**Topic: Inter Process Communication**

**Due on 21-September-2020, 04:15pm**

**Late Submission 22-September-2020, 04:15pm (Please note that assignments submitted late will not fetch full marks. Hence you are advised to submit your assignments by due date and time)**

**Mode of submission: MS Teams**

***Instructions:***

* You are required to upload the individual source (.c) files and one output file.
* You are required to follow proper naming convention for the files: ddmmyy118CS\*\*\*\*Q#.c.
* Apart from the source files you are also required to upload an output file containing the outputs of execution of all the assignments. The file name of the output file must be in the form: ddmmyy118CS\*\*\*\*Output.docx or .pdf
* The submitted programs will be checked for similarity through turnitin, before evaluation. So it is advisable not to borrow code from any source.

**Assignments:**

1. WAP to illustrate the use of PIPE for inter-process communication. Consider a parent process ‘P’ and its only child process ‘C’. P should write a message “Hello C\_PID”. C should read and display the message. C then writes “Hello P\_PID” which P should read and display. P\_PID and C\_PID are the PIDs of P and C respectively.
2. Extend the program in the previous Assignment (Skewed Process Tree P1 →P2→ ... → Pn) such that P1 accepts an integer dynamically, increments it by one and writes to a pipe. Process P2 then reads the value written by its parent, increments and writes to pipe. Subsequent children perform the same operation, except for the last one. The last process reads from the pipe increments the integer, and displays it.
3. WAP to verify that once a child process executes an exec() system call, its address space is changed to the address space of the executable file that it executes through exec() call.
4. Implement a program where a parent process forks two children: Child-1 will run "**ls -l**" command and send this as input to Child-2, which will run "**wc -l**". Parent will wait until both child processes terminate and then exit. The output of “**ls –l**” should not be displayed on the screen.

Note that this is equivalent to running, from UNIX shell prompt: $ ls -l | wc –l