Course Name: O Level (2nd Sem B1 Batch)
Topic: Function in C

Subject: C Language
Date: 03-April-2020

What is 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**. You can divide up your code into separate **functions**.

The function is a self contained block of statements which performs a coherent task of a same kind.

Types of functions :

- **Build in Functions**
- **User Defined Functions**

Built in Functions :

These functions are also called as **'Library Functions'**. These functions are provided by system. These functions are stored in library files.

Example:
scanf( ), printf( ), strcpy( ), strlwr( ), strcmp( ), etc.

User Defined Functions :

The functions which are created by user for program are known as **'User defined functions'**.

Advantages :

- It is easy to use, It reduces the size of a program.
- Debugging is more suitable for programs.
- It is easy to understand the actual logic of a program.
- Highly suited in case of large programs.
- By using functions in a program, it is possible to construct modular and structured programs.

FUNCTION TYPES:

- ✓ Function with No arguments and No return value
- ✓ Function with arguments but No return value
- ✓ Function with No arguments but Returns a value
- ✓ Function with arguments and return values

What is an argument ?

An argument is an entity used to pass the data from calling function to the called function.
There are two types of arguments:
1. Formal
2. Actual

Actual arguments:

The arguments that are passed in a function call are called actual arguments. These arguments are defined in the calling function.

Formal arguments:
The formal arguments are the parameters/arguments in a function declaration. The scope of formal arguments is local to the function definition in which they are used. Formal arguments belong to the called function. Formal arguments are a copy of the actual arguments. A change in formal arguments would not be reflected in the actual arguments.

There are two ways to call a function (used when invoking functions)

Call by Value:

- Copy of argument pass to a function
- Changes in function do not effect original

Call by reference:

- Passes original arguments to a function
- Changes in function effect original

<u>Try Yourself</u>

1. What is a function?
2. Difference between user defined and library functions.
3. How many types of an argument in functions.
4. Write some advantages of user defined functions.
5. Difference between call by value and call by reference.