

C Programming Logbook

Joseph Butterworth

06/10/20

Introduction to C Programming: Toolchain

<https://secure.ecs.soton.ac.uk/notes/ellabs/1/c1/c1.pdf>

https://en.wikibooks.org/wiki/C_Programming/Variables

	Listing 1	Listing 2	Listing 3
No error message	✓		
An error from the preprocessor		✓	
An error from the compiler			✓
An error from the linker			

Can declare multiple variables in one line as long as they are the same type.

```
int anumber, anothernumber, yetanothernumber;
```

Can assign it a value

```
int some_new_number = 4;
```

Can make multiple variables have the same value

```
anumber = anothernumber = yetanothernumber = 8;
```

A variable can be named anything as long as it is made of letters, numbers and underscores. It must begin with a letter.

int is an integer that can be stored as a signed 32 bit number.

char is a character in the ASCII character set. When defining a char it can be expressed as the character or its corresponding ASCII number.

float is an inexact representation of a real number, meaning it isn't always 100% accurate. It can store decimals, and numbers much greater and much smaller than an int can.

double is a double precision float, meaning it is a float that can store twice as much (floats on steroids!). It takes up more space than a float so is not preferable when space is at a minimum.

```
size_t size;  
int i;  
size = sizeof(i);
```

const int may be used to define a magic number, this is a variable that will stay the same within the code but may be needed to change between versions.

#define may also be used to define a magic number. #define is used within the preprocessor so if used incorrectly can cause more damage to the code than const int.

Cannot use command line, suspect c compiler installed incorrectly. Reinstalled minGW with all repositories. Then add a path to variable environment.

[illegible]

Initially unable to find the path to hello.c so moved it to E:\ making it easy to access. The code when copied did not compile as spaces were added to the stdio.h directory.

```
Select Command Prompt
C:\Users\ludde>gcc F:\1201_Programming\c1\hello.c -o hello
gcc: error: F:\1201: No such file or directory
gcc: error: Programming\c1\hello.c: No such file or directory
gcc: fatal error: no input files
compilation terminated.

C:\Users\ludde>gcc F:\1201_Programming\c1\hello.c -o hello
gcc: error: F:\1201_Programming\c1\hello.c: No such file or directory
gcc: fatal error: no input files
compilation terminated.

C:\Users\ludde>ls
'ls' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\ludde>gcc E:\hello.c -o hello
E:\hello.c:3:23: fatal error:  stdio .h : No such file or directory
# include < stdio .h >
^
compilation terminated.

C:\Users\ludde>gcc E:\hello.c -o hello
E:\hello.c:3:22: fatal error:  stdio.h : No such file or directory
# include < stdio.h >
^
compilation terminated.

C:\Users\ludde>
```

Cannot allow spaces in file or folder names as this will be read as a break by the terminal, and will not path to your file properly.

Using the dir command I am able to find all the files and folders in a directory.

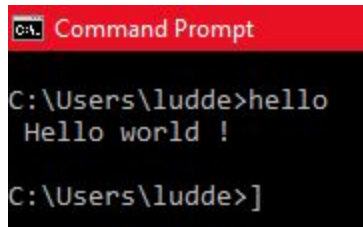
```
Command Prompt
C:\Users\ludde>dir F:\
Volume in drive F is KINGSTON
Volume Serial Number is 1D70-0F2C

Directory of F:\

06/10/2020  16:30    <DIR>          1055_Maths
06/10/2020  16:30    <DIR>          1200_Electronic_Circuits
06/10/2020  16:31    <DIR>          1201_Programming
06/10/2020  16:31    <DIR>          1202_Digital_Systems_and_Microprocessors
06/10/2020  16:35    <DIR>          1203_Mechanics
06/10/2020  16:33    <DIR>          1205_Solid_State_Devices
06/10/2020  16:33    <DIR>          1206_Electrical_Materials_and_Fields
06/10/2020  16:36    <DIR>          1207_Electronic_Systems
               0 File(s)                0 bytes
               8 Dir(s)  30,985,420,800 bytes free

C:\Users\ludde>
```

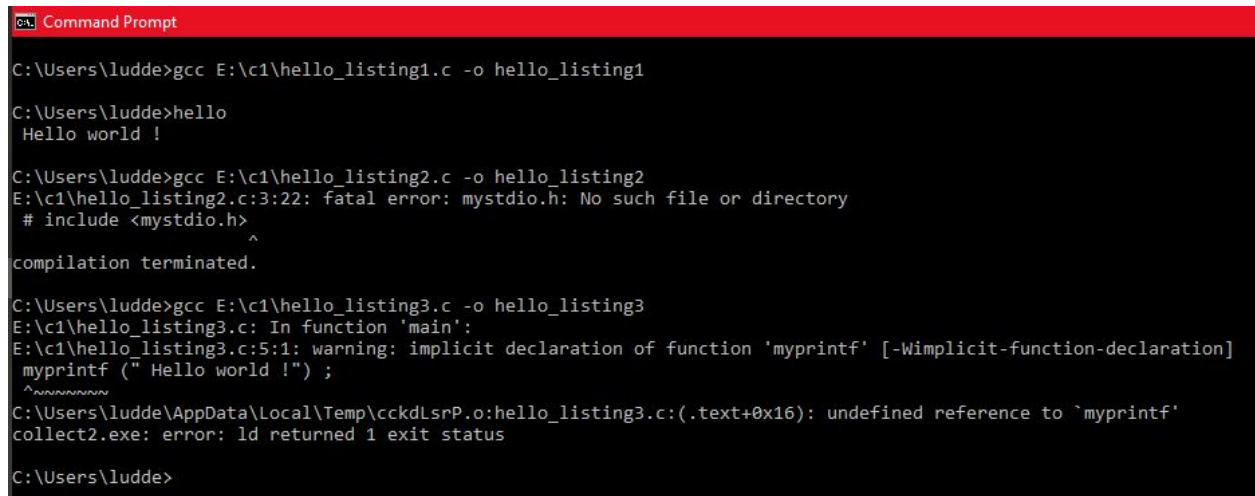
Now that the c has been compiled it is running on the command line. When I type hello, because of the program, the command line responds with “Hello World!”.



```
C:\Users\ludde>gcc E:\c1\hello_listing1.c -o hello_listing1
C:\Users\ludde>hello_listing1
Hello world !
C:\Users\ludde>]
```

The first listing works fine and is able to be compiled. This means “Hello world!” responded. If I were to remove the “*/” from the second line then the entire set of code would be commented and there would be nothing to compile.

The second listing had the wrong library in the code this means that when the preprocessor tried to install it it was unable to. This resulted in an error message of “No such file or directory”. The third listing was not able to be compiled in the linker because it used a different print function. This could not be compiled because this function was not defined.



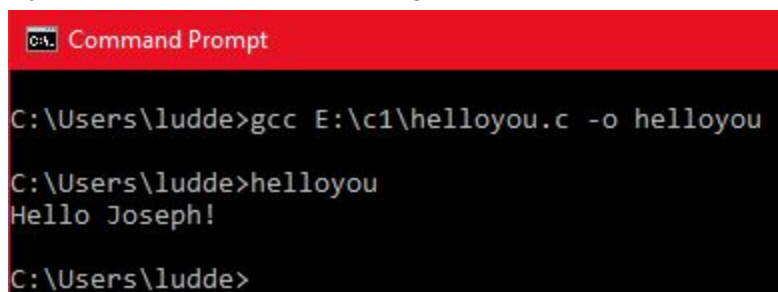
```
C:\Users\ludde>gcc E:\c1\hello_listing1.c -o hello_listing1
C:\Users\ludde>hello_listing1
Hello world !

C:\Users\ludde>gcc E:\c1\hello_listing2.c -o hello_listing2
E:\c1\hello_listing2.c:3:22: fatal error: mystdio.h: No such file or directory
# include <mystdio.h>
^
compilation terminated.

C:\Users\ludde>gcc E:\c1\hello_listing3.c -o hello_listing3
E:\c1\hello_listing3.c: In function 'main':
E:\c1\hello_listing3.c:5:1: warning: implicit declaration of function 'myprintf' [-Wimplicit-function-declaration]
myprintf (" Hello world !") ;
^~~~~~
C:\Users\ludde\AppData\Local\Temp\cckdLsrP.o:hello_listing3.c:(.text+0x16): undefined reference to `myprintf'
collect2.exe: error: ld returned 1 exit status
C:\Users\ludde>
```

My predictions from the preparation activity were relatively accurate. I was able to predict what the errors in the code would be for listings 1 & 2. I was incorrect in saying there would be an error in the compiler.

With the hello you program I was able to edit the variable so that the program would call me by my name (or whatever name I gave it).



```
C:\Users\ludde>gcc E:\c1\helloyou.c -o helloyou
C:\Users\ludde>helloyou
Hello Joseph!
C:\Users\ludde>
```