



TUTORIAL #2: Introduction to C

**Faculty of Engineering and Applied Science
Operating Systems SOFE-3950U**

74171 Group 11
Awais Ahmad - 100785259
Mithusan Arulampalam - 100787123
Jawad Kohja - 100786085
Mohammad Mohammadkhani - 100798165

Date: January 26, 2023

Tutorial 2

Conceptional Questions:

- 1. What are some of the benefits of the C language? Name an example where the C language is used (hint there are many!).**

Some benefits of the C language are it has a low cost, fast execution speed, easy debugging, among other benefits. An example where the C language is used is the Linux operating system.

- 2. What is a compiler, what does it do?**

A compiler translates the programming language's source code into machine code. It helps execute the code on the computer system which is written by the developer.

- 3. What is a makefile, what does it do?**

A makefile is a list that contains all the files that are needed to execute the program like the resource files and library files. The way that makefiles works is that once the code is executed the makefile will add all the incorporated files to help execute the task

- 4. Name 5 header files from the C library and explain their purpose.**

- `#include<stdio.h>`: perform input and output operations
- `#include<stdlib.h>`: perform standard utility functions
- `#include<math.h>`: used to perform mathematical operations like `pow()`
- `#include<time.h>`: perform time related functions such as `getDate()`
- `#include<string.h>`: perform operations on arrays of characters

- 5. Lookup one function from each of the header files and put down the function name and describe what it does**

- `stdio.h` - `fclose` - close a file/handle that is open
- `stdlib.h` - `exit(status)` - this terminates the execution of the program
- `math.h` - `sin(x)` - return the sin of a radian angle
- `time.h` - `clock()` - processor time used by the program
- `string.h` - `strstr()` - returns a pointer to the beginning of the first occurrence of the word from `string2` in `string1`.

Application Questions:

Task 1

```
mithusan@mithusan-VirtualBox:~/Desktop/tut2$ gcc task1.c -o t1
mithusan@mithusan-VirtualBox:~/Desktop/tut2$ ./t1
1 2 3 4 5 6 7 8 9 10
```

Task 2

```
mithusan@mithusan-VirtualBox:~/Desktop/tut2$ gcc task2.c -o t2
mithusan@mithusan-VirtualBox:~/Desktop/tut2$ ./t2
Greater Than, Greater Than, Less Than, Greater Than, Equal To
```

Task 3

```
mithusan@mithusan-VirtualBox:~/Desktop/tut2$ gcc task3.c -o t3
mithusan@mithusan-VirtualBox:~/Desktop/tut2$ ./t3
hello world
```

Task 4

```
captainmetal@captainmetal-VirtualBox:~/Desktop/Tasks$ ./t4
1 is Odd
2 is Even
3 is Odd
4 is Even
5 is Odd
6 is Even
7 is Odd
8 is Even
9 is Odd
10 is Even
```

Task 5

```
mithusan@mithusan-VirtualBox:~/Desktop/tut2$ gcc task5.c -o t5 -lm
mithusan@mithusan-VirtualBox:~/Desktop/tut2$ ./t5
62.072538
(86.000000,15.000000) (83.000000,77.000000)

58.309519
(35.000000,92.000000) (93.000000,86.000000)

44.821870
(21.000000,27.000000) (49.000000,62.000000)

48.270074
(59.000000,26.000000) (90.000000,63.000000)

38.626416
(26.000000,36.000000) (40.000000,72.000000)

68.505474
(68.000000,29.000000) (11.000000,67.000000)

65.000000
(30.000000,23.000000) (82.000000,62.000000)

41.868843
(35.000000,2.000000) (67.000000,29.000000)

36.055513
(58.000000,67.000000) (22.000000,69.000000)

48.270074
(56.000000,42.000000) (93.000000,11.000000)
```