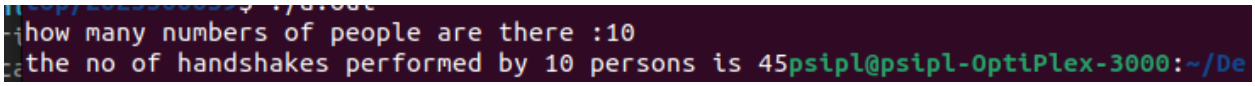
Name	Debjit Ghosal
UID no.	2023300065
Experiment No.	6

AIM:	**Apply the concept of recursion to solve a given problem**
Program 1	
PROBLEM STATEMENT :	Write a recursive function to count the number of handshakes if there are 'n' no. of people in a room and each person shakes hands with the other exactly once
ALGORITHM:	Function handshakes(n); If n equals 2 return 1 else return handshake(n-1)+n-1 function main num=0 print”enter the number of people are there ” input num’ num = handshakes(n) the no of handshakes performed by n people are num return 0

PROGRAM:	#include<stdio.h>
	int handshake(int n)
	{
	if(n==2)
	return 1;
	else
	return handshake(n-1)+(n-1);
	
	}
	
	int main()
	{
	int n,d;

	printf("how many numbers of people are there :");
	scanf("%d",&n);
	d=handshake(n);
	printf("the no of handshakes performed by %d persons is %d",n,d);
	}

RESULT:

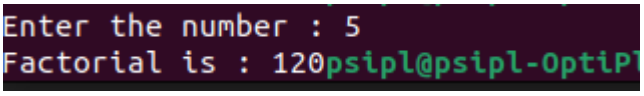


Program 2	
PROBLEM STATEMENT :	Write a recursive function to find the factorial of a number and test it

ALGORITHM/ Pseudo code :	function fact(num):
	if num equal 1:
	return 1
	else:
	return fact(num - 1) * num
	
	function main():
	num = 0
	factorial = 1
	print "Enter number to find factorial of: "
	input num
	factorial = fact(num)
	print "Factorial of", num, "is", factorial
	return 0

PROGRAM:	#include<stdio.h>
	
	int fact(int n)
	{
	if(n<=0)
	return 1;
	else
	return fact(n-1)*n; ;

	}
	int main ()
	{
	int n;
	printf("Enter the number : ");
	scanf("%d",&n);
	int s=fact(n);
	printf("Factorial is : %d \n",s);
	
	return 0;
	}

RESULT:	
Program 3	
PROBLEM STATEMENT:	Write a recursive function which returns the nth term of the fibonacci series. Call it from main() to find the 1st n numbers of the fibonacci series.

ALGORITHM:	function fibonacci(num):
	if num equals 1 or num equals 2:
	return 1
	else:
	return fiboonacci(n-1)+ Fibonacci(n-2)
	
	function main():
	n=0
	d=0
	num=0
	printf("enter the term of the fibonacci");
	d= fibonacci(n)
	print"the term of fibonacci series is n"
	return 0

PROGRAM:	#include<stdio.h>
	
	int fib(int n)
	{
	if(n==1)

```
                return 1;
        else if(n==2)
                return 1;
                return fib(n-1)+fib(n-2);       ;

}
int main ()
{
int n;
printf("Enter the fibonacci term : ");
scanf("%d",&n);
int s=fib(n);
printf("The fibonacci series is : %d",s);


return 0;
}#include<stdio.h>

int fib(int n)
{
                if(n==1)
                return 1;
        else if(n==2)
                return 1;
                return fib(n-1)+fib(n-2);       ;

}
int main ()
{
int n;
printf("Enter the fibonacci term : ");
scanf("%d",&n);
int s=fib(n);
printf("The fibonacci series is : %d",s);


return 0;
}#include<stdio.h>

int fib(int n)
```

```
{
                if(n==1)
                return 1;
        else if(n==2)
                return 1;
                return fib(n-1)+fib(n-2);       ;

}
int main ()
{
int n;
printf("Enter the fibonacci term : ");
scanf("%d",&n);
int s=fib(n);
printf("The fibonacci series is : %d",s);


return 0;
}
```

RESULT:

```
Enter the fibonacci term : 4
The fibonacci series is : 3psipl@ps
```

Program 4

PROBLEM STATEMENT: Write a recursive function to find the sum of the digits of a number.

ALGORITHM:

```
 function sum(n):
if n divided by 10 equals 0:
    return n
Else:
   return n modulo 10 + sum(n divided by 10)


function main():

 num = 0
    print "Enter a num"
     input num
   print "Sum of digits of", num, "is", sum(num)
    return 0
```

PROGRAM:

```
#include<stdio.h>
int sum(int n)
{
        if(n<10)
                return n;
        else
                return n%10+sum(n/10);
}

int main()
{
int n,d;
printf("Enter the digits: ");
scanf("%d",&n);
sum(n);
printf("The sum of digits is :%d",n,sum(n));
}
```

RESULT:

```
Enter the digits: 12345
The sum of digits is :15
```

Program 5

PROBLEM STATEMENT:

Given a number n, print following pattern without using any loop.

input: n = 16

Output: 16 11 6 1 -4 1 6 11 16

Input: n = 10

Output: 10 5 0 5 10

ALGORITHM:

Function mirror pattern
If n>0
  Print n
  Print"n-5"
  Print n
 Else
  Print n

Main function:

	int n print "Enter the max term : " input n pattern (num) print new line

PROGRAM:	#include<stdio.h>
	void print(int n)
	{
	if(n>0)
	{
	printf("%d ",n);
	print(n-5);
	printf("%d ",n);
	}
	else
	printf("%d ",n);
	
	}
	int main()
	{
	int n;
	printf("Enter the max term : ");
	scanf("%d",&n);
	print(n);
	return 0
	;
	
	}

RESULT:

```
psipl@psipl-OptiPlex-3000:~/Desk
Enter the max term : 16
 16  11  6  1  -4  1  6  11  16
```

```
Enter the max term : 15
 15  10  5  0  5  10  15 psipl
```

CONCLUSION:	We learnt the concept of applying Recursion to solve problems