



課程(一)Hello World !

Marius Suzuki



大綱

- C語言簡介
- IDE
- 編譯器
- 環境設定
- Hello, World!

C語言簡介



TIOBE: 程式語言流行排名

Jan 2025	Jan 2024	Change	Programming Language		Ratings	Change
1	1			Python	23.28%	+9.32%
2	3	▲		C++	10.29%	+0.33%
3	4	▲		Java	10.15%	+2.28%
4	2	▼		C	8.86%	-2.59%
5	5			C#	4.45%	-2.71%
6	6			JavaScript	4.20%	+1.43%
7	11	▲	<div>C語言家族 $10.29 + 10.15 + 8.86 + 4.45 = 38.16\%$</div>		2.61%	+1.24%
8	9	▲			2.41%	+0.95%
9	8	▼			2.37%	+0.77%
10	12	▲		Fortran	2.04%	+0.94%



C語言簡介

- C vs Python?
- 於1972首次出現
- 靈活度極高, 很考驗程式設計師的程度
- 仍是一個被廣泛使用的語言
- 特點: 低階記憶體存取(Low-level memory access)

IDE





IDE

IDE (整合開發環境)

- 編寫程式碼
 - 文字編輯器
 - 編輯 Source code (源代碼)
 - Syntax(語法) highlighter
 - 自動完成(Autocomplete)
 - 自動除錯(Debugging)



IDE





IDE

VS Code

Why VS Code?

- 其中一個最受歡迎的IDE
- 能夠配合大多數程式語言使用
- 可以安裝插件(Plugin)跟主題(Theme)





插件(Plugin)

What is a plugin?

- 增強/擴充程式的功能
- 常見例子: Chrome 插件

Why using plugins?

- 方便程式的使用
- 個人的使用偏好和習慣可以被滿足

編譯器(Compiler)

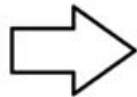


編譯器(Compiler)

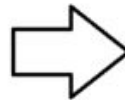
```
#include<stdio.h>

int main()
{
    printf("Hello, World!\n");
    return 0;
}
```

hello_world.c



Compiler



```
0110011000100010011000111
1100000001111111110000001
1111000110101010001100011
0011000100010011000111110
0000001111111110000001111
1000110101010001100011001
1000100010011000111110000
0001111111110000001111100
0110101010001100011001100
0100010011000111110000000
1111111110000001111100011
0101010001100011001100010
```

hello_world.o

編譯器(Compiler)



GCC

Any questions ???



環境設定





環境設定

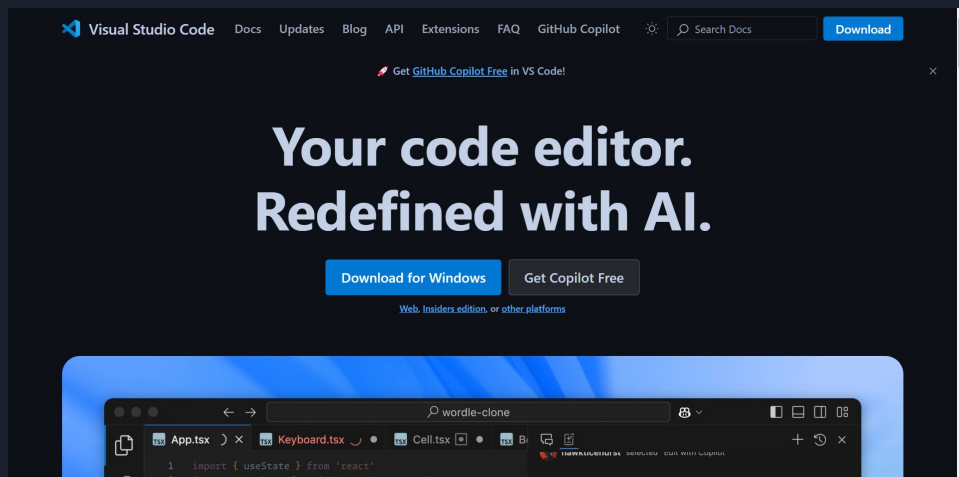
- 打開連結

<https://hackmd.io/@marius404/By6tmcOfkg>

Environmental Setup

Installation of VS Code

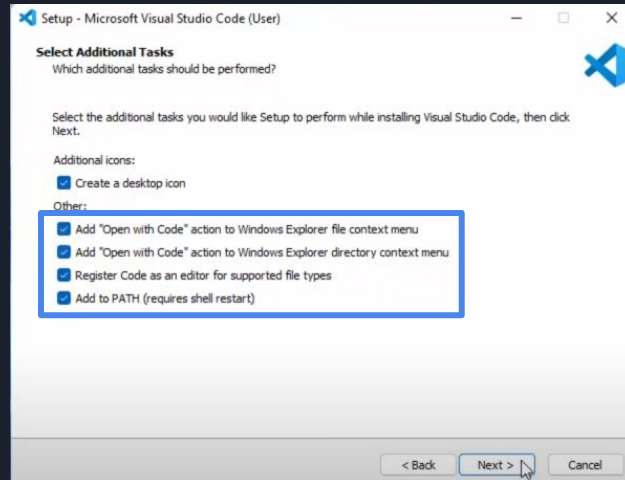
- 打開第一條連結
- 選擇適用於自己電腦OS的版本



Environmental Setup

Installation of VS Code

- 執行安裝檔
- 記得勾選以下選項

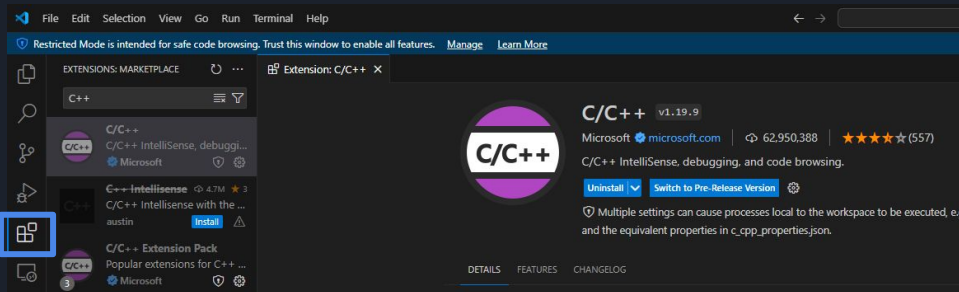


Environmental Setup

Installation of VS Code

- Install C/C++ Plugin
 - 選擇“Extension”
 - 輸入C/C++ 到搜尋欄
 - 按“Install”

“Extension” 按鈕



Environmental Setup

Installation of GCC compiler (Windows)

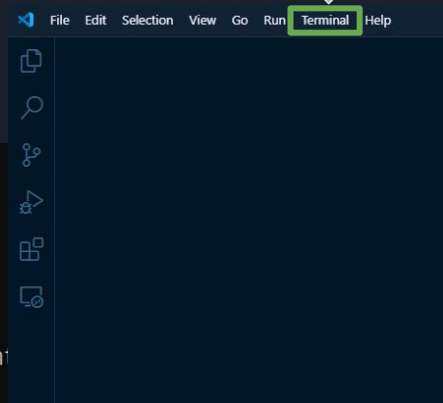
1. 在VS Code裡面打開Terminal (終端機)
2. 輸入 `Get-ExecutionPolicy` 之後按 Enter
3. 如果看到 `Restricted` 的話...
 - a. 輸入 `Set-ExecutionPolicy AllSigned` 然後按 Enter
 - b. 按 `Y` 然後按 Enter
 - c. Redo Step 2



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\WINDOWS\system32> Get-ExecutionPolicy
Restricted
PS C:\WINDOWS\system32> Set-ExecutionPolicy AllSigned
Execution Policy Change
The execution policy helps protect you from scripts that you do not trust. Changing the execution policy might
you to the security risks described in the about_Execution_Policies help topic at
https://go.microsoft.com/fwlink/?LinkID=135170. Do you want to change the execution policy?
[Y] Yes [A] Yes to All [N] No [L] No to All [S] Suspend [?] Help (default is "N"): Y
```



Environmental Setup

Installation of GCC compiler (Windows)

1. Copy 文字檔裡面 `Set-ExecutionPolicy Bypass` 開頭的指令到Terminal 裡面然後按 Enter
2. Wait for it...

```
PS C:\WINDOWS\system32> Set-ExecutionPolicy Bypass -Scope Process -Force; [System.Net.ServicePointManager]::SecurityProtocol = [System.Net.ServicePointManager]::SecurityProtocol -bor 3072; iex ((New-Object System.Net.WebClient).DownloadString('https://community.chocolatey.org/install.ps1'))
Forcing web requests to allow TLS v1.2 (Required for requests to Chocolatey.org)
Getting latest version of the Chocolatey package for download.
Not using proxy.
Getting Chocolatey from https://community.chocolatey.org/api/v2/package/chocolatey/2.4.1.
Downloading https://community.chocolatey.org/api/v2/package/chocolatey/2.4.1 to C:\Users\hinlu\AppData\Local\Temp\chocolatey\chocoInstall\chocolatey.zip
Not using proxy.
Extracting C:\Users\hinlu\AppData\Local\Temp\chocolatey\chocoInstall\chocolatey.zip to C:\Users\hinlu\AppData\Local\Temp\chocolatey\chocoInstall
Installing Chocolatey on the local machine
Creating ChocolateyInstall as an environment variable (targeting 'Machine')
Setting ChocolateyInstall to 'C:\ProgramData\chocolatey'
WARNING: It's very likely you will need to close and reopen your shell
before you can use choco.
```

Environmental Setup

Installation of GCC compiler (Windows)

1. 輸入 `choco info mingw` 然後按Enter
2. 如果Terminal出現下面這個Package(套件)的話
 - a. 輸入 `choco install mingw` 然後按Enter
 - b. Wait for it...

```
PS C:\WINDOWS\system32> choco info mingw
Chocolatey v2.4.1
mingw 13.2.0 [Approved] Downloads cached for licensed users
Title: MinGW-w64 | Published: 10/7/2023
Package approved by Windos on Nov 30 2024 17:25:20.
Package testing status: Passing on Oct 07 2023 19:13:06.
Number of Downloads: 1394092 | Downloads for this version: 321149
Package url https://community.chocolatey.org/packages/mingw/13.2.0
Chocolatey Package Source: https://bitbucket.org/xoviat/chocolatey-packages
Package Checksum: 'ORgZ1jPrbuQdltDAA03tzkJuP8WhA9+E4uQLCltrsTbpJ/UV79rtIBtebjLusBcCMG7F1tm0GMIjw0eXrtVQvg==' (SHA512)
Tags: compiler gcc mingw mingw-w64
Software Site: http://mingw-w64.org/
Software License: n/a
Summary: GCC for Windows 64 & 32 bits.
Description: Mingw-w64 is an advancement of the original mingw.org project, created to support the GCC compiler on Windows systems. It has forked it in 2007 in order to provide support for 64 bits and new APIs. It has since then gained widespread use and distribution.

1 packages found.
```



Environmental Setup

Installation of GCC compiler (Windows)



Check whether GCC is properly installed

1. 輸入 `gcc --version` 然後按Enter
2. 如果以下的字串有出現在Terminal的話就安裝完成了

```
PS C:\WINDOWS\system32> gcc --version
gcc.exe (Rev3, Built by MSYS2 project) 13.2.0
Copyright (C) 2023 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.
```

Environmental Setup

Installation of GCC compiler (Mac)



- 打開“Terminal”
- 輸入 `xcode-select --install` 然後按Enter
- 輸入 `g++ -v` OR `clang++ -v` 檢查是否有正確安裝

```
Apple logo icon ~  
> g++ -v  
Apple clang version 15.0.0 (clang-1500.3.9.4)  
Target: arm64-apple-darwin23.4.0  
Thread model: posix  
InstalledDir: /Library/Developer/CommandLineTools/usr/bin
```

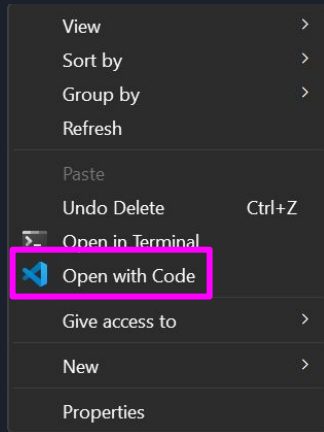

All the magic starts here
— “hello world”



All the magic starts here

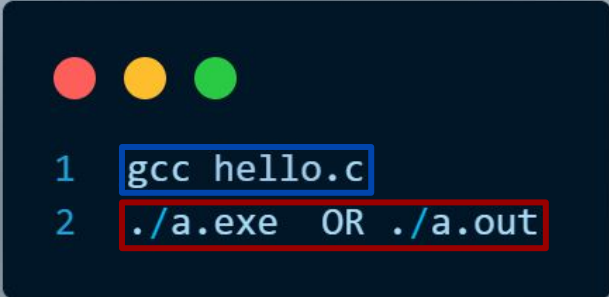
- Prerequisites

- 在檔案總管打開桌面的路徑
- 按右鍵以VS Code打開
- 創建一個新的檔案並命名為 `hello.c`



All the magic starts here

- Compiling your code



A terminal window with a dark blue background and three colored window control buttons (red, yellow, green) at the top left. It contains two lines of text: line 1 is 'gcc hello.c' and line 2 is './a.exe OR ./a.out'. The first line is enclosed in a blue rectangular box, and the second line is enclosed in a red rectangular box.

```
1 gcc hello.c
2 ./a.exe OR ./a.out
```

`gcc`: 編譯源代碼

`./xxx`: 執行Compiler所產生的執行檔

All the magic starts here

- Code explanation

```
1 #include <stdio.h>
2
3 int main(){
4     printf("Hello nccucs\n");
5     return 0;
6 }
```

Library (函式庫):

- 像一個工具箱
- 內有不同的Function(函式)
 - (e.g. printf from stdio.h)
- 增強開發效率

All the magic starts here

- Code explanation




```
1  #include <stdio.h>
2
3  int main(){ ←
4      printf("Hello nccucs\n");
5      return 0;
6  }
```

Main function(主程式)

- 程式的入口處

All the magic starts here

- Code explanation




```
1  #include <stdio.h>
2
3  int main(){
4      printf("Hello nccucs\n");
5      return 0;
6  }
```

Basic Syntax (語法)

- 每一個指令的最後必須加上分號

All the magic starts here

- Code explanation



```
1 #include <stdio.h>
2
3 int main(){
4     printf"Hello nccucs\\n";
5     return 0;
6 }
```

Basic Syntax (語法)

- 在使用printf()輸出文字(字串)的時候必須加上雙引號

All the magic starts here

- Code explanation



```
1 #include <stdio.h>
2
3 int main(){
4     printf("Hello nccucs\n");
5     return 0;
6 }
```

printf() function:

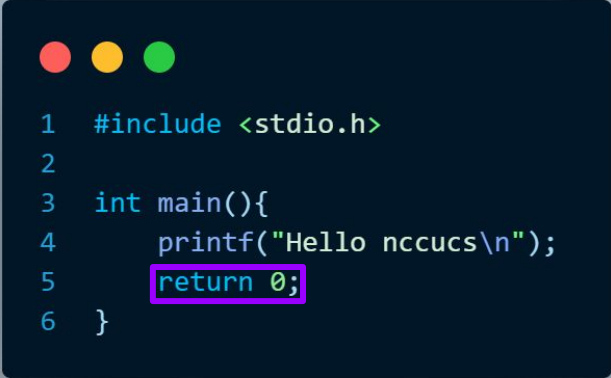
- Output function (輸出函式)
- 輸出文字到Terminal(終端機)

\n:

- 用於跳行

All the magic starts here

- Code explanation

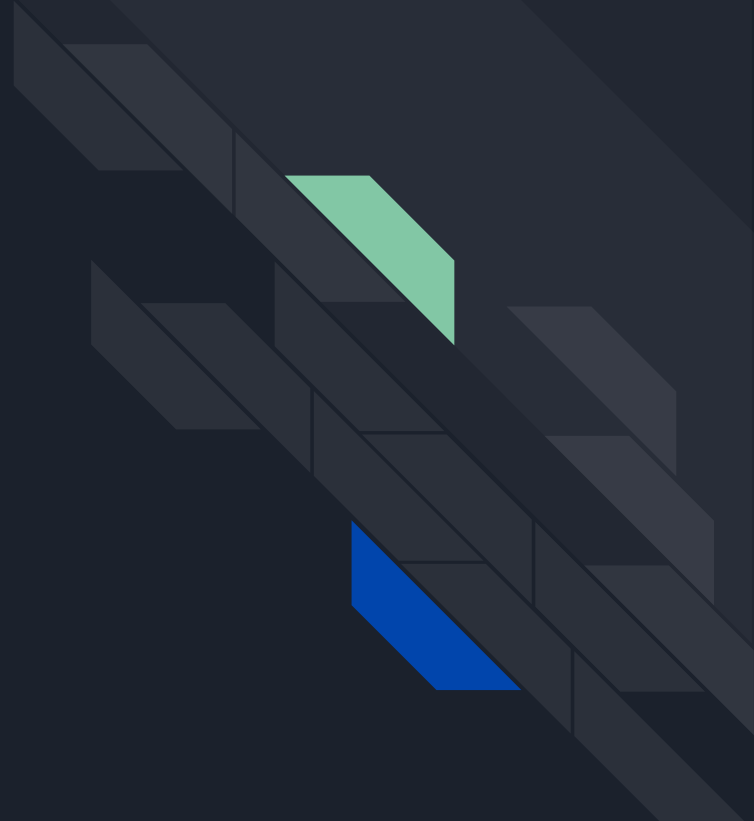


```
1  #include <stdio.h>
2
3  int main(){
4      printf("Hello nccucs\n");
5      return 0;
6  }
```

Return value (回傳值):

- 程式執行結束後所回傳的數值
- 0代表成功執行, 如果不是0代表程式有問題 (Bug)

Any questions ???



END

