USN □□□□□□□□□□

CMRIT
CMR INSTITUTE OF TECHNOLOGY, BENGALURU.
ACCREDITED WITH A+ GRADE BY NAAC
CELEBRATING 25 YEARS

Internal Assessment Test 1 – Oct 2023

Sub:	UNIX PROGRAMMING				Sub Code:	18CS56	Branch:	CSE
Date:	07/11/2022	Duration:	90 mins	Max Marks: 50	Sem / Sec:	V SEM / A, B, C		OBE

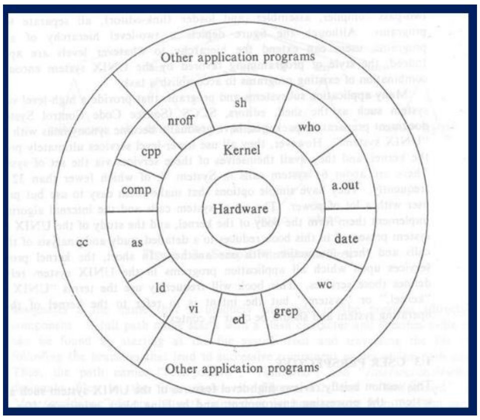
Answer any FIVE FULL Questions — MARKS | CO | RBT

1 (a) With a neat diagram, explain the UNIX architecture along with the role played by kernel and shell in sharing the workload. [07] CO1 L1

Diagram -------- 3 M

Explanation on each layer ---------- 4M

SOLUTION



The UNIX Architecture is divided as two components:

- Kernel
- Shell

Kernel: The Kernel interacts with machine hardware. The Shell interacts with the user. The Kernel is the core of Operating system. Loaded into memory when system is booted. Communicates directly with the hardware. User programs uses kernel services to access the hardware.

Working of Kernel: kernel works through “System Calls”. Manages the system memory. Schedules the processes. Decides the priorities of the processes etc.

Shell: provides the user an interface to Operating System. Shell is an environment in which user can run commands, programs and scripts. Different types of shells are available.

Unix supports two types of Shell:

1. Bourne shell (Bash) : $ is the default prompt
- C shell : % is the default prompt

(b) Explain the printf command along with syntax and example. Convert the decimal number 192 to octal and hexadecimal number using printf command. [03] CO1 L2

Definition syntax & Example ---------- 2 M

Conversion ---------- 1M

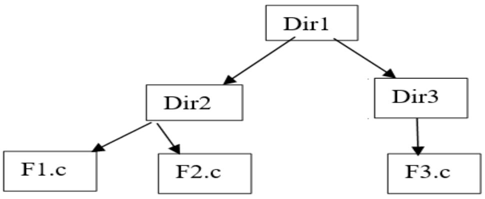
SOLUTION

Printf- an alternative to echo

Used to print messages on the terminal.

Ex1: printf “Hello World”
Ex2: printf “Hello \n World”

	Format specifiers: %s - Strings %30s - same as strings but 30 spaces wide %d - Decimal Integer %o - Octal Integer %x - Hexadecimal Integer %f - Floating point numbers printf "The number 192 in octal is %o and in hexadecimal is %x" 192 192			
2 (a)	Define Internal and External commands. How the shell acts when the given command is internal and external. Mention the command to find the type (Internal/External) of the given command. **Definition ----------- 2M** **Shell behavior --------------- 2M** **Type command ------------ 2 M** **SOLUTION** Most commands in UNIX are external in nature as they are stored in some location. Ex: who, date, cal, ls etc., Built-in commands are said to be Internal Commands. Ex: echo, printf For the internal commands, the shell will not look in to the location (PATH). For external commands, the shell will look in to the location (PATH) to find the executables for these commands - Type command is used to find whether a given command is internal or external. Ex: type echo o/p: echo is a shell builtin Ex: type ls o/p: /bin/sh	[06]	CO1	L2
(b)	Consider yourself in the terminal as User1 (Which is not root), which command will you use to change to the root access. And mention the pre-requisites to have access to the root user. **Su command ------------- 2 M** **Prerequisites ------------ 2M** **SOLUTION** **Su command is used to change to the root access.** • su stands for switch user. • User can switch to any other user, provided the password. • Ex1: su Password: ******** Ex2: Su user • Changes to the specified user environment with the help of password. • su command runs a separate subshell. User will come to parent shell by pressing ctrl +d or exit.	[04]	CO1	L3

3 (a)	With the help of commands to create directory and file, create the following hierarchy. And represent the creation with appropriate sequence of commands.  **mkdir ---------2M** **cd ------------- 2M** **touch/cat ----------- 2M** **SOLUTION** mkdir Dir1 cd Dir1 mkdir Dir2 Dir3 cd Dir2 touch F1.c touch F2.c cd .. cd Dir3 touch F3.c cd .. cd ..	[06]	CO1	L3

(b)	From the file Hierarchy give in Q3(a), to the following changes. 1. Create a copy of F3.c from Dir3 to Dir2. 2. Move F1.c of Dir1 to Dir3 3. Create a new file F4.c in Dir3. **cp ---------------- 2M** **mv --------------- 1M** **Touch/cat ------- 1M** **SOLUTION** cp /Dir3/F3.c /Dir2/F3.c mv /Dir1/F1.c Dir3 cd Dir3 touch F4.c	[04]	CO1	L3
4 (a)	Define the following commands with syntax, 3 options and example along with the output: • cp 2. date 3. Rm **cp along with option ----------- 2M** **date along with option --------- 2M** **rm along with options -----------2M** **SOLUTION** **cp : copying a file** cp command copies a file or a group of files. Creates an exact image of the file with different name on the disk. The syntax requires at least two files name as arguments.When both are ordinary files, contents of file1 are copied to file2.	[06]	CO1	L1

Ex1: cp file1 file2

If the file do not exist, it will be created. Else the file contents will be overwritten. The destination can also be a directory.

File is copied in to the directory as follows:

Ex2: cp file1 Dir1

Multiple files can be copied to a directory as follows.

Ex3: cp file1 file2 file3 Dir2

Options

- -i: Interactive copying
- -R: copying directory structure.

Date: Displaying the system date.

The UNIX system maintains an internal Clock. date command is used to display the current date and time to the nearest second.

```
Ex 1: date
O/p 1: Sat Oct  8 12:05:13 PM UTC 2022
```

The date command can be used with suitable format specifiers as arguments. Each format is preceded with a + symbol and %operator.

Format specifiers:

- %m - month (Number format)
- %h - name of the month (Aug, Oct)
- %d - the day of the month (1 t0 31)
- %y - the last two digits of the year
- %H, M, S - the hour, minutes and seconds
- %D - the date in the format mm/dd/yy
- %T - Time in the format of hh:mm:ss

```
Ex 1:   # date +%m O/p 1:     10
Ex 2:   # date +%h O/p 2:     Oct
Ex 3:   # date +%d O/p 3:     08
Ex 4:   # date +%y O/p 4:     22
```

rm: Deleting Files: The command deletes one or more files.

Ex1: rm file1

Ex2: rm file1 file2 file3

Options

-i: interactive Deletion

- -y removes the file. Any other option leaves the file unchanged.

-r: recursive deletion (-R)

- Recursively deletes the files in a directory along with directory.

Ex3: rm -r *

-f : Force Removal

- Prompts for removal if a file is write protected.

Ex4: rm -rf *

(b) Differentiate cp and mv command. Explain the changes that happens to the file if both commands are used.

[04] CO1 L3

Cp --------- 1M

Mv --------- 1M

Differentiation ------------ 2M

SOLUTION

mv: Renaming Files

Move command has two distinct functions: Renames a file or directory. Moves group of files to different directory.

```
Ex1: mv  one.txt one.c
```

If the destination file does not exist, it will be created. If the file is existing , it overwrites the contents of the destination file. Group of files can be moved to a directory as follows:

```
Ex2: mv file1 file2 file3 Dir1
```

cp : copying a file

cp command copies a file or a group of files. Creates an exact image of the file with different name on the disk. The syntax requires at least two files name as arguments. When both are ordinary files, contents of file1 are copied to file2.

```
Ex1: cp file1 file2
```

If the file do not exist, it will be created. Else the file contents will be overwritten. The destination can also be a directory.

mv command renames the file and removes the source argument. Cp command creates a new file with the same contents and also retains the source file.

5 (a) When you interact with shell using the command ls -l, it displays multiple columned output. Explain those in detail. [07] CO1 L3

Ls -l ------------ 1M

Explanation of each fields in detail is 0.5 (12*0.5 = 6 M)

SOLUTION

ls: Listing the files and its attributes

Lists the names of files in a particular UNIX directory. ls without parameters displays the files in the directories.

```
Ex 1: # ls
o/p 1: bench.py  hello.c
```

Options for ls command

-a : displays all files including the hidden files

-l : long listing files in the directory.

```
Ex: ls -l
O/p:
total 8
-rw-r--r-- 1 root root 114 Dec 26  2020 bench.py
-rw-r--r-- 1 root root 185 Sep  9  2018 hello.c
```

Displays most attributes of the files such as permissions, file size, ownership etc. -l option stands for listing. ls -l command has seven attributes as shown below:

The list is preceded with the words **total 8,** that indicates the total blocks occupied by these files.

```
total 8
-rw-r--r-- 1 root root 114 Dec 26  2020 bench.py
-rw-r--r-- 1 root root 185 Sep  9  2018 hello.c
```

File Types and Permissions (1st Column)

Shows the type and permission associated with each file. The first character - indicates an ordinary file, if its d , that indicates a directory.

Three types of permissions can be given to a file.

- R : Read permission
- W : Write Permission
- X : Execute Permission

Links (2nd Column)

	Indicates the number of links associated with each file. UNIX lets a file have as many as names the user want even though there is a single file on the disk. A link count having more than one indicates that the file has more than one name. **Ownership (3rd Column)** When user creates a file, he automatically becomes the owner. The owner has full authority to tamper the file. This privilege is not accessible to other users except root. **Group Ownership (4th Column)** When opening a user account, the system administrator also assigns the user to some group. It represents the group owner of that file. **File Size (5th Column)** Shows the size of the file in bytes. Represents the character count of the file. **Last Modification Time (6,7,8th Columns)** Indicates the last modification time of the file. A file is modified only if the contents are changed. If permissions are changed, the time of the files remain unchanged. If file is less than a year old, then year will not be displayed. **File name (9th Column)** Displays the file name in alphabetical order. UNIX filenames can be up to 255 characters.			
(b)	Differentiate Relative and Absolute pathnames. **Definition ----------- 1M** **Differentiation ------ 2M**	[03]	CO1	L1
6 (a)	Consider a file [File1.txt] with following permissions for users, group and others: -rwx r-x r-x Write the commands to change the permissions as follows using relative and absolute permissions: 1. Provide write access to Others. 2. Remove execute access to User. **Chmod o+w filename ---------- 1M** **Chmod 755 filename ---------- 1M** **Chmod u-x filename ---------- 1M** **Chmod 655 filename ---------- 1M** **SOLUTION** A file or directory is created with a default set of permissions. The default setting write protects a file from all, except the user. The **chmod - change mode** command is used to set the permissions for one or more files. Permissions for all users can be set using **chmod** command. Chmod command can be run only by the user and the root. The command can be used in two ways: 1. Relative Manner 2. Absolute Manner Chmod o+w filename Chmod 755 filename Chmod u-x filename Chmod 655 filename	[05]	CO2	L3
(b)	Explain features of UNIX. **Explanation of any five Features of UNIX ---------------- 5M** 1. **Multi- tasking:** Unix is a multi-tasking operating system. It has the ability to support concurrent execution of 2 or more active process (instance of program in execution is called process). 2. **Multi-user operating system:** Unix is multi-user operating system. It has ability to support more than one user to login into the system simultaneously and execute programs. Difference: the diiference between multi –tasking and multi-user system	[05]	CO1	L1

	is, In multi-tasking, different tasks the process running concurrently belongs to one user. In multi-user environment different tasks belong to diff users. 3. **Portable:** Unix operating system is highly portable. Compared to other os, it is very easy to port.unix on to different hardware platforms with minimal or no modifications at all i.e., because it ways developed in c language which is high level language 4**. Inter machine communication:** The development of communication protocols like tcp/ip has made possible by unix operating system to usera to exchange information in the form of email and shared data. 5. **Security:** As Unix is multi-user system, there is every chance that a user may intrude into a another users area either unintentionally. But, Unix offers solid security at various levels beginning from the system startup level to accessing files as well as saving data in an encrypted form. 6. **Documentation:** Unix has good library of utilited and commands that have been used to develop newer applications. 7. **File and directory system:** One of the very important key feature of any unix system is that allows users to organize and maintain these files/directories easily and maintain these files/directories easily and efficiency.			

CO PO Mapping

Course Outcomes		Modules covered	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
CO1	Explain Unix Architecture, File system and use of Basic Commands	**M1**	3	2	3	2	2	-	0	0	0	0	0	0	0	0	0	0
CO2	Illustrate Shell Programming and to write Shell Scripts	**M2**	3	2	3	2	2	0	0	0	0	0	0	0	0	0	1	0
CO3	Categorize, compare and make use of Unix System Calls	**M3M4 M5**	3	2	3	2	2	0	0	0	0	0	0	0	0	2	0	0
CO4	Build an application/service over a Unix system.	**M1, M2, M3, M4, M5**	3	2	3	2	2	0	0	0	0	0	0	0	0	2	2	0

COGNITIVE LEVEL	REVISED BLOOMS TAXONOMY KEYWORDS
L1	List, define, tell, describe, identify, show, label, collect, examine, tabulate, quote, name, who, when, where, etc.
L2	summarize, describe, interpret, contrast, predict, associate, distinguish, estimate, differentiate, discuss, extend
L3	Apply, demonstrate, calculate, complete, illustrate, show, solve, examine, modify, relate, change, classify, experiment, discover.
L4	Analyze, separate, order, explain, connect, classify, arrange, divide, compare, select, explain, infer.

L5	Assess, decide, rank, grade, test, measure, recommend, convince, select, judge, explain, discriminate, support, conclude, compare, summarize.

PROGRAM OUTCOMES (PO), PROGRAM SPECIFIC OUTCOMES (PSO)				CORRELATION LEVELS	
PO1	Engineering knowledge	PO7	Environment and sustainability	0	No Correlation
PO2	Problem analysis	PO8	Ethics	1	Slight/Low
PO3	Design/development of solutions	PO9	Individual and team work	2	Moderate/ Medium
PO4	Conduct investigations of complex problems	PO10	Communication	3	Substantial/ High
PO5	Modern tool usage	PO11	Project management and finance		
PO6	The Engineer and society	PO12	Life-long learning		
PSO1	Develop applications using different stacks of web and programming technologies				
PSO2	Design and develop secure, parallel, distributed, networked, and digital systems				
PSO3	Apply software engineering methods to design, develop, test and manage software systems.				
PSO4	Develop intelligent applications for business and industry				