

計算機程式設計

Computer Programming

Course Introduction

Instructor: 林英嘉

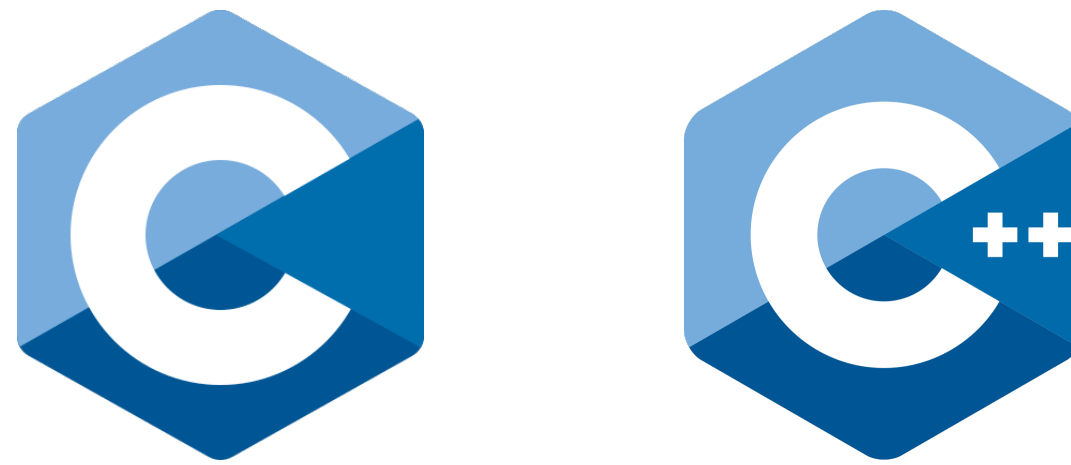
2024/09/09

自我介紹

- 林英嘉
- 2024/02 國立成功大學資訊工程學系博士畢業
- 現職：國立清華大學博士後研究員
 - 專長：自然語言處理

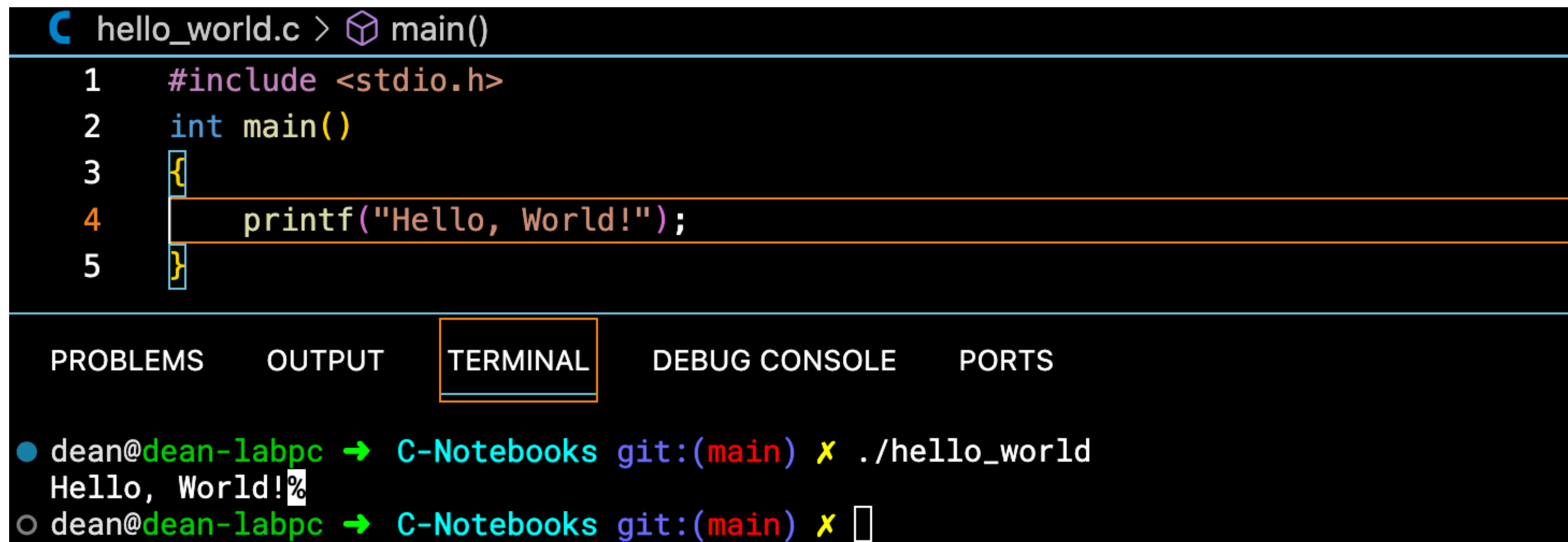
這門課教什麼？

- 程式設計
- C/C++



什麼是程式設計？

```
1 #include <stdio.h>
2 int main()
3 {
4     printf("Hello, World!");
5 }
```



The screenshot shows a code editor with a dark theme. At the top, a tab is labeled 'hello_world.c > main()'. The code is the same as shown in the previous block. Below the code, there are tabs for 'PROBLEMS', 'OUTPUT', 'TERMINAL', 'DEBUG CONSOLE', and 'PORTS'. The 'TERMINAL' tab is selected and highlighted with an orange border. The terminal shows the command 'dean@dean-labpc → C-Notebooks git:(main) X ./hello_world' and the output 'Hello, World!%'. Below this, another terminal entry shows 'dean@dean-labpc → C-Notebooks git:(main) X' followed by a cursor.

```
C hello_world.c > main()
1  #include <stdio.h>
2  int main()
3  {
4      printf("Hello, World!");
5  }

PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE  PORTS

● dean@dean-labpc → C-Notebooks git:(main) X ./hello_world
Hello, World!%
○ dean@dean-labpc → C-Notebooks git:(main) X █
```

為什麼我們還要學程式設計？

- 大型語言模型 (Large Language Model, LLM) 雖然很強，但會出錯
 - 只有你比他更強，你才能除錯
- 面試官通常不會讓你使用 LLM 或是 Copilot
 - 本身的實力決定是否取得offer

為什麼我們要學C，不學Python?

- 使用場景：
 - C語言的使用場景非常多，包含韌體、嵌入式系統、作業系統等...
 - Python多用於資料分析、機器學習

為什麼我們要學C，不學Python?

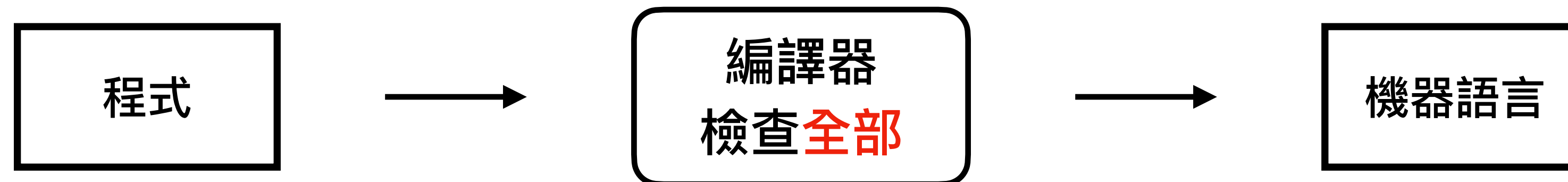
- 執行速度：
 - Python是直譯式語言 (Interpreted language)
 - C是編譯式語言 (Compiled language) — faster

直譯式 vs. 編譯式

直譯式語言 (e.g., Python)



編譯式語言 (e.g., C)



為什麼我們要學C，不學Python?

- 程式語言邏輯：
 - C的程式碼比Python包含更多計算機細節

C與Python簡單例子

C

```
#include <stdio.h>
int main() {
    int a = 10;           // 宣告變數a並賦值
    int *p = &a;          // 宣告指標p並將指標p指向a的記憶體地址
    printf("a的值: %d\n", a);
    printf("a的記憶體地址: %p\n", p);
    printf("通過指標p訪問a的值: %d\n", *p);
}
```

Python

```
a = 10
print("a的值:", a)
```

為什麼我們要學C，不學Python?

- 程式語言邏輯：
 - C的程式碼比Python包含更多計算機細節
 - 學習C對資工系的學生是很好的訓練
 - 記憶體管理

上課方式

- 原則上每週授課2小時+1小時的上機隨堂小考（從下禮拜開始）
 - 最晚截止繳交時間為當天16:00之前
 - 我跟助教都會在隨堂考期間回答各位的問題
- 作業5次
- 期中考2次、期末考1次

配分比重

- 平時隨堂考3% * 15次
- 作業5% * 5次
- 期中考10%、期末考20%

隨堂考與作業內容

- 原則上為撰寫程式
 - 隨堂考難度較低
 - 作業難度較高

作業繳交的內容

- 程式碼 (.c檔案)
- 說明檔 (Word或Markdown)

期中/期末考內容

- 上機考 (maybe) + 紙本 (為主)
- 不可開書、不可Google、不可使用生成式AI

上機考不可以使用 VSCode / Cursor 等編輯器



Copilot



CURSOR



←

→

C-course-materials [SSH: LABPC]

1

test.c 1, U

test.c

1

#include <stdio.h>

2

int gcd(int a, int b) {

if (b == 0) return a;

return gcd(b, a % b);

}

24

PROBLEMS 1

OUTPUT

TERMINAL

DEBUG CONSOLE

PORTS

zsh

+

↓

🗑

...

^

×

dean@dean-labpc → C-course-materials git:(main) ✕

SSH: LABPC

main*

↺

⊗ 1

⚠ 0

🔊 0

Ln 2, Col 8

Spaces: 4

UTF-8

LF

{ } C

Linux

教材

- 不用買書，簡報檔都會放在
 - <https://github.com/mcps5601/C-course-materials>
- 參考書籍：
 - C語言教學手冊第四版，作者：洪維恩，出版社：旗標。
 - C Programming: A Modern Approach, 2nd Edition. Author: K. N. King. Published by the W. W. Norton & Company.

生成式AI使用原則

- 隨堂小考和作業可使用
- 請註明AI生成段落
- 使用建議：
 - 請幫我寫出判斷最大公因式的函數 (X)
 - 請幫我檢查我寫的最大公因式函數哪裡錯誤 (O)
 - 請告訴我C語言 void 代表什麼意思？(O)

生成式AI使用原則小結

	隨堂考	作業	期中/期末考
生成式AI	O	O	X
Copilot	X	-	X
生成式AI使用規範	須於程式碼中加入註解說明AI生成段落	須於程式碼註解與報告中詳細說明AI生成段落	X
參考他人code	X (不可討論)	X (可口頭討論)	X (不可討論)

Syllabus (NCHU)

- https://onepiece.nchu.edu.tw/cofsys/plsql/Syllabus_main?v_strm=1131&v_class_nbr=1677

Syllabus

	Exam
	HW deadline

Week	Topic	Note
1	Course Rules and C Basics	
2	Data Types and Operators	
3	Program Control (1): if-else, switch	HW1
4	Program Control (2): for, while, do-while	
5	Midterm Exam (1)	
6	Arrays: Unidirectional Arrays, Multidirectional Arrays	
7	Functions (1): Introduction to Functions, Scope, Visibility	HW2
8	Functions (2): Recursions, Applications of Recursion	
9	Pointers (1): Introduction to Pointers	
10	Pointers (2): Pointers and Arrays	HW3
11	Midterm Exam (2)	
12	Strings (1): Introduction to Strings in C	
13	Strings (2): Pointers and Strings	HW4
14	Structures: struct, typedef	
15	Memory Management	
16	Final Exam	HW5
17	Self-Learning: File Input / Output (I/O)	
18	Self-Learning: From C to C++	

如果你有問題

- 歡迎來信問問題或是約時間！
 - yingjia.lin.public@gmail.com

課程目標

- 充分掌握C語言基礎
- 認識基本計算機程式概念，為未來的課程打好基礎