National University of Computer and Emerging Sciences, Lahore Campus

Course:	Operating Systems	Course Code:	CS 2006
Program:	BS (CS)	**Semester:**	Spring 2023
Due Date	12-April-2023 at 11:59 pm	Total Marks:	40 marks
Type:	Assignment 2	**Page(s):**	2

Important Instructions:

1. Submit the solution in a zipped file named as your roll number., i.e., 21L-1111.zip
2. **You are not allowed to copy solutions from other students. We will check your code for plagiarism using plagiarism checkers. If any sort of cheating is found, heavy penalties will be given to all students involved.**
3. Late submission of your solution is not allowed.

Question: Custom Shell [30+10= 40 Marks]

Task 1:

This assignment is a continuation of your first assignment. In this assignment you must extend the shell that you have created in the assignment # 1. Now your shell should support command line pipes and FIFOs as well as input/Output and error redirection.

Example:

$ mkfifo fifo1

$ man ls 1> ls.dat

$ cat fifo1 | grep ls | wc -l

$ sort ls.dat | tee fifo1 | wc -l

The command can have any number of | symbols, such as

command1 | command 2 | command3 > file.txt

Task 2:

The next task is to implement a history feature that allows the user to access the most recently entered commands. The user will be able to access up to 10 commands by using the features. The commands will be consecutively numbered starting at 1, and the numbering will continue past 10. For Example, if the user has entered 35 commands, the 10 most recent commands will be numbered 26 to 35.

The user will be able to list the command history by entering the command.

$ history

As an example, assume that the history consists of the commands (from most to least recent):

ps, ls -l, top, cal, who, date

The command history will output:

6: ps

5: ls -l

4: top

3: cal

2: who

1: date

Your program should support two techniques for retrieving commands from the command history:

1. When the user enters !!, the most recent command in the history is executed.
2. When the user enters a single! followed by an integer N, the Nth command in the history is executed.

If user enters !!, the ps command will be performed, if the user enters !4 the top command will be executed. Any command executed in this fashion should be echoed on the user's screen. The command should also be placed in the history buffer as the next command.

The program should also manage basic error handling. If there are no commands in the history, entering. !! should results in a message "No commands in history." If there is no command corresponding to the number entered with the single! the program should output "No such command in history."

GOOD LUCK!