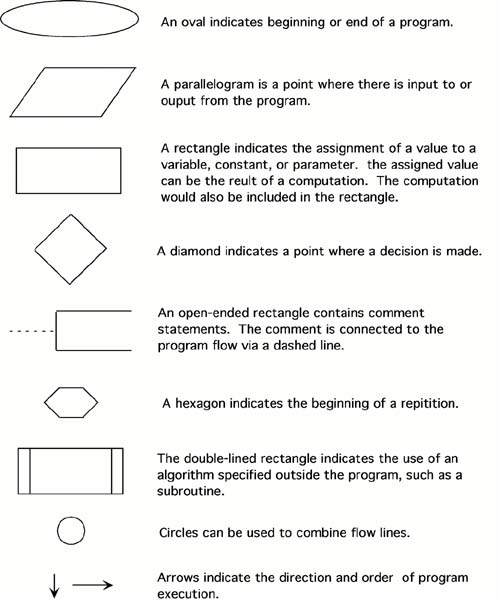
**Flowcharts and Pseudocode**

***Flowcharts*** are written with program flow from the top of a page to the bottom.  Each command is placed in a box of the appropriate shape, and arrows are used to direct program flow.  The following shapes are often used in flowcharts:

***Pseudocode*** is a method of describing computer algorithms using a combination of natural language and programming language.

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. This problem and solution are from Nyhoff, pg 206:

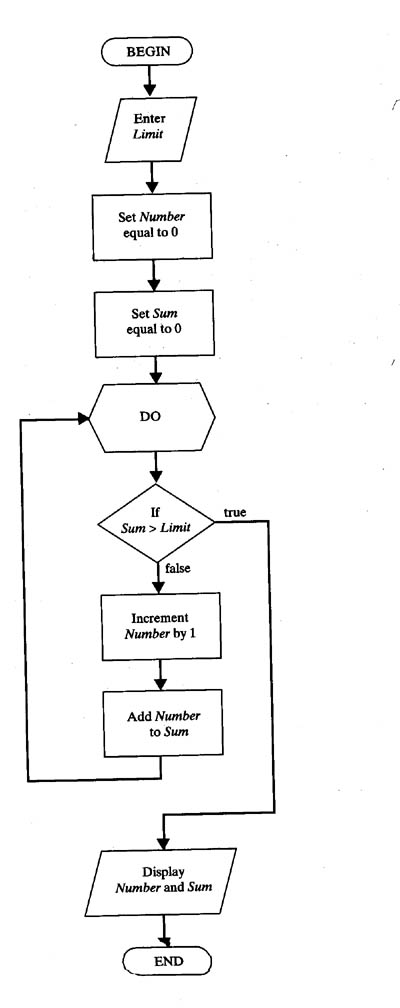
For a given value, *Limit*, what is the smallest positive integer *Number* for which the sum

*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*.

**Flowchart:**

**Differences Between Psuedocode and Flowcharts**

***\*Layout***

**Flowcharts-** for a program provides a graphical structure that allows programmers to follow the logical structure of the code.

**Pseudocodes-**uses words as opposed to a pictorial representation to illustrate the logic of the algorithm.

***\*Benefits***

**Flowchart**

**•** are especially beneficial for smaller concepts and problems.

•provide an easy method of communication about the logic and offer a good starting point for the project because they are easier to create than pseudocode in the beginning stages.

**Psuedocodes**

**•**is more efficient for larger programming problems.

•provides a beneficial bridge to the project code because it closely follows the logic that the code will.

•also helps programmers share ideas without spending too much time creating code, and it provides a structure that is not dependent on any one programming language.

***\*Structure***

**Flowchart -** uses symbols and shapes to create a diagram of only the essential parts of the algorithm or problem.

**Psuedocodes -** uses a linear text-based structure to organize the logic of the programming problem**.**

***\*Depth***

Both pseudocode and flowcharts can be either broad and cover only basic concepts of the algorithm or project, or they can cover more detail, making the jump to creating the codes easier. However, pseudocode has more flexibility with detail; too much detail in a flowchart can cause confusion.

**ADVANTAGES AND DISADVANTAGES OF FLOWCHAT AND PSEUDOCODES**

**Flowchart**

**Advantages:**

• Communication: Flowcharts are better way of communicating the logic of a system to all concerned or involved.

• Effective analysis: With the help of flowchart, problem can be analysed in more effective way therefore reducing cost and wastage of time.

• Proper documentation: Program flowcharts serve as a good program documentation, which is needed for various purposes, making things more efficient.

• Efficient Coding: The flowcharts act as a guide or blueprint during the systems analysis and program development phase.

• Proper Debugging: The flowchart helps in debugging process.

• 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

**Disadvantages:**

• Complex logic: Sometimes, the program logic is quite complicated. In that case, flowchart becomes complex and clumsy. This will become a pain for the user, resulting in a waste of time and money trying to correct the problem

• Alterations and Modifications: If alterations are required the flowchart may require re-drawing completely. This will usually waste valuable time.

• Reproduction: As the flowchart symbols cannot be typed, reproduction of flowchart becomes a problem.

**Pseudocodes**

**Advantages:**

•It is easier to develop a program from a pseudo code as compared to the flow chart. Programmers do not have to think about syntax, we simply have to concentrate on the underline logic. The focus is on the steps to solve a problem rather than how to use the computer language.

•Often it is easy to translate pseudocode into a programming language, a step which can be accomplished by less experienced

•The uses of words and phrases in pseudo code, which are in the lines of basic computer operations simplify the translation from the pseudo code algorithm to the specific programming language.

•Unlike flow charts, pseudo code is at and does not tend to run over many pages. Its simple structure and readability make it easier to modify.

•The pseudocode allows programmers to work in different computer languages to talk to others they can be reviewed by groups easier than the real code.

**Disadvantages:**

•The main disadvantages are that it does not provide a visual representation of the programming logic.

•There are no accepted standards for writing the pseudo code. Programmers use their own styles of writing pseudo code.

•The pseudo code cannot be compiled nor executed and there is no real formative of a syntax of rules. It is simply one step, an important one, in producing the final code.

**References:**

<http://www.owlnet.rice.edu/~ceng303/manuals/fortran/FOR3_3.html>

<https://www.techwalla.com/articles/differences-between-psuedocode-and-flowcharts>

<https://eternalsunshineoftheismind.wordpress.com/2013/02/20/advantages-and-disadvantages-of-flowchart/comment-page-1/>

<https://www.buildprogrammer.com/advantages-and-disadvantages-of-pseudo-code/>