# C Programming Language

Jamal Hussein

jah1g12@ecs.soton.ac.uk

Electronic and Computer Science

October 29, 2015

UNIVERSITY OF Southampton

# Outline

# What is C?

- C language developed by Dennis Ritchie at Bell Labs in 1972
- used in reimplementing Unix operating system
- C standards:
  - "K & R" C
  - ANSI C (C89 or C90): the most portable version of C
  - C99: extended C89 by adding new features such as new data types and variable length arrays
- We use GCC compiler which uses, by default, ANSI C

# C language features

- Provides lowlevel access to memory
- used in system programming
  - Operating systems: such as Linux
  - Microcontrollers: cars and planes
  - embedded systems: phones, portable electronics, ..
  - ...
- used in derivation of C++, Objective C, C#
- C has enormous influence on other languages: Java, PHP, Python, ...
- High-level but close to the hardware
- Fast: allows low-level programming
- compiles to native code
- C lacks: garbage collection, OOP, ...

# Installation

In this tutorial we use the gcc compiler form the command line, to write C codes, any text editors can be used: notepad (Windows), vim (Linux, Mac OS).

- Linux: included with most linux distributions
  - you can check it by entering this into the command line: gcc -v
- Mac OS: you need to install Xcode
  `https://developer.apple.com/xcode/`
- Windows: download and install MinGW, ensure that bin subdirectory is in PATH
  `http://www.mingw.org`

# Hello World!

```c
#include <stdio.h>

int main(void)
{
        /* This program prints
                Hello World!*/
        printf("Hello World!\n");
        return 0;
}
```

# Hello World!

- C standard library header files include function definitions, variable declarations.

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

- Other header files are `math.h`, `stdlib.h`, `string.h`, `time.h`.
- the main entry of the of C programs, returns integer (int) and has no parameters (void) followed by a curly bracket

```
int main(void)
{
```

# Hello World!

- comments can span multiple lines

```
/* This program prints
        Hello World! */
```

- this C statement prints "Hello World!", statements end with a semicolon, "\n" is an escape character means "newline"

```
printf("Hello World!\n");
```

- main function return an integer (0 indicates the program ends normally)

```
return 0;
}
```

# Complie and run

- save the code in hello.c (.c is the extension used for c language programs)
- compile the program by entering to the command line

```
gcc hello.c
```

- this create an executable file a.out on Linux and Mac OS, and a.exe on Windows
- now you can run it by typing ./a.out (Linux MacOS) or a.exe (Windows)
- you can change the name of the output file with:

```
gcc hello.c -o hello
```

# C keywords and Identifiers

- C keywords

| | | | | | |
|---|---|---|---|---|---|
| auto | break | case | char | const | continue |
| default | do | double | else | enum | extern |
| float | for | goto | if | int | long |
| register | return | short | signed | sizeof | static |
| struct | switch | typedef | union | unsigned | void |
| volatile | while | | | | |

# C keywords and Identifiers

- Identifiers and variable names
  - consists of letters (A..Z, a..z), digits (0..9), and underscore(_).
  - must begin with a letter or an underscore.
  - cannot be a reserved word (keywords).
  - cannot contain special characters.
- C special characters

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| , | < | > | . | _ | ( | ) | ; | $ | : |
| % | [ | ] | # | ? | ' | & | { | } | " |
| ^ | ! | * | / | \| | - | \ | | + | |

# Datatypes

- Integer Types

| datatype | size | range |
|---|---|---|
| char | 1 byte | -128 to 127 or 0 to 255 |
| unsigned char | 1 byte | 0 to 255 |
| signed char | 1 byte | -128 to 127 |
| int | 2 or 4 bytes | -32,768 to 32,767 or -2,147,483,648 to 2,147,483,647 |
| unsigned int | 2 or 4 bytes | 0 to 65,535 or 0 to 4,294,967,295 |
| short | 2 bytes | -32,768 to 32,767 |
| unsigned short | 2 bytes | 0 to 65,535 |
| long | 4 bytes | -2,147,483,648 to 2,147,483,647 |
| unsigned long | 4 bytes | 0 to 4,294,967,295 |

# Datatypes

- Integer Types
- in `unsigned` the most significant bit (MSB) will not be used as sign (+ or -)
- The header file `limits.h` has many useful constants to check the range of different datatypes: `SCHAR_MIN`, `SCHAR_MAX`, `UCHAR_MAX`, `INT_MIN`, `INT_MAX`, `UINT_MAX`, `...`

# Datatypes

- Floating-Point Types

| datatype | size | range | precision |
|---|---|---|---|
| float | 4 byte | 1.2E-38 to 3.4E+38 | 6 decimal places |
| double | 8 byte | 2.3E-308 to 1.7E+308 | 15 decimal places |
| long double | 10 byte | 3.4E-4932 to 1.1E+4932 | 19 decimal places |

- The header file `float.h` provides constants to check the range of float datatypes : `FLT_MIN, FLT_MAX, ...`
- the sizes and ranges may be different on you computer based on the platform you use (hardware and OS)

# sizeof operator

```
#include <stdio.h>
int main() {
        printf("sizeof(char) is %d byte(s)\n",
                (int) sizeof(char));
        printf("sizeof(short) is %d byte(s)\n",
                (int) sizeof(short));
        printf("sizeof(int) is %d byte(s)\n",
                (int) sizeof(int));
        ...
        return 0;
}
```

# Literals

| datatype | literal |
|---|---|
| `char` | `'c'`, `'\t'`, `'\u02C0'`, `99` |
| `int` | `85` (decimal), `0213` (octal), `0x4b` (hexa) |
| `unsigned int` | `30u` |
| `long` | `30l` |
| `unsigned long` | `30ul` |
| `float` | `3.14159f`, `.58f`, `123e4f` |
| `double` | `3.14159`, `.58d`, `123e4` |

- escape sequence characters: `\'` (' character), `\"` (" character), `\?` (? character), `\a` (Alert or bell), `\b` (Backspace), `\f` (Form feed), `\n` (Newline), `\r` (Carriage return), `\t` (Horizontal tab), `\v` (Vertical tab).