Flow chart

A flow chart is a graphical or symbolic representation of a process. Each step in the process is represented by a different symbol and contains a short description of the process step. The flow chart symbols are linked together with arrows showing the process flow direction. This diagrammatic representation illustrates a solution to a given problem, was developed by Herman Goldstein and John von Neumann in the 1940's.

Flowcharts are used in analyzing, designing, documenting or managing a process or program in various fields. Flow charts are used in designing and documenting complex processes or programs. They help visualize what is going on and thereby help the viewer to understand a process.

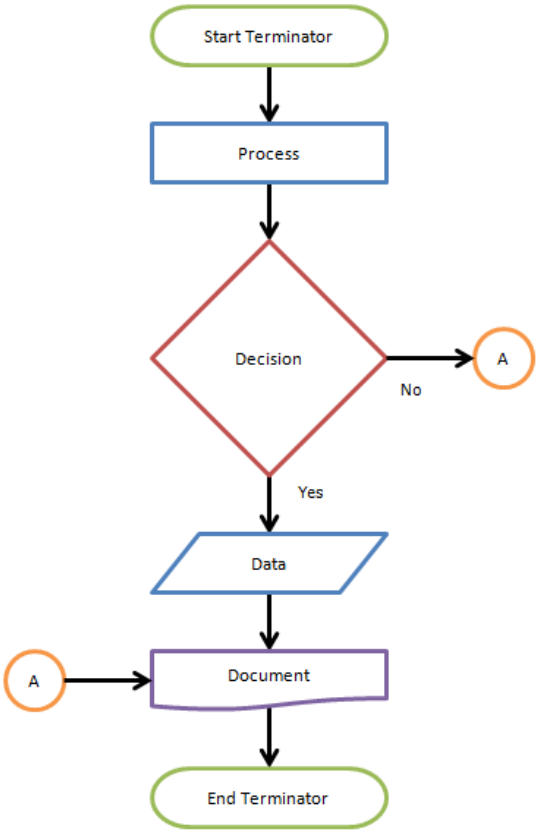
A flow chart can therefore be used to:

- Define and analyze processes.
- Build a step-by-step picture of the process for analysis, discussion, or communication.
- Define, standardize or find areas for improvement in a process.

SYMBOLS USED IN FLOW CHART:

Different flow chart symbols have different meanings. The most common flow chart symbols are:

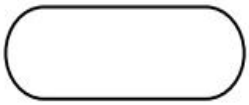
- **Terminator**: An oval flow chart shape indicating the start or end of the process.
- **Process**: A rectangular flow chart shape indicating a normal process flow step.
- **Decision**: A diamond flow chart shape indication a branch in the process flow.
- **Connector**: A small, labeled, circular flow chart shape used to indicate a jump in the process flow. (Shown as the circle with the letter “A”, below.)
- **Data**: A parallelogram that indicates data input or output (I/O) for a process.
- **Document**: Used to indicate a document or report



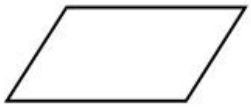
Flowchart

- *Flowchart* is a pictorial representation of an algorithm
- Uses symbols (boxes of different shapes) that have standardized meanings to denote different types of instructions
- Actual instructions are written within the boxes
- Boxes are connected by solid lines having arrow marks to indicate the exact sequence in which the instructions are to be executed
- Process of drawing a flowchart for an algorithm is called *flowcharting*

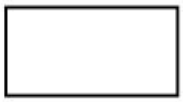
Basic Flowchart Symbols



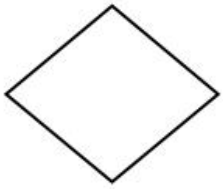
Terminal



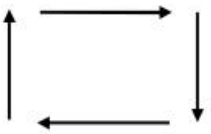
Input/Output



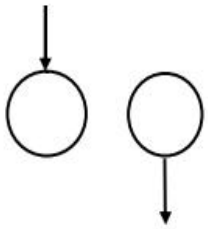
Processing



Decision

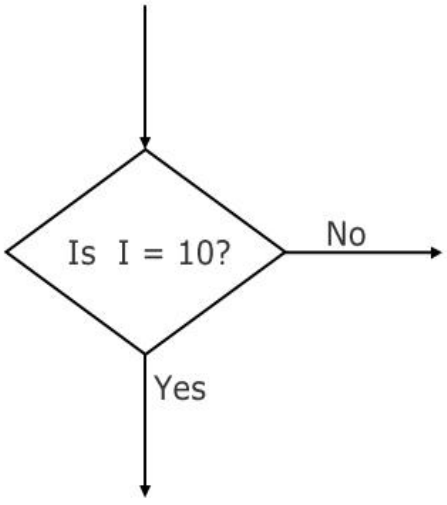


Flow lines

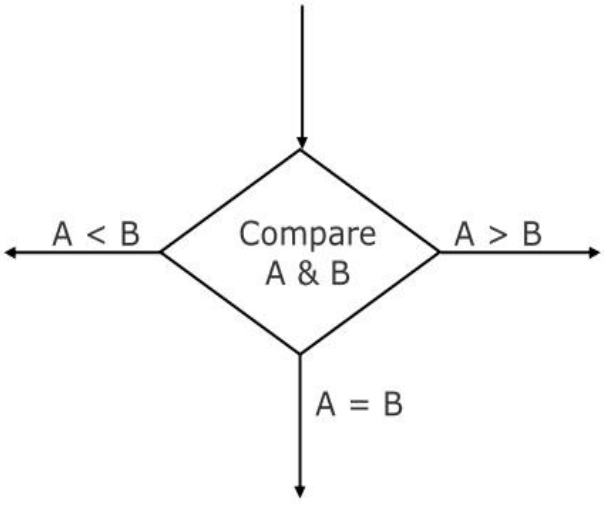


Connectors

Examples of Decision Symbol

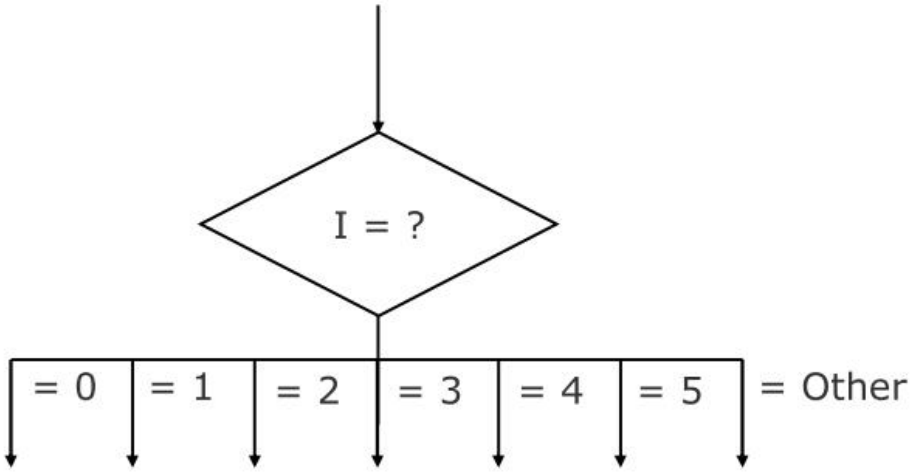


(a) A two-way branch decision.



(b) A three-way branch decision.

Examples of Decision Symbol



(c) A multiple-way branch decision.

Sample Flowchart (Example 3)

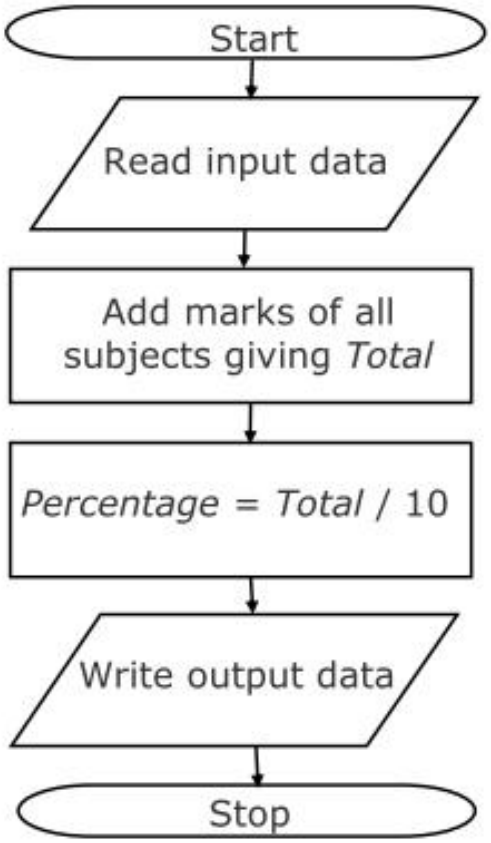
A student appears in an examination, which consists of total 10 subjects, each subject having maximum marks of 100.

The roll number of the student, his/her name, and the marks obtained by him/her in various subjects are supplied as input data.

Such a collection of related data items, which is treated as a unit is known as a record.

Draw a flowchart for the algorithm to calculate the percentage marks obtained by the student in this examination and then to print it along with his/her roll number and name.

Sample Flowchart (Example 3)

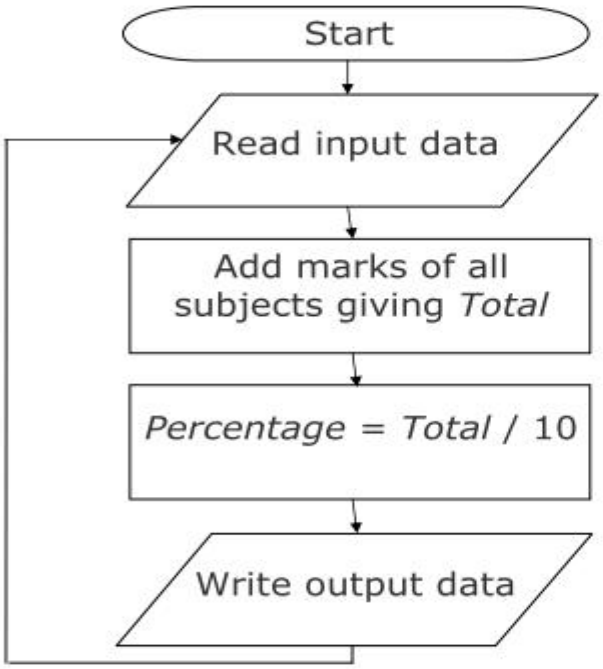


Sample Flowchart (Example 4)

50 students of a class appear in the examination of Example 3.

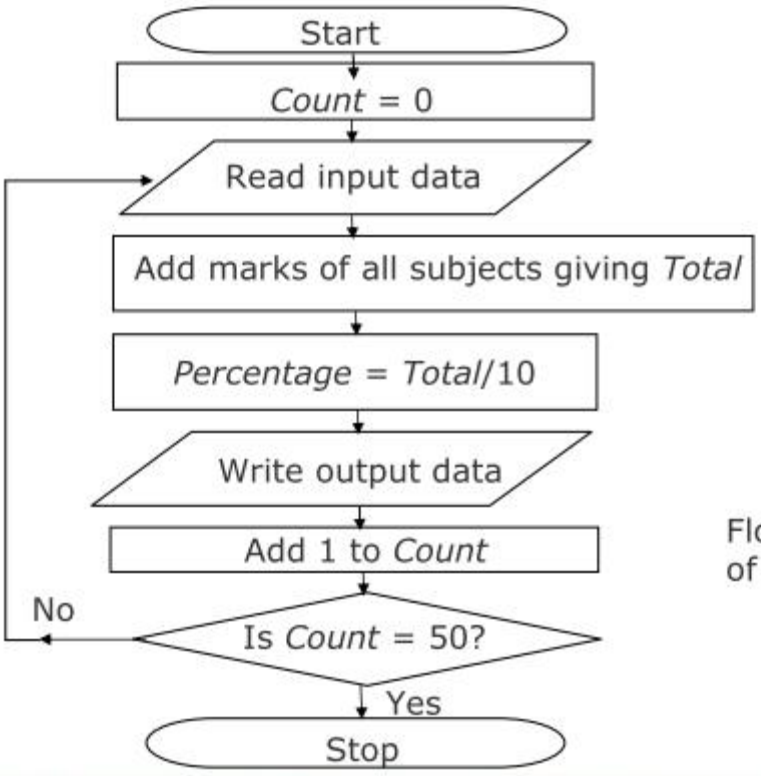
Draw a flowchart for the algorithm to calculate and print the percentage marks obtained by each student along with his/her roll number and name.

Sample Flowchart (Example 4)



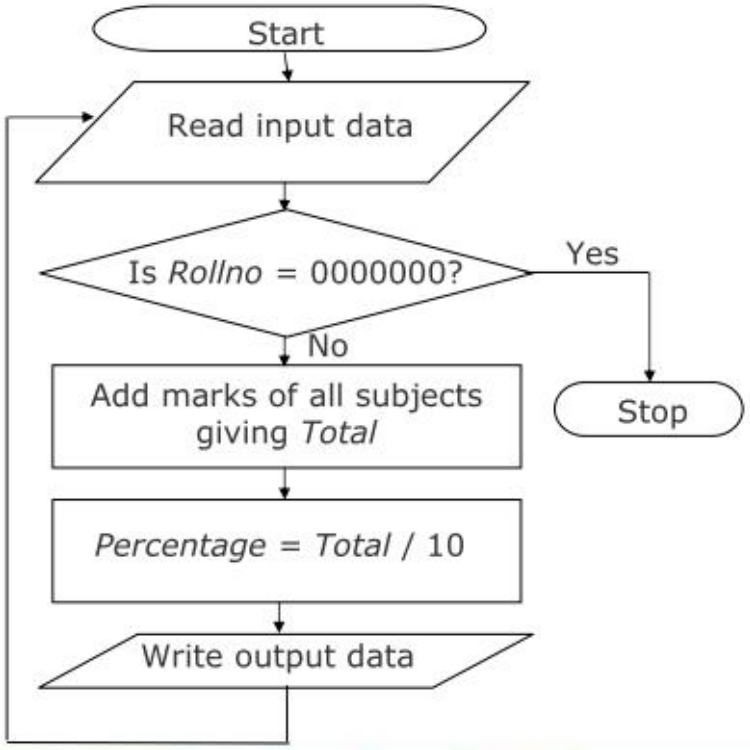
Flowchart for the solution of Example 4 with an infinite (endless) process loop.

Sample Flowchart (Example 4)



Flowchart for the solution of Example 4.

Sample Flowchart (Example 4)



Generalized flowchart for the solution of Example 4 using the concept of **trailer record**. Here the process loop is terminated by detecting a special non-data record.

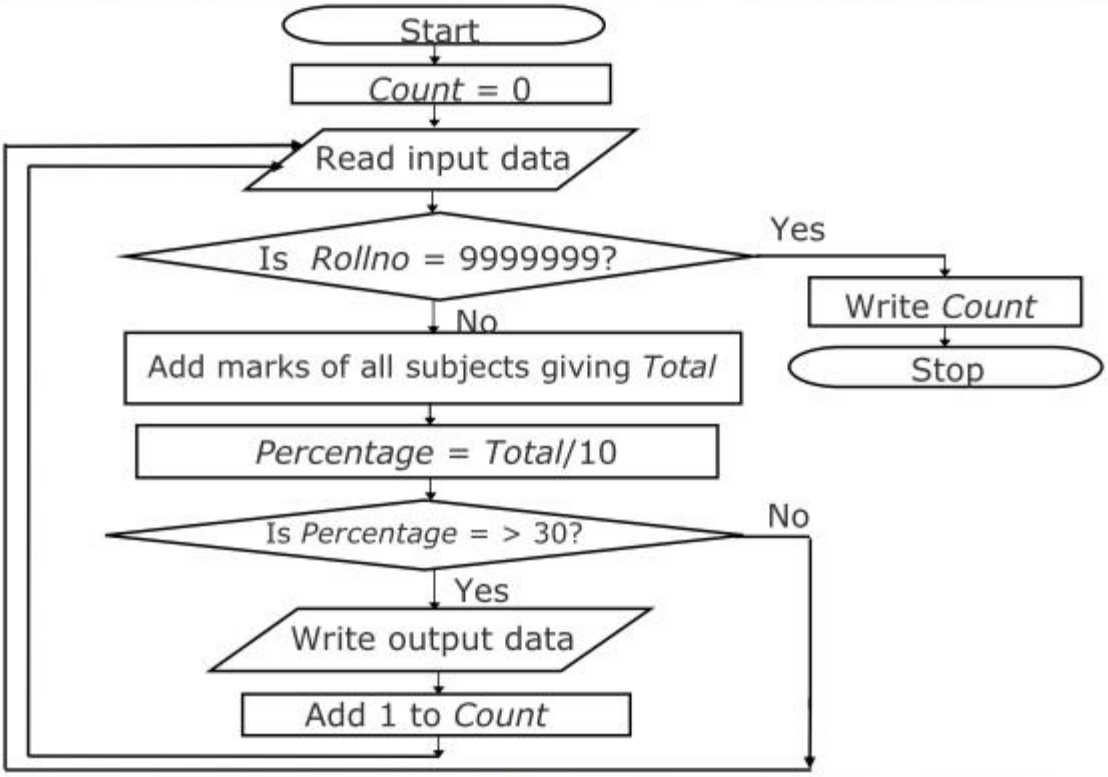
Sample Flowchart (Example 5)

For the examination of Example 3, we want to make a list of only those students who have passed (obtained 30% or more marks) in the examination.

In the end, we also want to print out the total number of students who have passed.

Assuming that the input data of all the students is terminated by a trailer record, which has sentinel value of 9999999 for Rollno, draw a flowchart for the algorithm to do this.

Sample Flowchart (Example 5)



Sample Flowchart (Example 6)

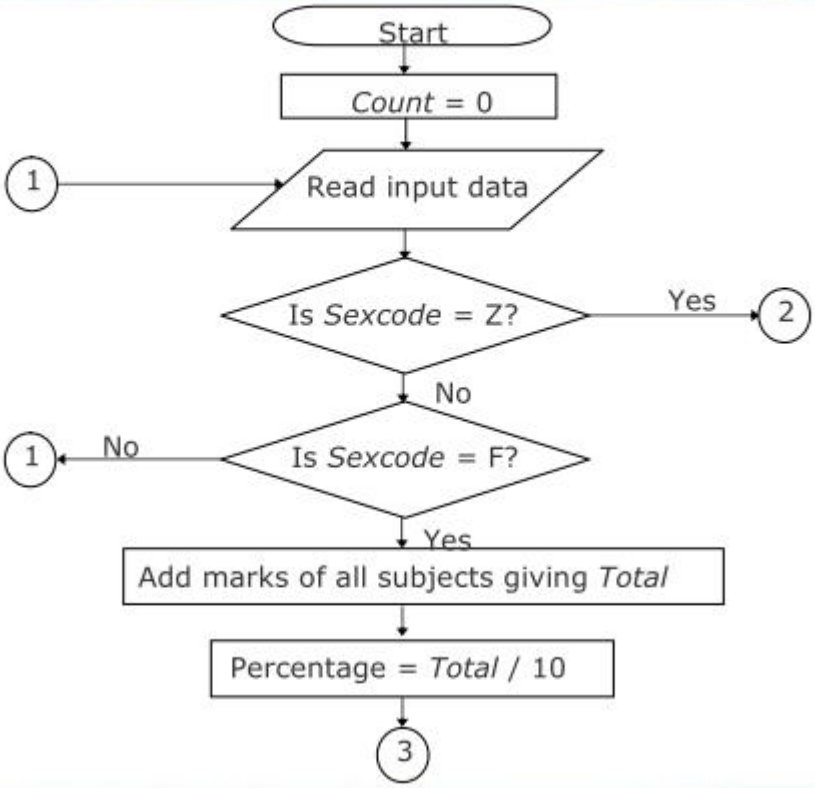
Suppose the input data of each student for the examination of Example 3 also contains information regarding the sex of the candidate in the field named *Sexcode* having values M (for male) or F (for female).

We want to make a list of only those female students who have passed in second division (obtained 45% or more but less than 60% marks).

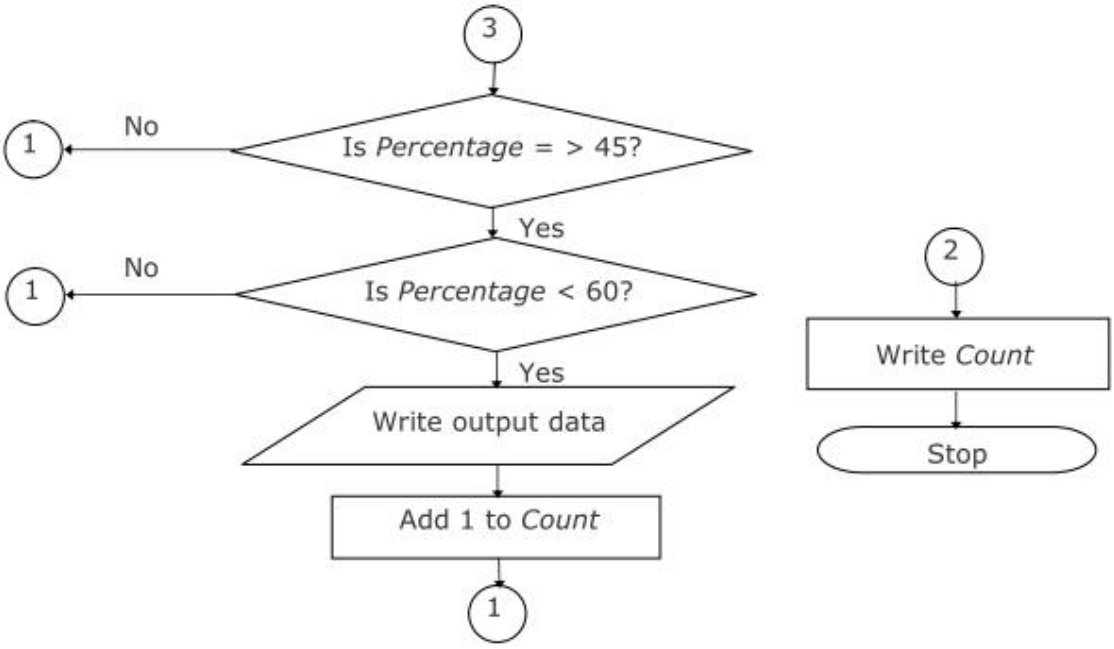
In the end, we also want to print out the total number of such students.

Assuming that the input data of all the students is terminated by a trailer record, which has a sentinel value of Z for Sexcode, draw a flowchart for the algorithm to do this.

Sample Flowchart (Example 6)



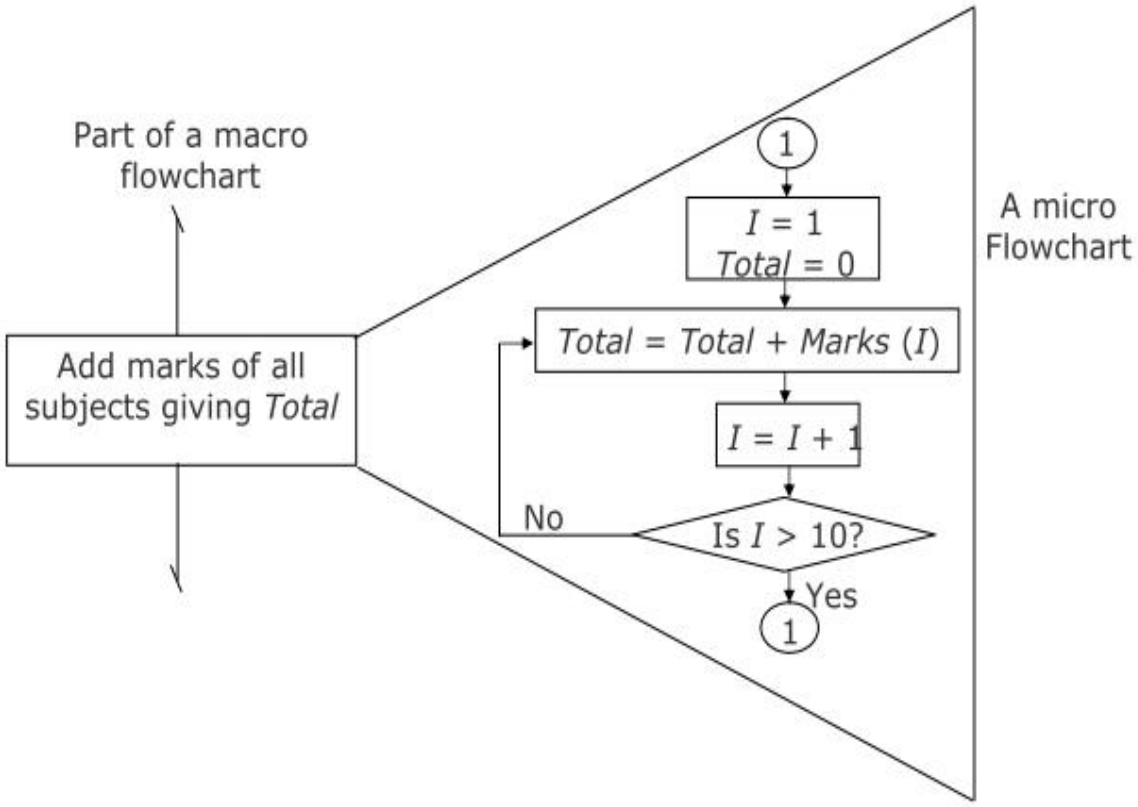
Sample Flowchart (Example 4)



Levels of Flowchart

- Flowchart that outlines the main segments of a program or that shows less details is a ***macro flowchart***
- Flowchart with more details is a ***micro flowchart***, or detailed flowchart
- There are no set standards on the amount of details that should be provided in a flowchart

Example of Micro Flowchart



Flowcharting Rules

- First chart the main line of logic, then incorporate detail
- Maintain a consistent level of detail for a given flowchart
- Do not chart every detail of the program. A reader who is interested in greater details can refer to the program itself
- Words in the flowchart symbols should be common statements and easy to understand

Flowcharting Rules

- Be consistent in using names and variables in the flowchart
- Go from left to right and top to bottom in constructing flowcharts
- Keep the flowchart as simple as possible. Crossing of flow lines should be avoided as far as practicable
- If a new flowcharting page is needed, it is recommended that the flowchart be broken at an input or output point.
- Properly labeled connectors should be used to link the portions of the flowchart on different pages

Advantages of Flowchart

- Better Communication
- Proper program documentation
- Efficient coding
- Systematic debugging
- Systematic testing

Limitations of Flowchart

- Flowcharts are very time consuming and laborious to draw (especially for large complex programs)
- Redrawing a flowchart for incorporating changes/ modifications is a tedious task
- There are no standards determining the amount of detail that should be included in a flowchart

Exercise:

Draw the flow chart to the following:

1. To multiply to numbers.
2. To divide two numbers.
3. To check whether the given number is even or odd.
4. To swap two numbers.
5. To check whether the given number is lesser than 10 or not.
6. To convert Fahrenheit to Celsius.
7. To check the greater number in the given two numbers.
8. To calculate simple interest.
9. To check whether the given number is prime or not.
10. To calculate area of the given rectangle.