Problem solving Techniques(Part-2)

(Algorithms, Flowchart and Pseudocodes)

Algorithms

- An algorithm is a step-by-step procedure or formula for solving a problem. It's a precise set of instructions that take inputs (data) and produce outputs (results) in a finite number of steps. Key characteristics of algorithms include:
- **Clear and Unambiguous:** Each step must be well-defined and understandable.
- **Finite:** The algorithm must terminate after a finite number of steps.
- **Effective:** It should solve the problem correctly for all possible inputs.

Example of algorithm to find maximum number:

- Algorithm to find the maximum number in a list of integers:
- 1. Start with the first number in the list as the current maximum.
- 2. Iterate through the list: - If the current number is greater than the current maximum, update the current maximum.
- 3. After iterating through the list, the current maximum is the maximum number.
- 4. End.

Flowchart:

- A flowchart is a visual representation of an algorithm using various symbols and arrows to illustrate the flow of control. It provides a way to understand the steps involved in a process or algorithm visually, making it easier to identify errors and improvements.
- **Symbols commonly used in flowcharts:**
- **Start/End:** Indicates the beginning or end of a process.
- **Process:** Represents a specific action or operation.
- **Decision:** Represents a conditional operation (yes/no or true/false).
- **Input/Output:** Shows where data enters or leaves the process.
- **Flow lines:** Arrows showing the direction of the process flow.

Pseudocode:

- Pseudocode is an informal high-level description of the operating principle of a computer program or other algorithm. It uses structured English to describe the logic of the algorithm without focusing on the specifics of programming syntax. Pseudocode helps programmers and designers to plan algorithms and understand the flow of the program.
- Pseudocode is used to:
- Plan and design a program or algorithm
- Communicate ideas and logic to others
- Test and refine the algorithm before writing code
- Document the logic and steps involved
- It is not meant to be compiled or executed directly, but rather serves as a blueprint for writing the actual code in a programming language.

Example of Pseudocode:

- Example of pseudocode:
- IF (x > 0)
- THEN PRINT "x is positive"
- ELSE
- PRINT "x is not positive“
- ENDIF

Thank you