

# **COMP0204: Introduction to Programming for Robotics and AI**

## **Development and Design of Program**

Course lead: Dr Sophia Bano

Lecture by: Dr Yunda Yan

MEng Robotics and AI

UCL Computer Science

# Development and Design of Program

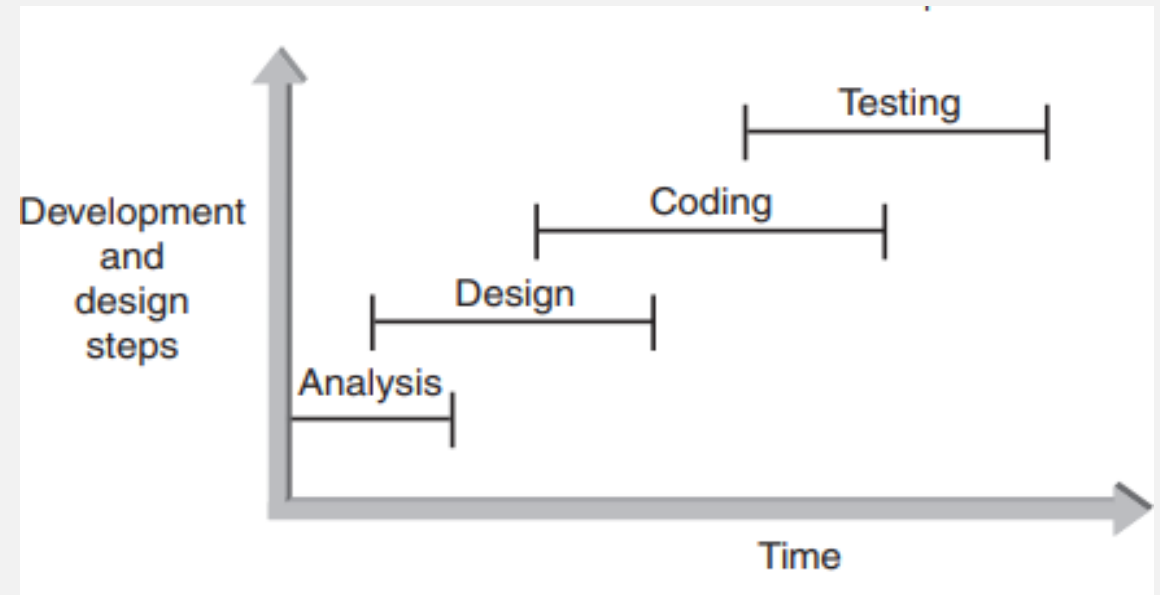
- Begins with a statement of a problem or a specific request for a program, which is referred to as a **program requirement**.
- Four steps :

**Step 1: Analyze the Problem**

**Step 2: Develop a Solution**  
(Use / Design an algorithm)

**Step 3: Code the Solution**  
(Write the Program)

**Step 4: Test and Correct the Program**



# 1. Analyze the Problem

- a. Determine and understand the output items the program must produce.
  - b. Determine the input items.
- 
- Together, these two items are referred to as the problem's **input/output (I/O)**.
  - Only after determining a problem's, I/O you can select specific steps for transforming inputs into outputs.

## 2. Develop a Solution

- In this step, you select the exact set of steps, called an “**algorithm**”, to be used to solve the problem.

### ALGORITHM

- Step-by-step sequence of instructions
- Describes how the data is to be processed to produce the desired outputs
- “What method will you use to solve this problem?”

### 3. Code the Solution (Write the Program)

- This step consists of writing a C program that corresponds to the solution developed in Step 2.

## 4. Test and Correct the Program

- The purpose of testing is to verify that a program works correctly and fulfills its requirements.
- Computer terminology, a program error is called a **bug**
- **Debugging** is the process of finding and resolving bugs within computer programs, software, or systems

# Coding an Algorithm




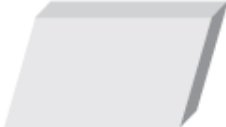
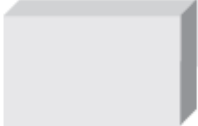


# Algorithm

An algorithm (step by step instructions) can be represented by:

1. Pseudocode
2. Formula
3. Flow Chart



# Flow Chart

Symbol	Name	Description
	Terminal	Indicates the beginning or end of a program
	Input/output	Indicates an input or output operation
	Process	Indicates computation or data manipulation
	Flow lines	Used to connect the other flowchart symbols and indicate the logic flow
	Decision	Indicates a program branch point

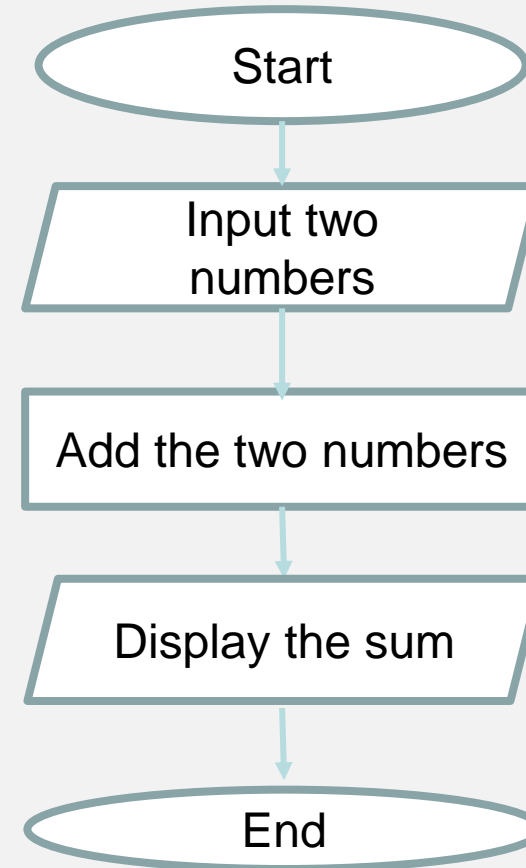
# Example 1

Give an algorithm to find the sum of two numbers entered by the user.

1. Pseudocode
2. Formula
3. Flow Chart

# Example 1

- **Pseudo Code:**
  - a) Input two numbers.
  - b) Add the two numbers.
  - c) Display the sum.
- **Formula:**  $\text{sum} = a + b$
- **Flow Chart:**



# Example 1

```
#include <stdio.h>

int main() {
    // Initialize variables
    double num1, num2, sum;

    // Input: Get the first number from the user
    printf("Enter the first number: ");
    scanf("%lf", &num1);

    // Input: Get the second number from the user
    printf("Enter the second number: ");
    scanf("%lf", &num2);

    // Calculate the sum of the two numbers
    sum = num1 + num2;

    // Output: Display the sum to the user
    printf("The sum of %.2lf and %.2lf is %.2lf\n", num1, num2, sum);

    return 0;
}
```

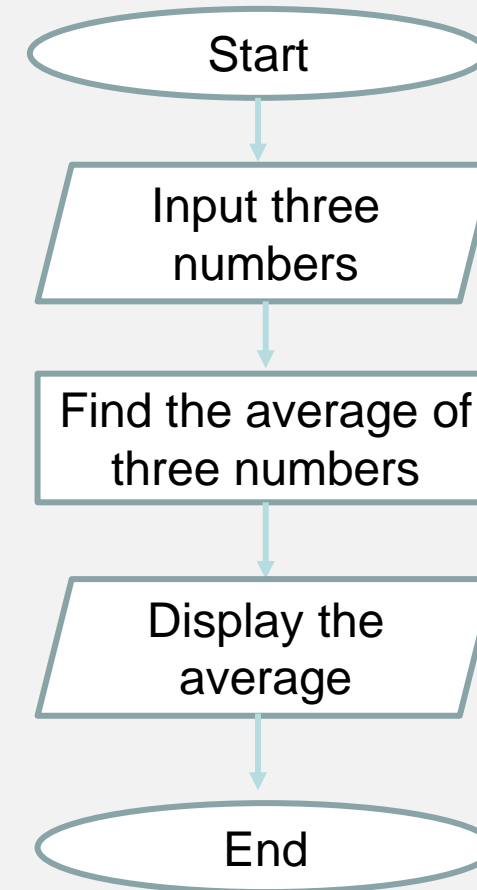
```
Enter the first number: 5.1
Enter the second number: 3.5
The sum of 5.10 and 3.50 is 8.60
```

## EXAMPLE 2

- Give an algorithm for determining the average of three numbers.
1. Pseudocode
  2. Formula
  3. Flow Chart

## Example 2

- **Pseudo Code:**
  - a) Input three numbers.
  - b) Find the average of three numbers.
  - c) Display the average.
- **Formula:**  $\text{avg} = (a+b+c)/3$
- **Flow Chart:**



# Example 2

```
#include <stdio.h>

int main() {
    double num1, num2, num3, average;

    // Input: Get three numbers from the user
    printf("Enter the first number: ");
    scanf("%lf", &num1);

    printf("Enter the second number: ");
    scanf("%lf", &num2);

    printf("Enter the third number: ");
    scanf("%lf", &num3);

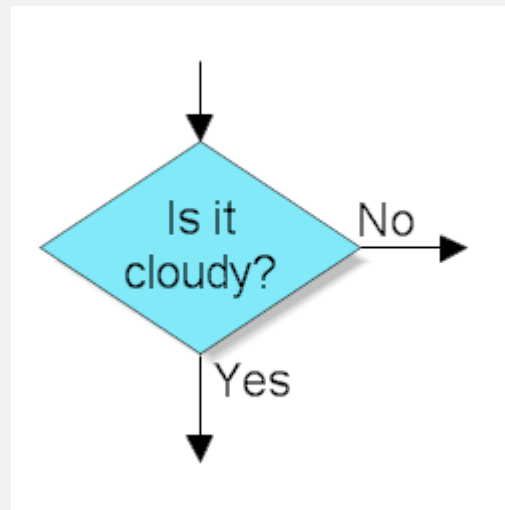
    // Calculate the average of the three numbers
    average = (num1 + num2 + num3) / 3.0;

    // Output: Display the average to the user
    printf("The average of %.2lf, %.2lf, and %.2lf is %.2lf\n", num1, num2, num3, average);

    return 0;
}
```

```
Enter the first number: 5
Enter the second number: 7.5
Enter the third number: 2
The average of 5.00, 7.50, and 2.00 is 4.83
```

# Decision block in the flow chart





# Exercise

- Design a program that takes a student's score as input and checks if the score is greater than or equal to 60. If the score is greater than or equal to 60, it should output "Pass," otherwise, it should output "Fail."
- Write a pseudocode and create a flowchart for this program.

# Designing Algorithms: A Methodology

- Read the problem, identifying the input and the output.
- What variables are needed?
- What computations are required to achieve the output?
- Usually, the first steps in your algorithm bring input values to the variables.
- Usually, the last steps display the output
- So, the middle steps will do the computation.
- If the process is to be repeated, add loops.