# FACULTY OF ENGINEERING & TECHNOLOGY

# CSPS103: Object Oriented Programming

# Lecture-09

Preeti Singh

Department of Computer Science & Engineering
Rama University,  Kanpur

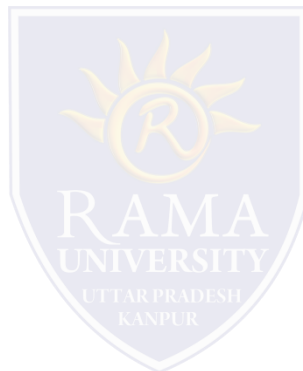preeti.ru@ramauniversity.ac.in

# OBJECTIVES

**In this lecture, you will learn to:**

❖**C++ Functions**

  ❖**Function declaration**

  ❖**Function definition**

  ❖**Calling a Function**

# C++ FUNCTIONS

❑A function groups a number of program statements into a unit and gives it a name.

❑This unit can then be invoked from other parts of the program.

❑The function's code is stored in only one place in memory, even though the function is executed many times in the course of the program's execution.

❑Functions help to reduce the program size when same set of instructions are to be executed again and again.

❑A general function consists of three parts, namely, function declaration (or prototype), function definition and function call.
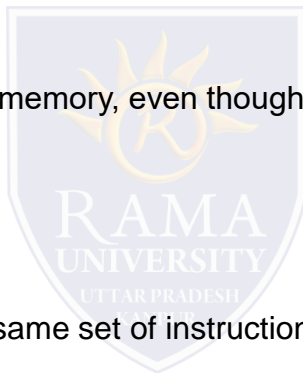
## Function declaration — prototype:

❑ A function has to be declared before using it, in a manner similar to variables and constants.

❑A function declaration tells the compiler about a function's name, return type, and parameters and how to

call the function.

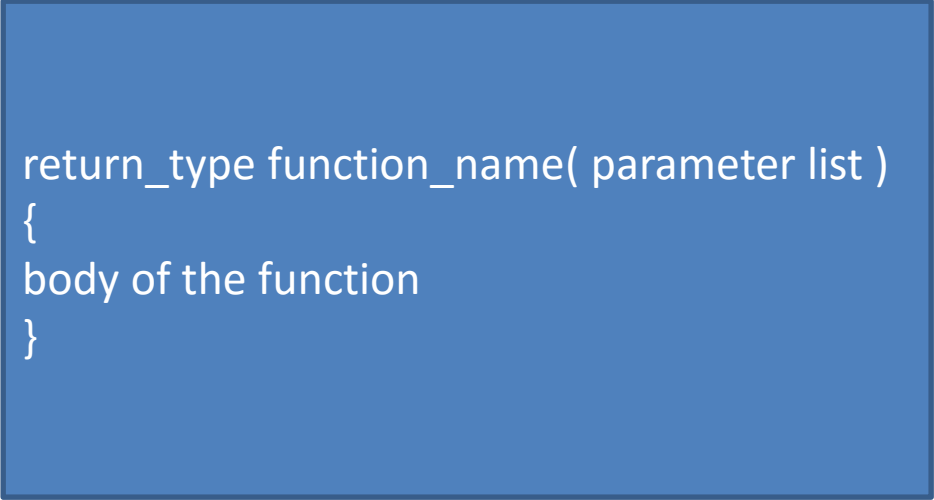❑The general form of a C++ function declaration is as follows:
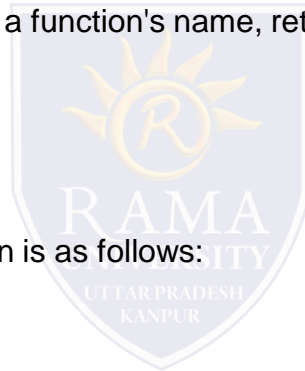
return_type function_name( parameter list );

## Function definition

❑ The function definition is the actual body of the function.

❑The function definition consists of two parts namely, function header and function body.

❑The general form of a C++ function definition is as follows:

```
return_type function_name( parameter list )
{
body of the function
}
```

## Calling a Function

❑To use a function, you will have to call or invoke that function.

❑To call a function, you simply need to pass the required parameters along with function name.

❑If function returns a value, then you can store returned value.
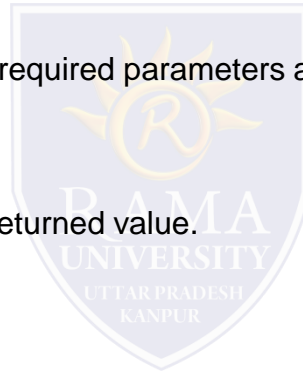
## A c++ program calculating factorial of a number using functions

```cpp
#include<iostream.h>
#include<conio.h>
int factorial(int n); //function declaration
int main(){
int no, f;
cout<<"enter the positive number:-";
cin>>no;
f=factorial(no); //function call
cout<<"\nThe factorial of a number"<<no<<"is"<<f;
return 0;
}
int factorial(int n) //function definition
{ int i , fact=1;
for(i=1;i<=n;i++){
fact=fact*i;
}
return fact;
}
```

# REFERENCES

- Kernighan, Brian W., and Dennis M. Richie. The C Programming Language. Vol. 2. Englewood Cliffs: Prentice-Hall, 1988.

- King, Kim N., and Kim King. C programming: A Modern Approach. Norton, 1996.

- Bjrane Stroustrup, "C++ Programming language",3rd edition, Pearson education Asia(1997)

- Lafore R."Object oriented Programming in C++",4th Ed. Techmedia,New Delhi(2002).

- Yashwant Kenetkar,"Let us C++",1stEd.,Oxford University Press(2006)

- B.A. Forouzan and R.F. Gilberg,CompilerScience,"A structured approach using C++" Cengage Learning, New Delhi.

- https://www.javatpoint.com/cpp-tutorial

- https://www.tutorialspoint.com/cplusplus/index.htm

- https://ambedkarcollegevasai.com/wp-content/uploads/2019/03/CPP.pdf

- https://onlinecourses.nptel.ac.in/noc20_cs07/unit?unit=3&lesson=19

## Multiple Choice Question:

**Q1. Where does the execution of the program starts?**

a) user-defined function
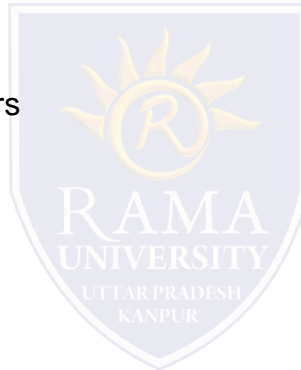
b) main function

c) void function
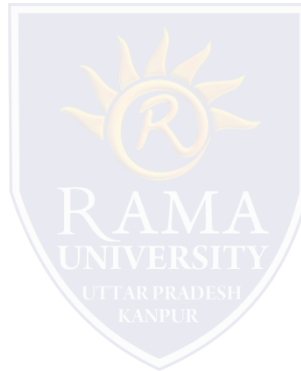
d) else function

## Multiple Choice Question:

**Q2. What are mandatory parts in the function declaration?**

    a) return type, function name

    b) return type, function name, parameters

    c) parameters, function name

    d) parameters, variables

## Multiple Choice Question:

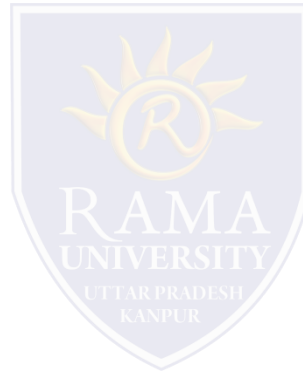**Q3. which of the following is used to terminate the function declaration?**

a) :

b) )

c) ;

d) ]

## Multiple Choice Question:

**Q4. How many can max number of arguments present in function in the c99 compiler?**
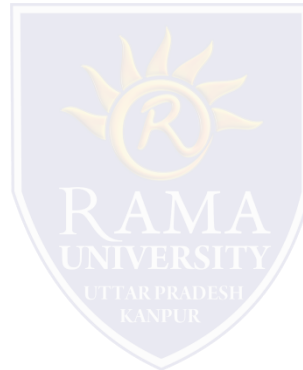
a) 99

b) 90

c) 102

d) 127

## Multiple Choice Question:

**Q5. What is the scope of the variable declared in the user defined function?**

a) whole program

b) only inside the {} block

c) the main function

d) header section

# Summary

## In this lecture, you learned that:

➢ Concept of function.

➢ Concept of

    1) Function declaration
    2) Function definition
    3) Calling a Function