Chapter 2.
Getting Start –
Program Structure, Printing, and Comments

I-Fen Chao
Assistant Professor

上傳檔案：

1. **WORD report**

   (1). 請用portal上的WORD report來改

   (2). 請delete report上面的圖檔，貼上自己的圖檔

   (3). 回答問題，請放最後

2. **1.cpp, 2.cpp…** 等程式檔案 **(請用題號標明程式)**

3. 如果多個程式，務必每個程式開一個**project(專案)**

流程圖

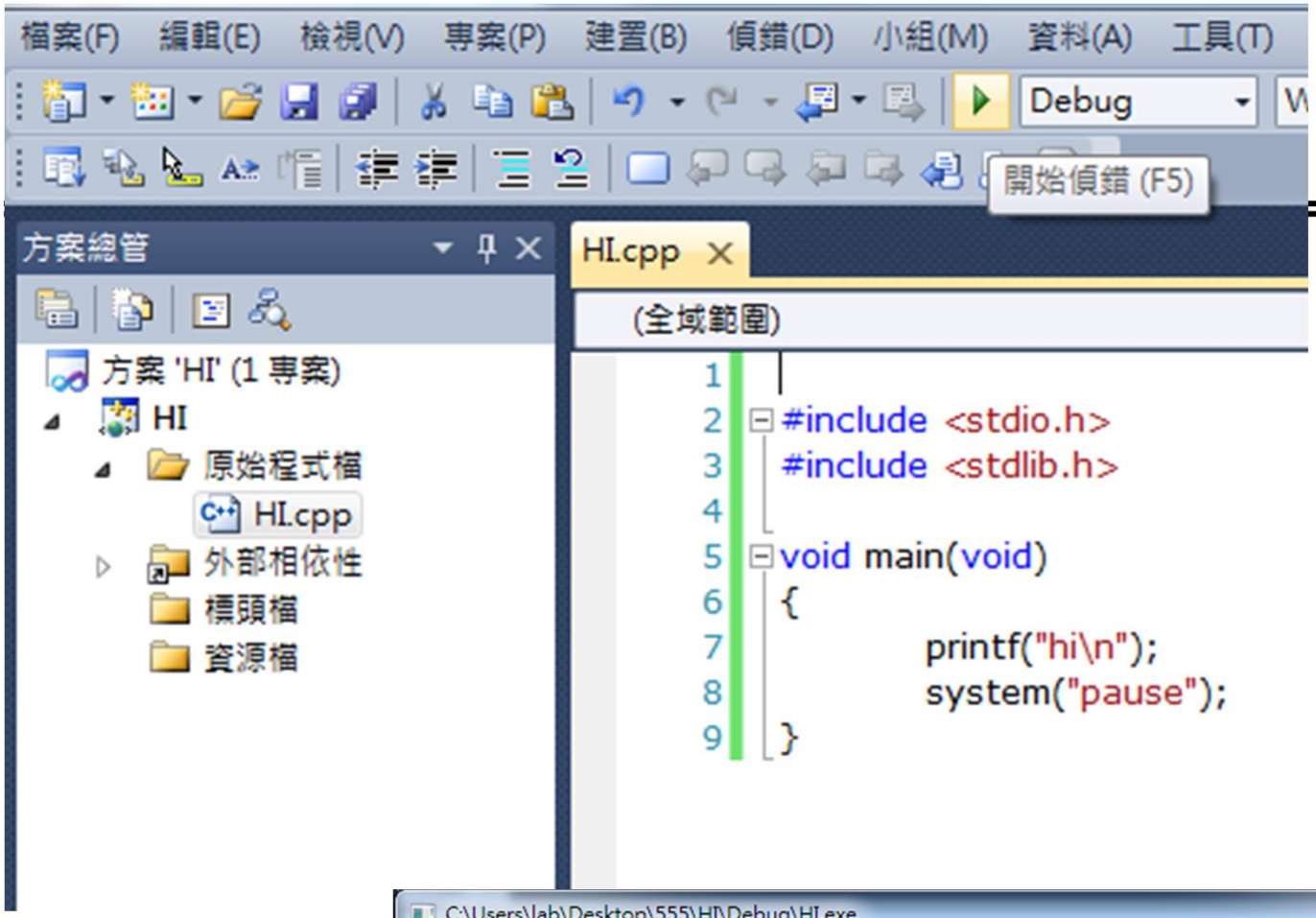
- 不須程式全部細節
- 必須畫橢圓，方形等形狀
- 描述完成的順序(可以用中文描述)

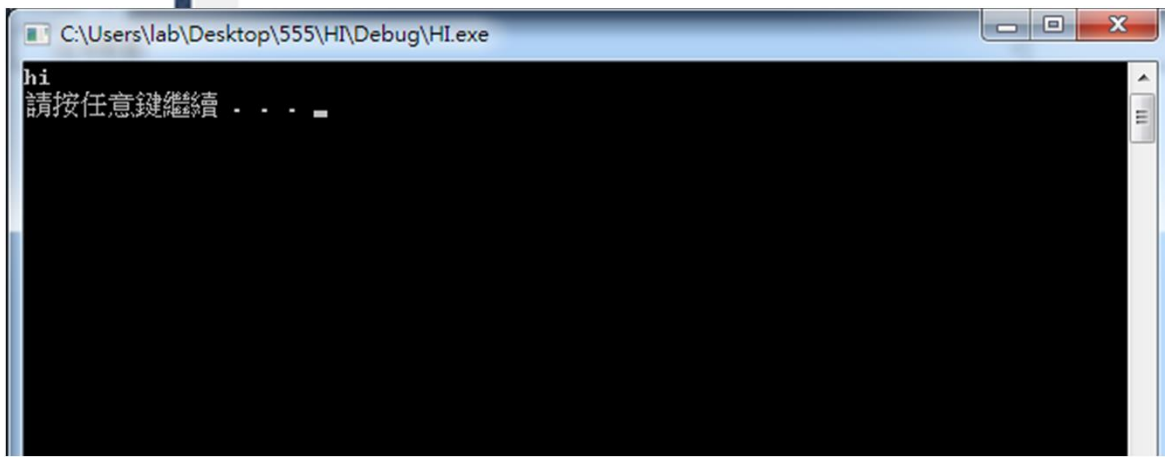
- 沒有標準答案
- 主要幫助同學了解寫程式的流程順序
- 請寫在筆記本
- 期末收回更改分數
- 請寫整齊，少寫或太亂，助教會給低分(但不會仔細看流程圖內容)


最新消息
教材
作業
學習討論
成績
學生/助教
出缺勤
課程內容
翻轉教室
1071學期EEA129 B班
◆作業清單 ◆新增 ◆作業討論 ◆
作業清單
1. 課堂project
NO
進度
作業名稱/內容
附檔
上傳數
截止日
成績公告
自由繳交
刪除
編輯
1
C Basic Programming
Class project1: practice printf, int
upload .cpp and WORD report
檔案
137
2018/9/17
立即
N
刪除
編輯
2
C Basic Programming
Homework1: practice printf
upload .cpp and WORD report
檔案
16
2018/9/23
立即
N
刪除
編輯
2. 回家homework

Portal上傳後
1. IP　　2. 時間

超時再Portal上傳，(即使是1分鐘)，
portal都會顯示紅色時間。


檔案(F) 編輯(E) 檢視(V) 專案(P) 建置(B) 偵錯(D) 小組(M) 資料(A) 工具(T)
Debug
開始偵錯 (F5)
方案總管
方案 'HI' (1 專案)
HI
原始程式檔
HI.cpp
外部相依性
標頭檔
資源檔
HI.cpp
(全域範圍)
#include <stdio.h>
#include <stdlib.h>
void main(void)
{
printf("hi\n");
system("pause");
}


C:\Users\lab\Desktop\555\HI\Debug\HI.exe
hi
請按任意鍵繼續 . . .

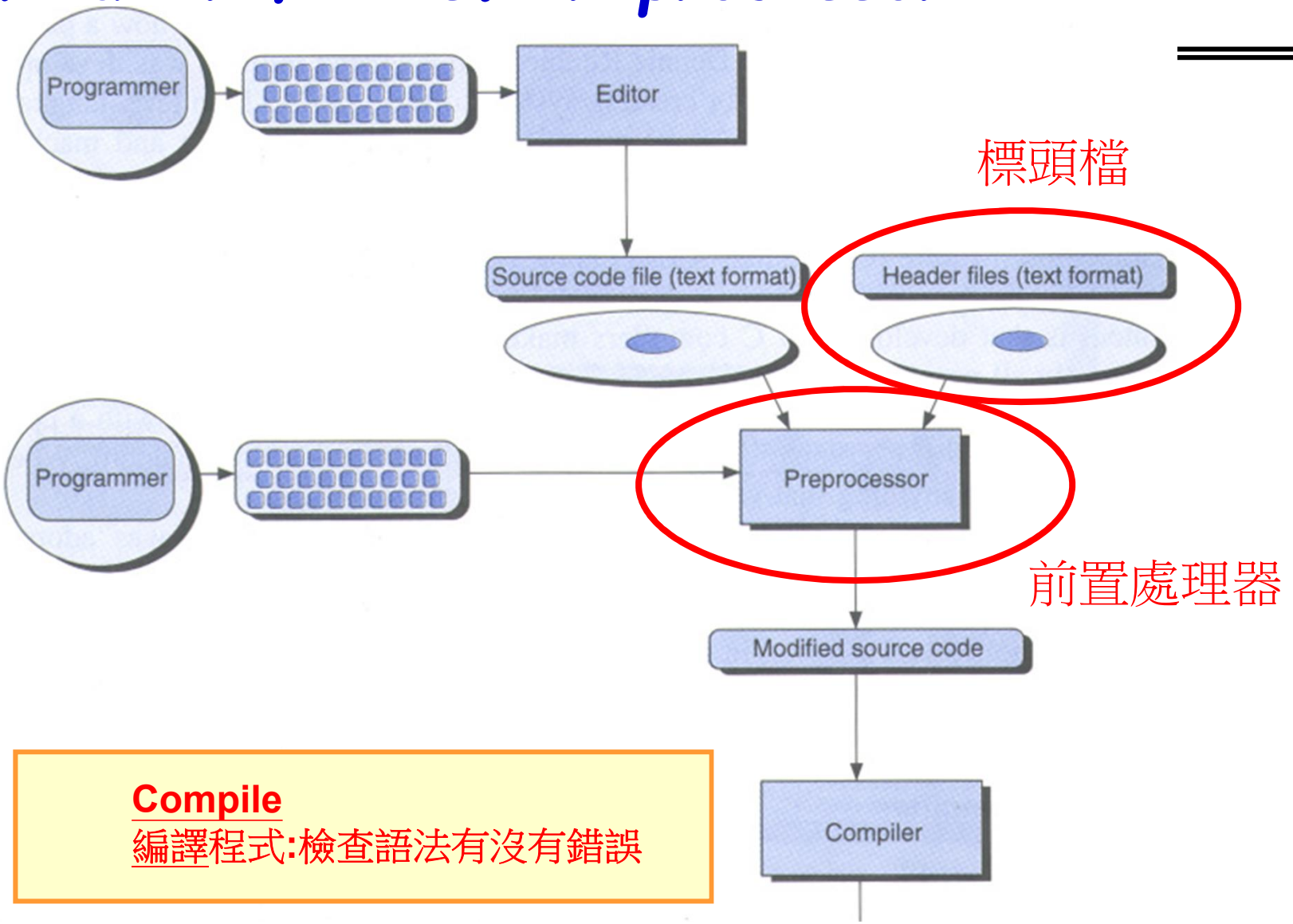
What is "Header file"

Source code

```
#include <stdio.h>
void main(void)
{
    printf("Hello World!!");
}
```

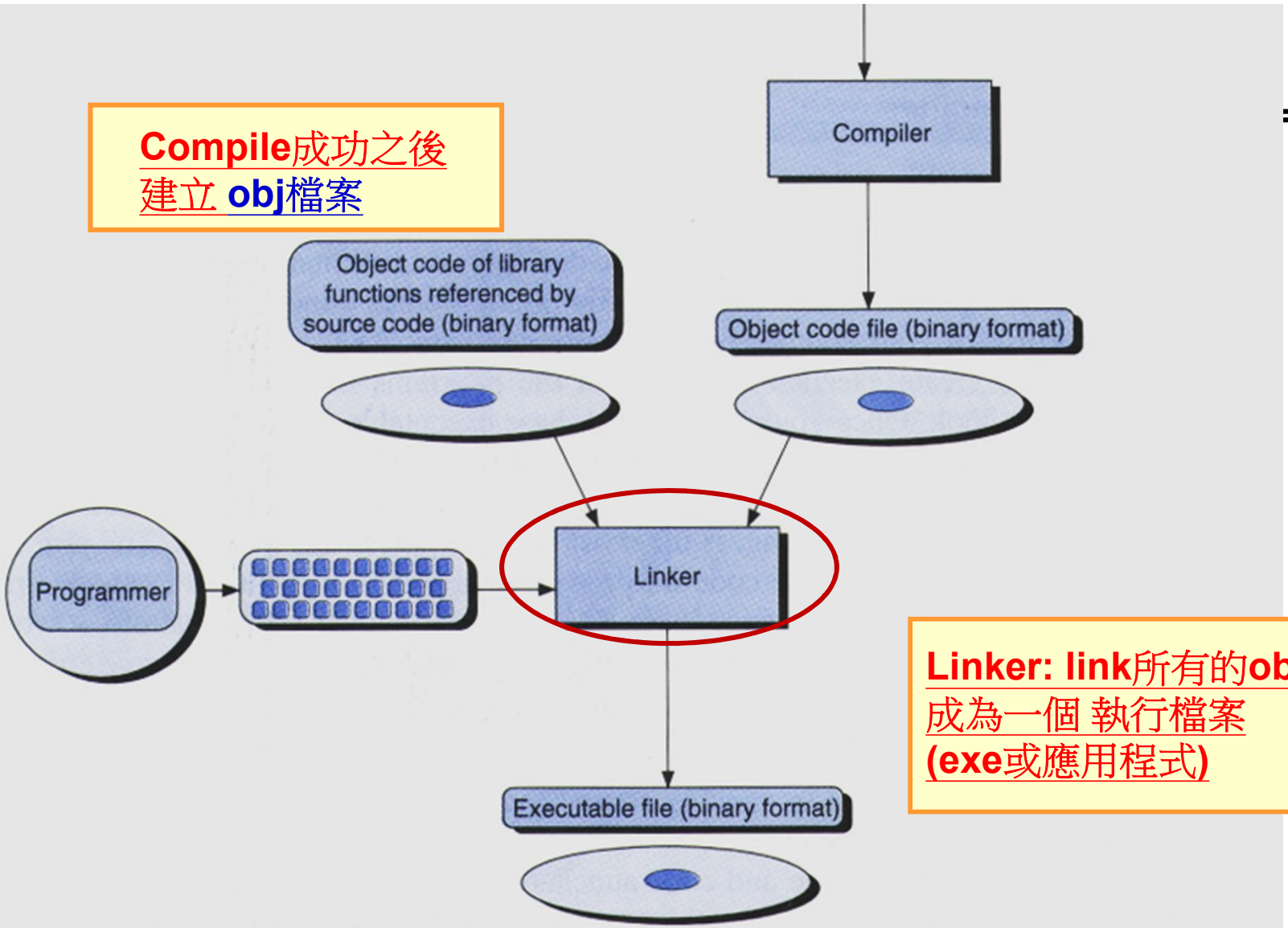
- What dose #include <stdio.h> mean?
  - header files (標頭檔)
  - Preprocessor (前置處理器)

Header file vs. Preprocessor



Compile
編譯程式:檢查語法有沒有錯誤

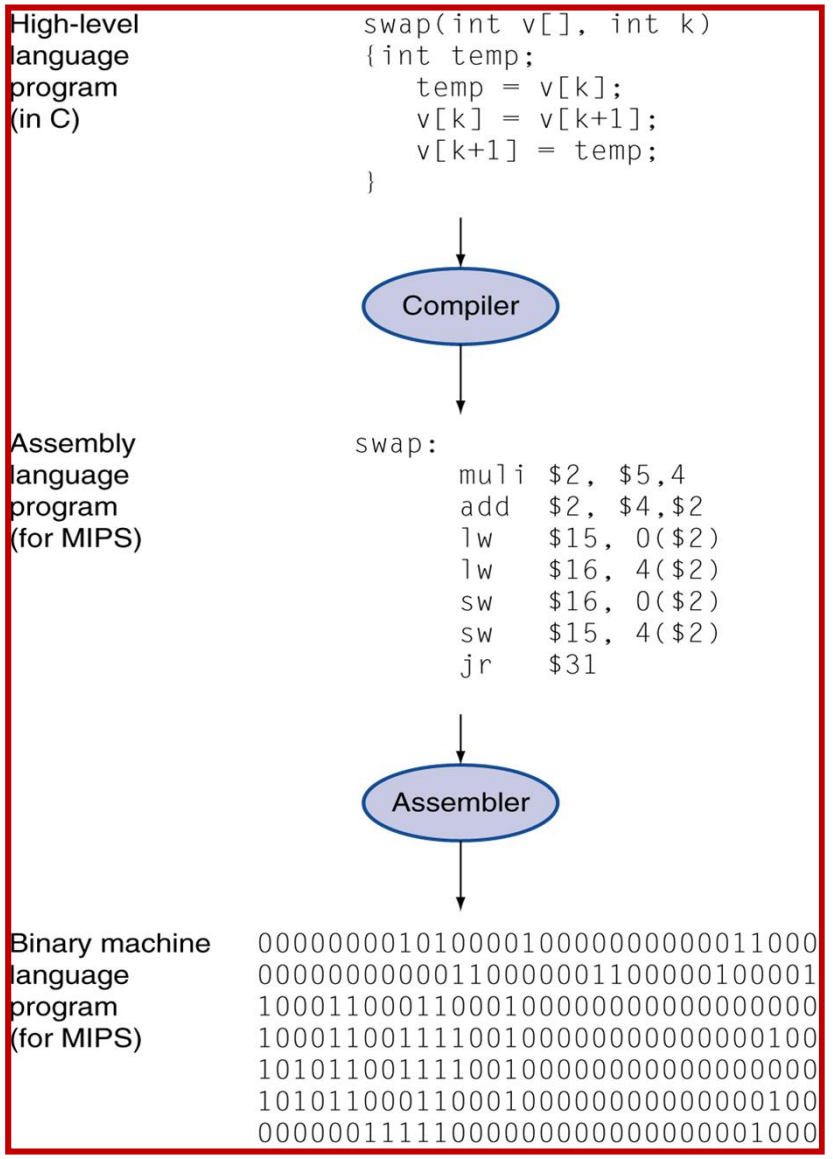
Compile成功之後
建立 obj檔案



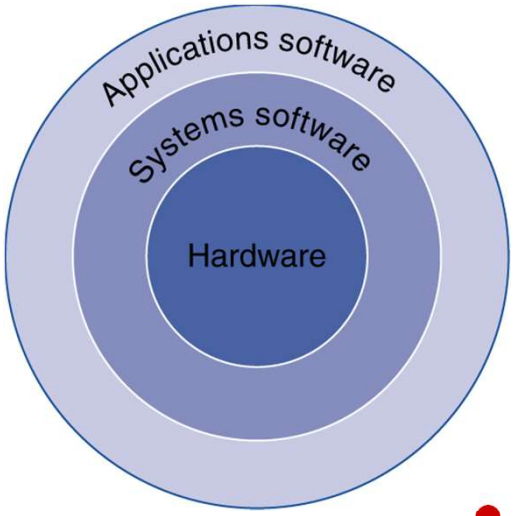
Linker: link所有的obj檔案
成為一個 執行檔案
(exe或應用程式)

Levels of Program Code

- High-level language
  - Level of abstraction closer to problem domain
  - Provides for productivity and portability
- Assembly language
  - Textual representation of instructions
- Hardware representation
  - Binary digits (bits)
  - Encoded instructions and data



Below Your Program



- Application software
  - Written in high-level language
- System software
  - Compiler: translates HLL code to machine code
  - Operating System: service code
    - + Handling input/output
    - + Managing memory and storage
    - + Scheduling tasks & sharing resources
- Hardware
  - Processor, memory, I/O controllers

Header file

- You write program codes
- C library functions
  - e.g. printf, in “stdio.h”

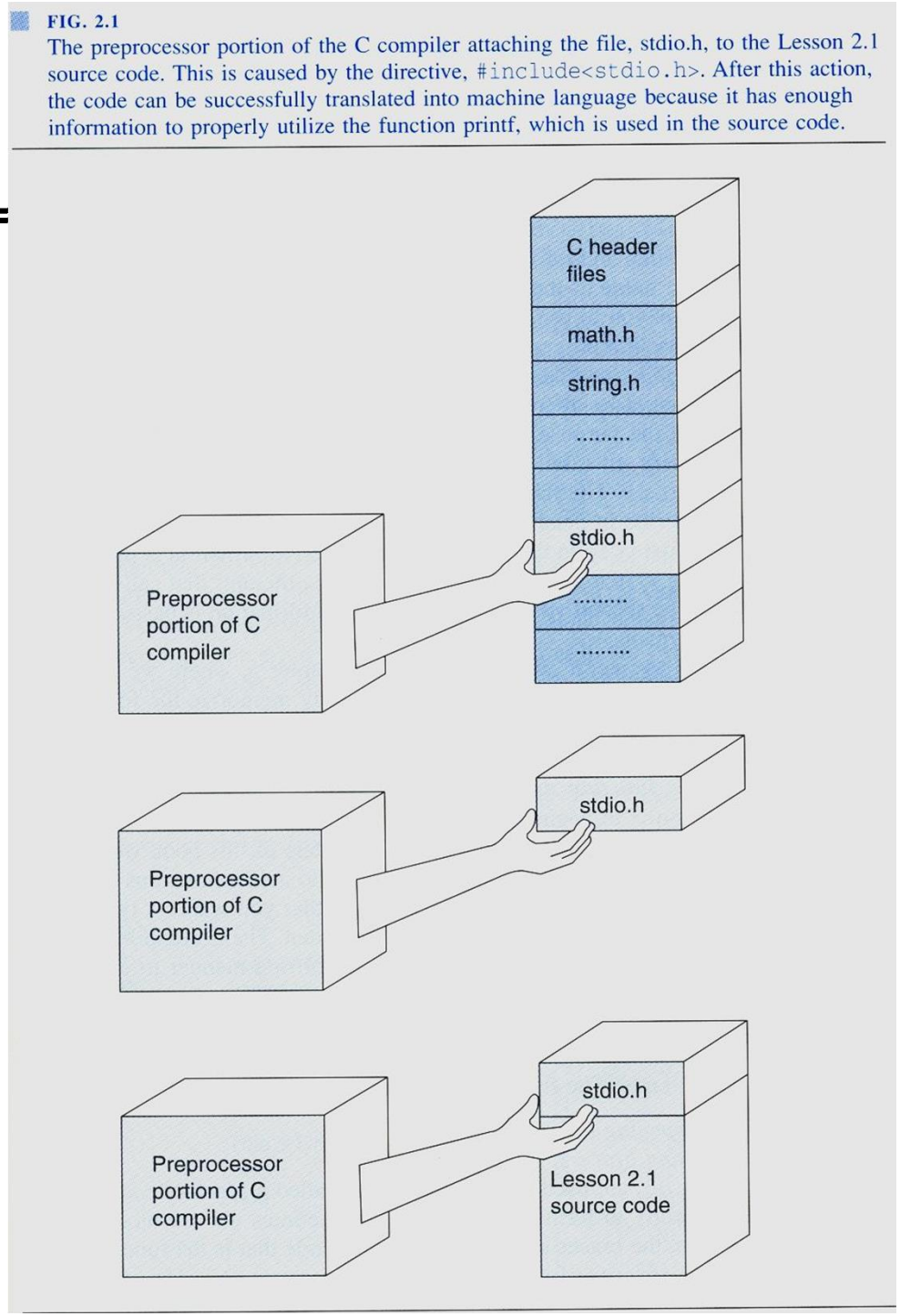
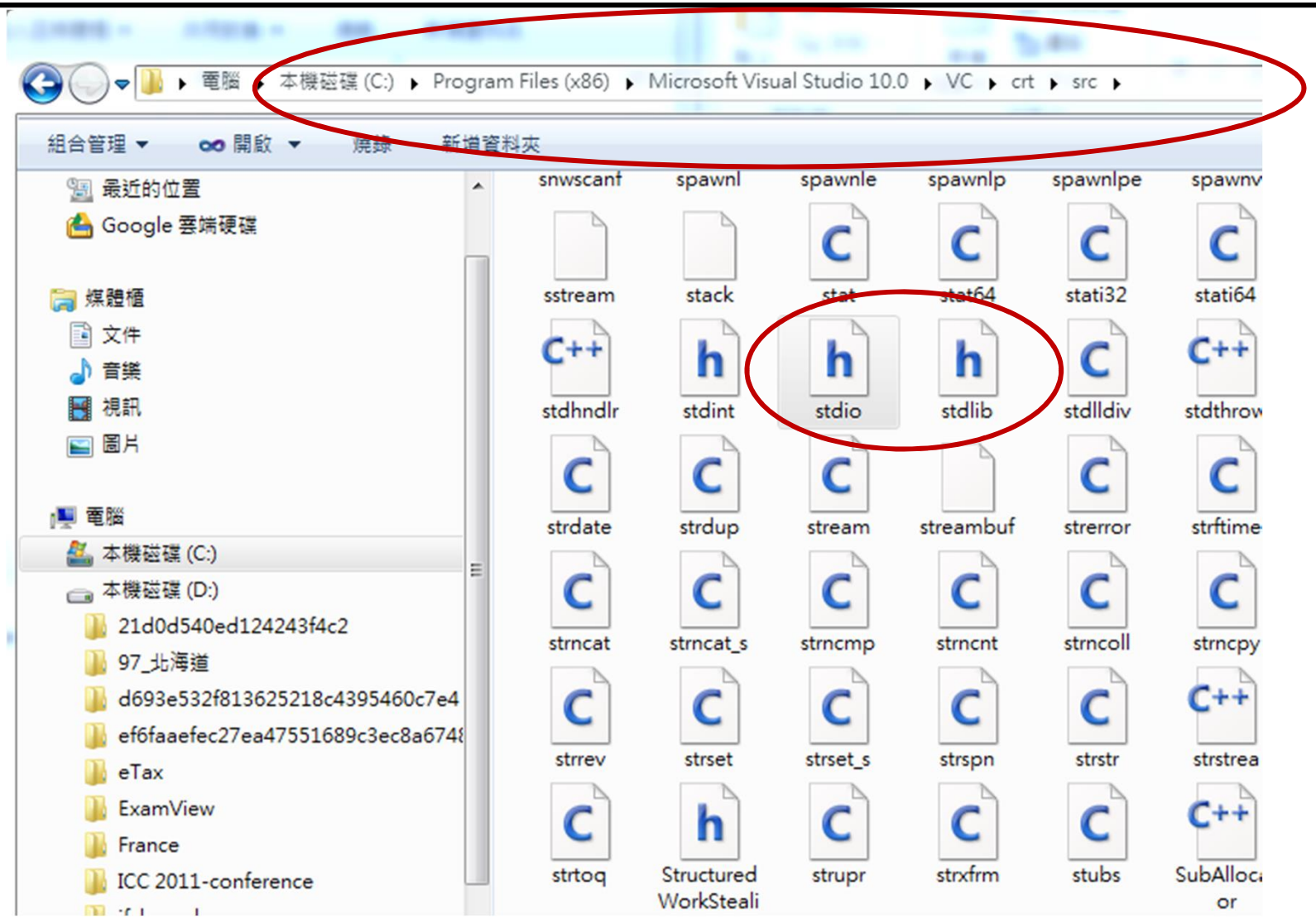


FIG. 2.1
The preprocessor portion of the C compiler attaching the file, stdio.h, to the Lesson 2.1 source code. This is caused by the directive, `#include<stdio.h>`. After this action, the code can be successfully translated into machine language because it has enough information to properly utilize the function printf, which is used in the source code.

Header files



main program

Source code

```
#include <stdio.h>
void main(void)
{
    printf("Hello World!!");
}
```

- What dose `void main(void)` mean?

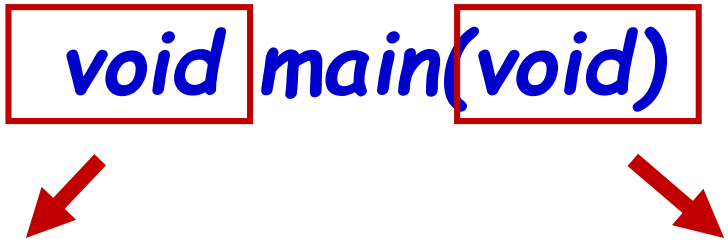
➔ function (函式)

➔ main fuction

➔ void

void main(void)

- The first and self-defined function
- The execution entrance point (The C program starts here)



No return value !

沒有傳回值

No parameter !

沒有傳入參數

What dose { and } mean?

Source code

```
#include <stdio.h>
void main(void)
{
    printf("Hello World!!");
}
```

- What dose { and } mean?

➔ Opening brace {

➔ Closing brace }

➔ A block of code {…}

printf

Source code

```
#include <stdio.h>
void main(void)
{
    printf("Hello World!!");
}
```

- What dose `printf("Hello World");` mean?

➔ Statements in C

➔ A function of the C stdanard I/O library

➔ End with ;

The form of a function call

Source code

```
#include <stdio.h>
void main(void)
{
    printf ("Hello World!!");
}
```

函式名稱

Function name

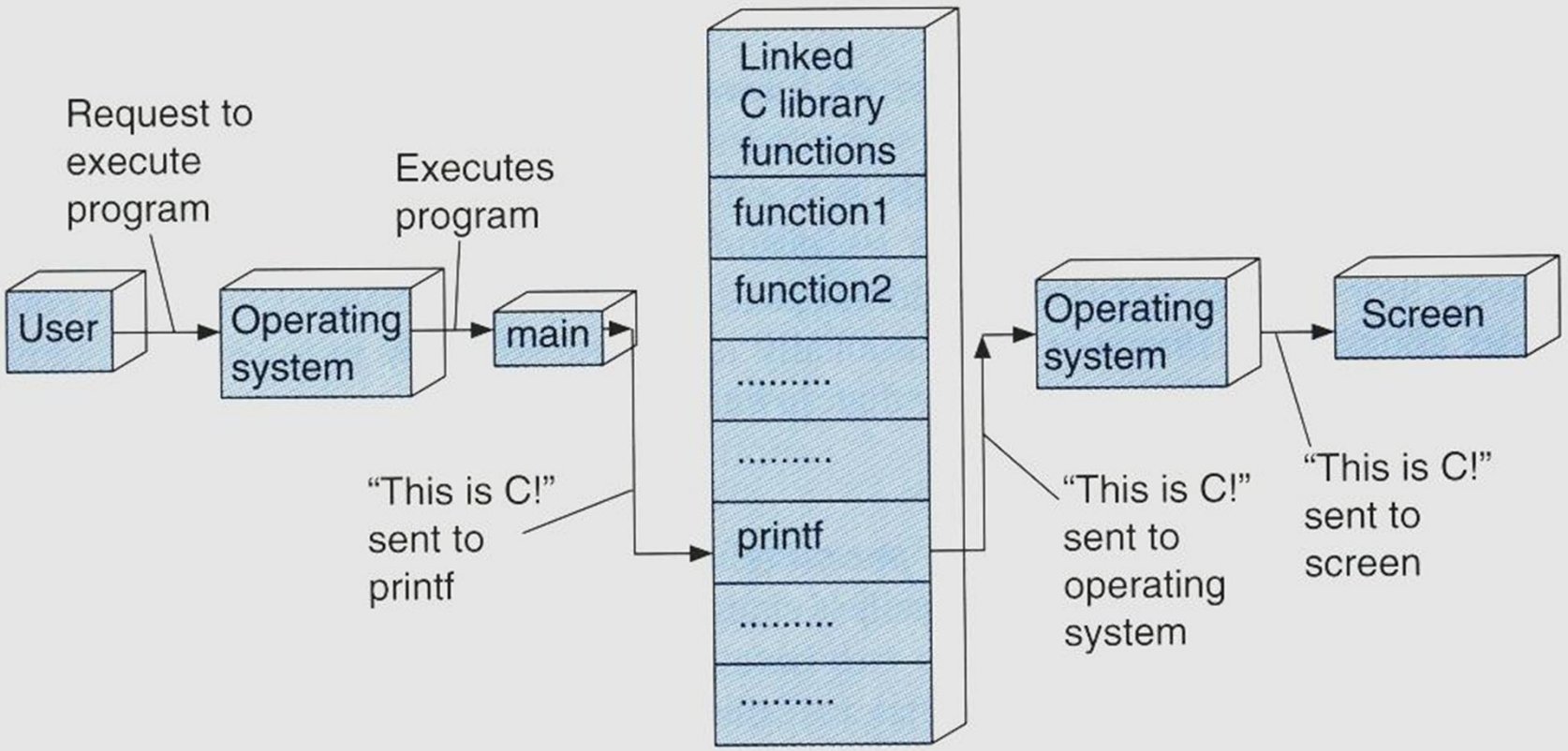
傳入的參數

Information to send

Link - printf function

=

FIG. 2.2
Calling the printf function from the C library. For this lesson's program, only the printf function is directly linked to main.



The look of a program

Source code

```
#include <stdio.h>
void main(void)
{
    printf ("Hello World!!");
}
```

Source code

```
#include <stdio.h>
void main(void)
{printf ("Hello World!!");}
```

Writing Comments

- The structure of comments
  - /* …… */
- The location of comments
  - Everywhere
  - No nested comment
- Write valuable comments
  - Documentation, readable codes, maintenance

```
/* Wrong comment 1, no end asterisk and slash
/* Wrong comment 2, no end slash *
/ *Wrong comment 3, there is a blank between /and * */
```

Source code

```
#include <stdio.h>
void main(void)
{
    printf("Hello, How are you?");
    printf("Fine, thank you.");
}
```

Output

```
Hello, How are you?
Fine, thank you.
```

Source code

```
#include <stdio.h>
void main(void)
{
    printf("Hello, How are you?\n");
    printf("Fine, thank you.");
}
```

- **What does \n mean?**

➔ **Linefeed symbol**

Variable

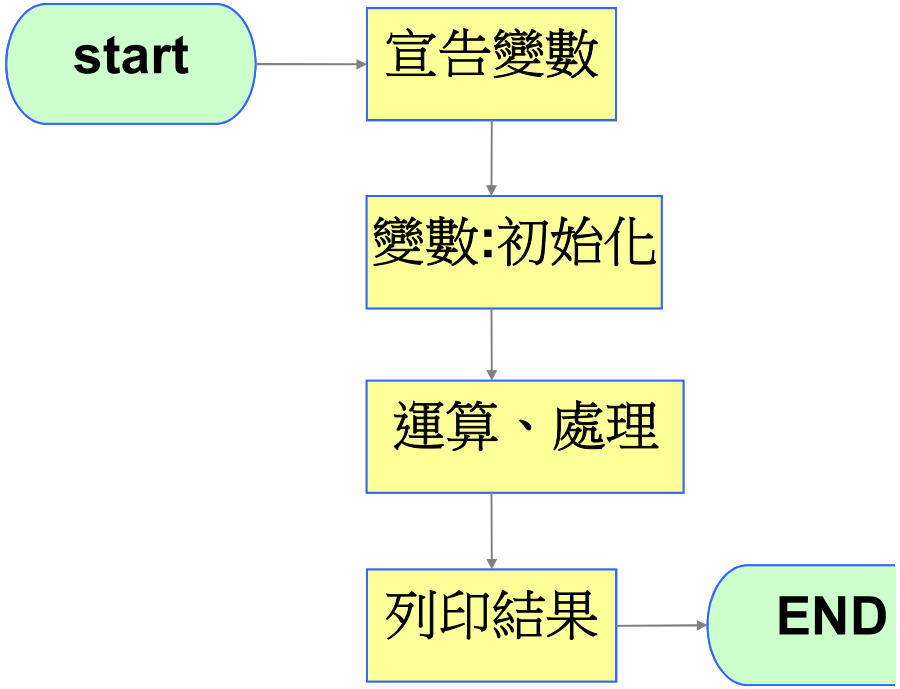
- **Naming**: 命名 **Declaring**: 宣告
- **Assigning**: 指定 **Printing** Values

main.cpp

(全域範圍)

```
#include <stdio.h>
#include <stdlib.h>

void main(void)
{
1   //declare variable
    int a;

2   //assignment
    a=2;
3   //output
    printf("a = %d\n", a);

    //assignment
    a=10;
    //output
    printf("a = %d\n", a);

    system("pause");
}
```

```
D:\program\tt1\Debug\tt1.exe
a = 2
a = 10
請按任意鍵繼續 . . .
```



這樣會有甚麼現象?

```
//output
printf("a = %d\n"    );
```

Important features of a C program (1)

- The main function name must be main
- The program body must start from {
- The program body must end with }
- A C statement must end with ;
- A C statement is case sensitive

Important features of a C program (2)

- You can add blank(s) between tokens in a C statement but not acceptable to add blank(s) within a token
- Uses character escape sequences that consist of a backslash followed by other character(s) in string literals to represent special characters and actions
- Make your comments stand out. Do not hide them. Adopt a style that is neat and orderly. Set aside time for writing comments in your programs

Character escape sequences (1)

\0	Null character	字串(string)的結束字元
\a	Alert/bell	產生一個音效或視覺上的警示
\b	Backspace	將目前游標位置行往前移一格
\f	Form feed	將目前游標位置移動至新一頁的開始位置(如: 印表機退出一頁)
\n	New line	移動至下一行的初始位置
\r	Carriage return	將目前游標位置移往本行第一個位置

Character escape sequences (2)

\t	Horizontal tab	水平移動至本行的某一個位置
\v	Vertical tab	垂直移動至某一行的位置
\0ddd	Octal constant	使用8進位表示數字, d必須小於7
\xddd \Xddd	Hexadecimal constant	使用16進位表示數字
\\	Backslash	顯示「 \ 」字元
\'	Single quote	顯示「 ' 」字元

Character escape sequences (3)

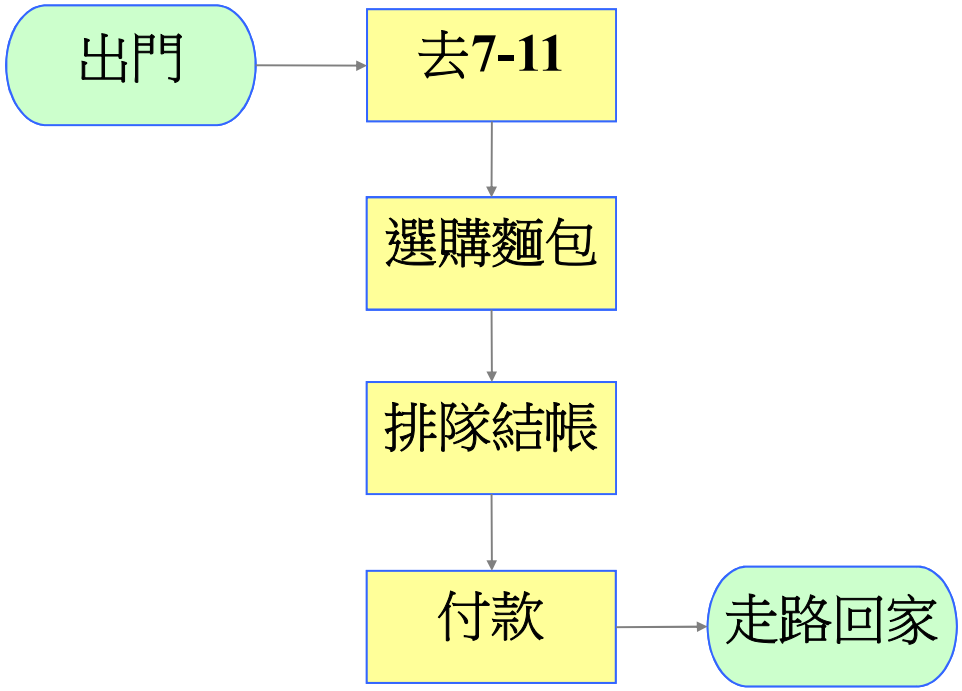
\"	Double quote	顯示「 " 」字元
%%	Percent	顯示「 % 」字元
\?	Question mark	顯示「 ? 」字元

```
#include <stdio.h>
void main(void)
{
    printf ("Welcome to");
    printf ("London!");
    printf("\nHow do we\njump\n\ntwo lines?\n");
    printf { "\n");
    printf("It will rain\ntomorrow\n");
}
```

=

```
#include <stdio.h>
void main(void)
{
    printf("Listen to the beep now. \a");
    printf("\nWhere is the 't' in cat\b?\n\n");
    printf("I earned $50 \r Where is the money?\n);
    printf("The rabbit jumps \t\t two tabs.\n\n");
    printf("Welcome to\
    New York! \n\n");
    printf("From " "Russia \
    with" "Love.\n");
    printf("Print 3 double quotes
}
```

程式流程圖:
循序性結構



程式流程圖:
循序性結構