

Functions in C Plus Plus

Module 2 - Call by Value and Call by Reference in C Plus Plus

 CBSE

Function Arguments

Arguments are the values you pass to a function when calling it. These values are assigned to the function's parameters.

 ICSE

Types of Arguments:

1. Positional Arguments:

- You pass values in the order of the function's parameters.
- Example: add(3, 4); (3 goes to a, 4 goes to b).

 NTSE

2. Default Arguments:

- Parameters can have default values, which are used if no argument is provided.
- Example:

 Banking & Insurance

```
int add(int a, int b = 5) {  
    return a + b;  
}
```

 Central Govt. Service

```
cout << add(3); // Uses default value for b (Output: 8)
```

 State Govt. Services

3. Pass-by-Value:

- A copy of the argument is passed, and changes inside the function do not affect the original value.
- Example:

```
void modify(int x) {  
    x = 10; // Only modifies the local copy  
}
```

 LAW Entrance

4. Pass-by-Reference:

- The actual variable is passed, so changes inside the function affect the original value.

- Use & to indicate reference:

```
void modify(int &x) {  
    x = 10; // Modifies the original variable  
}
```

 MBA Entrance

 Railways & Metro Services

Return Statement in C++

The **return statement** returns the flow of the execution to the function from where it is called. This statement does not mandatorily need any conditional statements. As soon as the statement is executed, the **flow of the program**

...many more

abhyasonline.in

**Course
&
Test Series**

 **CBSE**

 **ICSE**

 **NTSE**

 **Banking &
Insurance**

 **Central Govt.
Service**

 **State Govt.
Services**

 **LAW
Entrance**

 **MBA
Entrance**

 **Railways & Metro
Services**

...many more

abhyasonline.in

Introduction to 'C++' Language

stops immediately and returns the control from where it was called. The return statement may or may not return anything for a void function, but for a non-void function, a return value must be returned.

Syntax:

return[expression];

In C++, **Call by Value** and **Call by Reference** are two ways to pass arguments to functions.

Call by Value

In **Call by Value**, a copy of the actual parameter's value is passed to the function. Changes made inside the function do not affect the original variable.

How it works:

- A new memory location is created for the copied value.
- Modifications inside the function are limited to the copy.

Call by Reference

In **Call by Reference**, the actual parameter itself is passed to the function. Changes made inside the function affect the original variable.

How it works:

- The function operates directly on the memory address of the variable.
- No additional memory is allocated for a copy.

Key Differences Between Call by Value and Call by Reference

Aspect	Call by Value	Call by Reference
Data Passed	A copy of the variable's value.	The actual variable itself.
Memory Used	More (for the copy).	Less (no copy is made).
Effect on Original	No effect on the original variable.	Modifies the original variable.
Use	Safe, as changes are local.	Efficient for large data structures.

Solved Example: Swap Two Numbers using Call by Value

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

void swapByValue(int a, int b) {
    int temp = a;
    a = b;
    b = temp;
}
```

**Course
&
Test Series**

 **CBSE**

 **ICSE**

 **NTSE**

 **Banking &
Insurance**

 **Central Govt.
Service**

 **State Govt.
Services**

 **LAW
Entrance**

 **MBA
Entrance**

 **Railways & Metro
Services**

...many more

abhyasonline.in

Introduction to 'C++' Language

```

}
cout << "Inside swapByValue function: a = " << a << ", b = " << b << endl;
}

int main() {
    int x = 5, y = 10;
    cout << "Before swapping (main): x = " << x << ", y = " << y << endl;
    swapByValue(x, y); // Call by value
    cout << "After swapping (main): x = " << x << ", y = " << y << endl; //
    Unchanged
    return 0;
}

```

Output:
 Before swapping (main): x = 5, y = 10
 Inside swapByValue function: a = 10, b = 5
 After swapping (main): x = 5, y = 10

Solved Example: Swap Two Numbers using Call by Reference

```

#include <iostream>
using namespace std;

void swapByReference(int &a, int &b) {
    int temp = a;
    a = b;
    b = temp;
    cout << "Inside swapByReference function: a = " << a << ", b = " << b << endl;
}

int main() {
    int x = 5, y = 10;
    cout << "Before swapping (main): x = " << x << ", y = " << y << endl;
    swapByReference(x, y); // Call by reference
    cout << "After swapping (main): x = " << x << ", y = " << y << endl; // Swapped
    return 0;
}

```

Output:
 Before swapping (main): x = 5, y = 10
 Inside swapByReference function: a = 10, b = 5
 After swapping (main): x = 10, y = 5

Course
&
Test Series

Introduction to 'C++' Language

Assignment

Ques 1: Write a function findLargest(int a, int b, int c) that takes three integers as arguments and returns the largest number.

Task: Call this function from main() and display the result.

Example Input: 10, 25, 15

Example Output: 25

Ques 2: Write a function calculateInterest(float principal, float rate, int time) that calculates and returns the simple interest using the formula:

Simple Interest = (Principal × Rate × Time)/100

Task: Take principal, rate, and time as input in main() and call the function to display the result.

Example Input: Principal = 1000, Rate = 5%, Time = 2 years

Example Output: Simple Interest = 100



CBSE



ICSE



NTSE



Banking &
Insurance



Central Govt.
Service



State Govt.
Services



LAW
Entrance



MBA
Entrance



Railways & Metro
Services

...many more

abhyasonline.in

