

Name: Jogalekar Ishan Sagar

Reg no: 19BCE2250

Course: Operating System CSE2005

Slot: L35+L36

LAB Digital Assignment 1

1. Study of basic Linux commands.

a) date :-

To see current date

Output –

```
ishan@DELLG3Ishan: ~$ date
Wed Feb 24 10:17:59 IST 2021
ishan@DELLG3Ishan: ~$
```

b) ls :-

Listing all files inside directory

Output-

```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop$ ls
1.txt                               'Visual Studio Code.lnk'      kali-linux-2020.4-live-amd64.iso.torrent
'19BCE0439_VL2020210106624_AST01 (1).pdf' 'atom.exe - Shortcut.lnk'     'pycharm64.exe - Shortcut.lnk'
'Basic linux commands'                'brave'                       'studio64.exe - Shortcut.lnk'
'Counter-Strike Global Offensive.url'  'brave.exe - Shortcut.lnk'    '~$gineering Chemistry PROJECT.docx'
'LAB DA.txt'                          'cycle sheet .txt'           '~$WRL0005.tmp'
'Start Tor Browser.lnk'                'cycle sheet bash'
'Unix command.txt'                    desktop.ini
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop$
```

c) whoami :-

Current login user

Output-

```
ishan@DELLG3Ishan: ~  
ishan@DELLG3Ishan:~$ whoami  
ishan  
ishan@DELLG3Ishan:~$
```

d) cal :-

Calendar of given month

Output-

```
ishan@DELLG3Ishan: ~  
ishan@DELLG3Ishan:~$ cal feb 2021  
February 2021  
Su Mo Tu We Th Fr Sa  
    1  2  3  4  5  6  
  7  8  9 10 11 12 13  
14 15 16 17 18 19 20  
21 22 23 24 25 26 27  
28  
ishan@DELLG3Ishan:~$
```

e) mkdir :-

Make directory

Output-

```
ishan@DELLG3Ishan: ~  
ishan@DELLG3Ishan:~$ mkdir new  
ishan@DELLG3Ishan:~$ ls  
new  
ishan@DELLG3Ishan:~$
```

f) rmdir :-

Remove directory

Output-

```
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ rmdir new
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ cd new
-bash: cd: new: No such file or directory
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$
```

g) man :-

Manual of commands

Output-

```
ishan@DELLG3Ishan:~$ man ping
ishan@DELLG3Ishan:~$
```

```
PING(8)                                iputils                                PING(8)

NAME
    ping - send ICMP ECHO_REQUEST to network hosts

SYNOPSIS
    ping [-aAbBdDfhLnOqrRUvV46] [-c count] [-F flowlabel] [-i interval] [-I interface] [-l preload] [-m mark]
        [-M pmtudisc_option] [-N nodeinfo_option] [-w deadline] [-W timeout] [-p pattern] [-Q tos] [-s packetsize]
        [-S sndbuf] [-t ttl] [-T timestamp_option] [hop...] {destination}

DESCRIPTION
    ping uses the ICMP protocol's mandatory ECHO_REQUEST datagram to elicit an ICMP ECHO_RESPONSE from a host or
    gateway. ECHO_REQUEST datagrams ("pings") have an IP and ICMP header, followed by a struct timeval and then an
    arbitrary number of "pad" bytes used to fill out the packet.

    ping works with both IPv4 and IPv6. Using only one of them explicitly can be enforced by specifying -4 or -6.

    ping can also send IPv6 Node Information Queries (RFC4620). Intermediate hops may not be allowed, because IPv6
    source routing was deprecated (RFC5095).

OPTIONS
    -4
        Use IPv4 only.

    -6
        Use IPv6 only.

    -a
        Audible ping
```

h) cd :-

Change directory

Output-

```
ishan@DELLG3Ishan: ~/new
ishan@DELLG3Ishan:~$ mkdir new
ishan@DELLG3Ishan:~$ ls
new
ishan@DELLG3Ishan:~$ cd new
ishan@DELLG3Ishan:~/new$
```

```
ishan@DELLG3Ishan: ~
ishan@DELLG3Ishan:~$ mkdir new
ishan@DELLG3Ishan:~$ ls
new
ishan@DELLG3Ishan:~$ cd new
ishan@DELLG3Ishan:~/new$ cd ~
ishan@DELLG3Ishan:~$
```

```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ cd new
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop/new$ cd -
/mnt/c/Users/Dell/Desktop
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$
```

i) vi :-

VI editor in linux shell

Output-

```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ vi 1.txt
```

j) rm:-

Remove file

Output-

```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ rm new.txt
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ cat new.txt
cat: new.txt: No such file or directory
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$
```

k) pwd :-

Print working directory

Output-

```
ishan@DELLG3Ishan: ~$ pwd
/home/ishan
ishan@DELLG3Ishan: ~$
```

L) cat :-

Print data inside file

-b to print line no.

wc word count

Output-

```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop$ cat 1.txt
My name is Ishan
Reg no : 19BCE2250
Hello world !!
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop$
```

```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop$ cat 1.txt
My name is Ishan
Reg no : 19BCE2250
Hello world !!
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop$ cat 1.txt -b
 1 My name is Ishan
 2 Reg no : 19BCE2250
 3 Hello world !!
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop$ wc 1.txt
 3 11 51 1.txt
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop$
```

M) cp :-

Copy file

Output-

```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop
ishan@DELLG3Ishan:~/mnt/c/Users/Dell/Desktop$ cp 1.txt 2.txt
ishan@DELLG3Ishan:~/mnt/c/Users/Dell/Desktop$ cat 2.txt
My name is Ishan
Reg no : 19BCE2250
Hello world !!
ishan@DELLG3Ishan:~/mnt/c/Users/Dell/Desktop$
```

N) df :-

To summarise free space on disk drive

Output-

```
ishan@DELLG3Ishan: ~$ df
Filesystem      1K-blocks    Used Available Use% Mounted on
/dev/sdb         263174212 1520164 248215892   1% /
tmpfs            3192504      0    3192504   0% /mnt/wsl
tools           108578812 91396420 17182392   85% /init
none             3190104      0    3190104   0% /dev
none             3192504      4    3192500   1% /run
none             3192504      0    3192504   0% /run/lock
none             3192504      0    3192504   0% /run/shm
none             3192504      0    3192504   0% /run/user
tmpfs            3192504      0    3192504   0% /sys/fs/cgroup
C:\              108578812 91396420 17182392   85% /mnt/c
D:\              976628732 263990248 712638484  28% /mnt/d
ishan@DELLG3Ishan: ~$
```

O) du :-

To show disk space used by files or directories

Output-

```
ishan@DELLG3Ishan: ~$ du
4      ./..landscape
36     .
ishan@DELLG3Ishan: ~$
```

P) mv :-

Rename file

Output-

```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ mv 2.txt new.txt
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ cat 2.txt
cat: 2.txt: No such file or directory
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ cat new.txt
My name is Ishan
Reg no : 19BCE2250
Hello world !!
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$
```

Q) chmod :-

Change permissions of file

Output-

```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ chmod +r 1.txt
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ chmod -w 1.txt
chmod: 1.txt: new permissions are r-xrwxrwx, not r-xr-xr-x
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$
```

```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ chmod +r 1.txt
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ chmod -w 1.txt
chmod: 1.txt: new permissions are r-xrwxrwx, not r-xr-xr-x
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$
```

R) Ping :-

Network command to ping specific host

Output-

```

ishan@DELLG3Ishan:~$ ping www.google.com
PING www.google.com (142.250.67.132) 56(84) bytes of data.
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=1 ttl=117 time=50.2 ms
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=2 ttl=117 time=45.4 ms
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=3 ttl=117 time=46.0 ms
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=4 ttl=117 time=45.2 ms
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=5 ttl=117 time=56.0 ms
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=6 ttl=117 time=40.3 ms
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=7 ttl=117 time=39.3 ms
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=8 ttl=117 time=57.4 ms
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=9 ttl=117 time=25.2 ms
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=10 ttl=117 time=21.2 ms
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=11 ttl=117 time=37.1 ms
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=12 ttl=117 time=40.9 ms
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=13 ttl=117 time=43.4 ms
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=14 ttl=117 time=18.0 ms
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=15 ttl=117 time=43.4 ms
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=16 ttl=117 time=42.2 ms
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=17 ttl=117 time=27.2 ms
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=18 ttl=117 time=39.3 ms
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=19 ttl=117 time=40.1 ms
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=20 ttl=117 time=39.5 ms
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=21 ttl=117 time=49.4 ms
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=22 ttl=117 time=40.6 ms
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=23 ttl=117 time=41.0 ms
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=24 ttl=117 time=47.9 ms
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=25 ttl=117 time=41.6 ms
64 bytes from bom12s06-in-f4.1e100.net (142.250.67.132): icmp_seq=26 ttl=117 time=34.4 ms

```

S) host :-

Details of hostname

Output-

```

ishan@DELLG3Ishan:~$ host www.google.com
www.google.com has address 142.250.67.132
ns1.google.com has address 216.239.32.10
ns1.google.com has IPv6 address 2001:4860:4802:32::a
ns2.google.com has address 216.239.34.10
ns2.google.com has IPv6 address 2001:4860:4802:34::a
ns3.google.com has address 216.239.36.10
ns3.google.com has IPv6 address 2001:4860:4802:36::a
ns4.google.com has address 216.239.38.10
ns4.google.com has IPv6 address 2001:4860:4802:38::a
www.google.com has IPv6 address 2404:6800:4009:811::2004
ns1.google.com has address 216.239.32.10
ns1.google.com has IPv6 address 2001:4860:4802:32::a
ns2.google.com has address 216.239.34.10
ns2.google.com has IPv6 address 2001:4860:4802:34::a
ns3.google.com has address 216.239.36.10
ns3.google.com has IPv6 address 2001:4860:4802:36::a
ns4.google.com has address 216.239.38.10
ns4.google.com has IPv6 address 2001:4860:4802:38::a
ishan@DELLG3Ishan:~$

```


(b) Shell Programming :-

1. Handling the command line arguments :

Ans:-

Code :


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

```
echo "Name: $1";
```

```
echo "Reg no: $2";
```

```
echo "GPA : $3 ";
```

Output-



```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop
#!/bin/sh
echo "Name: $1";
echo "Reg no: $2";
echo "GPA : $3 ";
```



```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ bash cmd.sh 'Ishan' '19BCE2250' '9.13'
Name: Ishan
Reg no: 19BCE2250
GPA : 9.13
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$
```

2. String reversal :

Ans:

Code-

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

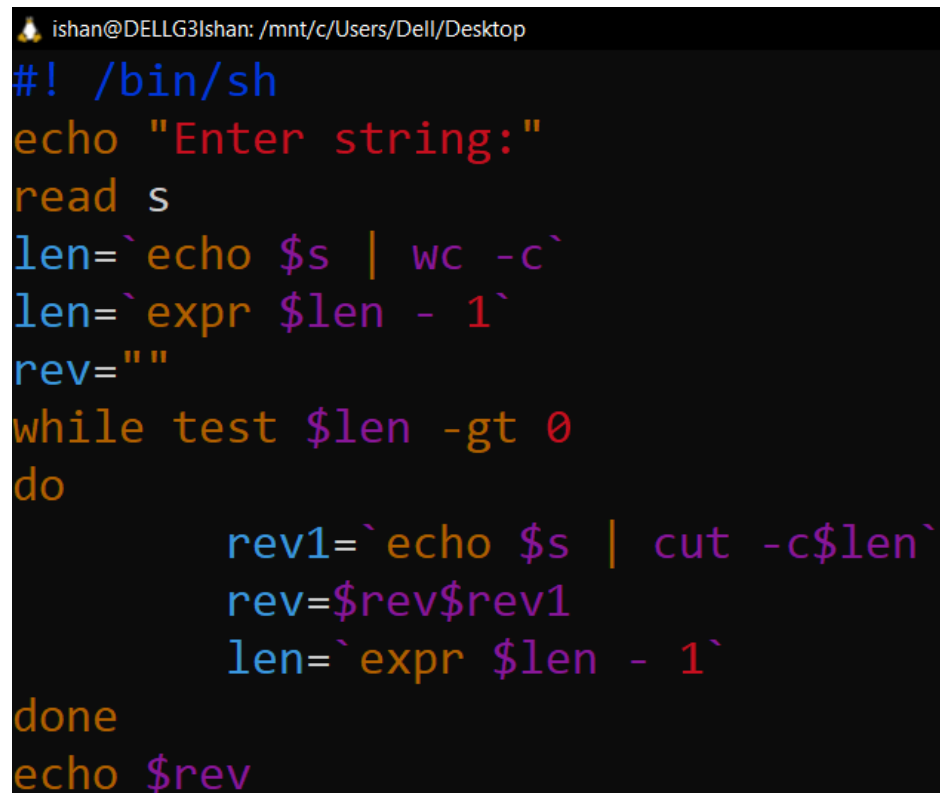
```
echo "Enter string:"
```

```
read s
```

```
len=`echo $s | wc -c`
```

```
len=`expr $len - 1`  
rev=""  
while test $len -gt 0  
do  
    rev1=`echo $s | cut -c$len`  
    rev=$rev$rev1  
    len=`expr $len - 1`  
done  
echo $rev
```

Output-



A terminal window screenshot with a dark background. The prompt is 'ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop'. The script being executed is as follows:

```
#!/bin/sh  
echo "Enter string:"  
read s  
len=`echo $s | wc -c`  
len=`expr $len - 1`  
rev=""  
while test $len -gt 0  
do  
    rev1=`echo $s | cut -c$len`  
    rev=$rev$rev1  
    len=`expr $len - 1`  
done  
echo $rev
```

```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ vi rev.sh
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ bash rev.sh
Enter string:
Ishan Jogalekar
rakelagoJ nahsI
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$
```

3. If-Else, Nested If Else, Switch cases in shell :

1) If-else :-

Ans :

Code-

```
#!/bin/sh
echo "Ishan Jogalekar - 19BCE2250"
echo "Enter no 1 : "
read a
echo "Enter no 2 : "
read b
if [ $a == $b ]
then
    echo "No 1 is equal to No 2"
else
    echo "No 1 is not equal to No 2"
fi
```

Output-

```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop
#!/bin/sh
echo "Ishan Jogalekar - 19BCE2250"
echo "Enter no 1 :"
read a
echo "Enter no 2 :"
read b
if [ $a == $b ]
then
    echo "No 1 is equal to No 2"
else
    echo "No 1 is not equal to No 2"
fi

~
```

```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ vi ifelse.sh
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ bash ifelse.sh
Ishan Jogalekar - 19BCE2250
Enter no 1 :
12
Enter no 2 :
12
No 1 is equal to No 2
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ bash ifelse.sh
Ishan Jogalekar - 19BCE2250
Enter no 1 :
45
Enter no 2 :
89
No 1 is not equal to No 2
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$
```

II) Nested If -else :-

Ans :

Code-

```
#!/bin/sh
echo "Ishan Jogalekar - 19BCE2250"
echo "Enter no:"
read num
if [ `echo "$num % 2" | bc` -eq 0 ]
then
echo "No is divisible by 2"
else
    if [ `echo "$num % 3" | bc` -eq 0 ]
    then
        echo "No is divisible by 3"
    fi
fi
```

Output-

```

ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop
#!/bin/sh
echo "Ishan Jogalekar - 19BCE2250"
echo "Enter no:"
read num
if [ `echo "$num % 2" | bc` -eq 0 ]
then
echo "No is divisible by 2"
else
    if [ `echo "$num % 3" | bc` -eq 0 ]
    then
        echo "No is divisible by 3"
    fi
fi
~
~
~
~
~
~

```

```

ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ vi nestedif.sh
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ bash nestedif.sh
Ishan Jogalekar - 19BCE2250
Enter no:
12
No is divisible by 2
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ bash nestedif.sh
Ishan Jogalekar - 19BCE2250
Enter no:
27
No is divisible by 3
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$

```

III) Switch case :-

Ans:

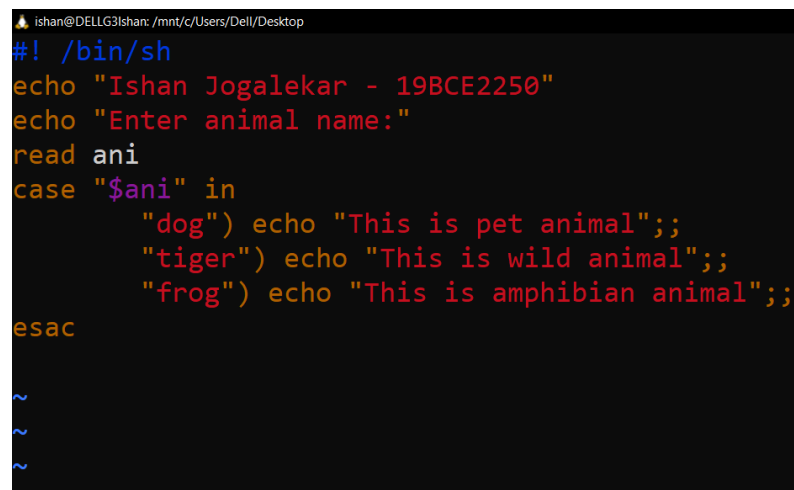
Code-

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

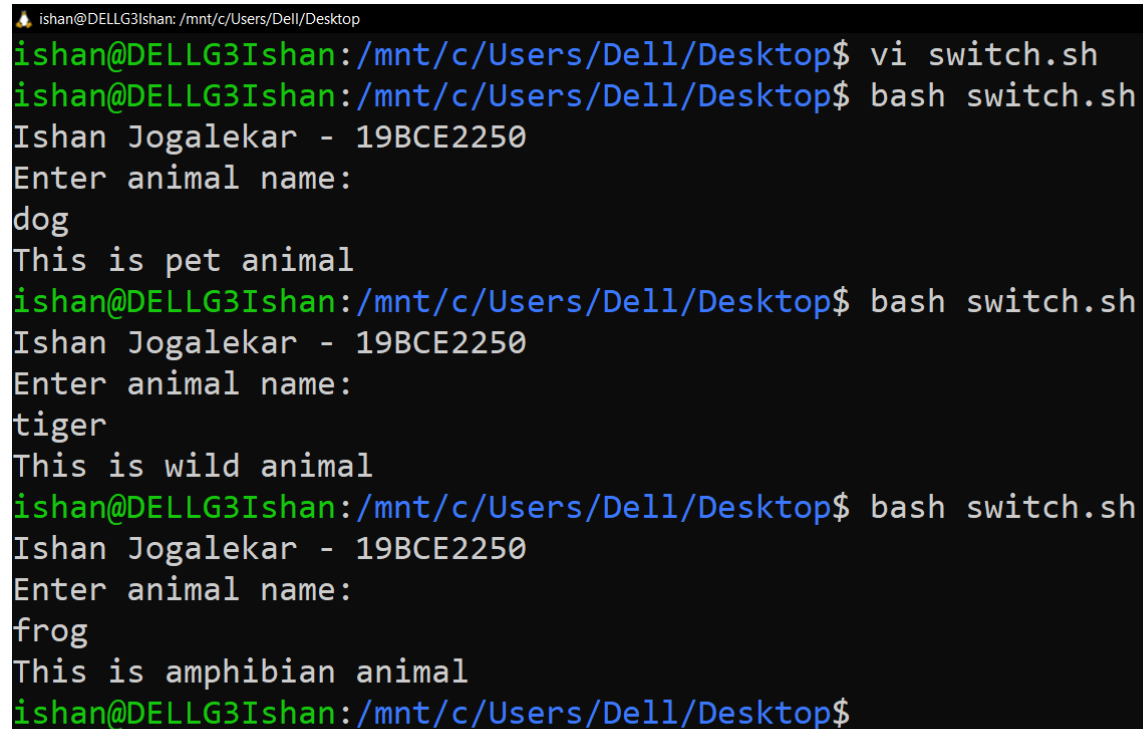
```
echo "Ishan Jogalekar - 19BCE2250"
```

```
echo "Enter animal name:"
read ani
case "$ani" in
    "dog") echo "This is pet animal";;
    "tiger") echo "This is wild animal";;
    "frog") echo "This is amphibian animal";;
esac
```

Output-



```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop
#!/bin/sh
echo "Ishan Jogalekar - 19BCE2250"
echo "Enter animal name:"
read ani
case "$ani" in
    "dog") echo "This is pet animal";;
    "tiger") echo "This is wild animal";;
    "frog") echo "This is amphibian animal";;
esac
~
~
~
```



```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop$ vi switch.sh
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop$ bash switch.sh
Ishan Jogalekar - 19BCE2250
Enter animal name:
dog
This is pet animal
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop$ bash switch.sh
Ishan Jogalekar - 19BCE2250
Enter animal name:
tiger
This is wild animal
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop$ bash switch.sh
Ishan Jogalekar - 19BCE2250
Enter animal name:
frog
This is amphibian animal
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop$
```

(c) Parent child process creation using fork() and exec() system call

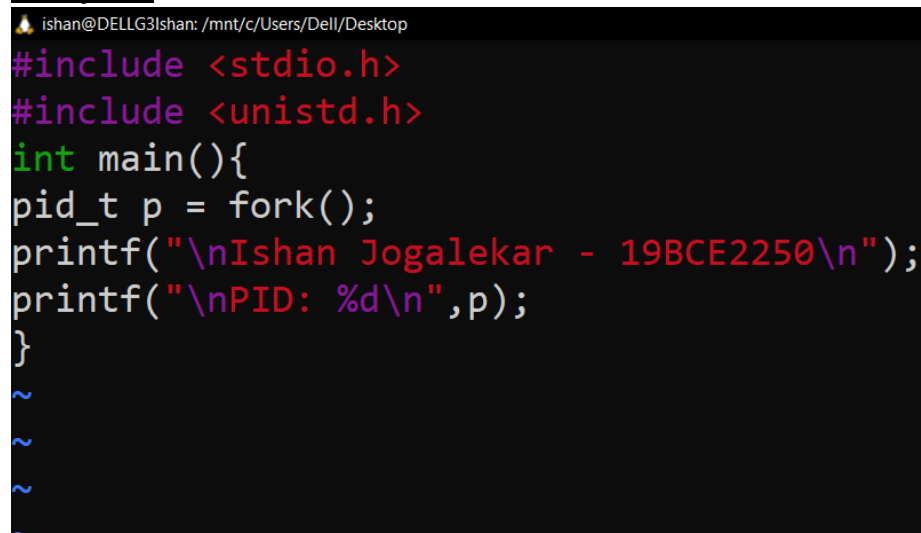
a. Checking the Process Identifier:-

Ans:

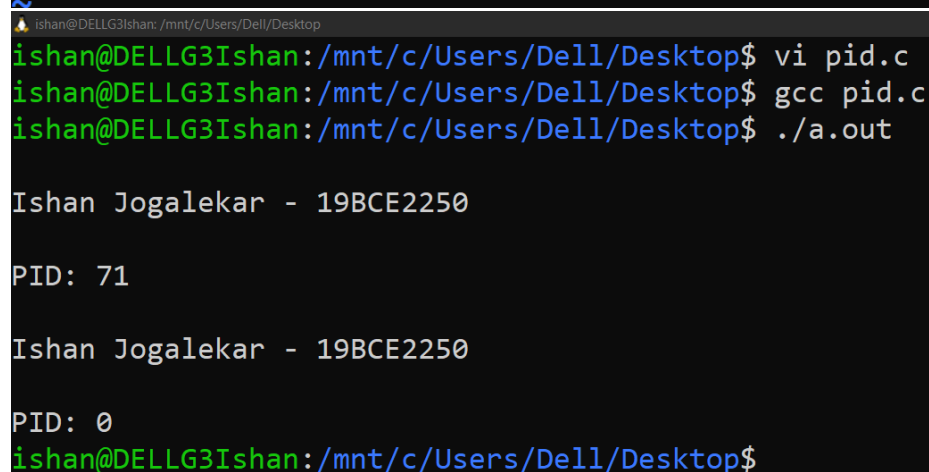
Code-

```
#include <stdio.h>
#include <unistd.h>
int main(){
    pid_t p = fork();
    printf("\nIshan Jogalekar - 19BCE2250\n");
    printf("\nPID: %d\n",p);
}
```

Output-



```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop
#include <stdio.h>
#include <unistd.h>
int main(){
    pid_t p = fork();
    printf("\nIshan Jogalekar - 19BCE2250\n");
    printf("\nPID: %d\n",p);
}
~
~
~
~
```



```
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ vi pid.c
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ gcc pid.c
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ ./a.out

Ishan Jogalekar - 19BCE2250

PID: 71

Ishan Jogalekar - 19BCE2250

PID: 0
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$
```


b. Assigning new task to child :-

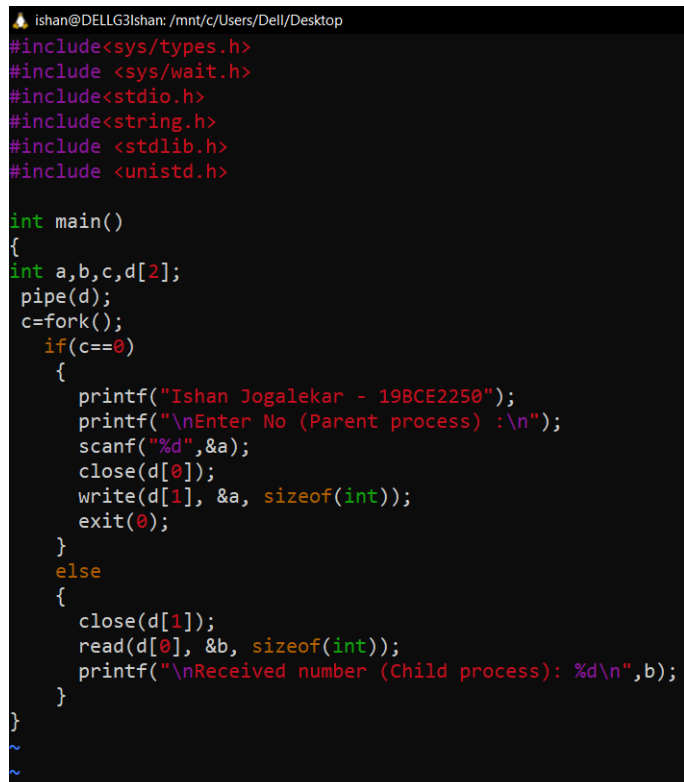
Ans:

Code:-

```
#include <stdio.h>
#include <unistd.h>
```

```
int main(){
    int a=fork();
    if (a==0)
    {
        printf("I am a child\n");
    }
    else
    {
        printf("I am a parent\n");
    }
    return 0;
}
```

Output-



```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop
#include<sys/types.h>
#include <sys/wait.h>
#include<stdio.h>
#include<string.h>
#include <stdlib.h>
#include <unistd.h>

int main()
{
    int a,b,c,d[2];
    pipe(d);
    c=fork();
    if(c==0)
    {
        printf("Ishan Jogalekar - 19BCE2250");
        printf("\nEnter No (Parent process) :\n");
        scanf("%d",&a);
        close(d[0]);
        write(d[1], &a, sizeof(int));
        exit(0);
    }
    else
    {
        close(d[1]);
        read(d[0], &b, sizeof(int));
        printf("\nReceived number (Child process): %d\n",b);
    }
}
```

```

🔥 ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ vi child.c
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ gcc child.c
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ ./a.out
Ishan Jogalekar - 19BCE2250
Enter No (Parent process) :
78

Received number (Child process): 78
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$

```

c. Providing the path name and program name to
exec() :

Ans:-

Code-

EXCE :

```

#include <stdio.h>
#include <unistd.h>

```

```

int main()
{
    int i;
    printf("\nIshan Jogalekar - 19BCE2250\n");
    printf("\nexcec.c File called by execev() file\n");
    return 0;
}

```

EXCEV :

```

#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
int main()
{

```

```

    char *args[]={"/.exec",NULL};

```

```
    execcvp(args[0],args);  
}
```

Output-

```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop  
#include <stdio.h>  
#include <unistd.h>  
  
int main()  
{  
    int i;  
    printf("\nIshan Jogalekar - 19BCE2250\n");  
    printf("\nexec.c File called by execcvp() file\n");  
    return 0;  
}  
~  
~  
~  
~
```

```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop  
#include<stdio.h>  
#include<stdlib.h>  
#include<unistd.h>  
int main()  
{  
  
    char *args[]={". /exec",NULL};  
    execcvp(args[0],args);  
}  
~  
~  
~  
~
```

```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop$ gcc exec.c -o exec  
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop$ gcc execcvp.c -o execcvp  
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop$
```

```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop$ gcc exec.c -o exec  
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop$ gcc execcvp.c -o execcvp  
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop$ ./execcvp  
  
Ishan Jogalekar - 19BCE2250  
  
exec.c File called by execcvp() file  
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop$
```

d. Synchronizing Parent and child process using wait()

:-

Ans:-

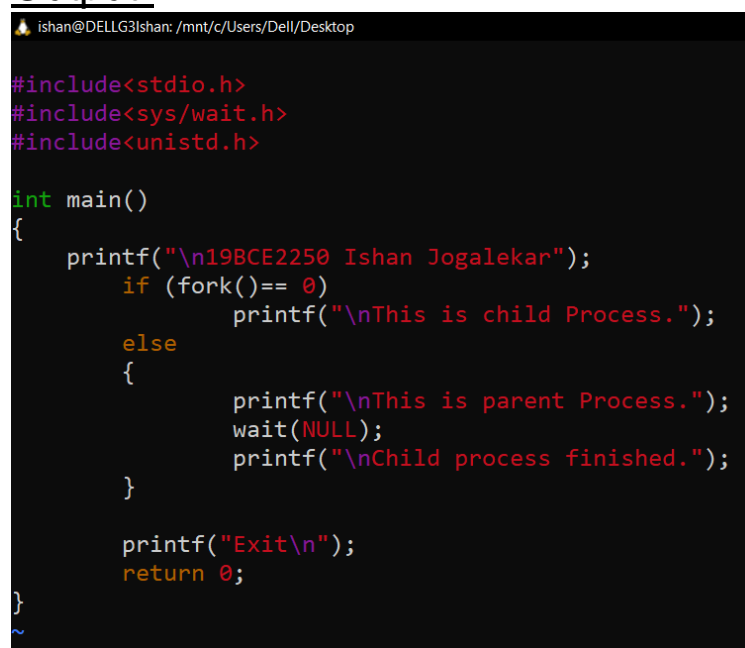
Code :

```
#include<stdio.h>
#include<sys/wait.h>
#include<unistd.h>

int main()
{
    printf("\n19BCE2250 Ishan Jogalekar");
    if (fork()== 0)
        printf("\nThis is child Process.");
    else
    {
        printf("\nThis is parent Process.");
        wait(NULL);
        printf("\nChild process finished.");
    }

    printf("Exit\n");
    return 0;
}
```

Output-



```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop
#include<stdio.h>
#include<sys/wait.h>
#include<unistd.h>

int main()
{
    printf("\n19BCE2250 Ishan Jogalekar");
    if (fork()== 0)
        printf("\nThis is child Process.");
    else
    {
        printf("\nThis is parent Process.");
        wait(NULL);
        printf("\nChild process finished.");
    }

    printf("Exit\n");
    return 0;
}
```

```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ gcc sync.c
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ ./a.out

19BCE2250 Ishan Jogalekar
19BCE2250 Ishan Jogalekar
This is child Process.Exit
This is parent Process.
Child process finished.Exit
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$
```

(d) The Collatz conjecture concerns what happens when we take

any positive integer n and apply the following algorithm:

$n = n/2$, if n is even $n = 3 \times n + 1$, if n is odd

The conjecture states that when this algorithm is continually

applied, all positive integers will eventually reach 1. For example,

if $n = 35$, the sequence is 35, 106, 53, 160, 80, 40, 20, 10, 5, 16,

8, 4, 2, 1. Write a C program using the `fork()` system call that

generates this sequence in the child process. The starting number

will be provided from the command line. For example, if 8 is

passed as a parameter on the Command line, the child process

will output 8, 4, 2, 1. Because the parent and child processes have

their own copies of the data, it will be necessary for the child to

output the sequence. Have the parent invoke the wait () call to

wait for the child process to complete before exiting the program :

Ans :-

Code-

```
#include<stdio.h>
#include<stdlib.h>
#include<sys/wait.h>
#include<unistd.h>

int main(int arg1, char **arg2){
    int n;
    printf("\nIshan Jogalekar - 19BCE2250\n");
    if(arg1==1){
        printf("\nArguments missing..\n");
        exit(0);
    }
    n = atoi(arg2[1]);
    if(n <= 0){
        printf("\nPositive interger required..\n");
        exit(0);
    }
    if(fork() == 0){
        while(n > 1){
            printf("%d ", n);
            if(n %2 == 0)
                n = n / 2;
            else
                n = 3 * n + 1;
        }
        printf("1\n\n");
    }
    else{

        wait(NULL);
    }
    return 0;
}
```

Output-

```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop
#include<stdio.h>
#include<stdlib.h>
#include<sys/wait.h>
#include<unistd.h>

int main(int arg1, char **arg2){
    int n;
    printf("\nIshan Jogalekar - 19BCE2250\n");
    if(arg1==1){
        printf("\nArguments missing..\n");
        exit(0);
    }
    n = atoi(arg2[1]);
    if(n <= 0){
        printf("\nPositive interger required..\n");
        exit(0);
    }
    if(fork() == 0){
        while(n > 1){
            printf("%d ", n);
            if(n %2 == 0)
                n = n / 2;
            else
                n = 3 * n + 1;
        }
        printf("1\n\n");
    }
    else{
        wait(NULL);
    }
    return 0;
}
-- INSERT --
```

```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ vi colltaz.c
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ gcc colltaz.c
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ ./a.out

Ishan Jogalekar - 19BCE2250

Arguments missing..
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ ./a.out -74

Ishan Jogalekar - 19BCE2250

Positive interger required..
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ ./a.out 74

Ishan Jogalekar - 19BCE2250
74 37 112 56 28 14 7 22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1
```

```
ishan@DELLG3Ishan: /mnt/c/Users/Dell/Desktop
ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$ ./a.out 23

Ishan Jogalekar - 19BCE2250
23 70 35 106 53 160 80 40 20 10 5 16 8 4 2 1

ishan@DELLG3Ishan:/mnt/c/Users/Dell/Desktop$
```