**CDAC MUMBAI**

**Concepts of Operating System**

**Assignment 2**

**Part A**

**What will the following commands do?**

1. **echo "Hello, World!"** = echo command just prints the string “Hello, World”. It ts command to show the output which insert by user in the console.
2. **name="Productive"** = name is simple variable name and “productive” is string which assign to the variable name.
3. **touch file.txt** = touch is command to create a file in Linux. file.txt is file name with extension .txt which created by using touch command.
4. **ls -a** = ls command is used to display list of the files and directories. ls with option a is used to show the hidden files.
5. **rm file.txt** = rm command is used to remove the file from directory. Remove file named as file.txt using rm command
6. **cp file1.txt file2.txt** = cp command uses for copy the content of file1.txt to file2.txt.
7. **mv file.txt /path/to/directory/** = mv command is used to move the file from one location to another location. In this mv command move the file file.txt to /path/to/directory/ this specified location.
8. **chmod 755 script.sh** =
9. **grep "pattern" file.txt** = grep command is used to search the specific word, text, line, file from the file or directory. This command displays a word pattern if it will present in file.txt file
10. **kill PID** = kill command is used to send a signal to a process.
11. **mkdir mydir && cd mydir && touch file.txt && echo "Hello, World!" > file.txt && cat file.txt =**

using mkdir command created a mydir directory.

Using cd command change the current directory to mydir.

Using touch command create a file.txt file inside the mydir directory.

Using echo command prints the string Hello world in the file.txt file.

Using cat command all content in file.txt file will display in the terminal.

1. **ls -l | grep ".txt" =** command use for search the specific word “.txt” and list out those files with their size, modified date and time, name of owner and its permission.
2. **cat file1.txt file2.txt | sort | uniq =** command is concatenate the two files named as file1.txt and file2.txt sort command is used sort the content of both the files and uniq command is used for removes the duplicate lines which are common in both the files.
3. **ls -l | grep "^d" =** ls -l list all the files and directories in the current directory with all the details and grep “^d” only display the lines start with d.
4. **grep -r "pattern" /path/to/directory/** = this command shows the pattern word in recursively through all the directory and subdirectory.
5. **cat file1.txt file2.txt | sort | uniq –d** = here in this code cat command can concatenate the two files file1.txt and file2.txt sord command the sorting all content of file1 and file2. Uniq -d command display the duplicate lines which will be present in the content of file1.txt and f2.txt.
6. **chmod 644 file.txt =**
7. **cp -r source \_directory destination\_directory** = in this command copy the directories, fills and subdirectories recursively from source directory to destination directory.
8. **find /path/to/search -name "\*.txt"** = this command used to search the files with a .txt extension within the /path/to/search directory.
9. **chmod u+x file.txt =** chmod command used to change the permission. In this command u means owner has permission to execute means x a program in file.txt.
10. **echo $PATH =** The PATH variable contains a list of directories that the shell searches through to find executable files for commands

**Part B**

**Identify True or False:**

1. **ls is used to list files and directories in a directory** = true.
2. **mv is used to move files and directories.** =true
3. **cd is used to copy files and directories**. = false
4. **pwd stands for "print working directory" and displays the current directory.** = true.
5. **grep is used to search for patterns in files**. = true
6. **chmod 755 file.txt gives read, write, and execute permissions to the owner, and read and execute permissions to group and others.** = true
7. **mkdir -p directory1/directory2 creates nested directories, creating directory2 inside directory1 if directory1 does not exist**. = true
8. **. rm -rf file.txt deletes a file forcefully without confirmation**. = true

**Identify the Incorrect Commands:**

1. **chmodx is used to change file permissions**. = This is incorrect command for change the file permission. For change the file permission chmod command will used.
2. **cpy is used to copy files and directories**=. incorrect command cp command will used to copy the files and directory.
3. **mkfile is used to create a new file.** = incorrect command to create a new file touch or nano command will be used.
4. **catx is used to concatenate files** = incorrect command to concatenate the file cat command is used followed by file name1 file name2.
5. **rn is used to rename files**. = there are mv used to rename a file. Mv command also used to move the file and directory.

**Part C**

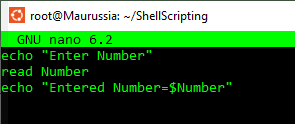
Question 1: Write a shell script that prints "Hello, World!" to the terminal.

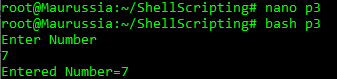


Question 2: Declare a variable named "name" and assign the value "CDAC Mumbai" to it. Print the value of the variable.

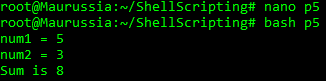
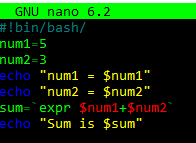


Question 3: Write a shell script that takes a number as input from the user and prints it.

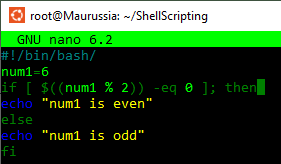




Question 4: Write a shell script that performs addition of two numbers (e.g., 5 and 3) and prints the result.

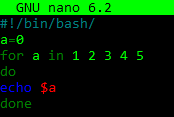


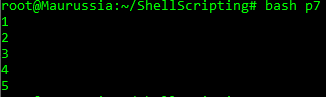
Question 5: Write a shell script that takes a number as input and prints "Even" if it is even, otherwise prints "Odd".



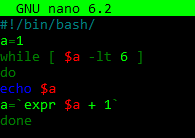


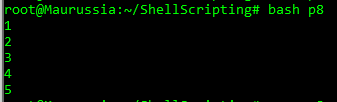
Question 6: Write a shell script that uses a for loop to print numbers from 1 to 5.



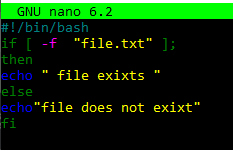


Question 7: Write a shell script that uses a while loop to print numbers from 1 to 5.



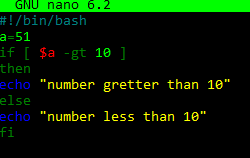


Question 8: Write a shell script that checks if a file named "file.txt" exists in the current directory. If it does, print "File exists", otherwise, print "File does not exist"



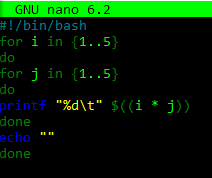


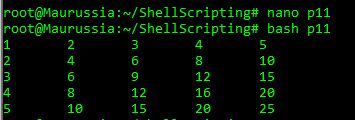
Question 9: Write a shell script that uses the if statement to check if a number is greater than 10 and prints a message accordingly.



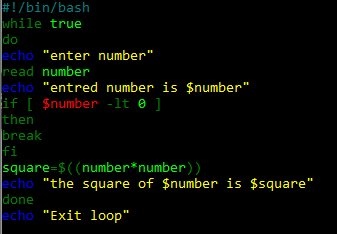


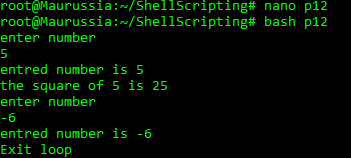
Question 10: Write a shell script that uses nested for loops to print a multiplication table for numbers from 1 to 5. The output should be formatted nicely, with each row representing a number and each column representing the multiplication result for that number.





Question 11: Write a shell script that uses a while loop to read numbers from the user until the user enters a negative number. For each positive number entered, print its square. Use the break statement to exit the loop when a negative number is entered.





Part E

1. Consider the following processes with arrival times and burst times:

| Process | Arrival Time | Burst Time |

|---------|--------------|------------|

| P1 | 0 | 5 |

| P2 | 1 | 3 |

| P3 | 2 | 6 |

Calculate the average waiting time using First-Come, First-Served (FCFS) scheduling.

|  |  |  |  |
| --- | --- | --- | --- |
| Process | Arrival Time | Brust Time | Waiting time |
| P1 | 0 | 5 | 0 |
| P2 | 1 | 3 | 4 |
| P3 | 2 | 6 | 6 |

P1 P2 P3

0 5 8 14

Arrival Waiting Time:-3.3

1. Consider the following processes with arrival times and burst times:

| Process | Arrival Time | Burst Time |

|---------|--------------|------------|

| P1 | 0 | 3 |

| P2 | 1 | 5 |

| P3 | 2 | 1 |

| P4 | 3 | 4 |

Calculate the average turnaround time using Shortest Job First (SJF) scheduling.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Process | Arrival | Brust | Waiting | Completion | ATA |
| P1 | 0 | 3 | 0 | 3 | 3 |
| P2 | 1 | 5 | 7 | 13 | 12 |
| P3 | 2 | 1 | 1 | 4 | 2 |
| P4 | 3 | 4 | 1 | 8 | 5 |

p1 p2 p3 p4

0 8 4 8 13

Average turn around time = 5.5

3. Consider the following processes with arrival times, burst times, and priorities (lower number indicates higher priority):

| Process | Arrival Time | Burst Time | Priority |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

| P1 | 0 | 6 | 3 |

| P2 | 1 | 4 | 1 |

| P3 | 2 | 7 | 4 |

| P4 | 3 | 2 | 2 |

Calculate the average waiting time using Priority Scheduling.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Process | Arrival time | Brust time | Priority | Waiting time |
| P1 | 0 | 6 | 3 | 6 |
| P2 | 1 | 4 | 1 | 0 |
| P3 | 2 | 7 | 4 | 10 |
| P4 | 3 | 2 | 2 | 2 |

P1 p2 p4 p1 p3

0 1 5 7 12 19

Avearage waiting Time:4.5

1. Consider the following processes with arrival times and burst times, and the time quantum for Round Robin scheduling is 2 units:

| Process | Arrival Time | Burst Time

|---------|--------------|------------

| P1 | 0 | 4 |

| P2 | 1 | 5 |

| P3 | 2 | 2 |

| P4 | 3 | 3 |

Calculate the average turnaround time using Round Robin scheduling.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Process | Arrival Time | Brust Time | Waiting Time | Complition Time | TAT |
| P1 | 0 | 4 | 6 | 10 | 10 |
| P2 | 1 | 5 | 9 | 15 | 14 |
| P3 | 2 | 2 | 2 | 6 | 4 |
| P4 | 3 | 3 | 7 | 13 | 10 |

P1 p2 p3 p4 p1 p2 p4pr p2pr

0 2 4 6 8 10 12 14 16

Avearage Turn Around Time=9.5

1. Consider a program that uses the fork() system call to create a child process. Initially, the parent process has a variable x with a value of 5. After forking, both the parent and child processes increment the value of x by 1. What will be the final values of x in the parent and child processes after the fork() call?

X=5

After fork()

Parent will be a 6 as well as child is 6