Name: Wan Muhammad Rayhan Arwindra

NPM: 1806241210

This week we learnt about processes and memory. A process is an instance of a program that is currently being executed. A program can have many processes which relate to it. A process's smallest executable unit is called a thread, and multiple threads of the same process can communicate with each other. Each process has a unique identifier to it, called a PID (process identifier).

To view all processes running within the system, we can type the command "ps". This command comes with many options and flags, all relating to what kind of information about processes currently running we want the system to show. For example, "ps -e" will show every single process currently running, and "ps -x" shows all processes which is owned by the user executing this command.

A PID is assigned to automatically assigned to a process when it is created. In Linux, there is a limit to how many PIDs a system can have at once. If that limit is exceeded, then the PID will reset to 300. This is because Linux has 300 reserved PIDs for processes within the kernel, these processes cannot be overwritten by newly created processes.

Each process also has a parent, the parent PID can be viewed in the PPID header when running ps with option "-o ppid" or "-ef". A parent of a process is the process which creates the child process. This is mainly done by using the fork() command. When a child process is killed, it sends the signal SIGCHILD to the parent, to notify the parent to read the exit status of the child. Until the parent does this, the child will remain in the process table.

If a child remains in the process table even though it has been terminated, it is called a "zombie process". A zombie process cannot be killed, at least not directly, since it is already dead. To remove a zombie process, we can kill it's parent process instead. That way, the parent of the zombie will be PID 1, that process will then call wait() and read the exit status of the zombie. Afterwards, the process will be removed from the process table.

Linux systems also have environment variables. For example, one default environment variable is $HOME, where if we change directory into that variable by "cd $HOME", we would be transported into our home directory. This is because $HOME is a variable containing the

path to our home directory. To create our own variable, we can type “[KEY]=[VALUE]” in the shell., where key is the name of the variable, and value is the value contained within the variable. To print the variable, we can use the echo command. To make the variable global, we use the “export” command. Finally, to make the variable permanent, we can go to our bash configuration file, and set the variable name with the command “export [KEY]=[VALUE]”.