

# **Programming languages and Program Development**

## **BBA 510113: Computer and Information Technology**

**Md. Intiaz Ahmed  
Lecturer, CSE  
Daffodil Institute of IT**

# Programming Languages



- The computer system is simply a machine and hence it cannot perform any work; therefore, in order to make it functional different languages are developed, which are known as programming languages or simply computer languages.
- Over the last two decades, dozens of computer languages have been developed. Each of these languages comes with its own set of vocabulary and rules, better known as syntax. Furthermore, while writing the computer language, syntax has to be followed literally, as even a small mistake will result in an error and not generate the required output.

# Programming Languages



Following are the major categories of Programming Languages –

- Machine Language
- Assembly Language
- High Level Language
- Machine Language or Code

This is the language that is written for the computer hardware. Such language is effected directly by the central processing unit (CPU) of a computer system.

# Programming Languages



- Assembly Language

It is a language of an encoding of machine code that makes simpler and readable.

- High Level Language

The high level language is simple and easy to understand and it is similar to English language. For example, COBOL, FORTRAN, BASIC, C, C+, Python, etc.

High-level languages are very important, as they help in developing complex software and they have the following advantages –

# Programming Languages



- Unlike assembly language or machine language, users do not need to learn the high-level language in order to work with it.
- High-level languages are similar to natural languages, therefore, easy to learn and understand.
- High-level language is designed in such a way that it detects the errors immediately.
- High-level language is easy to maintain and it can be easily modified.
- High-level language makes development faster.
- High-level language is comparatively cheaper to develop.
- High-level language is easier to document.

# Programming Languages



Examples of some frequently used programming languages are-

- C
- JAVA
- JAVASCRIPT
- C#
- Python
- PHP
- C++
- SQL

# Program Development



- Program development is the process of conceiving, specifying, designing, programming, documenting, testing, and bug fixing involved in creating and maintaining applications, frameworks, or other software components.
- Program development is a process of writing and maintaining the source code, but in a broader sense, it includes all that is involved between the conception of the desired software through to the final manifestation of the software, sometimes in a planned and structured process.
- Therefore, program development may include research, new development, prototyping, modification, reuse, re-engineering, maintenance, or any other activities that result in software products.

# Developing a Program



Developing a program includes the below steps:

- Analyze the problem.
- Design the program.
- Code the program.
- Debug the program.
- Formalize the solution.
- Document and maintain the program.



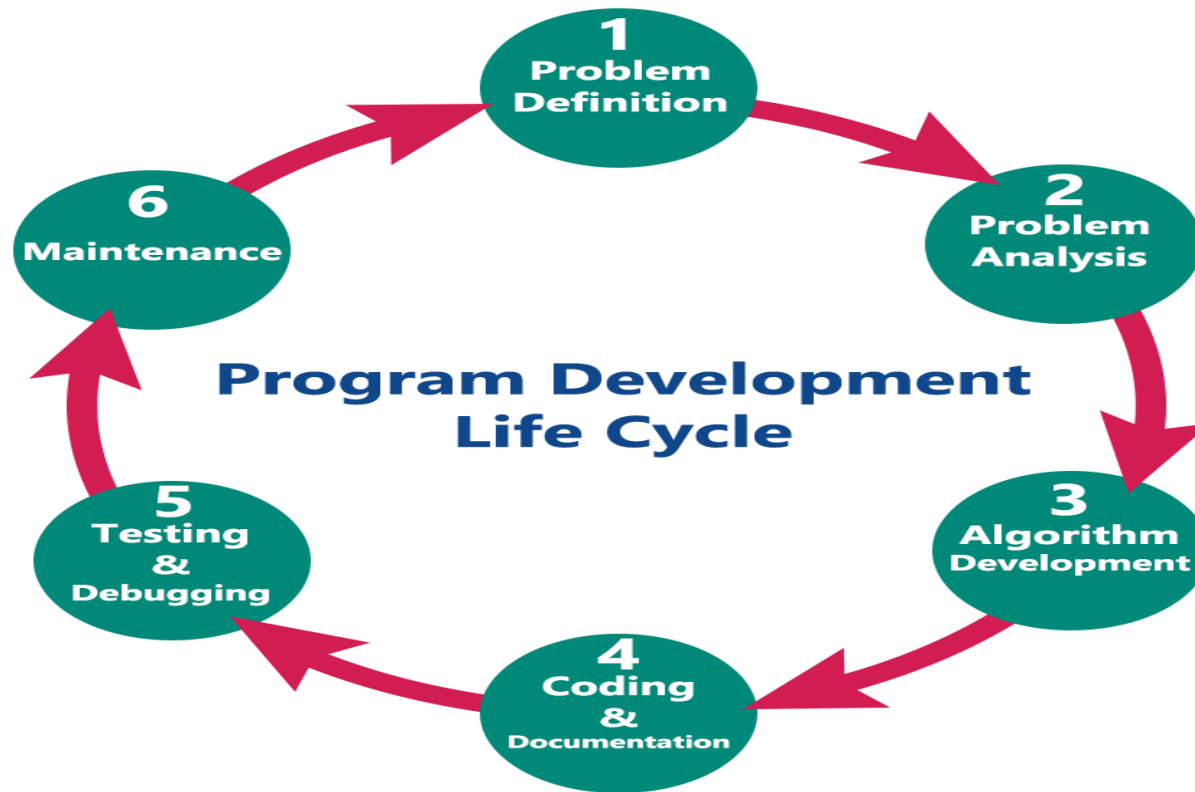
# Program Development Life Cycle(PDLC)



Generally, the program development life cycle contains 6 phases, they are as follows-

- Problem Definition
- Problem Analysis
- Algorithm Development
- Coding & Documentation
- Testing & Debugging
- Maintenance

# Program Development Life Cycle(PDLC)



# Program Development Life Cycle(PDLC)



- Problem Definition

In this phase, we define the problem statement and we decide the boundaries of the problem.

- Problem Analysis

In phase 2, we determine the requirements like variables, functions, etc. to solve the problem.

- Algorithm Development

During this phase, we develop a step by step procedure to solve the problem using the specification given in the previous phase.

# Program Development Life Cycle(PDLC)



- Coding & Documentation

This phase uses a programming language to write or implement the actual programming instructions for the steps defined in the previous phase.

- Testing & Debugging

During this phase, we check whether the code written in the previous step is solving the specified problem or not.

- Maintenance

During this phase, the program is actively used by the users. If any enhancements found in this phase, all the phases are to be repeated to make the enhancements.

# Algorithm



From programming point of view, an algorithm is a step-by-step procedure to resolve any problem. An algorithm is an effective method expressed as a finite set of well-defined instructions.

To understand these terms, consider a situation when someone asks you about how to go to a nearby KFC. What exactly do you do to tell him the way to go to KFC?

You will use Human Language to tell the way to go to KFC, something as follows –

**First go straight, after half kilometer, take left from the red light and then drive around one kilometer and you will find KFC at the right.**

# Algorithm



Here, you have used English Language to give several steps to be taken to reach KFC. If they are followed in the following sequence, then you will reach KFC –






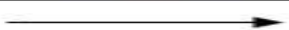


- 1. Go Straight**
- 2. Drive half kilometer**
- 3. Take left**
- 4. Drive around one kilometer**
- 5. Search for KFC at your right side**

# Flowcharts



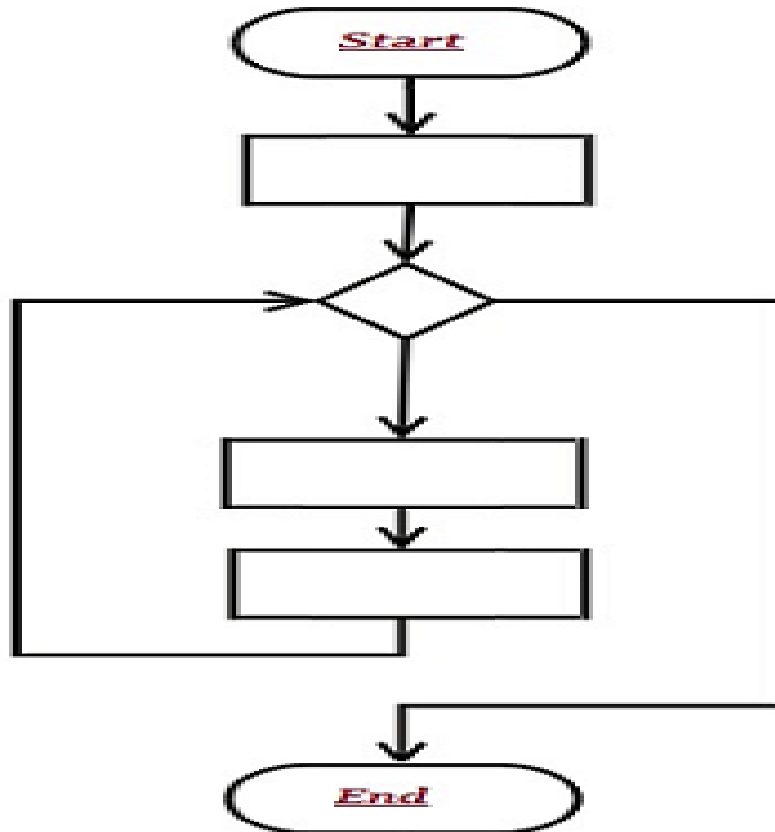
A flowchart is a blueprint that pictorially represents the algorithm and its steps. The steps of a flowchart do not have a specific size and shape rather it is designed in different shapes and sizes.

## Flowchart Symbols

Symbol	Meaning
	Start/Stop
	Process
	Input/Output
	Decision/Branching
	Connector
	Flow
	Manual Input
	Predefined Process

# Flowcharts

Sample flowcharts are given below-





# System Analysis and Design



Systems development is systematic process which includes phases such as planning, analysis, design, deployment, and maintenance. Here, in this tutorial, we will primarily focus on –

- **Systems analysis**
- **Systems design**

## ■ **Systems Analysis**

It is a process of collecting and interpreting facts, identifying the problems, and decomposition of a system into its components.

# System Analysis and Design



System analysis is conducted for the purpose of studying a system or its parts in order to identify its objectives. It is a problem solving technique that improves the system and ensures that all the components of the system work efficiently to accomplish their purpose.

Analysis specifies what the system should do.

## ■ Systems Design

It is a process of planning a new business system or replacing an existing system by defining its components or modules to satisfy the specific requirements. Before planning, you need to understand the old system thoroughly and determine how computers can best be used in order to operate efficiently.

# System Analysis and Design



System Design focuses on how to accomplish the objective of the system.

System Analysis and Design (SAD) mainly focuses on –

- Systems
- Processes
- Technology

# Software Development Life Cycle (SDLC)



SDLC is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process.

- SDLC is the acronym of Software Development Life Cycle.
- It is also called as Software Development Process.
- SDLC is a framework defining tasks performed at each step in the software development process.

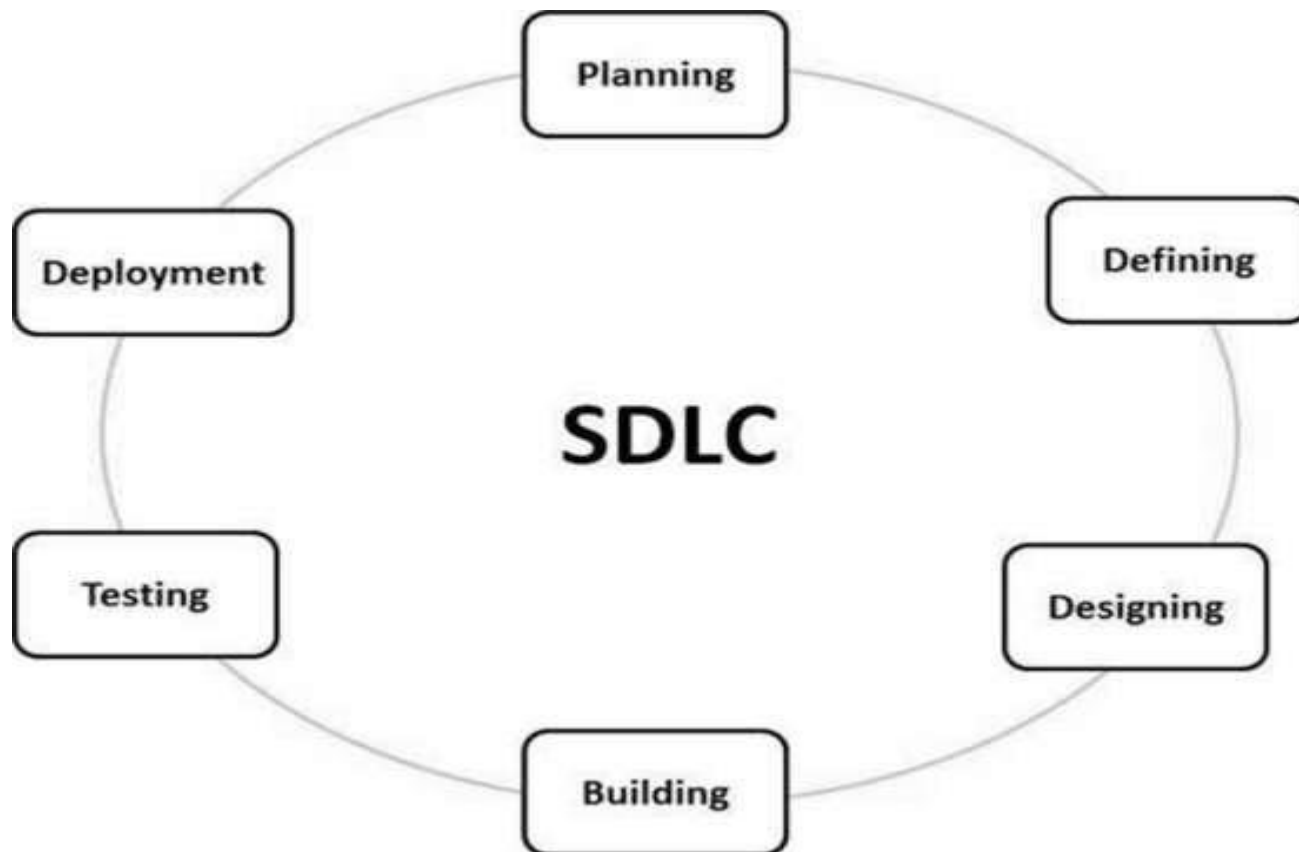
# Software Development Life Cycle (SDLC)



SDLC includes the following activities –

- Planning
- Defining
- Designing
- Building
- Testing
- Deployment

# Software Development Life Cycle (SDLC)



THANKS...

