Qu 1 Adam is working in a IT company. He has been given a task to reduce the load of a system by killing some of the run in linux operating system which commands will be use?

(a) Kill Processes by name
(b) Kill a process based on the process name
(c) Kill a single process at a time with the given process ID.

- PS -ef :- To display all running processes in system
- Killall Process_nam :- Kill all running with given name
- Pkill process_name :- Kill processes by matching their name or pattern.
- Kill PID :- Safely terminates one Specific process.
- Kill -9 :- forcefully kill a process (used ~~if~~ when it is stuck)
- Sleep 300 & :- To create a dummy process for testing
- Killall sleep :- To terminate all process with the same name.

Que 2 write a program for process creation in C.

a] Orphan Process

- Orphan process :- creates when the parent process terminates before the child process. the child process continue execution and is adopted by the system.
- nano orphan.c => To create and write a C program for orphan process.
- gcc orphan.c -o orphan => To compile the C program.
- ./orphan => To execute the compiled program
- ps -efl grep orphan => To verify the running orphan process.

b] Zombie process :-

- zombie process :- create when the child process terminates but the parent process does not call wait() The child remain in the zombie state until the parent exits.
- nano zombi.c => To create the zombie process program

⊙ gcc zomlie.c -o zomlie ⇒ To compile the zomlie program.

⊙ ./zomlie ⇒ To execute the zomlie process Program.

⊙ PS -el | grep Z ⇒ To chick the zomli process status.

Q3 Create the process using fork() system call

- Child Process creation
- Parent Process creation
- PPID and PID

⊙ fork() ⇒ This is used to create a new process After fork(), both parent and child process execute independently.

⊙ nano . fork.c ⇒ To create a C program using fork()

⊙ gcc fork.c -o fork ⇒ To compile the fork Program

⊙ ./ fork ⇒ To execute the program and observe Parent and child processes

⊙ PID => Process ID which is a unique number assigned to each runing Process by a Operating system.

⊙ PPID => ~~Front~~ Parent process ID which is a the process ID of the parent process that created the current process.