MODULE 2

LECTURE NOTE-13

FUNCTION

MONOLITHIC VS MODULAR PROGRAMMING:

1. Monolithic Programming indicates the program which contains a single function for the large program.
2. Modular programming help the programmer to divide the whole program into different modules and each module is separately developed and tested. Then the linker will link all these modules to form the complete program.
3. On the other hand monolithic programming will not divide the program and it is a single thread of execution. When the program size increases it leads inconvenience and difficult to maintain.

Disadvantages of monolithic programming: 1. Difficult to check error on large programs. 2. Difficult to maintain. 3. Code can be specific to a particular problem. i.e. it can not be reused.

Advantage of modular programming: 1. Modular program are easier to code and debug. 2. Reduces the programming size. 3. Code can be reused in other programs. 4. Problem can be isolated to specific module so easier to find the error and correct it.

FUNCTION:

A function is a group of statements that together perform a task. Every C program has at least one function, which is **main()**, and all the most trivial programs can define additional functions.

Function Declaration OR Function Prototype:

1. It is also known as function prototype .
2. It inform the computer about the three things

   a) Name of the function

   b) Number and type of arguments received by the function.

   c) Type of value return by the function

Syntax :

return_type function_name (type1 arg1 , type2 arg2);

OR

return_type function_name (type1 type2);

3. Calling function need information about called function .If called function is place before calling function then the declaration is not needed.

Function Definition:

1. It consists of code description and code of a function .

   It consists of two parts

   a) Function header

   b) Function coding

Function definition tells what are the I/O function and what is going to do.

Syntax:

```
return_type  function_name (type1  arg1 , type2 arg2)
 {
local variable;

statements ;

return (expression);

  }
```

2. Function definition can be placed any where in the program but generally placed after the main function .
3. Local variable declared inside the function is local to that function. It cannot be used anywhere in the program and its existence is only within the function.
4. Function definition cannot be nested.
5. Return type denote the type of value that function will return and return type is optional if omitted it is assumed to be integer by default.

USER DEFINE FUNCTIONS VS STANDARD FUNCTION:

User Define Fuction:

A function that is declare, calling and define by the user is called user define function. Every user define function has three parts as:

1. Prototype or Declaration
2. Calling
3. Definition

Standard Function:

The **C standard library** is a standardized collection of header files and library routines used to implement common operations, such as input/output and character string handling. Unlike other languages (such as COBOL, FORTRAN, and PL/I) C does not include built in keywords for these tasks, so nearly all C programs rely on the standard library to function.