SNS COLLEGE OF ENGINEERING

Kurumbapalayam (Po), Coimbatore – 641 107

An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

COURSE NAME : 20CS101 PROGRAMMING FOR PROBLEM SOLVING

I YEAR /I SEMESTER

Unit 2- C PROGRAMMING BASICS

Topic 1: Introduction to C Program And Fundamental Rules

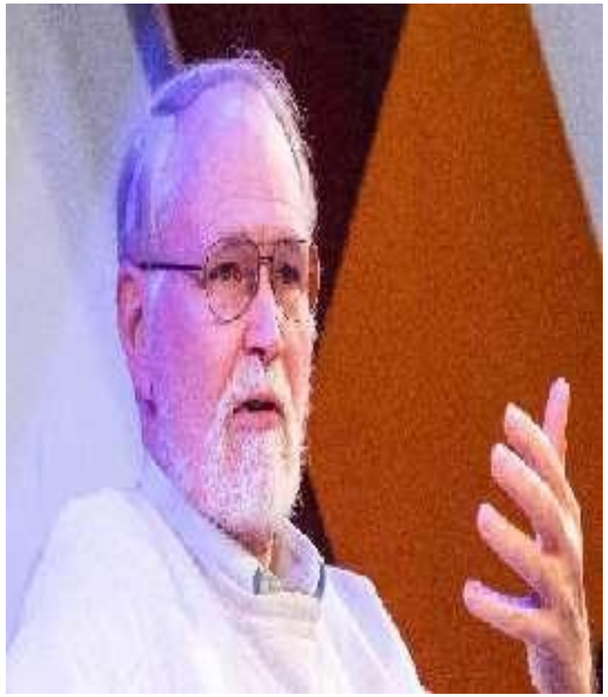
UNITII	C PROGRAMMING BASICS	9
Introduction to ‘C’ Programming –Fundamental rules – Structure of a ‘C’ program – Compilation and Linking processes –Constants, Variables, keywords, Identifier, Delimiters – Declaring and Initializing variables – Data Types – Operators and Expressions –Managing Input and Output operations – Decision Making and Branching –Looping statements –Illustrative programs.		

Introduction

- C programming language was developed in 1972 by **Dennis Ritchie** at Bell laboratories of AT&T (American Telephone & Telegraph), located in the U.S.A.
- C is a general-purpose, high-level language
- In 1978, **Brian Kernighan and Dennis Ritchie** produced the first publicly available description of C, now known as the K&R standard.

Conti...

Dennis Ritchie



Brian Kernighan

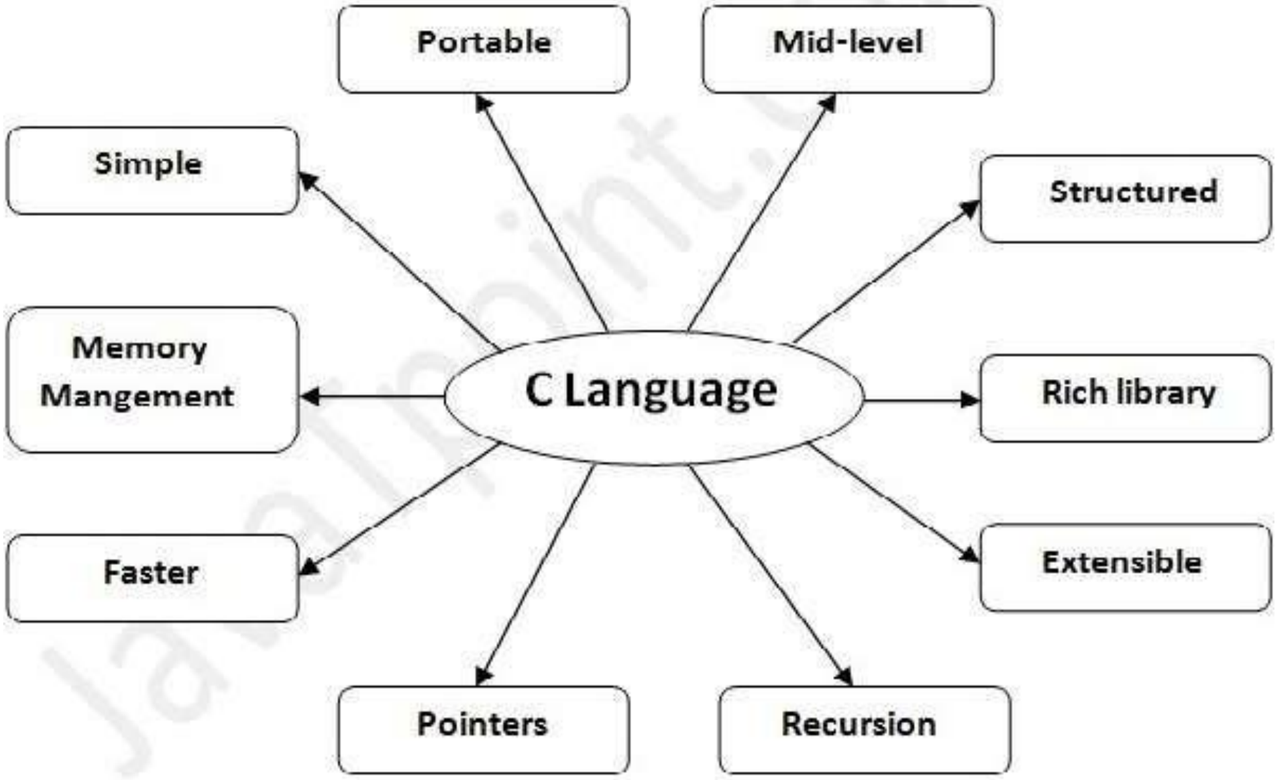
Facts about C

- C was invented to write an operating system called UNIX.
- C is a successor of B language which was introduced around the early 1970s.
- The language was formalized in 1988 by the American National Standard Institute (ANSI).
- The UNIX OS was totally written in C.

Conti...

- Today C is the most widely used and popular System Programming Language.
- Most of the state-of-the-art software have been implemented using C.
- Today's most popular Linux OS and RDBMS MySQL have been written in C.
- It inherits many features of previous languages such as B and BCPL(BASIC COMBINED PROGRAMMING LANGUAGE).

Features and Applications of C Language



Example of C Program

```
#include <stdio.h>

void main( )

{

printf("Hello C Language");

}
```

Compile and Execute C Program in linux

- Open a text editor and add the above-mentioned code.
- Save the file as *hello.c*
- Open a command prompt and go to the directory where you have saved the file.
- Type *gcc hello.c* and press enter to compile your code.
- If there are no errors in your code, the command prompt will take you to the next line and would generate *a.out* executable file.

Conti...

- If there are no errors in your code, the command prompt will take you to the next line and would generate *a.out* executable file.
- Now, type *./a.out or ./hello* to execute your program.
- You will see the output *"Hello World"* printed on the screen.

How to compile and run the c program in windows

- Open the turbo C++ editor
- To save: filename.c (Example: hello.c)
- Type the program
- Compilation: Alt+F9
- Run : Ctrl+F9

Description

- **#include <stdio.h>** includes the **standard input output** library functions.

The printf() function is defined in stdio.h

- **void main()** The **main() function is the entry point of every program**

in c language. Void- the main program does not return any value.

- **printf()** The printf() function is **used to print data** on the console.

Execution Flow

•C program (source code) is sent to preprocessor first. The preprocessor is responsible to **convert preprocessor directives into their respective values**. The preprocessor generates an expanded source code.

•Expanded source code is sent to compiler which compiles the code and converts it into assembly code.

Execution Flow

Basic Structure of C programs

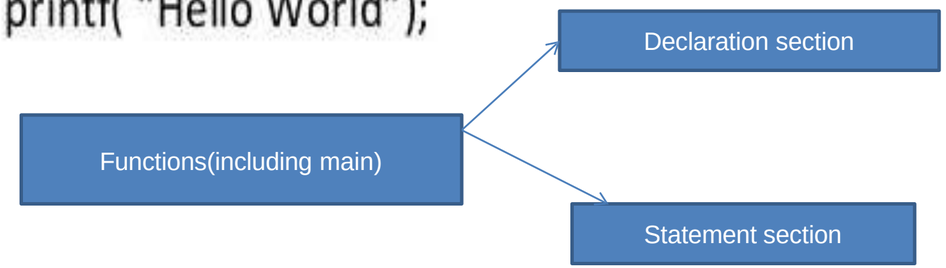
Documentation section represented used comment line //

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

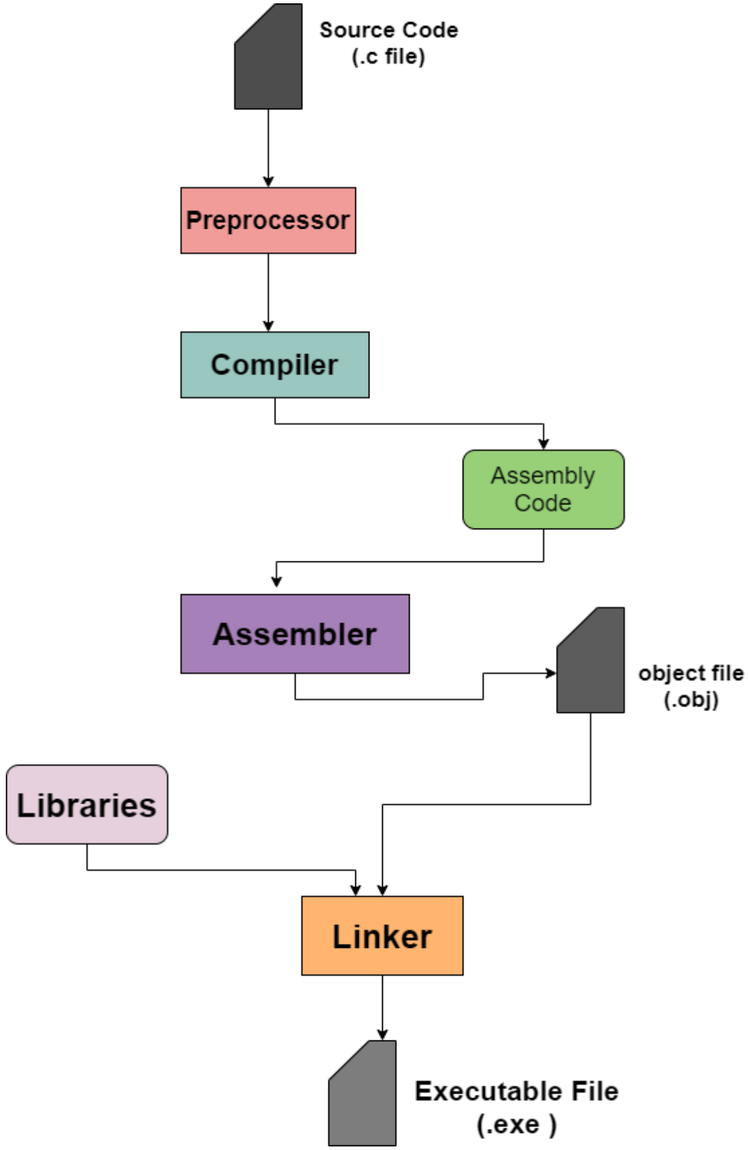
Preprocessor Directive

```
void main ( )
{
    printf( "Hello World");
}
```

void main ( )
- Every C program consists of one or more functions.
- main ( ) is a function. It is the first function to which control is passed from OS when a program is executed
- void indicates that the main function does not return any value



WHAT HAPPENS WHEN C CODE IS COMPILED



Compilation Process

The process of converting the source code written in any programming language(generally, mid-level or high-level language) into machine-level language that is understandable by the computer is known as Compilation. The software that is used for this conversion is known as a **Compiler.**

Source code is the set of instructions and statements written by a programmer using a human-readable computer programming language.

Compilation Process

Preprocessor (preprocessing)

Compiler (compiling)

Assembler (assembly)

Linker (linking)

Compilation Process

Step 0: Pre-processing of the source file

In this phase, pre-processing of the source file is done. The Pre-processor is a program that accepts the C source code file and then it performs the following tasks:

It will remove the comments from the source code.

Compilation Process

Step 1: Preprocessor

It is a program that processes the source program before passing them on to the compiler. At this step the pre-processors used in any C program are handled and the source code is made ready for compilation.

Each preprocessing statement must start with #, where # is called the preprocessor directive.

Each preprocessing directive is a single-line code statement. The word after # is called the preprocessor command.

#include
#define

Compilation Process

Step 2: Compiler

- The expanded code by the preprocessor is then passed on to the compiler.
- A compiler is a program that converts the **high-level language**(or **mid-level language**) code to the **assembly code**, which is then converted into the machine code, which the machine can understand.
- Therefore the preprocessed code given by the preprocessor to the compiler is then converted into assembly code by the compiler, which is then passed on to the **Assembler**.

Compilation Process

Step 3: Assembler

- The assembler converts the assembly code that it gets from the compiler into the **object code**.
- The extension of the file in this step becomes (**.obj**).
- Don't think that Assembler is a separate program generating the object code.

The Assembler is part of the compilation process of the C language source code

Compilation Process

Step 4: Linker

A linker is a tool that is used to link all the parts of a program together in order of execution.

The code after this stage becomes Executable Machine code.

There might be some situations when our program refers to the functions that are defined in some other files. Or, if the code for some program is too big, we can break it into two files, which will be compiled separately and then linked using the Linker.

Compilation Process

Step 4: Linker

In the C language compilation process, the Linker plays a very important role.

If your C program includes a header file, and you are using some function defined in that header file, then the Linker will link the required object code for the function in the library, to the object code of your program and package them together.

Example C Program

```
#include<stdio.h>
void main()
{
int a, b, sum;
printf("\nEnter two no: ");
scanf("%d %d", &a, &b);
sum = a + b;
printf("Sum : %d", sum);
}
```

OUTPUT:

Enter two no:5 6

Sum:11

Assessment 1

1. What is C Program?

Ans : _____

2. What are all the Features of C Program?

Ans : _____

References

1. Reema Thareja, “Programming in C”, Oxford University Press, Second Edition, 2016

Thank You