

Princeton University
COS 217: Introduction to Programming Systems
The gcc Command

```
#include <stdio.h>
int main(void)
/* Print "hello, world\n" to stdout. */
{
    printf("hello, world\n");
    return 0;
}
```

hello.c
Source code
C language
Contains preprocessor directives



Preprocess

`gcc -E hello.c > hello.i`

```
...
int printf(char *format, ...);
...
int main(void)
{
    printf("hello, world\n");
    return 0;
}
```

hello.i
Source code
C language
Contains declaration of printf() function
Missing definition of printf() function



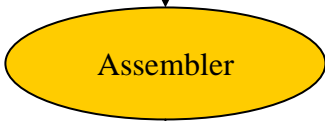
Compile

`gcc -Wall -ansi -pedantic -S hello.i`

Continued on next page

```
.section .rodata
cGreeting:
.asciz "hello, world\n"
.section .text
.globl main
.type main,@function
main:
pushl %ebp
movl %esp, %ebp
pushl $cGreeting
call printf
addl $4, %esp
movl $0, %eax
movl %ebp, %esp
popl %ebp
ret
```

hello.s
Source code
Assembly language
Missing definition of printf() function



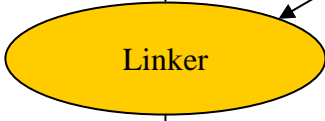
Assemble
gcc -c hello.s

```
100101000110100100100...
```

hello.o
Object code
Machine language
Missing definition of printf() function

```
11110010000010100100110...
```

libc.a
Library containing
machine language definition
of printf() function (and many
others)



Link
gcc hello.o -lc -o hello

```
001010000101000000111110...
```

hello
Executable code
Machine language

Shortcut:
gcc -Wall -ansi -pedantic hello.c -o hello