# Assignment 3

Operating System Lab (**CS341**)

Department of CSE, IIT Patna

**Date:-** 22-Jan-2019 **Time:-** 3 hours

## Instructions:

1. All the assignments should be completed and uploaded by 5 pm. Marks will be deducted for the submissions made after 5 pm.
2. Markings will be based on the correctness and soundness of the outputs. Marks will be deducted in case of plagiarism.
3. Proper indentation and appropriate comments (if necessary) are mandatory.
4. You should zip all the required files and name the zip file as ***roll_no*.zip**, eg. **1501cs11.zip.**
5. Upload your assignment (**the zip file**) in the following link: https://www.dropbox.com/request/QJHjH1dZ5kSGsLCAOsso

## Questions:

1. Create a pipe by a parent process and create a child process to read any message which is written to that pipe by the parent process.

2. Write two programs: one called client.c, the other called server.c. The client program lists a prompter and reads from the keyboard two integers and one of the characters '+' or '-'. The read information is transmitted with the help of the system call **execl()** to a child process, which executes the server code. After the child (server) process finishes the operation, it transmits the result to parent process (client) with the help of the system call **exit()**. The client process prints the result on the screen and also reprints the prompter, ready for a new reading.

3. Write two programs file1.c and file2.c
   Program file1.c uses these :
   (a) fork() to launch another process

(b) exec() to replace the program driving this process, while supplying arguments to file2.c to complete its execution
(c) wait() to complete the execution of the child process
(d) file1.c takes two arguments x( a number less than 1) and n (number of terms to be added, 1 or more). For example: file1 0.5 5
(e) When the child process finishes, the parent prints:
**Parent(PID=yyy) : Done**

Program file2.c requires two arguments to obtain the approximate value of $e^x$ by adding the first n terms in the relation: $e^x$ = 1+x+x 2/2!+x 3/3!+....... and prints the result in the format:

**Child(PID=yyy) : For x = 0.5 the first 5 terms yields 1.6484375**

Hint : Child-specific processing immediately following the fork() command should load file2.c into the newly created process using the exec() command. This exec() command should also pass 2 arguments to the child.

4. Write a C program that takes a file named **process.txt** as a command line parameter and takes the words which are in the position multiple of 3 and stored in the file. The selected words are written to the display and the input file is left unchanged. Compile the C file into an executable named ”**dstring**”. Name the C file **dstring.c**.
   Now write a C program (**pstring.c**) that implements a command called ”**pstring**” that you will invoke from the shell prompt. The syntax of the command is ”**pstring thread.txt**”. When you type the command, the command opens a new xterm window (terminal), and then take the words which are in the position multiple of 3 stored in the file thread.txt using the program ”**dstring**”. Look up the man pages for xterm, fork and the different variations of exec* system calls (such as execv, execve, execlp etc.) to do this assignment. Submit the C files **dstring.c** and **pstring.c**. Justify the outputs.