|  |  |  |  |
| --- | --- | --- | --- |
| **Fr. Conceicao Rodrigues College of Engineering Department of Computer Engineering** | | | |
| **Student’s Roll No** | **9603** | **Students Name** | **Zane Falcao** |
| **Date of Performance** |  | **SE Computer – Div** | **A** |

**Aim:** To study basics of Shell Scripting

**Lab Outcome:**

**CSL403.1: Demonstrate basic Operating system Commands, Shell scripts, System Calls and API wrt Linux.**

**Problem Statements:**

1.WAP that accepts user name and reports if user logged in.

echo "Enter username:"

read a

who > userlist

if grep $a userlist

then

echo "user logged in"

else

echo "user not logged in"

fi

OUTPUT:

zane@DESKTOP-LJHRUE9:~$ vim exp2.1.sh

zane@DESKTOP-LJHRUE9:~$ chmod +x exp2.1.sh

zane@DESKTOP-LJHRUE9:~$ ./exp2.1.sh

Enter username:

zane

user logged in

2.WAP that takes a filename as input and checks if it is executable, if not make it executable.

echo "Enter a filename: "

read filename

if [ -x "$filename" ]; then

echo "$filename is already executable."

else

chmod +x "$filename"

echo "$filename is now executable."

fi

OUTPUT:

zane@DESKTOP-LJHRUE9:~$ vim exp2.2.sh

zane@DESKTOP-LJHRUE9:~$ chmod +x exp2.2.sh

zane@DESKTOP-LJHRUE9:~$ ./exp2.2.sh

Enter a filename:

bb.txt

bb.txt is now executable

3.WAP to take string as command line argument and reverse it.

input\_string=$1

reversed\_string=`echo $input\_string | rev`

echo $reversed\_string

OUTPUT:

zane@DESKTOP-LJHRUE9:~$ vim exp2.3.sh

zane@DESKTOP-LJHRUE9:~$ chmod +x exp2.3.sh

zane@DESKTOP-LJHRUE9:~$ ./exp2.3.sh zane

atrib

4.Write a shell script to find the sum, the average and the product of the four integers entered

echo "Enter four numbers:"

read num1 num2 num3 num4

sum=$((num1 + num2 + num3 + num4))

average=$((sum / 4))

product=$((num1 \* num2 \* num3 \* num4))

echo "The sum of $num1, $num2, $num3, $num4 is $sum"

echo "The average of $num1, $num2, $num3, $num4 is $average"

echo "The product of $num1, $num2, $num3, $num4 is $product”

OUTPUT:

zane@DESKTOP-LJHRUE9:~$ vim exp2.4.sh

zane@DESKTOP-LJHRUE9:~$ chmod +x exp2.4.sh

zane@DESKTOP-LJHRUE9:~$ ./exp2.4.sh

Enter four numbers: 3 5 7 1

The sum of 3,5,7,1 is 16

The average of 3,5,7,1 is 4

The product of 3,5,7,1 is 105

5. Write a shell script to find out the unique words in a file and also count the occurrence of each of these words. We can say that the file under consideration contains many lines, and each line has multiple words.

echo "Enter file name"

read file

awk '

{

for (i = 1; i <= NF; i++){

count[$i]++

}

}

END {

for (word in count)

{ print word, count[word]

}

}' $file | sort -k2 -nr

OUTPUT

Enter file: src.txt

the 1

system’s 1

pager 1

manual 1

man 1

is 1

6.WAP which displays the following menu and executes the option selected by user:

1.ls 2. pwd 3. ls –l 4. ps -fe

echo "Enter an option:"

echo "1. ls"

echo "2. pwd"

echo "3. ls -l"

echo "4. ps -fe"

read option

case $option in

1) ls;;

2) pwd;;

3) ls -l;;

4) ps -fe;;

OUTPUT:

zane@DESKTOP-LJHRUE9:~$ vim exp2.6.sh

zane@DESKTOP-LJHRUE9:~$ chmod +x exp2.6.sh

zane@DESKTOP-LJHRUE9:~$ ./exp2.6.sh

Enter an option:

1. ls
2. pwd
3. ls -l
4. ps -fe

1

zane.txt exp2.1.sh exp2.2.sh exp2.3.sh exp.2.4.sh exp2.5.sh exp2.6.sh userlist

7.WAP that prompts user for a starting value & counts down from there.

echo "Enter the starting value: "

read value

while [ $value -gt 0 ]; do

echo $value

value=$((value - 1))

done

OUTPUT:

zane@DESKTOP-LJHRUE9:~$ vim exp2.7.sh

zane@DESKTOP-LJHRUE9:~$ chmod +x exp2.7.sh

zane@DESKTOP-LJHRUE9:~$ ./exp2.7.sh

Enter the starting value:

3

3

2

1

8.Create a data file called employee in the format given below:

EmpCode  Character, EmpName   Character , Grade  Character , Years of experience Numeric , Basic Pay   Numeric

* + Sort the file on EmpName.
  + Sort the file on

Decreasing order of basic pay

Increasing order of years of experience.

Display the number of employees whose details are included in the file.

Display all records with ‘smith’ a part of the employee name.

Display all records with EmpName starting with ‘B’.

echo "100,John James,A,5,25000

200,Sam Smith,B,7,30000

300,Zane ,C,4,20000

400,Ava Black,A,3,22000

500,Bridget Jones,B,6,28000

600,Sarah Green,C,5,24000" > employee

sort -k2 employee > sorted\_employee

sort -k5,5nr -k4,4n sorted\_employee > sorted\_employee2

echo "Number of employees: $(wc -l < employee)"

grep 'Smith' employee

grep '^[^,]\*,[B]' employee

OUTPUT:

zane@DESKTOP-LJHRUE9:~$ vim exp2.8.sh

zane@DESKTOP-LJHRUE9:~$ chmod +x exp2.8.sh

zane@DESKTOP-LJHRUE9:~$ ./exp2.8.sh

200,Sam Smith,B,7,30000

500, Bridget Jones,B,6,28000

300, Zane,C,4,20000

|  |  |  |  |
| --- | --- | --- | --- |
| **On time Submission(2)** | **Knowledge of Topic(4)** | **Implementation and Demonstration (4)** | **Total (10)** |
|  |  |  |  |
| **Signature of Faculty** |  | **Date of Submission** |  |