Unit 1: Introduction to Programming

Programming Languages

- Programming Languages are the languages design to express computation that can be performed by a machine (computer).
- Programming Languages are used to create programs that control the behavior of machine to express algorithm or mode of human communications with computer.

History of C

is structural and/or procedural language.
is problem solving language.
is artificial language used to communicate with computer by Implementing code/programs.

Language	Inventor / Developer	Purpose
1) BCPL	1967 Martin Richards	to develop UNIX
2) B	1970 Ken Thompson	operating system
3) C	1972 Dennis Ritchie	
4) Unix**(Operating System)**	1973 Ken Thompson & Dennis Ritchie	to work with ease on computer

★ **C** initially became widely known as the Development Language of ***UNIX*** Operating System.

- History and development of C is illustrated in following figure.

1960	ALGOL	International Group
	↓	
1967	BCPL	Martin Richards
	↓	
1970	B	Ken Thompson
	↓	
1972	Traditional C	Dennis Ritchie
	↓	
1978	K & R C	Kernighan and Ritchie
	↓	
1989	ANSI C	ANSI Committee
	↓	
1990	ANSI/ ISO C	ISO Committee
	↓	
1999	C99	Standardization Committee

Fig: - History of ANSI C

Features of C [Importance of C]

1. Simple & portable, can run on any hardware platform.
2. Faster execution, reliable, effective, flexible and small(32 key words)
3. Rich set of operators and library functions.
4. Supports systems programming & Ability to extend itself.
5. It is programmer's language, as we use C for System Programming.

Applications of C

1. To design computer games.
2. In business application
3. In embedded system.
4. In networking.
5. In database (File).
6. To develop system software.

Uses of C

Following System Software's (Programs) are written in C:

1. Operating Systems – Examples => **Unix & Linux**
2. Assembler, Compiler and interpreters.
    a. *Java Language Compiler is written in C/C++.*
    b. *Oracle and its interpreter is written in C.*
3. Network drivers.
4. Print spoolers.
5. Modem programs.

Source Code or Source Program

- **Program: -** Program is a set of instructions or statements that are grouped together to accomplish as task or tasks.
- Source Code is nothing but ***program text*** that we write.
- A source code of C language is based on building blocks. The building blocks are called functions.
- A C program (Source Code) is a collection of one or more functions.

Source Code

```
#include<stdio.h>
void main( )
{
printf("We always miss you\,Dennis Ritchie");
}
```

Output: -We always miss you, Dennis Ritchie

In above program both main( ) and printf( ) are functions

- main( ) is the first functions executed when our program begins to run(execute).
- main( ) is reserved function of operating system.
- printf( ) function causes its arguments to be printed on screen of the computer.
- In above program argument is mentioned in double inverted comma "We always miss you,Dennis Ritchie" is printed on screen as output.

Program Development Lifecycle (Steps to solve problem)

- Programs are developed to solve specific problem.
- If we have to solve any organizational need, or any mathematical problem, we can solve that by writing a program using programming language.
- Steps involved in program developments are as follows

Problem Definition:

- Become aware of the problem
- That problem should be clear and unambiguous
- Decide what inputs are available and what outputs are required.

Program Design:

- The problem should be analyzed to find the most efficient way in which it can be solved.
- An algorithm to solve the problem is developed based on the analysis.
- The algorithm may be expressed as a flowchart.
- The design specifies the type of data structures used and defines the relationship among the variable.
- The algorithm should be analyzed to check if it perform the desired actions using a reasonable amount of time and memory

Program Coding :

- Coding the algorithm using an appropriate language to form a program we can select any programming language which is easy to understand, easy for coding and also portable like 'C' language

Program Testing and Debugging

- The programmer locates and corrects as many errors as possible by eyeing the program before compiling it.
- If program contains errors or bugs then it will be debugged and errors are removed until program executes successfully.
- After successful execution of program, it should be tested thoroughly using all possible type of input.
- Testing detects inconsistencies in program (if any)

Documentation:

- Documentation is essential to make the program easy to understand
- It informs the readers about the working of the program, the mode of interaction and the purpose of variable used.
- A person who reads the program should be able to understand and modify the program using documentation.

Maintenance:

- The program may require changes depending on its applications
- The changes and updates required are performed during maintenance

All the 7 steps mentioned above are called as "software development life cycle.''

Program Compilation and Execution (Object (.obj) code and Executable(.exe)):

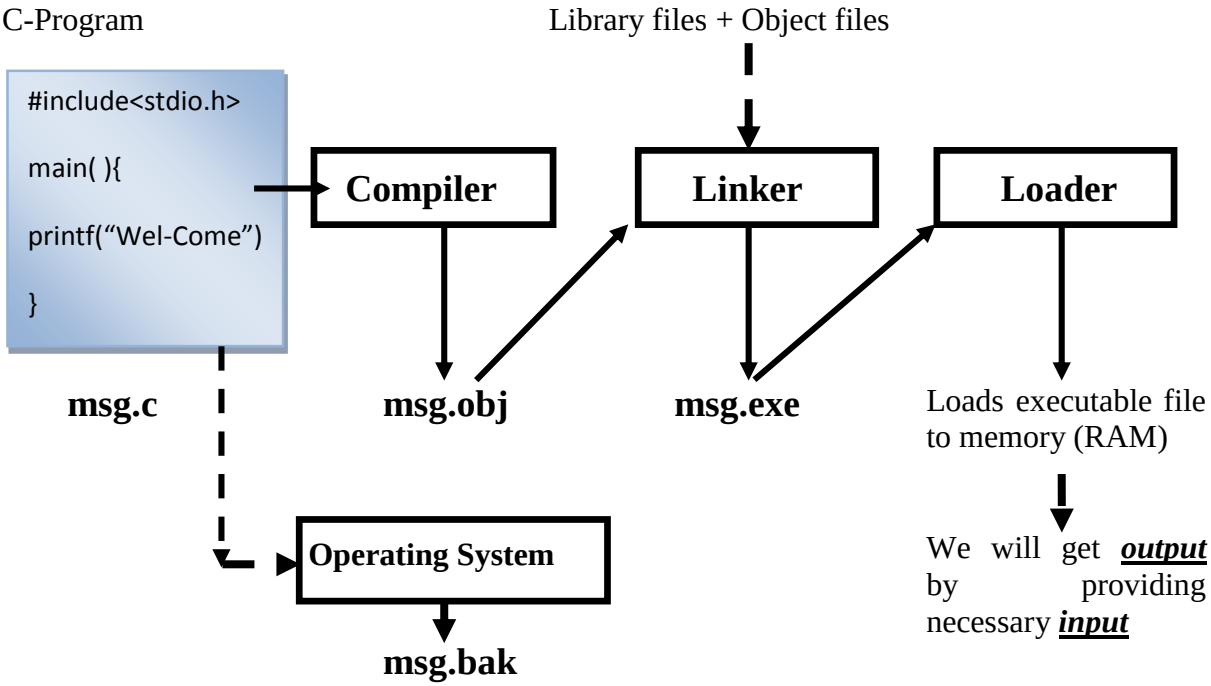


Fig:- C Program Execution Life Cycle

- A program written in higher level language must be translated into machine level language before it can be executed.
- Compiler is used to compile program
- In above diagram, C program by name 'msg.c' compiled with the help of compiler & compiler produces 'msg.obj' file as a output. Object file 'msg.obj' and library files will be given as input to linker which generates executable file 'msg.exe'.
- With the help of loader 'msg.exe' will be loaded to memory for execution & we will get output by giving input(if any).
- On other hand operating system produces 'msg.bak' file which contains C program code.

Algorithms

- Algorithm is a sequence of steps used to complete the task . or
- Algorithm is just a detailed sequence of simple steps that are needed to solve a problem
- The word Algorithm is derived from the phonetic pronunciation of the last name of Abu Ja' far Mohammed ibn Musa al –Khwarizmi Who was an Arabic mathematician who insentient a set of rules for performing the four basic arithmetic operation (addition, subtraction, multiplication and division) on decimal number.

Algorithm must satisfy following criteria or following properties or characteristics of algorithm

- It must be finite (algorithm must terminate)
- It must be unambiguous (Definiteness)
- It must be executable (Non- intuitiveness)
- It must have 0 or more input (input)
- It must have 1 or more output (output)
- It should be general so that it can solve any problem of a particular type for which it is constructed (completeness/ generally)
- For every input instance it halts with correct output (correct)

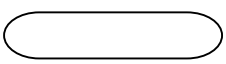
Conventions used in writing Algorithm:

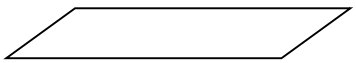
1. **Name of algorithm:**
   - Every algorithm is given a name which reflects the task performed by it.
2. **Introductory comments:**
   - The task performed by algorithm is described briefly any assumption made by algorithm are mentioned along with variable used and data types
3. **Steps :**
   - Any algorithm is made up of sequence of numbered steps. Each step is proceeded by a brief comment describing its function
4. **Comments** are included within the body of step by enclosing them in parentheses.

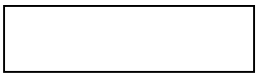
Flowchart

- Graphical or pictorial representation of algorithm is called flowchart.
- It shows logic of algorithm & flow of control.
- Flow chart uses different symbols to show action & arrow to show flow of control.
- Flow charts are language independent.
- **Symbols used in flowchart are as follows.**


Flow Control

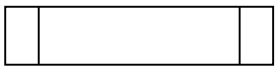

Start or Stop


Input or Output


Process or Computation


Decision


Connector



Subroutine

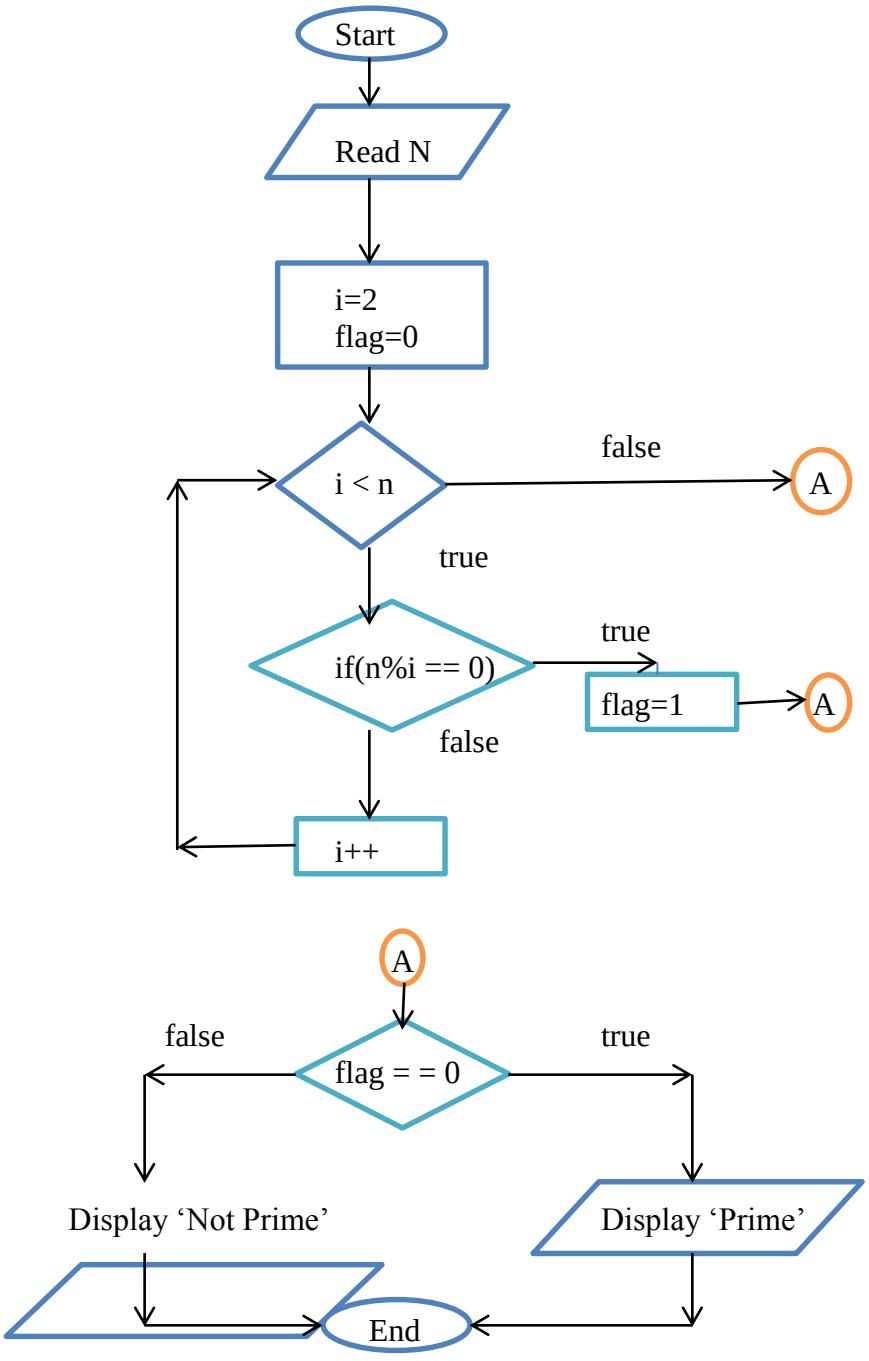
Secondary Storage

Flowchart and algorithm to check for prime number.

Algorithm to Check for Prime Number

1. Start
2. Read N
3. Set i=2
   flag=0
4. Repeat through 6 while(i< n)
5. if(n % i == 0)
   flag=1
   goto 7 [break]
   else
   goto 6
6. i++
7. if(flag==0)
   Display ‘Prime’
   else
   Display ‘Not Prime’
8. End

Flowchart to Check for Prime Number



Algorithm to compute roots of quadratic equation[$ax^2+bx+c=0$]…

Coefficients:- a,b,c

Discriminant:- D

Roots:- X1 , X2

1. Start
2. Read a,b,c
3. if(a == 0)
   display ‘It is linear equation’
   End
4. D = b*b – 4*a*c
5. if(D == 0)
   Display ‘Roots are real and imaginary’
   R = -b / (2*a)
   I = sqrt(D)/(2*a)
   Display (‘Root1=’ , R, ‘+j’, I)
   Display (‘Root2=’ , R, ‘-j’ , I)
   else if(D > 0)
   Display ‘Roots are real and unequal’
   X1=(-b+sqrt(D)/(2*a))
   X2=(-b-sqrt(D)/(2*a))
   Display(‘Root1=’, X1)
   Display(‘Root2=’,X2)
   else
   Display ‘Roots are real and equal’
   X1 = -b/(2*a)
   Display(‘Root1=’, X1)
   Display(‘Root2=’,X1)
6. End