

MATLAB APPLICATION IN CIVIL ENGINEERING

**KTU Civil Engineering Department
Spring Semester**

Lecturer:

Prof. Dr. Erol ŞADOĞLU

CourseTasks

- ❑ to have improved programming abilities
- ❑ to have gained abilities to construct flowcharts/algorithms in advance using programming languages
- ❑ to have used MATLAB programming language for solving civil engineering problems

2. Week

Computers and Programming

A *computer* is an automatic device that performs calculations, making decisions, and has capacity for storing and processing vast amounts of information. A computer has two main parts: hardware and software.

Hardware (=Donanim)

Hardware is the electronic and mechanical parts of the computer.

Hardware includes:

Input Units: Keyboard, Mouse, Scanner

Process Units: CPU, Central Processing Unit. This coordinates the operation of computer system and performs arithmetic logic operations.

RAM, Random Access Memory

HDD, Hard Disc Driver

FDD, Floppy Disc Driver

CD-ROM, Compact Disc – Read Only Memory

Output Units: Monitor, Printer, Plotter, Scanner, Modem, Speaker

Software (=Yazılım)

The software consists of all the programs running on the computer. It includes:

Operating System (OS): is a program written by manufacturer (e.g. *Microsoft*). It interface between computer and user. All the programs run under the OS.

Examples are: *MS-DOS, Windows, Unix, Linux, BEOS.*

Compilers: can also be called translator. Every computer language has its own compiler. The compiler translates the statements of program written in a high level language into a low level language, the machine code.

Examples are: *Fortran, C, C++, Java, Pascal, Basic.*

Application Programs are programs written by the users for their own needs.

For example: *Word, Excel, Logo, AutoCAD, Flash.*

Different types of programming language

Fortran -- Formula Translation, mainly for scientific and engineering people.

C/C++ -- System software.

Pascal -- Mainly for teaching purposes.

COBOL -- Mainly for business and banking purposes.

MATLAB (MATlab LABoratory)--mainly for scientific and engineering people.

Problem Solving with Computer

Problem solving with computers involves several steps:

1. Clearly define the problem.
2. Analyze the problem and formulate a method to solve it
3. Describe the solution in the form of an algorithm.
4. Draw a flowchart of the algorithm.
5. Write the computer program.
6. If there is any syntax error, make the correction
8. Compile and run the program (debugging).
9. Test the program (debugging)
10. Interpretation of results.

If the answer is not what you want, you have a logic error in your program.

Make the necessary corrections!!!!

Algorithms

The algorithm gives a step-by-step description of the solution. This may be written in a non-formal language and structure.

Flowchart

The flowchart is a means of visually presenting the flow of data through an information processing systems, the operations performed within the system and the sequence in which they are performed. In this lesson, **we shall concern ourselves with the program flowchart**, which describes what operations (and in what sequence) are required to solve a given problem.

At the end of this lesson,
you will be able to understand😊

- **the meaning of flowchart**
- **the basic parts of the flowchart such as flowchart symbols and the flow lines connecting these symbols.**
- **the advantages of flowchart**

MEANING OF A FLOWCHART

- A flowchart is a diagrammatic representation that illustrates the sequence of operations to be performed to get the solution of a problem.
- Flowcharts are generally drawn in the early stages of formulating computer solutions.
- Flowcharts facilitate communication between programmers and business people.
- These flowcharts play a vital role in the programming of a problem and are quite helpful in understanding the logic of complicated and lengthy problems.
- Once the flowchart is drawn, it becomes easy to write the program in any high level language.

Often we see how flowcharts are helpful in explaining the program to others.

Hence, it is correct to say that a flowchart is

a must for the better documentation of a complex program.

Flowchart Symbols



An oval is used to indicate the beginning



A parallelogram indicates the input or output of information.



A rectangle indicates a computation, with the result of the computation assigned to a variable.



A diamond indicates a point where a decision is made.

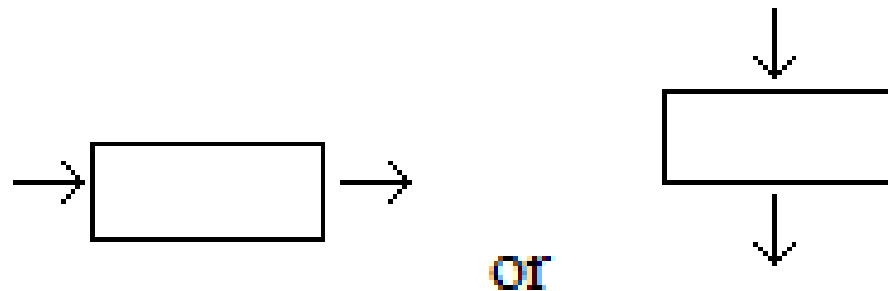


A hexagon indicates the beginning of the repetition structure.

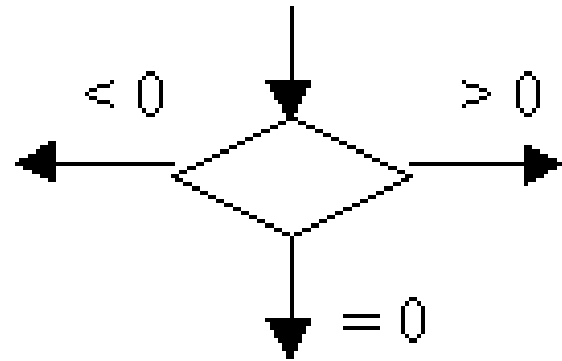
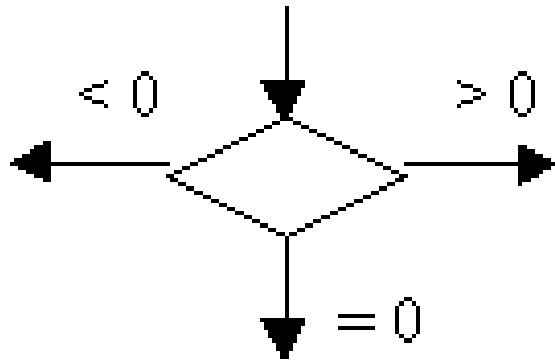


An arrow indicates the direction of flow of the algorithm. Circles with arrows connect the flowchart

- In drawing a proper flowchart, all necessary requirements should be listed out in logical order.
- The flowchart should be clear, and easy to follow.
- The usual direction of the flow of a procedure or system is from left to right or top to bottom.
- Only one flow line should come out from a process symbol.



- Only one flow line should enter a decision symbol, but two or three flow lines, one for each possible answer, should leave the decision symbol.



- If the flowchart becomes complex, it is better to use connector symbols to reduce the number of flow lines. Avoid the intersection of flow lines if you want to make it more effective and better way of communication.
- Ensure that the flowchart has a logical *start* and *finish*.
- It is useful to test the validity of the flowchart by passing through it with a simple test data.

ADVANTAGES OF USING FLOWCHARTS

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

Effective analysis: With the help of flowchart, problem can be analysed in more effective way.

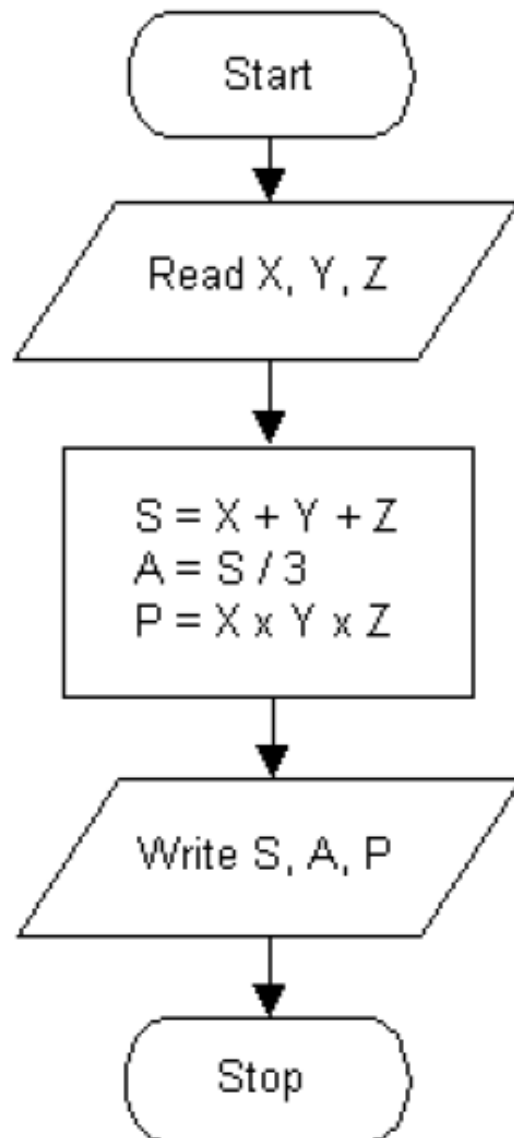
Proper documentation: Program flowcharts serve as a good program documentation, which is needed for various purposes.

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.

Flowchart



Pseudocode

Read X, Y, Z

Compute Sum (S) as $X + Y + Z$

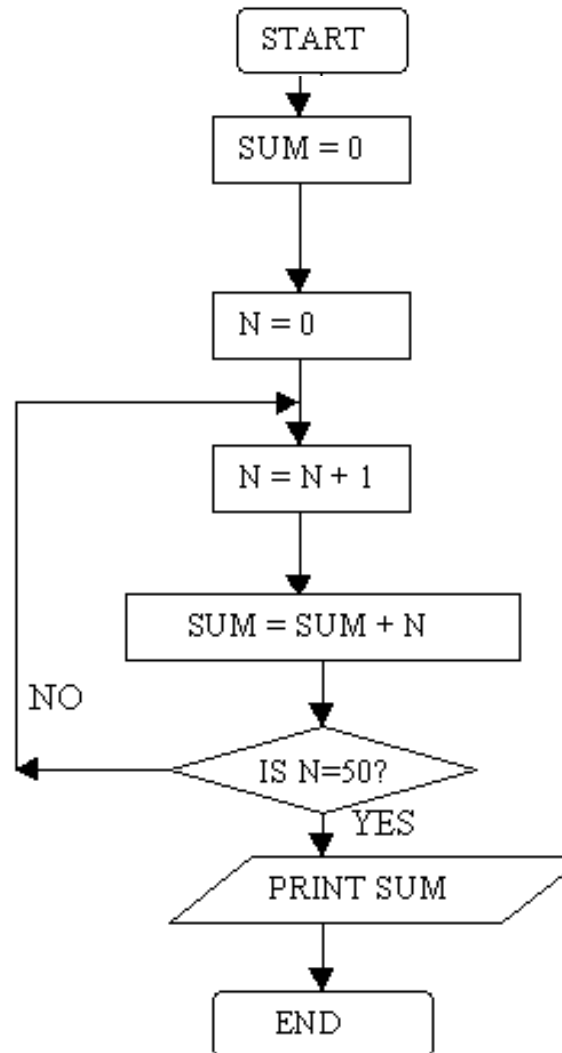
Compute Average (A) as $S / 3$

Compute Product (P) as $X \times Y \times Z$

Write (Display) the Sum, Average and Product

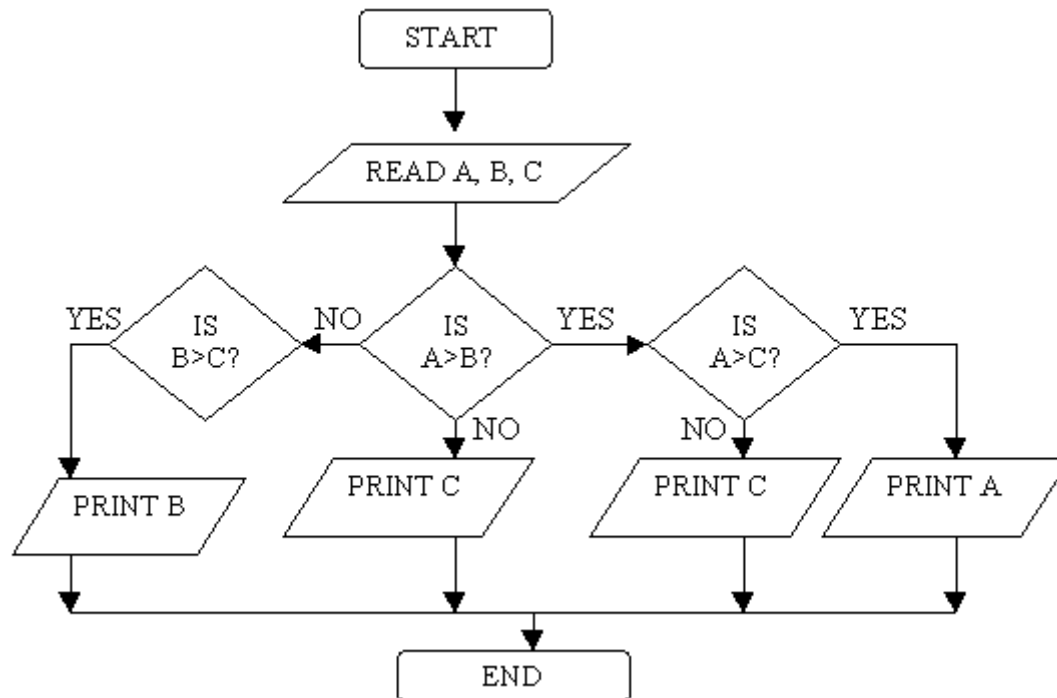
Example 1

Draw a flowchart to find the sum of first 50 natural numbers.



Example 2

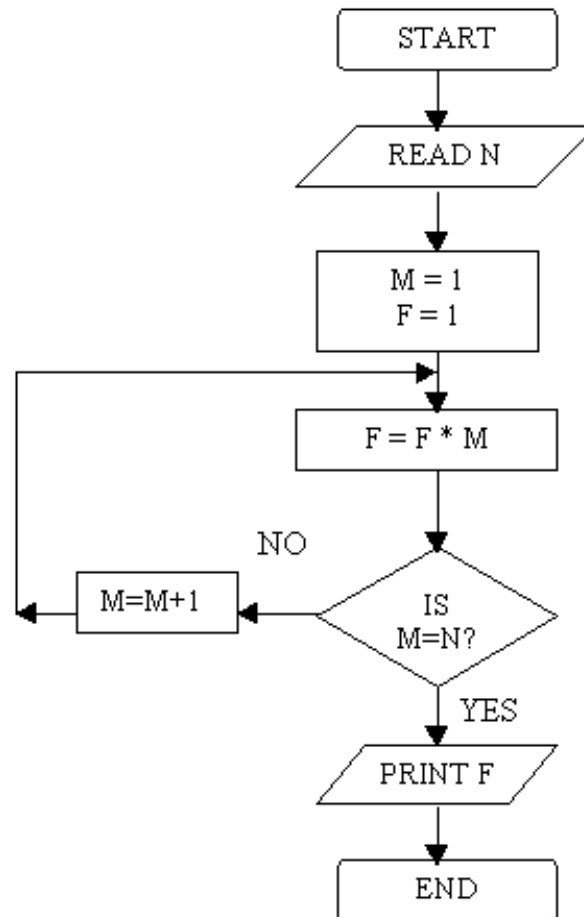
Draw a flowchart to find the largest of three numbers A, B, and C.



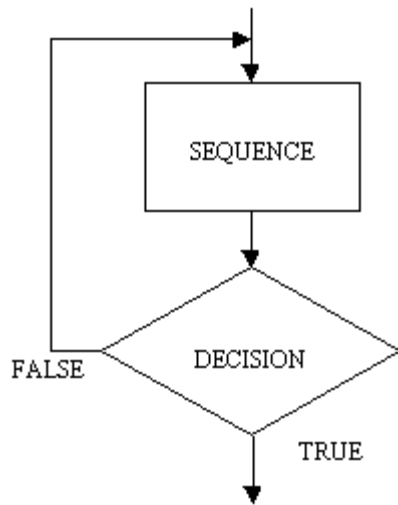
Example 3

Draw a flowchart for computing factorial N ($N!$) where

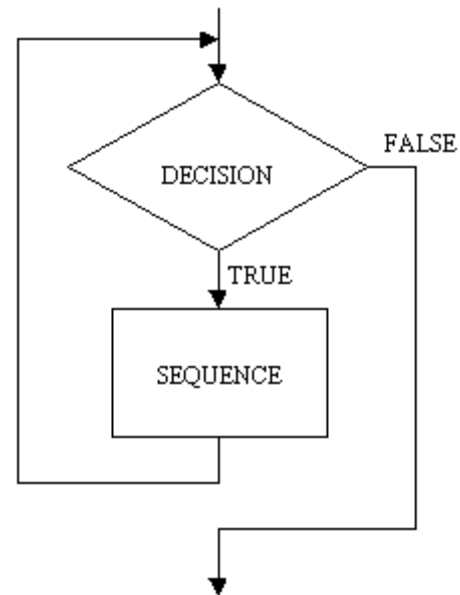
$$N! = 1 \times 2 \times 3 \times \dots \times N.$$



Repeat loop



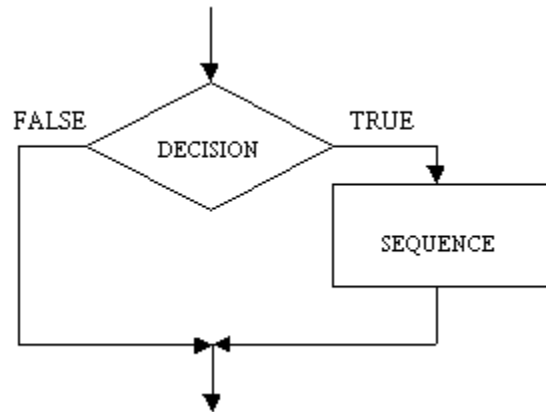
While loop



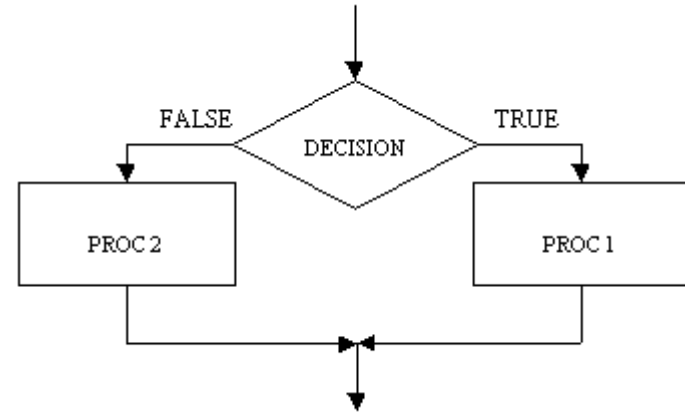
The while loop is basically the reverse of the repeat loop, the decision comes first, followed by the process

IF ... THEN

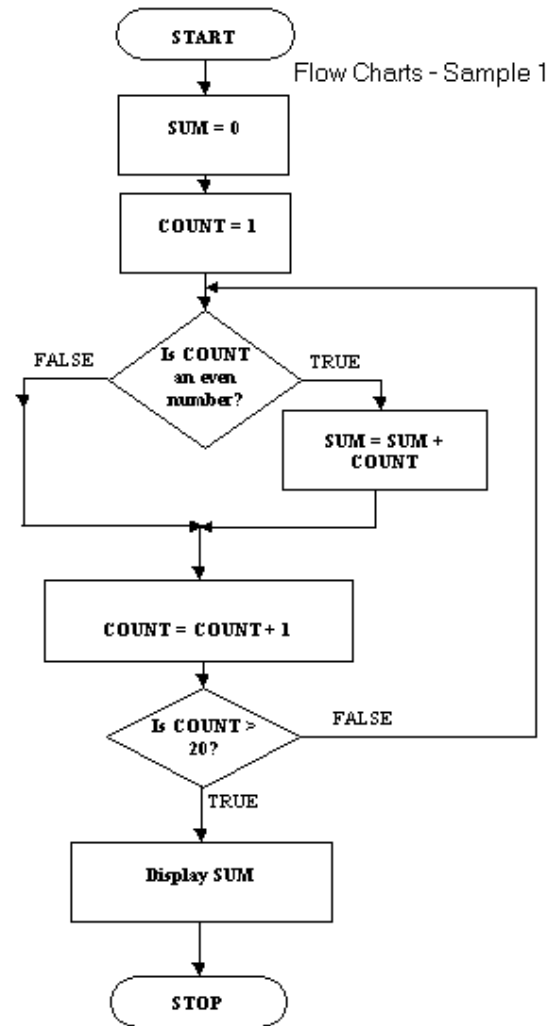
also known as the NULL ELSE,
meaning that there is no ELSE part



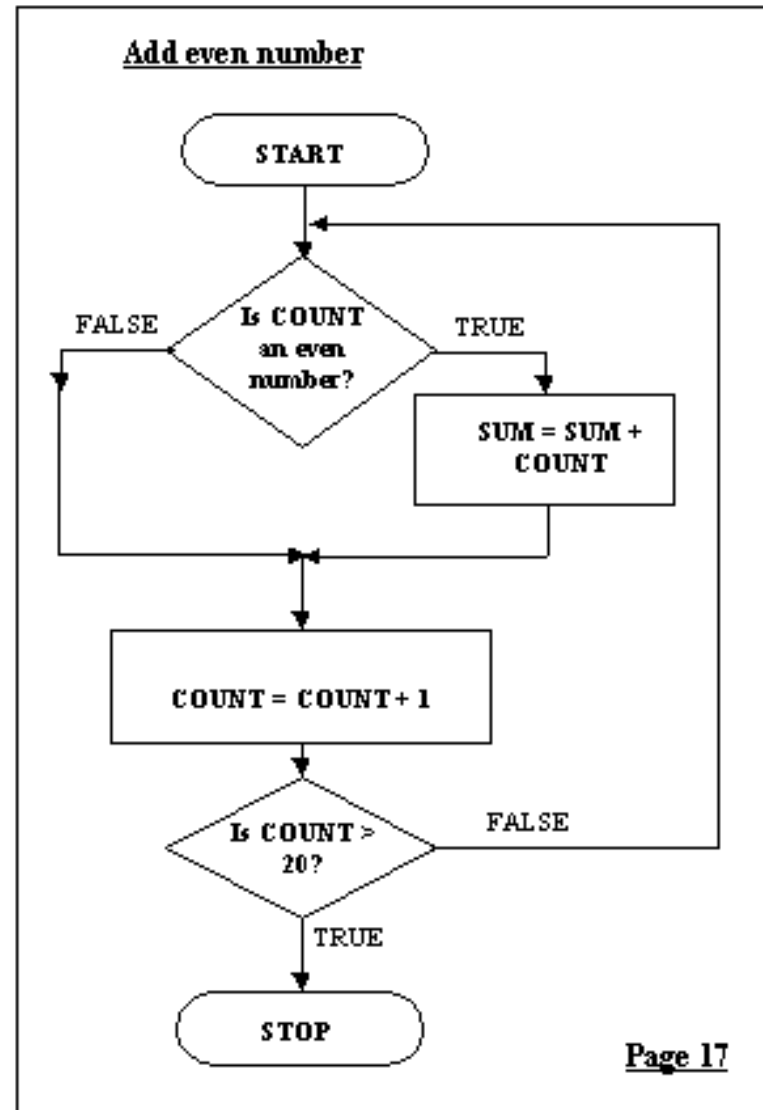
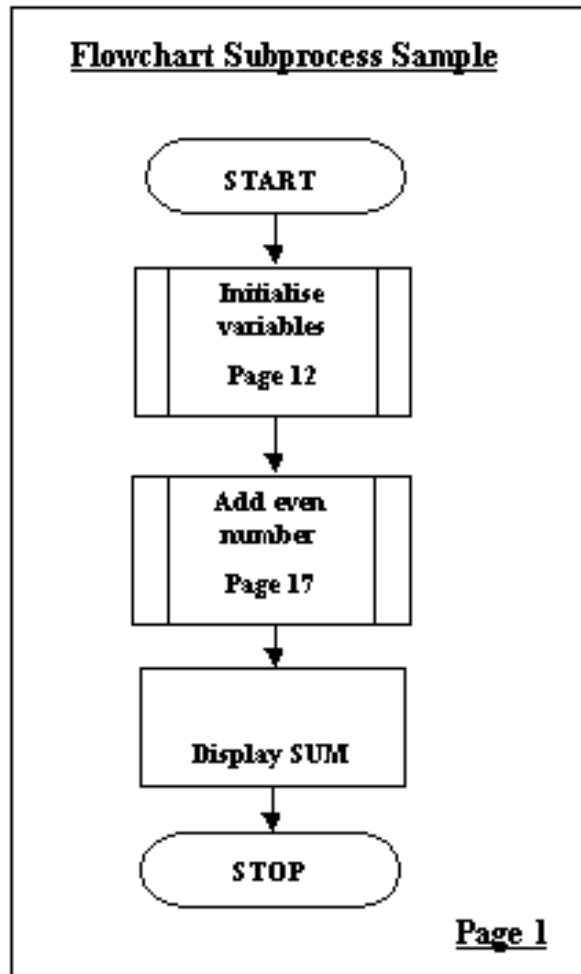
IF ... THEN ... ELSE



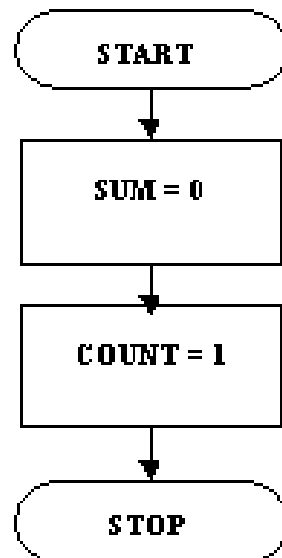
The algorithm sums all the even numbers between 1 and 20 inclusive and then displays the sum. It uses a repeat loop and contains a null else within the repeat loop.



an example of the use of subprocesses in flow charts

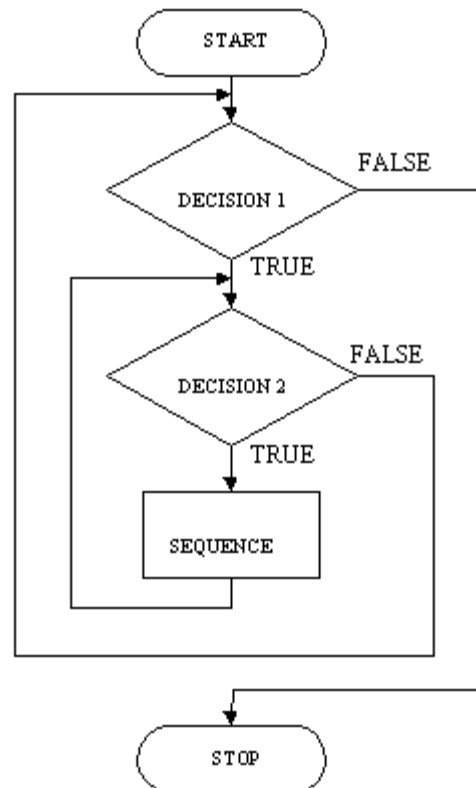


Initialise variables

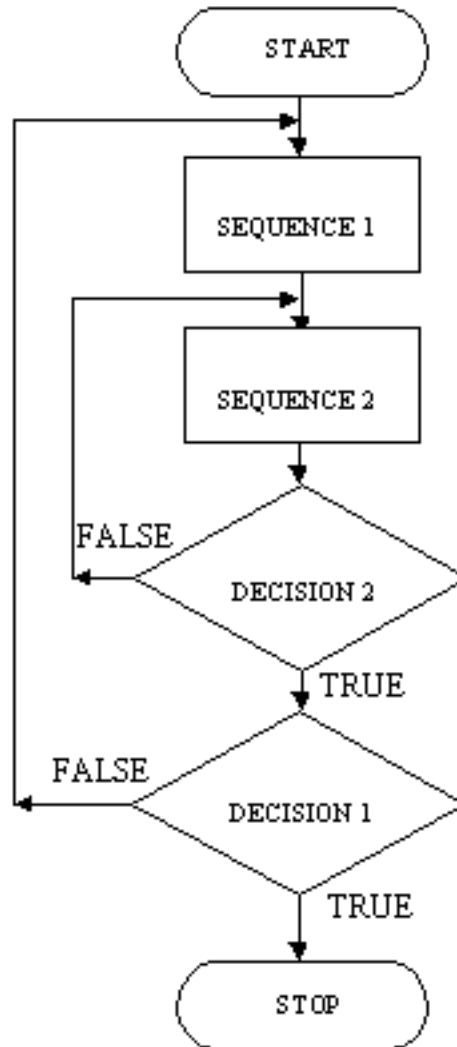


Using nested loops in flow charts

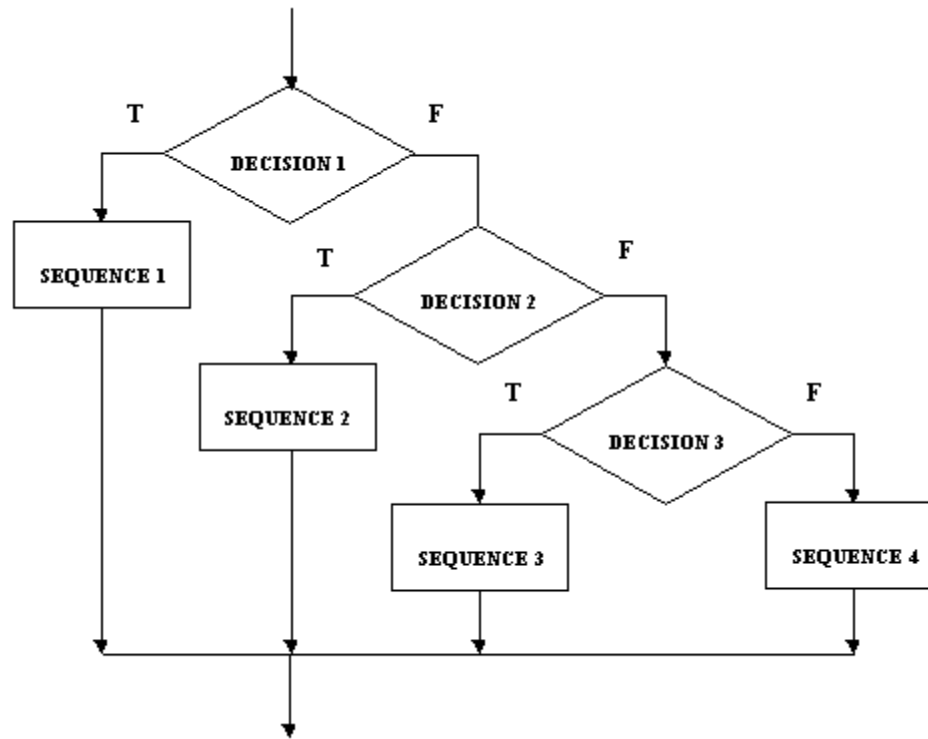
Each single step through the outer loop will lead to the complete iteration of the inner loop. Assume that the outer loop counts through 10 steps and the inner loop through 100 steps. The sequence in the inner loop will be executed $10 * 100$ times. Nested loops will do a lot of work



Like the while loop the nested repeat loop will see a great deal of work done.
If the outer loop does a thousand iterations and the inner loops does a thousand iterations then sequence 2 will be executed $1000 * 1000$ times.



Using multiway selection in flow charts



If decision 1 is true then sequence 1 is executed and the multiway selection is finished. If decision 1 is false then decision 2 is tested, if this is true then sequence 2 is done and the multiway selection is finished. If decision 2 is false, you get the picture.