GUJARAT TECHNOLOGICAL UNIVERSITY

MASTERS IN COMPUTER APPLICATION (Integrated MCA)

Year – II (Semester – IV) (W.E.F. January 2015)

Subject Name: Operating Systems

Subject Code: 4440603

1. Compute the addition of array elements and calculate average using shared memory. Average will be calculated using barrier. Array is declared as shared resource.

2. Producer-Consumer problem:
   1. Implement producer, consumer problem using bounded buffer where array is shared. When producer will put the element in the array, consumer will wait until producer will release the array and same operation is applicable for consumer. Implement mutual conclusion using semaphore. When array is full, producer will wait until consumer will consume at least one element. When buffer is empty, consumer will wait until producer will produce at least one element.
   2. Implement producer, consumer problem using un-bounded buffer where array is shared. When producer will put the element in the array, consumer will wait until producer will release the array and same operation is applicable for consumer. Implement mutual conclusion using semaphore. When buffer is empty, consumer will wait until producer will produce at least one element.

3. Reader-Writer problem:

   More than one Reader can read from a file, but, when one writer will write, no other writer and readers can operate on the file. When one writer will write, whole file is locked by that writer.

   Implement these features using mutual exclusion.

4. There is an array of elements of int. Generate histogram for calculating the occurrence of number of values within given range. Implement mutual exclusion and barrier in this application.

5. Dinning philosophers' problem"
   1. There are five philosophers sitting in rounded dining table and five fork are used for eating spaggati.
      When one philosopher is wishing to eat, he should check if both forks are free or not, otherwise he should
      wait for getting two forks.
   2. There is a room. Within room, only four philosophers are only allowed. After entering, they will seat on
      dining table. If forks are available, they will finish their dinner and then come out from room. Entrance in
      room and getting fork are controlled by semaphore.

6. Check the output of the following commands.
   date, ls, who, cal, ps, wc, cat, uname, pwd, mkdir, rmdir, cd, cp, rm, mv, diff, chmod, grep,

sed, head, tail,
cut, paste, sort, find.

7. Write shell script
    1. Accept numbers and perform addition, subtraction, division and multiplication.
    2. Accept the string and checks whether the string is palindrome or not.
    3. Accept number and check the number is even or odd, finds the length of the number, sum of the digits in the number.
    4. Accept strings and replace a string by another string.
    5. Accept filename and displays last modification time if file exists, otherwise display appropriate message.
    6. Fetch the data from a file and display data into another file in reverse order.

8. Write a script to find the global complete path for any file.

9. Write a script to broadcast a message to a specified user or a group of users logged on any terminal.

10. Write a script to copy the file system from two directories to a new directory in such a way that only the latest file is copied in case there are common files in both the directories.

11. Write a script to compare identically named files in two different directories and if they are same, copy one of them in a third directory.

12. Write a script to delete zero sized files from a given directory (and all its sub-directories).

13. Write a script to display the name of those files (in the given directory) which are having multiple links.

14. Write a script to display the name of all executable files in the given directory.

15. Write a script to display the date, time and a welcome message (like Good Morning should be displayed with “a.m.” or “p.m.” and not in 24 hours notation.

16. Write a script to display the directory in the descending order of the size of each file.

17. Write a script to implement the following commands:
    Tree (of DOS)
    which (of UNIX)

18. Write a script to make following file and directory management operations menu based:
    Display current directory

    List directory

    Make directory

    Change directory

    Copy a file

    Rename a file

Delete a file

Edit a file

19. Write a script which reads a text file and output the following
    Count of character, words and lines.
    File in reverse.
    Frequency of particular word in the file.
    Lower case letter in place of upper case letter.

20. Write a shell script to check whether the named user is currently logged in or not.

21. Write A Script To Perform Following String Operations Using Menu:
    1. COMPARE TWO STRINGS.
    2. JOIN TWO STRINGS.
    3. FIND THE LENGTH OF A GIVEN STRING.
    4. OCCURRENCE OF CHARACTER AND WORDS
    5. REVERSE THE STRING.

22. Write a script to check whether a given string is palindrome or not.

23. Write a script to check whether a given number is palindrome or not.22. Write a script to display all words of a file in ascending order.

24. Write a script to display all lines of a file in ascending order.

25. Write a script to display the last modified file.

26. Write a shell script to add the statement #include <stdio.h> at the beginning of every C source file in current directory containing printf and fprintf.

27. Write a script that behaves both in interactive and non-interactive mode. When no arguments are supplied, it picks up each C program from current directory and lists the first 10 lines. It then prompts for deletion of the file. If the user supplies arguments with the script, then it works on those files only.

28. Write a script that deletes all leading and trailing spaces in all lines in a file. Also remove blank lines from a file. Locate lines containing only printf but not fprintf.