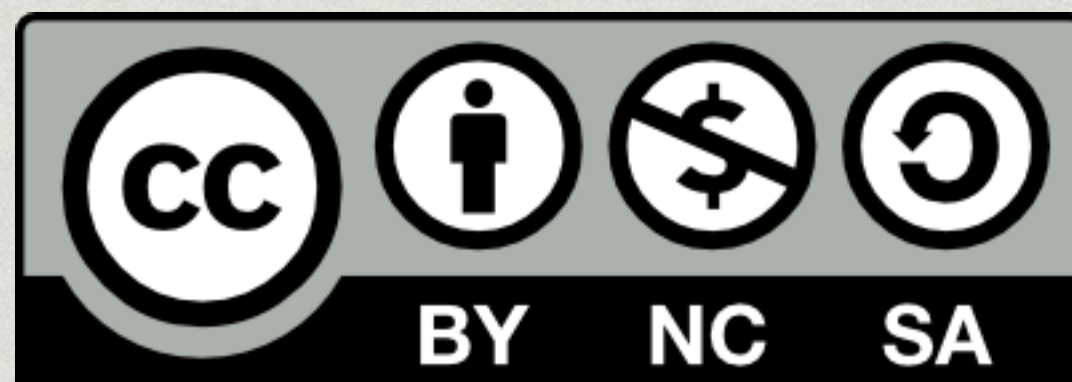


COMPILE, LINK, EXECUTION

DATE HUANG
BAMBOOFOX



About me

- ✱ 毫無回應，就只是個伊達
- ✱ **Date Huang**
- ✱ **NCU CSIE**

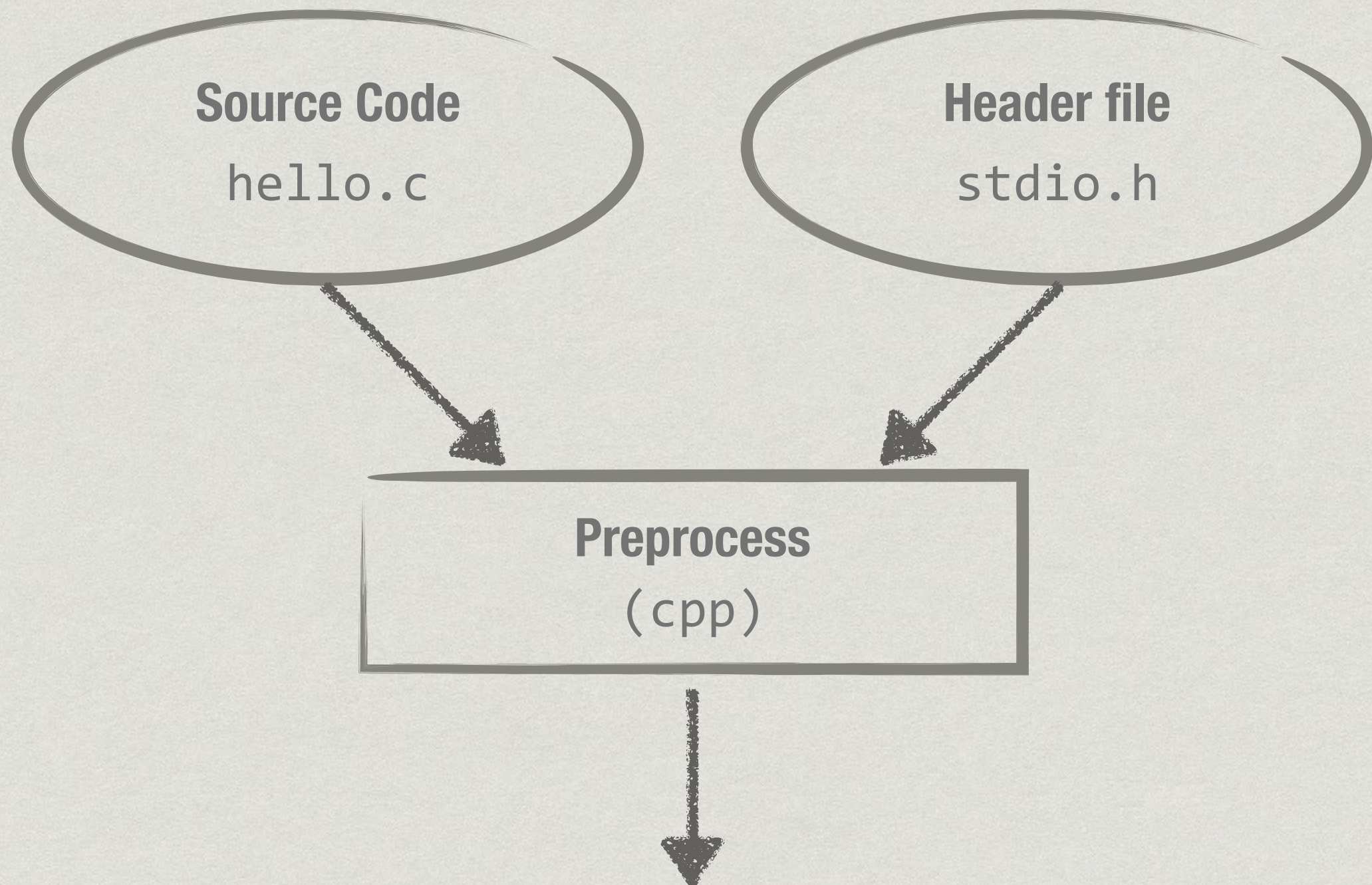
Demo Code

* https://github.com/tjjh89017/compile_link

Compilation Flow

- * **Preprocess**
- * **Compilation**
- * **Assembly**
- * **Link**

Compilation Flow





Preprocessed Code
hello.i



Compilation
(cc1)



Assembly Code
hello.S





Assembler
(as)



Object file
hello.o

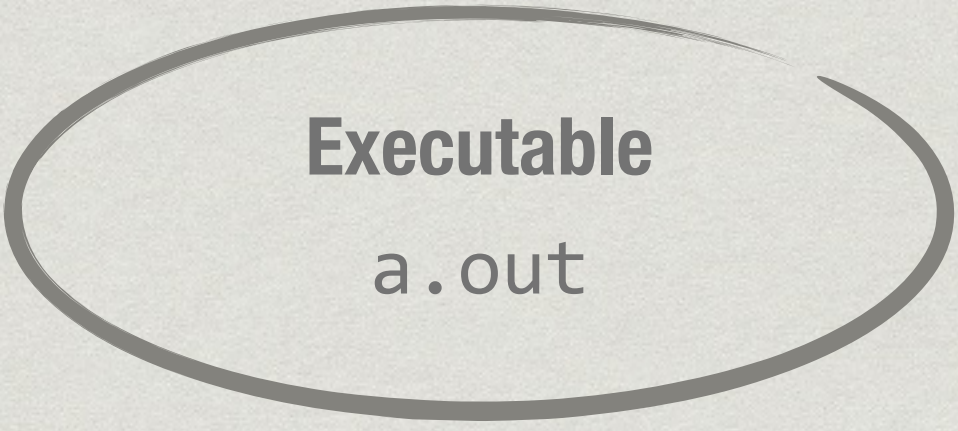


Linker
(ld)



Static Library
libc.a





Executable

a.out

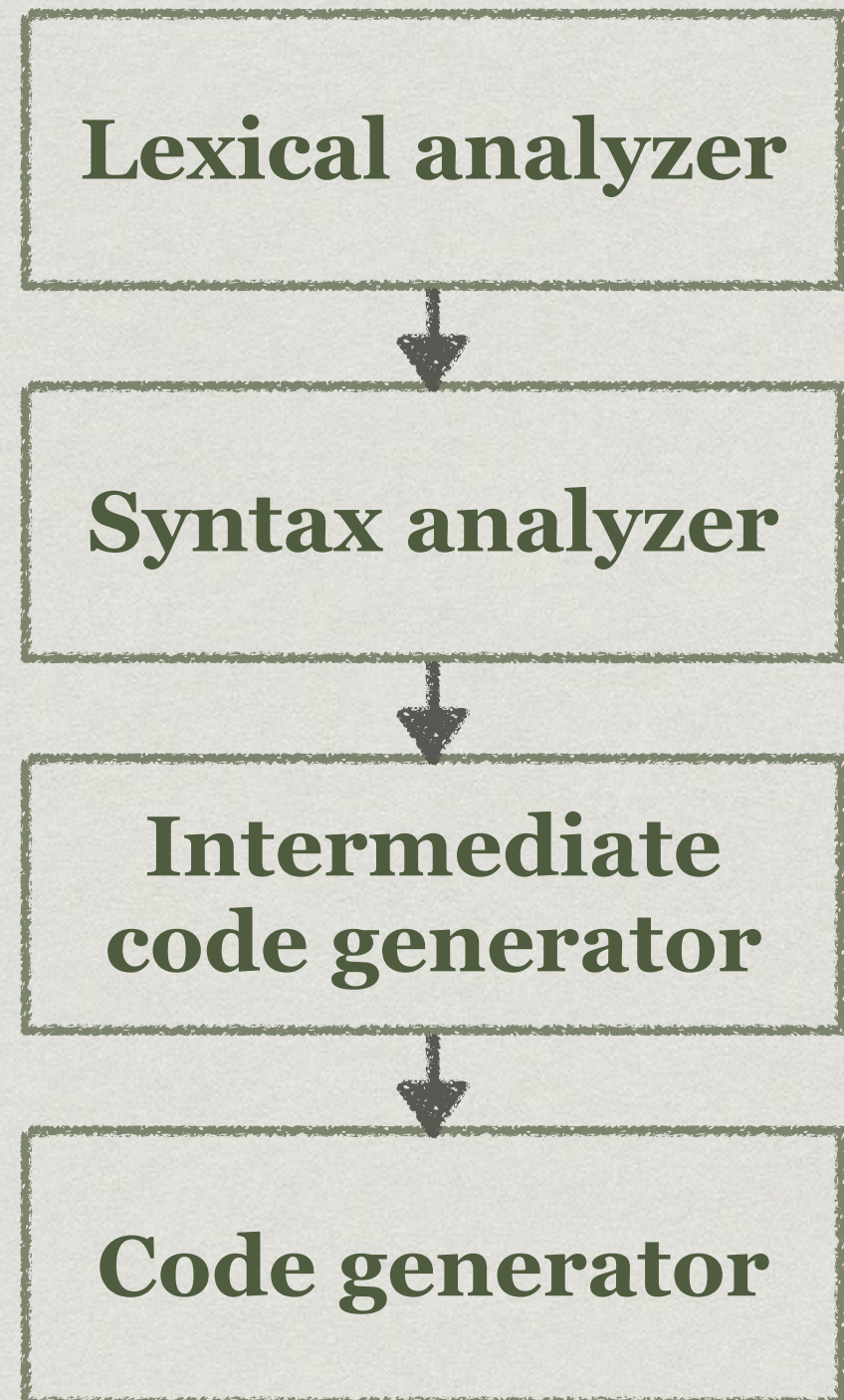
Preprocess

- * **Macro expansion**
- * **Header files**
- * **Conditional compilation**
- * **Line control (Debug)**
- * **Lines beginning '#' as directives**

```
#define PI 3.1459f  
  
#include <stdio.h>  
  
#if VERSION >= 2  
  
#ifdef __LINUX__  
  
etc.
```


Compilation

- * Lexical analyzer
- * Syntax analyzer
- * Intermediate code generator
- * Optimization (optional)
- * Code generator



Assembly

- * Turn Assembly Code into Machine Code

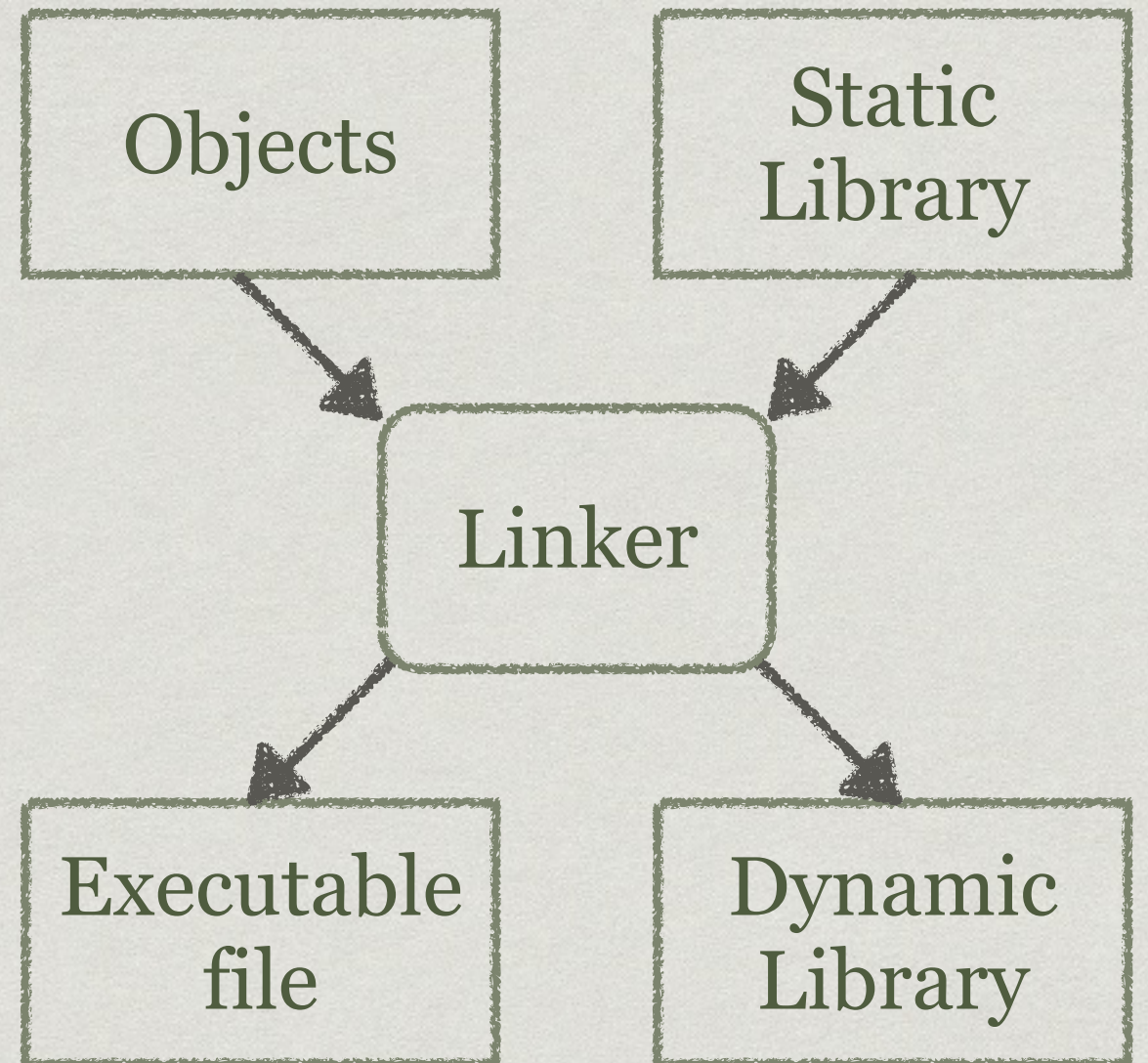
MOV AL, 61h

↓

B0 61

Link

- * **Combine objects and static libraries.**
- * **Replace symbols with “address” or “offset”.**



Section

- * **Text Section**
 - * **Code!!!**
- * **Data Section**
 - * **Global variable with initial value**
- * **BSS Section**
 - * **Global variable without initial value**


```
int a = 0;           // .data
int b;               // .bss
const double PI = 3.14159f; // .rodata

int main(){

    int c = 321;      // stack
    char *d = malloc(16); // heap

    return 0;
}
```


Static Linking

- * **Combine the library you need into program.**
- * **Filename Extension**
 - * **In Unix-like: .a**
 - * **In Windows: .lib**

Dynamic Linking

- * All program share one library.
- * Filename Extension
 - * In Unix-like: .so
 - * In Windows: .dll

Static Linking

- ✱ **Advantage**

- ✱ **Faster than dynamic linking**
- ✱ **Portable (Machine)**

- ✱ **Disadvantage**

- ✱ **Makes program bigger**
- ✱ **Modifying Library needs rebuild program**

Dynamic Linking

- ✱ **Advantage**

- ✱ **Modifying Library doesn't need to rebuild program**
- ✱ **Portable (Library Version)**

- ✱ **Disadvantage**

- ✱ **Need additional library**
- ✱ **Slower than static linking**

Relocation

- ✱ While linking, all data and functions will be different place.
- ✱ Replace symbols with “address” or “offset”.

`call rip+0x0`

`e8 00 00 00 00`

↓

`call rip+0xffffffffe75`

`e8 75 fe ff ff`

Override Dynamic Library

- * **Use environment variable LD_PRELOAD to load custom library**
- * `LD_PRELOAD=./libtest.so ./main`

Reference

- ✱ **How A Compiler Works: GNU Toolchain**
 - ✱ <http://www.slideshare.net/jserv/how-a-compiler-works-gnu-toolchain>
- ✱ **The Internals of "Hello World" Program**
 - ✱ <http://www.slideshare.net/jserv/helloworld-internals>
- ✱ **Internal and External Linkage in C**
 - ✱ https://github.com/u1240976/mess_note/blob/master/talk/internal_and_external_linkage_in_C.rst