Reg. No.									

Manipal Institute of Technology
MANIPAL

Constituent Institute of Manipal University

VI SEMESTER B.TECH
COMPUTER SCIENCE & ENGG. MAKEUP EXAMINATIONS, JUNE 2017
SUBJECT : LINUX BASICS AND PROGRAMMING(CSE 3286)
(OPEN ELECTIVE)
REVISED CREDIT SYSTEM
DATE: 24-06-2017

TIME:03 HOURS MAX.MARKS : 50

Instructions to Candidates:

- Answer **ALL** questions.
- Missing data, if any, may be suitably assumed.

1A. Explain Linux Operating System architecture with a neat diagram. 3M

1B. Explain the following commands. 3M

i. date

ii. printf

1C. Consider the following directory structure in a system. 4M

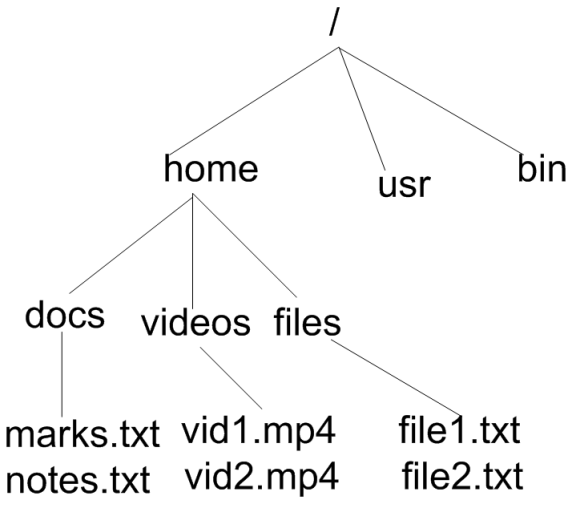

Figure : Q1C

Assume the user is in /home. Write commands for

i. Displaying the current directory.
ii. Changing the current directory to *videos*.
iii. Creating a new directory named *HindiFilms* under *videos*.
iv. Creating two files named *lyrics1.txt* and *lyrics2.txt* under *HindiFilms*.
v. Changing the current directory to / using relative pathname.
vi. Changing the current directory to */home* using absolute pathname.
vii. List all files and folders in */home* in columnar format
viii. remove the directory named *HindiFilms*

2A. Consider the directory structure as in Qn 1(C). Assume the user is in */home* folder. Write commands for each of the following. 3M

i. Copy all files under *files* folder to *docs* folder.
ii. Copy all files under *docs* folder to *videos* folder interactively.
iii. Move all files under *videos* folder to *files* folder.
iv. Remove all files under *videos* folder beginning with *v*.
v. Remove all files under *videos* folder ending with *.mp4*
vi. Count the number of lines, words and characters in the file *file1.txt*

2B. Consider the directory structure as in Qn. 1.(C). Write commands for each of the following. 3M

i. To compress the file *vid1.mp4* under */home/videos* using gzip.
ii. To determine the size in bytes of the compressed file.
iii. To recursively compress all files and folders under *home* folder using gzip.
iv. To create an archive of files under *files* folder using tar.
v. To display the content of the archive in the previous question
vi. To uncompress the file created in the previous to previous question

2C. Write commands for each of the following. 4M

i. Write the command to know inode number of a file named *abc.txt*
ii. Using relative permissions add write,execute permission to a user for a file named *abc.txt*.
iii. Using absolute permissions, set read,write permission to *user*, read,execute permission to group and read permission to others on a file name *abc.txt*.
iv. Change ownership to *userA*, group ownership to *member* to a file named *abc.txt*.
v. Create a hard link named *abcl* to a file *abc.txt*.
vi. Change the modification time of a file named *abc.txt* to 20th April, 2017 6 AM.
vii. Search and display all files under *root* directory.
viii. Search and delete all files beginning with *v* under current directory.

3A. Consider the following C++ program for reversing an input string. Please note that numbering at left are only for indication and should not be considered as part of program. 3M

```
1 #include<iostream>
2 #include<string.h>
3
4 using namespace std;
5
6 int main() {
7   int i;
8   char temp;
9   char str[10];
10  cout<<"Enter a string";
11  cin>>str;
12
13  //The following code for string reversal
14  for(i=0;i<=strlen(str)/2;i++) {
15    temp = str[i];
16    str[i]=str[strlen(str)-1-i];
17    str[strlen(str)-1-i]=temp;
18  }
19  cout<<"String reversed is "<<str;
20  return 0;
21 }
```

i. Write the command for compiling the above program for debugging in gdb.

ii. When run on gdb the above program would halt. Write the command to know where the above program would halt and details of stack trace.

iii. Write the command in gdb for setting breakpoint at line 14.

iv. Write the gdb command to display value of variable *i* automatically on reaching breakpoint.

v. Write the command to display information of all breakpoints and displays.

vi. Indicate the output when a string "helo" is entered. Provide correction to the program.

3B. Explain the mechanism of child process creation, running a program in the created child process, waiting for the child to finish execution. Differentiate between orphaned process and zombie process. 4M

3C. Assume there is a file *emp.txt* with 10 lines. The file consists of employee id, name separated with "|" symbol. Write commands for the following. 3M

i. To display the file *emp.txt* paginated with header "Employee Details"

ii. To display the first 5 lines of *emp.txt* file.

iii. To display the last 3 lines of *emp.txt* file.

iv. To copy the first 8 lines of *emp.txt* to a file employees and to display the lines at the same time.

v. To display the first 10 characters of each line in *emp.txt*.

vi. To display the 2nd column in the file *emp.txt*.

4A. Explain the following commands. 3M

i. paste command

ii. sort command

4B. Write commands for the following. 4M

i. Search for a string *soccer* in a file *game.txt*.

ii. Search for a string *tendulkar* in two files named *cricket1.txt* and *cricket2.txt* as a single command.

iii. Search for a string *rahul* ignoring case in a file *cricket.txt*.

iv. Search in a file named *cricket.txt* not containing string *devilliers*.

v. Count the number of times a pattern *rahul* exists in a file *cricket.txt*.

vi. Search in all files ending with *.txt* and display the file names in which string *ganguli* is found.

vii. Search for a string starting with a *r* at the beginning of a line in a file *cricket.txt*

viii. Search for a string ending with a *i* at the end of a line in a file *cricket.txt*.

4C. Read two numbers and a file name from the user. Check if the first number is less than or equal to second. If yes, continue, otherwise display an error message. Check if the file exists and is readable. If no, display an error message. Using sed command display the lines numbered from the first number till the second number in the file. Write a shell script for this. 3M

5A. Explain case statement after writing its syntax. Read a number from the user. Long list the files and directories if the input number is 1. If the number is 2 display today's date, if the number is 3 display all processes and exit if the number is 4. Display an error message if the number is not any one of them. Demonstrate this using a case statement. 3M

5B. Write a shell script to accept a command line argument. The argument indicates a file name. If the number of arguments is lesser than one, read the file name from the user. Check whether the file exists. If the file does not exist, terminate the shell script with a non-zero return code. If the file exists, check whether file is readable, if so, display appropriate message. If not, check whether writable, if so display appropriate message. After this read a choice from the user. If the user enters 1 repeat reading another file name and checking. If not terminate the script with a zero exit status. 4M

5C. Write a shell script to display the content of each file in the current directory using for statement. 3M