

# SWT11022: Practical for Fundamentals of Programming

Department of Information & Communication Technology

Faculty of Technology

South Eastern University of Sri Lanka

Time: - 08.30 am - 11.30 am

Labsheet 08

**Title:** Introduction to the Functions

## Objective:

- Understand and practice basic function creation and calling.
- Understand and practice functions with arguments and return values.
- Understand and practice recursive functions.
- Understand and practice using standard library functions.

## Practical 1: Basic Function

### Steps:

#### 1. Function Declaration:

- **Declare a function prototype for a function that calculates the square of an integer.**

```
int square (int num);
```

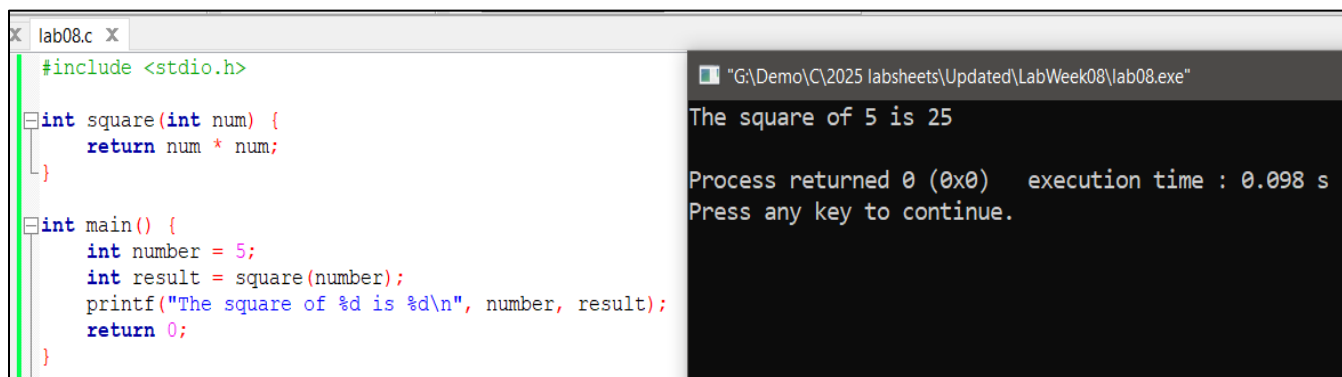
#### 2. Function Definition:

- **Define the square function to calculate the square of an integer.**

```
int square (int num) {  
    return num * num;  
}
```

#### 3. Function Call:

- **Call the square function to calculate and print the square of a number.**



The screenshot shows a C program in a text editor and its execution output in a terminal window. The code defines a function `square` that takes an integer and returns its square, and a `main` function that calls `square` with the value 5 and prints the result.

```
#include <stdio.h>  
  
int square(int num) {  
    return num * num;  
}  
  
int main() {  
    int number = 5;  
    int result = square(number);  
    printf("The square of %d is %d\n", number, result);  
    return 0;  
}
```

The terminal output shows the program's execution, displaying the message "The square of 5 is 25" and indicating that the process returned 0 with an execution time of 0.098 seconds.

```
"G:\Demo\C\2025 labsheets\Updated\LabWeek08\lab08.exe"  
The square of 5 is 25  
Process returned 0 (0x0)   execution time : 0.098 s  
Press any key to continue.
```

## Practical 2: Function with Arguments and Return Values

### Steps:

#### 1. Function Declaration:

- Declare a function prototype for a function that calculates the sum of two integers.

```
int add (int a, int b);
```

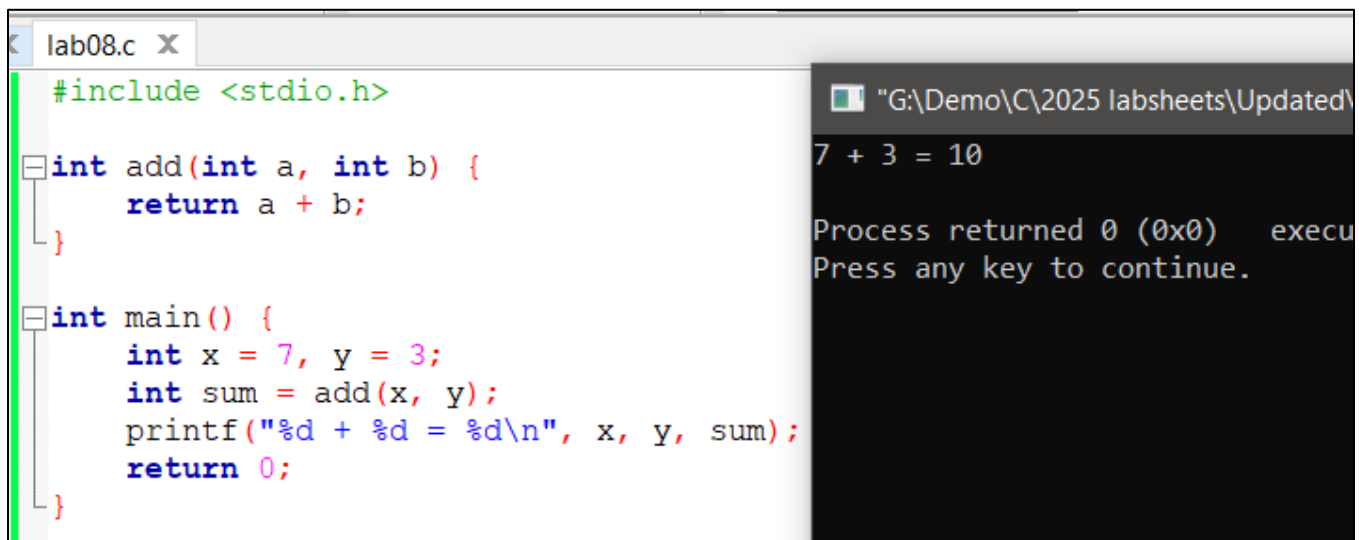
#### 2. Function Definition:

- Define the add function to calculate the sum of two integers.

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

#### 3. Function Call:

- Call the add function with two numbers and display the result.



The screenshot shows a code editor window titled 'lab08.c' with the following C code:

```
#include <stdio.h>  
  
int add(int a, int b) {  
    return a + b;  
}  
  
int main() {  
    int x = 7, y = 3;  
    int sum = add(x, y);  
    printf("%d + %d = %d\n", x, y, sum);  
    return 0;  
}
```

To the right of the code editor is a terminal window showing the execution output:

```
"G:\Demo\C\2025 labsheets\Updated\  
7 + 3 = 10  
  
Process returned 0 (0x0)   execu  
Press any key to continue.
```

### Practical 3: Recursive Function

#### Steps:

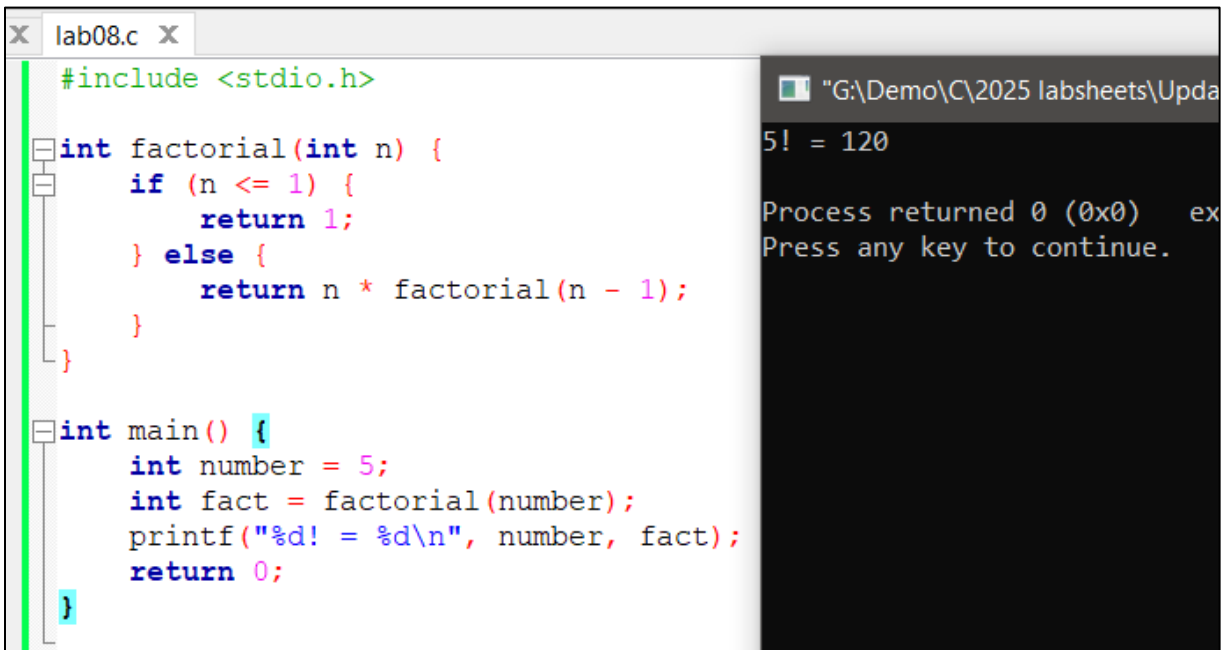
##### 1. Recursive Function:

- Create a recursive function to calculate the factorial of a positive integer.

```
int factorial(int n) {  
    if (n <= 1) {  
        return 1;  
    } else {  
        return n * factorial(n - 1);  
    }  
}
```

##### 2. Function Call:

- Call the factorial function with a positive integer and display the result.



The screenshot shows a code editor window titled 'lab08.c' on the left and a terminal window on the right. The code in the editor defines a recursive factorial function and a main function that calls it with the value 5. The terminal output shows the result of the calculation: '5! = 120'.

```
#include <stdio.h>  
  
int factorial(int n) {  
    if (n <= 1) {  
        return 1;  
    } else {  
        return n * factorial(n - 1);  
    }  
}  
  
int main() {  
    int number = 5;  
    int fact = factorial(number);  
    printf("%d! = %d\n", number, fact);  
    return 0;  
}
```

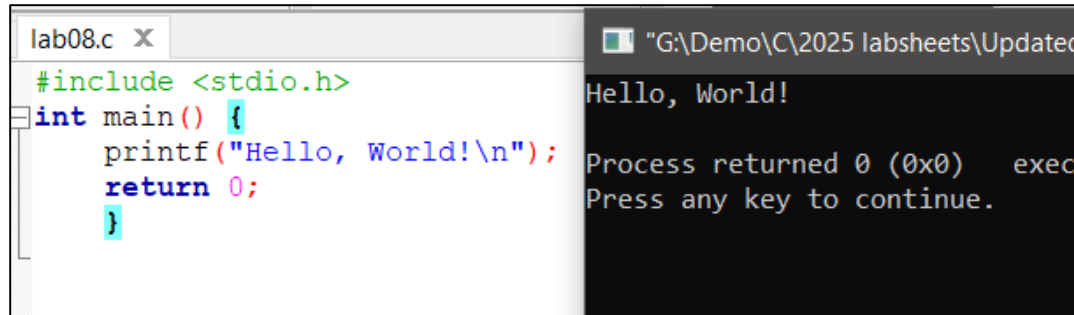
"G:\Demo\C\2025 labsheets\Upda  
5! = 120  
Process returned 0 (0x0) ex  
Press any key to continue.

## Practical 4: Using Standard Library Functions

### Steps

#### 1. Using printf:

- Use the printf function from the standard library to display a message.



The screenshot shows a code editor window titled 'lab08.c' with the following C code:

```
#include <stdio.h>

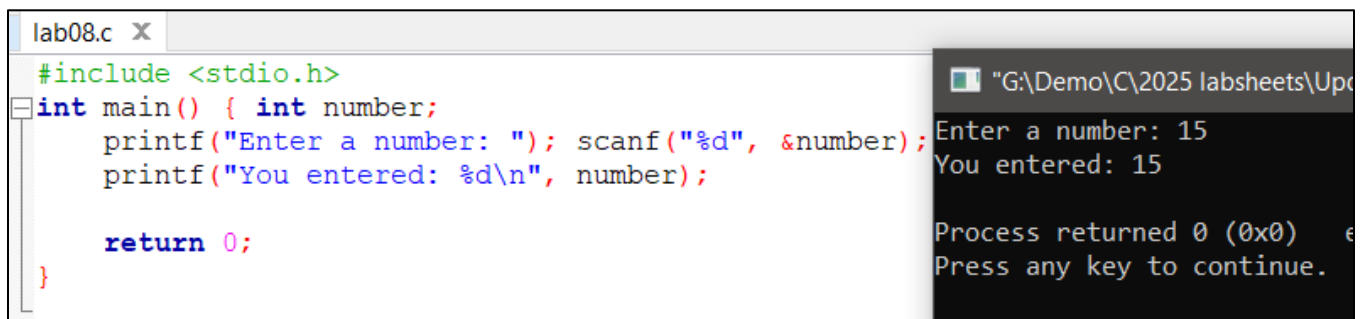
int main() {
    printf("Hello, World!\n");
    return 0;
}
```

To the right of the code editor is a terminal window showing the output of the program:

```
"G:\Demo\C\2025 labsheets\Updated
Hello, World!
Process returned 0 (0x0)   exec
Press any key to continue.
```

#### 2. Using scanf:

- Use the scanf function from the standard library to input a number and display it.



The screenshot shows a code editor window titled 'lab08.c' with the following C code:

```
#include <stdio.h>

int main() { int number;
    printf("Enter a number: "); scanf("%d", &number);
    printf("You entered: %d\n", number);

    return 0;
}
```

To the right of the code editor is a terminal window showing the output of the program:

```
"G:\Demo\C\2025 labsheets\Upd
Enter a number: 15
You entered: 15
Process returned 0 (0x0)   e
Press any key to continue.
```

## **Tasks**

1. **Write a function to check whether a given number is prime.**
  - Function prototype: `int isPrime(int num);`
  - The function should return 1 if the number is prime, otherwise return 0.
  - Implement `main()` to test the function.
2. **Write a recursive function to calculate the nth Fibonacci number.**
  - Function prototype: `int fibonacci(int n);`
  - The function should return the nth Fibonacci number using recursion.
  - Implement `main()` to display the Fibonacci sequence up to a given number.
3. **Write a function to find the greatest common divisor (GCD) of two numbers using recursion.**
  - Function prototype: `int gcd(int a, int b);`
  - The function should use the Euclidean algorithm for computing GCD.
  - Implement `main()` to test the function.

## **Discussion**

- Discuss the importance of function declaration and definition.
- Explain how return values are used in function calls.
- Explain how function arguments are passed.
- Discuss how the return value is used in calculations.
- Explain the concept of recursion.
- Discuss the importance of standard library functions.

## **Report Submission Guidelines**

- Submit the Report by 04/04/2025.
- Report Structure
  - Practical No
  - Date of Submission
  - Title
  - Objective of the practical.
  - Tasks
  - Discussion
  - Challenges
  - References