

FLOWCHARTING

The flowchart is a means of visually presenting the flow of data through an information processing systems, the operations performed within the system and the sequence in which they are performed. In this lesson, we shall concern ourselves with the program flowchart, which describes what operations (and in what sequence) are required to solve a given problem. The program flowchart can be likened to the blueprint of a building. As we know a designer draws a blueprint before starting construction on a building. Similarly, a programmer prefers to draw a flowchart prior to writing a computer program. As in the case of the drawing of a blueprint, the flowchart is drawn according to defined rules and using standard flowchart symbols prescribed by the American National Standard Institute (ANSI), Inc.

MEANING OF A FLOWCHART

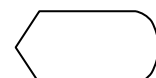
A flowchart is a diagrammatic representation that illustrates the sequence of operations to be performed to get the solution of a problem. Flowcharts are generally drawn in the early stages of formulating computer solutions. Flowcharts facilitate communication between programmers and business people. These flowcharts play a vital role in the programming of a problem and are quite helpful in understanding the logic of complicated and lengthy problems. Once the flowchart is drawn, it becomes easy to write the program in any high level language. Often we see how flowcharts are helpful in explaining the program to others. Hence, it is correct to say that a flowchart is a must for the better documentation of a complex program.

GUIDELINES FOR DRAWING A FLOWCHART

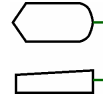
Flowcharts are usually drawn using some standard symbols; however, some special symbols can also be developed when required. Some standard symbols, which are frequently, required for flowcharting many computer programs are shown in Fig. 1

INPUT/OUTPUT SYMBOLS

- **Document** symbol represents a document or report that is prepared by hand or printed by a computer.
- **Multiple Copies of One Document**--The document copies should be numbered in the upper, right-hand corner.
- **Input/Output; Journal/Ledger** symbol can represent any input or output on a program flowchart. Also represents accounting journals or ledgers in a document flowchart.
- **Display** symbol represents information displayed by an online output device such as a terminal, monitor, or screen.
- **Online Keying** symbol represents data entry by an online device such as a terminal or personal computer.

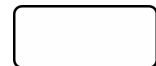


- **Terminal or Personal Computer** symbol combines the display and online keying symbols to represent terminals and personal computers.
- **Transmittal Tape** symbol represents manually prepared control totals which are to be compared to computer totals for control purposes.



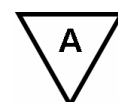
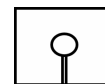
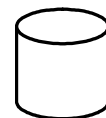
PROCESSING SYMBOLS

- **Computer Processing** symbol represents a process performed by a computer, which usually results in a change in data or information.
- **Manual Operation** symbol represents a processing operation that is performed manually.
- **Auxiliary Operation** represents a processing operation carried out by a device other than a computer, e.g., an optical character scanner.
- **Off-line Keying Operation** symbol represents an operation that uses an off-line keying device, such as a cash register or keying to a disk.



STORAGE SYMBOLS

- **Magnetic disk** symbol represents data stored permanently on a magnetic disk. Frequently used to represent master files and databases.
- **Magnetic Tape** symbol represents data stored on a magnetic tape. Sometimes represents transaction files.
- **Diskette** symbol represents data stored on a floppy disk or zip disk.
- **Online Storage** symbol represents data stored in a temporary online file on a direct-access medium such as a magnetic disk.
- **File** symbol represents a file of documents that are manually stored and retrieved. Letter indicates the ordering sequence: A = alphabetic order; D = date order; N = numeric order.



FLOW AND MISCELLANEOUS SYMBOLS

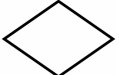
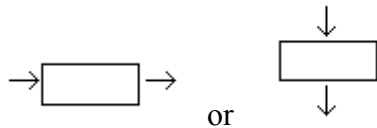
- **Document or Processing Flow** symbol represents the direction of processing or document flow. Normal flow is top to bottom and left to right. 
- **Data/Information Flow** symbol represents the direction of data/information flow. Often used to show data being copied from one document to another. 
- **Communication Link** symbol represents the transmission of data from one location to another via communication lines. 
- **On-page connector** symbol connects processing from one location to another on the same page. Used to avoid crisscrossing lines. 
- **Off-page connector** symbol connects the processing flow between two different pages. Signals the exit from one page and the corresponding entrance on another page. 
- **Terminal** symbol represents the beginning, end, or a point of interruption in a process or program. Also used to indicate an external party. 
- **Decision** symbol represents a decision-making step. Used in a program flowchart to show branching to alternate paths. 
- **Annotation** symbol provides for the addition of descriptive comments or explanatory notes as clarification. 

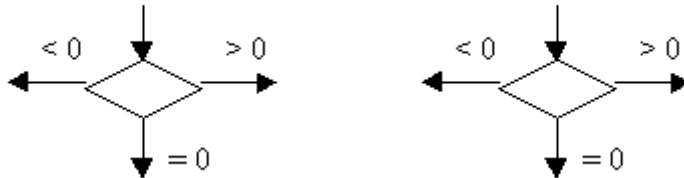
Fig. 1 Flowchart Symbols

The following are some guidelines in flowcharting:

- a. In drawing a proper flowchart, all necessary requirements should be listed out in logical order.
- b. The flowchart should be clear, neat and easy to follow. There should not be any room for ambiguity in understanding the flowchart.
- c. The usual direction of the flow of a procedure or system is from left to right or top to bottom.
- d. Only one flow line should come out from a process symbol.



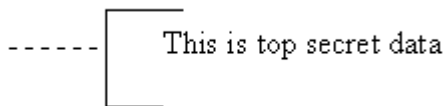
- e. Only one flow line should enter a decision symbol, but two or three flow lines, one for each possible answer, should leave the decision symbol.



- f. Only one flow line is used in conjunction with terminal symbol.



- g. Write within standard symbols briefly. As necessary, you can use the annotation symbol to describe data or computational steps more clearly.



- h. If the flowchart becomes complex, it is better to use connector symbols to reduce the number of flow lines. Avoid the intersection of flow lines if you want to make it more effective and better way of communication.
- i. Ensure that the flowchart has a logical *start* and *finish*.
- j. It is useful to test the validity of the flowchart by passing through it with a simple test data.

ADVANTAGES OF USING FLOWCHARTS

The benefits of flowcharts are as follows:

1. Communication: Flowcharts are better way of communicating the logic of a system to all concerned.
2. Effective analysis: With the help of flowchart, problem can be analysed in more effective way.
3. Proper documentation: Program flowcharts serve as a good program documentation, which is needed for various purposes.
4. Efficient Coding: The flowcharts act as a guide or blueprint during the systems analysis and program development phase.
5. Proper Debugging: The flowchart helps in debugging process.
6. Efficient Program Maintenance: The maintenance of operating program becomes easy with the help of flowchart. It helps the programmer to put efforts more efficiently on that part

LIMITATIONS OF USING FLOWCHARTS

1. Complex logic: Sometimes, the program logic is quite complicated. In that case, flowchart becomes complex and clumsy.
2. Alterations and Modifications: If alterations are required the flowchart may require re-drawing completely.
3. Reproduction: As the flowchart symbols cannot be typed, reproduction of flowchart becomes a problem.
4. The essentials of what is done can easily be lost in the technical details of how it is done.

FEW EXAMPLES ON FLOWCHARTING

Now we shall present few examples on flowcharting for proper understanding of this technique. This will help in student in program development process at a later stage.

Example 1

Draw a flowchart to find the sum of first 50 natural numbers.

Answer: The required flowchart is given in Fig. 2.

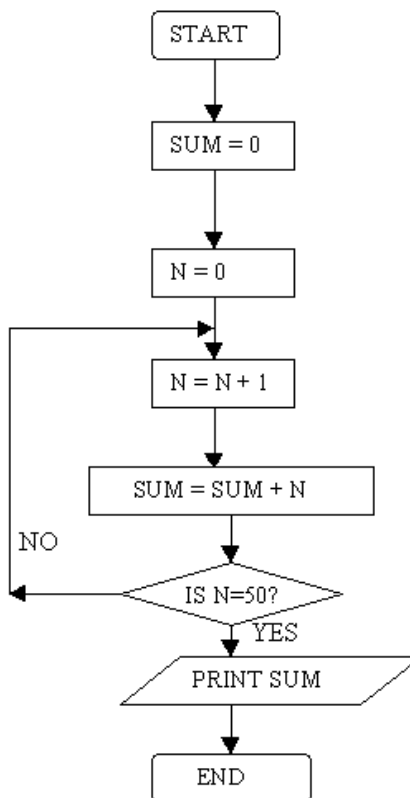


Fig. 2 Sum of first 50 natural numbers

Fig 2.2 Flowchart for computing the sum of first 50 natural numbers.

Example 2

Draw a flowchart to find the largest of three numbers A,B, and C.

Answer: The required flowchart is shown in Fig 3

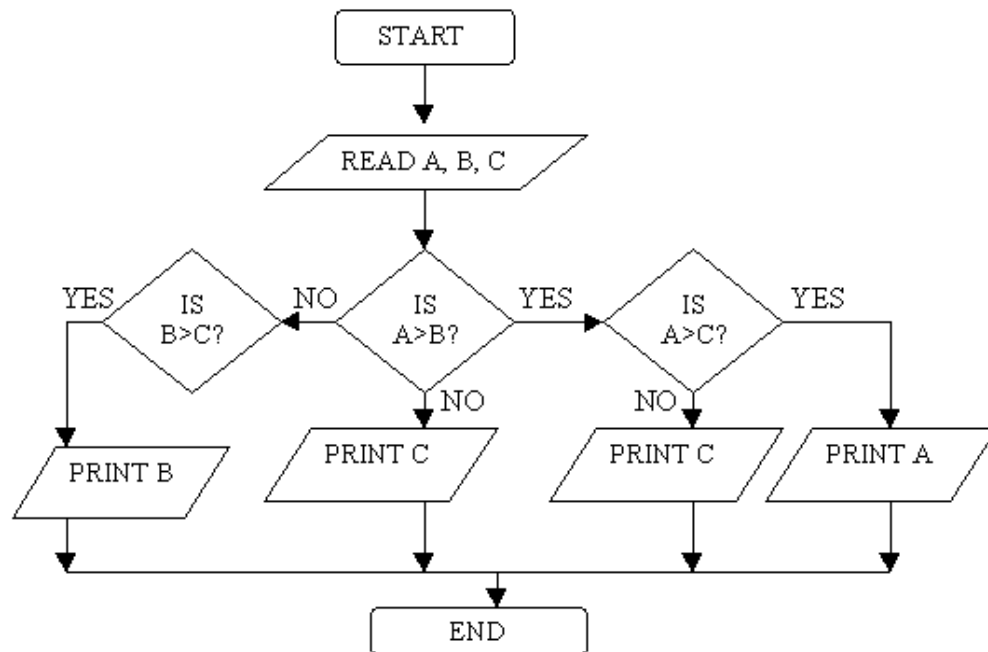


Fig 3 Flowchart for finding out the largest of three numbers

Example 3

Draw a flowchart for computing factorial N (N!)

Where $N! = 1 \times 2 \times 3 \times \dots \times N$.

The required flowchart has been shown in fig 4

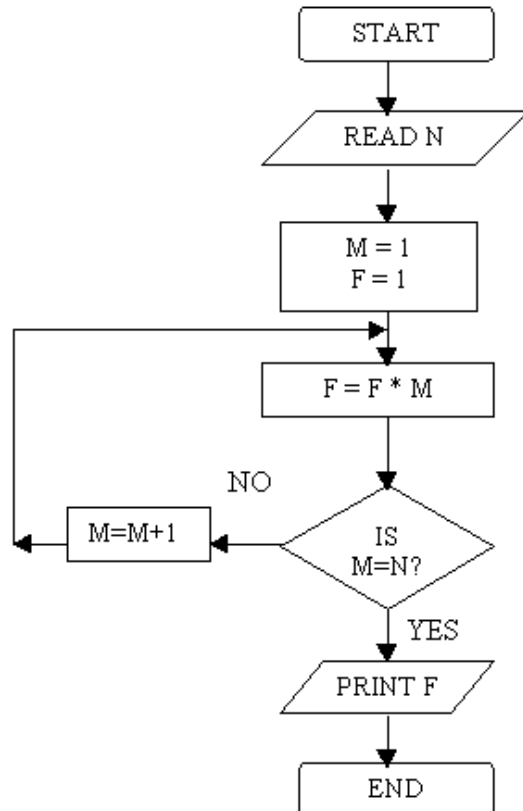


Fig 4 Flowchart for computing factorial N