

Department of Cybersecurity Technologies Engineering

PROGRAMMING ESSENTIALS

Lecture 2: Algorithms and Flowchart



Dr. Hayder Kareem Algabri

hayder_kareem_algabri@hilla-unc.edu.iq



2024-2025

Lecture 2: Algorithms and Flowchart

Contents

1. PROBLEM SOLVING	2
2. Algorithms:	3
Use of the Algorithms:	4
What is the need for algorithms?	4
Properties of Algorithm:	5
Qualities of a Good Algorithm.....	6
Algorithm 1: Add two numbers entered by the user.....	6
Algorithm 2: Find the largest number among three numbers	7
3. Flowchart In Programming	8
Symbols Used In Flowchart	8
General Rules for Flowcharting	9
Examples of flowcharts in programming	10

*Lecture 2: Algorithms and Flowchart***1. PROBLEM SOLVING**

Can you think of a day in your life which goes without problem solving? Answer to this question is of course, No. In our life we are bound to solve problems. In our day-to-day activity such as purchasing something from a general store and making payments, depositing fee in school, or withdrawing money from bank account. All these activities involve some kind of problem solving. It can be said that whatever activity a human being or machine do for achieving a specified objective comes under problem solving. To make it clearer, let us see some other examples.

Example1: If you are watching a news channel on your TV and you want to change it to a sports channel, you need to do something i.e. move to that channel by pressing that channel number on your remote. This is a kind of problem solving.

Example 2: One Monday morning, a student is ready to go to school but yet he/she has not picked up those books and copies which are required as per timetable. So here picking up books and copies as per timetable is a kind of problem solving.

Example 3: If someone asks to you, what is time now? So seeing time in your watch and telling him is also a kind of problem solving. Example 4: Some students in a class plan to go on picnic and decide to share the expenses among them. So calculating total expenses and the amount an individual have to give for picnic is also a kind of problem solving.

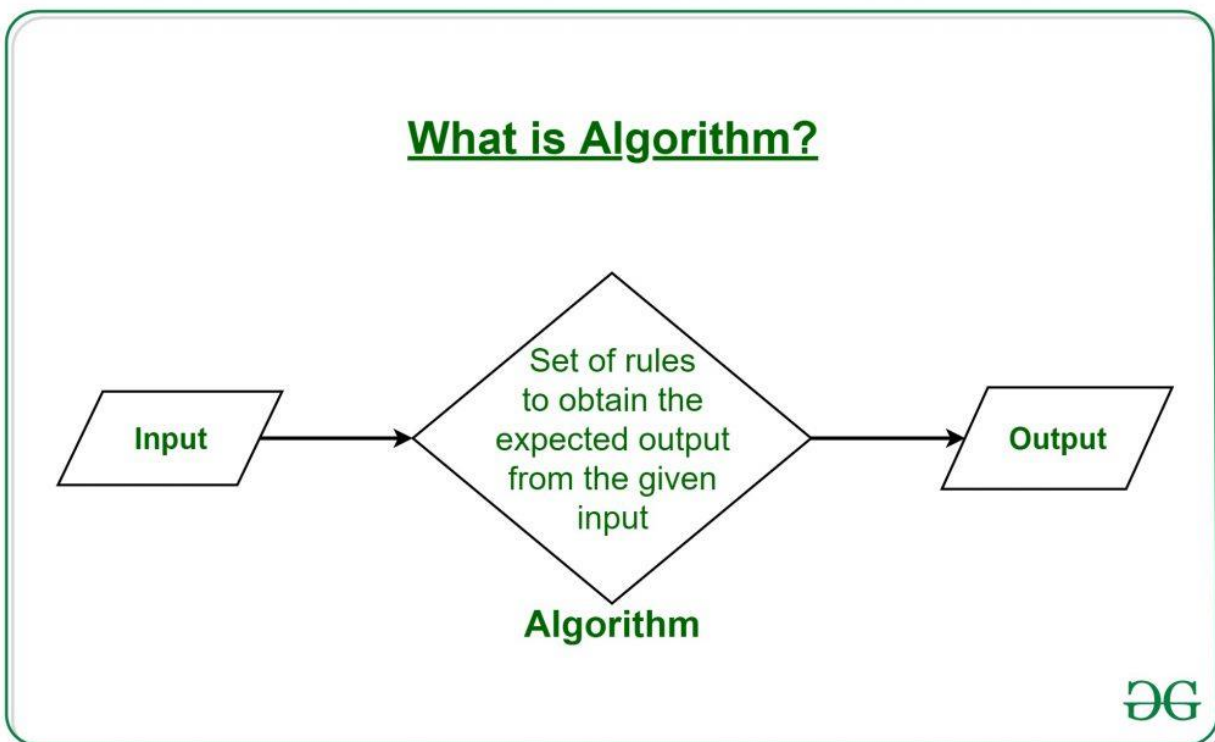
Lecture 2: Algorithms and Flowchart

2. Algorithms:

Algorithm can be defined as: “A sequence of activities to be processed for getting desired output from a given input.”

Another defines an algorithm as: “A formula or set of steps for solving a particular problem. To be an algorithm, a set of rules must be unambiguous and have a clear stopping point”. There may be more than one way to solve a problem, so there may be more than one algorithm for a problem.

In mathematics, computer science, and related subjects, an algorithm is a finite sequence of steps expressed for solving a problem. An algorithm can be defined as “a process that performs some sequence of operations in order to solve a given problem”. Algorithms are used for calculation, data processing, and many other fields.



*Lecture 2: Algorithms and Flowchart***Use of the Algorithms:**

Algorithms play a crucial role in various fields and have many applications. Some of the key areas where algorithms are used include:

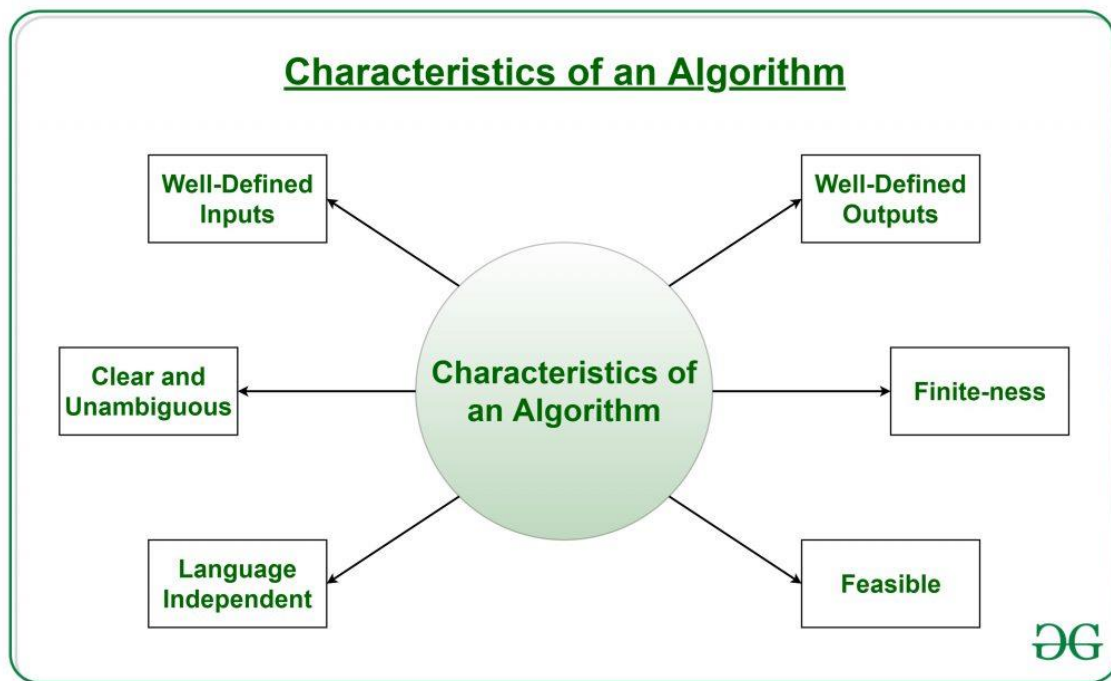
1. **Computer Science:** Algorithms form the basis of computer programming and are used to solve problems ranging from simple sorting and searching to complex tasks such as artificial intelligence and machine learning.
2. **Mathematics:** Algorithms are used to solve mathematical problems, such as finding the optimal solution to a system of linear equations or finding the shortest path in a graph.
3. **Operations Research:** Algorithms are used to optimize and make decisions in fields such as transportation, logistics, and resource allocation.
4. **Artificial Intelligence:** Algorithms are the foundation of artificial intelligence and machine learning, and are used to develop intelligent systems that can perform tasks such as image recognition, natural language processing, and decision-making.
5. **Data Science:** Algorithms are used to analyze, process, and extract insights from large amounts of data in fields such as marketing, finance, and healthcare.

What is the need for algorithms?

1. Algorithms are necessary for solving complex problems efficiently and effectively.
2. They help to automate processes and make them more reliable, faster, and easier to perform.

Lecture 2: Algorithms and Flowchart

3. Algorithms also enable computers to perform tasks that would be difficult or impossible for humans to do manually.
4. They are used in various fields such as mathematics, computer science, engineering, finance, and many others to optimize processes, analyze data, make predictions, and provide solutions to problems.

**Properties of Algorithm:**

- It should terminate after a finite time.
- It should produce at least one output.
- It should take zero or more input.
- It should be deterministic means giving the same output for the same input case.
- Every step in the algorithm must be effective i.e. every step should do some work.

Lecture 2: Algorithms and Flowchart

Qualities of a Good Algorithm

- Input and output should be defined precisely.
- Each step in the algorithm should be clear and unambiguous.
- Algorithms should be most effective among many different ways to solve a problem.
- An algorithm shouldn't include computer code. Instead, the algorithm should be written in such a way that it can be used in different programming languages.

Algorithm 1: Add two numbers entered by the user

Step 1: Start

Step 2: Declare variables num1, num2 and sum.

Step 3: Read values num1 and num2.

Step 4: Add num1 and num2 and assign the result to sum.

sum ← num1 + num2

Step 5: Display sum

Step 6: Stop

Lecture 2: Algorithms and Flowchart

Algorithm 2: Find the largest number among three numbers

Step 1: Start

Step 2: Declare variables a,b and c.

Step 3: Read variables a,b and c.

Step 4: If $a > b$

If $a > c$

Display a is the largest number.

Else

Display c is the largest number.

Else

If $b > c$

Display b is the largest number.

Else

Display c is the greatest number.

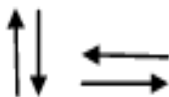
Step 5: Stop

Lecture 2: Algorithms and Flowchart

3. Flowchart In Programming

A flowchart is a diagrammatic representation of an algorithm. A flowchart can be helpful for both writing programs and explaining the program to others.

Symbols Used In Flowchart

Symbol	Name	Function
	Process	Indicates any type of internal operation inside the Processor or Memory
	input/output	Used for any Input / Output (I/O) operation. Indicates that the computer is to obtain data or output results
	Decision	Used to ask a question that can be answered in a binary format (Yes/No, True/False)
	Connector	Allows the flowchart to be drawn without intersecting lines or without a reverse flow.
	Predefined Process	Used to invoke a subroutine or an Interrupt program.
	Terminal	Indicates the starting or ending of the program, process, or interrupt program
	Flow Lines	Shows direction of flow.

Lecture 2: Algorithms and Flowchart

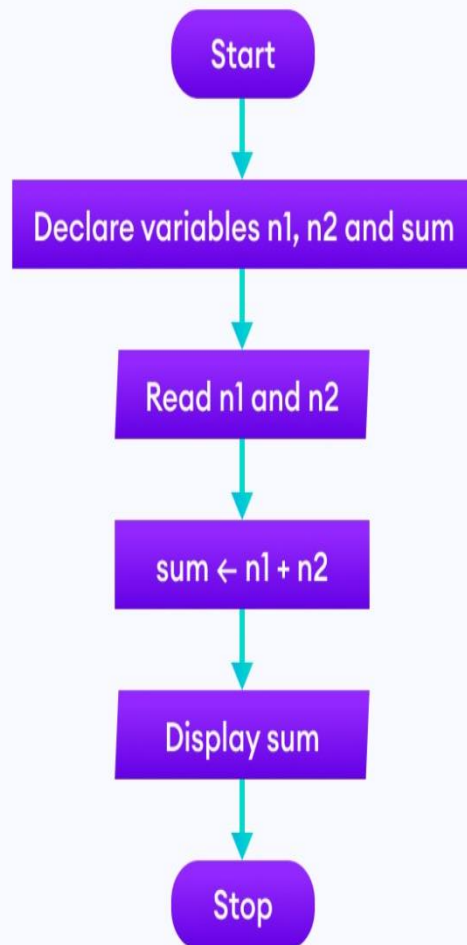
General Rules for Flowcharting

1. All boxes of the flowchart are connected with Arrows. (Not lines)
2. Flowchart symbols have an entry point on the top of the symbol with no other entry points. The exit point for all flowchart symbols is on the bottom except for the Decision symbol.
3. The Decision symbol has two exit points; these can be on the sides or the bottom and one side.
4. Generally a flowchart will flow from top to bottom. However, an upward flow can be shown as long as it does not exceed 3 symbols.
5. Connectors are used to connect breaks in the flowchart. Examples are:
 - From one page to another page.
 - From the bottom of the page to the top of the same page.
 - An upward flow of more than 3 symbols
6. Subroutines and Interrupt programs have their own and independent flowcharts.
7. All flow charts start with a Terminal or Predefined Process (for interrupt programs or subroutines) symbol.
8. All flowcharts end with a terminal or a contentious loop.

Lecture 2: Algorithms and Flowchart

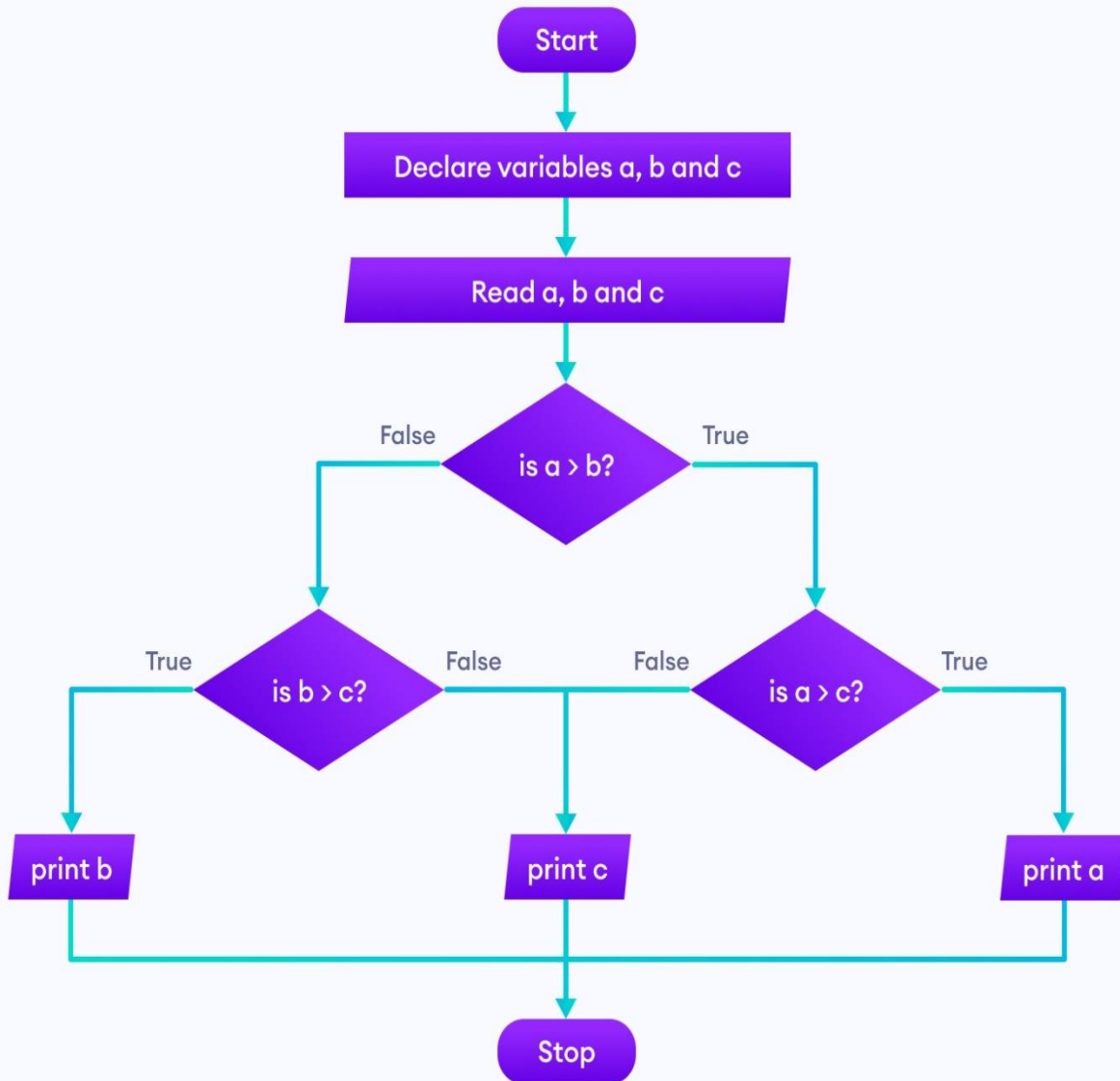
Examples of flowcharts in programming

1. Add two numbers entered by the user.



Lecture 2: Algorithms and Flowchart

2. Find the largest among three different numbers entered by the user.

**Advantages of using Flowcharts**

As we discussed flow chart is used for representing algorithm in pictorial form. This pictorial representation of a solution/system is having many advantages. These advantages are as follows:

Lecture 2: Algorithms and Flowchart

- 1) Communication: A Flowchart can be used as a better way of communication of the logic of a system and steps involve in the solution, to all concerned particularly to the client of system.
- 2) Effective analysis: A flowchart of a problem can be used for effective analysis of the problem.
- 3) Documentation of Program/System: Program flowcharts are a vital part of a good program documentation. Program document is used for various purposes like knowing the components in the program, complexity of the program etc.
- 4) Efficient Program Maintenance: Once a program is developed and becomes operational it needs time to time maintenance. With help of flowchart maintenance become easier.
- 5) Coding of the Program: Any design of solution of a problem is finally converted into computer program. Writing code referring the flowchart of the solution become easy.