**Disclaimer:** The information below are all from the sources available online. Please refer to the citations if needed.

Padua, John Christian P.

Flowchart

*One way of representing steps is trough graphical or visual, in programming it is called a* ***flowchart***. Like other flowcharts, the data or command is arranged sequentially – meaning from the first down to the last step of algorithms, workflow or process. It makes use of various basic shapes as a stand or representation of a command. Flowcharts also have arrows, basically, to guide the flow or process.

It was originated from computer science as a tool for representing algorithms and programming logic but had extended to use in all other kinds of processes. We know it now in this contemporary world as a use for various kinds of processes, but originally, it was designed for the field of computer science representing algorithms and programming. Today, they *play an extremely important role in displaying information and assisting reasoning mainly to make it easier to understand*, since it is much easier and convenient to look at organized figures rather that scanning voluminous text commands or instructions. ((n.d.). Retrieved from <https://www.visual-paradigm.com/tutorials/flowchart-tutorial/>.)

Please refer to the table below for the list of shapes used in doing a flowchart.

|  |  |  |
| --- | --- | --- |
| Shape | Name | Purpose |
|  | Terminator | The terminator symbol represents the starting or ending point of the system. |
|  | Process | A box indicates some particular operation. |



Padua, John Christian P.

|  |  |  |
| --- | --- | --- |
| Shape | Name | Purpose |
|  | Document | This represent printouts, such as documents or reports. |
|  | Decision | A diamond represents a decision or branching point. Lines coming out from the diamond indicates different possible situations, leading to different sub-processes. |
|  | Data | It represents information entering or leaving the system. An input might be an order from a customer. Output can be a product to be delivered. |
|  | On-Page Reference | This symbol would contain a letter inside. It indicates that the flow continues on a matching symbol containing the same letter somewhere else on the same page. |
|  | Off-Page Reference | This symbol would contain a letter inside. It indicates that the flow continues on a matching symbol containing the same letter somewhere else on a different page. |
|  | Delay or Bottleneck | Identifies a delay or a bottleneck. |
|  | Flow | Lines represent the flow of the sequence and direction of a process. |

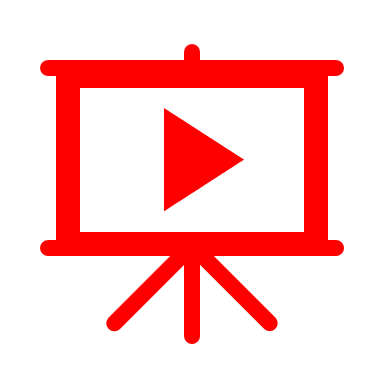


Padua, John Christian P.

When to Draw Flowchart?

Using a flowchart has a variety of benefits:

* It helps to clarify complex processes.
* It identifies steps that do not add value to the internal or external customer, including delays; needless storage and transportation; unnecessary work, duplication, and added expense; breakdowns in communication.
* It helps team members gain a shared understanding of the process and use this knowledge to collect data, identify problems, focus discussions, and identify resources.
* It serves as a basis for designing new processes.

Those were the basics of flowchart that a beginner must know or at least be familiar at, you can also search for more shapes that are used for flowcharting, however the mentioned shapes above the basics amongst other more. If you wat more technicalities visit the YouTube link below.

<https://www.youtube.com/watch?v=iJmcgQRk048>



Padua, John Christian P.

Pseudocodes

**Pseudocode**is a method of describing computer algorithms using a combination of natural language and programming language.  It is essentially an intermittent step towards the development of the actual code.  It allows the programmer to formulate their thoughts on the organization and sequence of a computer algorithm without the need for actually following the exact coding syntax.  Although pseudocode is frequently used there are no set of rules for its exact implementation.  In general, here are some rules that are frequently followed when writing pseudocode:

* The usual Fortran symobols are used for arithmetic operations (+, -, \*, / , \*\*).
* Symbolic names are used to indicate the quantities being processed.
* Certain Fortran keywords can be used, such as PRINT, WRITE, READ, etc.
* Indentation should be used to indicate branches and loops of instruction.

Here is an example problem, including a flowchart, pseudocode, and the final Fortran 90 program.  This problem and solution are from Nyhoff, pg 206:

For a given value, *Limit*, what is the smallest positive integer *Number* for which the sum

*Sum*= 1 + 2 + ... + *Number*

is greater than *Limit*.  What is the value for this *Sum*?

Pseudocode:

    Input:    An integer *Limit*  
    Ouput:   Two integers: *Number* and *Sum*

1.  Enter *Limit*  
2.  Set *Number* = 0.  
3.  Set *Sum* = 0.  
4.  Repeat the following:  
    a.  If *Sum*> *Limit*, terminate the repitition, otherwise.  
    b.  Increment *Number*by one.  
    c.  Add *Number*to *Sum*and set equal to *Sum*.  
5.  Print *Number* and *Sum*.

(n.d.). Retrieved from https://www.visual-paradigm.com/tutorials/flowchart-tutorial/.

Resources:

Padua, John Christian P.

((n.d.). Retrieved from <https://www.visual-paradigm.com/tutorials/flowchart-tutorial/>

<https://www.youtube.com/watch?v=iJmcgQRk048>

(n.d.). Retrieved from https://www.visual-paradigm.com/tutorials/flowchart-tutorial/.