

Particulars of the Experiments Performed

CONTENTS

Expt No.	Date	Experiment	Marks Obtained	Page No.
1.	25/10/21	Shell script to find if the given year is leap year or not	45/11/PS	2.
2.	25/10/21	Area of circle	5/11/PS	4.
3.	25/10/21	Check whether given no. is +ve, -ve or 0.	5/11/PS	6.
4.	25/10/21	Find the biggest of 3 numbers	5/11/PS	8.
5.	8/11/21	factorial of a number	5/11/PS	10.
6.	8/11/21	Compute the gross salary of an employee	5/11/PS	12.
7.	8/11/21	Convert temperature F to C°	5/11/PS	14.
8.	8/11/21	Perform arithmetic operations on given 2 numbers	5/11/PS	16.
9.	15/11/21	sum of even numbers upto n		18.
10.	15/11/21	print the combinations of numbers 123		20.
11.	15/11/21	Power of a number		22.
12.	15/11/21	sum of n natural numbers		24.
13.	29/11/21	display the pass class of students		26.
14.	29/11/21	Fibonacci series upto n		30.

Particulars of the Experiments Performed

CONTENTS

Expt No.	Date	Experiment	Marks Obtained	Page No.
15.	29/11/21	Count the number of vowels in a file	10/12/21	32
16.	29/11/21	Check number of lines, words and characters in a file	10/12/21	34
17.	3/01/22	C/C++ program that outputs contents of its environment list	10/12/21	36
18.	3/01/22	C/C++ program to emulate Unix "In" command	10/12/21	38
19.	3/01/22	C/C++ POSIX compliant program to print POSIX defined config	10/11/8	40
20.	3/01/22	C/C++ prog to demonstrate Interprocess communication b/w a reader process and writer process	10/11/8	44
21.		other command need to know	10/11/8	48
22.		need to understand diff thing	10/11/21	50
23.		statement & go now	10/11/21	51
24.		address location of go now	10/11/21	51
25.		for loop with neglect statement	10/11/21	51
26.		N step using "switch"	10/11/21	51

Lab - 1

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

```
#!/bin/sh
```

```
echo "Enter the year"
```

```
read y
```

```
if [ `expr $y % 100` -eq 0 ]; then
```

```
if [ `expr $y % 4` -eq 0 ]; then
```

```
echo "leap year"
```

```
else
```

```
echo "Not a leap year"
```

```
fi
```

```
else
```

```
if [ `expr $y % 400` -eq 0 ]; then
```

```
echo "leap year"
```

```
else
```

```
echo "Not a leap year"
```

```
fi
```

```
fi
```

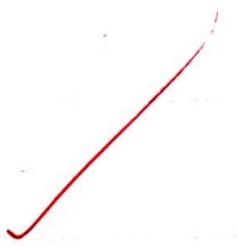
Teacher's Signature : _____

~~Output :-~~

Enter the year

2000

leap year



Lab - 2

Q. Write shell script to find the area of circle

```
#!/bin/sh
```

```
echo "Enter the radius:"
```

```
read radius
```

```
area = $(echo "3.14 * $radius * $radius / bc")
```

```
echo "Area of circle = \"$area sq. units\""
```

Teacher's Signature : _____

Output:-

Enter the radius

2 is my input at 3 both of these Node so, Area
Area of circle = 12.56 sq. units

1-2

Lab-3

Q. Write shell script to check whether given no. is positive, negative or zero.

#!/bin/sh

echo "Enter the number"

read n

if [\$n -eq 0]; then

echo "Given number is zero"

elif [\$n -lt 0]; then

echo "Given number is negative"

else

echo "Given number is positive."

fi

Teacher's Signature : _____

Output:-

Enter the number.

2

Given number is positive.

((odd | even) * even) + 4.2.5" add))
"Simple math" = "Add 20"

Lab-4

Q. Write a shell script to find biggest of 3 numbers.

```
#!/bin/sh
```

```
echo "Enter the numbers"
```

```
read a b c
```

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

```
    echo $a "A is greater"
```

```
elif [ $c -gt $a ]
```

```
then
```

```
    echo $c "C is greater"
```

```
fi
```

```
else
```

```
    if [ $c -gt $b ]
```

```
    then
```

```
        echo $c "C is greater"
```

```
    else
```

```
        echo $b "B is greater"
```

```
    fi
```

```
fi
```

Teacher's Signature : _____

Output :

Enter the numbers

1 2 3

3 is the greater number



Lab - 5

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

```
#!/bin/sh
```

```
echo "Enter the number"
```

```
read num
```

```
fact = 1
```

```
while [ $num -gt 1 ]
```

```
do
```

```
    fact=$((fact+num))
```

```
    num=$((num-1))
```

```
done
```

```
echo "The factorial of number"
```

~~```
echo $fact
```~~

Teacher's Signature : \_\_\_\_\_

wishes

• 8 days

Output :

Enter the number

5

The factorial of number

120



120

120

lab - 6

Q. Write a shell script to compute gross salary of an employee.

$\text{\#! /bin/sh}$

echo "Enter your basic salary"  
read bs.

hra = 'expr 20 \* \$bs / 100'

da = 'expr 10 + \$bs / 100'

echo "Gross salary is `expr \$bs + \$hra + \$da`"

Teacher's Signature : \_\_\_\_\_

Output:

Codes

Enter the basic salary

1200

Gross salary

1560



Explain result

(Gross salary) = 1560

(Element) = 1560

"Address of element of set"

## Lab - 7

Q. Write shell script to convert temperature farenheit to Celsius.

```
#!/bin/sh
```

```
echo "Enter the temperature in farenheit"
```

```
read tempf
```

```
tempc = $((echo "scale=2; (5/9)*($tempf-32)"/bc))
```

```
echo $tempc.
```

Teacher's Signature : \_\_\_\_\_

Output:

Enter the temperature in farenheit

68

The temperature in celsius is

28

## Lab-8

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

```
#!/bin/sh
```

```
echo "Enter two numbers"
```

```
read a b
```

```
echo "Enter your choice? (1 to 5)"
```

```
echo "1) Sum"
```

```
echo "2) Difference"
```

```
echo "3) Product"
```

```
echo "4) Quotient"
```

```
echo "5) Remainder"
```

```
echo "Enter your choice"
```

```
read n
```

```
case "$n" in
```

1) echo "The sum of \$a and \$b is `expr \$a + \$b / bc`"

2) echo "The difference of \$a and \$b is `expr \$a - \$b / bc`";;

3) echo "The product of \$a and \$b is `expr \$a \* \$b / bc`";;

4) echo "The Quotient of \$a and \$b is `expr \$a / \$b / bc`";;

5) echo "The Remainder of \$a by \$b is `expr \$a % \$b / bc`";;

```
esac
```

Teacher's Signature : \_\_\_\_\_

Output:

Enter 2 number

2 4

Enter your choice (1 to 5)

- 1) Sum
- 2) Difference
- 3) Product
- 4) Quotient
- 5) Remainder

Enter your choice

3

The product of 2 and 4 is 8.

F - 201

## Lab - 9

Q. Write shell script to find sum of even numbers upto n.

```
#!/bin/sh
```

```
echo "Enter the value of n"
```

```
read n
```

```
i=2
```

```
while [$i -le $n]
```

```
do
```

```
sum=$((sum+i))
```

~~```
i=$((i+2))
```~~
~~```
done
```~~

```
echo "Sum of even numbers upto n: "
```

```
echo $sum
```

Teacher's Signature : \_\_\_\_\_

Output:

Enter the value of  $n$

6

Sum of even numbers upto 6

12

"Even sum upto 6"

"(2 at 1) ? ends may wind"

"nude (1)"

"nudified (2)"

"nudified (3)"

"nudified (4)"

"nudified (5)"

"nudified (6)"

"nudified (7)"

"nudified (8)"

"nudified (9)"

"nudified (10)"

"nudified (11)"

"nudified (12)"

"nudified (13)"

"nudified (14)"

Expt. No. 10.

## Lab-10

Chaitanya

Q. Write shell script to print combinations of 123.

```
#!/bin/sh
```

```
echo "Combinations of 123"
```

```
for i in 123
```

```
do
```

```
for j in 123
```

```
do
```

```
for k in 123
```

```
do
```

```
/ if [$i -ne $j -ne $k -a $k -ne $i] ;
then
```

```
echo $i $j $k
```

```
fi
```

```
done
```

```
done
```

```
done
```

Teacher's Signature :

Output:

Combinations of 1 2 3

|   |   |   |
|---|---|---|
| 1 | 2 | 3 |
| 1 | 3 | 2 |
| 2 | 1 | 3 |
| 2 | 3 | 1 |
| 3 | 1 | 2 |
| 3 | 2 | 1 |

Total = 6

$(1+2+3)^2 = 9$

$(1+2)^2 = 4$

" Total steps needed to move 3 page  
is 9 "

## Lab - 11

Q. Write shell script to find power of a number

~~#!/bin/bash~~

count = 1

ans = 1

while [ ~~exact~~ \$count -le \$2 ]

do

ans = \$((ans \* \$1))

count = \$((count + 1))

done

~~/ echo "Power \$2 of number \$1 is : "~~

~~/ echo \$ans~~

Teacher's Signature :

Output:

sh lab11.sh 2 2

• ESL job scheduling using of Linux Node



"ESL job scheduling

ESL

ESL wif

ESL wif 2 wif

Linux job scheduling using of Linux Node

if if if nodes

if  
nodes

## Lab-12

Q. Write shell script to find sum of  $n$  natural numbers.

#!/bin/sh

count = 1

while [ \$count -le \$1 ]

do

ans = \$((\$ans + count))

count = \$((count + 1))

done

echo "Sum of \$1 natural no. is: \$ans"

echo \$ans

✓  
MM  
6/12/2021

Teacher's Signature :

output:

sh lab12.sh 3

sum of 3 natural numbers is :

6



## Lab-13

Q. Shell script to display the pass class of a student.

```
echo "Enter the no. of subjects : "
```

```
read n
```

```
fail = 0
```

```
pass = 0
```

```
l = 1
```

```
while [$l -le $n]
```

```
do
```

```
echo "Enter the marks for $lo subject : "
```

```
read cle
```

```
echo "Enter the SEE marks for $lo subject : "
```

```
read see
```

```
marks = $((cle + (see/2)))
```

```
if [$marks -ge 90]
```

```
then
```

```
echo "Grade is S in subject $lo"
```

```
elif [$marks -lt 90 -a $marks -ge 80]
```

```
then
```

```
echo " A in subject $lo"
```

```
elif [$marks -lt 80 -a $marks -ge 70]
```

```
then
```

```
echo " B in subject $lo"
```

```
elif [$marks -lt 70 -a $marks -ge 60]
```

```
then
```

```
echo " C in subject $lo"
```

Teacher's Signature :

Output:

Enter the no. of subjects : 3

Enter the <sup>CIE</sup> marks for subject 1 : 45

Enter the SEE marks for subject 1 : 92

Enter the CIE marks for subject 2 : 40

Enter the SEE marks for subject 2 : 90

Enter the CIE marks for subject 3 : 20

Enter the SEE marks for subject 3 : 35

Grade 8 in subject 1

Grade A in subject 2

Grade F in subject 3

No. of pass subjects = 2

No. of fail subjects = 1

`elif [ $marks -lt 60 -a $marks -ge 50 ]`

`then` ~~then \$i, \$marks, \$fail, \$pass~~ ~~number will not print~~

`echo " D in $i subject "`

`elif [ $marks -lt 50 -a $marks -ge 40 ]`

`then`

`echo " E in $i subject "`

`elif [ $marks -lt 40 ]`

`then`

`echo " Grade F in $i subject "`

`fi`

`if [ $marks -lt 40 ]`

`then`

`fail = $(($fail + 1))`

`else`

`pass = $(($pass + 1))`

`fi`

`r = $(($r + 1))`

`done`

`echo " The no. of pass subjects = $pass "`

`echo " No. of fail subjects = $fail "`

## Lab 14

Q. Shell script to find Fibonacci series upto n.

echo "Enter the no. till which Fibonacci series is needed"

read n

i = 2

a = 0

b = 1

echo "\$a"

echo "\$b"

while [ \$i -le \$n ]

do

c = \$((\$a+\$b))

a = \$b

b = \$c

echo "\$c"

i = \$((\$i+1))

done.

Teacher's Signature :

Output:

Enter the no. till which fibonacci series is needed : 6

Fibonacci series upto 6 numbers is : 1,1,2,3,5,8

For 7th number of fib series

For 8th number of fib series

For 9th number of fib series

For 10th number of fib series

$$((1+1.618))^\frac{1}{2} = 1.618$$

$$((1+1.618))^\frac{1}{2} = 2.236$$

$$((1+1))^\frac{1}{2} = 1$$

" 1.618 = Golden ratio or GP " also

" 1.618 = Golden ratio or GP " also

## Lab 15

Q. Shell script to count the number of vowels of a string

```
echo "Enter the string"
```

```
read st
```

```
len=`expr $st | wc -c`
```

```
len=`expr $len - 1`
```

```
count = 0
```

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

```
do
```

```
ch=`expr $str | cut -c $len`
```

```
case $ch in
```

[aeiouAEIOU]

```
count=`expr $count + 1`
```

```
esac
```

```
len=`expr $len - 1`
```

```
done
```

```
echo "No. of vowels : $count"
```

Output:

Enter the string : Aesopheodita

Number of vowels = 7

Ans

["d"]

((d+d))d = d

d = d

d = d

((d+i))d = i

Lab 16

Q. Shell script to check number of lines, words, characters in a file:

echo "Enter the filename"

read f

~~l = "wc -l \$f"~~

~~w = "wc -w \$f"~~

echo " no. of lines : \$l "

echo "no. of words : \$w "

echo "no. of characters : \$c "

Output:

Enter the filename: lab4prog3.sh

No. of lines = 16

No. of vowels = 54

No. of characters = 198

"pride and joy"

"I am a boy who loves to play with my dog."

"I love my dog."

"My dog is a very good boy."

[A C E I O U S Y]

(I like to play with my dog.)

"I am not a good boy."

"I am not a good boy."

- Q. Write a C/C++ program that outputs the contents of its environment list.

```
#include <stdio.h>
int main (int argc , char *argv[])
{
 int i;
 char *ptr;
 extern char **environ;
 for (ptr = environ; *ptr != 0; ptr++)
 printf ("%s\n", *ptr);
 return 0;
}
```

Output:

TMPDIR=/var/folders/b0/blwfxk556qs79s3rnf230msc000gn/T/

-- CFBundle Identifier = com.apple.Terminal

XPC\_FLAGS = 0x0

TERM=xterm-256color

SSH\_AUTH\_SOCK=/private/tmp/com.apple.launchd.OpdID06Hgz/Listeners

XPC\_SERVICE\_NAME = 0

TERM\_PROGRAM = Apple-Terminal

TERM\_PROGRAM\_VERSION = 440

TERM\_SESSION\_ID = 699CB65C-78C4-4592-9F9A-2A77F59547A8

SHELL=/bin/zsh

HOME=/Users/bhauya singh

USER = bhauya singh

PATH = /Library/Frameworks/Python.framework/Versions/3.10/bin:/usr/local/bin:/sbin:/sbin:/Library/Apple/usr/bin:/Users/bhauya singh/developer/flutter/bin

SH\_LVL = 1

PWD = /Users/bhauya singh/Desktop

OLDPWD = /Users/bhauya singh

LC\_CTYPE = UTF-8

- = /Users/bhauya singh/Desktop/.la.out

## Lab-18

Q. Write a C/C++ program to emulate unix ln 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[3],
 "-s") == 0))
 {
 printf("Usage: ./a.out [-s] <old-file> <new-link>/n");
 return 1;
 }
 if (argc == 4)
 {
 if ((symlink(argv[2], argv[3]) == -1))
 printf("Cannot create hard link /n");
 else
 printf("Hard link created/n");
 }
 return 0;
}
```

Teacher's Signature : \_\_\_\_\_

## Output:

Usage : ./a.out [-s] <org-file> <new-link>  
[root@localhost usp1]# ./a.out 1 2 3 4

Usage : ./a.out [-s] <org-file> <newlink>  
[root@localhost usp1]# ./a.out 1.c2

Hard link created

[root@localhost usp1]# ls -l

-rw-r--r-- 2 root 657 Mar 27 16:44 1a.c  
-rw-r--r-- 2 root root 657 Mar 27 16:44 2

[root@localhost usp1]# ./a.out 1a.c 2

cannot create hard link

## Lab-19

Q. Write a C/C++ POSIX compliant program that prints the POSIX defined configuration options supported on any given system using feature test macros.

```
#define _POSIX_SOURCE
#define _POSIX_C_SOURCE 199309L
#include <stdio.h>
#include <unistd.h>
int main()
{
 #if defined(_POSIX_JOB_CONTROL)
 printf("System supports job control\n");
 #else
 printf("System does not support job control\n");
 #endif
 #if defined(_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");
 #endif
 #if defined(_POSIX_CHOWN_RESTRICTED)

```

Teacher's Signature :

Output:

System supports job control

System supports saved sets - UID and saved  
get-GID

chown-restricted option is 1

Pathname trunc option is 1

Disable character for terminal files is 0

(N = 32768) ??

(L = 0, L = 0, N = 0, P = 0) ??

(L = 0, L = 0, N = 0, P = 0) ??

(L = 0, L = 0, N = 0, P = 0) ??

(0 number)

## Lab-20

- Q. Write a C/C++ program which demonstrates interprocess communication between a reader process and a writer process. Use mkfifo, open, read, write and close APIs in your program.

```
#include <sys/types.h>
#include <unistd.h>
#include <fcntl.h>
#include <sys/stat.h>
#include <string.h>
#include <errno.h>
#include <stdio.h>

Put main (int argc, char *argv[])
{
 int fd;
 char buf[256];
 if (argc != 2 && argc != 3)
 {
 printf("Usage ./s <file> [<arg>]\n", argv[0]);
 return 0;
 }
 mkfifo(argv[1], S_FIFO | S_IRWXU | S_IRWXG |
 S_IWUSR);
 if (argc == 2)
 {
 fd = open(argv[1], O_RDONLY | O_NONBLOCK);
 }
```

Teacher's Signature :

SP 18-05

P1

Output:

[root@localhost usp]# ./a.out FIFO1 "This is USP & CD Lab."

After this opens New Terminal

(\* Terminal 2 \*)

[root@localhost/]# ./a.out FIFO1

This is USP & CD Lab