



How to write Pseudocode and draw Flowchart

Behram Khan





Key Pseudocode Statements

1. SEQUENCE

Represents tasks that execute one after another.

2. WHILE Loop

Repeats a set of instructions while a condition is true.

3. FOR Loop

Iterates over a set range or collection.

4. IF-THEN-ELSE

Executes actions based on a condition.



Key Pseudocode Statements

5. CALL

Invokes a class or function.



Key Pseudocode Statements

SEQUENCE

Input: READ, OBTAIN, GET

Output: PRINT, DISPLAY, SHOW

Compute: COMPUTE,
CALCULATE, DETERMINE

Initialize: SET, INIT

Add: INCREMENT, BUMP

Sub: DECREMENT

FOR

FOR iteration bounds

sequence

ENDFOR

WHILE

WHILE condition

sequence

ENDWHILE

IF-THEN-ELSE

IF condition THEN

sequence 1

ELSE

sequence 2

ENDIF



Common Pseudocode Commands

- Print – Display output.
- Calculate – Perform calculations.
- Read/Input – Take user input.
- Add/Subtract – Perform arithmetic operations.
- Display – Show results.



Guidelines for Writing Pseudocode

1. Capitalize initial keywords.
2. Use one statement per line.
3. Indent to show structure and hierarchy.
4. End multi-line structures with appropriate END keywords (ENDIF, ENDWHILE, etc.).
5. Avoid specific programming language syntax; use plain language.
6. Aim for simplicity, conciseness, and readability.



Example of IF-THEN-ELSE

The IF-THEN-ELSE structure allows branching based on conditions with two possible outcomes.

```
IF user's age > 21 THEN
    Display "Welcome"
ELSE
    Display "Access Denied"
ENDIF
```



Example of FOR Loop

The FOR loop iterates over a collection or range.

```
FOR each day in month  
    Calculate daily active users  
ENDFOR
```




Example of WHILE Loop

The WHILE loop repeats actions while a condition remains true.

```
WHILE active users < 100  
    Send reminder email  
ENDWHILE
```

Example of pseudocode for an algorithm that generates paychecks:

```
WHILE total employees is greater than 50
```

```
    Auto-generate paychecks
```

```
        Display employee name
```

```
        Display hours worked
```

```
        Display paid time off
```

```
    IF salaried employee THEN
```

```
        Display yearly salary divided by pay periods in a year
```

```
    ELSE
```

```
        Display hourly rate
```

```
    ENDIF
```

```
IF hourly employee THEN
```

```
    Multiply hourly rate by hours worked to calculate pay
```

```
ENDIF
```

```
Subtract withholdings from calculated pay
```

```
IF employee requested paper check THEN
```

```
    Print paycheck
```

Example of
pseudocode for an
algorithm that
generates paychecks:

```
IF employee last name starts with A-H THEN
    Print paycheck on Monday
ELSE IF employee last name starts with I-P THEN
    Print paycheck on Wednesday
ELSE IF employee last name starts with Q-Z THEN
    Print paycheck on Friday
ENDIF

ELSE
    Initiate direct deposit
ENDIF

ENDWHILE

WHILE not all employees have confirmed receipt of paychecks
    Send reminder email
ENDWHILE
```



Explanation of Each Section

1.Outer WHILE Loop for Paycheck Generation:

- This loop runs while the number of total employees is greater than 50. Inside this loop, paychecks are generated, and each employee's pay information is displayed.

2.Display Employee Information:

- Displays details like the employee's name, hours worked, and paid time off.

3.Conditional Check for Salaried or Hourly Employee:

- If the employee is salaried, it calculates their pay as yearly salary divided by the number of pay periods in a year.
- If not salaried, the hourly rate is displayed.



Explanation of Each Section

4. **Hourly Employee Pay Calculation:**

- If the employee is hourly, it calculates pay by multiplying the hourly rate by hours worked.

5. **Withholdings Subtraction:**

- Withholdings (like taxes or other deductions) are subtracted from the gross pay.

6. **Paper Check Request and Printing:**

- If an employee has requested a paper check, the paycheck is printed.
- Based on the employee's last name, the check is printed on a designated day:
 - Last names A-H: Monday
 - Last names I-P: Wednesday
 - Last names Q-Z: Friday



Explanation of Each Section

7. **Direct Deposit Option:**

- If the employee did not request a paper check, direct deposit is initiated.

8. **Final WHILE Loop for Confirmation:**

- A loop runs to send reminder emails until all employees have confirmed receipt of their paychecks.



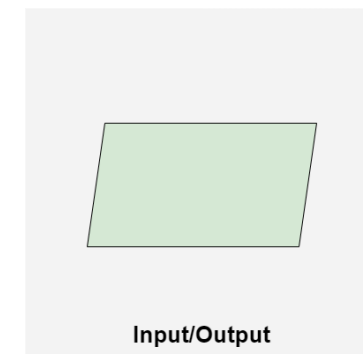
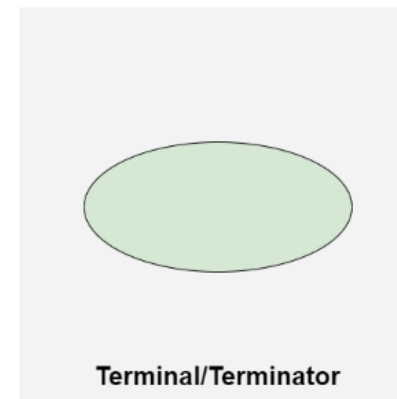
Flowchart: Symbols used in Flowchart Designs

1. Terminal/Terminator

The oval symbol indicates Start, Stop a program. Terminal is the first and last symbols in the flowchart.

2. Input/Output

A parallelogram denotes any function of input/output type. Program instructions that take input from input devices and display output on output devices are indicated with parallelogram in a flowchart.





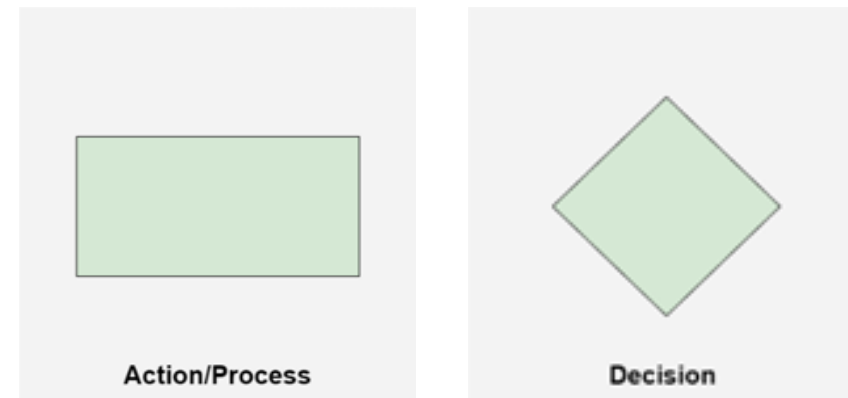
Flowchart: Symbols used in Flowchart Designs

3. Action/Process

A box represents arithmetic instructions, specific action or operation that occurs as a part of the process. All arithmetic processes such as adding, subtracting, multiplication and division are indicated by action/process symbol.

4. Decision

Diamond symbol represents a decision point. Decision based operations such as yes/no question or true/false are indicated by diamond in flowchart.





Flowchart: Symbols used in Flowchart Designs

5. Flow lines

Flow lines indicate the exact sequence in which instructions are executed. Arrows represent the direction of flow of control and relationship among different symbols of flowchart.



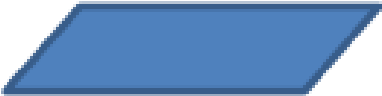
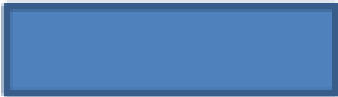



Flow Arrow



Flowchart: Symbols used in Flowchart Designs

Symbols Used In Flowchart

Symbol	Purpose	Description
	Flow line	Indicates the flow of logic by connecting symbols.
	Terminal (Stop/Start)	Represents the start and the end of a flowchart.
	Input/Output	Used for input and output operation.
	Processing	Used for arithmetic operations and data-manipulations.
	Decision	Used for decision making between two or more alternatives.



Rules For Creating a Flowchart

A flowchart is a graphical representation of an algorithm. It should follow some rules while creating a flowchart

Rule 1: Flowchart opening statement must be 'start' keyword.

Rule 2: Flowchart ending statement must be 'end' keyword.

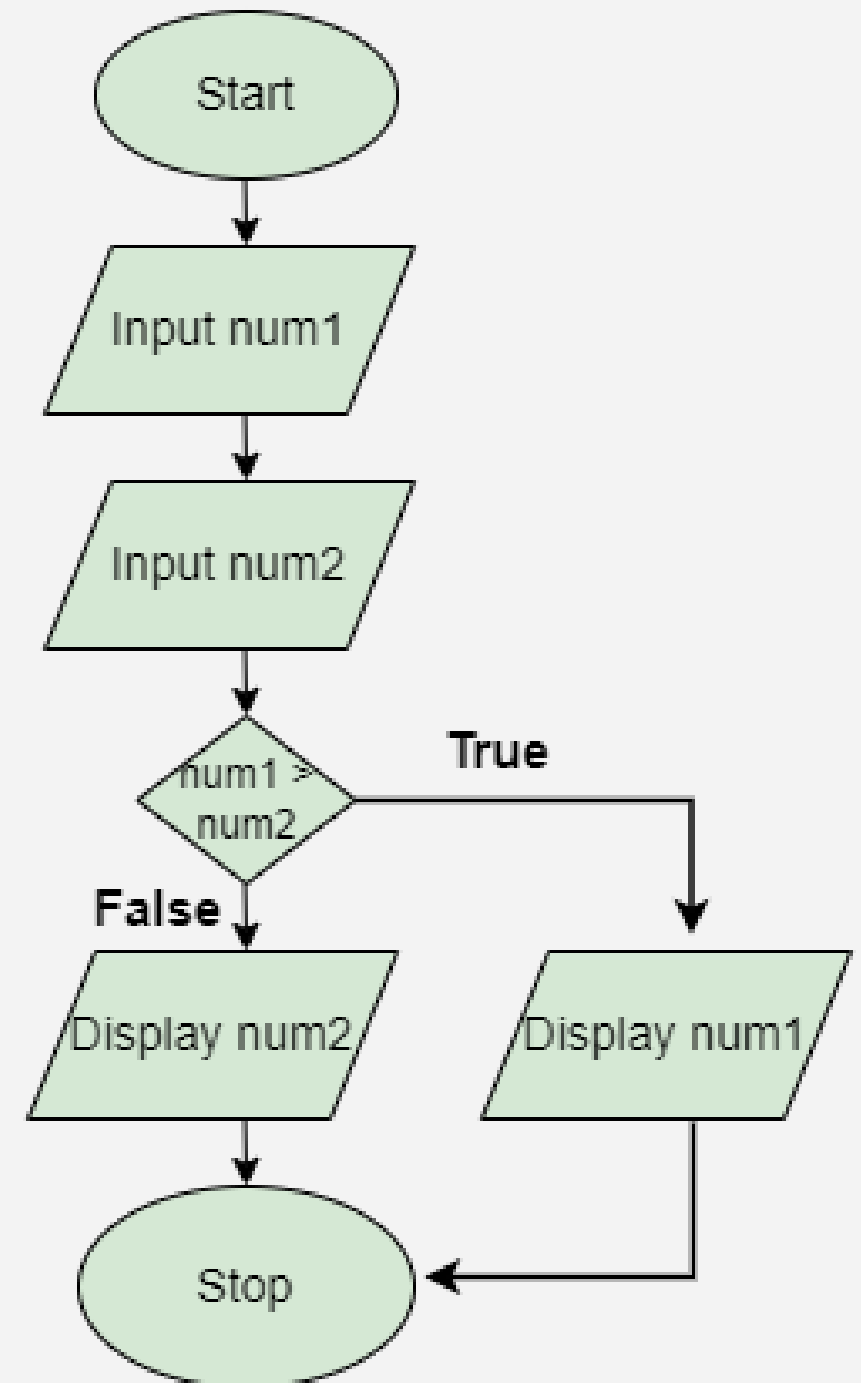
Rule 3: All symbols in the flowchart must be connected with an arrow line.

Rule 4: Each decision point should have two or more distinct outcomes.

Rule 5: Flow should generally move from top to bottom or left to right.

Example of a Flowchart

Draw a flowchart to input two numbers from the user and display the largest of two numbers.





Example of a Flowchart

- **Start:** The process begins with the Start symbol, indicating the start of the program.
- **Input num1:** The first number, represented as num1, is entered.
- **Input num2:** The second number, represented as num2, is entered.

Decision (num1 > num2): A decision point checks if num1 is greater than num2.

- If True, the process moves to the next step where num1 will be displayed.
- If False, the process moves to display num2.
- **Stop:** The process ends with the Stop symbol, signaling the conclusion of the program.