

Particulars of the Experiments Performed

CONTENTS

Expt No.	Date	Experiment	Marks Obtained	Page No.
1	25/10/21	Write the shell script to find if the given year is leap or not		1-2
2	25/10/21	Write the shell script to find the area of a circle		3-4
3	25/10/21	Write the shell script to check whether the number is zero / positive / negative		5-6
4	25/10/21	Write the shell script to find the biggest of three numbers		7-8
5	8/11/21	Write the shell script to find the factorial of a number		9-10
6	8/11/21	Write the shell script to compute the gross salary of an employee		11-12
7	8/11/21	Write the shell script to convert the Fahrenheit to Celsius		13-14
8	8/11/21	Write the shell script to perform arithmetic operations on given two numbers		15-16
9	15/11/21	Write the shell script to find sum of even numbers upto n		17-18
10	15/11/21	Write the shell script to print to print the combinations of numbers 123		19-20
11	15/11/21	Write a shell script to find the power of a number		21-22
12	15/11/21	Write a shell script to find the sum of n natural numbers		23-24

Particulars of the Experiments Performed

CONTENTS

Expt. No. 1

Date 25/5/12

Page No. 2

Output

1) Enter the year:

2016

Leap Year

2) Enter the year:

1800

Not a leap year

Q) Write the shell script to find if the given year is leap or not.

Lab Program 1

```
#!/bin/sh
echo "Enter the year :"
read year
val1 = `expr $year % 400`
val2 = `expr $year % 100`
val3 = `expr $year % 4`
if [ $val1 -eq 0 ] ; then
    echo "Leap Year"
elif [ $val2 -eq 0 ] ; then
    echo "Not a Leap Year"
elif [ $val3 -eq 0 ] ; then
    echo "Leap Year"
else
    echo "Not a Leap Year"
fi
```

Teacher's Signature: _____

Expt. No. 2

Date 25/10/21

Page No. 4

Output

Enter the radius of Circle:

12

452.39

Q2 Write the shell script to find the area of a circle.

#!/bin/sh

echo "Enter the radius:"

read r

area = `echo "scale=2; 3.14 * \$r * \$r" | bc`

echo "\$area"

Notepad

Teacher's Signature :

Output

1) Enter a number =

12

Positive Numbers

2) Enter a number =

-8

Negative Numbers

3) Enter a number =

0

Zero Number

Q3) Write the shell script to check whether the number is zero
positive / negative

#!/bin/bash

echo "Enter a number = "

read n

if [\$n -gt 0]; then

else "Positive numbers"

elif [\$n -lt 0]; then

else "Negative numbers"

else "Zero numbers"

fi

N/A

25/02/21

Output

1) Enter 3 numbers:

4
5
6

6 is largest

2) Enter 3 numbers =

10
13
12

13 is largest

3) Enter 3 numbers =

20

1

20 is largest

Lab Program

Q4. Write the shell script to find the largest of these numbers.

```
#!/bin/sh
else "Gives 3 numbers!"
```

```
read a
```

```
read b
```

```
read c
```

```
if [ $a -gt $b ] && [ $a -gt $c ] ; then
```

```
else "$a is largest"
```

```
elif [ $b -gt $a ] && [ $b -gt $c ] ; then
```

```
else "$b is largest"
```

```
else
```

```
else "$c is largest"
```

~~if~~

~~fi~~

~~else~~

Output

Enter a number:

5

Factorial is 120

Q. Write the shell script to find the factorial of a number.

Lab Program 5

```
#!/bin/sh
echo "Enter a number: "
read n
a=1
while [ $n -gt 0 ]
do
    a=$((a*n))
    n=$((n-1))
done
```

echo "Factorial is \$a"

N

8

Output

Enter the basic salary:

12000

Basic salary = 12000

DA = 1200

HSA = 2400

The gross salary of the employee is 15600

Q1: Write the shell script to compute the gross salary of an employee.

#!/bin/sh

else "Enter the basic salary:"

read basic

DA = 'expr \$basic \(* 10 / 100)'

HSA = 'expr \$basic \(* 20 / 100)'

gross_salary = 'expr \$basic + \$DA + \$HSA'

echo "Basic salary = \$basic"

echo "DA = \$DA"

echo "HSA = \$HSA"

else "The gross salary of the employee is \$gross_salary"

8/11/21

Date 8/11/21

Expt. No. 7

Page No. 14

Output

Enter the temperature in fahrenheit :

123

The temperature in celsius is 50.05

Write a shell script to convert a fahrenheit to celsius

```
#!bin/bash  
else "Gives the temperature in fahrenheit : "
```

```
read f
```

```
scale = $(echo "scale=2;($f-32)*(5/9)" | bc )  
echo "The temperature in celsius is $scale"
```

123

50.05

Teacher's Signature :

Output

Enters two numbers:

21

55

Addition = 76

Subtraction = -34

Multiplication = 1155

Division = .3818

Q1. Write one shell script to perform arithmetic operations on given two numbers.

#!/bin/bash

else "Enters two numbers : "

read a

read b

vala='expr \$a + \$b'

valb='expr \$a - \$b'

valc='expr \$a * \$b'

vald='expr "scale=4; \$a / \$b" | bc)

else "Addition = \$vala"

else "Subtraction = \$valb"

else "Multiplication = \$valc"

else "Division = \$vald"

Output

Enters a number :
4
Sum of even = 6

Expt. No. 9

Lab Programming

```
#!/bin/bash
echo "Enters a numbers : "
read n
sum=0
i=0
val='expr $i % 2'
while [ $val -eq 0 ]
do
    if [ $val -eq 0 ]
    then
        sum=$((sum+i))
    fi
    i=$((i+2))
done
echo "Sum of even = $sum"
```

OutPut

111
112
113
121
122
123
131
132
133
211
212
213
221
222
223
231
232
233
311
312
313
321
322
323
331
332
333

Expt. No. — 10 —

Date — 15/11/21 —
Page No. — 20 —

Lab Program 10

Ques Shell script to print the combinations of numbers 123.

#!/bin/bash

for i in 123

do

for j in 123

do

for k in 123

do

echo \$i \$j \$k

done

Teacher's Signature : _____

Expt. No. 11

Date 15/11/21

Page No. 22

Ques

Enter a number :

2

Enter its power :

3

Answer = 8

Lab Program 11

Quadratic a shell script to find the powers of a number

```
#!/bin/bash
```

```
echo "Enter a number:"
```

```
read n
```

```
echo "Enter its power:"
```

```
read a
```

```
mul=1
```

```
while [ $a -gt 0 ]
```

```
do
```

```
mul=$((mul * a))
```

```
a=$((a-1))
```

```
done
```

```
echo "Answer = $mul "
```

Teacher's Signature : _____

Expt. No. 12

Date 15/11/21

Page No. 24

Output

Enters a numbers:

6
Sum of 6 natural numbers is 21.

Q12= Write a shell script to find the sum of n natural numbers.

Lab Program 12

```
#!/bin/bash
echo "Enter a numbers:"
read n
sum=0
for ((i=1;i<=n;i=i+1))
do
    sum=$((sum+i))
done
echo "Sum of $n natural numbers is $sum."
```

N/121

Teacher's Signature :

Expt. No. 13

Date _____
Page No. 25

Output

Enter CIE and SEE marks of subject 1 :

23

You got F grade in subject 1.
Enter CIE and SEE marks of subject 2 :

11

You got F grade in subject 2.

Enter CIE and SEE marks of subject 3 :

15

You got F grade in subject 3.

Enter CIE and SEE marks of subject 4 :

16

You got C grade in subject 4.

Enter CIE and SEE marks of subject 5 :

12

You got F grade in subject 5.

Enter CIE and SEE marks of subject 6 :

50

You got F grade in subject 6.

Enter CIE and SEE marks of subject 7 :

11

You got D grade in subject 7.

Enter CIE and SEE marks of subject 8 :

13

You got D grade in subject 8.

3 subjects passed.

3 subjects failed.

Q1: Write a shell script to display the pass class of a student.
#!/bin/bash
for((i=1;i<=6;i+1))
do
echo "Enter CIE and SEE marks of a subject \$i : "
read cie
read see
marks=\$((cie+see))

case \$marks in
100) echo "You got S grade in subject \$i";;
9[0-9]) echo "You got S grade in subject \$i";;
8[0-9]) echo "You got A grade in subject \$i";;
7[0-9]) echo "You got B grade in subject \$i";;
6[0-9]) echo "You got C grade in subject \$i";;
5[0-9]) echo "You got D grade in subject \$i";;
4[0-9]) echo "You got E grade in subject \$i";;
3[0-9]) echo "You got F grade in subject \$i";;
2[0-9]) echo "You got F grade in subject \$i";;
1[0-9]) echo "You got F grade in subject \$i";;
>) echo "Not valid marks";;
esac

If [\$marks -gt 10]; then
countp=\$((countp+1))

Teacher's Signature : _____

else

countf = \$((countf + 1))

fi

done

echo "\$countp subjects passed."

echo "\$countf subjects failed."

Expt. No. — 14 —

Date _____

Output
Enter the value of n :

Lab Program 14

Page No. 27

Fibonacci series upto 5 terms :
5
0
1
1
2
3

Write a shell script to find fibonacci series upto n.

#!/bin/sh

echo "Enter the value of n : "

read n

x=0

y=1

i=2

echo "Fibonacci series upto \$n terms : "

echo "\$x"

echo "\$y"

while [\$i -le \$n]

do

i=`expr \$i + 1`

z=`expr \$x + \$y`

echo "\$z"

x=\$y

y=\$z

done

Teacher's Signature : _____

Expt. No. 15

Date _____

Lab Program 15

Page No. 28

Output

Enter a string : travel
The numbers of vowels : 8 2

```
Ques: Calculate a shell script to count the numbers of vowels of a string
#! /bin/bash
echo "Enter a string"
read string
count=0
l='expr "$string" : '\
for ((i=0;i<${#string};i++))
do
    c='expr "$string" : \'(.|\')\' '
    if [ "$c" = 'a' -o "$c" = 'e' -o "$c" = 'i' -o "$c" = 'o' -o "$c" = 'u' ]
    then
        count=$((count+1))
    fi
done
echo "The numbers of vowels : $count"
else "The numbers of vowels : $count"
```

Teacher's Signature : _____

Output

Enters the file name :
 lab5.sh
 No. of characters = 126
 No. of words = 24
 No. of lines = 10

Lab Program 16

Ques Write a shell script to check number of user words, characters in a file.

```
#!/bin/bash
echo "Enters the file name :"
read filenam
cat $filenam |wc -c
w=$cat $filenam |wc -w
l=$cat $filenam |wc -l
echo "No. of characters = $c"
echo "No. of words = $w"
echo "No. of lines = $l"
```

Date 3/11/22

No. 121

Page No. 30

卷之三

CEILL = 11: - 1 hour

SHELL = /bin/bash
SESSION_MANAGER=local/laptop:1000.ICE-unix/1537
DISPLAY=:10.0-3330-151.1537

QT_ACCESSIBILITY = 1
COLORTERM = truecolor
XDG_CONFIG_DIRS = /etc/xdg:~/.config:~/etc/xdg
XDG_MENU_PREFIX = gnome-
GNOME_DESKTOP_SESSION_ID = this-is-deprecated

LAWGAGE = en-IN : en

ENODE_BNODE_SESSION_HODDEBHOSEN

<https://www.wes/loss/recovery>

ESTATE PLANNING

SSM-AGENT-PID=1482

GTK_MODULES=gail:atk:tautola

$$\text{PwD} = \text{Phone} / \text{base}$$

LÖWENHE = borsca

THE SESSION DESKTOP = window

GPO - 2001 - 1994 - 14

XAUTHORITY=`/root/.xauthority`

WINDOWPATH = 2

Name _____

USERNAME = bmsca

卷之三

卷之三

Q17: Write a C/C++ program to read contents of its Environment list.

Lab Report 1

Teacher's Signature

`0x5=34; 42:st=37; 44:exr01; 32:*.bxz=01; 31:*.tgc=01; 31:*.axc=01; 31:*.
axj=01; 31:*.bx2=01; 31:*.lba=01; 31:*.l2h=01; 31:*.l2m=01;
31:*.tgc=01; 31:*.bx2=01; 31:*.bx=01; 31:*.tgc=01; 31:*.tgc2=01; 31:
31:*.tgc=01; 31:*.deb=01; 31:*.spu=01; 31:*.wai=01; 31:*.ear=01; 31:*.sw=0
.tgc=01; 31:.deb=01; 31:*.spu=01; 31:*.wai=01; 31:*.ear=01; 31:*.sw=0
.tgc=01; 31:.tgc=00; 36:*.nas=00; 36:*.wan=00; 36:*.oga=00;
01; 31:*.72=01; *.ogg=00; 36:*.nas=00; 36:*.wan=00; 36:*.oga=00;
36:*.opu=00; 36:*.spu=00; 36:*.spf=00; 36:`
`XDG_CURRENT_DESKTOP=ubuntu:GNOME`

`VTE_VERSION=6003`

`GNOME_Terminal_Screen=/usr/gnome/Terminal/screen/160x76x0-dbd540df-`

`850c_3da8260f2555`

`MANAGERPID=1306`

`LESSCLOSE=/usr/bin/lesspipe %s %s`

`XDG_SESSION_CLASS=user`

`TERM=xterm-256color`

`LESSOPEN=/usr/bin/lesspipe %s`

`USER=smsce`

`GNOME_TERMINAL_SERVICE=1.97`

`DISPLAY=0`

`SHLVL=1`

`QT_IM_MODULE=tbar`

`PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/
/usr/local/games:/snap/bin`

`GDMSESSION=ubuntu`

`DNS_SESSION_BUS_ADDRESS=unix:path=/run/user/1000/bus`

`=.16b17`

Ques

Usage: ./a.out [-s] <configfile> <new-link>

```

Lab Program 18

Ques: Write a C/C++ program to emulate the rmixd command.

#include <stdio.h>
#include <sys/types.h>
#include <unistd.h>
#include <string.h>

int main(int argc, char *argv[])
{
    if(argc < 3 || argc > 4 || (argc == 4 && strcmp(argv[1], "-s")))
        printf("Usage: ./a.out [-s] <configfile> <new-link>\n");
    else
    {
        if(argc == 4)
            if((symbolic(argv[2]), argv[3])) == -1)
                printf("Cannot create symbolic link\n");
        else
            printf("Symbolic link created\n");
    }
    else
    if((link(argv[1], argv[2])) == -1)
        printf("Cannot create hard link\n");
    else
        printf("Hard link created\n");
}

```

Output

System supports job control
System supports saved set-UID and saved set-GID
System supports chmod-restricted option is 0
Pathname trunc option is 1
Disable & characters for terminal files is 0
Usage: • lassut [-s] <arg-file> <new-link>

```
#define _POSIX_SOURCE
#define _POSIX_C_SOURCE 199309L
#include <unistd.h>
#include <stropts.h>
int main ()
{
    ifdef _POSIX_JOB_CONTROL
        printf("System supports job control\n");
    else
        printf("System does not support job control\n");
    ifdef _POSIX_SAVED_IDS
        printf("System supports saved set-UID and saved set-GID\n");
    else
        printf("System does not support saved set-UID and saved set-GID\n");
    ifdef _POSIX_CHMOD_RESTRICTED
        printf("chmod-restricted option is 0\n");
    else
        printf("System does not support chmod-restricted option\n");
    ifdef _POSIX_NO_TRUNC
    
```

```
printf("Pathname trunc option is %d\n", _POSIX_NO_TRUNC);
#else
printf("System does not support system-wide pathname trunc
option\n");
#endif
#ifndef _POSIX_VDISABLE
printf("Disable character for terminal files is %d\n", _POSIX_VDISABLE);
#else
printf("System does not support _POSIX_VDISABLE\n");
#endif
return 0;
}
```

Expt. No. 20

Date 3/11/22

Lab Program 20

Output

usage ./lab20 <file> [<args>]

Ques

Write a C/C++ program which demonstrates interprocess communication between a reader process and a writer process. Use `open`, `read`, `write` and `close` for I/O.

Ans Program 20

close

{

fd = open(argv[1], O_WRONLY);
write(fd, argv[2], strlen(argv[2]));

}

close(fd);

}