**Experiment: 8**

PART A

(PART A: TO BE REFFERED BY STUDENTS)

**Aim:** **To study functions in C++**

**Learning Outcomes: Learner would be able to**

Interpret the scenario to decide on creating modules called as function.

**Task 1: Explain the commented lines.**

#include<iostream>

using namespace std;

int fun(int); //1.

int main()

{

int i = fun(10); //2.

printf("%d\n", --i);

return 0;

}

int fun(int i) //3.

{

return (i++);

}

**Task2: Identify output for below blocks of code. Give justification.**

#include<iostream>

using namespace std;

**void b();**

**void c();**

void a()

{

cout<<”I am in a”<<endl;

b();

}

void b()

{

cout<<”I am in b”<<endl;

c();

}

void c()

{

cout<<”I am in c”<<endl;

}

int main()

{

a();

b();

c();

return 0;

}

**Task 3: What values are printed out by the following C++ program?**

int f(int x) {

return x + 2;

}

int main() {

int x = 5;

cout<<f(x+2)<<endl;

cout<<f(f(x+2));

return 0;

}

**Task 4:** **What values are printed out by the following C++ program?**

#include<iostream>

using namespace std;

int confusion(int x, int y) {

x = 2\*x + y;

return x;

}

int main() {

int x = 2, y = 5;

y = confusion(y, x);

x = confusion(y, x);

cout<< x<<y;

return 0;

}

**Task 5: What is the output? Why did the addition have no effect after coming back to main()?**

#include<iostream>

using namespace std;

void sum(int a, int b, int c)

{

a+=c;

b+=c;

cout<<“Inside the function”<<a<<b;

}

int main()

{

int a, b;

cout<<“Enter two numbers”;

cin>>a>>b;

cout<<“Before adding \n”<<a<<b;

sum(a,b,10);

cout<<“After adding \n”<<a<<b;

return 0;

}

**Task 6:**  Write a function to find factorial of a number. Return type of function is int and takes 1 int parameter.

**Theory:**

**Function** is a collection of statements grouped together to do some specific task. Every C program has at least one function i.e. the main() function.

**Syntax of function declaration**

return\_type function\_name( parameter\_list );

**Syntax of function definition**

return\_type function\_name(parameter list)

{

// Function body

}

**Syntax of function call**

function\_name( parameter\_list );

#include <iostream>

/\* Addition function declaration \*/

int add(int num1, int num2);

/\* Main function definition \*/

int main()

{

/\* Variable declaration \*/

int n1, n2, sum;

/\* Input two numbers from user \*/

cout<<"Enter two numbers: ";

cin>>n1>>n2;

/\*

\* Addition function call.

\* n1 and n2 are parameters passed to add function.

\* Value returned by add() is stored in sum.

\*/

sum = add(n1, n2);

/\* Print value of sum \*/

cout<<"Sum = "<< sum;

return 0;

}

/\*\*

\* Addition function definition.

\*

\* Return type of the function is int.

\* num1 - First parameter of the function of int type.

\* num2 - Second parameter of the function of int type.

\*/

int add(int num1, int num2)

{

int s = num1 + num2;

/\* Return value of sum to the main function \*/

return(s);

}

PART B

(PART B: TO BE COMPLETED BY STUDENTS)

Students must submit the soft copy as per following segments within two hours of the practical. The soft copy must be uploaded on the portal at the end of the practical. The filename should be **PPS\_batch\_rollno\_experimentno Example: PPS\_B2\_B001\_Exp1**

|  |  |
| --- | --- |
| **Roll No.:** | **Name:** |
| **Prog/Yr/Sem:** | **Batch:** |
| **Date of Experiment:** | **Date of Submission:** |

**Task 1:**

**Task 2:**

**Task 3:**

**Task 4:**

**Task 5:**

**Task 6:**

**Task 7:**

**Conclusion (Learning Outcomes):** Reflect on the questions answered by you jot down your learnings about the Topic: Functions

**Home Work Questions:**

1. When is it necessary to give the function prototype (declaration)?
2. Differentiate between formal parameters and actual parameters.
3. Write a menu-based program in C++ (Use Switch case and do while) that uses a set of functions to perform the following operations:

a. Addition of two complex numbers

b. Subtraction of two complex numbers

c. Multiplication of two complex numbers