Computer Concepts and C Programming (06CCP13)

Unit VIII

User-Defined Function

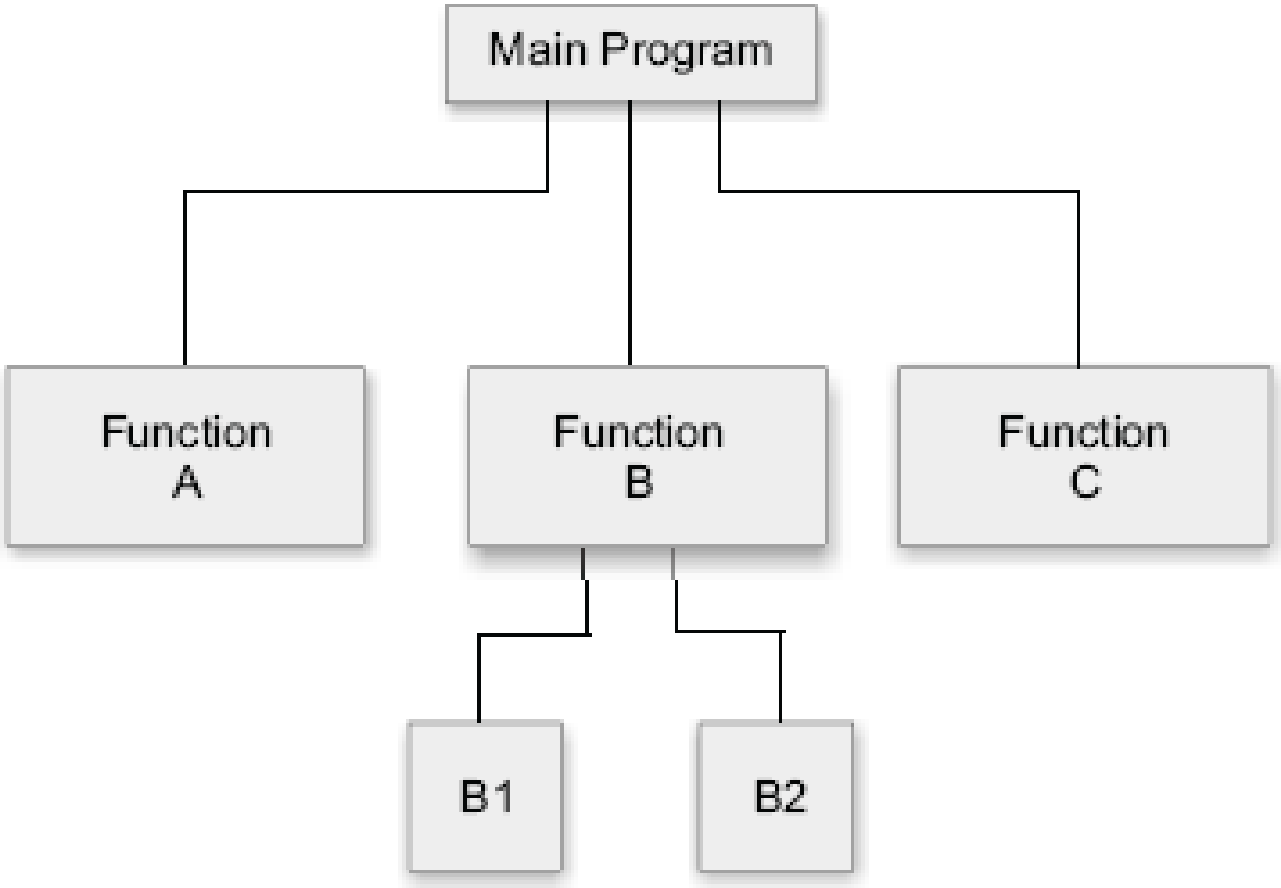
Functions are very much useful when a block of statements are has to be written / executed again and again. Functions are used when the program size is too large or complex. Functions are called to perform each task sequentially from the main program. Functions are also used to reduce the difficulty during the debugging a program. A function can be used any number of time as needed by the user in the same program or different program.

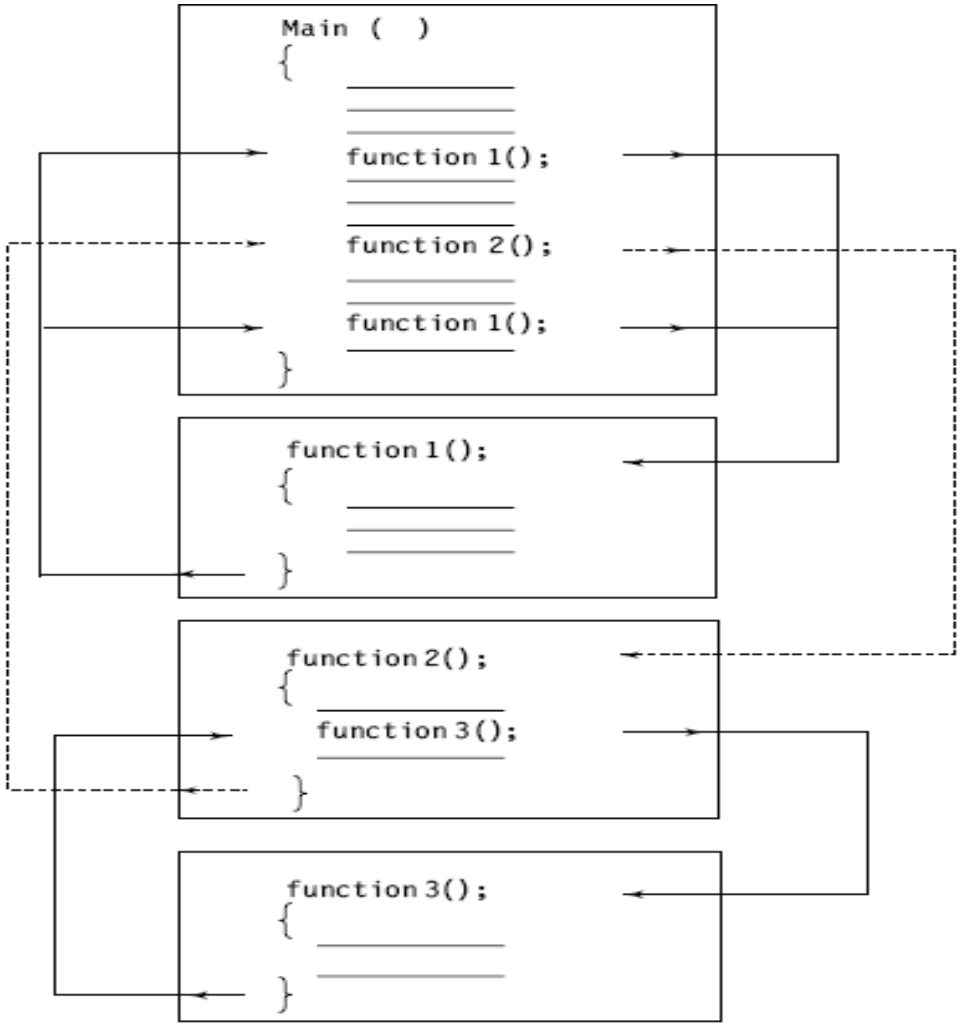
User-Defined Function

The program may become too large and complex and as a result the task of debugging, testing and maintaining becomes difficult. If a program is divided into functional parts, then each part may be independently coded and later combined into a single unit. These subprograms called ‘functions’ are much easier to understand, debug, and test.

User-Defined Function

- C functions can be classified into two categories, namely, library functions and user-defined functions.
- main is an example of user-defined functions. printf and scanf belong to the category of library functions. We have also used other library functions such as sqrt, cos, strcat, etc.
- The main distinction between these two categories is that library functions are not required to be written by us whereas a user-defined function has to be developed by the user at the time of writing a program.


Main Program
Function A
Function B
Function C
B1
B2


Main ( )
{
function 1();
function 2();
function 1();
}
function 1();
{
}
function 2();
{
function 3();
}
function 3();
{
}

User-Defined Function

Structure of function

```
Function-name(argument list)
{
local variable declarations;
executable statement1;
executable statement2;
-------------------------;
-------------------------    ;
return(expression);
}
```

User-Defined Function

Argument List

Ex: Quadratic(a,b,c)

Power(x,n)

Mul(a,b)

Category1: Functions with no arguments and no return values.
Category2: Functions with arguments and no return values.
Category3: Functions with arguments and return values.

Page 247