

GUJARAT TECHNOLOGICAL UNIVERSITY

MASTER OF COMPUTER APPLICATION

SEMESTER: III

Subject Name: **Programming Skills-V (OS/SS)**

Subject Code: **630007**

A. List of Practical Related to Operating System:

1. 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.
2. Write shell script
 - a) Accept numbers and perform addition, subtraction, division and multiplication.
 - b) Accept the string and checks whether the string is palindrome or not.
 - c) Accept number and check the number is even or odd, finds the length of the number, sum of the digits in the number.
 - d) Accept strings and replace a string by another string.
 - e) Accept filename and displays last modification time if file exists, otherwise display appropriate message.
 - f) Fetch the data from a file and display data into another file in reverse order.
3. Write a script to find the global complete path for any file.
4. Write a script to broadcast a message to a specified user or a group of users logged on any terminal.
5. 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.
6. 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.
7. Write a script to delete zero sized files from a given directory (and all its sub-directories).
8. Write a script to display the name of those files (in the given directory) which are having multiple links.
9. Write a script to display the name of all executable files in the given directory.
10. Write a script to display the date, time and a welcome message (like Good Morning etc.). The time should be displayed with "a.m." or "p.m." and not in 24 hours notation.
11. Write a script to display the directory in the descending order of the size of each file.
12. Write a script to implement the following commands:
Tree (of DOS)
which (of UNIX)
13. Write a script for generating a mark sheet after reading data from a file.
File contains student roll no, name, marks of three subjects.

14. 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

15. 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.

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

17. Write a Script for Simple Database Management System Operation.
Database File Contains Following Fields.
EMP_NO
EMP_NAME
EMP_ADDRESS
EMP_AGE
EMP_GENDER
EMP_DESIGNATION
EMP_BASIC_SALARY

Provide Menu Driven Facility For

VIEW RECORD BASED ON QUERY
ADD RECORD
DELETE RECORD
MODIFY RECORD.
COUNT TOTAL NUMBER OF RECORDS
EXIT

18. Write A Script To Perform Following String Operations Using Menu:
COMPARE TWO STRINGS.
JOIN TWO STRINGS.
FIND THE LENGTH OF A GIVEN STRING.
OCCURRENCE OF CHARACTER AND WORDS
E. REVERSE THE STRING.

19. Write a script to calculate gross salary for any number of employees
Gross Salary =Basic + HRA + DA.
HRA=10% and DA= 15%.

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

21. 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.
 23. Write a script to display all lines of a file in ascending order.
 24. Write a script to display the last modified file.
 25. 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`.
 26. 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.
 27. 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`.
-

B. List of Practical Related to System Software:

1. Write a program to identify whether a given number (both integer and floating point) is valid or not.
2. Write a program to check whether an entered identifier/variable is valid or not, according to the C syntax.
3. Write a program to get the following string pattern:
 - a. $(abc)^+$
 - b. $a^+b^+c^+$
 - c. $(alaab)^*b$
 - d. $(alb)^*abb$
 - e. $a(alab)^*$Similar type of expressions can be given. Use the LEX tool for doing the same. Find out how both the methods differ?
4. Write a DFA that accepts an employee code of the form:
 $\langle \text{section code} \rangle ::= l | \langle \text{section code} \rangle l$
 $\langle \text{numeric code} \rangle ::= d | \langle \text{numeric code} \rangle d$
 $\langle \text{employee code} \rangle ::= \langle \text{section code} \rangle \langle \text{numeric code} \rangle$
where l is a letter (a to z or A to Z) and d is a digit (0 to 9)
5. Write a C/C++ program to find integer, real number, identifiers and reserved words from a given C program. Write a LEX program for the same.
6. Write a Lexical Analyzer using Lex or Flex utility of UNIX for following:
 - a. A lexer to print out all numbers from a given file. (Hint: By default lex reads standard input.)
 - b. A lexer which adds line numbers to the given file and display the same onto the standard output.
 - c. A lexer which attempt to extract only comments from a C program and display the same on standard output.
 - d. A lexer which replaces all the occurrence of "rama" with "RAMA" and "sita" with "SITA".
 - e. A lexer which classifiers tokens as words, numbers or "other".

- f. This lexer prints only words followed by punctuation. If the following sentence was the input from standard input:
“ I was here”, they said.
But were they? I cannot tell.
It will print the words here, said, they, and tell. It will not print the punctuation ; only the words.

7. Write a program to eliminate left recursion from grammar.
8. Write a program to perform left factoring on a grammar.
9. Implement naïve top down parsing.
10. Implement naïve bottom up parsing.
11. Write a program that will find the FIRST SET of the grammar.
12. Write a program that will find the FOLLOW SET of the grammar.
13. Construct an operator precedence matrix for the operators of a grammar for expressions containing arithmetic, relational and Boolean operators.
14. Write a program to implement Operator Precedence Parser.
15. Write a program to implement Recursive Descent Parser.
$$E ::= T\{+T\}^*$$
$$T ::= V\{*V\}^*$$
$$V ::= \langle id \rangle$$
16. Write a program to implement LL(1) parser:
$$E ::= T E'$$
$$E' ::= +T E' \mid \epsilon$$
$$T ::= F T'$$
$$T' ::= * F T' \mid \epsilon$$
$$F ::= \langle id \rangle$$
17. Write a program to implement Pass – I of an assembler.
18. Write a program to implement Pass – II of an assembler.
19. Write a program that takes an assembly code with macro call as input and gives an assembly code with macros expanded as output.
20. Consider a macro and generate the following data structures:
 - a. PNT
 - b. MNT
 - c. MDT
 - d. APT
 - e. EVT