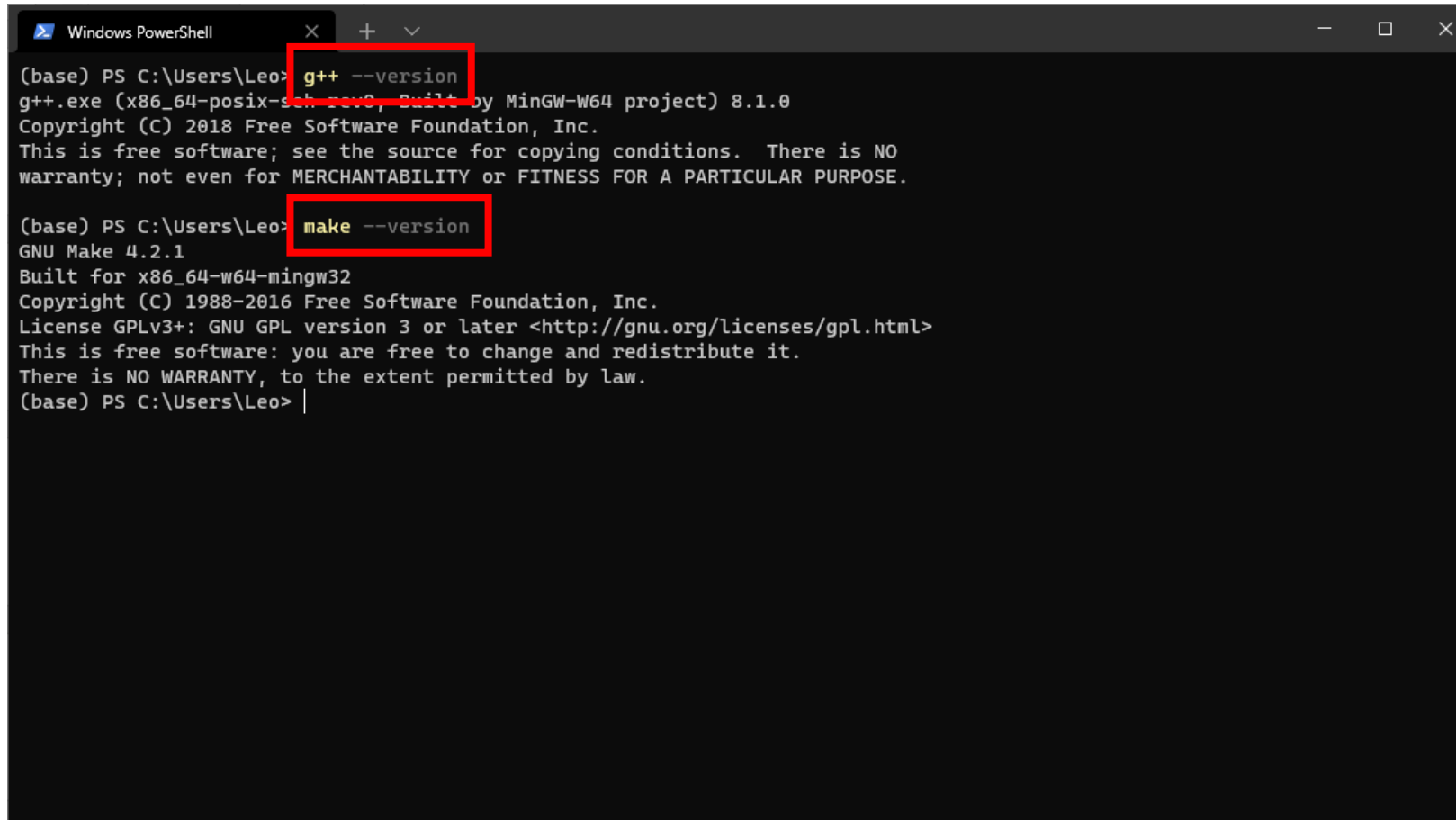
Click “New”



Paste the path and click “OK”



Open cmd or PowerShell to validate installation



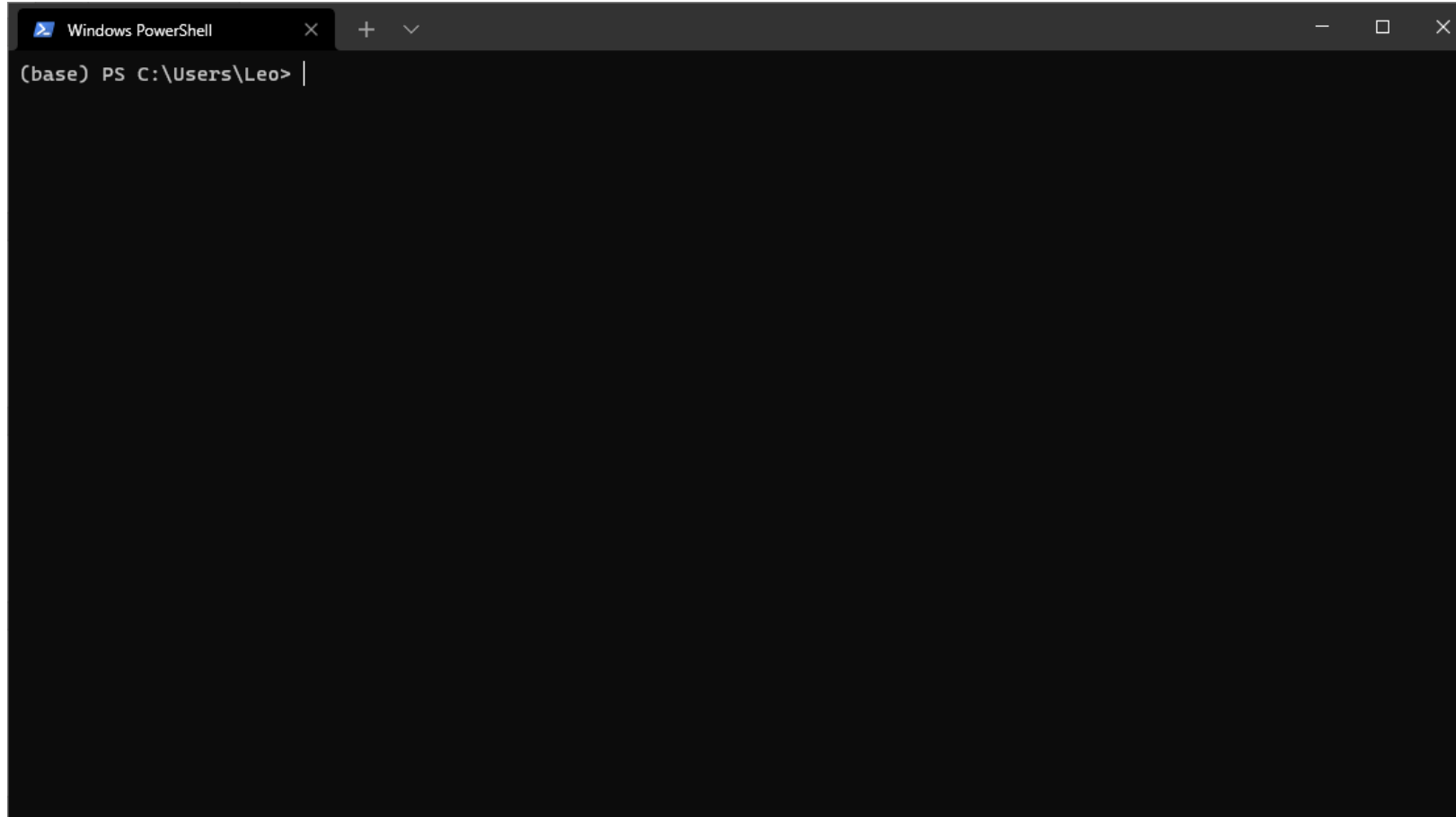
```
Windows PowerShell
(base) PS C:\Users\Leo> g++ --version
g++.exe (x86_64-posix-seh-rev0, Built by MinGW-W64 project) 8.1.0
Copyright (C) 2018 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

(base) PS C:\Users\Leo> make --version
GNU Make 4.2.1
Built for x86_64-w64-mingw32
Copyright (C) 1988-2016 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
(base) PS C:\Users\Leo> |
```

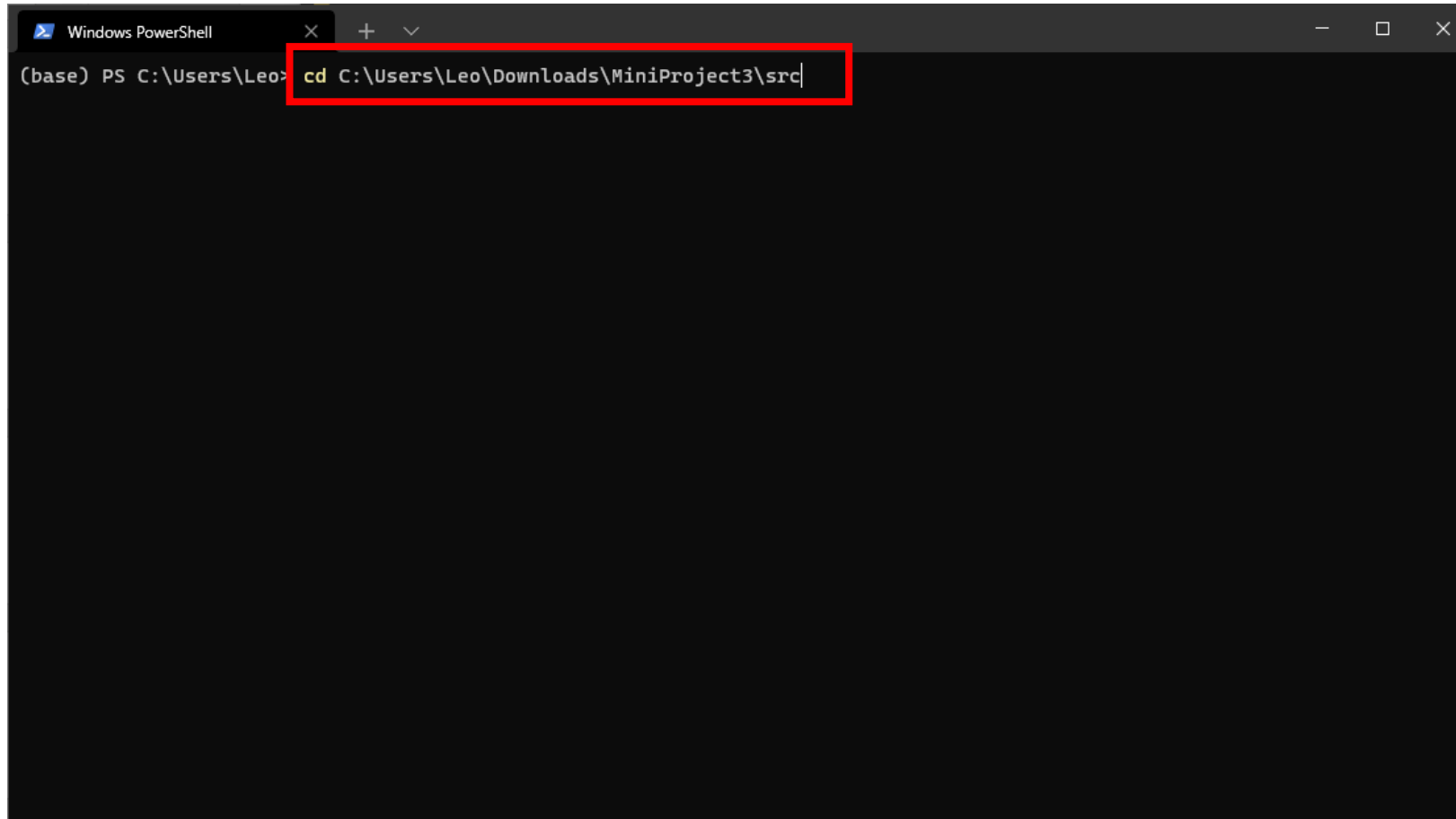
For Windows Users

1. Install mingw-w64
2. Compile your code
3. Execute your code
4. Troubleshooting

Open cmd or Powershell



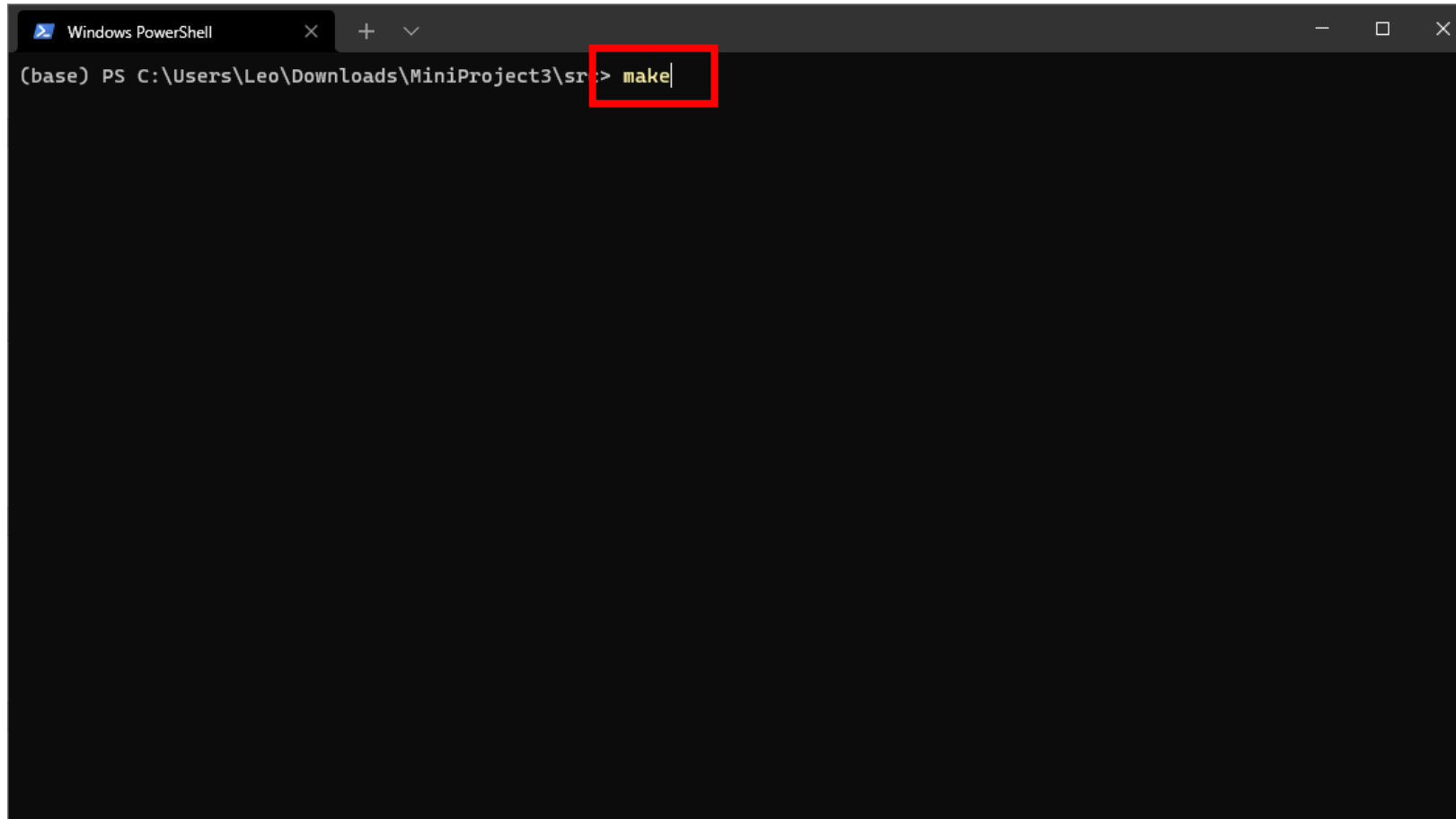
Change directory by entering
“cd [project_path]/src”



A screenshot of a Windows PowerShell terminal window. The title bar at the top reads "Windows PowerShell" and includes standard window controls (minimize, maximize, close). The terminal content shows the prompt "(base) PS C:\Users\Leo>" followed by the command "cd C:\Users\Leo\Downloads\MiniProject3\src". The command is highlighted with a red rectangular box. The terminal background is dark gray.

```
(base) PS C:\Users\Leo> cd C:\Users\Leo\Downloads\MiniProject3\src
```

Type “**make**” to compile your code



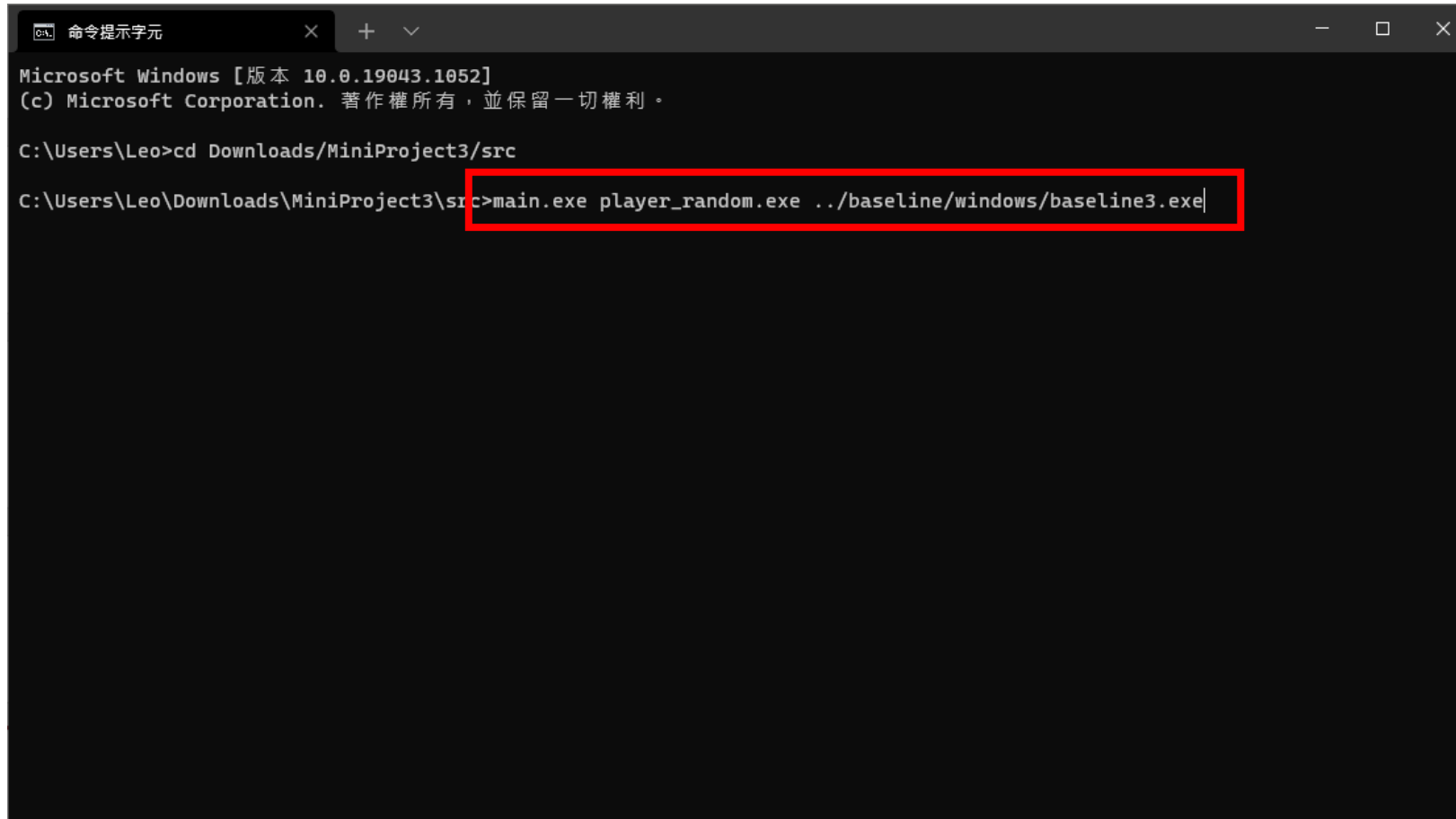
A screenshot of a Windows PowerShell terminal window. The title bar at the top reads "Windows PowerShell" and includes standard window controls (minimize, maximize, close). The terminal content shows the prompt "(base) PS C:\Users\Leo\Downloads\MiniProject3\src:" followed by the command "make" which is highlighted by a red rectangular box. The rest of the terminal area is dark and empty.

```
(base) PS C:\Users\Leo\Downloads\MiniProject3\src:> make|
```


For Windows Users

1. Install mingw-w64
2. Compile your code
3. Execute your code
4. Troubleshooting

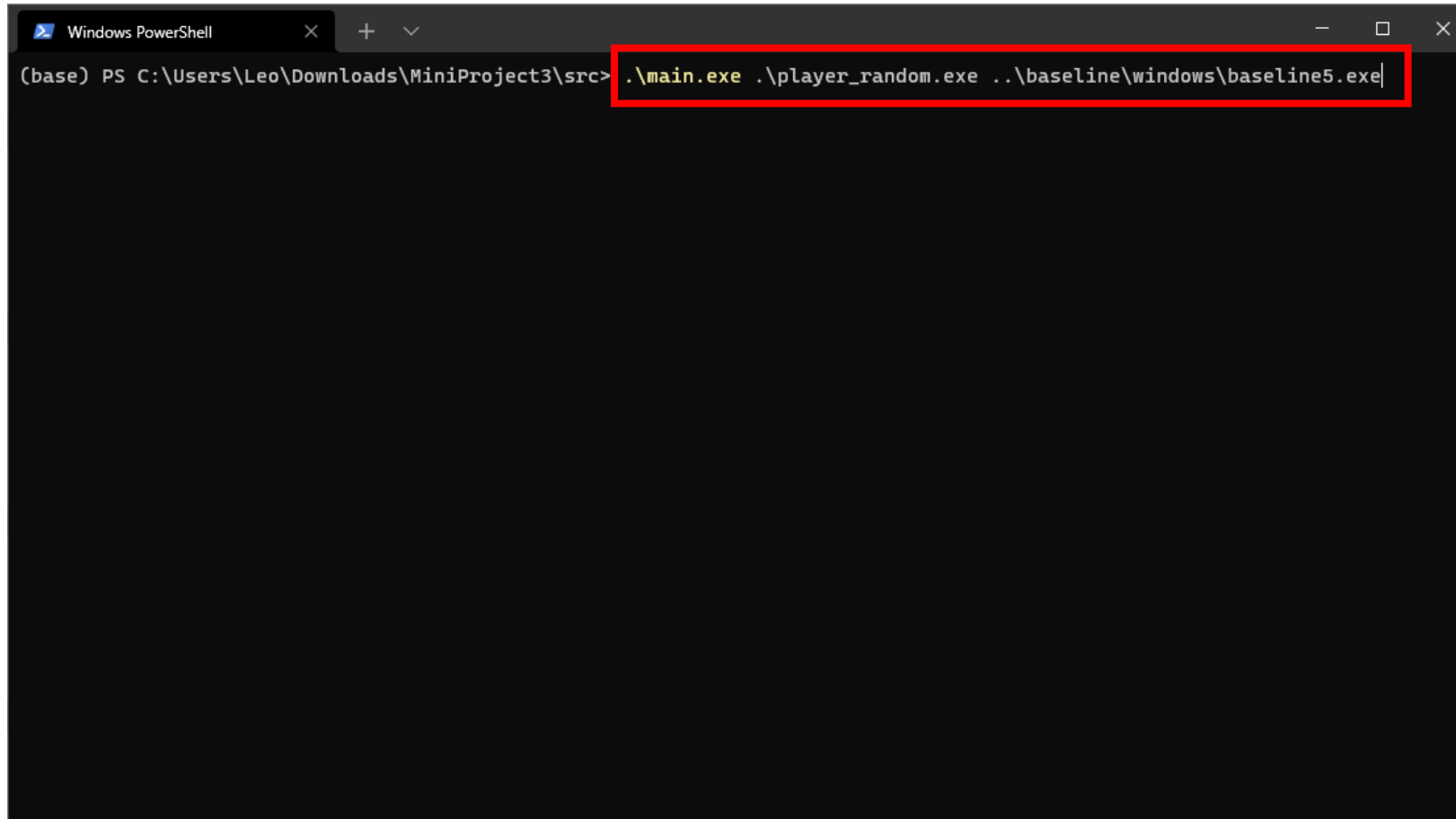
If you use cmd, type “**main.exe [AI1].exe [AI2].exe**” while in the “src” folder of your project



```
命令提示字元
Microsoft Windows [版本 10.0.19043.1052]
(c) Microsoft Corporation. 著作權所有，並保留一切權利。

C:\Users\Leo>cd Downloads/MiniProject3/src
C:\Users\Leo\Downloads\MiniProject3\src>main.exe player_random.exe ../baseline/windows/baseline3.exe|
```

If you use PowerShell, type “**.\main.exe** [AI1].exe [AI2].exe” while in the “src” folder of your project



```
Windows PowerShell
(base) PS C:\Users\Leo\Downloads\MiniProject3\src> .\main.exe .\player_random.exe ..\baseline\windows\baseline5.exe|
```

The screenshot shows a Windows PowerShell terminal window with a dark background. The title bar at the top reads "Windows PowerShell". The command prompt shows the current directory as "C:\Users\Leo\Downloads\MiniProject3\src" and the command ".\main.exe .\player_random.exe ..\baseline\windows\baseline5.exe|" is being typed. The command is highlighted with a red rectangular box.

For Windows Users

1. Install mingw-w64
2. Compile your code
3. Execute your code
4. Troubleshooting

Error removing file: (state/action)

- If you encounter this problem, do these steps and compile again

1. Comment line 255, 256 in main.cpp

```
254 // Reset action file
255 if (remove(file_action.c_str()) != 0)
256     std::cerr << "Error removing file: " << file_action << "\n";
257 // Take action
```

2. Comment line 271, 272 in main.cpp

```
270 // Reset state file
271 if (remove(file_state.c_str()) != 0)
272     std::cerr << "Error removing file: " << file_state << "\n";
273 return 0;
```

Error removing file: (state/action)

- If the solution on the previous page still cannot solve the problem
- You can try increasing the defined timeout limit on line 9 to 5 ~ 10

```
8  
9  #define TIMEOUT 10  
10
```

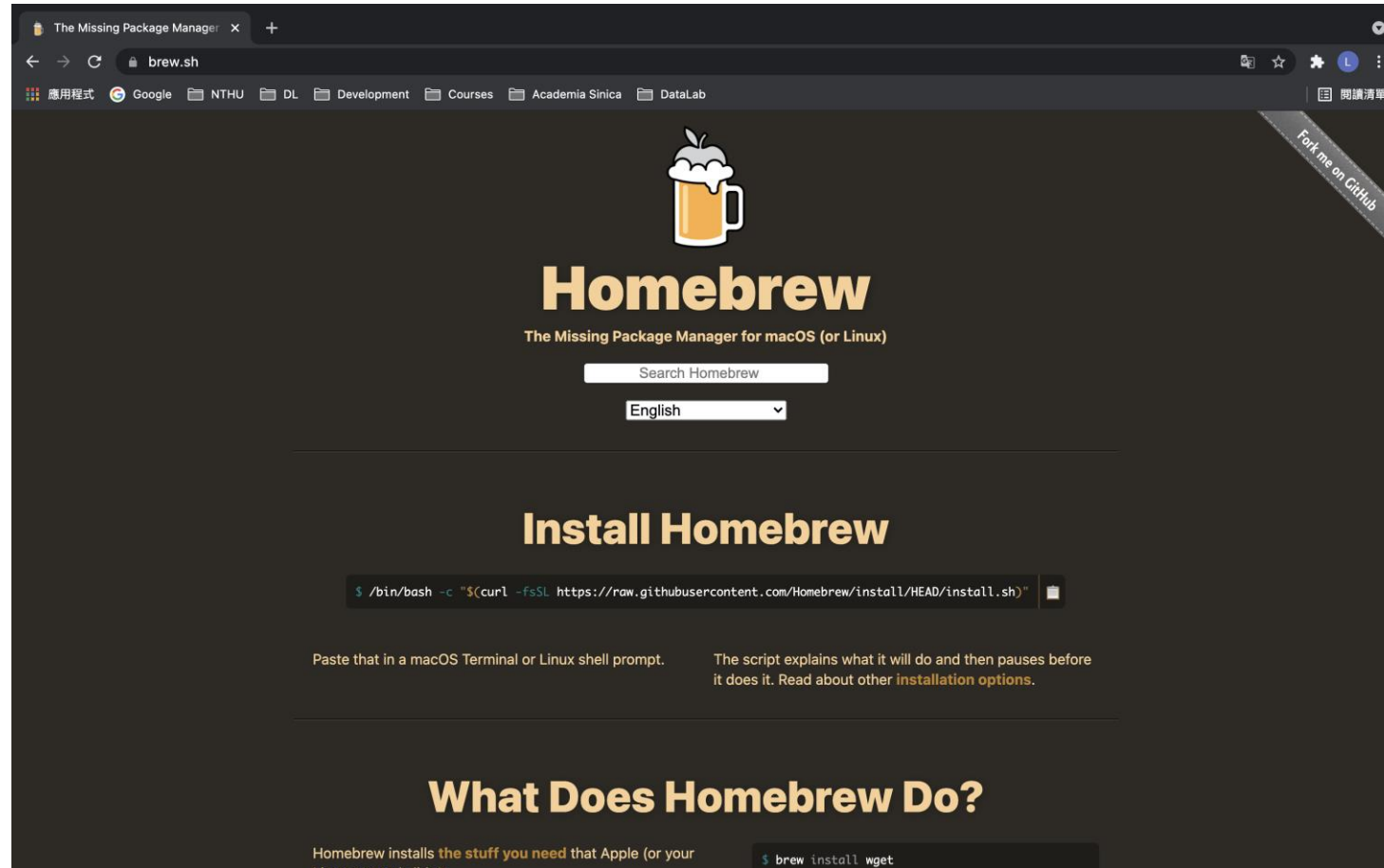
- This might solve the problem

For Mac Users

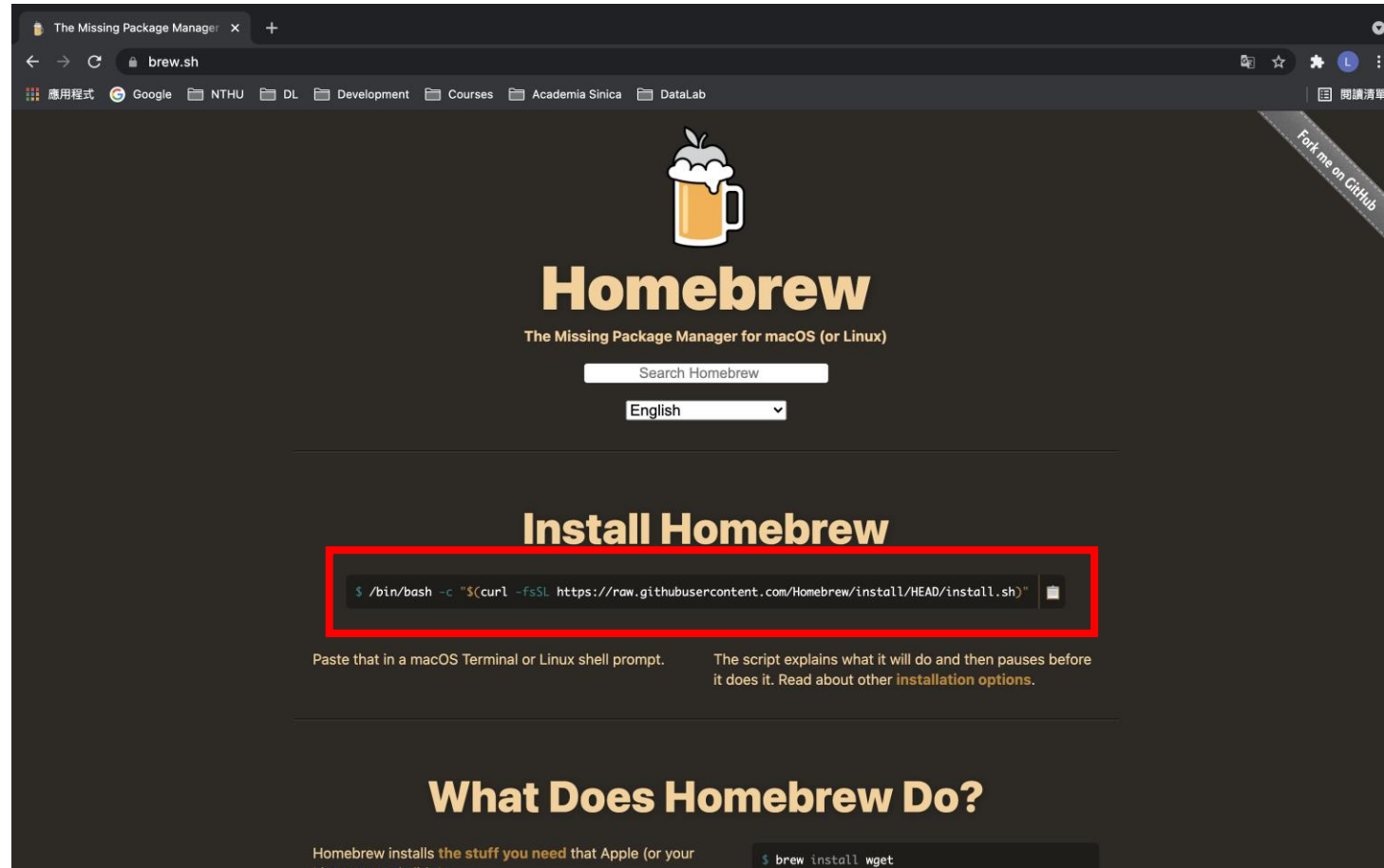
For Mac Users

1. Install Homebrew
2. Install Coreutils
3. Modify file permissions
4. Compile your code
5. Execute your code

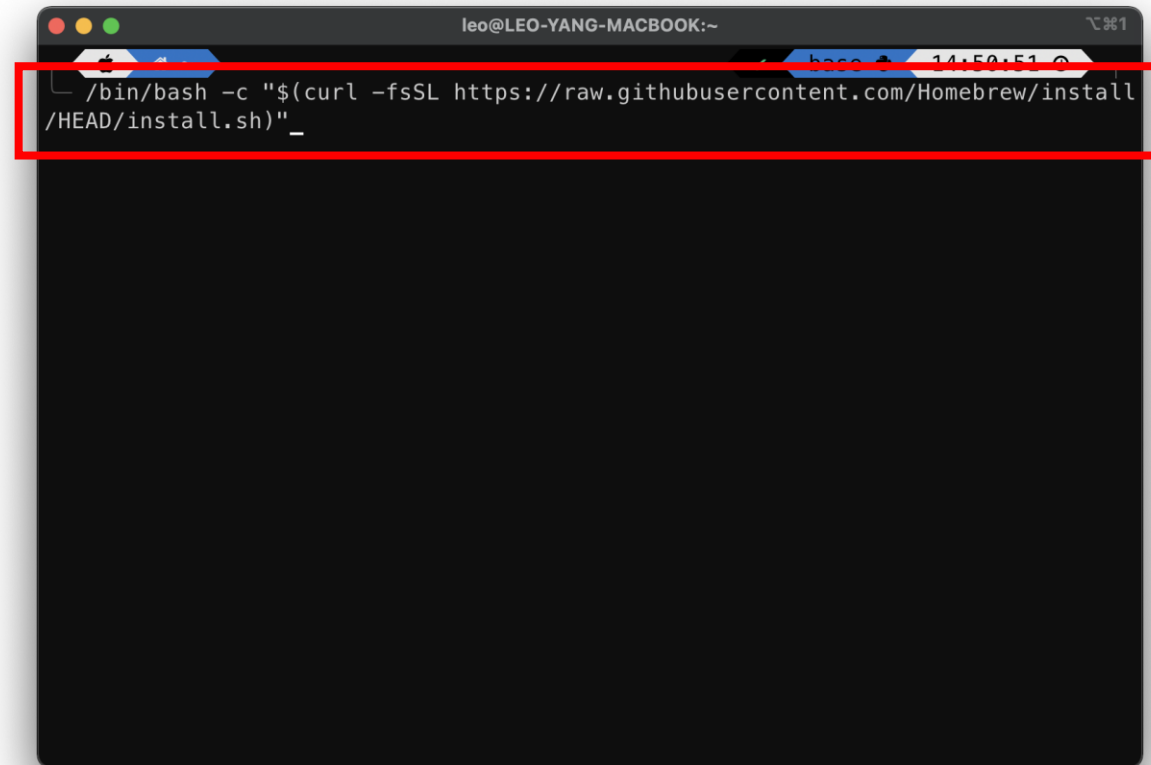
Go to [Homebrew homepage](https://brew.sh)



Copy the install command

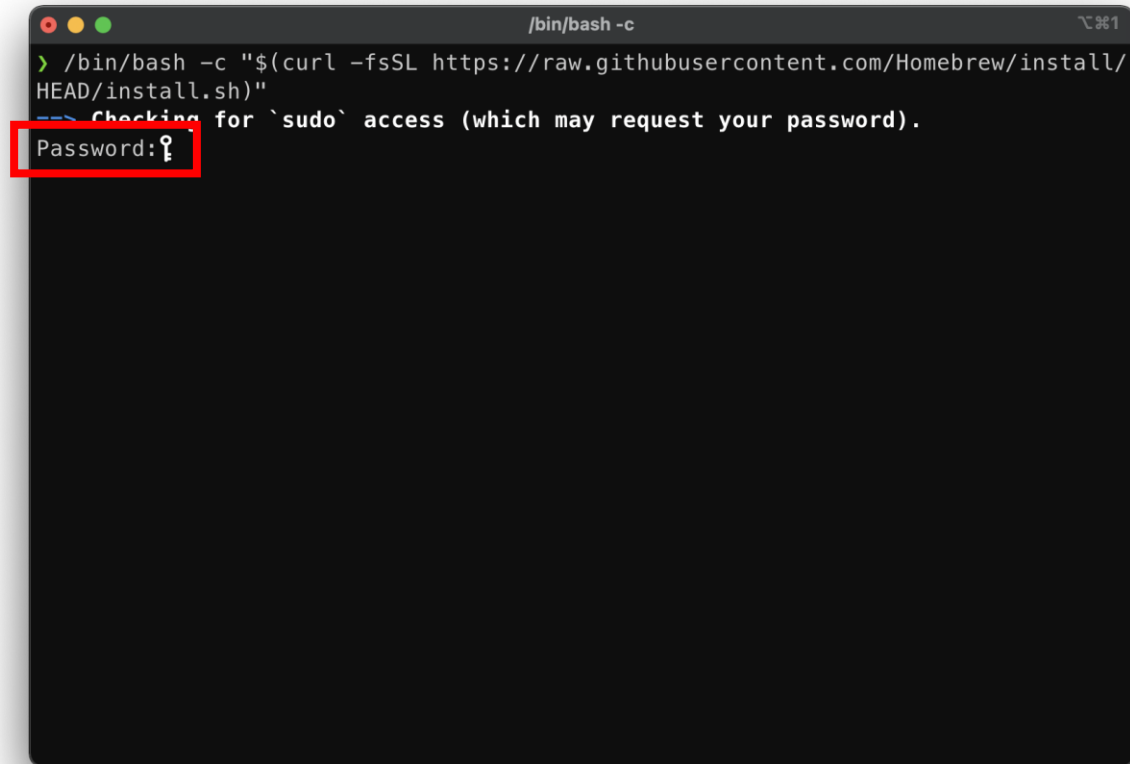


Paste the command in terminal and enter

A screenshot of a macOS terminal window. The title bar at the top reads 'leo@LEO-YANG-MACBOOK:~'. The terminal shows a command being entered: `/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"`. The command is highlighted with a red rectangular box. The terminal background is dark, and the text is white. The cursor is at the end of the command line.

```
leo@LEO-YANG-MACBOOK:~  
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

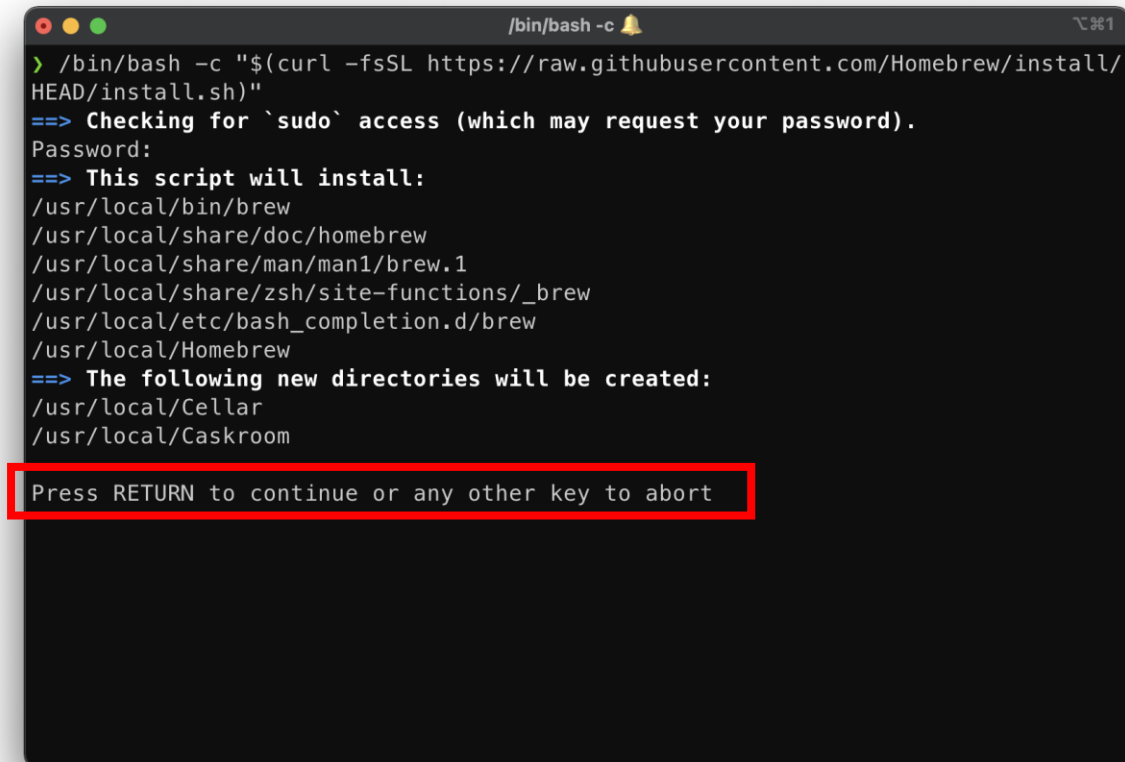
Enter your password





```
/bin/bash -c
> /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
--> Checking for `sudo` access (which may request your password).
Password: 
```

A terminal window with a dark background and light gray text. The window title bar shows standard macOS window controls (red, yellow, green buttons) and the text `/bin/bash -c`. The terminal content shows the execution of a command to install Homebrew. The prompt `>` is followed by the command `/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"`. Below this, the text `--> Checking for `sudo` access (which may request your password).` is displayed. The next line shows the prompt `Password:` followed by a cursor. A red rectangular box highlights the `Password:` text and the cursor.

Press enter to install

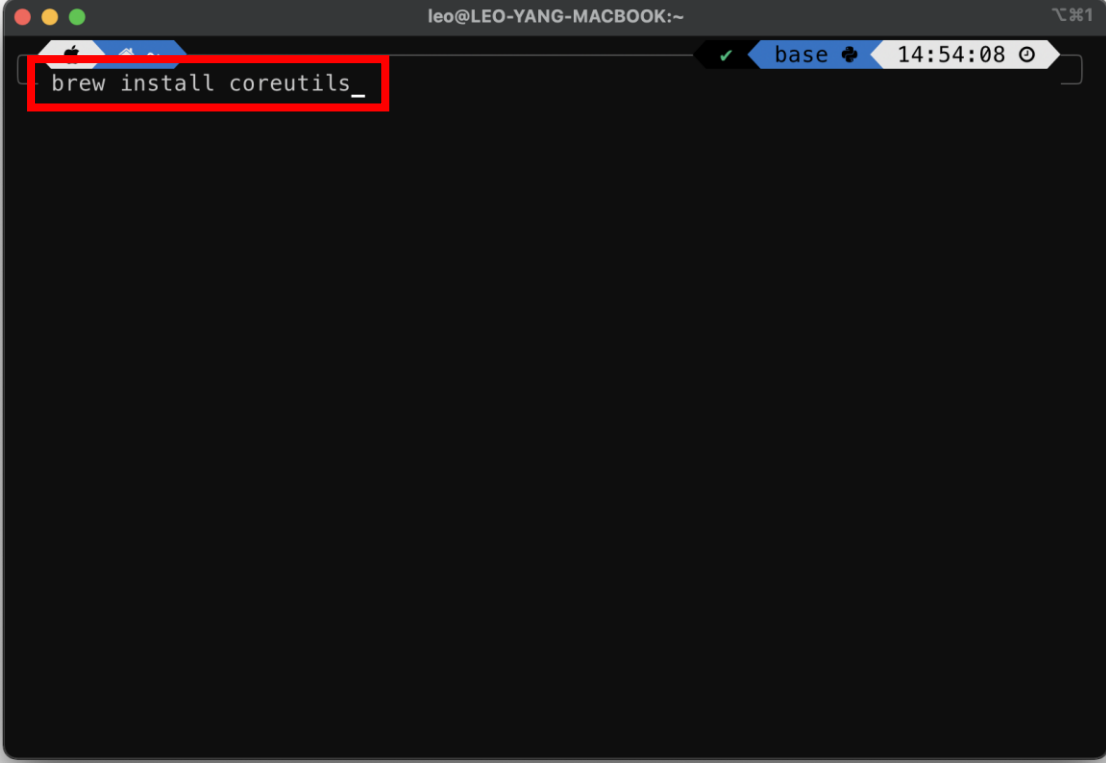
A terminal window with a dark background and light text. The title bar shows "/bin/bash -c" and a bell icon. The terminal content shows the execution of a curl command to fetch the Homebrew install script. It then prompts for a password, lists the files to be installed, and lists the new directories to be created. The final line, "Press RETURN to continue or any other key to abort", is highlighted with a red rectangular box.

```
/bin/bash -c  
> /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
==> Checking for `sudo` access (which may request your password).
Password:
==> This script will install:
/usr/local/bin/brew
/usr/local/share/doc/homebrew
/usr/local/share/man/man1/brew.1
/usr/local/share/zsh/site-functions/_brew
/usr/local/etc/bash_completion.d/brew
/usr/local/Homebrew
==> The following new directories will be created:
/usr/local/Cellar
/usr/local/Caskroom
Press RETURN to continue or any other key to abort
```

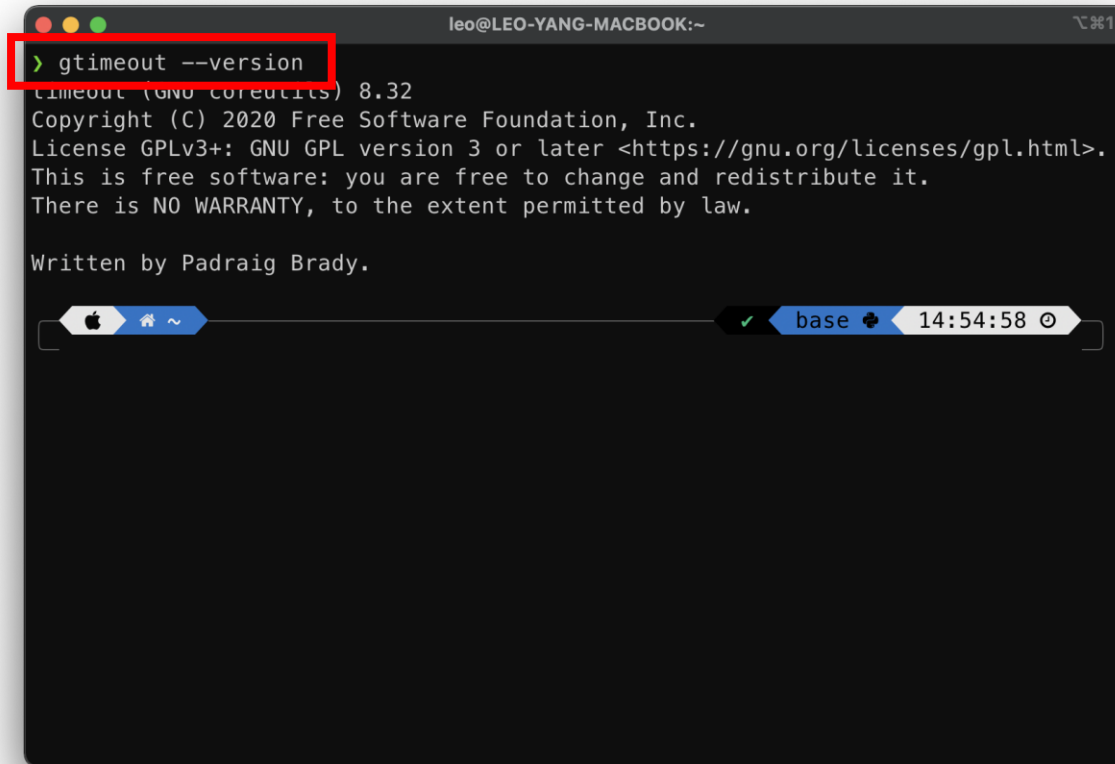
For Mac Users

1. Install Homebrew
2. Install Coreutils
3. Modify file permissions
4. Compile your code
5. Execute your code

Type “**brew install coreutils**” in terminal

A screenshot of a macOS terminal window. The title bar at the top reads 'leo@LEO-YANG-MACBOOK:~'. The terminal content shows the command 'brew install coreutils_' entered at the prompt. The command is highlighted with a red rectangular box. The terminal interface includes a dark background, a light-colored prompt, and a status bar at the top right showing 'base' and the time '14:54:08'.

Type “**gtimeout --version**”
to check if it works properly

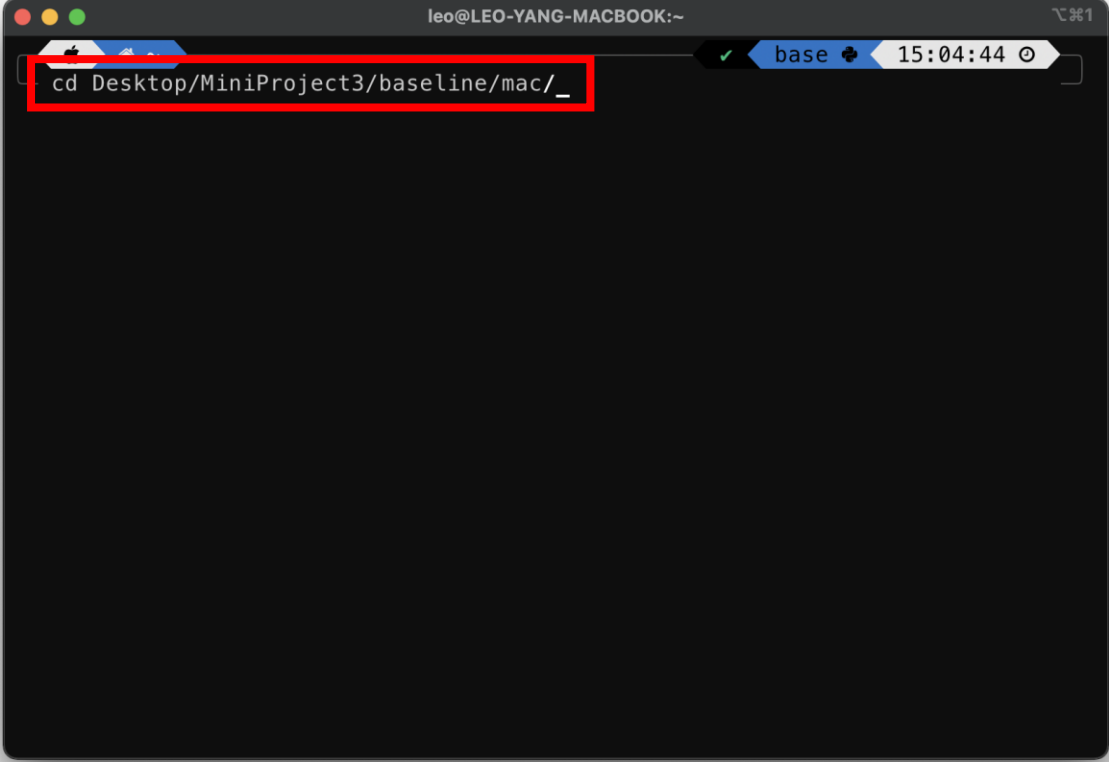
A screenshot of a macOS terminal window. The title bar at the top reads 'leo@LEO-YANG-MACBOOK:~'. The command prompt shows the command 'gtimeout --version' entered, which is highlighted by a red rectangular box. The output of the command is displayed below the prompt: 'gtimeout (GNU coreutils) 8.32', 'Copyright (C) 2020 Free Software Foundation, Inc.', 'License GPLv3+: GNU GPL version 3 or later <https://gnu.org/licenses/gpl.html>.', 'This is free software: you are free to change and redistribute it.', 'There is NO WARRANTY, to the extent permitted by law.', and 'Written by Padraig Brady.'. At the bottom of the terminal window, there is a status bar showing the Apple logo, a home icon, a tilde icon, a green checkmark, the text 'base', a plus icon, the time '14:54:58', and a magnifying glass icon.

```
leo@LEO-YANG-MACBOOK:~  
> gtimeout --version  
gtimeout (GNU coreutils) 8.32  
Copyright (C) 2020 Free Software Foundation, Inc.  
License GPLv3+: GNU GPL version 3 or later <https://gnu.org/licenses/gpl.html>.  
This is free software: you are free to change and redistribute it.  
There is NO WARRANTY, to the extent permitted by law.  
  
Written by Padraig Brady.
```


For Mac Users

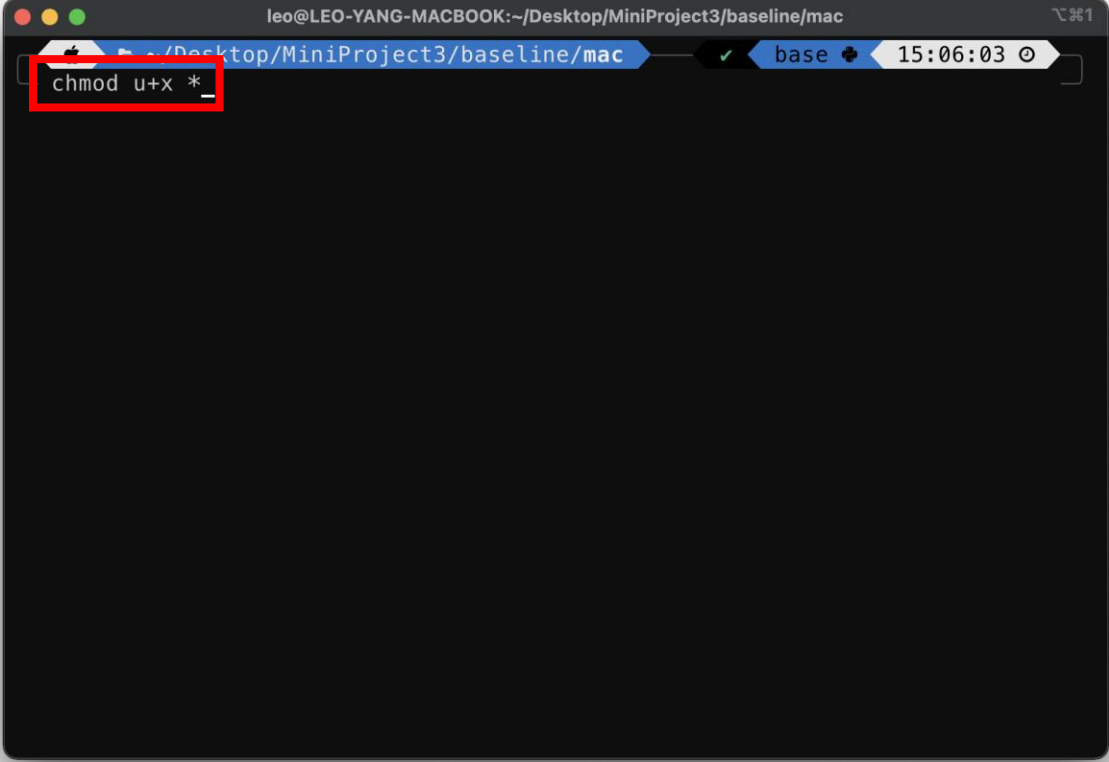
1. Install Homebrew
2. Install Coreutils
3. **Modify file permissions**
4. Compile your code
5. Execute your code

Change directory by entering
“cd [project_path]/baseline/mac” in terminal

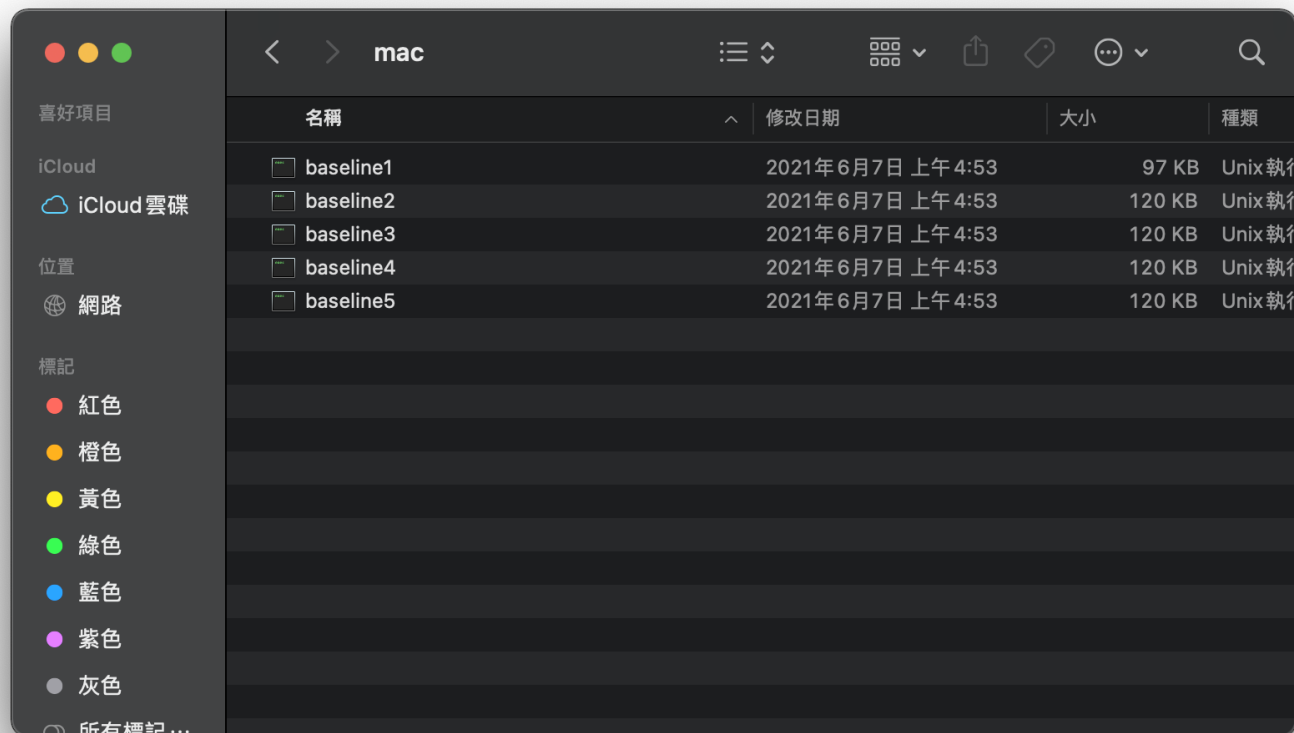
A screenshot of a macOS terminal window. The title bar at the top reads "leo@LEO-YANG-MACBOOK:~". The terminal has a dark background with white text. The command "cd Desktop/MiniProject3/baseline/mac/_ " is entered and is highlighted by a red rectangular box. Above the command line, there is a status bar showing a green checkmark, the word "base", a plus icon, and the time "15:04:44" followed by a magnifying glass icon. The terminal window also features standard macOS window controls (red, yellow, green buttons) in the top-left corner.

```
leo@LEO-YANG-MACBOOK:~  
cd Desktop/MiniProject3/baseline/mac/_ 
```

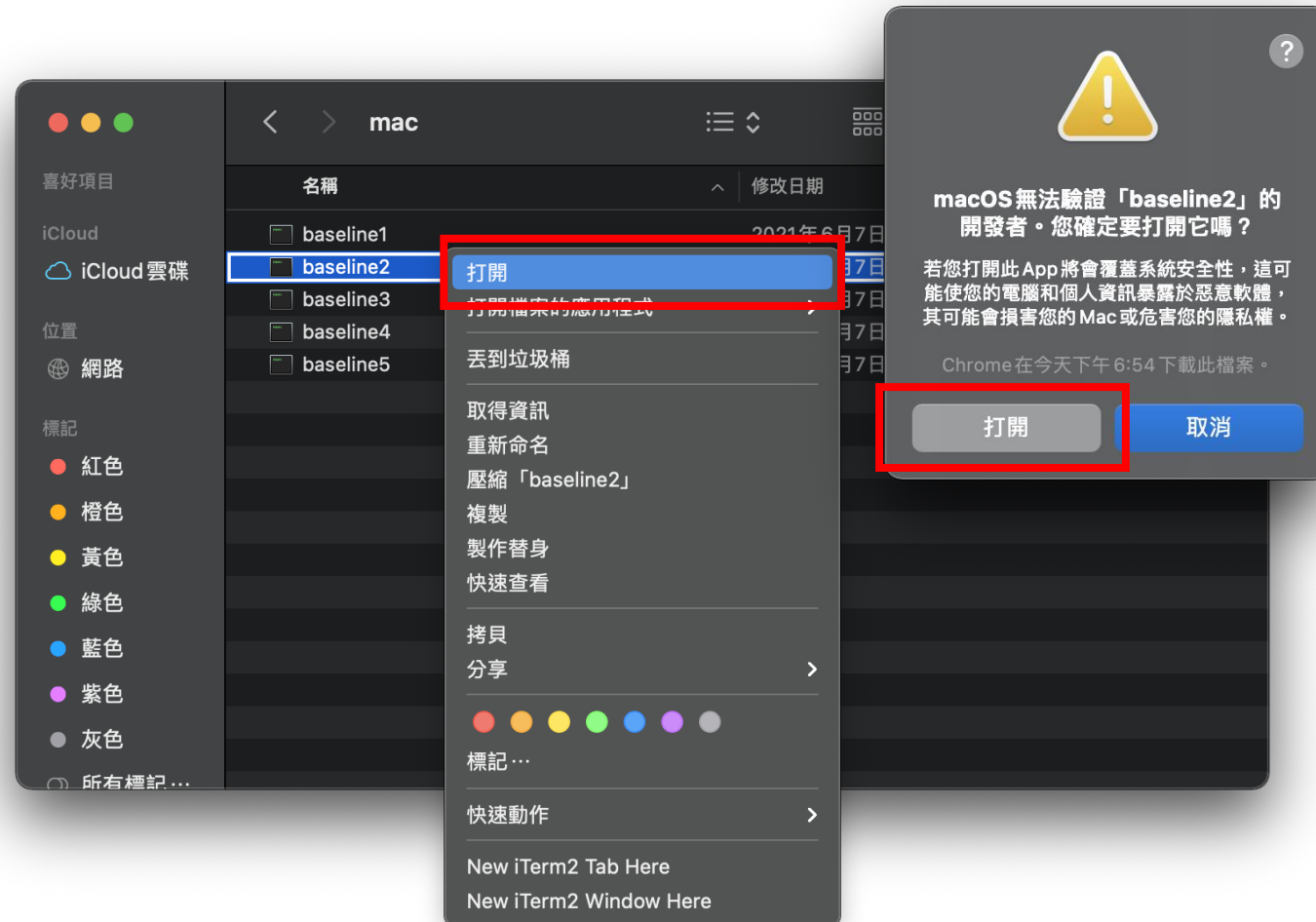
Type “**chmod u+x ***” in terminal
to enable execution permission

A screenshot of a macOS terminal window. The title bar at the top reads 'leo@LEO-YANG-MACBOOK:~/Desktop/MiniProject3/baseline/mac'. The terminal content shows the command 'chmod u+x *_' entered at the prompt. The command is highlighted with a red rectangular box. The terminal has a dark background and a light-colored prompt character. The window also shows standard macOS window controls (red, yellow, green buttons) and a status bar at the bottom with the text 'base' and a time indicator '15:06:03'.

Open “[project_path]/baseline/mac” in Finder



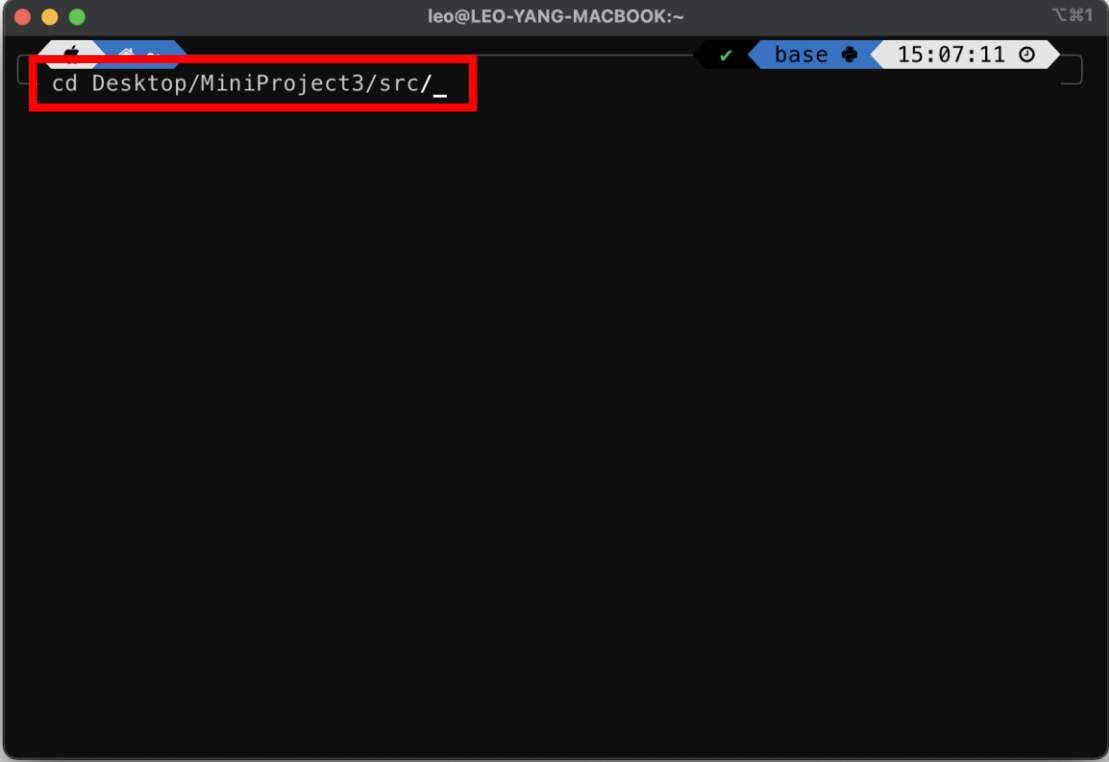
For all **baselines**, right click on the file and click “open”, then click “open” on the pop-up window



For Mac Users

1. Install Homebrew
2. Install Coreutils
3. Modify file permissions
4. **Compile your code**
5. Execute your code

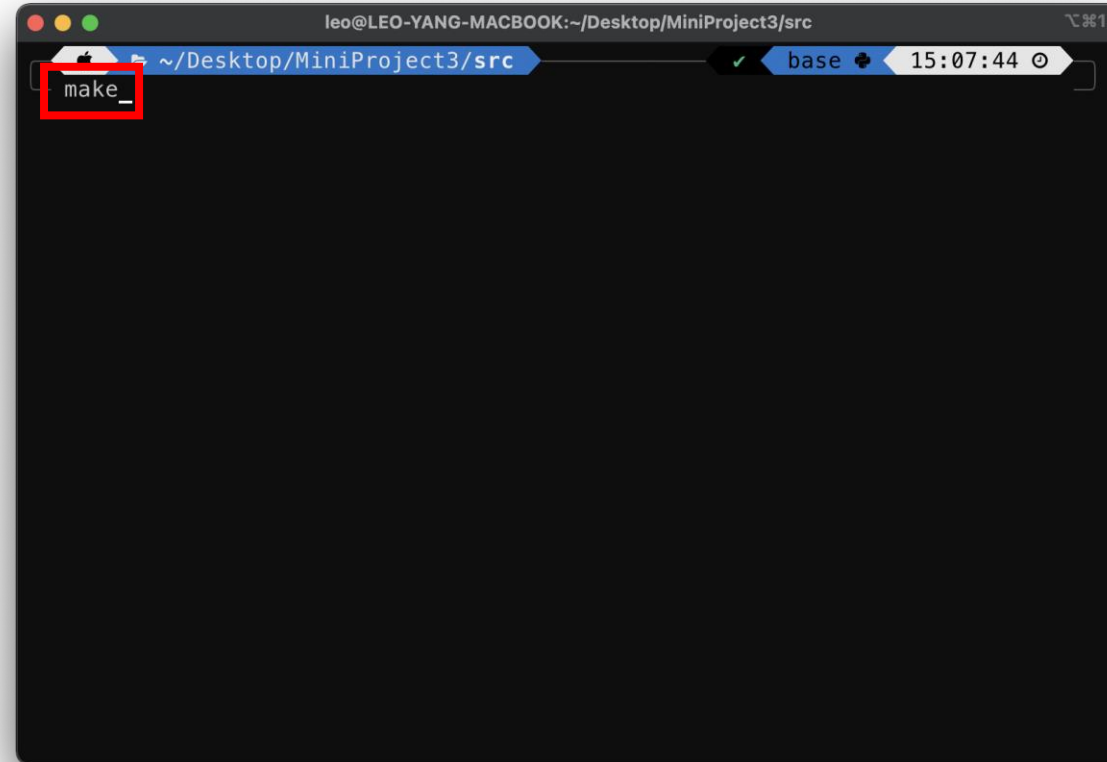
Change directory by entering
“cd [project_path]/src” in terminal



```
leo@LEO-YANG-MACBOOK:~  
base 15:07:11  
cd Desktop/MiniProject3/src/_
```

A terminal window on a Mac. The title bar shows 'leo@LEO-YANG-MACBOOK:~'. The terminal has a dark background with a light blue prompt 'base' and a timestamp '15:07:11'. The command 'cd Desktop/MiniProject3/src/_' is entered and highlighted with a red box.

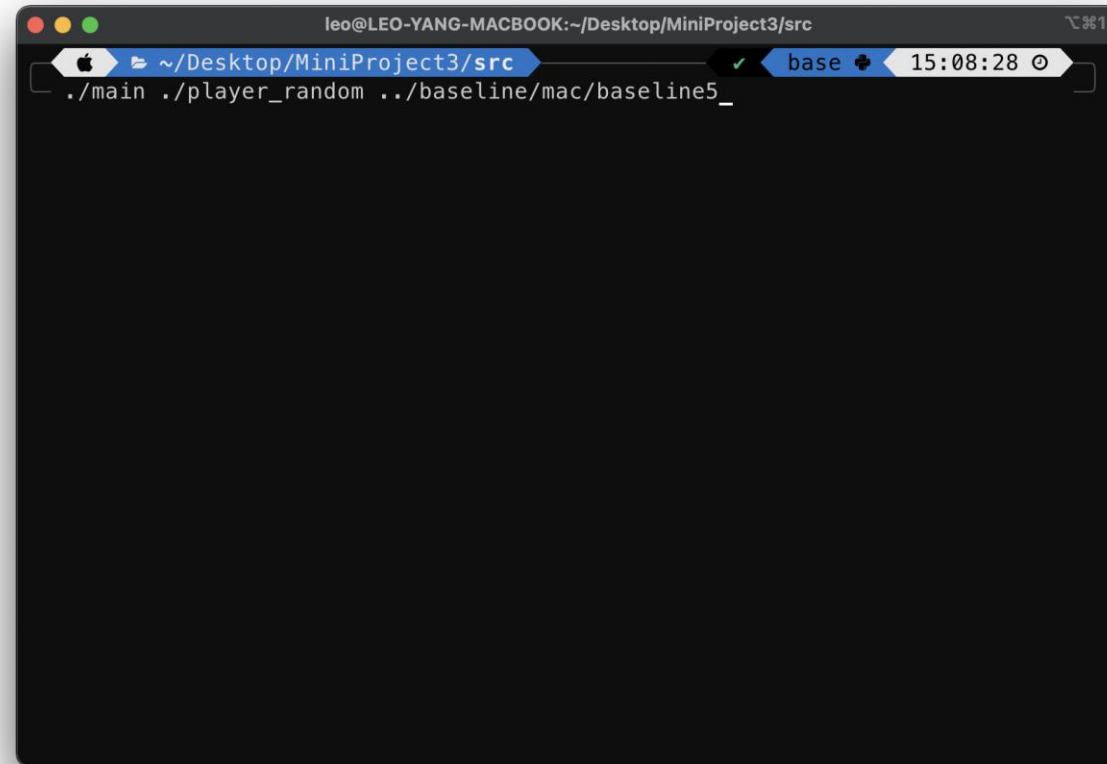
Type “**make**” to compile your code



For Mac Users

1. Install Homebrew
2. Install Coreutils
3. Modify file permissions
4. Compile your code
5. Execute your code

Type “**./main** **./[AI1]** **./[AI2]**” in terminal to execute while in the “src” folder of your project

A screenshot of a macOS terminal window. The title bar at the top reads 'leo@LEO-YANG-MACBOOK:~/Desktop/MiniProject3/src'. Below the title bar is a dark blue bar containing a home icon, the path '~/Desktop/MiniProject3/src', a green checkmark, the word 'base', and a clock icon showing '15:08:28'. The terminal area is black with white text. The command './main ./player_random ../baseline/mac/baseline5_' is entered on the first line.

```
leo@LEO-YANG-MACBOOK:~/Desktop/MiniProject3/src
~/Desktop/MiniProject3/src base 15:08:28
./main ./player_random ../baseline/mac/baseline5_
```

Happy Coding!