



Algorithms

Forms of algorithms representation

Algorithms

Algorithms can be expressed in many kinds of notation, including natural languages, pseudo-codes, flowcharts, and programs for computer. Natural language expressions of algorithms tend to be verbose and ambiguous, and are rarely used for complex or technical algorithms. Pseudo-code and flowcharts are structured ways to express algorithms that avoid many of the ambiguities common in natural language statements, while remaining independent of a particular implementation language. Programming languages are primarily intended for expressing algorithms in a form of program that can be executed by a computer.



Programming languages and their classification

Programming languages

- A **programming language** is an artificial language that can be used to control the behavior of a machine, particularly a computer. Programming languages, like natural languages, are defined by syntactic and semantic rules which describe their structure and meaning respectively. Many programming languages have some form of written specification of their syntax and semantics; some are defined only by an official implementation.
- Programming languages are used to facilitate communication about the task of organizing and manipulating information, and to express algorithms precisely. Some authors restrict the term "programming language" to those languages that can express *all* possible algorithms; sometimes the term "computer language" is used for more limited artificial languages.
- Thousands of different programming languages have been created, and new languages are created every year.



Flowchart

Introduction & Method of Problem Solving

The 6 steps in using a computer as a problem-solving tool

- Perform data analyzing
- Develop an *Algorithm* and a *Flowchart*.
- Write the program in a computer language.
- Enter the program into the computer.
- Test and debug the program.
- Run the program, input data, and get the results from the computer.

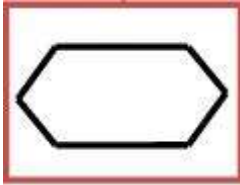
Definition of Algorithm and Flowchart

- An **Algorithm** is just a detailed **sequence** of simple **steps** that are needed to solve a problem.
- A **Flowchart** is a graphical representation of an algorithm.

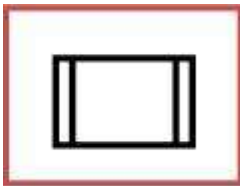
Basic Symbols

- Start/Stop
- Question, Decision (Use in Branching)
- Input/Output
- Connector (connect two flowcharts)
- Process, Instruction
- Comments, Explanations, Definitions.

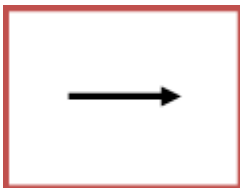
2 Additional Symbols



- Preparation (may be used with " loops" explained later)



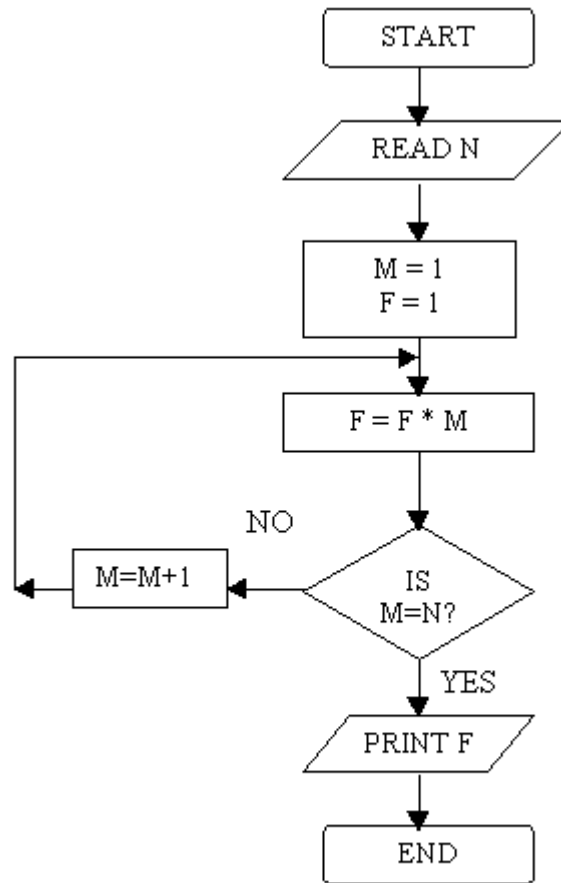
- Refers to separate flowchart ("Subprograms"(explained later) are shown in separate flowcharts).



- Flow line

\

Example:



Comments

A comment starts with a `/*` and ends with `*/`. Comments started in this way can span multiple lines but cannot be nested.

But comment that starts with `//` doesn't have necessity to be closed. And this comment span everything that is situated in the right side.

Example:

```
main()
{
    int Counter=0;    // Initializing Counter
    /* /* Invalid comment */ */
    /*
    *Valid comment
    */
    /*****
    *Valid comment.
    *****/
}
```