

DEV BHOOMI INSTITUTE OF TECHNOLOGY

Department of Computer Science and Engineering

Year: 1stSemester:1st



C Programming Lab-


LAB MANUAL

INDEX

S.No.	Experiments	Software Used	Remark
1.	To find the sum of individual digits of a given number		
2.	To print the Fibonacci series for 1 to n value		
3.	To print a prime numbers up to 1 to n		
4.	To calculate the sum. $Sum=1-x2/2!+ x4/4!- x6/6!+ x8/8!- x10/10!$		
5.	Programs that use recursive function to find the factorial of a given integer.		
6.	Program that use non recursive function to find the factorial of a given integer.		
7.	To find the GCD of two given integers by using the recursive function		
8.	Two integer operands and one operator form user, performs the operationand then prints the result.(Consider the operators +,-,*, /, % and use Switch Statement)		
9.	To find both the largest and smallest number in a list of integers.		
10.	To perform the addition of two matrices		
11.	Functions to insert a sub string into given main string from a given position.		
12.	To generate Pascal's triangle		
13.	To convert the given binary number to 2's complement		
14.	To read the two complex numbers and perform the addition and multiplicationof these two numbers.		
15.	Program which copies one file to another		

DEV BHOOMI INSTITUTE OF TECHNOLOGY

LAB MANUAL

	Course Name :Fundamentals of Computer & Programming Lab.	EXPERIMENT NO. 1	
	Course Code : PCS 101 Faculty : Mr. ShubhashishGoswami	Branch: CSE	Semester: I

Objective:

To find the sum of individual digits of a given number.

Description:

Sum of the individual digits means adding all the digits of a numberEx: 123 sum of digits is $1+2+3=6$

Algorithm:

Step 1: start
Step 2: read n
Step 3: initialize the s=0
Step 4: if $n < 0$ goto Step 7
Step 5: if $n \neq 0$ goto Step 6 else goto step 7
Step 6: store $n \% 10$ value in p
Add p value to s
Assign $n / 10$ value to n
Goto Step 5
Step 7: print the output
Step 8: stop

Program:

```
#include<stdio.h>
main()
{
int n,s,p;
clrscr();
printf("enter the vaue for n:\n");
scanf("%d",&n);
s=0;
if(n<0)
printf("The given number is not valid");
else
{
```

```
while(n!=0) /* check the given value =0 or not */
{
p=n%10;
n=n/10;
s=s+p;
}
printf("sum of individual digits is %d",s);
}
getch();
}
```

Output:


1.Enter the value for n: 333
Sum of individual digits is 9
2.Enter the value for n: 4733
Sum of individual digits is 17
3. Enter the value for n: -111
The given number is not valid

Outcome:

- 1.To understand basic structure of a C program.
2. To understand handling of number system in a C program

DEV BHOOMI INSTITUTE OF TECHNOLOGY

LAB MANUAL

	Course Name :Fundamentals of Computer & Programming Lab.	EXPERIMENT NO. 2	
	Course Code : PCS 101 Faculty : Mr. ShubhashishGoswami	Branch: CSE	Semester: I

Objective 2:To print the Fibonacci series for 1 to n value

Description

A fibonacci series is defined as follows

The first term in the sequence is 0

The second term in the sequence is 1

The sub sequent terms 1 found by adding the preceding two terms in the sequence

Formula: let t_1, t_2, \dots, t_n be terms in fibonacci sequence

$t_1=0, t_2=1, t_n=t_{n-2}+t_{n-1} \dots$ where $n>2$

Algorithm:

Step 1: start

Step 2: initialize the $a=0, b=1$

Step 3: read n

Step 4: if $n== 1$ print a go to step 7. else goto step 5

Step 5: if $n== 2$ print a, b go to step 7 else print a,b

Step 6: initialize $i=3$

i) if $i \leq n$ do as follows. If not goto step 7

$c=a+b$

print c

$a=b$

$b=c$

increment I value

goto step 6(i)

Step 7: stop

Program:

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

```
void main()
```

```
{
```

```
int a,b,c,n,i;
```

```
clrscr();
```

```
printf("enter n value");
scanf("%d",&n);a=0;
b=1;
if(n==1)
printf("%d",a);
else
if(n==2)
printf("%d%d",a,b);
else
{
printf("%d%d",a,b);
//LOOP WILL RUN FOR 2 TIME LESS IN SERIES AS THESE WAS
PRINTED IN ADVANCE
for(i=2;i<n;i++)
{
c=a+b;
printf("%d",c);
a=b;
b=c;
}
getch();
}
}
```

Output:


```
1. Enter n value : 5
0 1 1 2 3
2. Enter n value : 7
0 1 1 2 3 5 8
```

Outcome:

- 1 .To learn basic looping constructs in a C program,

DEV BHOOMI INSTITUTE OF TECHNOLOGY

LAB MANUAL

	Course Name : Fundamentals of Computer & Programming Lab.	EXPERIMENT NO. 3	
	Course Code : PCS 101 Faculty : Mr. ShubhashishGoswami	Branch: CSE	Semester: I

Objective 3: To print a prime numbers up to 1 to n

Description:

Prime number is a number which is exactly divisible by one and itself only Ex: 2, 3, 5, 7,

Algorithm:

Step 1: start
Step 2: read n
Step 3: initialize i=1, c=0
Step 4: if i<=n goto step 5
If not goto step 10
Step 5: initialize j=1
Step 6: if j<=1 do as the follow. If no goto step 7
 i) if i%j==0 increment c
 ii) increment j
 iii) goto Step 6
Step 7: if c== 2 print i
Step 8: increment i
Step 9: goto step 4
Step 10: stop

Program:

```
#include<stdio.h>
#include<conio.h>
void main()
{
    int n,i,fact,j;
    clrscr();
    printf("enter the number:");
    scanf("%d",&n);
    for(i=1;i<=n;i++)
    {
```

```
fact=0;
//THIS LOOP WILL CHECK A NO TO BE PRIME NO. OR NOT.
for(j=1;j<=i;j++)
{
if(i%j==0)
fact++;
}
if(fact==2)
printf("\n %d",i);
}
getch();
}
```

Output:

Enter the number : 5

2 3 5

Enter the number : 10

2 3 5 7

Enter the number : 12


2 3 5 7 11

Outcome:

- 1 .To understand basic looping constructs in C programs
2. To handle various mathematical problems.

DEV BHOOMI INSTITUTE OF TECHNOLOGY

LAB MANUAL

	Course Name : Fundamentals of Computer & Programming Lab.	EXPERIMENT NO. 4	
	Course Code : PCS 101 Faculty : Mr. ShubhashishGoswami	Branch: CSE	Semester: I

Objective 4: To calculate the sum. $\text{Sum} = 1 - x^2/2! + x^4/4! - x^6/6! + x^8/8! - x^{10}/10!$

Algorithm: main program:

- Step 1: start
- Step 2: declare x,i,n,s=0,c
- Step 3: read x value
- Step 4: for i=0 , n=0; i<=10; i=i+2, n++ goto step 5
- Step 5: $s = s + (\text{pow}(-1, n) * \text{pow}(x, i) / \text{fact}(i))$
- Step 6: print s value
- Step 7: stop

Sub program:

- Step 1: while x!=0 goto Step 2
- Step 2: $y = y + x$; $x = x - 1$
- Step 3: return y
- Step 4: return to main program

Program:

```
#include<stdio.h>
#include<math.h>
long fact(int);
void main()
{
    int x,i,n;
    float s=0,c;
    clrscr();
    printf("\n enter the value of x\t");
    scanf("%d",&x);
    /*perform the looping operation*/
    for(i=0,n=0;i<=10;i=i+2,n++)
        s=s+(pow(-1,n)*pow(x,i)/fact(i));
    printf("\n the result is %f",s);
    getch();
}
/* calling sub program*/
long fact(int x)
```

DEV BHOOMI INSTITUTE OF TECHNOLOGY

```
{  
longint y=1;  
while(x!=0)  
{  
y=y*x;  
x--;  
}  
return y;  
}
```


Output:

1.Enter the value of x : 1
The result is 0.540302
2 Enter the value of x: 2
The result is -0.416155

Outcome:

1 .To handle and calculate series based problems

LAB MANUAL

	Course Name :Fundamentals of Computer & Programming Lab.	EXPERIMENT NO. 5	
	Course Code : PCS 101 Faculty : Mr. ShubhashishGoswami	Branch: CSE	Semester: I

Objective 5: Programs that use recursive function to find the factorial of a given integer.

Description:

Factorial of a number is nothing but the multiplication of numbers from a given number to 1

Algorithm: main program

Step 1: start
Step 2: read n
Step 3: call sub program as f=fact(n)
Step 4: print f value
Step 5: stop

Sub program:

Step 1: initialize the f
Step 2: if n= 0 or n == 1 return 1 to main program if not goto step 3
Step 3: return n*fact(n-1) to main program

Program:

```
#include<stdio.h>
#include<conio.h>
int fact(int n)
{
int f;
if((n==0)||(n==1)) // check the condition for the n value
return(n);
else
f=n*fact(n-1); //calculate the factorial of n
return(f);
}
void main()
{
int n;
clrscr();
printf("enter the number :");
scanf("%d",&n);
```

```
printf("factorial of number%d",fact(n));  
getch();  
}
```

Output:


1. Enter the number : 5
Factorial of number: 120
2. Enter the number : 3
Factorial of number: 6

Outcome:

- 1 .To understand recursion based concepts in C programming.

DEV BHOOMI INSTITUTE OF TECHNOLOGY

LAB MANUAL

	Course Name :Fundamentals of Computer & Programming Lab.	EXPERIMENT NO. 6	
	Course Code : PCS 101 Faculty : Mr. ShubhashishGoswami	Branch: CSE	Semester: I

Objective 6: Program that use non recursive function to find the factorial of a given integer.

Description:

Factorial of a number is nothing but the multiplication of numbers from a given number to 1
Ex: $5! = 5*4*3*2*1 = 120$

Algorithm: main program

- Step 1: start
- Step 2: read n
- Step 3: call the sub program fact(n)
- Step 4: print the f value
- Step 5: stop

Sub program:

- Step 1: initialize the f=1
- Step 2: if $n==0$ or $n=1$ return 1 to main program. If not goto step 3
- Step 3: perform the looping operation as follows
For $i=1$ $i \leq n$; $i++$
- Step 4: $f=f*i$
- Step 5: return f value to the main program

Program:

```
#include<stdio.h>
#include<conio.h>
int fact(int n) //starting of the sub program
{
int f=1,i;
if((n==0)||(n==1)) // check the condition for n value
return(1);
else
for(i=1;i<=n;i++) // perform the looping operation for calculating the factorial
f=f*i;
return(f);
}
void main()
```

```
{  
int n;  
clrscr();  
printf("enter the number :");  
scanf("%d",&n);  
printf("factoria of number%d",fact(n));  
getch();  
}
```

Output:


1. Enter the number: 7
Factorial of number: 5040
2. Enter the number: 6
Factorial of number: 720
3. Enter the number: 8
Factorial of number: -25216

Outcome:

1 .To handle recursive problems without recursion based function call.

DEV BHOOMI INSTITUTE OF TECHNOLOGY

LAB MANUAL

	Course Name : Fundamentals of Computer & Programming Lab.	EXPERIMENT NO. 7	
	Course Code : PCS 101 Faculty : Mr. ShubhashishGoswami	Branch: CSE	Semester: I

Objective 7: To find the GCD of two given integers by using the recursive function

Description:

GCD means Greatest Common Divisor. i.e the highest number which divides the given number

Ex: GCD(12,24) is 12

Formula: $GCD = \text{product of numbers} / \text{LCM of numbers}$

Algorithm: main program

Step 1: start

Step 2: read a,b

Step 3: call the sub program GCD(a,b) for print the value

Step 4: stop

Sub program:

Step 1: if $n > m$ return GCD(n,m)

Step 2: if $n == 0$ return m else goto step 3

Step 3: return GCD (n,m%n)

Step 4: return to main program

Program:

```
#include<stdio.h>
#include<conio.h>
intgcdrecursive(intm,int n) // starting of the sub program
{
if(n>m)
returngcdrecursive(n,m);
if(n==0)
return m;
else
returngcdrecursive(n,m%n); // return to the main program
}
void main()
{
inta,b,igcd;
clrscr();
```

```
printf("enter the two numbers whose gcd is to be found:");  
scanf("%d%d",&a,&b);  
printf("GCD of a,b is %d",gcdrecursive(a,b)); // return to the sub program  
getch();  
}
```

Output:


1. enter the two numbers whose gcd is to be found:5,25
GCD of a,b is : 5
2. enter the two numbers whose gcd is to be found:36,54
GCD of a,b is : 18
3. enter the two numbers whose gcd is to be found:11,13
GCD of a,b is : 1

Outcome:

- 1 .To understand basic structure of a C program.
2. To understand handling of number system in a C program

DEV BHOOMI INSTITUTE OF TECHNOLOGY

LAB MANUAL

	Course Name :Fundamentals of Computer & Programming Lab.	EXPERIMENT NO. 8	
	Course Code : PCS 101 Faculty : Mr. ShubhashishGoswami	Branch: CSE	Semester: I

Objective 8:Two integer operands and one operator form user, performs the operation and then prints the result.(Consider the operators +,-,*,/, % and use Switch Statement)

Description:

To take the two integer operands and one operator from user to perform the some arithmetic operations by using the following operators like +,-,*,/, % ,Ex: 2+3=5

Algorithm:

Step 1: Start
Step 2: Read the values of a,b and operator
Step 3: if the operator is '+' then
R=a+b
Go to step 8
Break
Step 4: Else if the operator is '-' then
R=a-b
Go to step 8
Step 5: Else if the operator is '*' then
R=a*b
Go to step 8
Step 6: Else if the operator is '/' then
R=a/b
Go to step 8
Step 7: Else if the operator is '%' then
R=a%b
Go to step 8
Step 8: write R
Step 9:End

Program:

```
#include<stdio.h>
main()
{
char op;
```

```

float a,b,c;
clrscr();
printf("enter two operands:");
scanf("%d%d",&a,&b);
printf("enter an operator:");
scanf(" %c",&op);
switch(op) // used to select particular case from the user
{ case '+':printf("sum of two numbers %
2d %
2d is: %
d",a,b,a+b);
break;
case '-':printf("subtraction of two numbers %2d %2d is:
%d",a,b,a-b);
break;
case '*':printf("product of two numbers %2d %2d is:
%d",a,b,a*b);
break;
case '/':printf("quotient of two numbers %2d %2d is:
%d",a,b,a/b);
break;
case '%':printf("remainder of two numbers %2d %2d is:
%d",a,b,c);
break;
default:printf("please enter correct operator");
break;
}
getch();
}

```

Output:

```

1.enter two operands:2 3
enter an operator:+
sum of two numbers 2 3 is: 5
2.enter two operands:3 4
enter an operator: -
subtraction of two numbers 3 4 is: -1
3.enter two operands:3 5
enter an operator:*
product of two numbers 3 5 is: 15
4.enter two operands:5 2
enter an operator:/
quotient of two numbers 5 2 is: 2
5. enter two operands:5 2
enter an operator:%
remainder of two numbers 5 2 is: 1


```

Outcome:

- 1 .To understand basic structure of a C program.
2. To understand handling of number system in a C program

DEV BHOOMI INSTITUTE OF TECHNOLOGY

LAB MANUAL

	Course Name : Fundamentals of Computer & Programming Lab.	EXPERIMENT NO. 9	
	Course Code : PCS 101 Faculty : Mr. ShubhashishGoswami	Branch: CSE	Semester: I

Objective 09: To find both the largest and smallest number in a list of integers

Description:

This program contains n number of elements, in these elements we can find the largest and smallest numbers and display these two numbers

Algorithm:

Step 1: start
Step 2: read n
Step 3: initialize i=0
Step 4: if i<n do as follows. If not goto step 5
Read a[i]
Increment i
Goto step 4
Step 5: min=a[0], max=a[0]
Step 6: initialize i=0
Step 7: if i<n do as follows. If not goto step 8
If a[i]<min
Assign min=a[i]
Increment i goto Step 7
Step 8: print min,max
Step 9: stop

Program:

```
#include<stdio.h>
void main()
{
int a[10],i,n,min,max;
clrscr();
printf("enter the array size:");
scanf("%d",&n);
printf("Enter the elements of array");
```

```
for(i=0;i<n;i++) // read the elements of an array
scanf("%d",&a[i]);
min=a[0];
max=a[0];
for(i=0;i<n;i++)// read the elements of an array
{
if(a[i]<min)// check the condition for minimum value
min=a[i];
if(a[i]>max)//check the condition for maximum value
max=a[i];
}
printf("maximum value is:%d\n",max);
printf("minimum value is:%d\n",min);
getch();
}
```

Output:

```
1.enter the array size:4
Enter the elements of array 36 13 2 45
maximum value is:45
minimum value is:2
2.enter the array size:5
Enter the elements of array 6 2 1 3 8
maximum value is:8
minimum value is:1
3.enter the array size:5
Enter the elements of array-6 9 -9 2 5
maximum value is:9
minimum value is:-9
```

Outcome:

- 1 .To understand basic structure of a C program.
2. To understand handling of number system in a C program

Objective 10:To perform the addition of two matrices

Description:

To program takes the two matrixes of same size and performs the addition an also takes the two matrixes of different sizes and checks for possibility of multiplication and perform multiplication if possible.

Algorithm:

- Step 1: start
- Step 2: read the size of matrices A,B – m,n
- Step 3: read the elements of matrix A
- Step 4: read the elements of matrix B
- Step 5: select the choice for you want. If you select case 1 then gotomatricaddition. Else goto Step 7.
- Step 6: print Sum of matrix A and B
- Step 7: if you select case 2 then goto matrix multiplication
- Step 8: check if n=p, if not print matrices can not be multiplied
- Step 9: Otherwise perform the multiplication of matrices
- Step 10: Print the resultant matrix
- Step 11: Stop

Program:

```
#include<stdio.h>
void main()
{
int ch,i,j,m,n,p,q,k,r1,c1,a[10][10],b[10][10],c[10][10];
clrscr();
printf("*****");
printf("\n\t\tMENU");
printf("\n*****");
printf("\n[1]ADDITION OF TWO MATRICES");
printf("\n[2]MULTIPLICATION OF TWO MATRICES");
printf("\n[0]EXIT");
printf("\n*****");
printf("\n\tEnter your choice:\n");
scanf("%d",&ch);
if(ch<=2 &ch>0)
{
printf("Valid Choice\n");
}
switch(ch)
{
```

```

case 1:
printf("Input rows and columns of A & B Matrix:");
scanf("%d%d",&r1,&c1);
printf("Enter elements of matrix A:\n");
for(i=0;i<r1;i++)
{
for(j=0;j<c1;j++)
scanf("%d",&a[i][j]);
}
printf("Enter elements of matrix B:\n");
for(i=0;i<r1;i++)
{
for(j=0;j<c1;j++)
scanf("%d",&b[i][j]);
}
printf("\n =====Matrix Addition===== \n");
for(i=0;i<r1;i++)
{
For(j=0;j<c1;j++)
printf("%5d",a[i][j]+b[i][j]);
52
printf("\n");
}
break;
case 2:
printf("Input rows and columns of A matrix:");
scanf("%d%d",&m,&n);
printf("Input rows and columns of B matrix:");
scanf("%d%d",&p,&q);
if(n==p)
{
printf("matrices can be multiplied\n");
printf("resultant matrix is %d*%d\n",m,q);
printf("Input A matrix\n");
read_matrix(a,m,n);
printf("Input B matrix\n");
/*Function call to read the matrix*/
read_matrix(b,p,q);
/*Function for Multiplication of two matrices*/
printf("\n =====Matrix Multiplication===== \n");
for(i=0;i<m;++i)
for(j=0;j<q;++j)
{
c[i][j]=0;
for(k=0;k<n;++k)
c[i][j]=c[i][j]+a[i][k]*b[k][j];
}
}
}

```

```

}
printf("Resultant of two matrices:\n");
write_matrix(c,m,q);
}
/*end if*/
else
{
printf("Matrices cannot be multiplied.");
}
/*end else*/
break;
case 0:
printf("\n Choice Terminated");
exit();
break;
default:
printf("\n Invalid Choice");
53
}
getch();
}
/*Function read matrix*/
intread_matrix(int a[10][10],intm,int n)
{
inti,j;
for(i=0;i<m;i++)
for(j=0;j<n;j++)
scanf("%d",&a[i][j]);
return 0;
}
/*Function to write the matrix*/
intwrite_matrix(int a[10][10],intm,int n)
{
inti,j;
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
printf("%5d",a[i][j]);
printf("\n");
}
return 0;
}

```

Output:

```

1.
*****
MENU

```



```
*****
[1]ADDITION OF TWO MATRICES
[2]MULTIPLICATION OF TWO MATRICES
[0]EXIT
*****
```

Enter your choice:

1

Valid Choice

Input rows and columns of A & B Matrix:2

2

Enter elements of matrix A:

222

54

2

Enter elements of matrix B:

2222

====Matrix Addition====

4 4

4 4

MENU

```
[1]ADDITION OF TWO MATRICES
[2]MULTIPLICATION OF TWO MATRICES
[0]EXIT
*****
```

Enter your choice:

2

Valid Choice

Input rows and columns of A matrix:2

3

Input rows and columns of B matrix:2

2

Matrices cannot be multiplied.

MENU

```
[1]ADDITION OF TWO MATRICES
[2]MULTIPLICATION OF TWO MATRICES
[0]EXIT
*****
```

Enter your choice:

2

Valid Choice

Input rows and columns of A matrix:2

2

Input rows and columns of B matrix:2

2

matrices can be multiplied

resultant matrix is 2*2

Input A matrix

222

55

2

Input B matrix

2222

=====Matrix Multiplication=====

Resultant of two matrices:

8 8


8 8

Outcome:

- 1 .To understand basic structure of a C program.
2. To understand handling of number system in a C program

DEV BHOOMI INSTITUTE OF TECHNOLOGY

LAB MANUAL

	Course Name : Fundamentals of Computer & Programming Lab.	EXPERIMENT NO. 11	
	Course Code : PCS 101 Faculty : Mr. ShubhashishGoswami	Branch: CSE	Semester: I

Objective 11: Functions to insert a sub string into given main string from a given position

Description:

In this program we need to insert a string into another string from a specified position.

Algorithm:

- Step 1: start
- Step 2: read main string and sub string
- Step 3: find the length of main string(r)
- Step 4: find length of sub string(n)
- Step 5: copy main string into sub string
- Step 6: read the position to insert the sub string(p)
- Step 7: copy sub string into main string from position p-1
- Step 8: copy temporary string into main string from position p+n-1
- Step 9: print the strings
- Step 10: stop

Program:

```
#include <stdio.h>
#include <conio.h>
#include <string.h>
void main()
{
char a[10];
char b[10];
char c[10];
int p=0,r=0,i=0;
int t=0;
int x,g,s,n,o;
clrscr();
puts("Enter First String:");
gets(a);
```

```

puts("Enter Second String:");
gets(b);
printf("Enter the position where the item has to be inserted: ");
scanf("%d",&p);
r = strlen(a);
n = strlen(b);
i=0;
// Copying the input string into another array
while(i<= r)
{
c[i]=a[i];
i++;
}
o = n+r;
// Adding the sub-string
for(i=p;i<s;i++)
{
x = c[i];
if(t<n)
{
a[i] = b[t];
t=t+1;
}
a[o]=x;
o=o+1;
}
printf("%s", a);
getch();
}

```

Output:


```

1.enter first string:
computer
2.enter second string:
gec
3.enter the position where the item has to be inserted:3
comgecomputer

```

DEV BHOOMI INSTITUTE OF TECHNOLOGY

LAB MANUAL

	Course Name : Fundamentals of Computer & Programming Lab.	EXPERIMENT NO. 12	
	Course Code : PCS 101 Faculty : Mr. ShubhashishGoswami	Branch: CSE	Semester: I

Objective 12: To generate Pascal's triangle

Description:

Pascal's triangle which is used for a coefficient in the equation in polynomials.

Algorithm:

- Step 1: Start
- Step 2: Initialize m=0
- Step 3: Read n
- Step 4: If m<n goto step 5. if not goto step 12
- Step 5: initialize i=40-m
- Step 6: If i>0 is true do as follows. If not goto step 7
 - i. print white space
 - ii. decrement i
 - iii. goto Step 6
- Step 7: Initialize j=0
- Step 8: If j=m do as follows. If not goto Step 10
 - i) if(j==0||m==0)
 - ii) Initialize b=1 if not b=b*(m-j+1)/j
 - iii) Print white space, b .
 - iv) Goto Step 9
- Step 9: increment j, goto Step 8
- Step 10: print new line control
- Step 11: increment m, goto step 4
- Step 12: Stop

Program:

```
#include<stdio.h>
#include<conio.h>
void main()
{
    inti,p,r,x,binom=1;
    clrscr();
```

```
printf("enter the how many lines to print");
scanf("%d",&p);
i=0;
while(i<p) // check the condition
{
for(r=40-i;r>0;r--) // perform the looping operation until 0
printf(" ");
for(x=0;x<=i;x++)
{
if((x==0)||(i==0)) // check the condition
binom=1;
else
binom=binom*(i-x+1)/x;
printf("%d",binom);
printf(" ");
}
printf("\n");
i++;
}
getch();
}
```


Output:

enter the how many lines to print5

```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
```

DEV BHOOMI INSTITUTE OF TECHNOLOGY

LAB MANUAL

	Course Name : Fundamentals of Computer & Programming Lab.	EXPERIMENT NO. 13	
	Course Code : PCS 101 Faculty : Mr. ShubhashishGoswami	Branch: CSE	Semester: I

Objective 13: To convert the given binary number to 2's complement

Description:

In this program the given binary number is first converted the numbers 0 to 1 and 1 to 0. And finally add the 1 to the converted number. Then we will get the 2's complement number.

Algorithm: main program

Step 1: Start

Step 2: declare the subprogram "complement(char *a)"

Step 3: initialize the variable i

Step 4: read the binary number

Step 5: perform the loop operation. if it is true then follows. if not goto step 7

i) for(i=0;a[i]!='\0';i++)

ii) if(a[i]!='0' && a[i]!='1') then displayed the number is not valid. enter the correct number.

iii) Exit the loop

Step 6: call sub program 'complement(a)'

Step 7: stop

Sub program:

Step 1: initialize the variable i, c=0, b[160]

Step 2: l=strlen(a)

Step 3: perform the loop operation. if it is true then follows. if not goto

i) for(i=l-1; i>=0; i--)

ii) if(a[i]=='0') then b[i]='1' else

iii) b[i]='0'

Step 4: for(i=l-1; i>=0; i--) is true

i) if(i==l-1) then

ii) if(b[i]=='0') then b[i]='1' else

iii) b[i]='0', c=1 if not goto step 5

Step 5: if(c==1 && b[i]=='0') is true then

i) b[i]='1', c=0 if not goto Step 6
Step 6: if(c==1&&b[i]=='1') then b[i]='0',c=1
Step 7: displayed b[l]='\0'
Step 8: print b and return to main program

Program:

```
#include <stdio.h>
#include<conio.h>
void complement (char *a);
void main()
{
char a[16];
inti;
clrscr();
printf("Enter the binary number");
gets(a);
for(i=0;a[i]!='\0'; i++)
{
if (a[i]!='0' && a[i]!='1')
{
printf("The number entered is not a binary number. Enter the
correct number");
exit(0);
}
} complement(a);
getch();
A
If c==1
&&
b[i]==1
B[i]='0]
C=1
true
B
C
B[i]='\0'
Print 'b'
Return to
main
program
93
}
void complement (char *a)
{
int l, i, c=0;
```



```

char b[16];
l=strlen(a);
for (i=l-1; i>=0; i--)
{
if (a[i]=='0')
b[i]='1';
else
b[i]='0';
}
for(i=l-1; i>=0; i--)
{
if(i==l-1)
{
if (b[i]=='0')
b[i]='1';
else
{
b[i]='0';
c=1;
}
}
else
{
if(c==1 && b[i]=='0')
{
b[i]='1';
c=0;
}
else if (c==1 && b[i]=='1')
{
b[i]='0';
c=1;
}
}
}
b[l]='\0';
printf("The 2's complement is %s", b);
}

```

Output:


```

1.Enter the binary number101010
The 2's complement is 010110
Enter the binary number11111
The 2's complement is 00001
Enter the binary number2222
The number entered is not a binary number. Enter the correct number

```


DEV BHOOMI INSTITUTE OF TECHNOLOGY

LAB MANUAL

	Course Name : Fundamentals of Computer & Programming Lab.	EXPERIMENT NO. 14	
	Course Code : PCS 101 Faculty : Mr. ShubhashishGoswami	Branch: CSE	Semester: I

Objective 14: To read the two complex numbers and perform the addition and multiplication of these two numbers.

Description:

In this program the complex number means it contains the two parts .first one is real part and second one is imaginary part(2+3i).by taking these two complex numbers we can perform the addition and multiplication operation.

Algorithm:

Step 1: Start
Step 2: declare structure for complex numbers
Step 3: read the complex number
Step 4: read choice
Step 5: if choice=1 then addition operation will perform and it contains following steps
i) $w.\text{realpart} = w1.\text{realpart} + w2.\text{realpart};$
ii) $w.\text{imgpart} = w1.\text{imgpart} + w2.\text{imgpart};$ goto step 4
Step 6: if choice=2 then multiplication operation will perform and it contains following steps
i) $w.\text{realpart} = (w1.\text{realpart} * w2.\text{realpart}) - (w1.\text{imgpart} * w2.\text{imgpart});$
ii) $w.\text{imgpart} = (w1.\text{realpart} * w2.\text{imgpart}) + (w1.\text{imgpart} * w2.\text{realpart});$ goto step 4
Step 7: if choice=0 then exit operation will perform
Step 8: if $w.\text{imgpart} > 0$ then print $\text{realpart} + \text{imgpart}$ else
Print realpart .
Step 9: Stop

Program:

```
#include<stdio.h>
#include<math.h>
void arithmetic(intopern);
struct comp
{
```

```

double realpart;
double imgpart;
};
void main()
{
int opern;
clrscr();
printf("\n\n \t\t\t***** MAIN MENU *****");
printf("\n\n Select your option: \n 1 : ADD\n 2 : MULTIPLY\n 0 : EXIT \n\n\t\t Enter
your Option [ ]\b\b");
scanf("%d",&opern);
if(opern>2)
{
printf("invalid option");
}
else
{
switch(opern)
{
case 0:
exit(0);
case 1:
case 2:
arithmetic(opern);
default:
main();
}
}
getch();
}
void arithmetic(int opern)
{
struct comp w1, w2, w;
printf("\n Enter two Complex Numbers (x+iy):\n Real Part of First Number:");
scanf("%lf",&w1.realpart);
printf("\n Imaginary Part of First Number:");
scanf("%lf",&w1.imgpart);
printf("\n Real Part of Second Number:");
scanf("%lf",&w2.realpart);
103
printf("\n Imaginary Part of Second Number:");
scanf("%lf",&w2.imgpart);
switch(opern)
{
/*addition of complex number*/
case 1:

```

```

w.realpart = w1.realpart+w2.realpart;
w.imgpart = w1.imgpart+w2.imgpart;
break;
/*multiplication of complex number*/
case 2:
w.realpart=(w1.realpart*w2.realpart)-(w1.imgpart*w2.imgpart);
w.imgpart=(w1.realpart*w2.imgpart)+(w1.imgpart*w2.realpart);
break;
}
if (w.imgpart>0)
printf("\n Answer = %lf+%lfi",w.realpart,w.imgpart);
else
printf("\n Answer = %lf%lfi",w.realpart,w.imgpart);
getch();
main();
}

```

Output:


```

***** MAIN MENU *****
Select your option:
1 : ADD
2 : MULTIPLY
0 : EXIT
Enter your Option [ 1]
Enter two Complex Numbers (x+iy):
Real Part of First Number:2
Imaginary Part of First Number:2
Real Part of Second Number:2
Imaginary Part of Second Number:2
Answer = 4.000000+4.000000i

```

DEV BHOOMI INSTITUTE OF TECHNOLOGY

LAB MANUAL

	Course Name : Fundamentals of Computer & Programming Lab.	EXPERIMENT NO. 1	
	Course Code : PCS 101 Faculty : Mr. ShubhashishGoswami	Branch: CSE	Semester: I

Objective 15: Program which copies one file to another

Description:

In this program we have to use the file functions to perform the copy operation from one file to another file.

Algorithm:

- Step 1: Start
- Step 2: read command line arguments
- Step 3: check if no of arguments =3 or not. If not print invalid no of arguments
- Step 4: open source file in read mode
- Step 5: if NULL pointer, then print source file can not be open
- Step 6: open destination file in write mode
- Step 7: if NULL pointer, then print destination file can not be open
- Step 8 : read a character from source file and write to destination file until EOF
- Step 9: Close source file and destination file
- Step 10: Stop

Program:

```
#include <stdio.h>
#include <conio.h>
#include <process.h>
void main(int argc, char *argv[])
{
FILE *fs,*ft;
charch;
clrscr();
if(argc!=3)
{
puts("Invalid number of arguments.");
exit(0);
```

```

}
fs = fopen(argv[1],"r");
if(fs==NULL)
{
puts("Source file cannot be opened.");
exit(0);
}
ft = fopen(argv[2],"w");
if (ft==NULL) // check the condition if the file pointer is NULL or not
{
puts("Target file cannot be opened.");
fclose(fs);
exit(0);
}
while(1)
{
ch=fgetc(fs);
if (ch==EOF) // check the condition if the file is end or not
break;
else
fputc(ch,ft);
}
fclose(fs);
fclose(ft);
getch();
}

```

Output:

```

source.c
this is source text
ouput.c
Command line arguments
source.couput.c
source.c
this is source text
ouput.c
this is source text
Command line arguments
source.c
Invalid number of arguments.

```

Outcome:

- 1 . Construct programs that demonstrate effective use of file handling in C language.
2. To understand various file related functions.

