BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE PILANI,
K. K. BIRLA GOA CAMPUS

I SEMESTER 2019-2020
Operating Systems (CS F372)
Assignment 2 [Non-evaluative Assignment]

Question #1.

Write a program to create a process tree as shown in Figure #1 (if the command line argument is 2, 3) and print the process identifiers in reverse order of its creation. The number of processes parent process P has to create is the first command line argument and the number of levels in the tree is the second argument. The program should create simultaneously executing processes in the same level as quickly as possible before child's execution starts. Each process should print its process identifier and its parent process identifier by using the function getpid( ) and getppid( ) respectively. Your program should not create an orphan process while execution. The parent process should also print the exit status of its child processes.

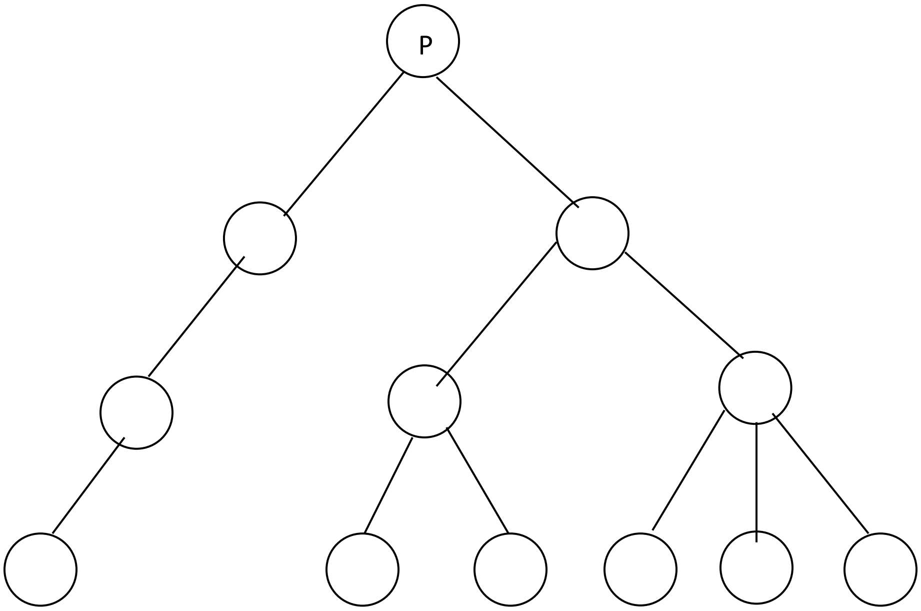

Figure #1

Question #2.

Since you are going to pursue a major in computers, you must be aware of how hacking is done (for your own benefit). We present to you a very sneaky way into "Some one's" user space where in you can edit your own marks for your assignment.

Aim: Change your Online #1 marks in "Some one's" user space at your own will without any penalty.

Input Format:

The input must be taken as follows

<filename> <ID No.> <New marks>

This input must be taken as arguments to the main function.

Format of the Marks file:

<ID No.> <\t> <Marks>

Assumptions:

- You must have super user privileges at all times.
- Create two groups 1.) Evaluator, 2.) Student *(please refer to figure1 at the end for the exact hierarchy expected of the directories)*
- There will be two evaluators in the evaluator group.
- Individual evaluator will have its own directories and subdirectories. One of them will contain your marks file.
- You should write your code in the Student's directory and not in the Evaluator's.
- There may be a directory of the same name as that of the file you want to search.
- The code will start searching the marks file from the Evaluator directory, down the hierarchy.
- There will be no space in the name of any file or directory.
- You can use 'chdir' system call as 'cd' command does not work in exec(). You must use exec() series system calls for executing the rest of the commands like 'pwd', 'ls', 'grep', 'rm'.

Procedure to change:

- The code should execute the following command
  ls <options> | grep <options> expression

You are expected to fork two processes (one for ls and the second one for grep taking the input as output of ls command) that will execute the above command.

- After execution of this command, check whether you have found the given marks file.
- If you haven't found the file in the current working directory (cwd), the process should access the shared space to check whether the flag is set. The flag will be set only if some other process already found the file. So this process can exit without proceeding further. If the flag is not set, fork as many number of children as the number of directories in the cwd.
- It is possible that some of the directories in your cwd have read only permissions. If this is the case, you are expected to change these permissions to read –write – execute mode for group, user and others.

  This can be done using the chmod() system call. After changing the permissions, you will be allowed to traverse down the hierarchy.
- In each child process, change the cwd to the directory for which the child has been forked in the previous step.
- Now, each of the child processes must search for the file in their respective cwd simultaneously.
- At any point, if the file is found,
  - The process will access the shared memory to check if any other process has already found the file and there is a flag set to 1 in the shared space. If the flag is already set, the process should readily terminate without continuing further.
  - If the flag is not set in the shared space, the process will,
    - Execute pwd command to get the path.
    - Write this path in a shared memory.
    - Set the flag to 1.
- A process will terminate normally if it does not find any directory after the execution of the command stated in the first step.
- Till all the children finish their execution, the main process must wait (No orphans should exist in the system).
- After the termination of all child processes, the main process will access the shared memory to change your marks in the file whose path is present.
- First search your ID no. in the given file and then edit your marks corresponding to it. You may need the creation of temporary file while the editing is being done.
- After your marks have been changed in the file, your main process should terminate.

- The following figure (Figure 2) gives the sample directory hierarchy for your reference.
- The marks file can be in any of the evaluator's directory.
- The figure below shows just a sample example for your understanding and not any concrete directory structure. So, please do not hardcode with reference to the figure below. Your code is expected to be generic.

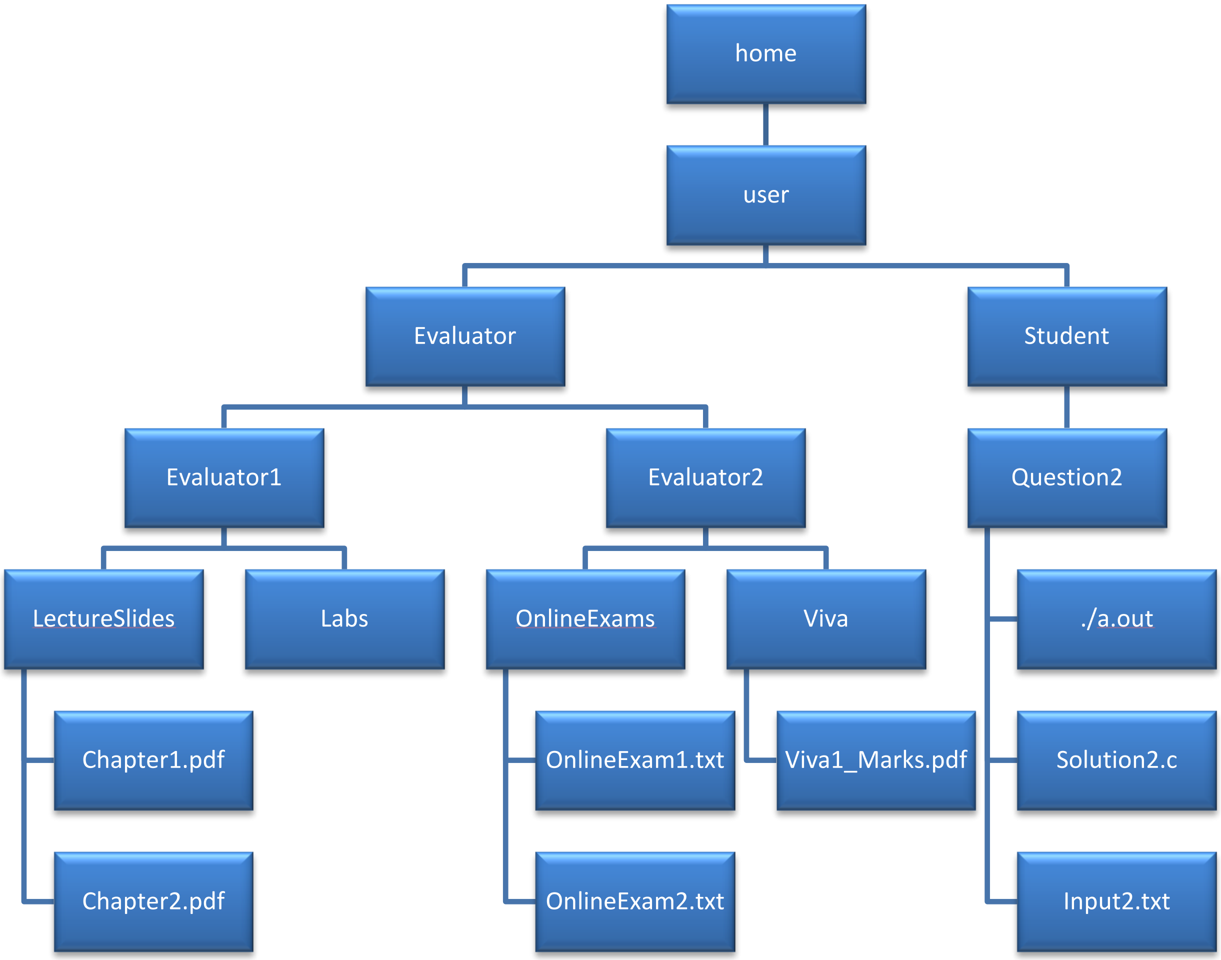

Figure 2 : Sample directory hierarchy with groups evaluator and student

Your code must also delete all the temporary files that have been created in the above procedure.

Please be realistic while giving yourself marks for Test 1 so that "no user" will get suspicious. If you have taken help from anyone, please be fair enough.

Please remove all unnecessary printfs, so that "no user" will get suspicious and you will end up safely with your marks.