Flow Chart

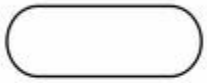
The flowchart is a diagrammatic representation of a sequence of logical steps of a program. Flowcharts use simple geometric shapes to depict processes and arrows to show relationships and process/data flow.

Flowchart Symbols

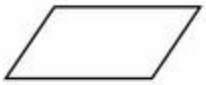
Here is a chart for some of the common symbols used in drawing flowcharts.

Symbol	Symbol Name	Purpose
	Start/Stop	Used at the beginning and end of the algorithm to show start and end of the program.
	Process	Indicates processes like mathematical operations.
	Input/ Output	Used for denoting program inputs and outputs.
	Decision	Stands for decision statements in a program, where answer is usually Yes or No.
	Arrow	Shows relationships between different shapes.
	On-page Connector	Connects two or more parts of a flowchart, which are on the same page.
	Off-page Connector	Connects two parts of a flowchart that are spread over different pages.

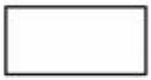
Basic Flowchart Symbols



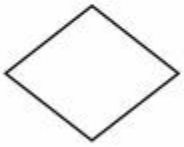
Terminal



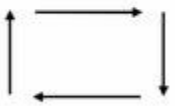
Input/Output



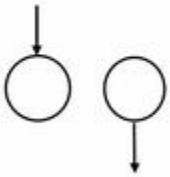
Processing



Decision

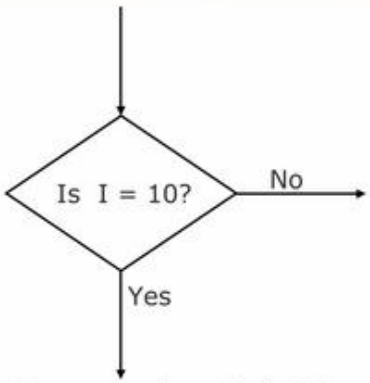


Flow lines

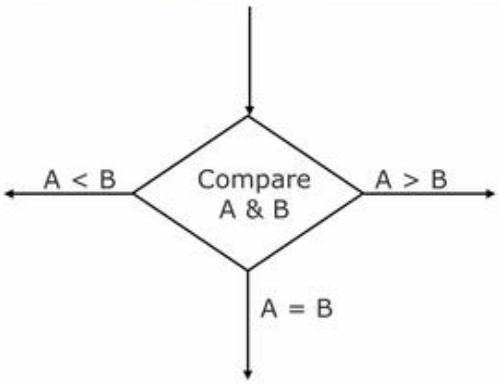


Connectors

Examples of Decision Symbol



(a) A two-way branch decision.



(b) A three-way branch decision.

Sample Flowchart (Example 3)

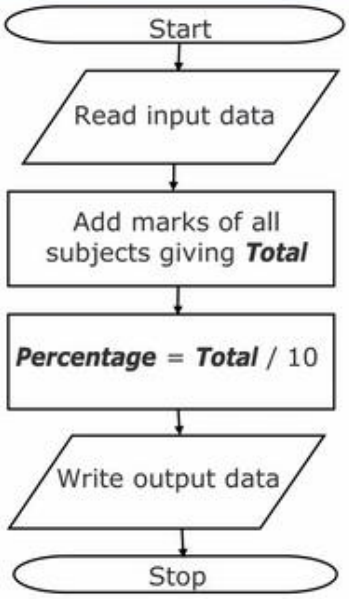
A student appears in an examination, which consists of total 10 subjects, each subject having maximum marks of 100.

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)



(contd...)

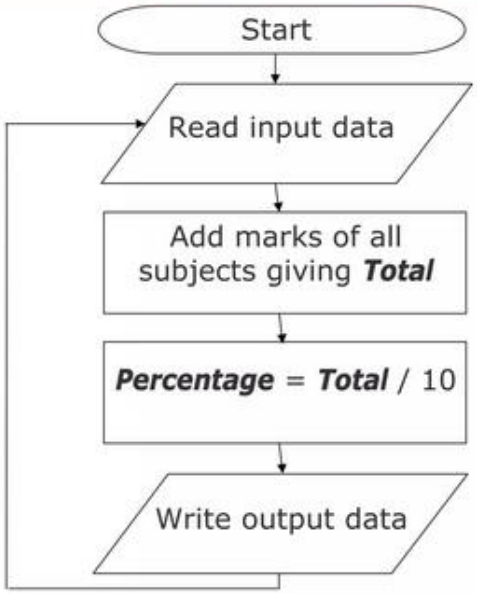
Sample Flowchart (Example 4)

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)

(contd...)

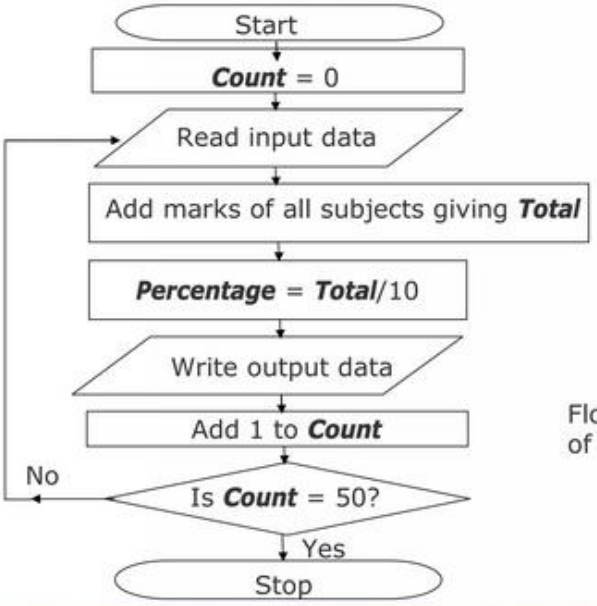


Flowchart for the solution of Example 4 with an infinite (endless) process loop.

Computer Fundamentals: Pradeep K. Sinha & Priti Sinha

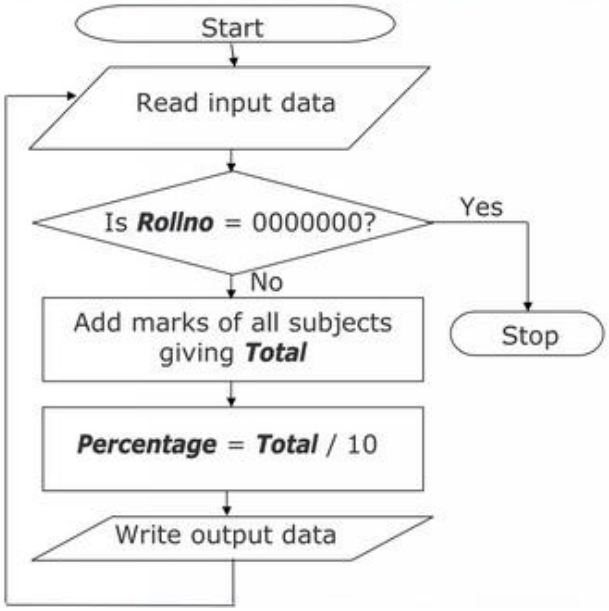
Sample Flowchart (Example 4)

(contd...)



Flowchart for the solution of Example 4.

Sample Flowchart (Example 4)



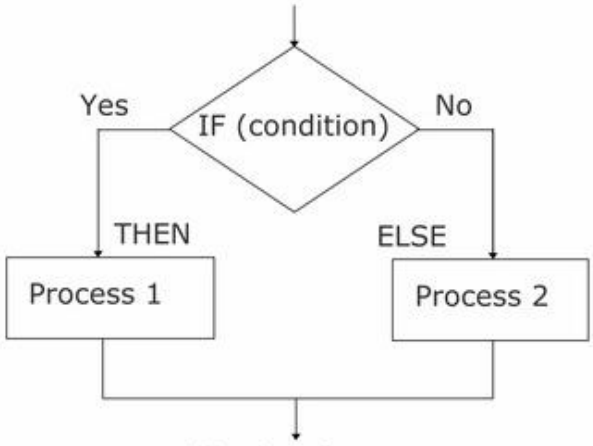
(contd...)

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.

Selection Logic

- Also known as decision logic, it is used for making decisions
- Three popularly used selection logic structures are
  1. IF...THEN...ELSE
  2. IF...THEN
  3. CASE

Selection Logic (IF...THEN...ELSE Structure)

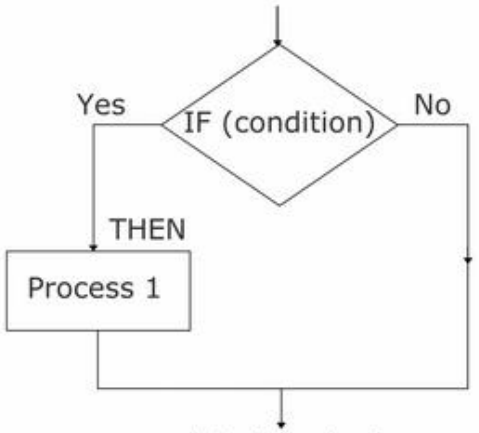


(a) Flowchart

```
:
IF Condition
    THEN    Process 1
    ELSE    Process 2
ENDIF
:
```

(b) Pseudocode

Selection Logic (IF...THEN Structure)

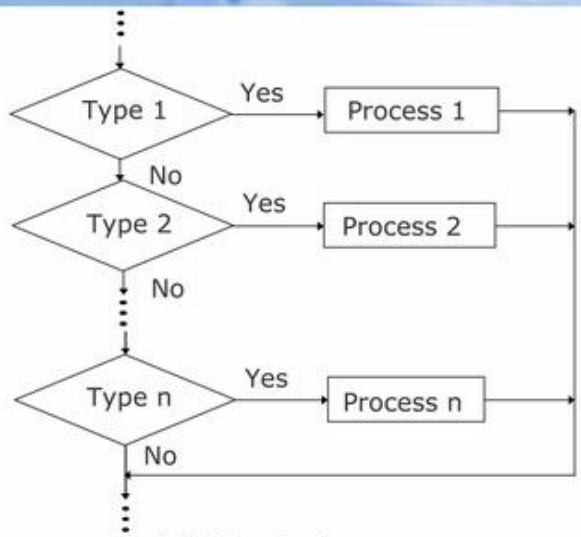


(a) Flowchart

```
:
IF Condition
    THEN    Process 1
ENDIF
:
```

(b) Pseudocode

Selection Logic (CASE Structure)



(a) Flowchart

```
CASE Type
    Case Type 1:    Process 1
    Case Type 2:    Process 2
    :
    Case Type n:    Process n
ENDCASE
```

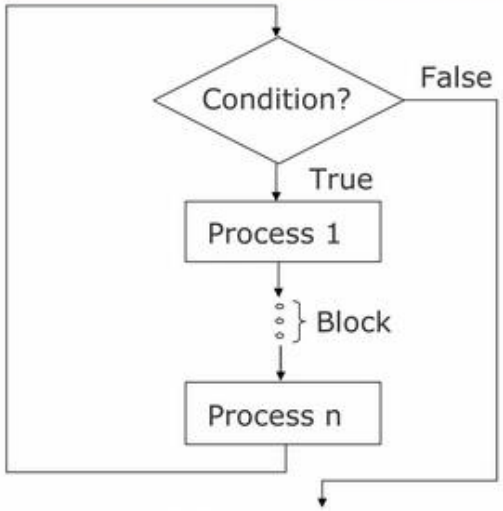
(b) Pseudocode

Computer Fundamentals: Pradeep K. Sinha & Priti Sinha

Iteration (or Looping) Logic

- Used to produce loops in program logic when one or more instructions may be executed several times depending on some conditions
- Two popularly used iteration logic structures are
  1. DO…WHILE
  2. REPEAT…UNTIL

Iteration (or Looping) Logic
(DO...WHILE Structure)

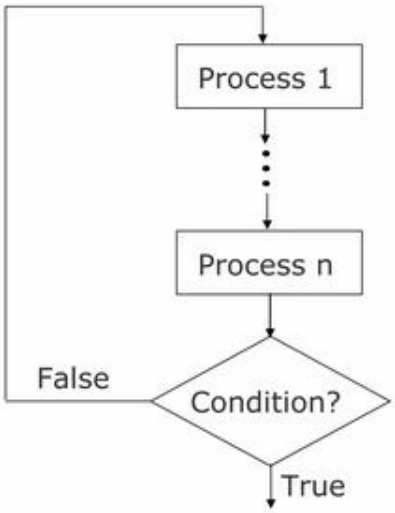


(a) Flowchart

```
:
DO WHILE Condition
    Process 1
    :
    Process n
ENDDO
:
```

(b) Pseudocode

Iteration (or Looping) Logic
(REPEAT...UNTIL Structure)



(a) Flowchart

```
:
REPEAT
    Process 1
    :
    Process n
UNTIL Condition
:
```

(b) Pseudocode

```
#include<stdio.h>
int main()
{
 int number;

 printf("Enter number:");
 scanf("%d",&number);

 if(number%2 == 0)
  printf("Entered number is even.");
 else
  printf("Entered number is odd.");

 return 0;
}
```

Output of the program:

```
Enter number:51
Entered number is odd.
```

Following flowchart will read a number from user. This number is checked using % operator to find whether it is odd or even.

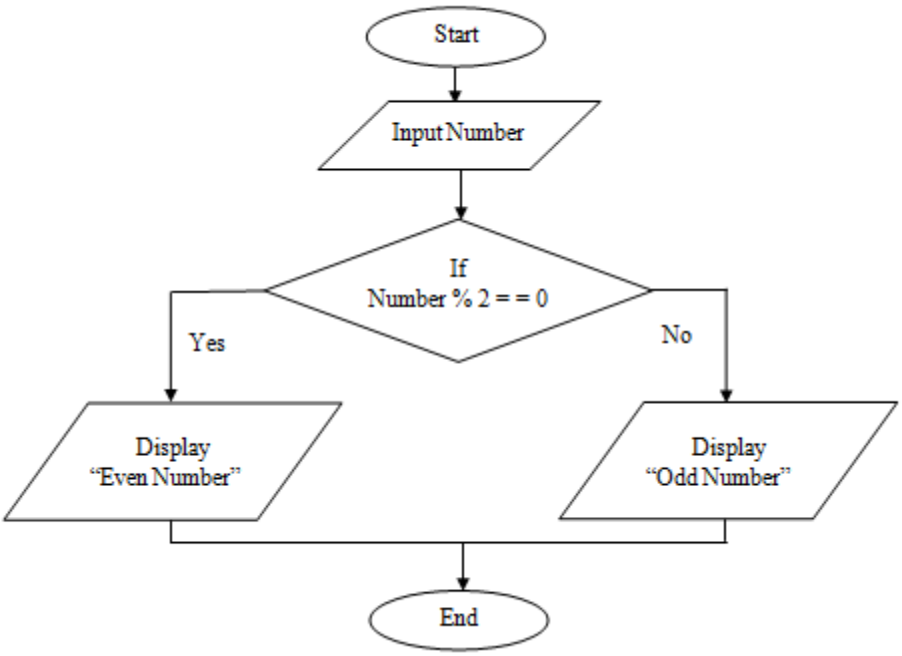


Figure: Flowchart to check Odd or Even Number

This C program will convert given uppercase letters into lover case.

```c
#include<stdio.h>
#include<string.h>
int main()
{
  char str1[10]="ABCD";
  int i;
  for(i=0; i<4; i++)
  {
    printf("%c",str1[i]+32);
  }
  return 0;
}
```

Output of the program:

abcd

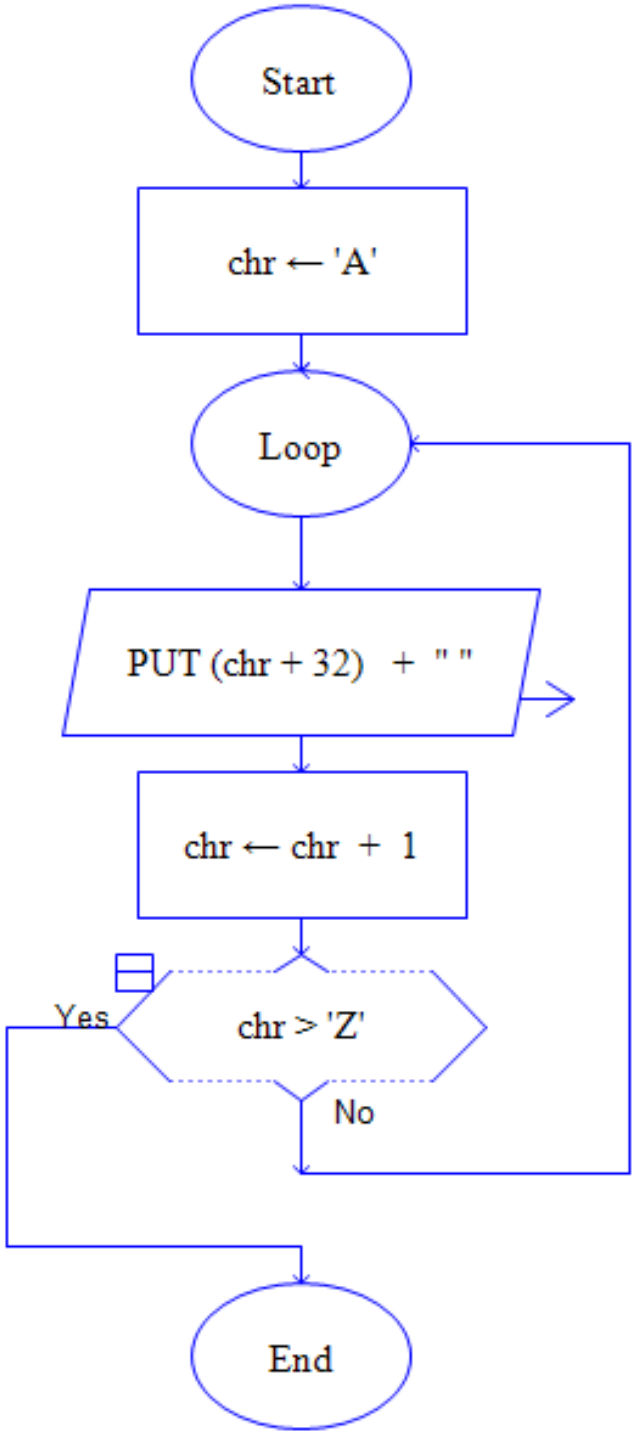


Fig.: Flowchart to convert Uppercase letters into Lowercase letters

Note:

- ASCII value of 'A' is 65, and 'a' is 97.

- If we add 32 into 65, we get 97 which is ASCII value of 'a'.
- Hence we add 32 into 'chr' variable in the flowchart. This process we repeat up to 'Z' in the flowchart using loop.

Flowchart to find Largest of Three Numbers

Following flowchart is used to find the largest number from the given three numbers.

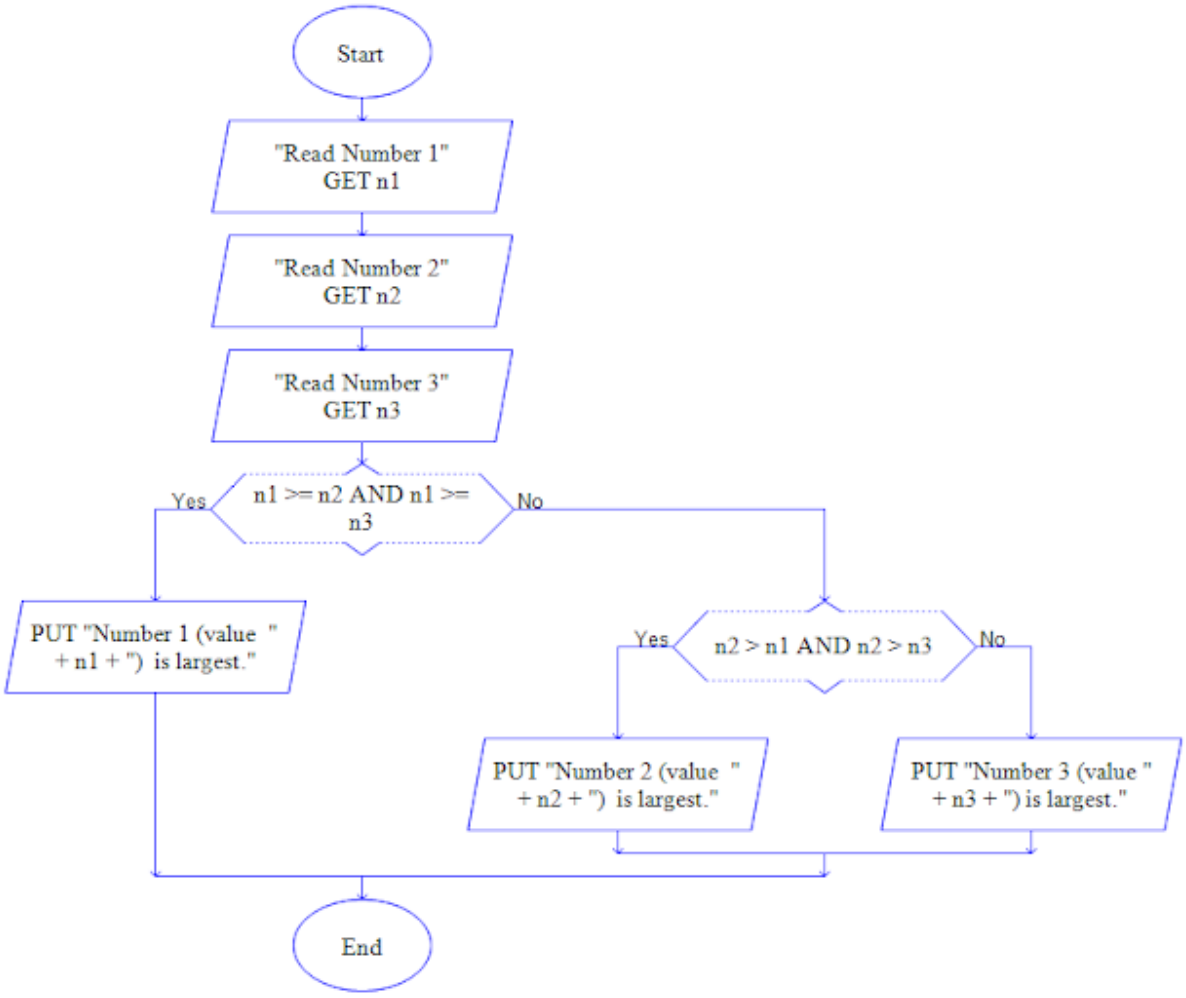


Fig.: Flowchart to find largest of given three numbers.