Expt. No. 01 Date 25/10/21

Page No. 01

Shell script to find if the given year is a leap or not

```
echo "Enter the year"
read leap
if [ $((leap % 400)) -eq 0 ]
then echo "The year is a leap year"
elif [ $((leap % 100)) -ne 0 ]
then echo "The year entered is not a leapyear"
elif [ $((leap % 4)) -eq 0 ]
then echo "The year entered is a leap year"
else
echo "The year entered is not a leap year"
fi
```

Teacher's Signature : _____

Enter the year
2024
The year is a leap year

Enter the year
2001
The year is not a leap year

Expt. No. 02 Date 25/10/21

Page No. 02

Shell script to find the area of a circle

```
echo "Enter the radius of the circle"
read rad
pi=3.14
area='echo $pi\*$rad\*$rad | bc'
echo "Area of the circle is $area"
```

Teacher's Signature : _____

Order for radius of the circle
2
Area of the circle = 12.56

Expt. No. 03

Date 25/10/21

Page No. 03

Shell script to check whether the number is zero/positive/negative

```
echo "Enter a number"
read no
if [$no -eq 0]
then echo "zero"
elif [$no -gt 0]
then echo "positive"
else
echo "negative"
fi
```

Teacher's Signature : _____

Enter a number
0
zero

Enter a number
5
positive

Enter a number
-1
negative

Expt. No. 04

Date 29/10/21

Page No. 04

Shell script to find the biggest of three numbers.

```
echo "Enter the first number"
read n1
echo "Enter the second number"
read n2
echo "Enter the third number"
read n3
if [$n1 -gt $n2 -a $n1 -gt $n3]
then echo "$n1 is the greatest"
elif [$n2 -gt $n1 -a $n2 -gt $n3]
then echo "$n2 is the greatest"
else
echo "$n3 is the greatest"
fi
```

Teacher's Signature : _____

Enter the first number
1
Enter the second number
2
Enter the third number
3
3 is the greatest

Expt. No. 05

Date 8/11/21

Page No. 05

Shell script to find the factorial of a number

```
echo "Enter a number"
read num
fact=1
for ((i=2; i<=num; i++))
do
fact=$((fact * i))
done
echo "$fact"
```

Teacher's Signature :

Enter a number
3
6

Exp No. 06

Date 8/11/21

Page No. 06

Shell script to compute the gross salary of an employee

```
echo "Enter the basic salary of the employee"
read basic
da = `echo 0.1 \* $basic | bc`
hra = `echo 0.2 \* $basic | bc`
gross = `echo $basic + $da + $hra | bc`
Echo "Gross salary : $gross"
```

Teacher's Signature

Enter the basic salary of the employee
1000
Expense salary : 1300.0

Expt. No. 07 Date 8/11/21

Page No. 07

Shell script to convert the temperature Fahrenheit to Celsius.

```
echo " Enter the temperature in Fahrenheit "
read temp
t = `echo " scale = 4 ; 5/9 ; " | bc `
cel = `echo $(( temp - 32 )) \* $t | bc `
echo "Temperature in celsius : $cel"
```

Teacher's Signature : _____

Enter the temperature in fahrenheit
32
Temperature in celsius : 0

Expt. No. 08 Date 8/11/21

Page No. 08

Shell script to perform arithmetic operations on given two numbers

```
echo "Enter two numbers"
read no1
read no2
echo "Addition : $((no1+no2))"
echo "Subtraction : $((no1-no2))"
echo "Multiplication : $((no1*no2))"
echo "scale=2; $no1/$no2;" | bc`
echo "Division : $d"
```

Teacher's Signature : _____

Enter two numbers
6
3
Addition : 9
Subtraction : 3
Multiplication : 18
Division : 2.00

Expt. No. 09

Date 15/11/21

Page No. 09

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

```
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 is $sum"
```

Teacher's Signature : _____

Enter the value of n
4
Sum of even numbers upto 4 is 6.

Expt. No. 10

Date 15/11/21

Page No. 10

Shell script to print the combinations of numbers 123

```
echo "The combinations of 123 are:"
for i in 123
do
for j in 123
do
for k in 123
do
echo "$i$j$k"
done
done
done
```

Teacher's Signature :

The combinations of 123 are:

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

Date 15/11/21

Expt. No. 11

Page No. 11

Shell script to find the power of a number

```
echo "Enter a number and power"
read a
read b
p=$b
res=1
while [ $b -gt 0 ]
do
res=`echo "$res * $a" | bc`
b=`echo $b-1 | bc`
done
echo "Result is $res"
```

Teacher's Signature :

Enter a number and power
2
1
Result is 2

Expt. No. 12

Date 15/11/21

Page No. 12

Shell script to find the sum of n natural numbers.

```
echo "Enter the value of n"
read n
sum = 0
for ((i=1 ; i<=n ; i++))
do
sum = $((sum + i))
done
echo "Sum of $n natural numbers is : $ sum"
```

Teacher's Signature : _____

Enter the value of n
3
Sum of 3 natural numbers is : 6

Shell script to display the pass class of a student

```
pass=0
fail=0
for((i=0; i<6; i++))
do
echo "Enter your cie marks out of 50"
read cie
echo "Enter your vsee marks out of 50"
read vsee
total=$(( cie + vsee ))
if [$total -gt 90 ]
then echo "S grade"
pass=$((pass+1))
elif [$total -gt 80 ]
then echo "A grade"
pass=$((pass+1))
elif [$total -gt 70 ]
then echo "B grade"
pass=$((pass+1))
elif [$total -gt 60 ]
then echo "C grade"
pass=$((pass+1))
elif [$total -gt 50 ]
then echo "D grade"
pass=$((pass+1))
```

Teacher's Signature : _____

Enter your cie marks out of 50
23
Enter your see marks out of 50
23
Fgrade
Enter your cie marks out of 50
12
Enter your see marks out of 50
12
Fail
Enter your cie marks out of 50
34
Enter your see marks out of 50
45
Bgrade
Enter your cie marks out of 50
45
Enter your see marks out of 50
45
Fgrade
Enter your cie marks out of 50
40
Enter your see marks out of 50
40
Bgrade
Enter your cie marks out of 50
34
Enter your see marks out of 50
32
Cgrade
Number of subjects passed : 5
Number of subjects failed : 1

```
elif [ $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 "
```

Shell script to find the Fibonacci series upto n

```
echo "Enter the value of n"
read n
a=0
b=1
for (( i=0; i<n; i++))
do
echo "$n"
fn = $(( a+b))
a=$b
b=$fn
done
```

Enter the value of n

6

0
1
1
2
3
5

Expt. No. 15

Date 29/11/21

Page No. 16

Shell script to count the number of vowels of a string

```
echo "Enter a string"
read string
count=0
l='expr "$string" : '.*'
for((i=0; i<$l; 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
string='expr "$string" : '.\(.*\)''
done
echo "The number of vowels : $count"
```

Teacher's Signature : _____

Enter a string
Hello
The number of vowels : 2.

Date 29/11/21

Expt. No. 16

Page No. 17

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

```
echo "Number of lines:"
cat prog15.sh | wc -l
echo "Number of words:"
cat prog15.sh | wc -w
echo "Number of characters:"
cat prog15.sh | wc -c
```

Teacher's Signature : _____

Number of lines :
15
Number of words :
55
Number of characters :
312

Date 3/1/22

Expt. No. 17

Page No. 18

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;
}
```

Teacher's Signature : _____

SSH_AGENT_PID=3207
HOSTNAME=localhost.localdomain
DESKTOP_STARTUP_ID=
SHELL=/bin/bash
TERM=xterm
HISTSIZE=1000
KDE_NO_IPV6=1
GTK_RC_FILES=/etc/gtk/gtkrc:/root/.gtkrc-1.2-gnome2
WINDOWID=44040273
OLDPWD=/root/tan
QTDIR=/root/lib/qt-3.3.
QTDINC=/root/lib/qt-3.3/include
USER=root
LS_COLORS=no=00;fi=00:di=00;34:l
GNOME_KEYRING_SOCKET=/tmp/keyring-vsDBVL/socket
SSH_AUTH_SOCK=/tmp/ssh-SEwJHJ3149/agent.3149
KDEDIR=/usr
SESSION_MANAGER=local/localhost.localdomain:/tmp/.
ICE-unix/1349
MAIL=/var/spool/mail/root
DESKTOP_SESSION=default
PATH=/usr/lib/qt-3.3/bin:/usr/kerberos/sbin:/usr/kurberos/
bin:/usr/local/bin:/sbin/usr/sbin:/usr/bin.
GDM_XSERVER_LOCATION=local
INPUTRC=/etc/inputrc
PWD=/root/tan/muspl
XMODIFIERS=@im=none
KDE_IS_PRELINKED=1
LANG=en_US.UTF-8

GDMSESSION=default
SSH_ASKPASS=/usr/libexec/openssh/gnome-ssh-askpass
HOME=/root
DISPLAY=:0.0
GL_BROKEN_FILENAMES=1
COLORTERM=gnome-terminal
XAUTHORITY=/tmp/gdm.3X7IUW
_=./a.out.

Date 3/1/22

Expt. No. 18 Page No. 20

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");
```

Teacher's Signature : _____

```
else
print ("Hard link created\n");
}
return 0;
}
```

./a.out [-s] <org_file><new_link>
./a.out 1234
./a.out [-s] <org_file><new_file>
./a.out 1.cz
Hard link created.
ls -l
-rw-r--r-- 2 root root 657 Mar 27 14:22 1a.c
-rw-r--r-- 2 root root 657 Mar 27 14:22 z
./a.out 1a.cz
Cannot creat hard link.
./a.out -s 1a.c zz
Symbolic link created.
ls -l
-rw-r--r-- 2 root root 657 Mar 27 14:22 1a.c
lrwxrwxrwx 1 root root 4 Apr 1 18:32 zz → 1a.c
readlink zz
1a.c.

Date 3/1/22

Expt. No. 19

Page No. 22

Write a C/C++ POSIX complaint 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()
{
#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 set-UID and set-GID\n");
#else
printf("System does not support saved set-UID and
set-GID\n");
#endif
#ifdef _POSIX_CHOWN_RESTRICTED
printf("chown_restricted option is %d\n",_POSIX_
CHOWN_RESTRICTED);
#else
printf("Systems does not support chown_restricted
```

Teacher's Signature : _____

```
option\n");
#endif
#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
#ifdef _POSIX_VDISABLE
printf("Disable characted for terminal files is %d\n",
_POSIX_VDISABLE);
#else
printf("System does not support _POSIX_VDISABLE\n");
#endif
return 0;
}
```

Teacher's Signature :

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

Date 3/1/22

Expt. No. 20

Page No. 24

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>
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);
```

Teacher's Signature : _____

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

Terminal 1
./a.out FIFO1 "This is USP lab"

Terminal 2
./a.out FIFO1
This is USP lab