vi leapyr.sh

chmod 777 leapyr.sh

./ leapyr.sh

enter the leap year

2000

leap year

./ leapyr.sh

enter the leap year

2015

not leap year

Expt. No. 1 Page No. 1

Shell script to find out if given year is leap or not.

```
#!/bin/sh
echo "enter the leap year"
read year
if [ $((year %.400)) -eq 0 ]
then
echo "leap year"
elif [ $((year %.100)) -eq 0 ]
then
echo "not leap year"
elif [ $((year %.4)) -eq 0 ]
then
echo "leap year"
else
echo "not leap year"
fi
```

25/10/21

Teacher's Signature :

```
vi area.sh
chmod 777 area.sh
./area.sh
Enter the radius
10
Area of the circle
314.1500
```

Date 25-10-21

Expt. No. 2 Page No. 2

Shell script to find the area of a circle.

```
#!/bin/sh
echo "Enter the radius"
read radius
echo "Area of the circle"
pi = 3.1415
ans = `echo $pi\*$radius\*$radius | bc`
echo $ans
```

25/10/21

Teacher's Signature :

vi

./prog3.sh

Enter the number

22

Positive

./prog3.sh

Enter the number

-10

Negative

Enter the number

0

Zero

Date 25-10-21

Expt. No. 3 Page No. 3

shell script to check whether the nuber is zero / positive / negative.

```
#!/bin/sh
echo "Enter the no."
read no
if [ $no -eq 0 ]
then
echo "Zero"
elif [ $no -lt 0 ]
then
echo "Negative"
else
echo "Positive"
fi
```

Teacher's Signature :

./biggest.sh
Enter the first no.
12
Enter the second no.
44
Enter the third no.
20
44 is the greatest

Date 25-10-21

Expt. No. 4 Page No. 4

Shell script to find the biggest of three numbers.

#!

```
#!/bin/sh
echo "Enter the first no"
read n1
echo "Enter the second no"
read n2
echo "Enter the third no"
read n3
if [ $n1 -gt $n2 ] && [ $n1 -gt $n3 ]
then
echo $n1 "is the greatest"
elif [ $n2 -gt $n1 ] && [ $n2 -gt $n3 ]
then
echo $n2 "is the greatest"
else
echo $n3 "is the greatest"
fi
```

25/10/21

Teacher's Signature :

```
./factorial.sh
Enter a number
6
The factorial is 720
```

Date 8-11-21

Expt. No. 5

Page No. 5

shell script to find the factorial of a number.

```
#!/bin/sh
echo "enter a number"
read num
f=1
while [ $num -gt 1 ]
do
f=$((f*num))
num=$((num-1))
done
echo "the factorial = $f"
```

8/11/21

Teacher's Signature : _____

./grosssalary
Enter the basic salary
1000
Gross salary = 1300

Date 8-11-21

Expt. No. 6 Page No. 6

shell script to compute the gross salary of an employee.

```
#!/bin/sh
echo "Enter the basic salary"
read basic
da = `expr $basic \* 10/100`
hra = `expr $basic \* 20/100`
gross_sal = $(( basic + $da + $hra ))
echo "gross sal = $gross_sal"
```

8/11/21

Enter the temp in fearheit
100
100 F is equal to 37 C

Date 8-11-21

Expt. No. 7 Page No. 7

Shell script to convert the temperature Farenheit to Celsius.

```
#!/bin/sh
echo "enter the temp in farenheit"
read f
c=`echo "scale=4 ; ($f-32)/1.8" | bc`
echo "$f F is equal to $c C"
```

N
8/11/21

1. add
2. sub
3. mul
4. div

select your choice (1-4)

3

Enter two numbers

2 5

Result = 10

Date 8-11-21

Expt. No. 8

Page No. 8

Shell script to perform arithmetic operations on given 2 numbers.

```
#!/bin/sh
echo "1. add"
echo "2. sub"
echo "3. mul"
echo "4. div"
echo "Select your choice (1-4)"
read choice
if [ $choice -eq 1 ]
then
echo "enter 2 numbers"
read a b
result = `expr "$a + $b" | bc`
echo "Result = $result"
elif [ $choice -eq 2 ]
then
echo "enter 2 numbers"
read a b
result = `expr "$a - $b" | bc`
echo "Result = $result"
elif [ $choice -eq 3 ]
then
echo "enter 2 numbers"
read a b
result = `expr "$a \* $b" | bc`
echo "Result = $result"
elif [ $choice -eq 4 ]
then
```

8/11/21

Teacher's Signature : _____

Date

Expt. No.

Page No. 9

```
echo "enter 2 numbers "
read a b
result = `expr "$a/$b" | bc`
echo "Result = $result"
else
echo "Wrong choice"
fi
```

enter no,
10
Sum = 20

Date 15-11-21

Expt. No. 9 Page No. 10

Shell script to find the sum of even numbers upto n.

```
#!/bin/bash
echo "enter no."
read n
i=2
while [ $i -lt $n ]
do
sum = $((sum+i))
i = $((i+2))
done
echo "Sum = $sum"
```

15/11/21

Teacher's Signature :

1 1 1
1 1 2
1 1 3
1 2 1
1 2 2
1 2 3
1 3 1
1 3 2
1 3 3
2 1 1
2 1 2
2 1 3
2 2 1
2 2 2
2 2 3
2 3 1
2 3 2
2 3 3
3 1 1
3 1 2
3 1 3
3 2 1
3 2 2
3 2 3
3 3 1
3 3 2
3 3 3

Date 15-11-21

Expt. No. 10

Page No. 11

Shell script to print the combinations of numbers 1 2 3.

```
#!/bin/sh
for ((i=1; i<=3; i++))
do
for ((j=1; j<=3; j++))
do
for ((k=1; k<=3; k++))
do
echo $i $j $k
done
done
done
```

15/11/21

Teacher's Signature : _____

Enter the no.
2
Enter the power
3
result = 8

Date 15-11-21

Expt. No. 11 Page No. 12

Shell Script to find powers of a number.

```
#!/bin/bash
echo "enter the no."
read number
echo "enter the power"
read power
result = 1
count = 0
while [ $power -ge 1 ]
do
result = `expr $result \* $number`
power = `expr $power - 1`
count = `expr $count + 1`
done
echo "result = $result"
```

15/11/21

input upper limit
6
15

Date 15-11-21

Expt. No. 12 Page No. 13

Shell Script to find the sum of n natural numbers.

```
#!/bin/sh
echo "input upper limit"
read max
sum=0
for((i=1;i<max;i++))
do
sum=$(expr $sum + $i)
done
echo $sum
```

15/11/21

Enter CIE marks
12
Enter SEE marks
11

Enter CIE marks
50
Enter SEE marks
50

Enter CIE marks
40
Enter SEE marks
48

Enter CIE marks
45
Enter SEE marks
90

Enter CIE mark
40
Enter SEE marks
90

4 subjects passed
1 subjects failed.

Date 29/11/21

Expt. No. 13 Page No. 14

Shell Script to display pass class of student.

```
#!/bin/bash
for ((i=1; i<=5; i=i+1))
do
echo "Enter CIE marks"
read cie
echo "Enter SEE marks"
read see
total=$((cie+see))
if [ $total -gt 40 ]
then
countpass=$((countpass+1))
else
countfail=$((countfail+1))
fi
done
echo "$countpass subjects passed"
echo "$countfail subjects failed"
```

Enter the number

4

The fibonacci series is :

0

1

1

2

Date 29/11/21

Expt. No. 14

Page No. 15

Shell Script to find the Fibonacci Series upto n.

```
#!/bin/bash
echo "Enter the number"
read n
a=0
b=1
echo "The fibonacci series is :"
if [ $n -gt 0 ]
then
for ((i=0; i<n; i++))
do
echo "$a"
fib=$((a+b))
a=$b
b=$fib
done
else
echo "given number is invalid"
echo "number must be greater 0"
fi
```

Teacher's Signature : _____

enter the string
hello world
3

Date 29/11/21

Expt. No. 15

Page No. 16

Shell Script to count the number of vowels on a string.

```
#!/bin/sh
echo "enter the string"
read name
vowels=$(echo $name | grep -o "[aeiouAEIOU]" | wc --lines)
echo "$vowels"
```

Teacher's Signature : _____

enter filename
text1.sh
No. of characters in text1.sh 47
No. of words in text1.sh 20
No. of lines in text1.sh 4

Date 29/11/21

Expt. No. 16 Page No. 17

Shell script to check no. of lines, words and characters in a file.

```
#!/bin/sh
echo "enter filename"
read file
w=`cat $file | wc -w`
c=`cat $file | wc -c`
l=`grep -c \".\" $file`
echo "No. of characters in $file = $c"
echo "No. of words in $file = $w"
echo "No. of lines in $file = $l"
```

Teacher's Signature :

SSH_AGENT_PID = 3207
HOSTNAME = localhost.localdomain
DESKTOP_STARTUP_ID =
SHELL = /bin/bash
TERM = xterm

Date 03/01/12

Expt. No. 17

Page No. 18

Write c/c++ program to output the content 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;
}
```

Teacher's Signature : _____

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

Usage : ./a.out [-s] <org-file> <new-link>
[root@localhost usp]# ./a.out 1.c 2
Hard link created

[root@localhost usp]# ls -l
-rw-r--r-- 2 root root 657 Mar 27 16:44 1a.c
-rw-r--r-- 2 root root 657 Mar 27 16:44 2

[root@localhost usp]# ./a.out 1a.c 2
Cannot create hard link

[root@localhost usp]# ls -l a.c zz
Symbolic link created

[root@localhost usp]# readlink zz
1a.c

Date 23/11/22

Expt. No. 18 Page No. 19

Write a C/C++ program to emulate the 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[1],"-s")))
 { printf("Usage: ./a.out [-s] <org_file> <new_link>\n");
   return 1;
 }
 if(argc==4)
 { if((symlink(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");
 }
return 0;
}
```

Teacher's Signature : _____

System supports job control
System supports saved set-UID's and saved set-GID's
chown_restricted option is 1
Pathname trunc option is 1
Disable character for terminal files is 0

Date 03/1/22

Expt. No. 19 Page No. 20

Write a C/C++ POSIX complaint program that prints 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()
{
#ifdef _POSIX_JOB_CONTROL
  printf("System supports job control\n");
#else
  printf("System does not support job control\n");
#endif
#ifdef _POSIX_SAVED_IDS
  printf("System supports saved UID's and saved set-GID's");
#else
  printf("System does not support saved set-UID and saved set-GID's\n");
#endif
#ifdef _POSIX_CHOWN_RESTRICTED
  printf("chown_restricted option is %d\n", _POSIX_CHOWN_RESTRICTED);
#else
  printf("System don't support chown_restricted option\n");
#endif
#ifdef _POSIX_VDISABLE
  printf("Disable characters for terminal files is %d\n", _POSIX_VDISABLE);
#endif
#else
```

Teacher's Signature :

Date

Expt. No. _____ Page No. 21

```
    printf("System wait support - PSMOUSE_DISABLE\n");
#endif
    return 0;
}
```

/* Terminal 1 - writer */
[root@localhost usp]# ./a.out FIFO1 "This is USP & CD lab"

/* Terminal 2 - reader */
[root@localhost /]# ./a.out FIFO1
This is USP & CD lab

Date 03-01-22

Expt. No. 20 Page No. 22

Write a C/C++ program that demonstrates interprocess communication between a reader & writer process. Use mkfifo, open, read, write & close API 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>
int 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_IFIFO|S_IRWXU|S_IRWXG|S_IRWXO);
   if (argc==2)
   { fd = open (argv[1], O_RDONLY|O_NONBLOCK);
   while (read (fd, buf, sizeof (buf))>0)
   printf ("%s", buf);
   }
   else
   { fd =open (argv[1], O_WRONLY);
     write ( fd, argv[2], strlen (argv[2])); }
close (fd);
}
```

Teacher's Signature : _____