Flowcharting*

Kenneth Leroy Busbee

This work is produced by The Connexions Project and licensed under the
Creative Commons Attribution License †

Abstract

An introduction to flowcharting including the demonstration of functions and various control structures.

1 Flowcharting Symbols

1.1 Terminal

The rounded rectangles, or terminal points, indicate the flowchart's starting and ending points.

Figure 1

1.2 Process

The rectangle depicts a process such as a mathematical computation, or a variable assignment.

Note: the C++ language equivalent is the statement.

*Version 1.7: Mar 5, 2012 6:00 am -0600

†http://creativecommons.org/licenses/by/3.0/

Figure 2

1.3 Input/Output

The parallelograms designate input or output operations.

Note: the C++ language equivalent is cin or cout.

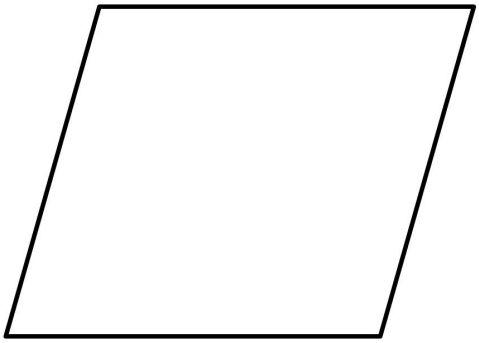

Figure 3

1.4 Connectors

Sometimes a flowchart is broken into two or more smaller flowcharts. This is usually done when a flowchart does not fit on a single page, or must be divided into sections. A connector symbol, which is a small circle

with a letter or number inside it, allows you to connect two flowcharts on the same page. A connector symbol that looks like a pocket on a shirt, allows you to connect to a flowchart on a different page.

On-Page Connector

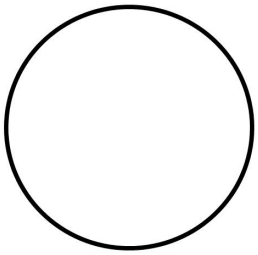

Figure 4

Off-Page Connector

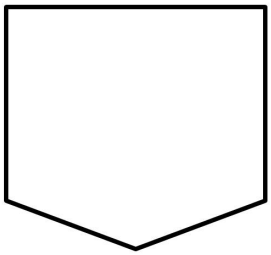

Figure 5

1.5 Decision

The diamond is used to represent the true/false statement being tested in a decision symbol.

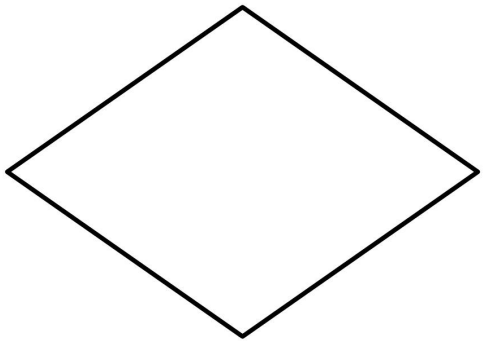

Figure 6

1.6 Module Call

A program module is represented in a flowchart by rectangle with some lines to distinguish it from process symbol. Often programmers will make a distinction between program control and specific task modules as shown below.

Note: C++ equivalent is the function.

Local module: usually a program control function.

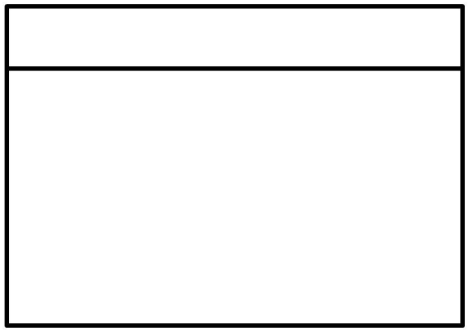

Figure 7

Library module: usually a specific task function.

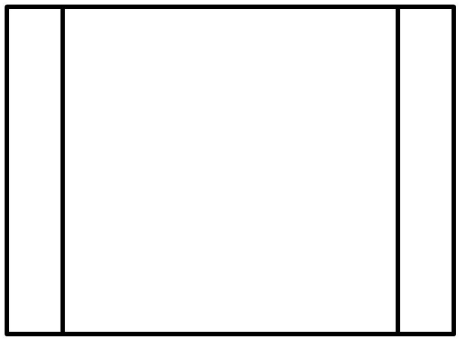

Figure 8

1.7 Flow Lines

Note: The default flow is left to right and top to bottom (the same way you read English). To save time arrowheads are often only drawn when the flow lines go contrary the normal.

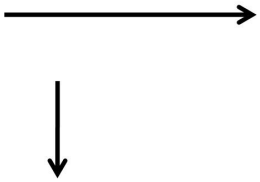

Figure 9

2 Examples

We will demonstrate various flowcharting items by showing the flowchart for some pseudocode.

2.1 Functions

Example 1: pseudocode: Function with no parameter passing

```
Function clear monitor
  Pass In: nothing
  Direct the operating system to clear the monitor
  Pass Out: nothing
Endfunction
```

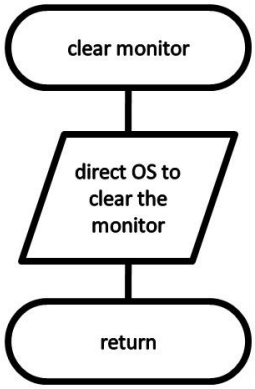

Figure 10: Function clear monitor

Example 2: pseudocode: Function main calling the clear monitor function

```
Function main
  Pass In: nothing
  Doing some lines of code
  Call: clear monitor
  Doing some lines of code
  Pass Out: value zero to the operating system
Endfunction
```

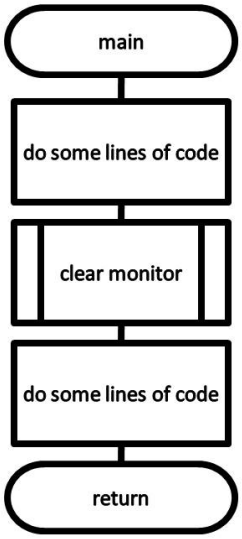

Figure 11: Function main

2.2 Sequence Control Structures

The next item is pseudocode for a simple temperature conversion program. This demonstrates the use of both the on-page and off-page connectors. It also illustrates the sequence control structure where nothing unusually happens. Just do one instruction after another in the sequence listed.

Example 3: pseudocode: Sequence control structure

```
Filename: Solution_Lab_04_Pseudocode.txt
Purpose:  Convert Temperature from Fahrenheit to Celsius
Author:   Ken Busbee; © 2008 Kenneth Leroy Busbee
Date:     Dec 24, 2008

Pseudocode = IPO Outline

input
  display a message asking user for the temperature in Fahrenheit
  get the temperature from the keyboard
processing
  calculate the Celsius by subtracting 32 from the Fahrenheit
  temperature then multiply the result by 5 then
  divide the result by 9. Round up or down to the whole number.
  HINT: Use 32.0 when subtracting to ensure floating-point accuracy.
```

```
output
  display the celsius with an appropriate message
  pause so the user can see the answer
```

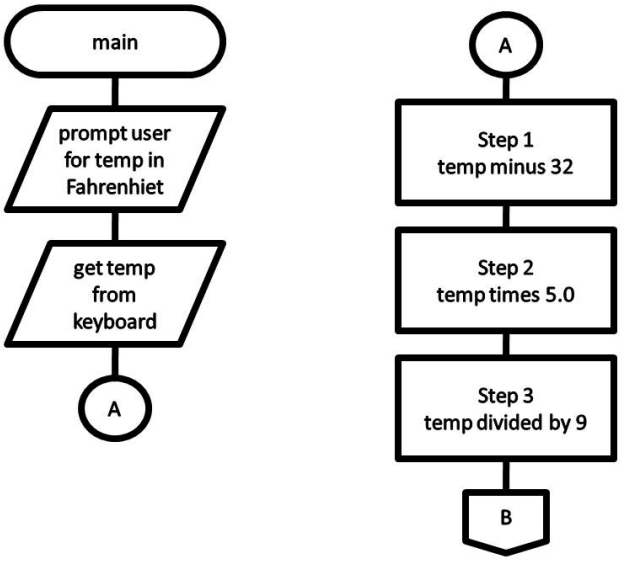

Figure 12: Sequence control structure

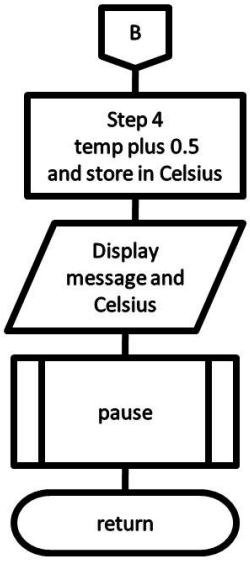

Figure 13: Sequence control structured continued

2.3 Selection Control Structures

Example 4: pseudocode: If then Else

```
If age > 17
  Display a message indicating you can vote.
Else
  Display a message indicating you can't vote.
Endif
```

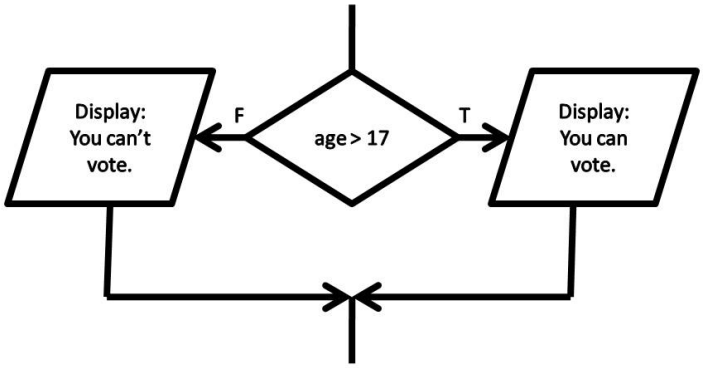

Figure 14: If then Else control structure

Example 5: pseudocode: Case

```
Case of age
  0 to 17   Display "You can't vote."
  18 to 64  Display "Your in your working years."
  65 +      Display "You should be retired."
Endcase
```

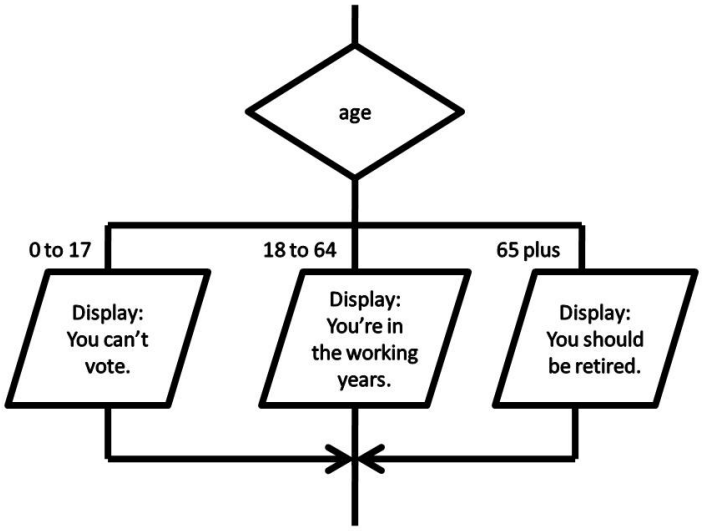

Figure 15: Case control structure

2.4 Iteration (Repetition) Control Structures

Example 6: pseudocode: While

```
count assigned zero
While count < 5
  Display "I love computers!"
  Increment count
Endwhile
```

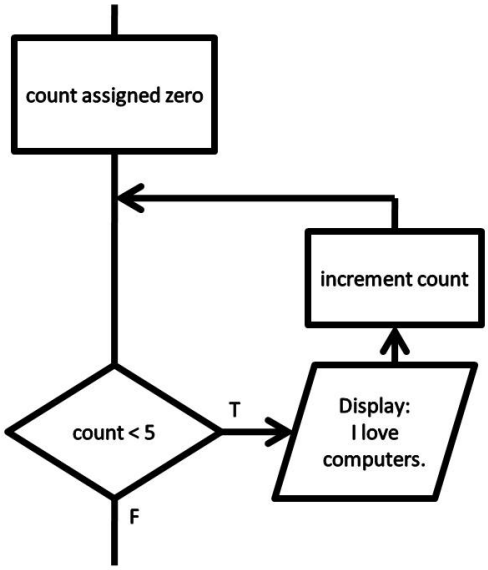

Figure 16: While control structure

Example 7: pseudocode: For

```
For x starts at 0, x < 5, increment x
  Display "Are we having fun?"
Endfor
```

The for loop does not have a standard flowcharting method and you will find it done in different ways. The for loop as a counting loop can be flowcharted similar to the while loop as a counting loop.

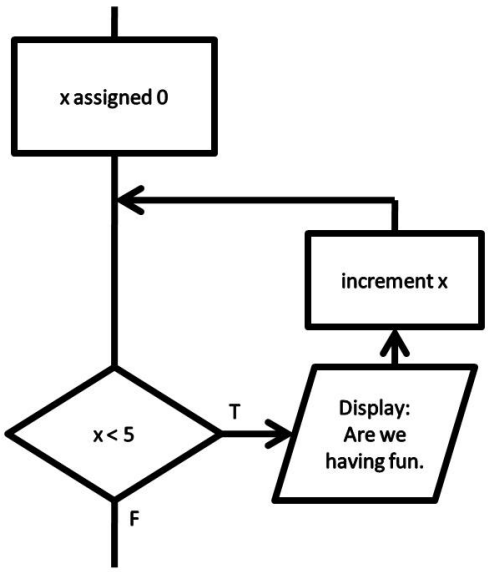

Figure 17: For control structure

Example 8: pseudocode: Do While

```
count assigned five
Do
  Display "Blast off is soon!"
  Decrement count
While count > zero
```

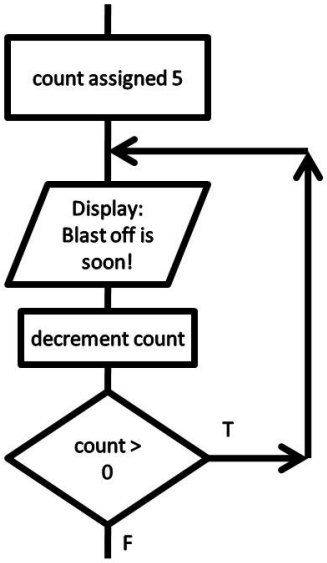

Figure 18: Do While control structure

Example 9: pseudocode: Repeat Until

```
count assigned five
Repeat
  Display "Blast off is soon!"
  Decrement count
Until count < one
```

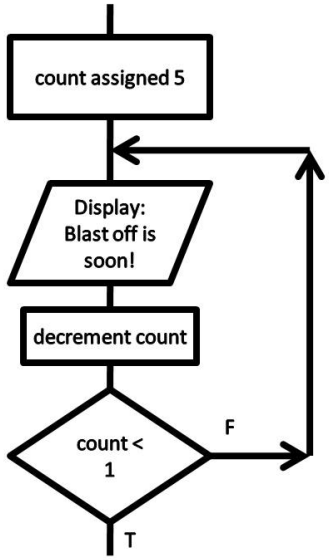

Figure 19: Repeat Until control structure

3 Definitions

Definition 1: flowcharting
A programming design tool that uses graphical elements to visually depict the flow of logic within a function.

Definition 2: process symbol
A rectangle used in flowcharting for normal processes such as assignment.

Definition 3: input/output symbol
A parallelogram used in flowcharting for input/output interactions.

Definition 4: decision symbol
A diamond used in flowcharting for asking a question and making a decision.

Definition 5: flow lines
Lines (sometimes with arrows) that connect the various flowcharting symbols.