



B M S. COLLEGE OF ENGINEERING (Autonomous Institution)

RECORD OF PRACTICAL WORK

NAME : RAJESTHWARI.....
SUBJECT : USP LAB.....
SEMESTER : V BRANCH : CSE
ROLL NO : USN : IBM19CS031

Particulars of the Experiments Performed CONTENTS

Expt No.	Date	Experiment	Marks Obtained	Page No.
01	25/10/21	Shell script = whether leap year or not		
2	25/10/21	Area of circle		
3	25/10/21	Positive, negative or zero		
4	25/10/21	Largest of 3 numbers		
5	8/11/21	Factorial of a number		
6	8/11/21	Gross salary		
7	8/11/21	Convert Fahrenheit to Celsius		
8	8/11/21	Arithmetic operations on 2 no		
9	15/11/21	Sum of even no upto n		
10	15/11/21	Combinations of 1, 2, 3		
11	15/11/21	Power of a number		
12	15/11/21	Sum of n natural nos		

Expt. No. 01

Output

```

chroot /bin/sh
./one.sh 2004
leap year
./one.sh 2005
Not a leap year

```

```

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

#!/bin/sh
if [ $(($1%400)) -eq 0 ]
then
echo "leap year"
elif [ $(($1%4))-eq 0 ]
then
echo "leap year"
elif [ $(($1%100))-ne 0 ]
then
echo "Not a leap year"
else
echo "Not a leap year"
fi

```

Expt. No. 02

Date 25/10/21
Page No. 02

Shell script to find the area of a circle.

Output

chmod 777 two.sh
./two.sh
Enter the radius
2
12.56 is the area of the circle

Output

clined 777 three.sh
. / three.sh 0
argument is equal to zero
. / three.sh 1
argument is positive
. / three.sh - 5
argument is negative

```
#!/bin/lsh
if 'test $1 -eq 0
then
echo "argument is equal to zero"
elif test $1 -lt 0
then
echo "argument is negative"
else
echo "argument is positive"
fi
```

shell script to check whether a number is
positive, negative or zero

Output

clined 777 four sh

1 four sh

Enter 3 numbers

10
20
30

30 is the greatest

1 four sh

Enter 3 numbers

-1
0
-4

0 is the greatest

1 four sh

Enter 3 numbers

80
9
-7

80 is the greatest

Shell script to find the biggest of 3 numbers

#!/bin/sh

echo "Enter 3 numbers"

read a

read b

read c

if [\$a -gt \$b -a \$a -gt \$c]

then

echo "\$a is the greatest"

elif [\$b -gt \$c -a \$b -gt \$a]

then

echo "\$b is the greatest"

elif [\$c -gt \$a -a \$c -gt \$b]

then

echo "\$c is the greatest"

fi

Output

```
chner 777 fact.sh
· /fact.sh
enter a number
4
Factorial of the number is : 24
· /fact.sh
enter a number
6
Factorial of the number is : 720
```

Shell script to find the factorial of a number

```
#!/bin/sh
echo "Enter a number"
read num
fact=1
for ((i=1; i<=num; i++))
do
    fact=$((fact * i))
done
echo "Factorial of the number is : $fact"
```

Expt. No. 06

Output

chmod +r salary.sh

./salary.sh

Enter the basic salary:

1000

Gross salary = 1300.0

./salary.sh

Enter the basic salary:

2500

Gross salary = 3250.0

Shell script to compute the gross salary of an employee

```
#!/bin/sh
echo "Enter the basic salary: "
read basic
da = `echo 0.1*$basic |bc`
hra = `echo 0.2*$basic |bc`
gross = `echo $basic + $da + $hra |bc`
echo "Gross salary = $gross"
```

Date 8/11/21

Expt. No. 07 Page No. 07

Output

Changed ITT conv.sh

'conv.sh

Enter the temperature in Fahrenheit

36

Temperature in Celsius is 9.2220

'conv.sh

Enter the temperature in Fahrenheit

39

Temperature in Celsius is 0

Output

- read str
- enter 2 numbers
- i.e
- 30

addition : 40

Subtraction : -20

Multiplication : 300

Division : .33

last str

enter 2 numbers

20

40

addition : 60

Subtraction : -20

Multiplication : 800

Division : .50

Shell script to perform arithmetic operations on given two numbers

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

```
echo "Enter 2 numbers"
```

```
read n1
```

```
echo "Addition : $((n1 + n2))"
```

```
echo "Subtraction : $((n1 - n2))"
```

```
echo "Multiplication : $((n1 * n2))"
```

```
d = `echo "scale=2.$((n1 / n2))" | bc`
```

```
echo "Division : $d"
```

Date 15/11/21

Expt. No. 09

Page No. 59

shell script to find the sum of even numbers upto n

Output

```
charat 777 sum.sh
`sum.sh
Enter the limit
^
sum is 10
`sum.sh
Enter the limit
^
sum is 21
```

Date 15/11/21

Page No. 10

Expt. No. 10

Shell script to print the combinations of numbers

111	112	113	114	115
116	117	118	119	111
119	120	121	122	123
124	125	126	127	128
129	130	131	132	133
134	135	136	137	138
139	140	141	142	143
144	145	146	147	148

oakleaf
Chord 777 comb str
1 comb-sh

Output

chmod 777 power.sh
./power.sh
Input number
9
Input power
3
Power is 8
Power is 16

Shell script to find the power of a number

```
#!/bin/bash
echo "Input number"
read no
echo "Input power"
read power
p=1
for (( i=1; i<=power; i++ ))
do
    p=$((p * no))
done
echo "Power is $p"
```

Input power
9
Power is 16

Output

cloned 777 even.sh

even.sh

Enter the limit :

4

Sum is : 6

even.sh

Enter the limit :

4

Sum is : 12

```
shell script to find the sum of n natural even  
numbers  
#!/bin/sh  
echo "Enter the limit : "  
read n  
sum=0  
for ((i=1; i<=n; i++))  
do  
    if [ $((i%2)) -eq 0 ]  
    then  
        sum=$((sum+i))  
    fi  
done  
echo "sum is : $sum "
```

Subject: Enter sic and sce(50) marks for subject 1 :

45

44

A Grade

Enter sic and sce(50) marks for subject 2:

46

40

A grade

Enter sic and sce(50) marks for subject 3:

40

10

Fail

Enter sic and sce(50) marks for subject 4:

15

20

Fail

Enter sic and sce(50) marks for subject 5:

30

22

D grade

Enter sic and sce(50) marks for subject 6:

18

20

Fail

Number of subjects passed : 3

Number of subjects failed : 3

Ques 13- Shell script to display pass class of a student.

```
#!/bin/bash
pass=0
fail=0
for ((i=1; i<=6; i++))
do
    echo "Enter sic and sce(50) marks for subject $i : "
    read sic
    read sce
    total=$((sic+sce))
    if [ $total -gt 90 ]
    then
        echo "S Grade"
    else
        pass=$((pass+1))
    fi
    if [ $total -gt 80 ]
    then
        echo "A Grade"
    else
        pass=$((pass+1))
    fi
    if [ $total -gt 70 ]
    then
        echo "B Grade"
    else
        pass=$((pass+1))
    fi
    if [ $total -gt 60 ]
    then
        echo "C Grade"
    else
        pass=$((pass+1))
    fi
    if [ $total -gt 50 ]
    then
        echo "D Grade"
    else
        pass=$((pass+1))
    fi
done
```

Expt. No. _____

Date _____
Page No. 14

```
if [ $total -gt 40 ]
then echo "E Grade"
pass=$((pass+1))
else
echo "Fail"
fail=$((fail+1))
fi
done
```

```
echo "Number of subjects passed: $pass"
echo "Number of subjects failed: $fail"
```

Expt. No. 14

Date _____
Page No. 15

File 14:- Shell script to find Fibonacci series upto n.

Output:- shmed 777 lab14.sh
Enter n:
3
0
1
1
2
3
4
5
#!/bin/sh
echo "Enter n: "
read n
f1=0
f2=0
for ((i=0; i<n; i+))
do
echo "f1"
file=\$f1\$f2
f1=\$f2
f2=\$file
done

Expt. No. — 15

Date — _____
Page No. — 15

Output:-
Opened file lab15.sh
Enter string

Good
The given string has 2 vowels.

Enter string

zzz

The given string has 0 vowels.

```
Lab15:- Shell script to count number of vowels of a string.  
#!/bin/sh  
echo "Enter string: "  
read str  
numcount=0  
v=""  
for (( i=0; i<${#v}; i++ ))  
do  
    s="${str:$i:1}"  
    if [ "$s" = 'a' -o "$s" = 'e' -o "$s" = 'i' -o "$s" = 'o'  
        -o "$s" = 'u' ]  
    then  
        numcount=$((numcount+1))  
    fi  
done  
echo "The number of vowels: $numcount."
```

Output :-
showed 444 lab16.sh

Enter filename
usp.txt

Number of characters in usp.txt : 37
Number of words in usp.txt : 8
Number of lines in usp.txt : 3

Expt. No. — 16 —

Date _____
Page No. 17

Lab16 - Shell script to check number of lines, words, characters
in a file.

```
#!/bin/sh
echo "Enter filename"
read file
c=$(cat $file | wc -c)
w=$(cat $file | wc -w)
l=$(grep -c "\." $file)
echo "Number of characters in $file : $c"
echo "Number of words in $file : $w"
echo "Number of lines in $file : $l"
```

Expt. No. 17Page No. 18Date 31/1/22

Ques:- Write a C/C++ program to print the contents
of the 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 ("%c", *ptr);
    return 0;
}
```

Expt. No. 18

Date 31/12/22

Page No. 19

```
#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] <symbolic link> <new file> \n");
    return 1;
}

if(argc==4)
{
    if((symbolicLink(argv[1], 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 link created \n");
}

return 0;
}
```

Ques 19:- Write C/C++ POSIX compliant program that prints the POSIX defined configuration options supported by any given system using feature test macros.

```
#define _POSIX_SOURCE
#define _POSIX_C_SOURCE 199309L
#include <stropts.h>
#include <unistd.h>

int main()
{
    ifdef _POSIX_JOB_CONTROL
        printf("System supports job control\n");
    else
        printf("System does not support job control\n");
    endif
    ifdef _POSIX_SAVED_IDS
        protof("System supports saved set-UID and saved set-GID");
    endif
    ifdef _POSIX_CHTDN_Restricted
        printf("System restricted edition is old\n", _POSIX_CHTDN_Restricted);
    else
        printf("System doesn't support system restricted edition\n");
    endif
    ifdef _POSIX_NO_TRUNC
        printf("Pathname Trunc option is option is old\n", _POSIX_NO_TRUNC);
    else
        printf("System doesn't support system-wide pathname trunc option\n");
    endif
}
```

```
#endif  
#if !POSIX_VDISABLE  
printf ("Disable character for terminal files is %d\n",  
    POSIX_VDISABLE);  
#else  
printf ("System doesn't support POSIX_VDISABLE\n");  
#endif  
return 0;  
}
```

Ques:- Write a C/C++ program which demonstrates interprocess communication between a reader process and a writer process. Use mkfifo, open, read, write and close API's in your program.

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

int main( int argc, char * argv[] )
{
    int fd;
    short buf[256];
    if (argc != 2 && argc != 3)
    {
        printf ("USAGE: %s <file> [<arg>]\n", argv[0]);
        return 0;
    }

    mkfifo ( argv[1], S_IRIFO | S_IWUWXU | S_IRWGXG | S_IRWXO );
    if (argc == 2)
    {
        printf ("%s", buf);
    }

    fd = open ( argv[1], O_RDONLY | O_NONBLOCK );
    while (read ( fd, buf, sizeof (buf)) > 0)
    }
```

Expt. No. _____

Date _____
Page No. 23

```
else
{
    fd = open("angul[1].J", O_WRONLY);
    write(fd, arr[2], strlen(arr[2]));
}
close(fd);
```