

Operating Systems Lab

Semester-5, B.tech(ICT)

Roll No:-1401010, Name:-Deep C. Patel

Assignment-2

1).

Write down a program which works like a scientific calculator and performs following operations:-

- 1) Floating point calculations
- 2) Trigonometric operations
- 3) Logarithms
- 4) Exponent etc.

(Hint: Use awk and bc)

Code:-

```
#!/bin/bash

#Created by Deep C. Patel - 1401010

clear

while true
do

    echo -e "\nEnter from following:-"
    echo "1).Basic Operations Calculator-Floating Point"
    echo "2).Trigonometric Calculations"
    echo "3).Log Calculations"
    echo "4).Exponential Calculations"
    echo "5).Exit"
    echo -e "Enter:\c"
    read choice

    case $choice in
        [1])
```

```
echo -e "\nEnter from following:-"
echo "1).Addition"
echo "2).Subtraction"
echo "3).Multiplication"
echo "4).Division"
echo "5).Modulus"
echo "6).Exit"
echo -e "Enter:\c"
read choice1

case $choice1 in
[1])
    echo -e "\nEnter X:\c"
    read x;

    echo -e "Enter Y:\c"
    read y;

    echo -e "\nSum=\c"
    echo "scale=3;$x+$y" | bc

    echo -e "\nPress any Key to Continue"
    read ch
    continue;;

[2])
    echo -e "\nEnter X:\c"
    read x;

    echo -e "Enter Y:\c"
    read y;

    echo -e "\nDifference=\c"
    echo "scale=3;$x-$y" | bc

    echo -e "\nPress any Key to Continue"
    read ch
    continue;;

[3])
    echo -e "\nEnter X:\c"
    read x;

    echo -e "Enter Y:\c"
    read y;

    echo -e "\nProduct=\c"
    echo "scale=3;$x*$y" | bc
```

```

        echo -e "\nPress any Key to Continue"
        read ch
        continue;;

[4])
    echo -e "\nEnter X:\c"
    read x;

    echo -e "Enter Y:\c"
    read y;

    echo -e "\nQuotient=\c"
    echo "scale=3;$x/$y" | bc

    echo -e "\nPress any Key to Continue"
    read ch
    continue;;

[5])
    echo -e "\nEnter X:\c"
    read x;

    echo -e "Enter Y:\c"
    read y;

    echo -e "\nModulus=\c"
    echo "$x%$y" | bc

    echo -e "\nPress any Key to Continue"
    read ch
    continue;;

[6]) continue;;
esac
continue;;

[2])
    echo -e "\nEnter Angle(degree): \c"
    read angle

    pi=$( echo "4*a(1)" | bc -l )
    rad=$( echo "$angle*($pi/180)" | bc -l )

    sin=$( echo "scale=5;s($rad)" | bc -l )
    cos=$( echo "scale=5;c($rad)" | bc -l )

```

```

        echo -e "\nSIN($deg):$sin"
        echo "COS($deg):$cos"
        echo -e "TAN($deg):\c"
        echo "scale=5;$sin/$cos" | bc -l
        echo -e "COSEC($deg):\c"
        echo "scale=5;1/$sin" | bc -l
        echo -e "SEC($deg):\c"
        echo "scale=5;1/$cos" | bc -l
        echo -e "COT($deg):\c"
        echo "scale=5;$cos/$sin" | bc -l

        echo -e "\nPress any Key to Continue"
        read ch
        continue;;

[3])
    echo -e "\nEnter Number:\c"
    read number
    echo -e "\nAnswer:\c"
    echo "l($number)/l(10)" | bc -l

    echo -e "\nPress any Key to Continue"
    read ch
    continue;;

[4])
    echo -e "\nEnter a Base:\c"
    read base
    echo -e "Enter Power:\c"
    read power

    echo -e "\nExponent:\c"
    echo "$base $power" | awk '{ print ((($1)^$2) ); }'

    echo -e "\nPress any Key to Continue"
    read ch
    continue;;

[5]) exit;;

*)
    echo -e "\nImproper Choice, Press any Key to
Continue"
    read ch
    continue;;

esac
done

```

Screenshot:-

```

❌ ◯ ramkabir@Ramkabir: /media/ramkabir/OS/Users/cmp/Deep/Semester 5/OS/Lab/Lab 2
SIN():.70710
COS():.70710
TAN():1.00000
COSEC():1.41422
SEC():1.41422
COT():1.00000

Press any Key to Continue

Enter from following:-
1).Basic Operations Calculator-Floating Point
2).Trigonometric Calculations
3).Log Calculations
4).Exponential Calculations
5).Exit
Enter:3

Enter Number:10000

Answer:4.00000000000000000001

Press any Key to Continue
❌ ◯ ramkabir@Ramkabir: /media/ramkabir/OS/Users/cmp/Deep/Semester 5/OS/Lab/Lab 2
Modulus=0

Press any Key to Continue

Enter from following:-
1).Basic Operations Calculator-Floating Point
2).Trigonometric Calculations
3).Log Calculations
4).Exponential Calculations
5).Exit
Enter:2

Enter Angle(degree): 45

SIN():.70710
COS():.70710
TAN():1.00000
COSEC():1.41422
SEC():1.41422
COT():1.00000

Press any Key to Continue
❌ ◯ ramkabir@Ramkabir: /media/ramkabir/OS/Users/cmp/Deep/Semester 5/OS/Lab/Lab 2

```

```
ramkabir@Ramkabir: /media/ramkabir/OS/Users/cmp/Deep/Semester 5/OS/Lab/Lab 2
Enter from following:-
1).Basic Operations Calculator-Floating Point
2).Trigonometric Calculations
3).Log Calculations
4).Exponential Calculations
5).Exit
Enter:1

Enter from following:-
1).Addition
2).Subtraction
3).Multiplication
4).Division
5).Modulus
6).Exit
Enter:5

Enter X:4
Enter Y:1

Modulus=0

Press any Key to Continue
```

```
ramkabir@Ramkabir: /media/ramkabir/OS/Users/cmp/Deep/Semester 5/OS/Lab/Lab 2
Enter from following:-
1).Basic Operations Calculator-Floating Point
2).Trigonometric Calculations
3).Log Calculations
4).Exponential Calculations
5).Exit
Enter:1

Enter from following:-
1).Addition
2).Subtraction
3).Multiplication
4).Division
5).Modulus
6).Exit
Enter:1

Enter X:4
Enter Y:5

Sum=9

Press any Key to Continue
```


Code:-

```
#!/bin/bash

#Created by Deep C. Patel - 1401010

clear

echo -e "\nEnter Path:"
read path

cd $path
clear

while true
do

    echo -e "\nEnter from following:-"
    echo "1).See Biggest Files"
    echo "2).See Smallest Files"
    echo "3).Exit"
    echo -e "Enter:\c"
    read choice

    case $choice in
        [1])
            set `ls -lR | awk '{ print $1 "\t" $5 "\t" $9 }' | awk '/^[-]/' | awk '{ print $2 "\t" $3 }' | sort -nr`

            echo $2

            stop=$1

            shift 2

            for true
            do
                if [ "$1" = "$stop" ]
                then

                    echo $2
                    shift 2

                else

                    break;

                fi
            done
        esac
    done
```



```

done

echo -e "\nPress any Key to Continue"
read ch
continue;;

[2])
    set `ls -lR | awk '{ print $1 "\t" $5 "\t" $9
}' | awk '/^[-]/' | awk '{ print $2 "\t" $3 }' | sort -n `

    echo $2

    stop=$1

    shift 2

    for true
    do
        if [ "$1" = "$stop" ]
        then

            echo $2
            shift 2

        else

            break;

        fi

    done

    echo -e "\nPress any Key to Continue"
    read ch
    continue;;

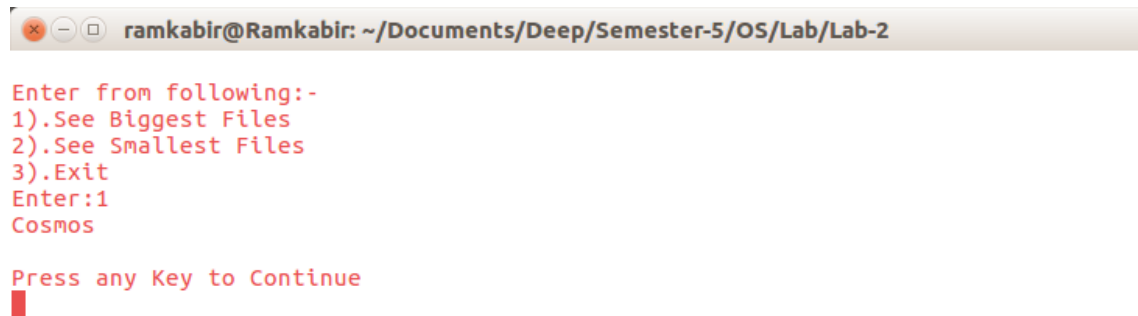
[3]) exit;;

*)
    echo -e "\nImproper Choice, Press any Key to
Continue"
    read ch
    continue;;

esac
done

```

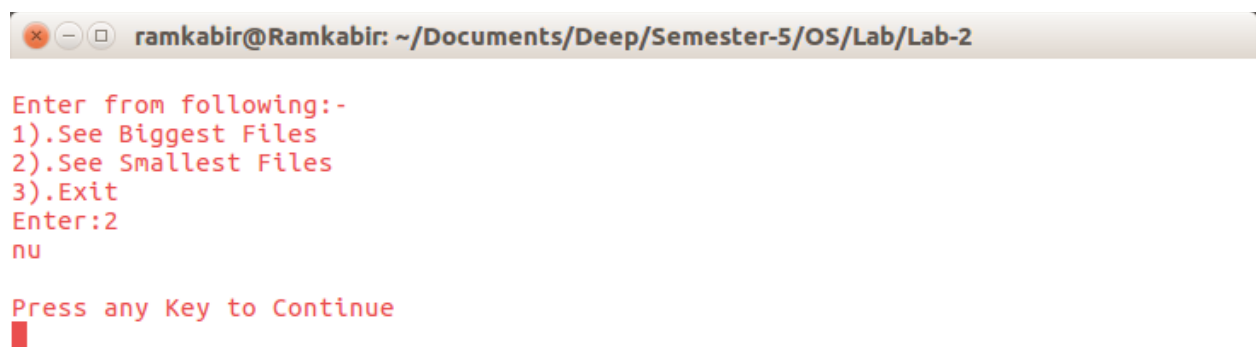
Screenshot:-



```
ramkabir@Ramkabir: ~/Documents/Deep/Semester-5/OS/Lab/Lab-2

Enter from following:-
1).See Biggest Files
2).See Smallest Files
3).Exit
Enter:1
Cosmos

Press any Key to Continue
```



```
ramkabir@Ramkabir: ~/Documents/Deep/Semester-5/OS/Lab/Lab-2

Enter from following:-
1).See Biggest Files
2).See Smallest Files
3).Exit
Enter:2
nu

Press any Key to Continue
```

3).

Run all the programs related to process creation/termination shown in the lectures (Also given at ftp).

Code 1:- f1.c

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

int main(void)
{
    int index;
    for (index = 1; index < 4; index++)
        fork();
    printf("Unix System Programming\n");
    exit(0);
}
```

Screenshot:-



A screenshot of a terminal window. The title bar shows the user 'ramkabir' and the directory '~/Documents/Deep/Semester-5/OS/Lab/Lab-2/3'. The terminal content shows the command './f1' being executed, which results in the output 'Unix System Programming' being printed twice. The prompt returns to the shell.

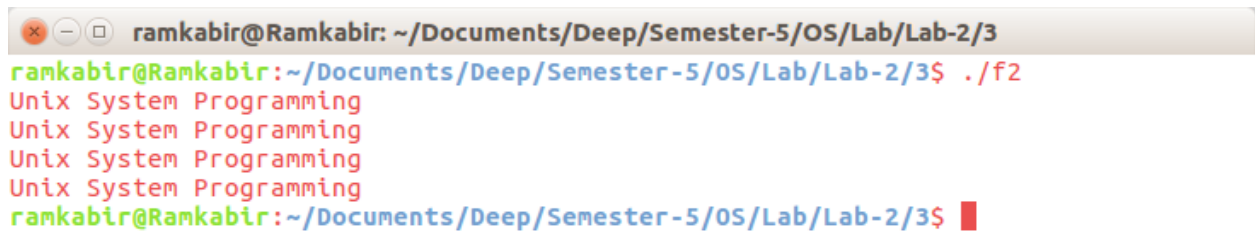
```
ramkabir@Ramkabir: ~/Documents/Deep/Semester-5/OS/Lab/Lab-2/3
ramkabir@Ramkabir:~/Documents/Deep/Semester-5/OS/Lab/Lab-2/3$ ./f1
Unix System Programming
Unix System Programming
ramkabir@Ramkabir:~/Documents/Deep/Semester-5/OS/Lab/Lab-2/3$
```

Code 2:- f2.c

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

int main(void)
{
    fork();
    fork();
    printf("Unix System Programming\n");
    exit(0);
}
```

Screenshot:-

A screenshot of a terminal window. The title bar shows the user 'ramkabir' and the directory '~/Documents/Deep/Semester-5/OS/Lab/Lab-2/3'. The terminal content shows the command './f2' being executed, which results in four lines of output: 'Unix System Programming'. The prompt 'ramkabir@Ramkabir:' is visible at the end of each line.

```
ramkabir@Ramkabir: ~/Documents/Deep/Semester-5/OS/Lab/Lab-2/3
ramkabir@Ramkabir:~/Documents/Deep/Semester-5/OS/Lab/Lab-2/3$ ./f2
Unix System Programming
Unix System Programming
Unix System Programming
Unix System Programming
ramkabir@Ramkabir:~/Documents/Deep/Semester-5/OS/Lab/Lab-2/3$
```

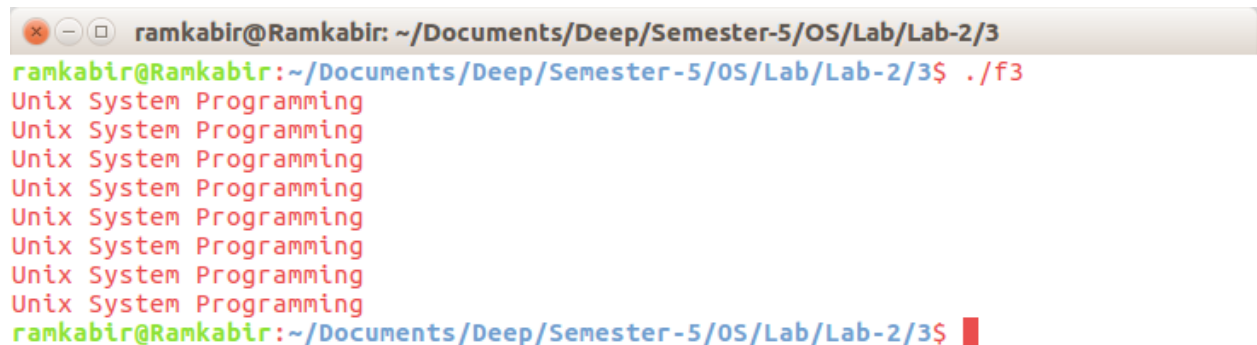
Code 3:- f3.c

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

int main(void)
{
    fork();
}
```

```
fork();
fork();
printf("Unix System Programming\n");
exit(0);
}
```

Screenshot:-



```
ramkabir@Ramkabir: ~/Documents/Deep/Semester-5/OS/Lab/Lab-2/3
ramkabir@Ramkabir:~/Documents/Deep/Semester-5/OS/Lab/Lab-2/3$ ./f3
Unix System Programming
Unix System Programming
Unix System Programming
Unix System Programming
Unix System Programming
Unix System Programming
Unix System Programming
Unix System Programming
ramkabir@Ramkabir:~/Documents/Deep/Semester-5/OS/Lab/Lab-2/3$
```

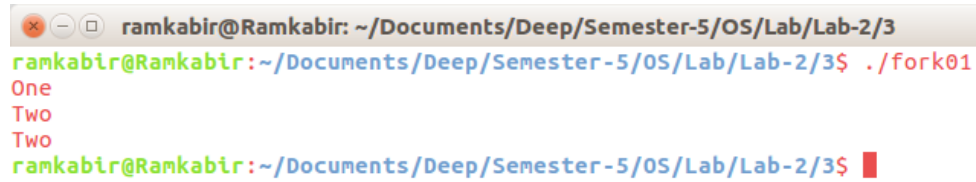
Code 4:- fork01.c

```
#include <sys/types.h>
#include <unistd.h> /* for fork() */

main()
{
    pid_t pid; /*holds process-id in parent*/

    printf("One\n");
    pid=fork();
    printf("Two\n");
}
```

Screenshot:-



```
ramkabir@Ramkabir: ~/Documents/Deep/Semester-5/OS/Lab/Lab-2/3
ramkabir@Ramkabir:~/Documents/Deep/Semester-5/OS/Lab/Lab-2/3$ ./fork01
One
Two
Two
ramkabir@Ramkabir:~/Documents/Deep/Semester-5/OS/Lab/Lab-2/3$
```

Code 5:- fork02.c

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


void main(void)
{
    pid_t pid; /*holds process-id in parent*/
    char answer[1];

    printf("Just One Process so far\n");
    pid=fork(); /* create new process */

    printf("\nPress Enter");
    gets(answer);

    if(pid == 0)
        printf("I am the child");
    else if(pid > 0)
        printf(" I am the parent, child has pid %d\n", pid);
    else
        printf("Fork returned error code, no child\n");
}
```

Screenshot:-



```
ramkabir@Ramkabir: ~/Documents/Deep/Semester-5/OS/Lab/Lab-2/3
ramkabir@Ramkabir:~/Documents/Deep/Semester-5/OS/Lab/Lab-2/3$ ./fork02
Just One Process so far

Press Enter
Press Enter
I am the parent, child has pid 6564
ramkabir@Ramkabir:~/Documents/Deep/Semester-5/OS/Lab/Lab-2/3$
```

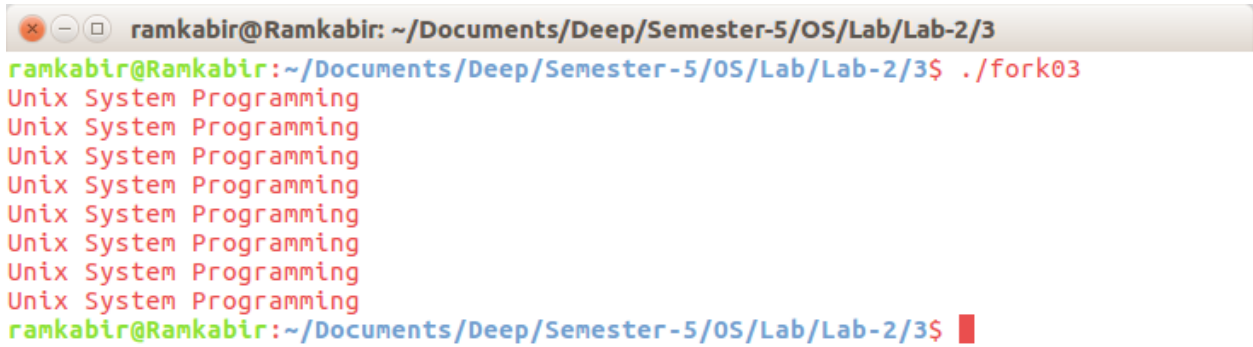
Code 6:- fork03.c

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

void doit(void)
{
    pid_t pid;
    fork();
    fork();
    printf("Unix System Programming\n");
    return;
}

int main(void)
{
    doit();
    printf("Unix System Programming\n");
    exit(0);
}
```

Screenshot:-



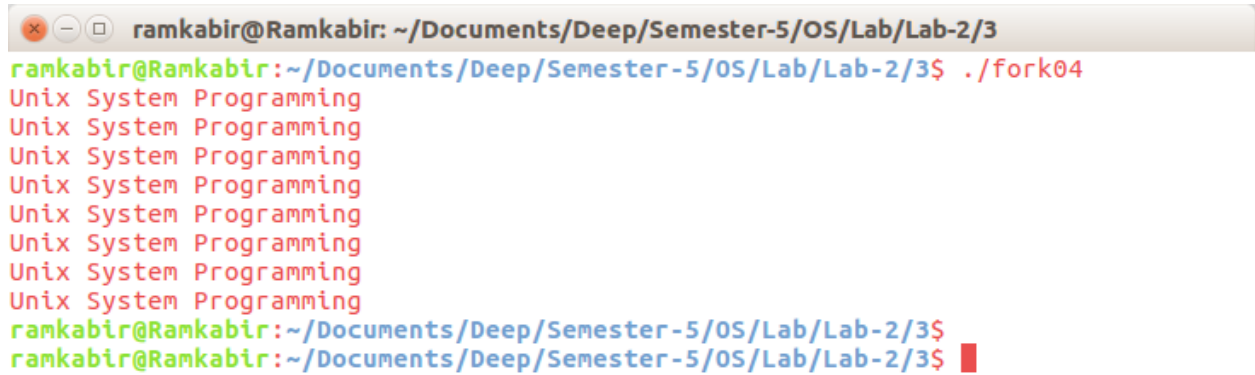
```
ramkabir@Ramkabir: ~/Documents/Deep/Semester-5/OS/Lab/Lab-2/3
ramkabir@Ramkabir:~/Documents/Deep/Semester-5/OS/Lab/Lab-2/3$ ./fork03
Unix System Programming
Unix System Programming
Unix System Programming
Unix System Programming
Unix System Programming
Unix System Programming
Unix System Programming
Unix System Programming
ramkabir@Ramkabir:~/Documents/Deep/Semester-5/OS/Lab/Lab-2/3$
```

Code 7:- fork04.c

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

int main(void)
{
    int index;
    for (index = 1; index < 4; index++)
        fork();
    printf("Unix System Programming\n");
    exit(0);
}
```


Screenshot:-



```
ramkabir@Ramkabir: ~/Documents/Deep/Semester-5/OS/Lab/Lab-2/3
ramkabir@Ramkabir:~/Documents/Deep/Semester-5/OS/Lab/Lab-2/3$ ./fork04
Unix System Programming
Unix System Programming
Unix System Programming
Unix System Programming
Unix System Programming
Unix System Programming
Unix System Programming
Unix System Programming
ramkabir@Ramkabir:~/Documents/Deep/Semester-5/OS/Lab/Lab-2/3$
ramkabir@Ramkabir:~/Documents/Deep/Semester-5/OS/Lab/Lab-2/3$
```

4).

Create one child process and make sure that child process runs first and then parent executes and vice versa (if it is possible).

Code:-Parent Process before Child Process

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

int main()
{
    pid_t pid;
    int pi,status;
    char buf[100];
```

```

bzero(buf, strlen(buf));

pid=fork();

//pi=wait(&status);

if(pid<0)
{
    printf("\n Error ");
    exit(1);
}
else if(pid==0)
{
    sprintf(buf, "\n\nHello I am the child process ");
    write(1, buf, strlen(buf));

    pi=getpid();

    bzero(buf, strlen(buf));

    sprintf(buf, "\nMy pid is: %d\n", pi);
    write(1, buf, strlen(buf));

    exit(0);
}
else
{
    sprintf(buf, "\n\nHello I am the parent process ");
    write(1, buf, strlen(buf));

    pi=getpid();

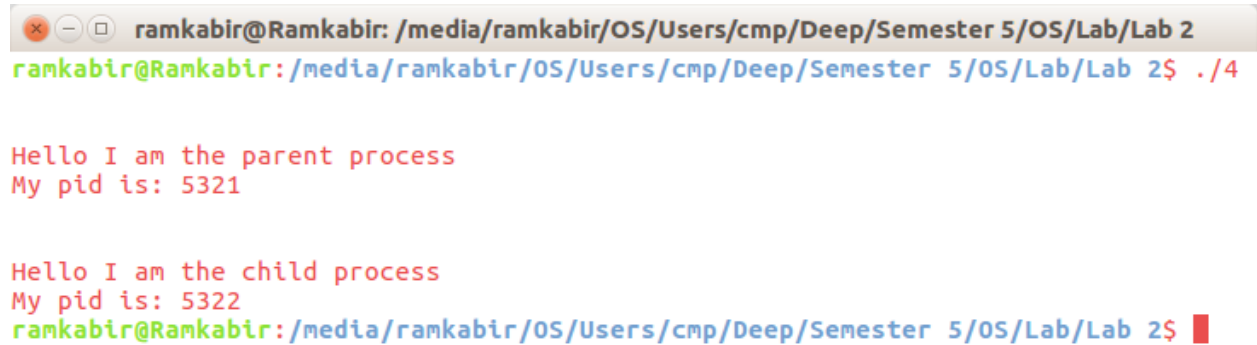
    bzero(buf, strlen(buf));

    sprintf(buf, "\nMy pid is: %d\n", pi);
    write(1, buf, strlen(buf));

    exit(0);
}
//getpid() to retrieve the process ID assigned to this
process;
return 0;
}

```

Screenshot:-

A terminal window with a title bar showing 'ramkabir@Ramkabir: /media/ramkabir/OS/Users/cmp/Deep/Semester 5/OS/Lab/Lab 2'. The prompt is 'ramkabir@Ramkabir: /media/ramkabir/OS/Users/cmp/Deep/Semester 5/OS/Lab/Lab 2\$./4'. The output shows the parent process printing 'Hello I am the parent process' and 'My pid is: 5321', followed by the child process printing 'Hello I am the child process' and 'My pid is: 5322'.

```
ramkabir@Ramkabir: /media/ramkabir/OS/Users/cmp/Deep/Semester 5/OS/Lab/Lab 2$ ./4
Hello I am the parent process
My pid is: 5321

Hello I am the child process
My pid is: 5322
ramkabir@Ramkabir: /media/ramkabir/OS/Users/cmp/Deep/Semester 5/OS/Lab/Lab 2$
```

Code:-Child Process before Parent Process

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

int main()
{
    pid_t pid;
    int pi,status;
    char buf[100];

    bzero(buf,strlen(buf));

    pid=fork();

    pi=wait(&status);

    if(pid<0)
```

```
{
    printf("\n Error ");
    exit(1);
}
else if(pid==0)
{
    sprintf(buf, "\n\nHello I am the child process ");
    write(1, buf, strlen(buf));

    pi=getpid();

    bzero(buf,strlen(buf));

    sprintf(buf, "\nMy pid is: %d\n", pi);
    write(1, buf, strlen(buf));

    exit(0);
}
else
{

    sprintf(buf, "\n\nHello I am the parent process ");
    write(1, buf, strlen(buf));

    pi=getpid();

    bzero(buf,strlen(buf));

    sprintf(buf, "\nMy pid is: %d\n", pi);
    write(1, buf, strlen(buf));

    exit(0);
}
//getpid() to retrieve the process ID assigned to this process;
    return 0;
}
```

Screenshot:-



```
ramkabir@Ramkabir: /media/ramkabir/OS/Users/cmp/Deep/Semester 5/OS/Lab/Lab 2
ramkabir@Ramkabir:/media/ramkabir/OS/Users/cmp/Deep/Semester 5/OS/Lab/Lab 2$ ./4

Hello I am the child process
My pid is: 5346

Hello I am the parent process
My pid is: 5345
ramkabir@Ramkabir:/media/ramkabir/OS/Users/cmp/Deep/Semester 5/OS/Lab/Lab 2$
```

5).

Create one child process and make it as zombie. (Use sleep(time);).

Code:-

```
//ps -e -o pid,ppid,stat,cmd

#include <stdlib.h>
#include <sys/types.h>
#include <unistd.h>
#include <stdio.h>
#include <string.h>

int main()
{
    pid_t pid;

    char buf[50];
```

```

bzero(buf, strlen(buf));

pid = fork ();

if (pid != 0)
{
    pid=getpid();

    sprintf(buf, "\nParent pid is: %d\n", pid);
    write(1, buf, strlen(buf));

    pause();    //Pausing the parent
}
else
{
    sprintf(buf, "\nChild pid is: %d\n", pid);
    write(1, buf, strlen(buf));

    sleep(2);    //Letting child live for 2 seconds before
becoming Zombie

    exit (0);
}

return 0;
}

```

Screenshot:-

```

ramkabir@Ramkabir: ~/Documents/Deep/Semester-5/OS/Lab/Lab-2
ramkabir@Ramkabir:~/Documents/Deep/Semester-5/OS/Lab/Lab-2$ ./5
Parent pid is: 6252
child pid is: 0

ramkabir@Ramkabir: ~
5097 3010 Sl /usr/bin/perl /usr/bin/shutter
5235 3010 Sl /usr/lib/gvfs/gvfsd-network --spawner :1.4 /org/gtk/gvfs/exec_s
5257 3010 Sl /usr/lib/gvfs/gvfsd-dnssd --spawner :1.4 /org/gtk/gvfs/exec_spa
5522 3010 Sl gedit /home/ramkabir/Documents/Deep/Semester-5/OS/Lab/Lab-2/5.c
5537 3010 Rl /usr/lib/gnome-terminal/gnome-terminal-server
5543 5537 Ss bash
5588 5537 Ss bash
5628 2 S [kworker/u8:1]
5711 3010 Sl /usr/lib/x86_64-linux-gnu/notify-osd
5720 2 S [kworker/2:2]
5724 2461 S /sbin/dhclient -d -q -sf /usr/lib/NetworkManager/nm-dhcp-helper
5898 3010 Sl /usr/lib/firefox/firefox
5991 2 S [kworker/1:0]
6000 2 S [kworker/u8:0]
6020 2 S [kworker/0:2]
6096 2 S [kworker/3:1]
6144 3010 Sl evince /home/ramkabir/Documents/Deep/Semester-5/OS/Lab/Lab-2/c
6161 2 S [kworker/1:1]
6180 2 S [kworker/0:0]
6244 2 S [kworker/u8:2]
6252 5543 S+ ./5
6253 6252 Z+ [5] <defunct>
6254 5588 R+ ps -e -o pid,ppid,stat,cmd
ramkabir@Ramkabir:~$

```