TSN3151: Laboratory 01

We will use this laboratory session as a quick introduction to what is planned for the rest of the trimester. Note that some of the planned laboratories may not be realized due to possible system conflicts and so on.

The objectives of the laboratory sessions are:-

- To have a very basic understanding of parallel processing using Message Passing Interface (MPI) protocol.
- The MPI session will also introduce the concept of distributed memory parallel processing.
- Shared memory parallel processing with be introduced using threads.

For this first session, we will just get used to the LINUX environment and some LINUX commands.

Guidelines:

- You are encouraged to conduct the laboratory exercises and assist each other in groups NOT larger than 4 people. This is to ensure that the laboratory sessions do not get distractingly noisy. In general, your tutor WILL NOT be demonstrating or lecturing during the laboratory session but will be around for you for clarification.

NOTE:

The laboratory PCs may not be setup appropriately. You can use your own laptop.

You can install Windows Subsystem for Linux (WSL) on your Windows laptop: https://pureinfotech.com/install-wsl-windows-11/

If you have a Macintosh or LINUX laptop, you don't have to do this – it's already built-in.

Introduction to Command Line Interface

Log in to the UBUNTU system. Start the Terminal application.

There are a few commands you need to know before you start on the CLI (the text screen). Once you started the terminal, spend some time to familiarizing yourself with the following commands that you can type in the CLI. The commands are in normal font whilst you can substitute those in *italics* with appropriate entries.

ls	list files and directories
mkdir	make a directory
cd *directory*	change to named directory
cd	change to home-directory
cd ..	change to parent directory
pwd	display the path of the current directory
cp *file1 file2*	copy file1 and call it file2
mv *file1 file2*	move or rename file1 to file2
rm *file*	remove a file
rmdir *directory*	remove a directory
cat *file*	display a file
more *file*	display a file a page at a time
grep *'keyword' file*	search a file for keywords
wc *file*	count number of lines/words/characters in file
command > *file*	redirect standard output to a file
command >> *file*	append standard output to a file
command < *file*	redirect standard input from a file
command1 | *command2*	pipe the output of command1 to the input of command2
man *command*	read the online manual page for a command

Additional software you will need

These pieces of software which are not in a standard Ubuntu installation are needed. They may already be on the lab PCs, but if you are using your own PC, you will need to install them.

Make sure your package repository is up to date.

```
> sudo apt update
```

You can install MPI by using the following command:

```
> sudo apt install openmpi-bin libopenmpi-dev
```

You can install the C compiler by using the following command:

```
> sudo apt install gcc
```

You can install the C++ compiler by using the following command:

```
> sudo apt install g++
```

You can install the Makefile system by using the following command:

```
> sudo apt install make
```

Familiarizing with environment and editor

I will leave this part for you to explore on your own. A few things to note:

(1) The initial screen you are greeted with when you login is a folder that is located at /home/<yourusername>/Desktop.

(2) The basic text editor that is available on all Linux system will be a program called "vi". However, it will take you a long time to learn how to use it if you are not familiar with it. For this laboratory, learn how to use "nano". If you are in WSL, you can also just edit your files with Notepad++ in Windows.

You can run (execute) the "nano" program simply by typing "nano" on the Command Line Interface (CLI).

Learn how to navigate your way around the system and also to familiarize yourself with the editor as you will need it.

(3) Write a C or C++ program (a "hello world" is enough) and figure out how to compile and run it from the command line.