

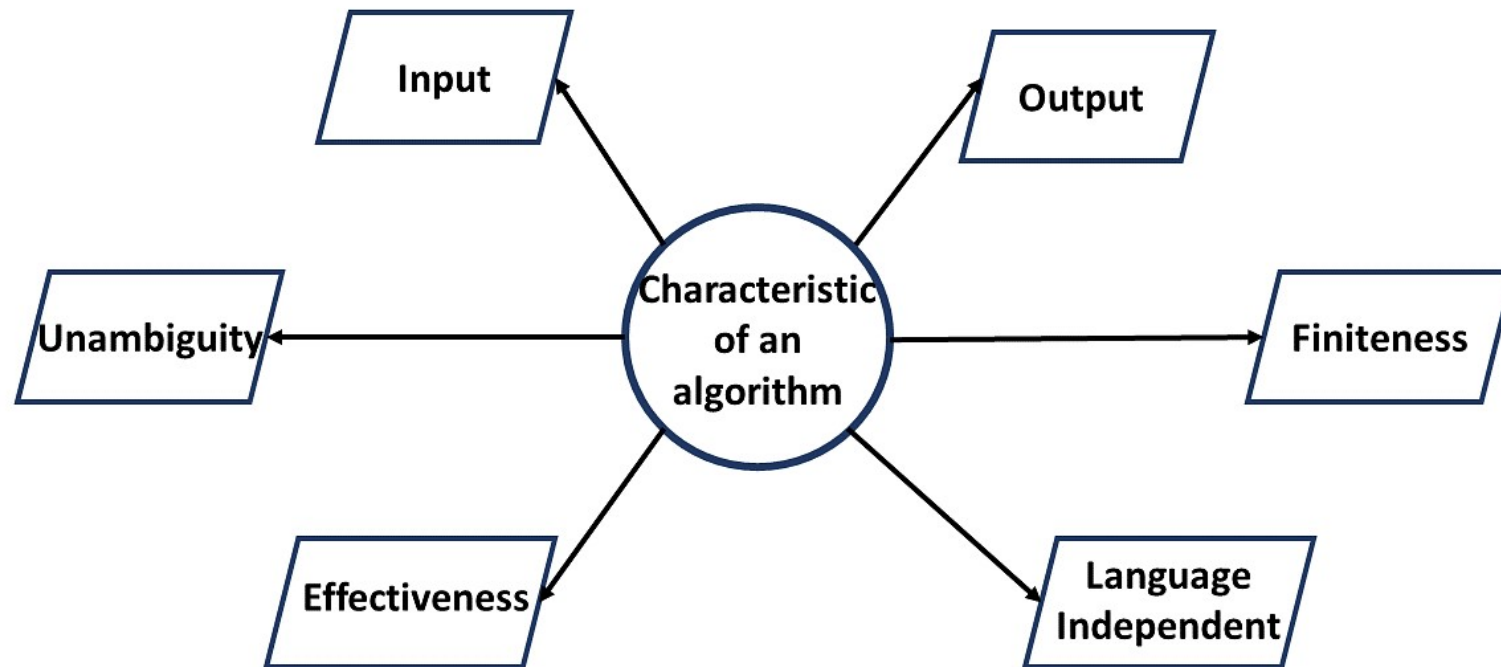
Algorithms & Flowcharts

K.N. Toosi

Fall 2022

Algorithms

- “algorithm” from the name of mathematician Al-khowarizmi which means a *procedure* or a *technique*.
- Algorithms are used for *planning* and *solving* problems.
- An algorithm is a **sequence of steps** to solve a particular problem, is an **ordered** set of **unambiguous steps** that **produces a result** and **terminates in a finite time**.
- Algorithms are generally developed **independently** of underlying languages, which means that an algorithm can be implemented in more than one programming language.



-
- **Input:** An algorithm requires some input values.
 - **Output:** At the end of an algorithm, you will have one or more outcomes.
 - **Unambiguity:** A perfect algorithm is defined as unambiguous, which means that its instructions should be clear and straightforward.
 - **Finiteness:** An algorithm must be finite. Finiteness in this context means that the algorithm should have a limited number of instructions, i.e., the instructions should be countable.
 - **Effectiveness:** Because each instruction in an algorithm affects the overall process, it should be adequate.
 - **Language independence:** An algorithm must be language-independent, which means that its instructions can be implemented in any language and produce the same results.




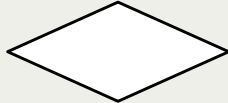

How to write algorithms?

- **Step 1:** Define your algorithms *input*. Many algorithms take in data to be processed.
 - e.g. to calculate the area of rectangle input may be the rectangle height and rectangle width.
- **Step 2:** Define the variables. Algorithm's variables allow you to use it for more than one place.
 - We can define two variables for rectangle height and rectangle width as HEIGHT and WIDTH (or H & W).
- **Step 3:** Outline the algorithm's operations. Use input variable for computation purpose.
 - e.g. to find area of rectangle multiply the HEIGHT and WIDTH variable and store the value in new variable (say) AREA. An algorithm's operations can take the form of multiple steps and even branch, depending on the value of the input variables.
- **Step 4:** Output the results of your algorithm's operations.
 - In case of area of rectangle output will be the value stored in variable AREA. if the input variables described a rectangle with a HEIGHT of 2 and a WIDTH of 3, the algorithm would output the value of 6.

Flowcharts

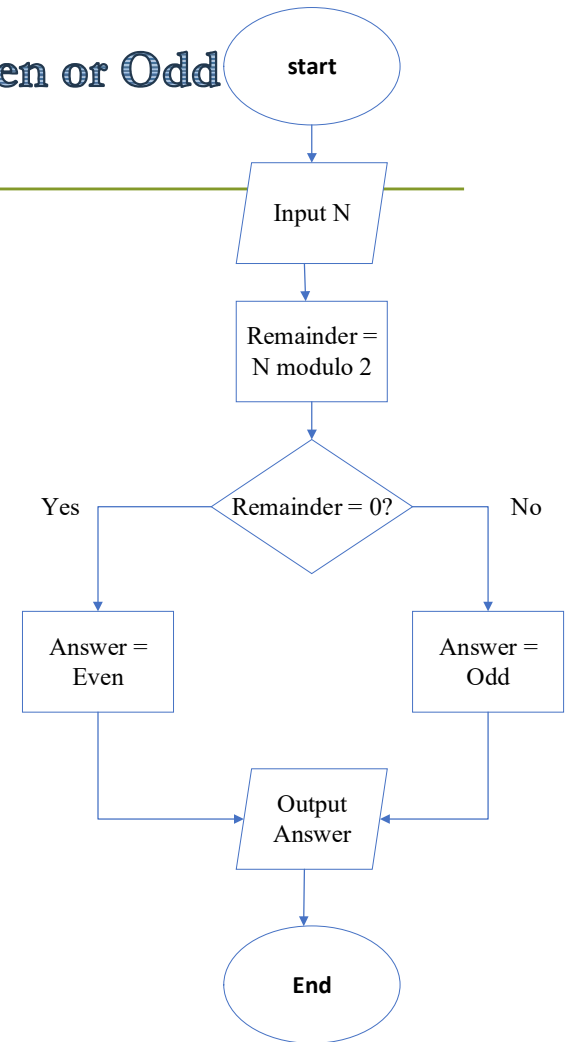
- The first design of flowchart goes back to 1945 which was designed by John Von Neumann. Flowchart uses different symbols to represent a solution to a problem.
- Flowchart is diagrammatic / Graphical representation of sequence of steps to solve a problem.

Draw Flowcharts

Name	Symbol	function
Oval		start and end of flowchart
Parallelogram		input and output operation
Rectangle		Processing: Used for arithmetic operations and data-manipulations
Diamond		Decision making. Used to represent the operation in which there are two/three alternatives, true and false etc
Arrows		Flow line Used to indicate the flow of logic by connecting symbols

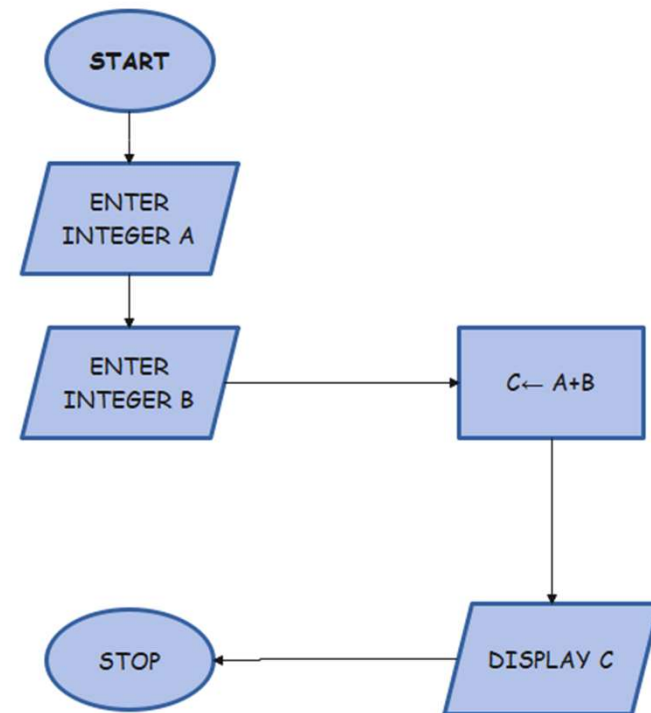
Example: Determine and Output Whether Number N is Even or Odd

- Step 1: Read number N.
- Step 2: Set remainder as N modulo 2.
- Step 3: If the remainder is equal to 0 then number N is even, else number N is odd.
- Step 4: Print output.



Example: Find the Sum of Two Numbers Entered

- Step 1: Read the Integer A.
- Step 2: Read Integer B.
- Step 3: Perform the addition by using the formula: $C = A + B$.
- Step 4: Print the Integer C.



Example: Swap Two Numbers using Temporary Variable

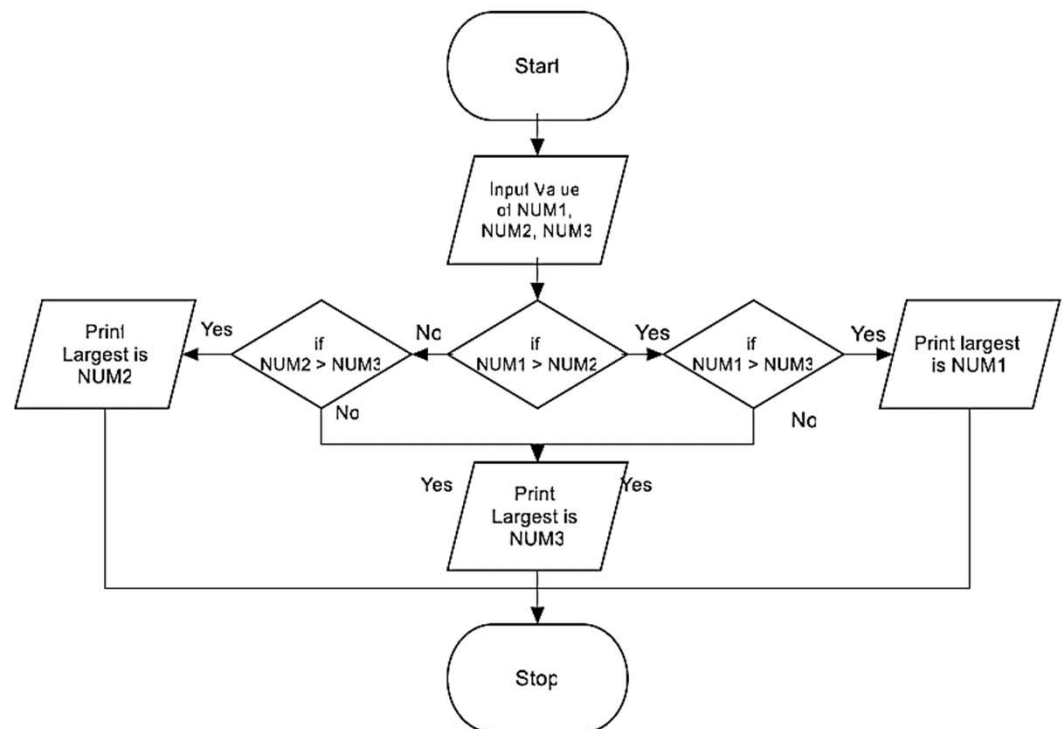
- Step-1 Input Two Numbers num1,num2
- Step-2 temp = num1
- Step-3 num1 = num2
- Step-4 num2 = temp
- Step-5 Display After Swap Values num1,num2

Swap Two Numbers without using temporary variable

- Step-1 Input Two Numbers Say A,B
- Step-2 Display Before Swap Values A, B
- Step-3 $A = A + B$
- Step-4 $B = A - B$
- Step-5 $A = A - B$
- Step-6 Display After Swap Values A, B

Find the largest of three numbers

- Step-1 Read three numbers say num1,num2, num3
- Step-2 if num1>num2 then go to step-5
- Step-3 IF num2>num3 THEN
 print num2 is largest
ELSE
 print num3 is largest
ENDIF
GO TO Step-6
- Step-5 IF num1>num3 THEN
 print num1 is largest
ELSE
 print num3 is largest
ENDIF
- Step-6 Stop



Find the Fibonacci series till term ≤ 1000

- Step 1: Declare the variables i, a, b, show.
- Step 2: Enter the values for the variables, a=0, b=1, show=0
- Step 3: Enter the terms of the Fibonacci series to be printed, i.e., 1000.
- Step 4: Print the first two terms of the series.
- Step 5: Loop the following steps:

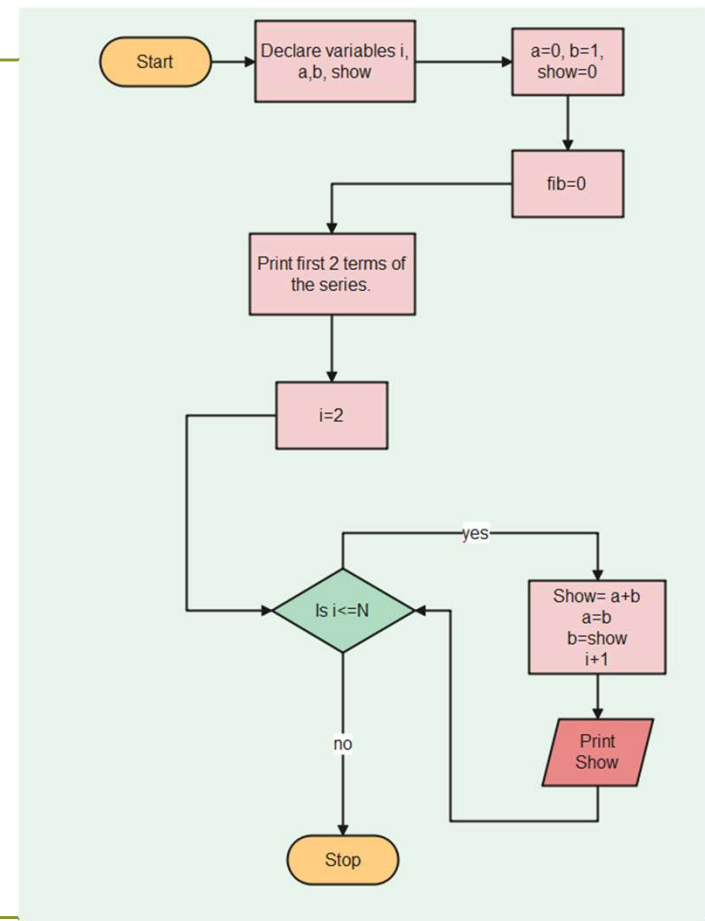
Show = a + b

a = b

b = show

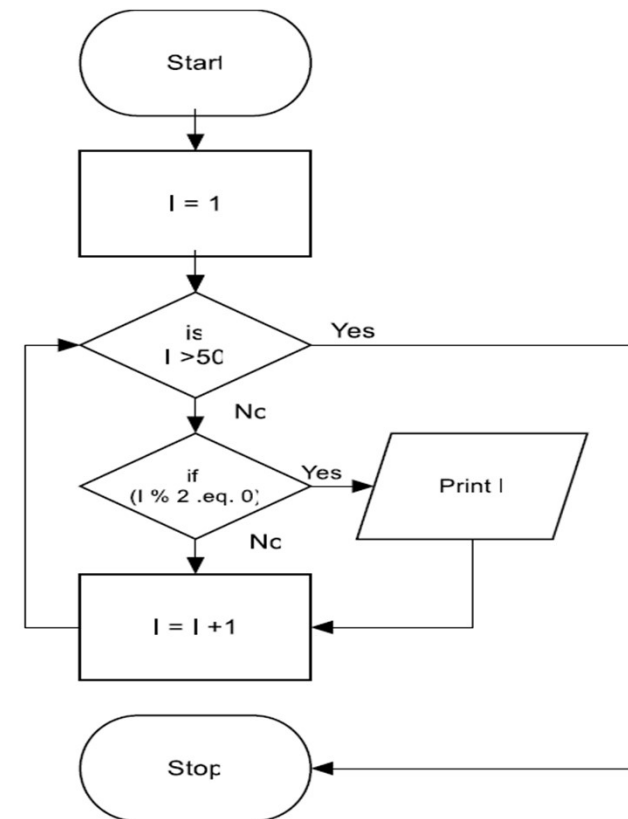
Add 1 to the value of i each time.

Print Show



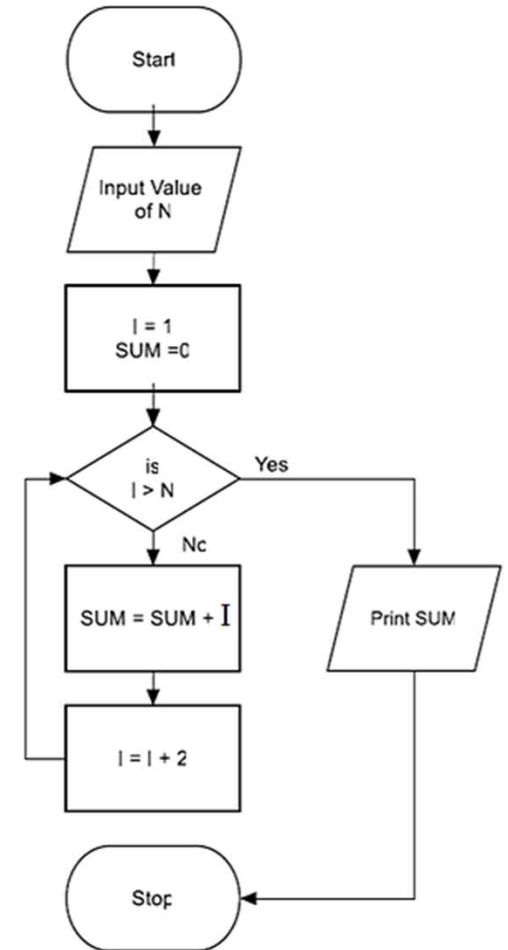
Find Even number between 1 to 50

- Step-1 $I = 1$
- Step-2 IF ($I > 50$) THEN
GO TO Step-6
ENDIF
- Step-3 IF ($(I \% 2) = 0$) THEN
Display I
ENDIF
- Step-4 $I = I + 1$
- Step-5 GO TO Step--2
- Step-6 Stop



Find sum of series $1+3+5+\dots+N$, Where N is positive odd Integer

- Step-1 Input Value of N
- Step-2 $I = 1$, $SUM = 0$
- Step-3 IF ($I > N$) THEN
 GO TO step 8
 ENDIF
- Step-4 $SUM = SUM + I$
- Step-5 $I = I + 2$
- Step-6 Go to step-3
- Step-7 Display value of SUM
- Step-8 Stop



Find sum of series $1 - X + X^2 - X^3 \dots X^N$

- Step-1 Input Value of N, X
- Step-2 $I = 1$, $SUM = 1$, $TERM = 1$
- Step-3 IF ($I > N$) THEN
 GO TO Step-9
 ENDIF
- Step-4 $TERM = - TERM * X$
- Step-5 $SUM = SUM + TERM$
- Step-6 $I = I + 1$
- Step-7 Go to step-3
- Step-8 Display value of SUM
- Step-9 Stop

