



**THE ONLY LIMIT TO WHAT YOU'LL
ACHIEVE IS WHAT YOU
CHOOSE TO LEARN**

• WORKSHOPS • INTERNSHIPS • HACKATHONS • PLACEMENT ASSISTANCE

Algonex INDUSTRY-STANDARD TRAININGS

+91 99597 89424 algonexacademy@gmail.com Bengaluru

Topics to Concentrate on



Quantitative:

- Speed Time and Distance
- Work and Time
- Boats and Streams
- Profit and Loss
- Ratios and Proportion
- Numbers and Fractions
- Area and Volume
- Probability
- Combinatorics (Less Priority)

Logical Reasoning:

- Sequence and Series
- Blood Relations
- Directions-Based/Positions Based
(i.e You move north, then east, etc)
- Reasoning Analogies
- Coding Decoding
- Clocks and Time

Verbal Aptitude/ English:

- Synonyms and Antonyms
- Fill in the blanks (Verbs/Adjectives/Pronoun etc)
- Sentence Formation/Passage Rearrangement
- Find the error
- Subject Verb Agreement
- (Grammar-Based Questions)

Pseudocode is language-independent. You don't worry about exact syntax, only logic

1. COMMON KEYWORDS:

- **BEGIN, END**
- **IF, ELSE, ELSE IF**
- **WHILE, FOR, DO**
- **PRINT, READ**
- **RETURN**

Example:

```
BEGIN  
  READ N  
  SUM ← 0  
  FOR i ← 1 TO N DO  
    SUM ← SUM + i  
  END FOR  
  PRINT SUM  
END
```

This prints the sum of first N numbers

2. Important Topics Infosys Focuses On



Conditional Logic

Nested IF, ELSE, SWITCH

Example: eligibility check (age, marks)

Loops

FOR, WHILE, DO-WHILE

Pattern printing, factorial, reverse of a number

Arrays

Finding largest/smallest

Searching & counting elements

Frequency of numbers

Strings

Palindrome check

Counting vowels/consonants

Reversing

Functions & Recursion

Factorial, Fibonacci

Binary Search

Math/Logic-based

Prime check

GCD/LCM

Armstrong numbers

Tracing & Debugging

Identify output

Find errors in pseudocode

Predict number of iterations

3. Infosys Pseudocode Question Patterns



Type 1: Predict the output

#16

```
BEGIN
  X ← 1
  WHILE X < 10 DO
    X ← X * 2
  END WHILE
  PRINT X
END
```

Type 2: Error identification

Given pseudocode may have missing END, wrong variable, wrong condition.

Type 3: Fill in the blanks

They may ask what condition/statement should be placed.

Type 4: Trace & logic check

Given a piece of pseudocode, you need to find the number of iterations, or final result

4. Shortcuts & Tricks

Loop Estimation Trick → Always check loop entry and exit conditions

Dry Run Method → Make a table with variables → step-by-step execution

Spot Common Errors → Missing END, wrong assignment (= vs <-)

Time-Saving Tip → If you know the logic (like factorial, sum, palindrome), don't dry run fully—just check edge values

5. Practice Examples

Write pseudocode for:

- Checking if a number is prime.
- Finding factorial (iterative & recursive).
- Printing Fibonacci series up to N.
- Reversing an array.
- Palindrome check for strings

BEGIN

$X \leftarrow 1$

WHILE $X < 10$ DO

$X \leftarrow X * 2$

END WHILE

PRINT X

END

- # Final Output? (A=9, B=8)

Impact Tip: In Infosys, pseudocode questions are usually moderate difficulty. They test logical thinking, not coding expertise. If you master tracing and dry runs, you'll score high in this section

Infosys-Level Pseudocode Examples

Example 1: Loop with Increment & Condition

BEGIN

$A \leftarrow 2$

$B \leftarrow 5$

WHILE $A < B$ DO

$A \leftarrow A + 2$

$B \leftarrow B - 1$

END WHILE

PRINT A, B

END

Dry Run:

- Iteration 1 $\rightarrow A=4, B=4 \rightarrow$ loop stops.

Output: 4 4

Example 2: Nested Loop

```
BEGIN
  COUNT ← 0
  FOR I ← 1 TO 3 DO
    FOR J ← 1 TO I DO
      COUNT ← COUNT + 1
    END FOR
  END FOR
  PRINT COUNT
END
```

Explanation:

- $I=1 \rightarrow J=1 \rightarrow \text{COUNT}=1$
- $I=2 \rightarrow J=1,2 \rightarrow \text{COUNT}=3$
- $I=3 \rightarrow J=1,2,3 \rightarrow \text{COUNT}=6$

Output: 6

Example 3: Tricky Condition

```
BEGIN
  X ← 0
  FOR I ← 1 TO 5 DO
    IF I MOD 2 = 0 THEN
      X ← X + I
    ELSE
      X ← X - I
    END IF
  END FOR
  PRINT X
END
```

Explanation:

- $I=1 \rightarrow \text{odd} \rightarrow X=-1$
- $I=2 \rightarrow \text{even} \rightarrow X=1$
- $I=3 \rightarrow \text{odd} \rightarrow X=-2$
- $I=4 \rightarrow \text{even} \rightarrow X=2$
- $I=5 \rightarrow \text{odd} \rightarrow X=-3$

Output: -3

Example 4: Factorial via Loop

```
BEGIN
  FACT ← 1
  N ← 5
  FOR I ← 1 TO N DO
    FACT ← FACT * I
  END FOR
  PRINT FACT
END
```

Output: 120

(Infosys loves factorial/Fibonacci/palindrome style questions.)

Example 5: Array Maximum



```
BEGIN
  ARR ← [3, 9, 2, 7, 5]
  MAX ← ARR[1]
  FOR I ← 2 TO 5 DO
    IF ARR[I] > MAX THEN
      MAX ← ARR[I]
    END IF
  END FOR
  PRINT MAX
END
```

Output: 9

Example 6: Palindrome Check

```
BEGIN
  STR ← "LEVEL"
  REV ← ""
  FOR I ← LENGTH(STR) DOWNT0 1 DO
    REV ← REV + STR[I]
  END FOR
  IF STR = REV THEN
    PRINT "PALINDROME"
  ELSE
    PRINT "NOT PALINDROME"
  END IF
END
```

Output: PALINDROME

Example 7: Fibonacci Series (Common in Infosys)

```
BEGIN
  N ← 6
  A ← 0
  B ← 1
  PRINT A, B
  FOR I ← 3 TO N DO
    C ← A + B
    PRINT C
    A ← B
    B ← C
  END FOR
END
```

Output: 0 1 1 2 3 5

Example 8: Logical Trace



```
BEGIN
  A ← 5
  B ← 10
  C ← 0
  WHILE A < B DO
    A ← A + 2
    B ← B - 3
    C ← C + 1
  END WHILE
  PRINT A, B, C
END
```

Dry Run:

- Iter 1: A=7, B=7, C=1 → loop ends.

Output: 7 7 1

Example 9: Error Identification (Infosys Style)

```
BEGIN
  SUM ← 0
  FOR I ← 1 TO 5
    SUM ← SUM + I
  PRINT SUM
END
```

Error: Missing END FOR.

Corrected version will output 15

Example 10: Pattern Printing (High-Frequency)

```
BEGIN
  N ← 4
  FOR I ← 1 TO N DO
    FOR J ← 1 TO I DO
      PRINT "*"
    END FOR
    PRINT NEWLINE
  END FOR
END
```

```
*
**
***
****
```