**System Software and Programming Techniques**

**Flow Charts: Software, Algorithm Design**

**By Odiscious Dozier**

**Trident University**

INTRODUCTION

In 1911, the expression “Use a picture. It’s worth a thousand words.” was something newspaper editor Arthur Brisbane conveyed while discussing journalism and publicity (The history of a picture's worth, n.d.). As time moved forward the use of images in professional settings grew and is now a mandatory part of today’s professional environment.

This discussion is about flowcharts and answers, how can a person utilize a big visual display (BVD) to convey as much information as possible, in the least amount of space, reading a minimal amount of text? Also, are there any special symbols or more relevant symbols that could be used to convey a message to a group of people that may be familiar with the same type of documentation or BVD?

FLOWCHARTS

The importance of flowcharts is not limited to just one field or discipline. They are widely used throughout the modern professional world. Flowcharts have proven to be an excellent tool for, process documentation, training materials, workflow management, continuous improvement, trouble-shooting guides, regulatory management, quality control management, and programming (Hebb, n.d.).

In designing the user’s interface, it is important to clearly understand the true workflow of the user. Flowcharts have proven to be an invaluable tool in representing decision making and logic while utilizing the advantages of a BVD. With appropriate requirements gathering techniques, programmers have the ability to document and diagram a user’s experience as they navigate through a program. This helps the product owners discern if the observed navigation is what they truly want or not. Changes can then be made in the design phase and not the implementation phase of the overall software project.

In designing an algorithm, text, alone, can be quite laborious to maintain a working knowledge of order of operations or precedence and relationships. This is one reason why pseudocode is not frequently used by software engineers, rather heavily used by non-computer professionals; as, the difficulty of the problems being solved move from known to unknown with the greater skill of the software engineer. Flowcharts show these concepts in a very clear and concise manner that engineers can easily understand, utilize or modify. The ease of structuring an algorithm helps with the solving of complex problems and for this reason is considered an essential part of the documentation process of any computer program.

Getting a good visual representation of a problem is a powerful ability for software developers. Flowcharts can be a diagrammatic representation of a flow of a problem and as such they provide excellent means of documentation in the software development life cycle (SDLC), but also act as a means to communicate with several developers or teams of developers – regardless of time, provided they’re kept current (Ravichandra, 2001). This increase of appeal is enhanced with the simplicity and efficiency of representing the program logic of a problem.

In testing, as stated earlier, desk-checking is a test practice that enables developers to test their solutions without writing any code (King, 1999). They write the entirety of their problem solution and write data sets to go through each step of the algorithm to prove its general functionality. Depending on the size and difficulty of the particular algorithm that is to be tested, several data sets may be required. Flowcharts provide a method for easy desk-checking.

STANDARD SYMBOLS

There are several standard or common symbols used in the creation/design of flowcharts. Figure 1 is a list of images and names of the standard symbols used in today’s practice of flowcharts.

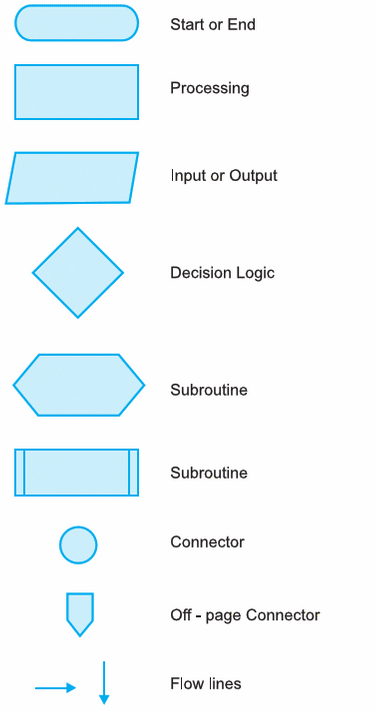


Figure 1: Standard Symbols for flowcharts (Ram, 20117)

SPECIAL SYMBOLS

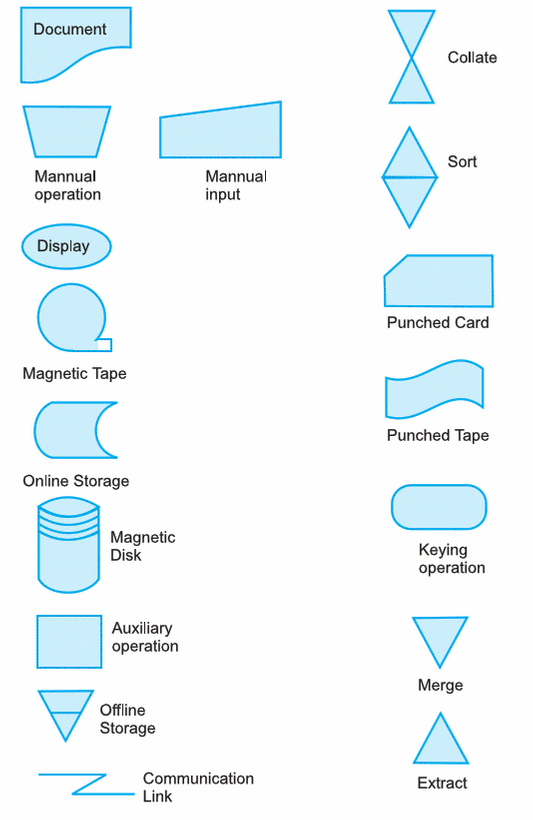
There are several special symbols used in the creation/design of flowcharts. Figure 2 is a list of images and names of the special symbols used in today’s practice of flowcharts.

Figure 2: Special Symbols for flowcharts (Ram, 2007)

REFERENCES

Ajit, P. (2012), Department of Computer Science & Engineering., IIT KGP, Lecture 8 Assembly Language Programming, Retrieved from. <http://www.youtube.com/watch?v=HXYhBCpDoVc>

Hebb, N. (n.d.). The Top 5 Reasons To Use Flowcharts. Retrieved September 14, 2015, from <http://www.breezetree.com/articles/top-reasons-to-flowchart.htm>

King, P. (1999, August 31). Desk-checking. Retrieved September 14, 2015

Ravichandra, D. (2001). Algorithms, Flowcharts, and Programming Languages. In Introduction to computer and communication. New Delhi: Tata McGraw-Hill.

The history of a picture's worth. (n.d.). Retrieved September 14, 2015, from <http://www2.cs.uregina.ca/~hepting/research/web/words/history.html>