


# Hello World!

CS2263 – Systems Software Development




1

## Learning Outcomes

At the conclusion of this lecture students should be able to:

- Describe the evolution of C
- List some versions of C
- Explain the actions of a very simple C program
- Compile and execute a single file C program



2

## First C Program (from Chapter 1)

```
// first.c
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char * * argv)
{
    int a = 5;
    int b = 17;
    printf("main: a = %d, b = %d, argc = %d\n", a, b, argc);
    return EXIT_SUCCESS;
}
```



3

## Where Did C Come From?

- Ken Thompson, 1960s: BCPL ► B
- Dennis Ritchie: 1971: NB ("New B") then C
- 1973: UNIX rewritten in C



4

## Standardization of C

### K&R

- Described in Kernighan and Ritchie, "The C Programming Language" (1978)
- De facto standard

### C89/C90

- ANSI standard X3.159-1989 (completed in 1988; formally approved in December 1989)
- International standard ISO/IEC 9899:1990

### C99

- International standard ISO/IEC 9899:1999
- Incorporates changes from Amendment 1 (1995)

### C11 (late 2011): Minor updates



5

## Features as of C99

- Intermingled declarations and code:  

```
for(int i=0; i<n; i++){...}
```
- New data types, including bool (stdbool.h) and complex (complex.h)
- Variable length arrays (optional in C11, but supported by gcc):  

```
int n;  
scanf("%d",&n);  
int array[n];
```
- One-line comments:  

```
// InLine Comment
```



6

## C99 and GCC

- "C99 is substantially completely supported as of GCC 4.5"
- FCS Linux has 4.8.x
- compile with `-std=c99` flag.



7

## C-based Languages

- C++ includes all the features of C, but adds classes and other features to support object-oriented programming.
- Java is based on C++ and therefore inherits many C features.
- C# is a more recent language derived from C++ and Java.
- Perl has adopted many of the features of C.



8

## Properties of C

- Low-Level
- Small
- Permissive



9

## C is...

*"A language that combines all the elegance and power of assembly language with all the readability and maintainability of assembly language."*

— New Hacker's Dictionary

*"... one of the main causes of the fall of the Roman Empire was that, lacking zero, they had no way to indicate successful termination of their C programs."*

— Robert Firth



10

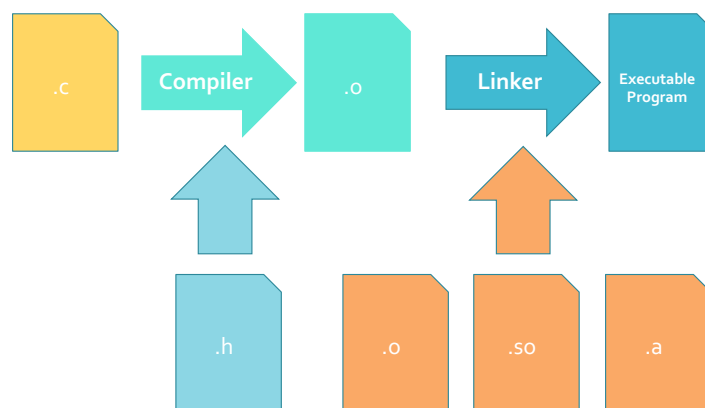
## First C Program (from Chapter 1)

```
// first.c
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char * * argv)
{
    int a = 5;
    int b = 17;
    printf("main: a = %d, b = %d, argc = %d\n", a, b, argc);
    return EXIT_SUCCESS;
}
```



11

## Compilation Process



12

## Compiling and Linking

Three steps:

1. **Preprocessor**: process commands beginning with #
  - `gcc -E first.c`
2. **Compiler**: translates C code into machine instructions (object code)
  - `gcc -c first.c`
3. **Linker**: combine object code with other code (e.g. libraries) needed to produce executable program.
  - `gcc -o first first.o`
  - OR-
  - `gcc -o first first.c`



13

## Command Line Execution

- `./first`  
main: a = 5, b = 17, argc = 1
- `./first one two`  
main: a = 5, b = 17, argc = 3
- `./first one two > first.txt`
- `cat first.txt`  
main: a = 5, b = 17, argc = 3
- What do we know?

```
// first.c
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char * * argv)
{
    int a = 5;
    int b = 17;
    printf("main: a = %d, b = %d, argc = %d\n", a, b, argc);
    return EXIT_SUCCESS;
}
```



14

## ForNextDay()

- Read Chapter 1 of the Text
- Create `first.c`
- Compile it to preprocessor (`-E`) and examine it
- Compile and link it to an executable named `mine` (`-o mine`)
- Execute it with several words, as in the example slide.
- Write an explanation of what the `printf ( )` statement is doing.

