|  |
| --- |
| Experiment No.3 |
| Write shell scripts programming. |
| Date of Performance: 16/02/2024 |
| Date of Submission: 23/02/2024 |

**Aim:** Write Shell Scripts to do the following:

1. Perform the basic arithmetic operations

2. Display top 10 processes in Ascending order.

3. Display processes with highest memory usage.

4. Display current logged in user and log name.

5. Display current shell, home directory, kernel version.

**Objective:** The shell is the operating system's command-line interface (CLI) and interpreter for the set of commands that are used to communicate with the system. A shell script is usually created for command sequences in which a user has a need to use repeatedly in order to save time.

**Theory:**

Shell is a user program or its environment is provided for user interaction. It is a command prompt within Linux where you can type commands. It is a program that takes your commands from the keyboard and gives them to the OS to perform. Shell is not part of system KERNAL but it uses system KERNAL to execute programs, create files,etc. A Shell Script is a text file that contains a sequence of commands for a UNIX based OS. It is called a Shell Script because it combines into a "Script" in a single file a sequence of commands, that would otherwise have to be presented to the system from a keyboard one at a time. A Shell Script is usually created for command sequences for which a user has a repeated need. You initiate the sequence of commands in Shell Script by simply entering the name of the Shell Script on a command line.

**Types of Shell Script :-**

1. **sh** - Simple Shell
2. **bash** - Bourne Again Shell
3. **ksh** - Korne Shell
4. **csh** - C Shell
5. **ssh** - Secure Shell

To use a particular Shell type the Shell name at the command prompt. Eg:- $csh - It will switch the current Shell to C Shell. To view the current Shell that is being used, type echo $ SHELL at the command prompt.

**Program:**

echo "Arithematic Operations: "

num1=10

num2=20

sum=$(($num1+$num2))

echo "The sum of $num1 and $num2 is $sum"

diff=$(($num1-$num2))

echo "The difference of $num1 and $num2 is $diff"

divide=$(($num1/$num2))

echo "The division of $num1 and $num2 is $divide"

multiply=$(($num1\*$num2))

echo "The multiplication of $num1 and $num2 is $multiply"

modulus=$(($num1%$num2))

echo "The modulus of $num1 and $num2 is $modulus"

echo "Conditional Statements: "

if [ sum%2==0 ]

then

echo "The sum of $num1 and $num2 i.e $sum is even."

else

echo "The sum of $num1 and $num3 i.e $sum is odd."

fi

echo "Top 10 process based on Memory Usage: "

top -b -o -%MEM | head -n 10

echo "Top 10 process based on CPU utilization:"

top -b -o -%CPU | head -n 10

echo "Top 10 process based on their PID's: "

top -b -o -PID | head -n 10

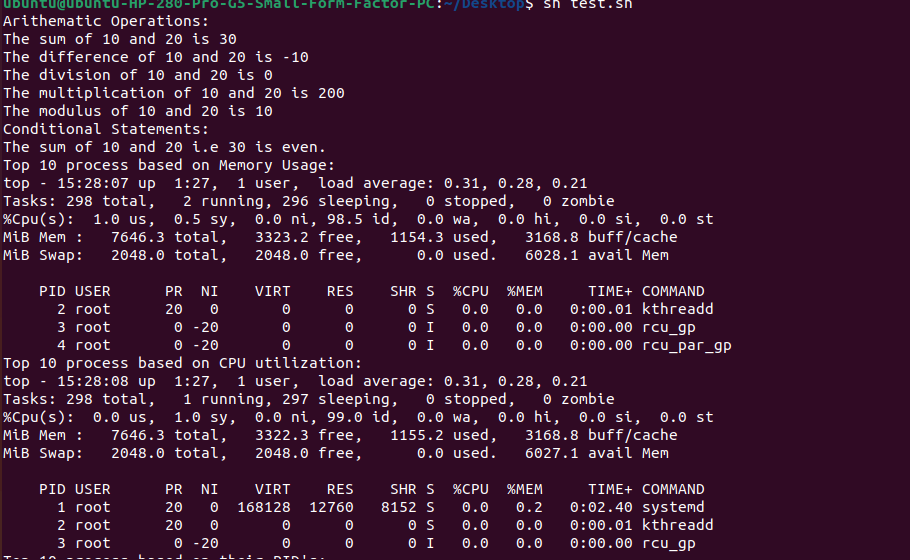
echo "The current logged in user is $(whoami)"

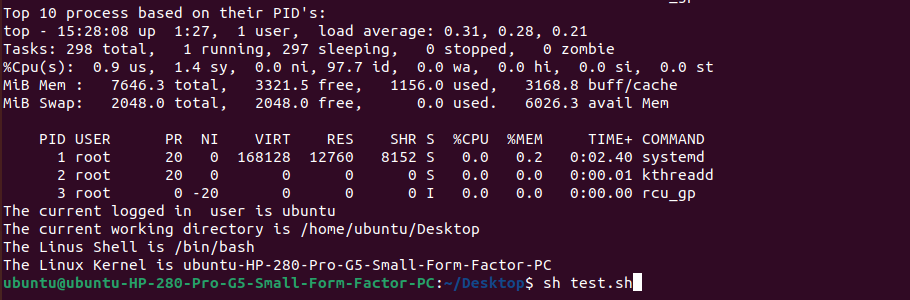
echo "The current working directory is $(pwd)"

echo "The Linus Shell is $SHELL"

echo "The Linux Kernel is $(uname -n)"

**Result:**





**Conclusion:**

Shell scripting in Linux provides automation, enabling users to streamline repetitive tasks and save time. It offers flexibility for customization, allowing scripts to be tailored to specific needs and workflows. With batch processing capabilities, multiple commands can be executed sequentially, enhancing efficiency. Seamless integration with Linux commands empowers users to leverage the full potential of the command line interface for increased productivity.