LEARN PROGRAMMING

STUDY GROUP - SESSION #5

Weekly: Wednesday 19:15 to 22:15
R307 G29

TODAY?

1. Functions
2. Types of Functions
3. Using Functions
    a. Declaration
    b. Definition
    c. Calling
4. Function Arguments
5. Return Value
6. Categories
7. PRACTICE..PRACTICE..PRACTICE

Functions

In programming, function refers to a segment that **groups code** to perform a specific task.

Depending on whether a function is **predefined** or **created by programmer**; there are two types of function:

1. Library Function
2. User-defined Function

Type of Functions

Library Function

These functions are the built-in function in C++ programming.

Programmer can use library function by invoking function directly; they don't need to write it themselves.

Include Header file

```
#include <iostream>
#include <cmath>

using namespace std;

int main()
{
      double number, squareRoot;
      cout << "Enter a number: ";
      cin >> number;

      // sqrt() is a library function to calculate square root
      squareRoot = sqrt(number);
      cout << "Square root of " << number << " = " << squareRoot;
      return 0;
}
```

Using function of cmath

Type of Functions

User-defined Function

C++ allows programmer to define their own function.

A user-defined function ***groups code to perform a specific task*** and that group of code is given a name(identifier).

When the function is invoked from any part of program, it will execute the code defined in the body of function.

3 Step process

```
# include <iostream>
using namespace std;

int add(int, int);

int main() {
    ... .. ...
    sum = add(num1, num2);
}

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

2 Step process

```
#include <iostream>

void function_name() {
    ... .. ...
    ... .. ...
}

int main() {
    ... .. ...
    function_name();
    ... .. ...
}
```

Type of Functions

User-defined Function

- Sum of two numbers
- Add function

```cpp
#include <iostream>
using namespace std;

// Function prototype (declaration)
int add(int, int);

int main()
{
      int num1, num2, sum;
      cout<<"Enters two numbers to add: ";
      cin >> num1 >> num2;

      // Function call
      sum = add(num1, num2);
      cout << "Sum = " << sum;
      return 0;
}

// Function definition
int add(int a, int b)
{
      int add;
      add = a + b;

      // Return statement
      return add;
}
```

1.Declaration

3.Calling

3.Definition

USING FUNCTIONS

1.Declaration

- Compiler needs to know function before it is being used.
- Declaration is only telling compiler that there is a function definition coming later with a specific name and arguments.

3.Definition

- An actual body of function, which contains the lines of code that need to be packed in a function block.

3.Calling

- Using function where needed by calling name of functions with arguments (if needed) and a return value (if defined)

Note: It is not necessary to define prototype if user-defined function exists before `main()` function.

3 Step process

```
# include <iostream>
using namespace std;

int add(int, int);

int main() {
    ... .. ...
    sum = add(num1, num2);
}

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

2 Step process

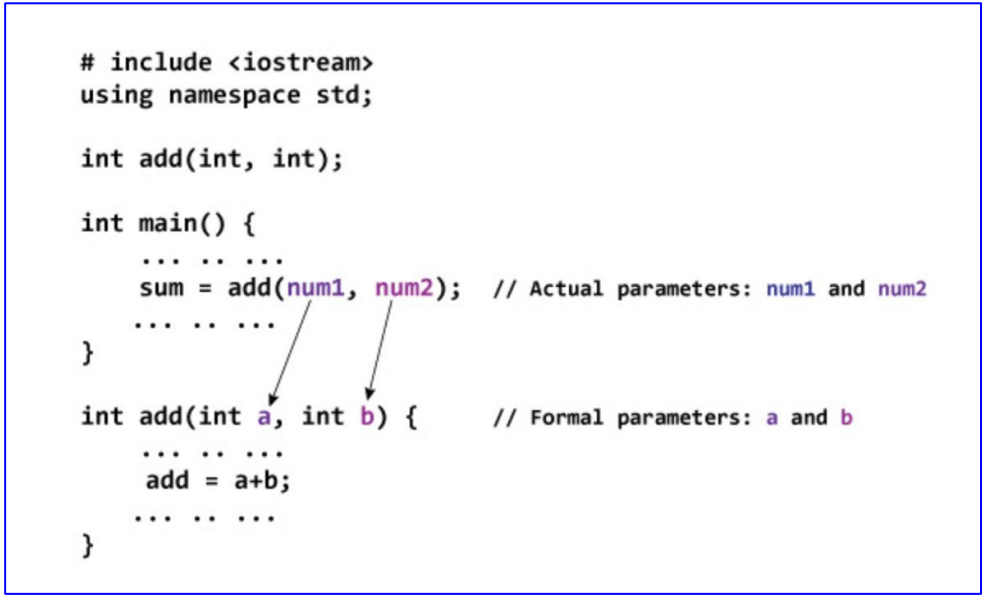
```
#include <iostream>

void function_name() {
    ... .. ...
    ... .. ...
}

int main() {
    ... .. ...
    function_name();
    ... .. ...
}
```

Function Arguments

- An argument (parameter) refers to the data.
- Which is defined, passed to a function in following steps.
  - Declaration (data type)
  - Definition
  - Calling
- In example, **int** data type arguments are declared in declaration of function, defined formal names **a** and **b** and are passed value from variables **num1** and **num2** respectively.



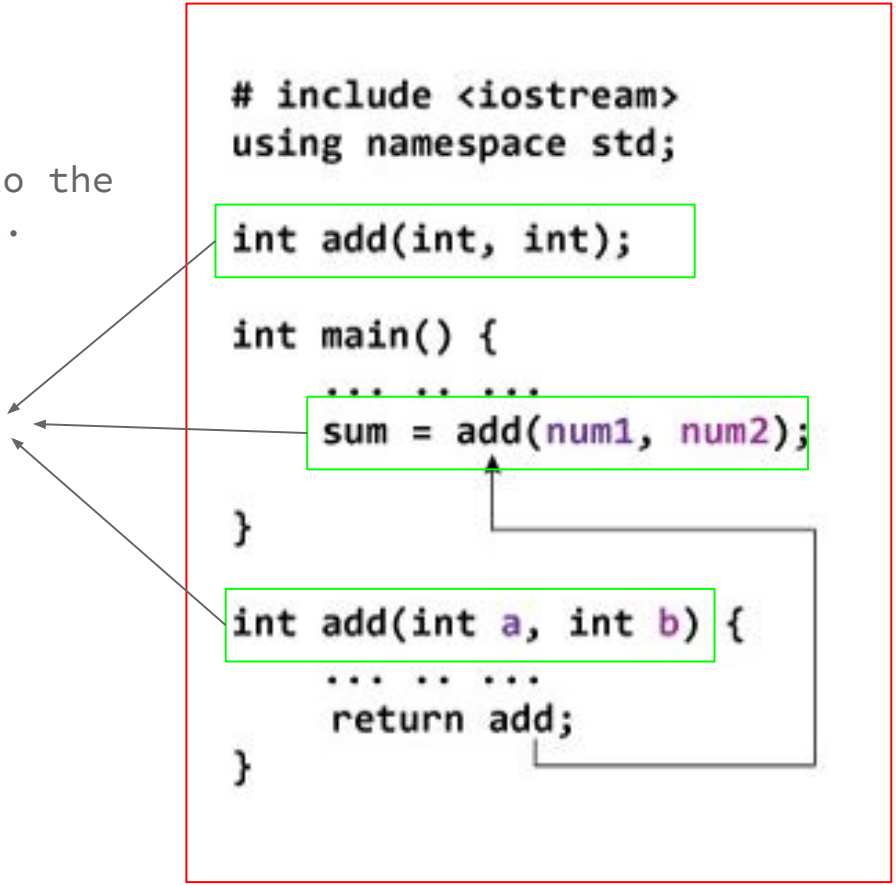
Example sum of two numbers

Return Value

A function can return a single value to the calling program using return statement.

- Datatype **int** used as a return type
- The **function add** will return a value that is saved in a variable **sum.**

If no value is return use **void** as datatype.



Categories

For better understanding of arguments and return in functions, user-defined functions can be categorised as:

1. Function with **no argument** and **no return value**
2. Function with **no argument** but **return value**
3. Function with **argument** but **no return value**
4. Function with **argument** and **return value**

C++ Functions example

Task For Today? (using functions)

1. Program to solve quadratic equation;

The Quadratic Formula: For $ax^2 + bx + c = 0$, the values of x which are the solutions of the equation are given by:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

2. C++ Programs to “Check if is a Prime Number”
   By Creating a Function; Use all categories of functions
   (Hint: Create 4 separate programs)

Next Session Topics?

Arrays

- One-dimensional
- Two-dimensional
- Three-dimensional

THANK YOU

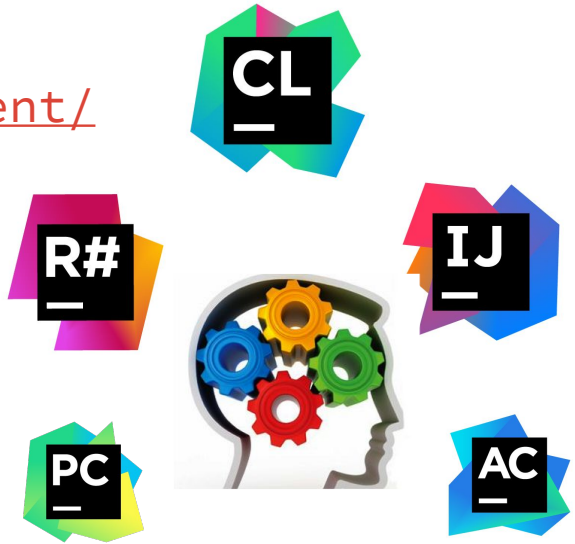
Let's get started

TOOL & TECH

Integrated Development Environment (IDE)

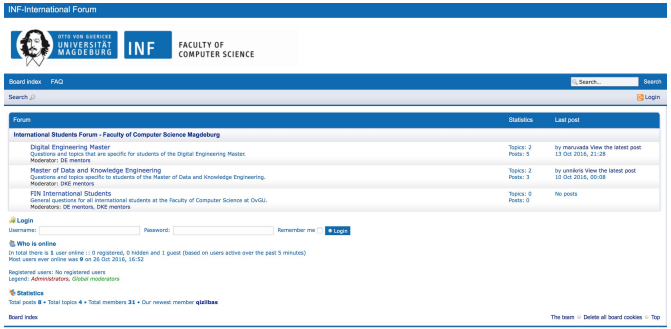
JETBRAINS: https://www.jetbrains.com/student/

CLION - C/C++ (Install THIS)



HOW TO REACH US?

1. FORUM for International students?
2. Link: https://iif.cs.uni-magdeburg.de/index.php
3. “Study group programming”
4. Login with your university ID.
5. Post your questions on FORUM.
6. Moderators will reply.



OFFICIAL COMMUNICATION ---> FORUM ONLY