Heritage Institute of Technology
Kolkata – 700107

Academic Year: 2023-2024,Odd Semester

Sub: Operating Systems Lab
Subject Code: INFO3152
Branch: Information Technology
3rd Year 5th Semester

ASSIGNMENTS:

Shells

1. Write a shell script to calculate the sum of digits of any number entered through keyboard.

2. Write a program that displays the result of division of one integer by another and informs if the user tries to divide an integer by zero.

3. Write a shell program that takes a number from user and prints the reverse of the number.

4. Write a shell script to check whether a given number is prime or not.

5. Write a shell script to determine whether two numbers input through keyboard are prime to each other.

6. Write a shell script to show the maximum of three numbers provided as command line arguments.

7. Write a shell script to read date and temperature of last seven says and show the average temperature of the last seven days.

8. Ramesh's basic salary (BASIC) is input through keyboard. His dearness allowance (DA) is 52% of BASIC. House rent allowance (HRA) is 15% BASIC. Contributory provident fund is 12% of (BASIC + DA). Write a shell script to calculate his gross salary and take home salary using the following formula:
   Gross salary = BASIC + DA + HRA + (BASIC + DA) * 0.12
   Take home salary = BASIC + DA + HRA - (BASIC + DA) * 0.12

9. Write a shell script to calculate the grade of students from average marks, where the marks will be taken from another file. Determine the grade as per the following rule:

90 – 100	***O***
80 – 89	***E***
70 – 79	***A***
60 – 69	***B***
50 – 59	***C***
49 – 40	***F***

10. Write a shell script that produces a shell calculator to perform the following operations:
    - Addition
    - Subtraction
    - Multiplication
    - Division

11. Write a shell script to sort a list of elements using bubble sort technique.

12. (a) How to print any string so that the cursor will stay at the same line?

12. (b) Write a shell script to print the following pattern for any number of lines:

```
        *
      * * *
    * * * * *
  * * * * * * *
* * * * * * * * *
```

13. Write a shell script which counts the number of consonants and vowels in a given sentence.

14. Accept a string from the terminal and echo a suitable message if it doesn't have at least ten characters.

15. Write a shell script to drop the lines which are matched with a given word.

16. How will you quickly move to the fifth word of a line and replace its four characters with something else.

17. Write a shell script which deletes all lines containing the word *UNIX* in the files supplied as arguments to this shell script.

18. Write a sequence to display the list of users as well as the number of users connected to the system.

19. Write the shell script to check the entered user is a valid user or not.

20. Write a shell script to print last twenty commands issued by the user. The user name is supplied as a command line argument to the script (use bash-history file).

21. Write a shell script, which gets executed the moment a user logs in. It should display the message "GOOD MORNING" or "GOOD AFTERNOON" or "GOOD EVENING" depending upon the time at which the user logs in.

22. Write a shell program, which displays the message" welcome" and prints the date when you login to your system.

23. Write a shell script, which reports names and sizes of all files in a directory (directory should be supplied as an argument to the shell script) which size exceeds 500 bytes. The filenames should be printed in decreasing order of their sizes. The total number of such files should also be reported.

24. Write a shell script, which displays a list of all files in the current directory to which you have read, write and executes permissions.

25. Write a shell command that accept a filename as argument and displays the last modification time, if the file exists and a suitable message if it does not.

26. Write a shell script which finds out the following:
a. the last modification time of a file
b. whether the command line input string is a valid user or not

27. Write a shell script to display the files created or updated fourteen days before from the current date.

28. Develop a shell script which displays all files with all attributes those have created or modified in the month of November.

29. Write a shell script that shows the names of all the non-directory files in the current directory and calculates the sum of the size of them.

30. Write a shell script to list the name of files under the current directory started with vowels.

31. Write a shell script that accepts two directory names bar1 and bar2 as arguments and deletes those files in bar2 which are identical to their names in bar1.

32. Write a shell script which reads a directory name and compares the files in the current directory which has more files and how much more files.

33. Write a shell script to check the entered file is a blank file or not. If not found blank then display the contents of the file.

34. Write a shell script to find the total number of words, characters, lines in the given file (supplied as command line argument) and check if it is a regular file or not.

35. Write a shell script to concatenate two files and count the number of characters, number of words and number of lines in the resultant concatenated file.

36. Write a shell script to take the two filename as an input and if they are not duplicate file then concatenate them otherwise delete the second one.

37. Write a shell script, which will receive either the filename or the filename with its full path during execution. This script should print information bout the file as giving by `ls -l` command and display it in an informative manner.

38. The file `/etc/password` contains information about all the users. However, it is difficult to decipher the information stored in it. Write a shell script which would receive the log name during execution, obtain information about it from `/etc/password` and display this information on screen in easily understandable format.

39. Write a shell script to make a password-based menu-driven program, which will give three chances to enter the password in case of wrong password. If the given password is correct then the program will show the

a) Number of users currently logged in.
b) Calendar of current month.
c) Date in the format: dd / mm / yyyy.
d) Exit

40. Device a menu-driven shell program which accepts values 1 to 4 and performs actions depending upon the number keyed in:

1. List of files
2. Present date
3. Users of the system
4. Quit to UNIX.

Processes:

41. Write a program to get the PID of parent and child process.

42. Write a program to kill all processes whose PID is even.

43. Implement an orphan process using `fork`.

44. Implement a zombie process using `fork`.

45. Write a program with a local variable and a global variable. Initialize both of them. The program should `fork` a child process and the child should increment both the variables by one. After this operation, both the parent and the child should print the values of the variables.

46. Implement the assignment no. 47 using `vfork` for spawning the child.

47. Write a program to create a process which will run as a background process for fifty seconds and at the time of execution it will print the system information.

48. Display the process in the system every thirty seconds but five times.

49. Write a program for a process which cannot be killed by pressing `Ctrl + c` and again restore the default status of it. (Print necessary messages where required).

50. Write a program that creates two child processes. Each of the child process prints numbers from 1 to 10. Each time a child prints a number it also prints its own PID and parent PID. The parent waits for both of its child to finish execution and prints “Good Bye” before exiting.

51. Write a program that creates three child processes. The first child process executes the command “`who`”, the second child process executes the command “`ls -l`” and the third child process executes the command “`date`”. The parent process waits for all the child processes to finish and prints the termination status of the child.

Signals

52. Write a program which takes a value of delay as command line argument and creates a child process. The parent process waits for the child process to finish its job up to the supplied delay value. If the child terminates within the delay the parent prints the termination status and PID of the child process. On not receiving from the child it kills the child process forcefully.

53. What is a signal? Show appropriate programs that demonstrate the following signals:
a. SIGINIT
b. SIGHUP
c. SIGCHLD

54. Write a program to create a child process and send a SIGCHLD signal to the parent process.

55. Write a C program to Kill the child process from parent and print the signal number after receiving acknowledgement.

56. Write a program to get an interrupt from machine and display that value of that signal.

57. Process A and Process B normally sleep, except when process A receives signal `SIGUSR1` and process B receives signal `SIGUSR2`, when both the processes print the message “I am awake”. Write a program to incorporate it.

Semaphores

58. Write a program to implement the producer-consumer problem’s bounded buffer solution using Semaphore.

59. Write a program implementing two separate threads, one keeps printing 1-5 and halts for one second & the other repeatedly print A,B,C,D,E and for one second. Both share the same output terminal. So the same resource should be shared within a critical section. Implement the solution using Semaphore.

Threads

60. Implement the multithread concept with a program

61. Write a multi-threaded program where the main thread gets an integer number range from the user and then creates two child threads; one thread calculates the sum of all numbers in the range and prints it, and the second thread finds prime numbers in the range and prints them. The child thread must terminate by returning a value. The parent thread must wait for the child threads to finish, and it must also print the return values of the child threads.

62. Write a program to print the odd and even numbers in a range parallelly using two different Threads. The range should start from 0 and the end of the range should be taken from the user input and passed to the threads as argument.

Inter process Communication – Pipes

63. Write a program to send a message through pipe.

64. Write a program to communicate between two processes using pipes.

65. Write a code in C to implement how a process communicates with another process using the `pipe` function.

66. Write a code in C to implement how a process communicates with another process using the `popen` function.

67. Write a program that creates a one-way pipe between a parent and child process. The parent process gets a string from standard input and sends the string to the child. The child prints the reverse of the string. Both parent and child terminate when the string “quit” is input.

68. Write a program that creates a two-way pipe between a parent and child process. The parent process gets an integer from standard input and sends the integer and sends the result to the parent process, which then prints it. Both the parent and child terminate when the number 0 is input.

69. Write a program that creates one-way pipe between a parent and three child processes. The parent process gets an integer number range from standard input and divides the entire range to three child processes. Each of the child after receiving the sub-range, searches for prime numbers in that range and prints them.