



COMPUTER PROGRAMMING LAB MANUAL

B. Tech., CSE

AURORA'S SCIENTIFIC TECHNOLOGICAL & RESEARCH ACADEMY
(Affiliated to JNTU Hyderabad and Approved by AICTE)
Bandlaguda, Hyderabad.

Index

S. No	Content	Page no
1	Preface	2
2	Objective and Relevance	4
3	Outcomes	5
4	Code Of Conduct	6
5	List Of Experiments	9
6	Syllabus analysis	12
7	Session Plan	16
8	Equipment Required	19
9	Experiment Write Up	26

Preface

C is a general-purpose high level language that was originally developed by Dennis Ritchie for the UNIX operating system. It was first implemented on the Digital Equipment Corporation PDP-11 computer in 1972. The UNIX operating system and virtually all UNIX applications are written in the C language. C has now become a widely used professional language for various reasons. Easy to learn, structured language, it produces efficient Programs, it can handle low-level activities. It can be compiled on a variety of computers.

Facts about C

C was invented to write an operating system called UNIX. C is a successor of B language which was introduced around 1970. The language was formalized in 1988 by the American National Standard Institute (ANSI). By 1973 UNIX OS almost totally written in C. Today C is the most widely used System Programming Language. Most of the state of the art software has been implemented using C.

Why to use C?

C was initially used for system development work, in particular the Programs that make-up the operating system. C was adopted as a system development language because it produces code that runs nearly as fast as code written in assembly language. Some examples of the use of C might be:

- Operating Systems
- Language Compilers
- Assemblers
- Text Editors
- Print Spoolers
- Network Drivers
- Modern Programs
- Data Bases
- Language Interpreters
- Utilities

C Program File

All the C Programs are written into text files with extension ".c" for example *hello.c*. You can use "vi" editor to write your C Program into a file. This tutorial assumes that you know how to edit a text file and how to write Programming instructions inside a Program file.

C Compilers

When you write any Program in C language then to run that Program you need to compile that Program using a C Compiler which converts your Program into a language understandable by a computer. This is called machine language (i.e. binary format). So before proceeding, make sure you have C Compiler available at your computer. It comes along with all flavors of UNIX and Linux. If you are working over UNIX or Linux then you can type *gcc -v* or *cc -v* and check the result. You can ask your system administrator or you can take help from anyone to identify an available C Compiler at your computer. If you don't have C compiler installed at your computer then you can use below given link to download a GNU C Compiler and use it.

Objective and Relevance

- To review basic computer systems concepts
- To be able to understand the different computing environments and their components
- To review the history of computer languages
- To be able to list and describe the classifications of computer languages
- To understand the steps in the development of a computer Program To review the system development life cycle
- To write Programs in C to solve the problems.
- To implement linear data structures such as lists, stacks, queues.
- To implement simple searching and sorting methods.

Outcomes

- This subject contributes to having students practice their writing skills with project document and report writing.
- This subject contributes to developing student critical thinking through tutorial and lab exercises on solving problems. They will also practice more in written assignments, Programming exercises, and project.
- This subject contributes to team work with group-based project for students to practice team spirit.

Code of Conduct

- Students should report to the labs concerned as per the timetable.
- Students who turn up late to the labs will in no case be permitted to perform the experiment scheduled for the day.
- After completion of the experiment, certification of the staff in-charge concerned in the observation book is necessary.
- Students should bring a notebook of about 100 pages and should enter the readings/observations/results into the notebook while performing the experiment.
- The record of observations along with the detailed experimental procedure of the experiment performed in the immediate previous session should be submitted and certified by the staff member in-charge.
- Not more than three students in a group are permitted to perform the experiment on a set up.
- The group-wise division made in the beginning should be adhered to, and no mix up of student among different groups will be permitted later.
- The components required pertaining to the experiment should be collected from Lab – in - charge after duly filling in the requisition form.
- When the experiment is completed, students should disconnect the setup made by them, and should return all the components/instruments taken for the purpose.
- Any damage of the equipment or burnout of components will be viewed seriously either by putting penalty or by dismissing the total group of students from the lab for the semester/year.
- Students should be present in the labs for the total scheduled duration.
- Students are expected to prepare thoroughly to perform the experiment before coming to Laboratory.
- Procedure sheets/data sheets provided to the students' groups should be maintained neatly and are to be returned after the experiment.

List of Experiments Syllabus as per JNTUH**Week 1**

- a) Write a C Program to find the factorial of a positive integer.
- b) Write a C Program to find the roots of a quadratic equation.

Week 2

- a) Write a C Program to determine if the given number is a prime number or not.
- b) A Fibonacci sequence is defined as follows: the first and second terms in the sequence are 0 and 1. Subsequent terms are found by adding the preceding two terms in the sequence. Write a C Program to generate the first n terms of the sequence.

Week 3

- a) Write a C Program to construct a pyramid of numbers.
- b) Write a C Program to calculate the following Sum: $\text{Sum} = 1 - X^2 / 2! + X^4 / 4! - X^6 / 6! + X^8 / 8! - X^{10} / 10!$.

Week 4

- a) The least common multiple (LCM) of two positive integers a and b is the smallest integer that is evenly divisible by both a and b. Write a C Program that reads two integers and calls LCM (a, b) function that takes two integer arguments and returns their LCM. The LCM (a, b) function should calculate the least common multiple by calling the GCD (a, b) function and using the following relation: $\text{LCM}(a, b) = ab / \text{GCD}(a, b)$
- b) Write a C Program that reads two integers n and r to compute the nCr value using the following relation: ${}^nC_r(n, r) = n! / r! (n-r)!$. Use a function for computing the factorial value of an integer.

Week 5

- a) Write C Program that reads two integers x and n and calls a recursive function to compute X^n .
- b) Write a C Program that uses a recursive function to solve the Towers of Hanoi problem.
- c) Write a C Program that reads two integers and calls a recursive function to compute nC_r value.

Week 6

- a) Write a C Program to generate all the prime numbers between 1 and n, where n is a value Supplied by the user using Sieve of Eratosthenes Algorithm.
- b) Write a C Program that uses non recursive function to search for a Key value in a given list of integers. Use linear search method.

Week 7

- a) Write a menu-driven C Program that allows a user to enter n numbers and then choose between finding the smallest, largest, sum, or average. The menu and all the choices are to be functions. Use a switch statement to determine what action to take. Display an error message if an invalid choice is entered.
- b) Write a C Program that uses non recursive function to search for a Key value in a given sorted list of integers. Use binary search method.

Week 8

- a) Write a C Program that implements the Bubble sort method to sort a given list of integers in ascending order.
- b) Write a C Program that reads two matrices and uses functions to perform the following:
 - 1. Addition of two matrices
 - 2. Multiplication of two matrices

Week 9

- a) Write a C Program that uses functions to perform the following operations:
 - 1. To insert a sub-string into a given main string from a given position.
 - 2. To delete n characters from a given position in a given string.
- b) Write a C Program that uses a non-recursive function to determine if the given string is a palindrome or not.

Week 10

- a) Write a C Program to replace a substring with another in a given line of text.
- b) Write a C Program that reads 15 names each of up to 30 characters, stores them in an array, and uses an array of pointers to display them in ascending (i.e. alphabetical) order.

Week 11

- a) 2's complement of a number is obtained by scanning it from right to left and complementing all the bits after the first appearance of a 1. Thus 2's complement of 11100 is 00100. Write a C Program to find the 2's complement of a binary number.
- b) Write a C Program to convert a positive integer to a roman numeral. Ex. 11 is converted to XI.

Week 12

- a) Write a C Program to display the contents of a file to standard output device.
- b) Write a C Program which copies one file to another, replacing all lowercase characters with their uppercase equivalents.

Week 13

- a) Write a C Program to count the number of times a character occurs in a text file. The file name and the character are supplied as command-line arguments.
- b) Write a C Program to compare two files, printing the first line where they differ.

Week 14

- a) Write a C Program to change the nth character (byte) in a text file. Use fseek function.
- b) Write a C Program to reverse the first n characters in a file. The file name and n are specified on the command line. Use fseek function.

Week 15

- a) Write a C Program to merge two files into a third file (i.e., the contents of the first file followed by those of the second are put in the third file).
- b) Define a macro that finds the maximum of two numbers. Write a C Program that uses the macro and prints the maximum of two numbers.

Reference Books:

1. Mastering C, K.R. Venugopal and S.R. Prasad, TMH Publishers.
2. Computer Programming in C, V. Rajaraman, PHI.
3. Programming in C, Stephen G. Kochan, Fourth Edition, Pearson Education.
4. C++: The complete reference, H. Schildt, TMH Publishers.

Syllabus Analysis:

S.No	Name of the Experiment	Unit No	Text /Reference Books
1	a) Write a C Program to find the factorial of a positive integer. b) Write a C Program to find the roots of a quadratic equation.	1	Text Book 1
2	a) Write a C Program to determine if the given number is a prime number or not. b) A Fibonacci sequence is defined as follows: the first and second terms in the sequence are 0 and 1. Subsequent terms are found by adding the preceding two terms in the sequence. Write a C Program to generate the first n terms of the sequence.	1	Text Book 1
3	a) Write a C Program to construct a pyramid of numbers. b) Write a C Program to calculate the following Sum: $\text{Sum} = 1 - X^2/2! + X^4/4! - X^6/6! + X^8/8! - X^{10}/10!$	1	Text Book 1
4	a) The least common multiple (LCM) of two positive integers a and b is the smallest integer that is evenly divisible by both a and b. Write a C Program that reads two integers and calls LCM (a, b) function that takes two integer arguments and returns their LCM. The LCM (a, b) function should calculate the least common multiple by calling the GCD (a, b) function and using the following relation: $\text{LCM}(a, b) = ab / \text{GCD}(a, b)$ b) Write a C Program that reads two integers n and r to compute the ncr value using the following relation: ${}^nC_r(n, r) = n! / r! (n-r)!$. Use a function for computing the factorial value of an integer.	2	Text Book 1
5	a) Write C Program that reads two integers x and n and calls a recursive function to compute X^n . b) Write a C Program that uses a recursive function to solve the Towers of Hanoi problem. c) Write a C Program that reads two integers and calls a recursive function to compute nC_r value.	2	Text Book 1
6	a) Write a C Program to generate all the prime numbers between 1 and n, where n is a value Supplied by the user using Sieve of Eratosthenes Algorithm. b) Write a C Program that uses non recursive function to search for a Key value in a given list of integers. Use linear search method.	2	Text Book 1

7	<p>a) Write a menu-driven C Program that allows a user to enter n numbers and then choose between finding the smallest, largest, sum, or average. The menu and all the choices are to be functions. Use a switch statement to determine what action to take. Display an error message if an invalid choice is entered.</p> <p>b) Write a C Program that uses non recursive function to search for a Key value in a given sorted list of integers. Use binary search method.</p>	2	Text Book 1
8	<p>a) Write a C Program that implements the Bubble sort method to sort a given list of integers in ascending order.</p> <p>b) Write a C Program that reads two matrices and uses functions to perform the following:</p> <ol style="list-style-type: none"> 1. Addition of two matrices 2. Multiplication of two matrices 	2	Text Book 1
9	<p>a) Write a C Program that uses functions to perform the following operations:</p> <ol style="list-style-type: none"> 1. To insert a sub-string into a given main string from a given position. 2. To delete n characters from a given position in a given string. <p>b) Write a C Program that uses a non-recursive function to determine if the given string is a palindrome or not.</p>	3	Text Book 1
10	<p>a) Write a C Program to replace a substring with another in a given line of text.</p> <p>b) Write a C Program that reads 15 names each of up to 30 characters, stores them in an array, and uses an array of pointers to display them in ascending (i.e. alphabetical) order.</p>	3	Text Book 1
11	<p>a) 2's complement of a number is obtained by scanning it from right to left and complementing all the bits after the first appearance of a 1. Thus 2's complement of 11100 is 00100. Write a C Program to find the 2's complement of a binary number.</p> <p>b) Write a C Program to convert a positive integer to a roman numeral. Ex. 11 is converted to XI.</p>	3	Text Book 1
12	<p>a) Write a C Program to display the contents of a file to standard output device.</p> <p>b) Write a C Program which copies one file to another, replacing all lowercase characters with their uppercase equivalents.</p>	5	Text Book 1
13	<p>a) Write a C Program to count the number of times a character occurs in a text file. The file name and the character are supplied as command-</p>	5	Text Book 1

	line arguments. b) Write a C Program to compare two files, printing the first line where they differ.		
14	a) Write a C Program to change the nth character (byte) in a text file. Use fseek function. b) Write a C Program to reverse the first n characters in a file. The file name and n are specified on the command line. Use fseek function.	5	Text Book 1
15	a) Write a C Program to merge two files into a third file (i.e., the contents of the first file followed by those of the second are put in the third file). b) Define a macro that finds the maximum of two numbers. Write a C Program that uses the macro and prints the maximum of two numbers.	5	Text Book 1

Session Plan:

S.No	Week No	Unit No	Activity	Remarks
1	1	1	a) Write a C Program to find the factorial of a positive integer. b) Write a C Program to find the roots of a quadratic equation.	JNTUH
2	2	1	a) Write a C Program to determine if the given number is a prime number or not. b) A Fibonacci sequence is defined as follows: the first and second terms in the sequence are 0 and 1. Subsequent terms are found by adding the preceding two terms in the sequence. Write a C Program to generate the first n terms of the sequence.	JNTUH
3	3	1	a) Write a C Program to construct a pyramid of numbers. b) Write a C Program to calculate the following Sum: $\text{Sum} = 1 - X^2 / 2! + X^4 / 4! - X^6 / 6! + X^8 / 8! - X^{10} / 10!$	JNTUH
4	4	2	a) The least common multiple (LCM) of two positive integers a and b is the smallest integer that is evenly divisible by both a and b. Write a C Program that reads two integers and calls LCM (a, b) function that takes two integer arguments and returns their LCM. The LCM (a, b) function should calculate the least common multiple by calling the GCD (a, b) function and using the following relation: $\text{LCM}(a, b) = ab / \text{GCD}(a, b)$ b) Write a C Program that reads two integers n and r to compute the ncr value using the following relation: ${}^nC_r(n, r) = n! / r! (n-r)!$. Use a function for computing the factorial value of an integer.	JNTUH
5	5	2	a) Write C Program that reads two integers x and n and calls a recursive function to compute X^n . b) Write a C Program that uses a recursive function to solve the Towers of Hanoi problem. c) Write a C Program that reads two integers and calls a recursive function to compute nC_r value.	JNTUH
6	6	2	a) Write a C Program to generate all the prime numbers between 1 and n, where n is a value Supplied by the user using Sieve of Eratosthenes Algorithm. b) Write a C Program that uses non recursive function to search for a Key value in a given list of integers. Use linear search method.	JNTUH

7	7	2	<p>a) Write a menu-driven C Program that allows a user to enter n numbers and then choose between finding the smallest, largest, sum, or average. The menu and all the choices are to be functions. Use a switch statement to determine what action to take. Display an error message if an invalid choice is entered.</p> <p>b) Write a C Program that uses non recursive function to search for a Key value in a given sorted list of integers. Use binary search method.</p>	JNTUH
8	8	2	<p>a) Write a C Program that implements the Bubble sort method to sort a given list of integers in ascending order.</p> <p>b) Write a C Program that reads two matrices and uses functions to perform the following:</p> <ol style="list-style-type: none"> 1. Addition of two matrices 2. Multiplication of two matrices 	JNTUH
9	9	3	<p>a) Write a C Program that uses functions to perform the following operations:</p> <ol style="list-style-type: none"> 1. To insert a sub-string into a given main string from a given position. 2. To delete n characters from a given position in a given string. <p>b) Write a C Program that uses a non-recursive function to determine if the given string is a palindrome or not.</p>	JNTUH
10	10	3	<p>a) Write a C Program to replace a substring with another in a given line of text.</p> <p>b) Write a C Program that reads 15 names each of up to 30 characters, stores them in an array, and uses an array of pointers to display them in ascending (i.e. alphabetical) order.</p>	JNTUH
11	11	3	<p>a) 2's complement of a number is obtained by scanning it from right to left and complementing all the bits after the first appearance of a 1. Thus 2's complement of 11100 is 00100. Write a C Program to find the 2's complement of a binary number.</p> <p>b) Write a C Program to convert a positive integer to a roman numeral. Ex. 11 is converted to XI.</p>	JNTUH
12	12	5	<p>a) Write a C Program to display the contents of a file to standard output device.</p> <p>b) Write a C Program which copies one file to another, replacing all lowercase characters with their uppercase equivalents.</p>	JNTUH
13	13	5	<p>a) Write a C Program to count the number of times a character occurs in a text file. The file name and the character are supplied as command-line arguments.</p>	JNTUH

			b) Write a C Program to compare two files, printing the first line where they differ.	
14	14	5	a) Write a C Program to change the nth character (byte) in a text file. Use fseek function. b) Write a C Program to reverse the first n characters in a file. The file name and n are specified on the command line. Use fseek function.	JNTUH
15	15	5	a) Write a C Program to merge two files into a third file (i.e., the contents of the first file followed by those of the second are put in the third file). b) Define a macro that finds the maximum of two numbers. Write a C Program that uses the macro and prints the maximum of two numbers.	JNTUH

Equipment Required:**Hardware:**

No. of System	:	60(IBM)
Processor	:	PIV™ 1.67 GHz
RAM	:	512 MB
Hard Disk	:	40 GB
Mouse	:	Optical Mouse
Network Interface card	:	Present

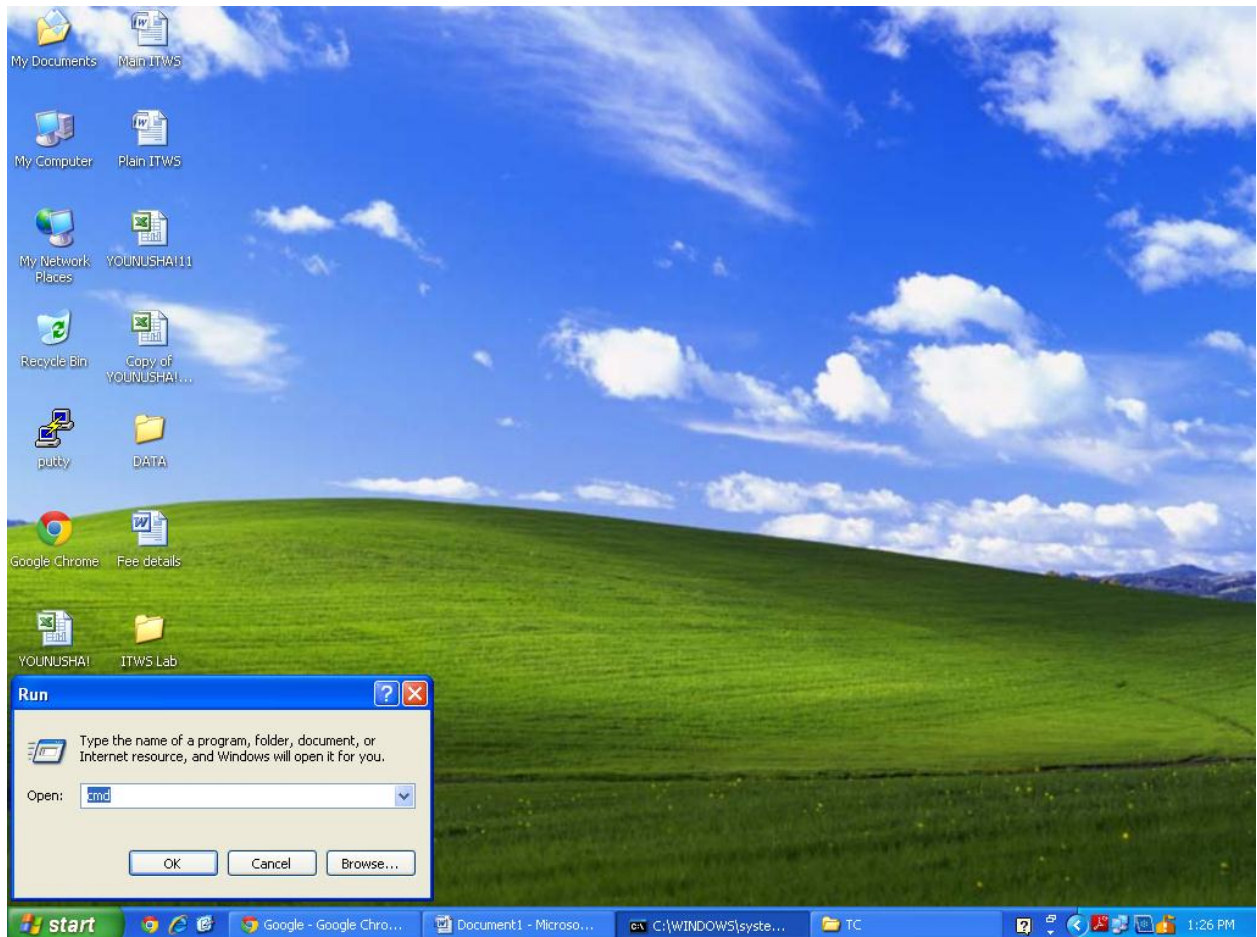
Software:

Operating System	:	Window XP
Application Software	:	Turbo C.

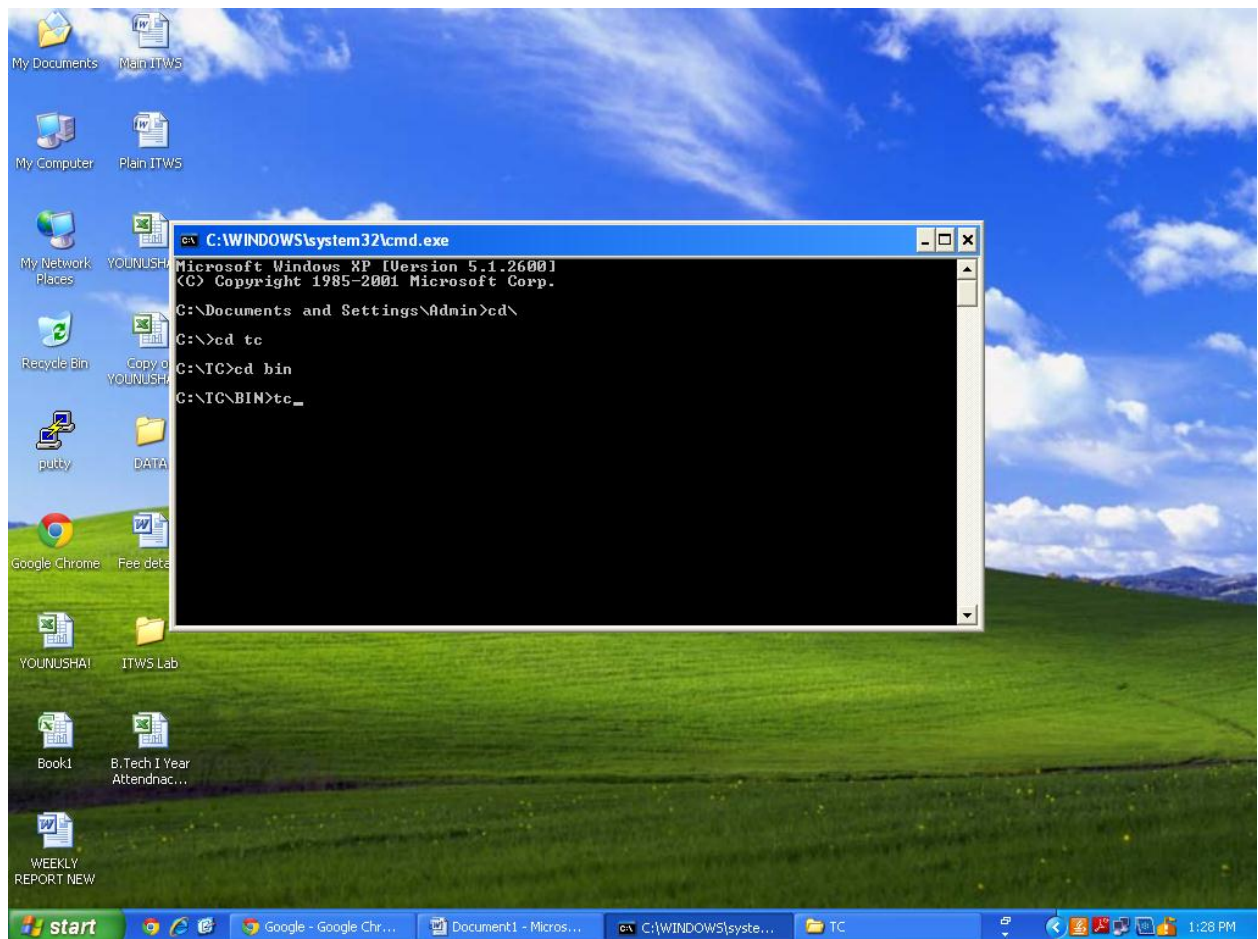
Experimental Write Up:

1 Introduction

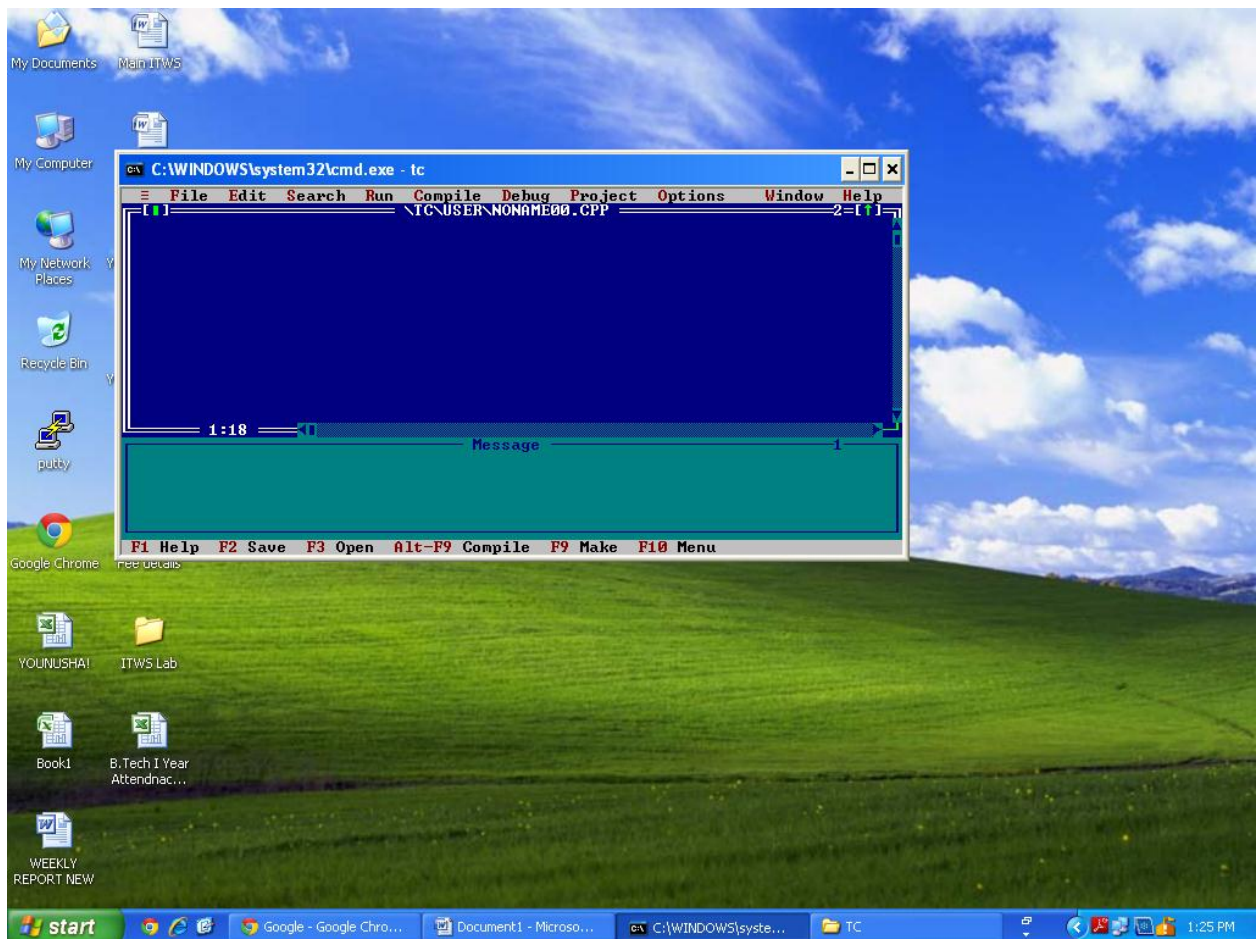
How to Open Application



Step 1: Open Command Prompt Window



Step 2: Open TC with Command Prompt Window

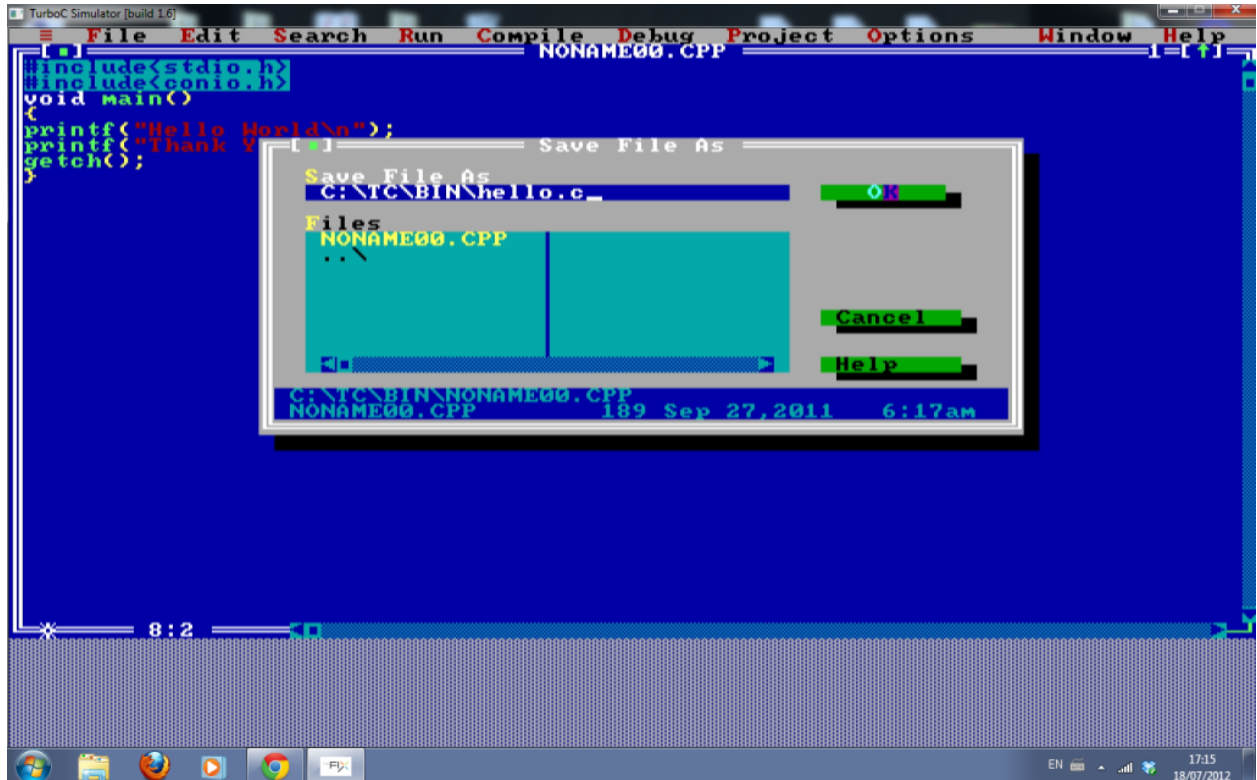


```
#include<stdio.h>
```

```
void main()
```

```
{  
    printf ("Hello World\n");  
    printf ("Thank You");  
    getch ();  
}
```

I hope that's enough for a basic explanation of the Program. If you still have doubts please ask through comments. Now let's **RUN this Program using Turbo C**. Before going into the steps, you may **SAVE** your C Program. Select "**File**" from menu -> click-> **Save**. Name the files as ->**hello.c** or some other name with a .c extension. See the screen shot below.



How to Compile a C Program in Turbo C:

The first step is compiling. Compiling makes sure your Program is **free of syntax errors**. However compiler won't check for any logical/Algorithmic errors. There is a lot of process that happens while the compiler compiles a Program – which we will discuss later in coming articles. To do compiling - **Select -> Compile** from menu and **click-> compile**. See the image below.



After compiling, you will see a dialog box as shown below. If the compilation is success – you will see a **“success”** message. Else you will see the number of errors. Both are shown using screen shots.

The screen shot of a “success” compilation

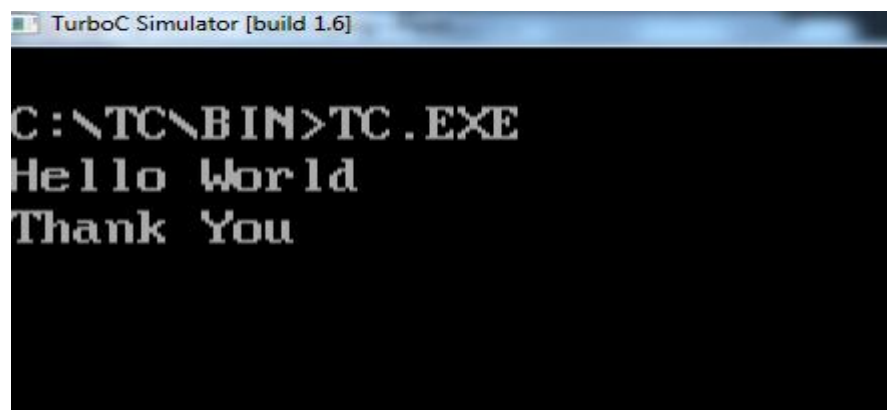


How to RUN a C Program in Turbo C compiler?

To RUN the Program – you may select ->Run from menu and click -> Run (as shown in the image below).



Now you will see the output screen as shown in the screen shot below.

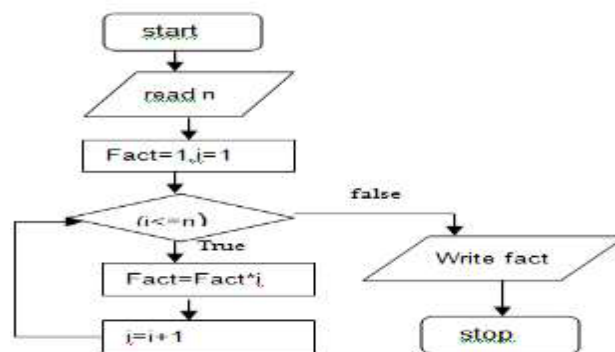


Week – 1:

(a) **Aim : Write a Program to find factorial of a positive integer.**

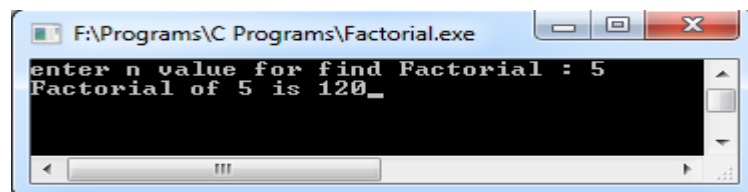
Algorithm:

Step-1: start
 Step-2: Read n
 Step-3: Initialize fact to 1 and i=1
 Step-4: while (i<=n) else goto step-6
 Step-4.1: fact=fact*i;
 Step-4.2: sum=i+1
 Step-5: goto step-4
 Step-6: print fact
 Step-7: stop

Flowchart:**Program:**

```

#include<stdio.h>
#include<conio.h>
void main()
{
    int n,fact=1,i;
    printf("enter n value for find Factorial : ");
    scanf("%d",&n);
    for(i=1;i<=n;i++)
        fact=fact*i;
    printf("Factorial of %d is %d",n,fact);
    getch();
}
  
```

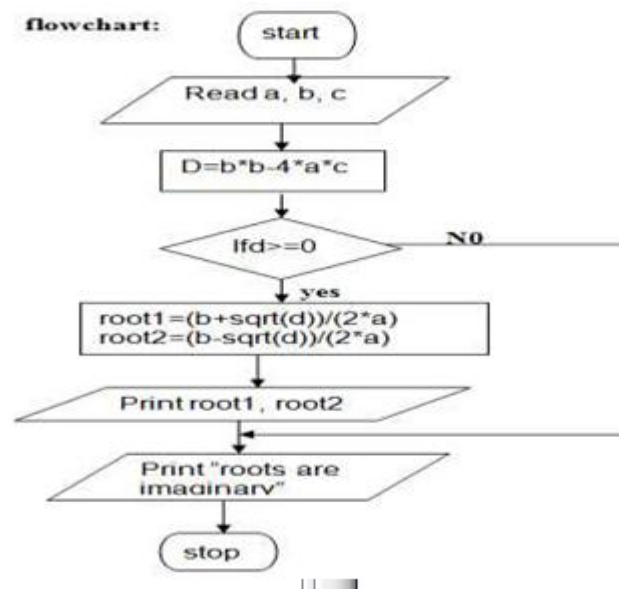
Output:

(b) **Aim :** Write a C Program to find the roots of a quadratic equation.

Algorithm:

Step-1:start
 Step-2:READ values of a , b and c ,
 Step-3:calculate value of discriminant $D=b^2-4ac$
 Step-4:if D is less than zero then goto step-6
 step4.1:1 roots $root1 = (-b+D)/2a$ and $root2 = (-b-D)/2a$
 step-5:print $root1, root2$
 step-6:print "roots are imaginary"
 step-7 :stop

Flowchart:

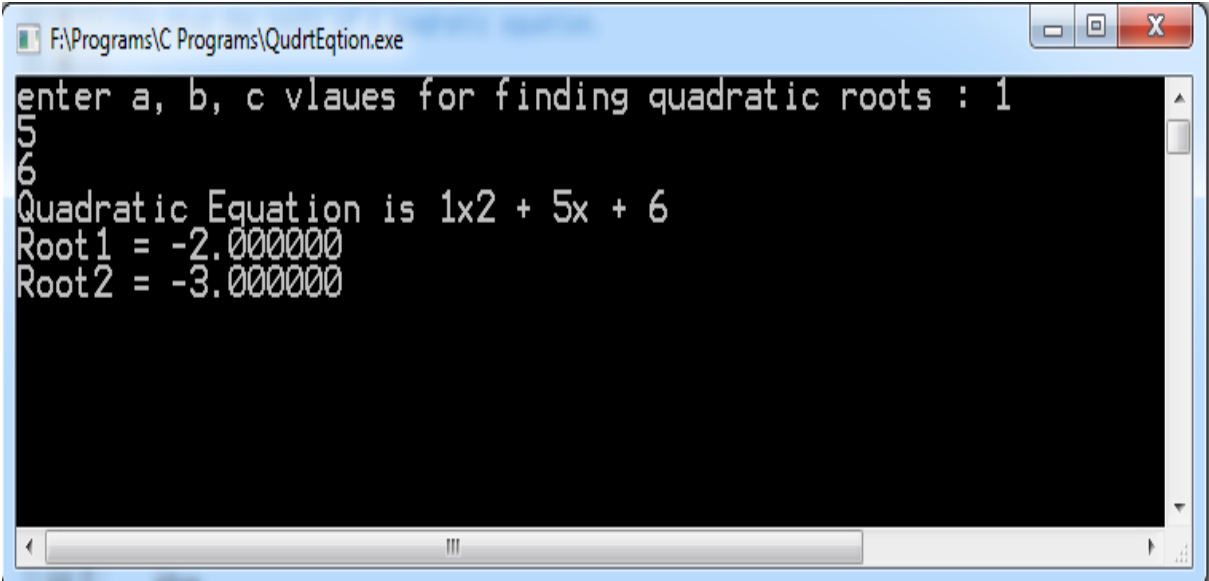


Program:

```

#include<stdio.h>
#include<conio.h>
#include<math.h>
void main()
{
    int a,b,c; float r1,r2,D;
    printf("enter a, b, c vlaues for finding quadratic roots : ");
    scanf("%d%d%d",&a,&b,&c);
    printf("Quadratic Equation is %dx2 + %dx + %d",a,b,c);
    D=b*b-4*a*c;
    if(D<0)
    {
        printf("Roots are imaginary");
    }
}
  
```

```
    else
    {
        r1 = (-b + sqrt(D)) / (2 * a);
        r2 = (-b - sqrt(D)) / (2 * a);
        printf("\nRoot1 = %f\nRoot2 = %f", r1, r2);
    }
    getch();
}
```

Output:

```
F:\Programs\C Programs\QudrtEqtion.exe
enter a, b, c vlaues for finding quadratic roots : 1
5
6
Quadratic Equation is 1x2 + 5x + 6
Root1 = -2.000000
Root2 = -3.000000
```

Viva Questions:

1. What is an Algorithm?
2. What is Flowchart?
3. What is Factorial?
4. Define preprocessor?
5. How to find the roots of quadratic equations?

Week – 2:

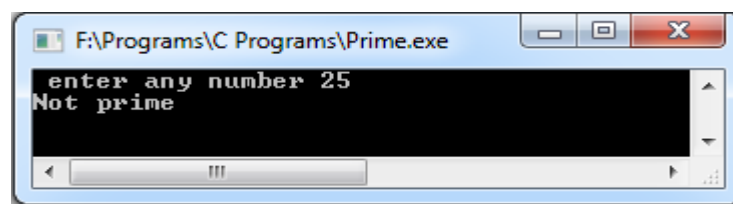
- (a) **Aim :** Write a C Program to determine if given number is prime number or not.

Algorithm:

Step1: start
Step2: read n
Step3: assign i=1,nf=0
Step4: if(i<=n) else goto step5
 Step 4.1 :if(n%i==0)
 Step 4.2:nf=nf+1
 Step 4.3:i=i+1 goto step4
Step5: if(nf==2)
 Step 4.1: write “given number is prime”
 Step 4.2: otherwise write “given number is not prime”
Step 6: stop

Program:

```
#include <stdio.h>
#include<conio.h>
void main()
{
    int n,i,nf=0;
    clrscr();
    printf(" enter any number ");
    scanf("%d",&n);
    for(i= 1; i<= n; i++)
    {
        if(n%i == 0)
            nf++;
    }
    if(nf == 2)
        printf("given number is prime");
    else
        printf("Not prime");
    getch();
}
```

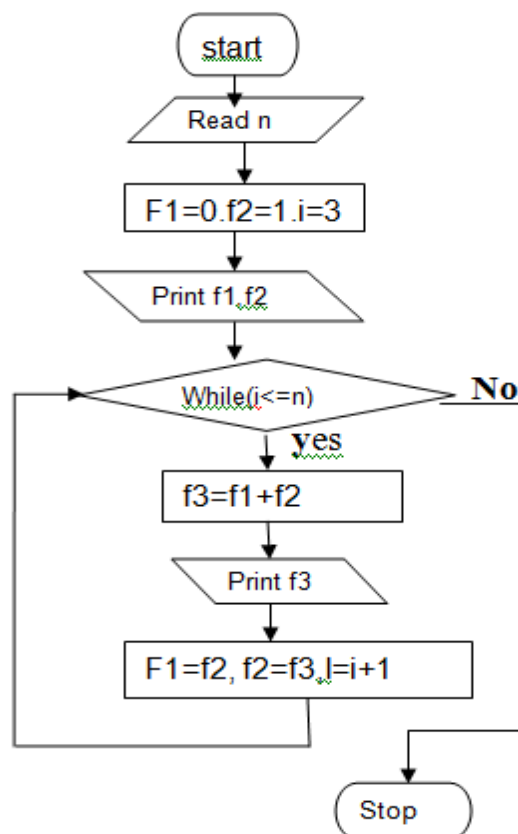
Output:

(b) **Aim :** Write a C Program to generate the first n terms of the fibonacci sequence.

Algorithm:

1. start.
2. get the number upto which Fibonacci series is generated.
3. assign $f1=0, f2=1$ and print it.
4. Add $f1$ & $f2$ and assign it to $f3$. print it.
5. Assign $f2$ to $f1$ and $f3$ to $f2$.
6. goto step 4 until the number is not zero.
7. stop.

Flowchart:



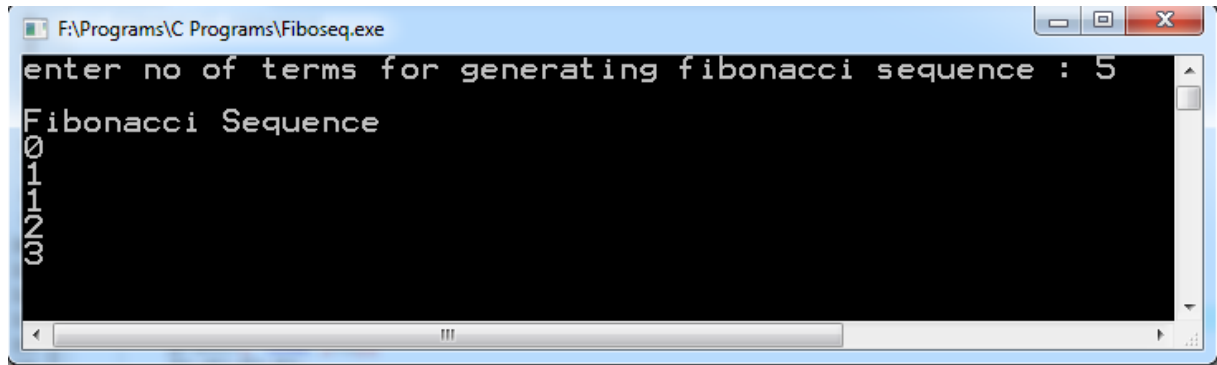
Program:

```

#include <stdio.h>
void main()
{
    int f1=0, f2=1, n, i, f3;
    printf("enter no of terms for generating fibonacci sequence : ");
    scanf("%d", &n);
    printf("\nFibonacci Sequence ");
    printf("\n%d\n%d", f1, f2);
    for(i=3; i<=n; i++)

```

```
{  
    f3=f1+f2;  
    printf("\n%d",f3);  
    f1=f2;f2=f3;  
}  
getch();  
}
```

Output:

```
F:\Programs\C Programs\Fiboseq.exe  
enter no of terms for generating fibonacci sequence : 5  
Fibonacci Sequence  
0  
1  
1  
2  
3
```

Viva Questions:

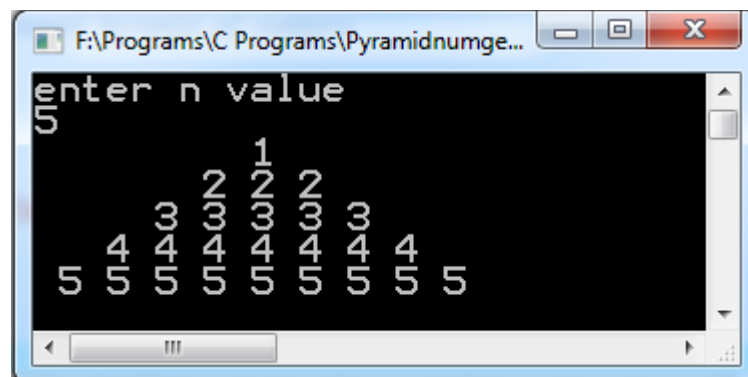
1. What is prime number?
2. What is Fibonacci series?
3. What are the various types of unconditional statements?
4. What are the various types of conditional statements?
5. Expand <stdio.h>?

Week – 3:

(a) **Aim : Write a C Program to Construct Pyramid of numbers**

Program:

```
#include<stdio.h>
void main()
{
    int i,j,n;
    printf("enter n value\n");
    scanf("%d",&n);
    for(i=1;i<=n;i++)
    {
        for(j=n-1;j>=i;j--)
            printf(" ");
        for(j=1;j<=i;j++)
            printf("%2d",i);
        for(j=i-1;j>=1;j--)
            printf("%2d",i);
        printf("\n");
    }
    getch();
}
```

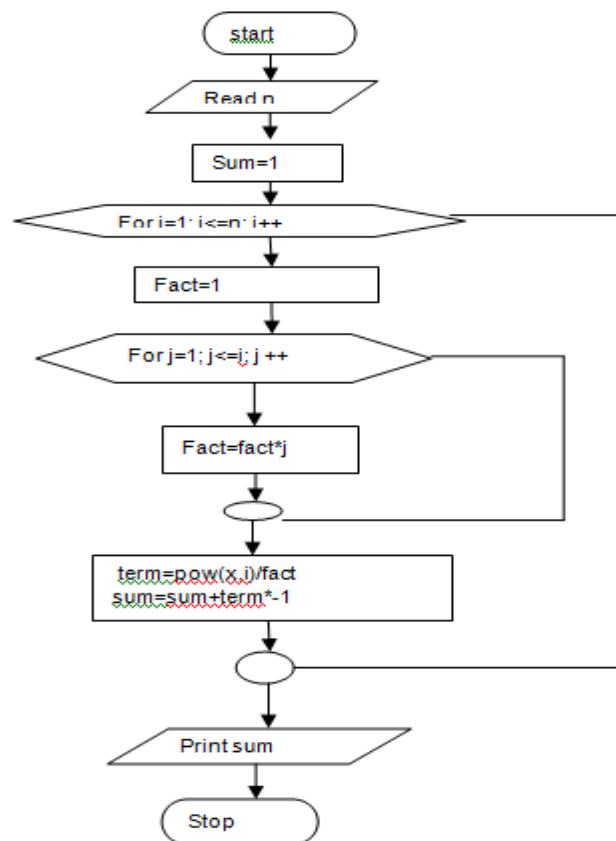
Output:

(b) **Aim: Write a C Program to calculate the following Sum: $\text{Sum} = 1 - X^2 / 2! + X^4 / 4! - X^6 / 6! + X^8 / 8! - X^{10} / 10!$**

Algorithm

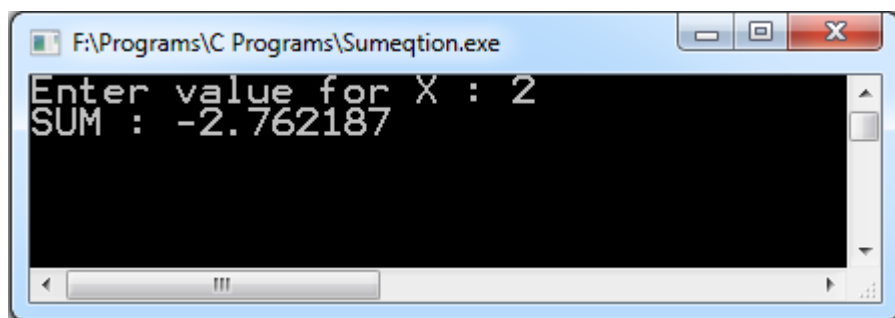
- Step-1: start
- Step-2: Read x
- Step-3: initialize values for i=2, sum=1
- Step-4: check whether the value of I is less than or equal to the value of n
- Step-5: Repeat for i=1 to n
- Step-6: initialize values for j=1, fact=1
- Step-7: Repeat for j=1 to goto step-10
- Step-8: compute the value of fact=fact*j
- Step-9: incremented j by 1
- Step-10: compute s=s+POW(x,i)/fact
- Step-11: increment i by 1
- Step-12: print the value of sum
- Step-13: stop

Flowchart:



Program:

```
#include <stdio.h>
#include <math.h>
void main()
{
    int i,j;
    float sum=0,x,term,fact,sign=-1;
    printf("Enter value for X : ");
    scanf("%f",&x);
    for(i=2;i<=10; i=i+2)
    {
        fact=1;
        for(j=1; j<=i; j++)
            fact = fact*j;
        term=pow(x,i)/fact*sign;
        sum=sum+term;
        sign=-1;
    }
    printf("SUM : %f",sum);
    getch();
}
```

Output:**Viva Questions:**

1. What are various types of loop statements?
2. What is the difference between while and do-while statements?
3. List out the C features?
4. What are the basic data types in C?
5. What is an Expression?

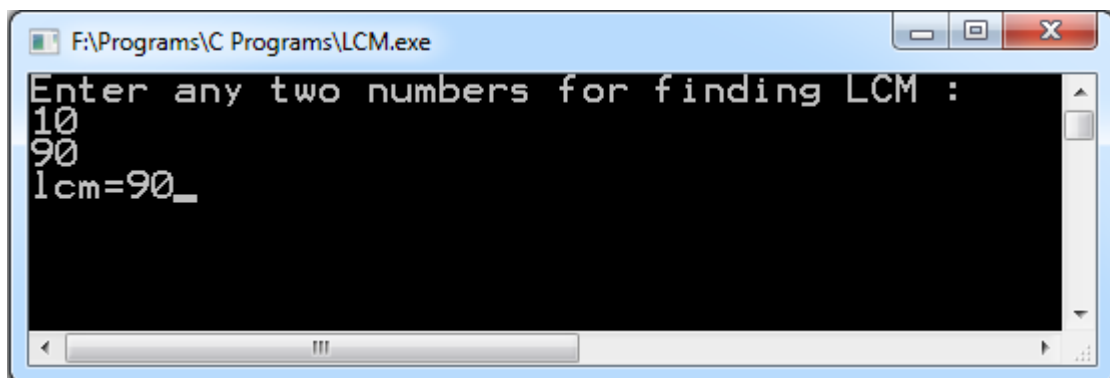
Week – 4:

(a) **Aim :** The least common multiple (lcm) of two positive integers a and b is the smallest integer that is evenly divisible by both a and b. Write a C Program that reads two integers and calls lcm (a, b) function that takes two integer arguments and returns their lcm. The lcm (a, b) function should calculate the least common multiple by calling the gcd (a, b) function and using the following relation: $LCM(a,b) = ab / gcd(a,b)$.

Program:

```
#include<stdio.h>
void main()
{
    int a,b,lcm;
    int gcd(int,int);
    printf("Enter any two numbers for finding LCM :\n");
    scanf("%d%d",&a,&b);
    lcm=(a*b)/gcd(a,b);
    printf("lcm=%d",lcm);
    getch();
}

int gcd(int a,int b)
{
    int r;
    while(a%b!=0)
    {
        r=a%b;
        a=b;
        b=r;
    }
    return(b);
}
```

Output:

(b) **Aim :** Write a C Program that reads two integers n and r to compute the nC_r value using the following relation: ${}^nC_r (n,r) = n! / r! (n-r)! .$ Use a function for computing the factorial value of an integer.

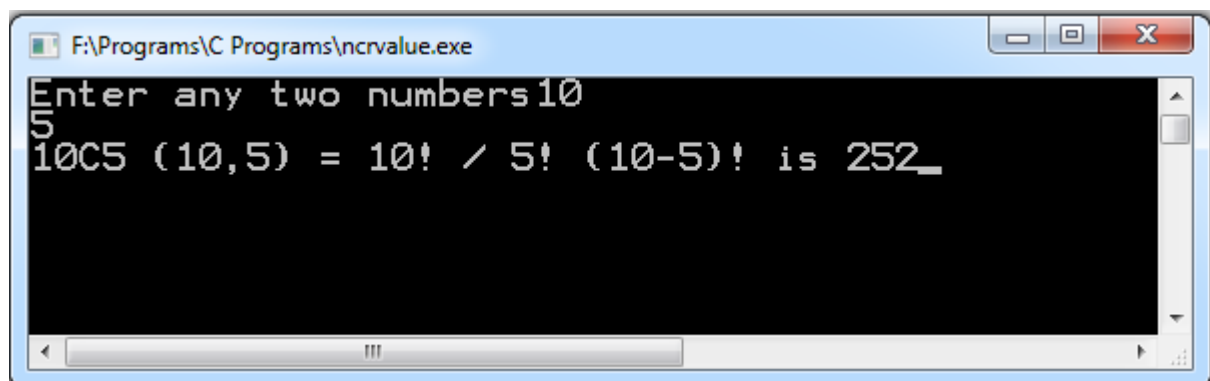
Program:

```
#include<stdio.h>

void main()
{
    int n,r,ncr;
    int fact(int);
    printf("Enter any two numbers");
    scanf("%d%d",&n,&r);
    ncr=fact(n)/(fact(r)*fact(n-r));
    printf("%dC%d (%d,%d) = %d! / %d! (%d-%d)! is %d",n,r,n,r,n,r,n,ncr);
    getch();
}

int fact(int n)
{
    int i,f=1;
    for(i=1;i<=n;i++)
        f=f*i;
    return(f);
}
```

Output:



Viva Questions:

1. What is meaning of GCD and how to find out GCD?
2. What is meaning of LCM and how to find out LCM?
3. Define scope of variable?
4. Define extent of a variable?
5. What is an identifier?

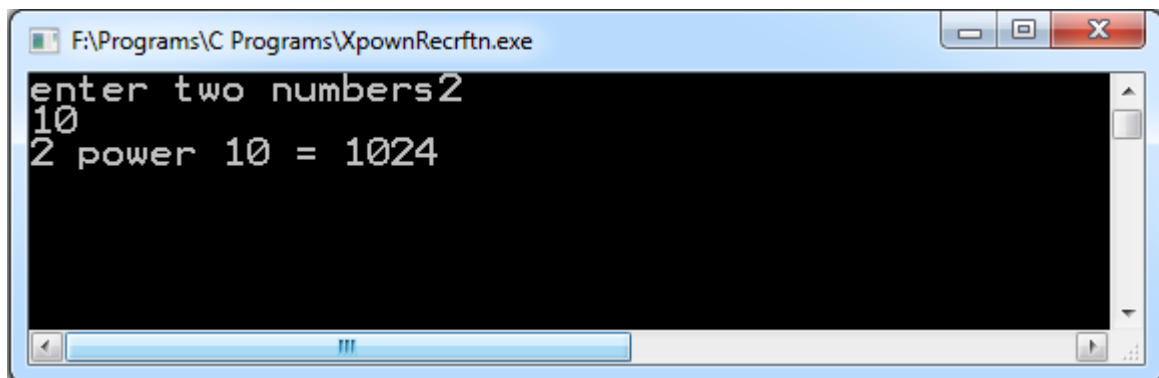
Week – 5:

(a) **Aim: Write C Program that reads two integers x and n and calls a recursive function to compute X^n**

Program:

```
#include<stdio.h>
void main()
{
    int x,n;
    int pow(int,int);
    printf("enter two numbers");
    scanf("%d%d",&x,&n);
    printf("%d power %d = %d",x,n,pow(x,n));
    getch();
}

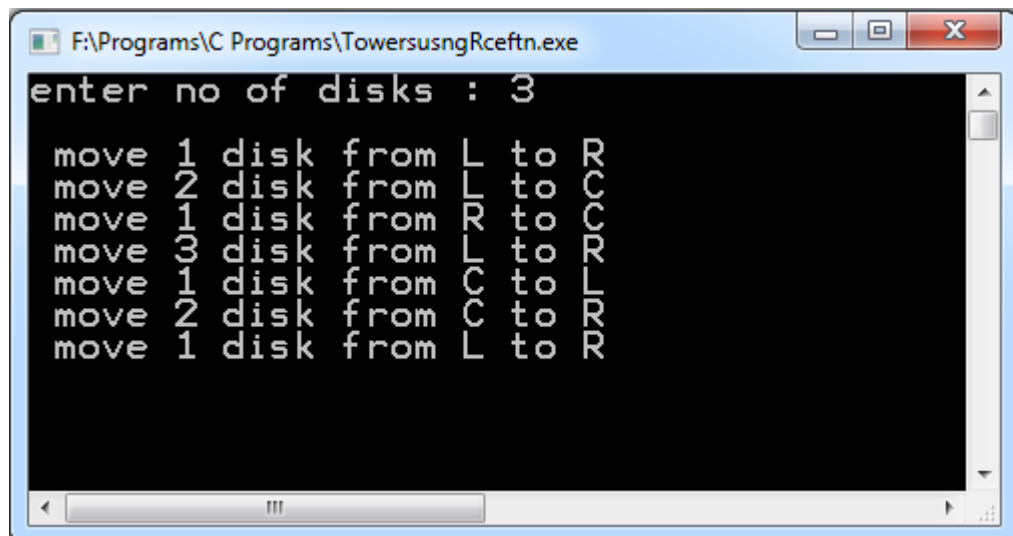
int pow(int m, int n)
{
    if(n==1)
        return m;
    else
        return(m*pow(m,n-1));
}
```

Output:

(b) Aim : Write a Program to Towers of Hanoi using recursive functions**Program:**

```
#include<stdio.h>
void main()
{
    int n;
    void towers(int,char,char,char);
    printf("enter no of disks : ");
    scanf("%d",&n);
    towers(n,'L','R','C');
    getch();
}

void towers(int n,char L ,char R,char C)
{
    if(n==1)
        printf("\n move %d disk from %c to %c",n,L,R);
    else
    {
        towers(n-1,L,C,R);
        printf("\n move %d disk from %c to %c",n,L,R);
        towers(n-1,C,R,L);
    }
}
```

Output:

```
F:\Programs\C Programs\TowersusngRceftn.exe
enter no of disks : 3

move 1 disk from L to R
move 2 disk from L to C
move 1 disk from R to C
move 3 disk from L to R
move 1 disk from C to L
move 2 disk from C to R
move 1 disk from L to R
```

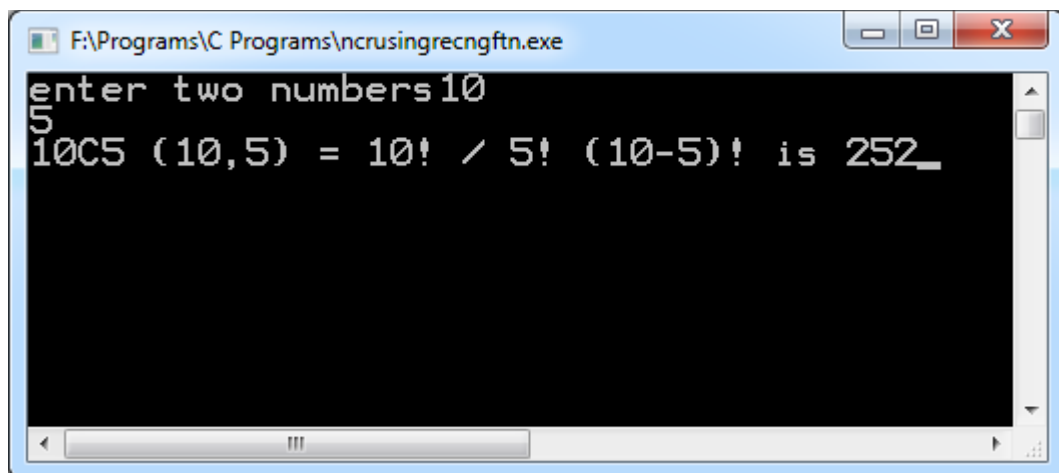
(c) **Aim :** Write a C Program that reads two integers and calls a recursive function to compute nC_r value.

Program:

```
#include<stdio.h>
void main()
{
    int n,r;
    int ncr(int,int);
    printf("enter two numbers");
    scanf("%d%d",&n,&r);
    printf("%dC%d (%d,%d) = %d! / %d! (%d-%d)! is %d",n,r,n,r,n,r,n,ncr(n,r));
    getch();
}

int ncr(int n,int r)
{
    if((r==0)||(r==n))
        return 1;
    else
        return(ncr(n-1,r-1)+ncr(n-1,r));
}
```

Output:



Viva Questions:

1. What is Function?
2. What is Recursive Function?
3. What is the difference keyword and identifier?
4. What is actual parameter?
5. What is an argument?

Week – 6:

(a) **Aim :** Write a C Program to generate all the prime numbers between 1 and n, where n is a value supplied by the user using Sieve of Eratosthenes Algorithm.

Program:

```
#include<stdio.h>
void main()
{
    int i,j,numbers[200],n;
    int primes[200];
    printf("enter n : ");
    scanf("%d",&n);
    for (i=0;i<n;i++)
    {
        numbers[i]=i+2;
    }
    /*sieve the non-primes*/
    for (i=0;i<n;i++)
    {
        if (numbers[i]!=-1)
        {
            for (j=2*numbers[i]-2;j<n;j+=numbers[i])
                numbers[j]=-1;
        }
    }
    /*transfer the primes to their own array*/
    j = 0;
    for (i=0;i<n;i++)
    {
        if (numbers[i]!=-1)
            primes[j++] = numbers[i];
    }
    printf("Prime numbers 1 to %d",n);
    for (i=0;i<j;i++)
        printf("\n%d ",primes[i]);
    getch();
}
```

Output:

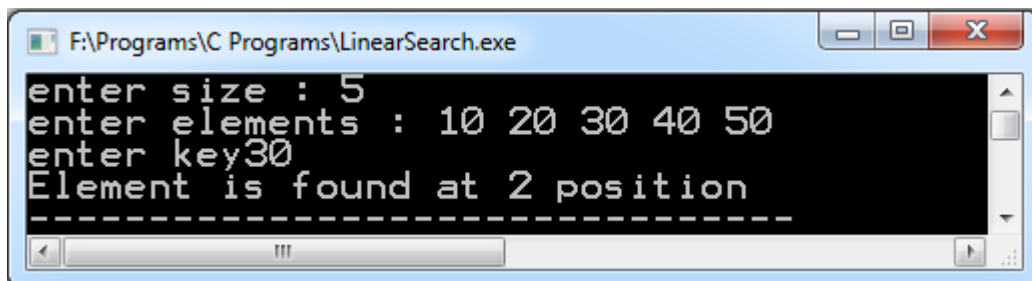
```
F:\Programs\C Programs\Primeusngalgoritm.exe
enter n : 15
Prime numbers 1 to 15
2
3
5
7
11
13
```

(b) Aim : Write a C Program that uses non recursive function to search for a Key value in a given list of integers. Use linear search method.

Program:

```
#include<stdio.h>
void main()
{
    int a[100],n,key,i,pos;
    int linear(int[],int,int,int);
    printf("enter size : ");
    scanf("%d",&n);
    printf("enter elements : ");
    for(i=0;i<=n-1;i++)
        scanf("%d",&a[i]);
    printf("enter key");
    scanf("%d",&key);
    pos=linear(a,0,n-1,key);
    if(pos==-1)
        printf("Element is not found");
    else
        printf("Element is found at %d position",pos);
}
int linear(int a[],int low,int high, int key)
{
    int i;
    for(i=low;i<=high;i++)
    {
        if(key==a[i])
            return i;
    }
    return -1;
}
```

Output:



Viva Questions:

1. What is Linear Search?
2. What is an array?
3. What is key?
4. What is Eratosthenes Algorithm?
5. What is the use of return keyword?

Week – 7:

(a) **AIM:** Write a menu-driven C Program that allows a user to enter n numbers and then choose between finding the smallest, largest, sum, or average. The menu and all the choices are to be functions. Use a switch statement to determine what action to take. Display an error message if an invalid choice is entered.

Program:

```
#include<stdio.h>
int smallest(int[],int);
int largest(int[],int);
int sum(int[],int);
float average(int[],int);
void main()
{
    int a[100],i,n,ch;
    printf("enter number of elemets");
    scanf("%d",&n);
    printf("enter elements");
    for(i=0;i<n;i++)
        scanf("%d",&a[i]);
    while(1)
    {
        printf("\n1:smallest\n2:largest\n3:sum\n4:average \n5:exit\n");
        printf("enter your choice");
        scanf("%d",&ch);
        switch(ch)
        {
            case 1: printf("smallest number=%d",smallest(a,n));
                    break;
            case 2: printf("largest number=%d",largest(a,n));
                    break;
            case 3: printf("sum of elements=%d",sum(a,n));
                    break;
            case 4: printf("average=%f",average(a,n));
                    break;
            case 5: exit(1);
            default:printf("invalid options");
        }
    }
}

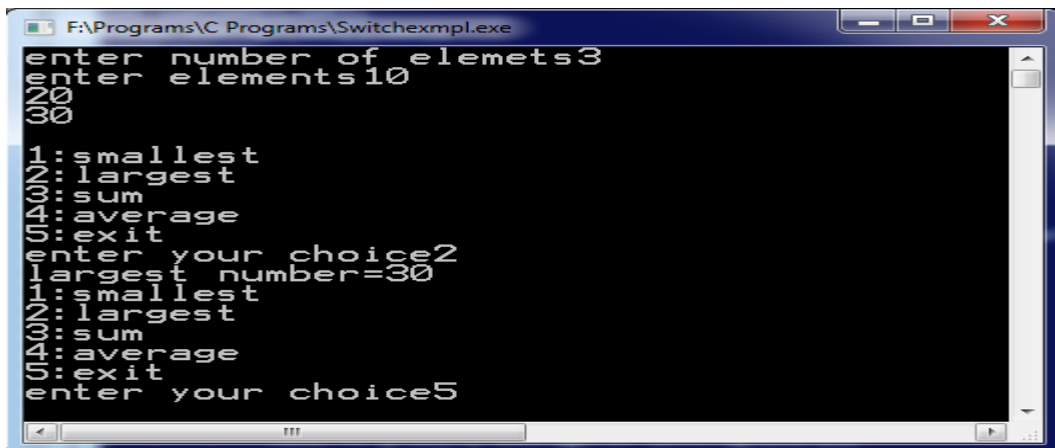
int smallest(int a[],int n)
{
    int i,min=a[0];
    for(i=1;i<n;i++)
    {
        if(a[i]<min)
```

```
        min=a[i];
    }
    return min;
}

int largest(int a[],int n)
{
    int i,max=a[0];
    for(i=1;i<n;i++)
    {
        if(a[i]>max)
            max=a[i];
    }
    return max;
}

int sum(int a[],int n)
{
    int i,sum=0;
    for(i=1;i<n;i++)
        sum=sum+a[i];
    return sum;
}

float average(int a[],int n)
{
    float avg;
    avg=sum(a,n)/n;
    return(avg);
}
```

Output:

```
F:\Programs\C Programs\Switchexmpl.exe
enter number of elemets3
enter elements10
20
30

1:smallest
2:largest
3:sum
4:average
5:exit
enter your choice2
largest number=30
1:smallest
2:largest
3:sum
4:average
5:exit
enter your choice5
```

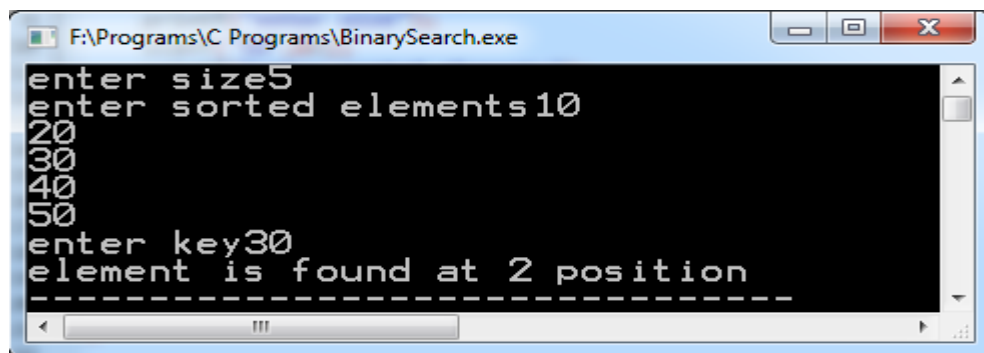
(b) Aim : Write a C Program that uses non recursive function to search for a Key value in a given sorted list of integers. Use binary search method.

Program:

```
#include<stdio.h>

int binary(int[],int,int,int);
void main()
{
    int a[100],i,n,key,pos;
    printf("enter size");
    scanf("%d",&n);
    printf("enter sorted elements");
    for(i=0;i<n;i++)
        scanf("%d",&a[i]);
    printf("enter key");
    scanf("%d",&key);
    pos=binary(a,0,n-1,key);
    if(pos==-1)
        printf("element is not found");
    else
        printf("element is found at %d position",pos);
}

int binary(int a[],int low,int high,int key)
{
    int mid;
    while(low<=high)
    {
        mid=(low+high)/2;
        if(key==a[mid])
            return mid;
        else if(key>a[mid])
            low=mid+1;
        else
            high=mid-1;
    }
    return -1;
}
```


Output:

```
F:\Programs\C Programs\BinarySearch.exe
enter size5
enter sorted elements10
20
30
40
50
enter key30
element is found at 2 position
-----
```

Viva Questions:

1. What is an array?
2. What is binary search?
3. What is the difference between Linear and Binary search?
4. What is switch statement?
5. What is the difference between switch and nested if statement?

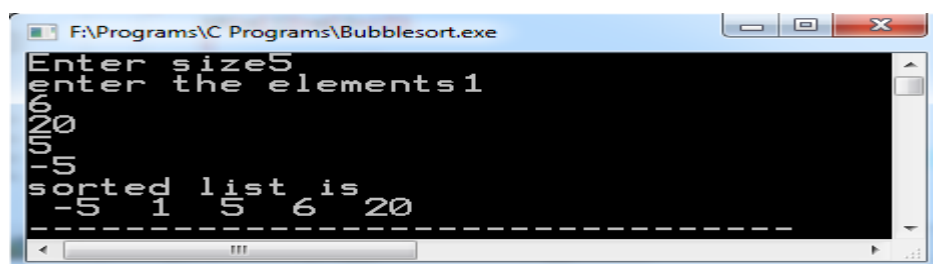
Week – 8:

(a) **Aim :** Write a C Program that implements the Bubble sort method to sort a given list of integers in ascending order.

Program:

```
#include<stdio.h>
void main()
{
    int a[100],n,i;
    void bubble(int[] ,int);
    printf("Enter size");
    scanf("%d",&n);
    printf("enter the elements");
    for(i=0;i<n;i++)
        scanf("%d",&a[i]);
    bubble(a,n);
    printf("sorted list is \n");
    for(i=0;i<n;i++)
        printf(" %d ",a[i]);
}

bubble(int a[], int n)
{
    temp,i,j;
    for(i=0;i<n-1;i++)
    {
        for(j=0;j<n-i-1;j++)
        {
            if(a[j]>a[j+1])
            {
                temp=a[j];
                a[j]=a[j+1];
                a[j+1]=temp;
            }
        }
    }
}
```

Output:

```
F:\Programs\C Programs\Bubblesort.exe
Enter size5
enter the elements1
20
-5
6
-5
sorted list is
-5 1 5 6 20
```

(b) Aim : Write a C Program that reads two matrices and uses functions to perform the following: i) Addition of two matrices ii) Multiplication of two matrices

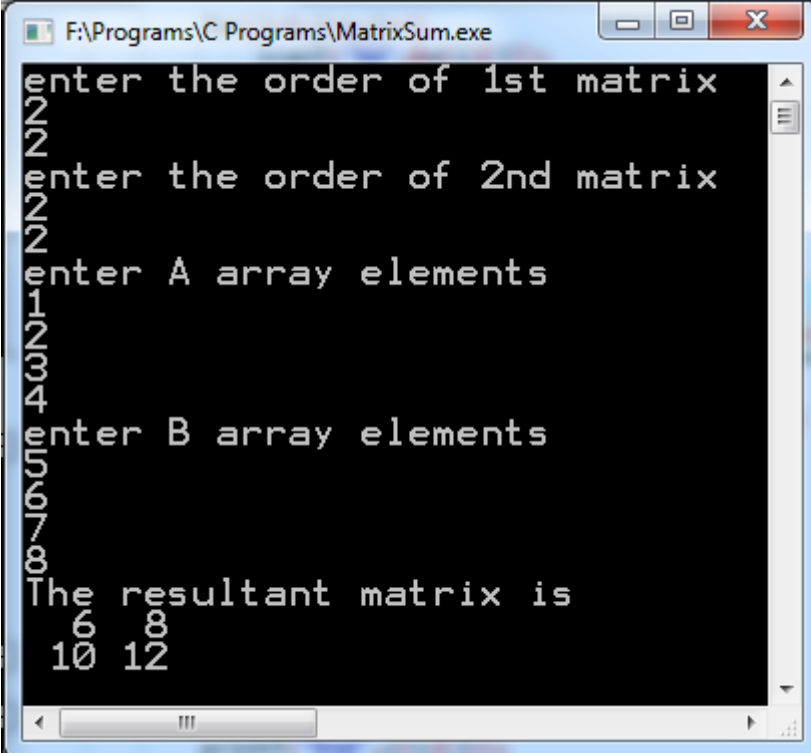
i) Addition of two matrices

Program:

```
#include<stdio.h>
#include<conio.h>
int i,j,p,q,m,n;
void main()
{
    int a[10][10],b[10][10];
    void matrixadd(int a[10][10],int b[10][10],int,int);
    printf("enter the order of 1st matrix\n");
    scanf("%d%d",&m,&n);
    printf("enter the order of 2nd matrix\n");
    scanf("%d%d",&p,&q);
    if((m==p) &&(n==q))
    {
        printf("enter A array elements\n");
        for(i=0;i<p;i++)
            for(j=0;j<q;j++)
                scanf("%d",&a[i][j]);
        printf("enter B array elements\n");
        for(i=0;i<p;i++)
            for(j=0;j<q;j++)
                scanf("%d",&b[i][j]);
        matrixadd(a,b,p,q);
    }
    else
        printf("matrix addition is not possible");
    getch();
}

void matrixadd(int a[10][10],int b[10][10],int p,int q)
{
    int c[10][10],i,j;
    for(i=0;i<p;i++)
        for(j=0;j<q;j++)
            c[i][j]=a[i][j]+b[i][j];
    printf("The resultant matrix is \n");
    for(i=0;i<p;i++)
    {
        for(j=0;j<q;j++)
        {
            printf("%3d",c[i][j]);
        }
    }
}
```

```
        printf("\n");  
    }  
}
```

Output:

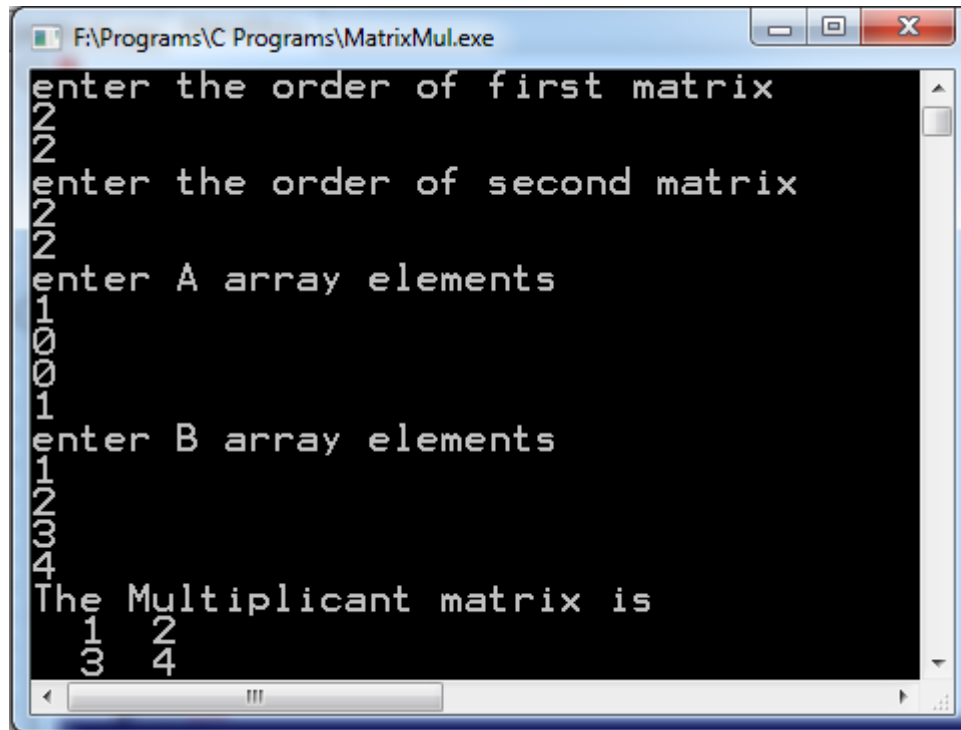
```
F:\Programs\C Programs\MatrixSum.exe  
enter the order of 1st matrix  
2  
2  
enter the order of 2nd matrix  
2  
2  
enter A array elements  
1  
2  
3  
4  
enter B array elements  
5  
6  
7  
8  
The resultant matrix is  
6 8  
10 12
```

ii) Multiplication of two matrices**Program:**

```
#include<stdio.h>
#include<conio.h>
int i,j,p,q,p1,q1;
void main()
{
    int a[10][10],b[10][10];
    void matrixmul(int a[10][10],int b[10][10],int,int,int);
    printf("enter the order of first matrix\n");
    scanf("%d%d",&p,&q);
    printf("enter the order of second matrix\n");
    scanf("%d%d",&p1,&q1);
    if(p1==q)
    {
        printf("enter A array elements\n");
        for(i=0;i<p;i++)
            for(j=0;j<q;j++)
                scanf("%d",&a[i][j]);
        printf("enter B array elements\n");
        for(i=0;i<p1;i++)
            for(j=0;j<q1;j++)
                scanf("%d",&b[i][j]);
        matrixmul(a,b,p,q1,q);
    }
    else
        printf("matrix multiplication is not possible");
    getch();
}

void matrixmul(int a[10][10],int b[10][10],int m ,int q,int n)
{
    int c[10][10],i,j,k;
    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("The Multiplicant matrix is \n");
    for(i=0;i<m;i++)
```

```
    {  
        for(j=0;j<q;j++)  
        {  
            printf("%3d",c[i][j]);  
        }  
        printf("\n");  
    }  
}
```

Output:**Viva Questions:**

1. How many types of arrays are there?
2. What is sort? How many types of sort techniques?
3. What is condition for performing a bubble sort?
4. What is condition for performing a matrix addition?
5. What is condition for performing a matrix multiplication?

Week – 9:

(a) Aim : Write a C Program that uses functions to perform the following operations: i) To insert a substring into a given main string from a given position. ii) To delete n characters from a given position in a given string.

i) To insert a sub-string into a given main string from a given position.

Program:

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

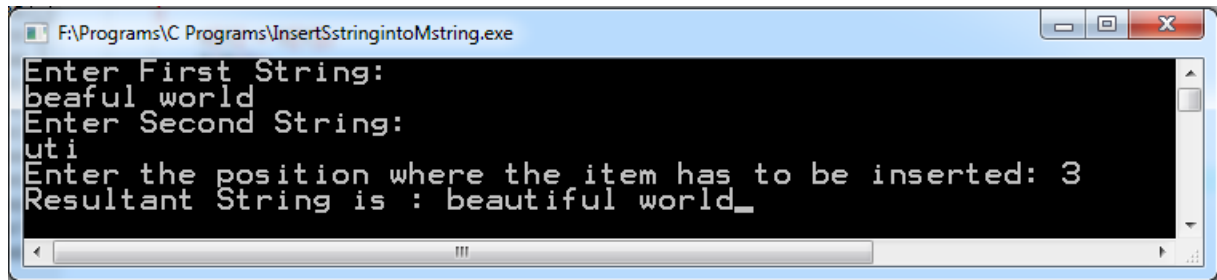
main()
{
    char a[10];
    char b[10];
    char c[10];
    int p=0,r=0,i=0;
    int t=0;
    int x,g,s,n,o;
    puts("Enter First String:");
    gets(a);
    puts("Enter Second String:");
    gets(b);
    printf("Enter the position where the item has to be inserted: ");
    scanf("%d",&p);
    r = strlen(a);
    n = strlen(b);
    i=0;
    while(i <= r)
    {
        c[i]=a[i];
        i++;
    }
    s = n+r;
    o = p+n;

    // 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("Resultant String is : %s", a);
    getch();
}

```

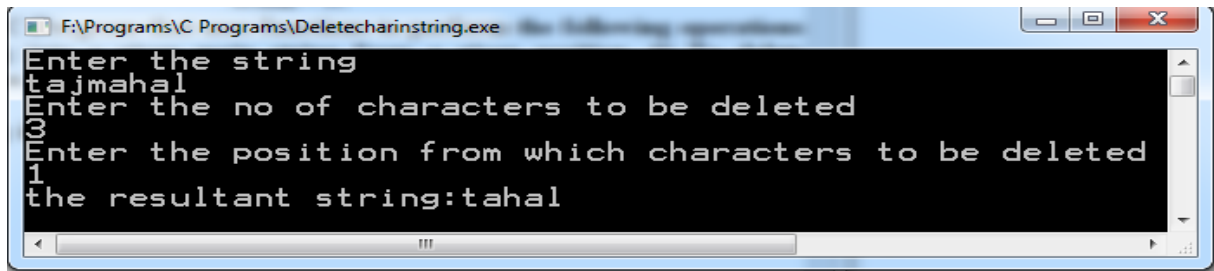
Output:**ii) To delete n characters from a given position in a given string.****Program:**

```

#include<stdio.h>
int p,n;
main()
{
    char s1[50];
    void delete(char s1[],int p,int n);
    printf("Enter the string\n");
    scanf("%s",s1);
    printf("Enter the no of characters to be deleted\n");
    scanf("%d",&n);
    printf("Enter the position from which characters to be deleted\n");
    scanf("%d",&p);
    delete(s1,p,n);
    getch();
}

void delete(char s1[20],int p,int n)
{
    int i=p,j=p+n;
    while(s1[j]!='\0')
    {
        s1[i++]=s1[j++];
        s1[i]='\0';
    }
    printf("the resultant string:%s\n",s1);
}

```

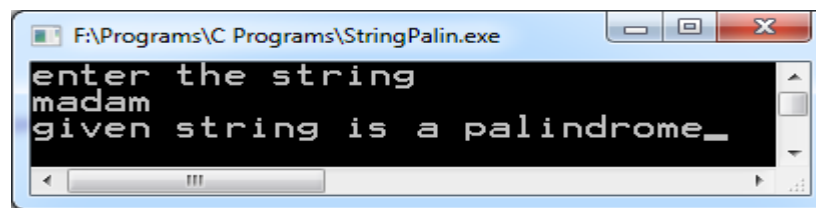

Output:

- (b) **Aim :** Write a C Program that uses a non recursive function to determine if the given string is a palindrome or not.

Program:

```
#include<stdio.h>
#include<string.h>
void main()
{
    char s[30];
    int n;
    printf("enter the string\n");
    scanf("%s",s);
    n=palin(s);
    if(n==1)
        printf("given string is a palindrome");
    else
        printf("not palindrome");
    getch();
}

int palin(char s[])
{
    int i,j,l,flag=1;
    l=strlen(s);
    i=0,j=l-1;
    while(i<=l/2 )
    {
        if(s[i]!=s[j])
        {
            flag=0;
            break;
        }
        i++;
        j--;
    }
    return(flag);
}
```

Output:

```
F:\Programs\C Programs\StringPalin.exe
enter the string
madam
given string is a palindrome_
```

Viva Questions:

1. What is String?
2. Which command is used for to comparing a string?
3. Which command is used for to find out a substring?
4. What is Palindrome?
5. What is condition for performing Palindrome?

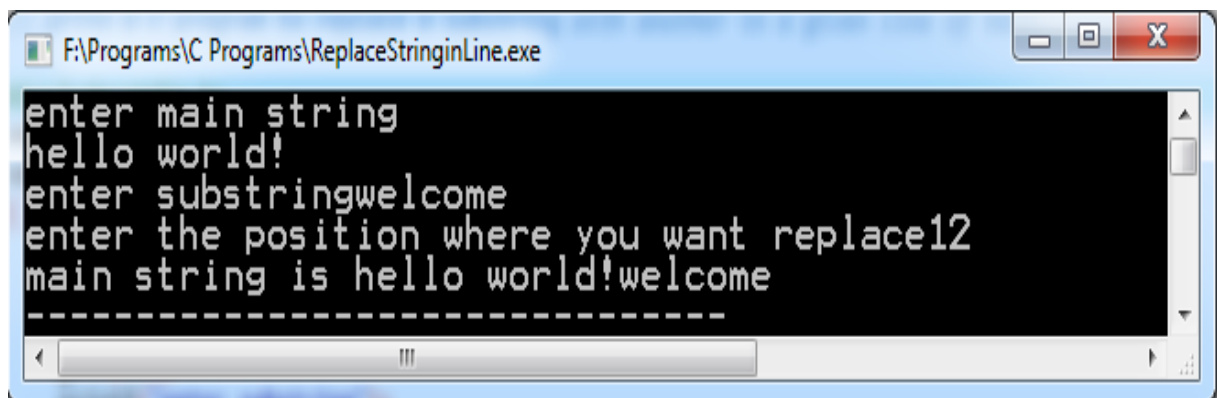
Week – 10:

- (a) **Aim :** Write a C Program to replace a substring with another in a given line of text.

Program:

```
#include<stdio.h>
void replace(char[100],char[100],int);
void main()
{
    char s1[100],s2[100];
    int d;
    printf("enter main string\n");
    gets(s1);
    printf("enter substring");
    gets(s2);
    printf("enter the position where you want replace");
    scanf("%d",&d);
    replace(s1,s2,d);
    getch();
}

void replace(char s1[100],char s2[100],int d)
{
    int l,i,j;
    l=strlen(s1);
    for(i=0,j=d;s2[i]!='\0';i++,j++)
        s1[j]=s2[i];
    if(l<j)
        s1[j]='\0';
    printf("main string is %s",s1);
}
```

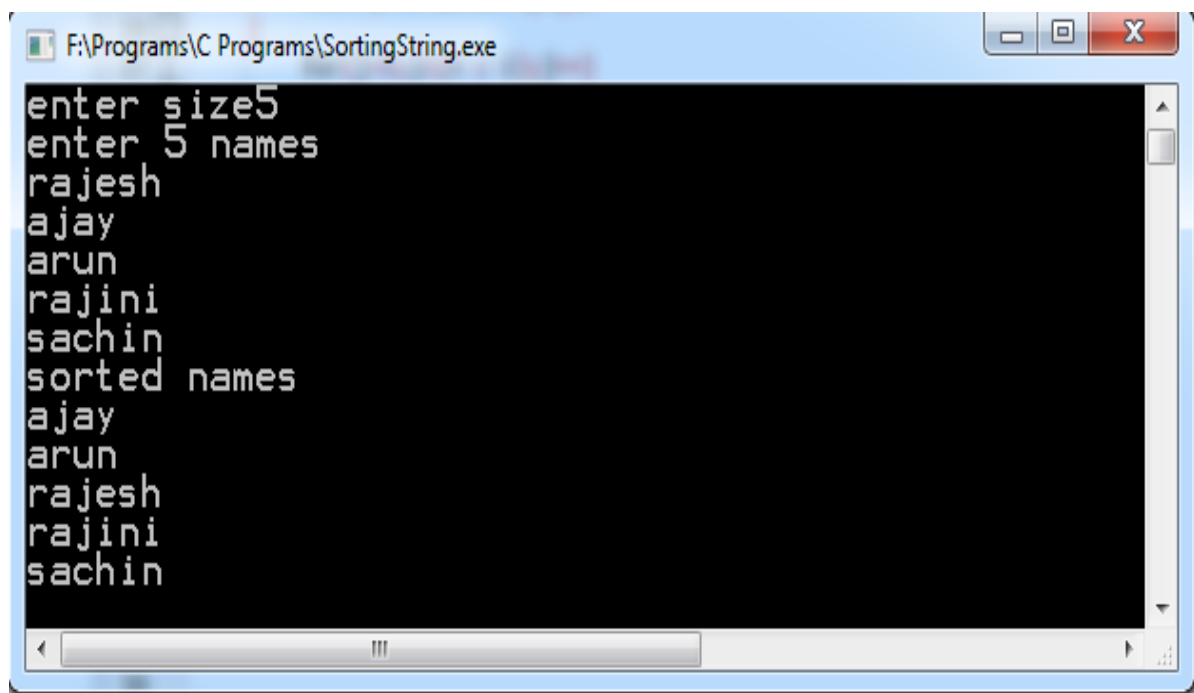
Output:

```
F:\Programs\C Programs\ReplaceStringInLine.exe
enter main string
hello world!
enter substringwelcome
enter the position where you want replace12
main string is hello world!welcome
-----
```

(b) Aim : Write a C Program that reads 15 names each of up to 30 characters, stores them in an array, and uses an array of pointers to display them in ascending (ie. alphabetical) order.

Program:

```
#include<stdio.h>
#include<string.h>
void main()
{
    int j,i,n;
    char names[15][30],*p[15],temp[30];
    printf("enter size");
    scanf("%d",&n);
    printf("enter %d names\n",n);
    for(i=0;i<n-1;i++)
    for(i=0;i<n;i++)
        scanf("%s",&names[i]);
    {
        for(j=0;j<(n-1-i);j++)
        {
            if(strcmp(names[j],names[j+1])>0)
            {
                strcpy(temp,names[j]);
                strcpy(names[j],names[j+1]);
                strcpy(names[j+1],temp);
            }
        }
    }
    for(i=0;i<n;i++)
        p[i]=names[i];
    printf("sorted names\n");
    for(i=0;i<n;i++)
        printf("%s\n",*(p+i));
}
```

Output:

```
F:\Programs\C Programs\SortingString.exe
enter size5
enter 5 names
rajesh
ajay
arun
rajini
sachin
sorted names
ajay
arun
rajesh
rajini
sachin
```

Viva Questions:

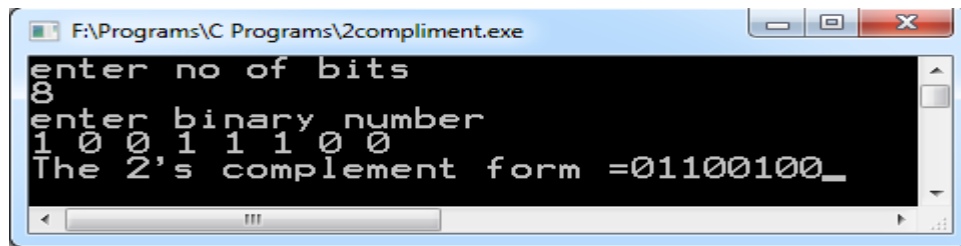
1. Which command is used to copy the strings?
2. Which command is used to combine the strings?
3. What is a pointer?
4. What are the various types of string functions?
5. What is use of the strcmp() function?

Week – 11:

(a) **Aim : 2's complement of a number is obtained by scanning it from right to left and complementing all the bits after the first appearance of a 1. Thus 2's complement of 11100 is 00100. Write a c Program to find the 2's complement of a binary number.**

Program:

```
#include<stdio.h>
void main()
{
    int a[10],i,n;
    printf("enter no of bits\n");
    scanf("%d",&n);
    printf("enter binary number\n");
    for(i=0;i<n;i++)
        scanf("%d",&a[i]);
    for(i=0;i<n;i++)
    {
        if(a[i]==0)
            a[i]=1;
        else
            a[i]=0;
    }
    for(i=n-1;i>=0;i--)
    {
        if(a[i]==0)
        {
            a[i]=1;
            break;
        }
        else
        {
            a[i]=0;
            if(a[i-1]==0)
            {
                a[i-1]=1;
                break;
            }
        }
    }
    printf("The 2's complement form =");
    for(i=0;i<n;i++)
        printf("%d",a[i]);
    getch();
}
```

Output:

(b) **Aim :** Write a C Program to convert a positive integer to a roman numeral. Ex. 11 is converted to XI.

Program:

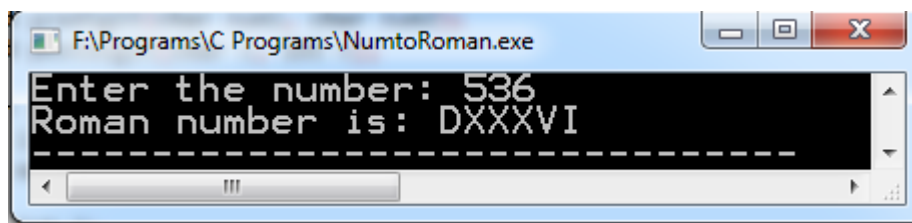
```
#include <stdio.h>
void predigit(char num1, char num2);
void postdigit(char c, int n);
char romanval[1000]; int i = 0;
int main()
{
    int j;
    long number;
    printf("Enter the number: ");
    scanf("%d", &number);
    if (number <= 0)
    {
        printf("Invalid number");
        return 0;
    }
    while (number != 0)
    {
        if (number >= 1000)
        {
            postdigit('M', number / 1000);
            number = number - (number / 1000) * 1000;
        }
        else if (number >= 500)
        {
            if (number < (500 + 4 * 100))
            {
                postdigit('D', number / 500);
                number = number - (number / 500) * 500;
            }
            else
            {
                predigit('C', 'M');
                number = number - (1000-100);
            }
        }
    }
}
```

```
else if (number >= 100)
{
    if (number < (100 + 3 * 100))
    {
        postdigit('C', number / 100);
        number = number - (number / 100) * 100;
    }
    else
    {
        predigit('L', 'D');
        number = number - (500 - 100);
    }
}
else if (number >= 50 )
{
    if (number < (50 + 4 * 10))
    {
        postdigit('L', number / 50);
        number = number - (number / 50) * 50;
    }
    else
    {
        predigit('X','C');
        number = number - (100-10);
    }
}
else if (number >= 10)
{
    if (number < (10 + 3 * 10))
    {
        postdigit('X', number / 10);
        number = number - (number / 10) * 10;
    }
    else
    {
        predigit('X','L');
        number = number - (50 - 10);
    }
}
else if (number >= 5)
{
    if (number < (5 + 4 * 1))
    {
        postdigit('V', number / 5);
        number = number - (number / 5) * 5;
    }
    else
    {

```



```
        predigit('I', 'X');
        number = number - (10 - 1);
    }
}
else if (number >= 1)
{
    if (number < 4)
    {
        postdigit('I', number / 1);
        number = number - (number / 1) * 1;
    }
    else
    {
        predigit('I', 'V');
        number = number - (5 - 1);
    }
}
}
printf("Roman number is: ");
for(j = 0; j < i; j++)
    printf("%c", romanval[j]);
return 0;
}
void predigit(char num1, char num2)
{
    romanval[i++] = num1;
    romanval[i++] = num2;
}
void postdigit(char c, int n)
{
    int j;
    for (j = 0; j < n; j++)
        romanval[i++] = c;
}
```

Output:**Viva Questions:**

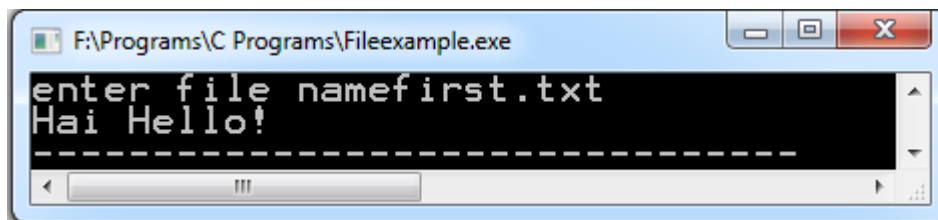
1. What is Binary number?
2. What is 1's and 2's compliment?
3. How to convert 1's compliment to 2's compliment?
4. How to convert binary to decimal?
5. How to convert octal to binary?

Week – 12:

- (a) **Aim : Write a C Program to display the contents of a file to standard output device**

Program:

```
#include<stdio.h>
void main()
{
    FILE *f1;
    char filename[100],ch;
    printf("enter file name");
    gets(filename);
    f1=fopen(filename,"r");
    while((ch=getc(f1))!=EOF)
        putchar(ch);
    fclose(f1);
}
```

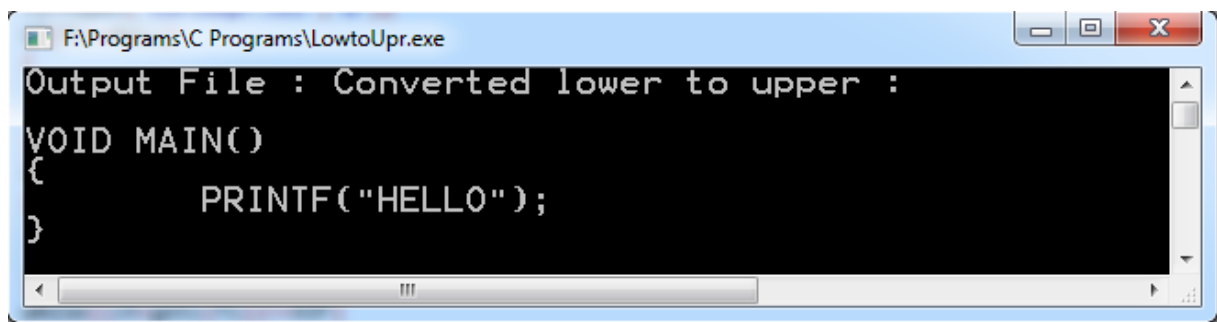
Output:

- (b) **Aim : Write a C Program which copies one file to another, replacing all lowercase Characters with their uppercase equivalents.**

Program:

```
#include<stdio.h>
#include<ctype.h>
void main()
{
    FILE *f1,*f2,*f3;
    char ch;
    f1=fopen("Hello.c","r");
    f2=fopen("LwrtoUpr.txt","w");
    if(f1==NULL)
    {
        printf("unable to open file");
        exit(1);
    }
}
```

```
if(f2==NULL)
{
    printf("unable to open file");
    exit(1);
}
while((ch=getc(f1))!=EOF)
{
    if(islower(ch))
        putc(ch-32,f2);
    else
        putc(ch,f2);
}
fclose(f1);
fclose(f2);
f3=fopen("LwrtoUpr.txt","r");
printf("Output File : Converted lower to upper :\n\n");
while((ch=getc(f3))!=EOF)
    putchar(ch);
fclose(f3);
getch();
}
```

Output:

```
F:\Programs\C Programs\LowtoUpr.exe
Output File : Converted lower to upper :
VOID MAIN()
{
    PRINTF("HELLO");
}
```

Viva Questions:

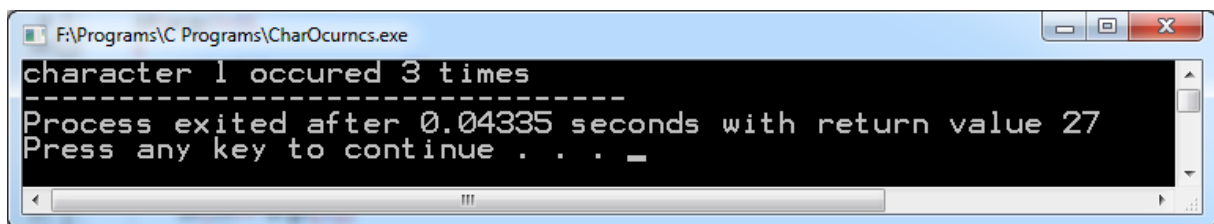
1. What is file?
2. What are file modes?
3. What are the various operations performed on the file?
4. What is the use of file pointer?
5. Which command is used for to convert lower to upper?

Week – 13:

(a) **Aim :** Write a C Program to count the number of times a character occurs in a text file. The file name and the character are supplied as command-line arguments

Program:

```
#include<stdio.h>
void main(int argc,char *argv[])
{
    FILE *f1;
    char ch;
    int count=0;
    if(argc!=3)
    {
        printf("error occurred");
        exit(1);
    }
    f1=fopen(argv[1],"r");
    while((ch=getc(f1))!=EOF)
    {
        if(ch==*argv[2])
            count++;
    }
    fclose(f1);
    printf("character %c occurred %d times",*argv[2],count);
}
```

Output:

b) **Aim :** Write a C program to compare two files, printing the first line where they differ.

Program:

```
#include <stdio.h>
void compare_two_binary_files(FILE *,FILE *);

int main(int argc, char *argv[])
{
    FILE *fp1, *fp2;

    if (argc < 3)
    {
        printf("\nInsufficient Arguments: \n");
    }
}
```

```
        printf("\nHelp:./executable <filename1> <filename2>\n");
        return;
    }
    else
    {
        fp1 = fopen(argv[1], "r");
        if (fp1 == NULL)
        {
            printf("\nError in opening file %s", argv[1]);
            return;
        }

        fp2 = fopen(argv[2], "r");

        if (fp2 == NULL)
        {
            printf("\nError in opening file %s", argv[2]);
            return;
        }

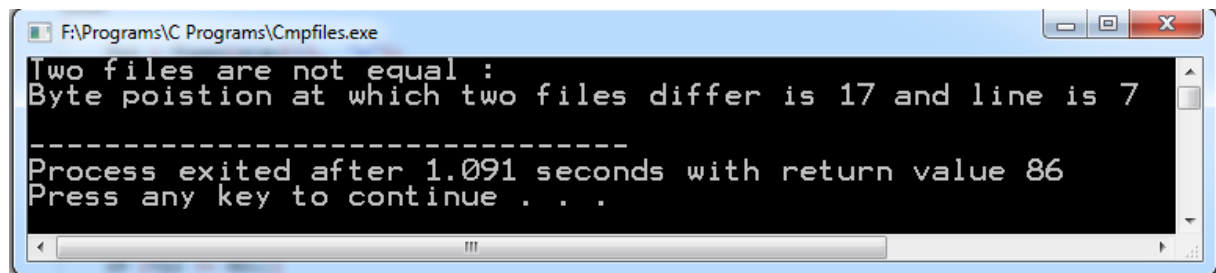
        if ((fp1 != NULL) && (fp2 != NULL))
        {
            compare_two_binary_files(fp1, fp2);
        }
    }
}

void compare_two_binary_files(FILE *fp1, FILE *fp2)
{
    char ch1, ch2;
    int flag = 0;

    while (((ch1 = fgetc(fp1)) != EOF) && ((ch2 = fgetc(fp2)) != EOF))
    {
        if (ch1 == ch2)
        {
            flag = 1;
            continue;
        }
        else
        {
            fseek(fp1, -1, SEEK_CUR);
            flag = 0;
            break;
        }
    }

    if (flag == 0)
```

```
{  
    printf("Two files are not equal :\n");  
    printf("Byte poistion at which two files differ is %d and line is %c\n", ftell(fp1)+1,  
fgetc(fp1));  
}  
else  
{  
    printf("Two files are Equal\n ", ftell(fp1)+1);  
}  
}
```

Output:**Viva Questions:**

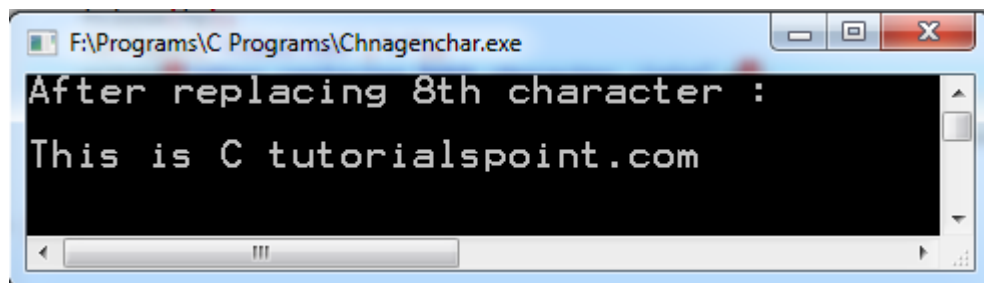
1. What is command line argument?
2. What is the use of fseek pointer?
3. What is the use of ftell pointer?
4. What is the difference between getc and fgetc?
5. What is the difference between fgetc and fgets?

Week – 14:

(a) **Aim :** Write a C Program to change the nth character (byte) in a text file. Use fseek function.

Program:

```
#include <stdio.h>
void main (int argc,char *argv[])
{
    FILE *fp,*f1;
    char ch;
    int m;
    if(argc!=3)
    {
        printf("error");
        exit(1);
    }
    fp = fopen(argv[1],"w+");
    fputs("This is H tutorialspoint.com", fp);
    m=atoi(argv[2]);
    fseek(fp,m,0);
    fputc("C", fp);
    fclose(fp);
    f1=fopen("file1.txt","r");
    printf("After replacing %dth character :\n\n",m);
    while((ch=getc(f1))!=EOF)
        putchar(ch);
    fclose(f1);
    getch();
}
```


Output:

(b) **Aim :** Write a C Program to reverse the first n characters in a file. The file name and n are specified on the command line. Use fseek function.

Program:

```
#include <stdio.h>
void main (int argc, char *argv[])
{
    FILE *fp;
    int m,i;
    char ch;
    if(argc!=3)
    {
        printf("error");
        exit(1);
    }
    fp = fopen(argv[1], "w");
    fputs("This is ctutorialspoint.com", fp);
    fclose(fp);
    m=atoi(argv[2]);
    fp = fopen(argv[1], "r");
    fseek(fp, m, 0);
    printf("first %d character in reverse\n", m);
    for(i=m; i>=0; i--)
    {
        ch=getc(fp);
        putchar(ch);
        fseek(fp, -2, 1);
    }
    fclose(fp);
}
```

Output:



Viva Questions:

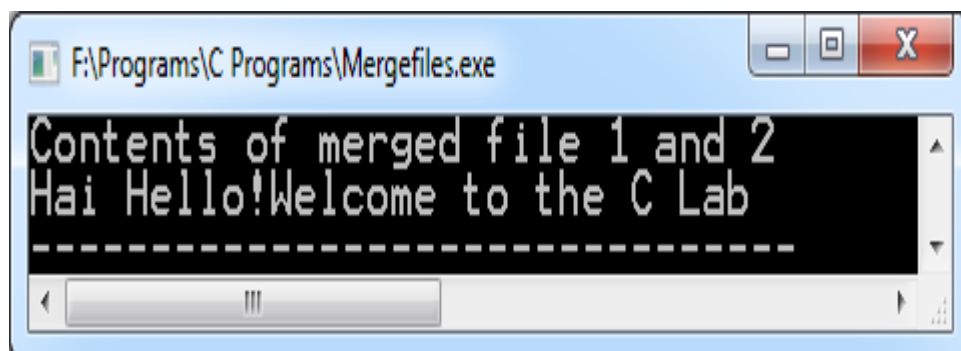
1. What is the use of fclose() function?
2. What is the use of fopen() function?
3. What is the use of FILE?
4. What is the use of r+?
5. What is the use of delete() function?

Week – 15:

(a) **Aim :** Write a C Program to merge two files into a third file (i.e., the contents of the first file followed by those of the second are put in the third file).

Program:

```
#include<stdio.h>
void main()
{
    FILE *f1,*f2,*f3;
    char ch;
    f1=fopen("first.txt","r");
    f2=fopen("second.txt","r");
    f3=fopen("third.txt","w");
    while((ch=getc(f1))!=EOF)
        putc(ch,f3);
    fclose(f1);
    while((ch=getc(f2))!=EOF)
        putc(ch,f3);
    fclose(f2);
    fclose(f3);
    printf("Contents of merged file 1 and 2 \n");
    f3=fopen("third.txt","r");
    while((ch=getc(f3))!=EOF)
        putchar(ch);
    fclose(f3);
}
```

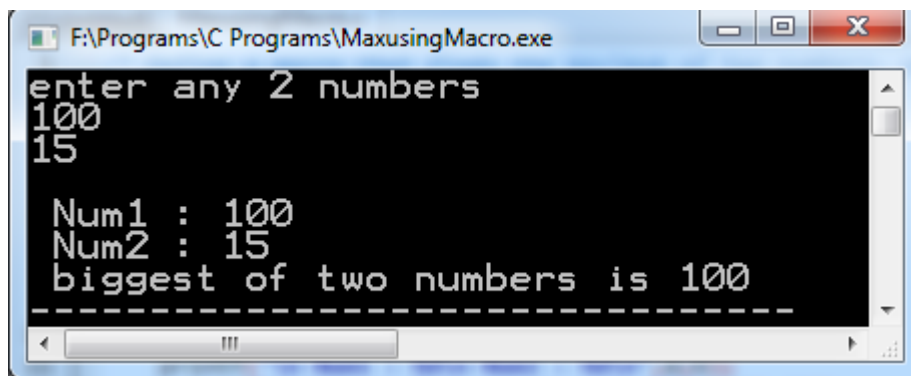
Output:

(b) Aim : Define a macro that finds the maximum of two numbers. Write a C Program that uses the macro and prints the maximum of two numbers.

Program:

```
#define MAX(a, b) ((a) < (b) ? (b) : (a))
#include<stdio.h>
void main()
{
    int a,b;
    printf("enter any 2 numbers\n");
    scanf("%d%d",&a,&b);
    printf("\n Num1 : %d\n Num2 : %d\n",a,b);
    printf(" biggest of two numbers is %d",MAX(a,b));
    getch();
}
```

Output:



Viva Questions:

1. What is merging?
2. What is macro?
3. What is procedure?
4. What is ternary operator?
5. Which header file is used for solving mathematical equations?