Fundamentals of Programming: Functions in C++

Abdul Haseeb

Agenda

- Definition of Function
- Declaration of Function
- Calling a Function
- Types of functions
- Function prototype
- Void vs return type
- Formal and Actual Parameters
- Call by value vs call by Reference
- Type casting in functions

Function

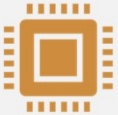
A **block of code** that performs some specific task.

Problem: Write code to Draw a circle on screen, and color it.

Let's say we need the code for above mentioned task for multiple times at different time instances and with different color

Function

Put commonly or repeatedly done tasks together, to avoid writing the same code again and again for different inputs.



A function is a block of code that runs only when it is called, exception is the main function (which is called automatically by OS)

Function

- A piece of code which takes input, processes it and produces output


```
int max(int x, int y)
{
    if (x > y)
        return x;
    else
        return y;
}

// main function that doesn't receive any parameter and
// returns integer
int main()
{
    int a = 10, b = 20;

    // Calling above function to find max of 'a' and 'b'
    int m = max(a, b);

    cout << "m is " << m;
    return 0;
}
```

Advantages of functions

- Code Organization
- Code Readability
- Code Reusability

Types of functions

- Two Types of functions:
  - Pre-defined/Standard Library Functions
  - User defined functions

C++ User Defined function

- C++ allows users to create their own functions
- User defines a function and groups the code in a block
- User calls the function and it is executed

C++ function declaration


```
returnType functionName (parameter1, parameter2,...) {
    // function body
}
```

1. Return Type, Function name are mandatory
2. Parameters are optional

Function Declaration

```
// function declaration
void greet() {
    cout << "Hello World";
}
```

Here,

- the name of the function is `greet()`
- the return type of the function is `void`
- the empty parentheses mean it doesn't have any parameters
- the function body is written inside `{}`

Return type

- It can be void, which means function doesn't return any value.
- We have to omit return statement (Except the empty return)

```
void displayNumber() {
    // code
}
```

- If function has a return type like int, string or float, then it should return a value of that type.
- Any code after return statement is not executed, and return cant be empty.

```
int add (int a, int b) {
   return (a + b);
}
```

Calling a function

- We call the function by writing the function name.
- If function returns some value, we store it in a variable or we can direct it to console screen.
- If function doesn't return any value, we can simply write it's name only.

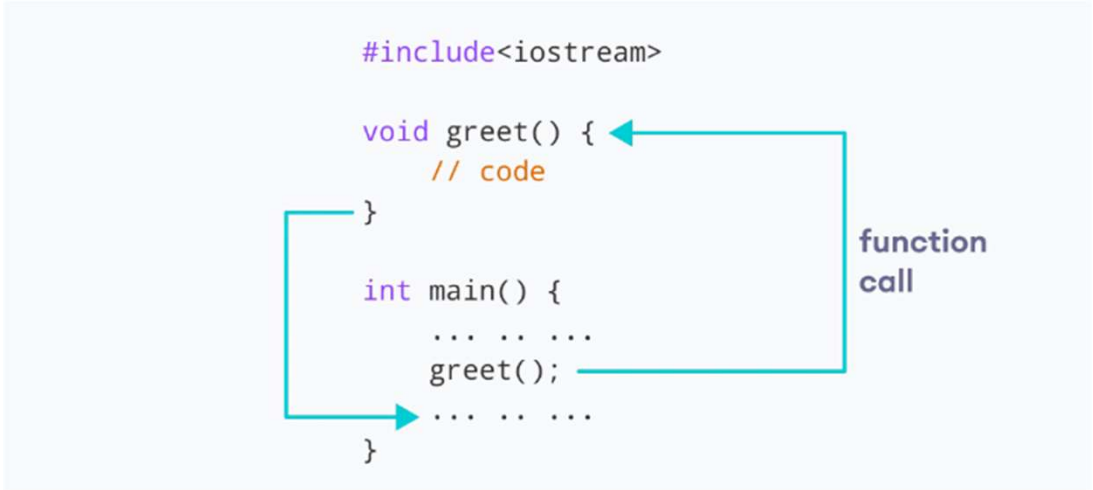
Example of calling a function

```
int main() {

    // calling a function
    greet();

}
```

How function works in C++



Activity

DEFINE A FUNCTION NAMED AS FIND_MAX, WHICH FINDS OUT MAXIMUM NUMBER AMONG ANY TWO GIVEN NUMBERS AND RETURNS THE MAXIMUM NUMBER