In-Class Exercise: Demonstrating the Three Programming Structures Using Flowgorithm

Objective: Students will use **Flowgorithm** to create and run simple flowcharts that demonstrate the three fundamental programming structures:

- 1. Sequence
- 2. Selection (Decision-Making)
- 3. Loop (Iteration)

Instructions

- 1. **Download and Install Flowgorithm** (if not already installed).
- 2. Follow the given tasks to create flowcharts demonstrating each programming structure.
- 3. Run and test each flowchart.
- 4. Save and submit your Flowgorithm files.

Flowgorithm Flowchart Symbols & Their Purpose

Symbol	Name	Purpose
Oval	Terminator (Start/End)	Marks the beginning and end of a flowchart.
Parallelogram	Input/Output	Used to take user input or display output.
Rectangle	Process	Represents calculations, assignments, and processing steps.
○ Diamond	Decision	Used for selection structures (IF, IF-ELSE) to make decisions.
S Arrow	Flowline	Connects flowchart components and determines the flow of execution.

Flowgorithm Programming Structures

Sequence (Linear Execution)

A **sequence** structure executes instructions in order, one after another.

- Use **Input** to get values.
- Use **Process** for calculations or variable assignments.
- Use **Output** to display the result.

Example:

- 1. Input Name
- 2. Display "Hello, " + Name

Selection (Decision-Making)

A **selection** structure is used for conditional branching in Flowgorithm (making decisions using IF/ELSE). It uses a **Decision (Diamond)** symbol in the flowchart to branch into different paths based on a condition.

Simple IF-ELSE

• In a simple selection, the flowchart splits into two paths (true/false) based on one condition.

Example:

```
1. Input Number
2. IF Number MOD 2 == 0 THEN
    Output "Even"
3. ELSE
    Output "Odd"
```

Nested IF-ELSE

• Nested selection structures are used when multiple conditions need to be checked in sequence.

Example:

```
1. Input Score
   IF Score >= 90 THEN
        Output "Grade A"
   ELSE IF Score >= 80 THEN
        Output "Grade B"
   ELSE IF Score >= 70 THEN
        Output "Grade C"
   ELSE
   Output "Fail"
```

Loop (Iteration)

A **loop** structure allows repeating a set of instructions multiple times. Always ensure there is an exit condition to prevent infinite loops.

WHILE Loop

• A WHILE loop executes as long as a specified condition remains true.

Example (print numbers 1 to 5):

```
Set Counter = 1
WHILE Counter <= 5
Output Counter`
Counter = Counter + 1
```

• A **FOR** loop executes a *fixed number of times*.

Example (print numbers 1 to 5):

```
FOR i FROM 1 TO 5
Output i
```

DO-WHILE Loop (Post-Test Loop)

• A **DO-WHILE** loop executes the loop body **at least once**, then continues *while* a condition remains true (check at the end).

Example (ask for input until the user enters a positive number):

```
DO
Input Number
WHILE Number <= 0
```

Nested Loops (Loop Inside a Loop)

A **nested loop** is a loop inside another loop, used for grid-like repetition or when one loop controls repetition of another.

Example (Multiplication Table with nested FOR loops):

```
FOR i FROM 1 TO 5

FOR j FROM 1 to 5

Output i * j
```

Common Flowgorithm Functions

Function Usage Example

```
Input Input Name
Output Output "Hello, " + Name

If-Else IF Age >= 18 THEN Output "Adult" ELSE Output "Minor"

While WHILE i <= 5 DO Output i
    (with loop body i = i + 1)

For FOR i FROM 1 TO 5 DO Output i

Do-While DO Input Number WHILE Number <= 0</pre>
```

Tasks

Task 1: Sequence Structure (Linear Execution)

• **Problem Statement:** Create a flowchart that asks the user for their name, age, and favorite color. Then, display a greeting message including their details.

• Flowchart Components:

- Input: Ask for name, age, and color
- Process: Concatenate the response into a message
- Output: Display the message

• Expected Output Example:

```
Enter your name: Alice
Enter your age: 20
Enter your favorite color: Blue
Output: Hello Alice! You are 20 years old and your favorite color is Blue.
```

Task 2: Selection Structure (Decision-Making using IF/ELSE)

• **Problem Statement:** Create a flowchart that asks the user for a number. If the number is **even**, display "The number is even"; otherwise, display "The number is odd."

• Flowchart Components:

- o Input: Ask the user for a number
- Process: Use an **if-else** condition to check if the number is even (i.e., number mod 2 = 0)
- Output: Display "Even" or "Odd"

• Expected Output Example:

```
Enter a number: 7
Output: The number is odd.

Enter a number: 8
Output: The number is even.
```

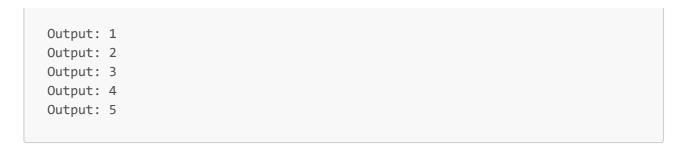
Task 3: Loop Structure (Using a WHILE Loop or FOR Loop)

Problem Statement: Create a flowchart that prints numbers from 1 to 5 using a loop.

• Flowchart Components:

- Process: Initialize a counter variable to 1
- o Loop: Use a while loop or for loop to repeat the process until the counter reaches 5
- o Output: Print numbers from 1 to 5

• Expected Output Example:



Task 4: Loop Structure (User-Defined Range)

• Modify **Task 3** to allow the user to input a number **N**, then print numbers from **1 to N** using a loop.