Lab 0

Quick tips on how to write, compile and run a C program

CSI3131 Operating Systems

Write “Hello World Code” in C

- To write a C program and save it in Linux or macOS, you can follow these steps:
  - Open a text editor. Linux and macOS come with various text editors pre-installed, such as Nano, Vim, or Emacs. Alternatively, you can use a graphical text editor like Sublime Text, Atom, or Visual Studio Code.

  - Create a new file in the text editor and enter the C program code. For example, you can write the "Hello, World!" program:

```c
#include <stdio.h>

int main() {
    printf("Hello, World!\n");
    return 0;
}
```

Write “Hello World Code” in C

- Save the file with a **.c** extension. Choose a meaningful filename for your program, such as **hello.c** Make sure to use the **.c** extension to indicate that it's a C program.
- Choose a directory where you want to save your program. For example, you can save it in your home directory or create a new folder specifically for your C programs.
- Open the terminal on your Linux or macOS system.
- Navigate to the directory where you saved the C program using the **cd** command. For example, if you saved the program in your home directory, you can use the ‘cd’ command to navigate there.

Compile and Run the code

- Compile the C program using a C compiler. On Linux and macOS, the most common C compiler is **gcc**. In the terminal, use the following command to compile the program:

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

- This command tells **gcc** to compile the **hello.c** file and create an executable file named **hello**.
- After successful compilation, you can run the program by entering the following command in the terminal:

```
./hello
```

- This command executes the compiled program, and you should see the "Hello, World!" message printed in the terminal.

Main structure of C program

- Here's an explanation of each line in the "Hello, World!" program written in C:

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

This line is a preprocessor directive that tells the C compiler to include the **stdio.h** header file, which stands for "standard input/output." This header file provides functions like **printf()** that allow you to perform input and output operations.

Main structure of C program

```
int main() {
```

This line declares the **main()** function, which is the entry point of a C program. It indicates that the program's execution will start from this point. The **int** before **main()** specifies that the function returns an integer value.

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

This line uses the **printf()** function to print the text "Hello, World!" to the console. The **printf()** function is part of the **stdio.h** library and is used for formatted output. The text to be printed is enclosed within double quotes. The **\n** represents the newline character, which moves the cursor to the next line after printing the message.

Main structure of C program

```
return 0;
```

This line is a return statement. It ends the **main()** function and returns the value 0 to the operating system. By convention, a return value of 0 typically indicates successful execution of the program.

The program ends after the **return 0**; statement, and control is handed back to the operating system.

Summary

- Using previous slides you must have learned:
  - How to write a C program
  - Compile the code to get the executable file
  - Run the executable file to get the output