Flowchart

Mrs. Shinde Manjusha Anant
Department of Information Technology
Rajarshi Shahu Mahavidyalaya(Autonomous),Latur

LearnProgramCode

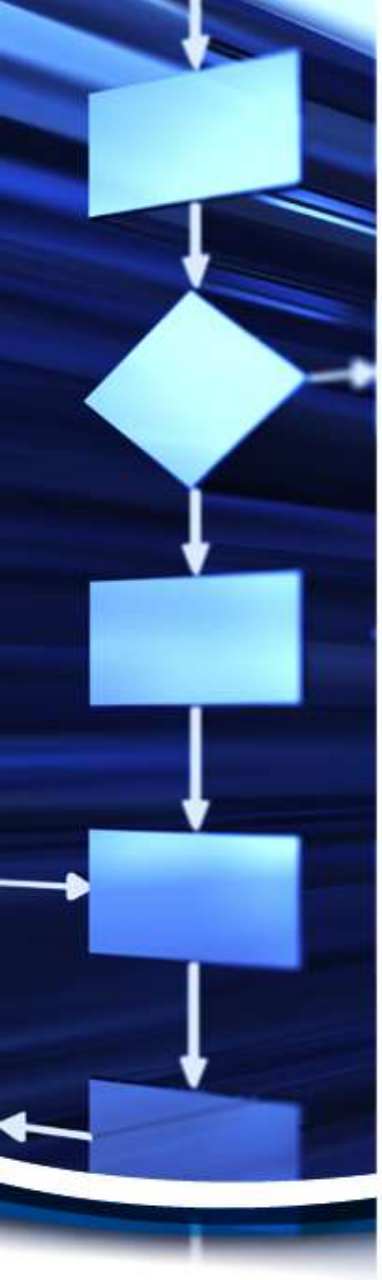
What is Flowchart?

- The graphical or the pictorial form of an algorithm is called the Flowchart.
- A flowchart is a **graphical representation of the problem** solving process.

The definition of flowchart can be divided into the following two parts:

- **Flow -** It is a representation of a series of logic operations to satisfy specific requirements. It can be irregular, naturally, unfixed or full of problems.

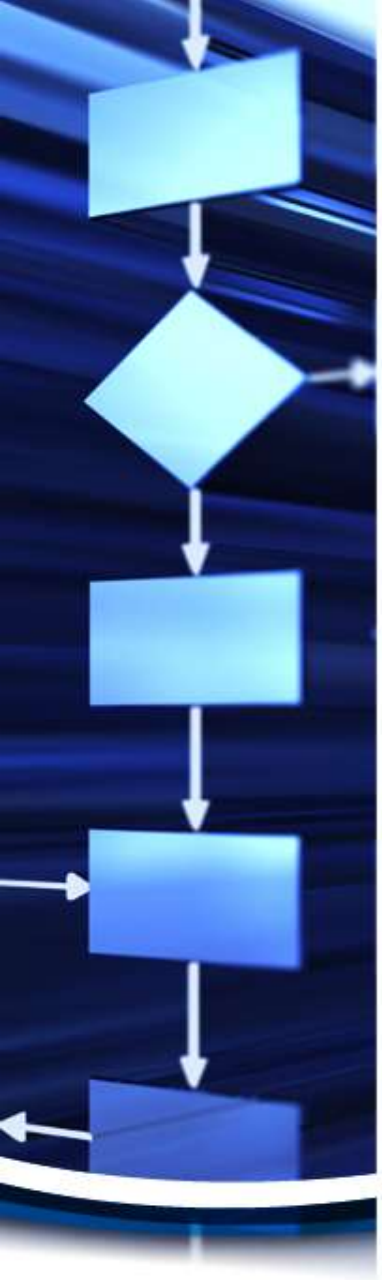
- **Chart -** It is a presentation and a written description of some regular , common parts of the flow.



HISTORY OF FLOWCHARTS

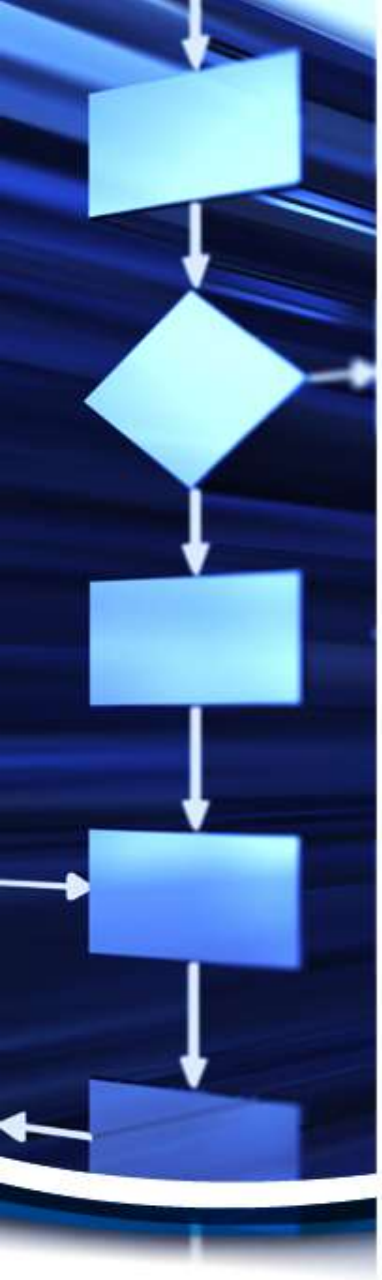
- It is not clear who was the true inventor of flowcharts, but the first standardized documentation on flow chart was first introduced by Frank and Lillian Gilbreth in 1921. Both presented the graphic-based method in a presentation titled: "Process Charts: First Steps in Finding the One Best Way to do Work", to members of the American Society of Mechanical Engineers (ASME).
- In 1949, flowchart to be used for planning computer programs and quickly became one of the most popular tools in designing computer algorithms and programs.
- Nowadays, flow chart is an important productivity tool, serving employees in various documentation and functions.

{ C }
LearnProgramCode



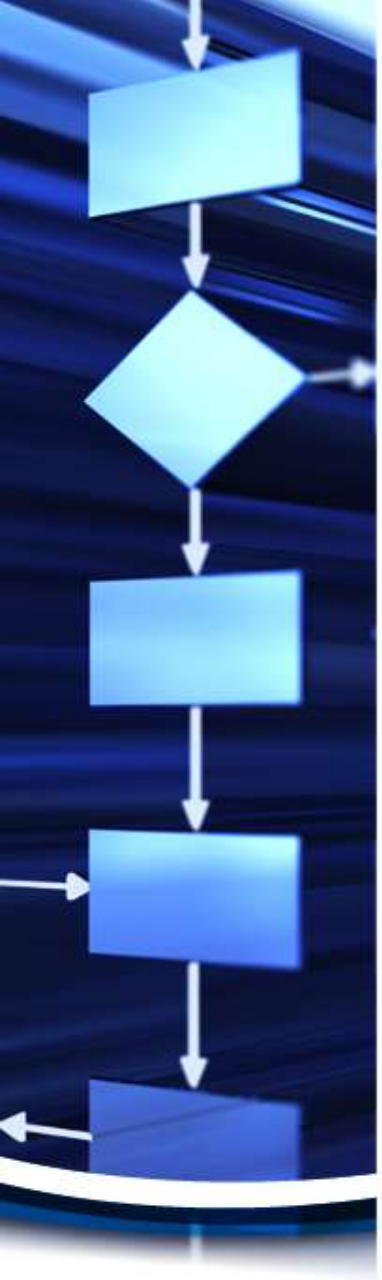
Features of Flowchart

- A good way of communicating the logic of a program.
- It maintains good program documentation.(DFD)
- It can be used to standardize a process for effective analysis, efficiency.
- It helps a programmer to maintain unnecessary steps in a program and improve it, and pay attention to that part of the information flow, which is to be modified.
- Using charts, we can create an oral presentation and can organize a group or individual projects.
- Generally three concepts are used to draw a flowchart:
- 1. Simple 2. Decision Making 3. Looping

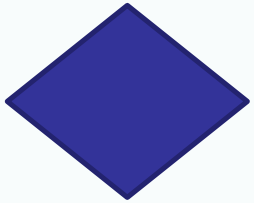
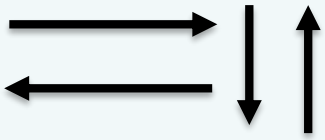


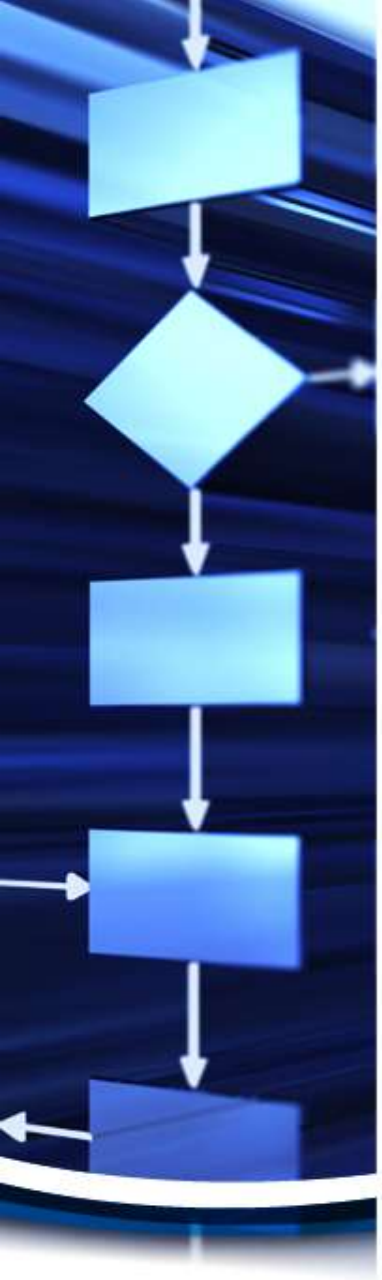
RULES TO DRAW A FLOWCHARTS

- The process in a flowchart can be expressed through boxes and arrows with different sizes.
- We can easily highlight a certain element and the relationships between each part.
- Draw relevant symbols of program flowchart on the page.
- Connect the steps with arrow according to their correct order.
- A flowchart will always be drawn from TOP to BOTTOM showing the exact order of the steps.

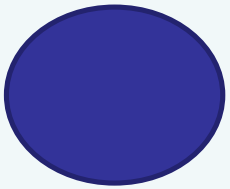
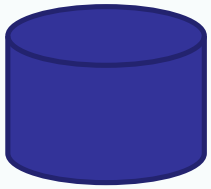


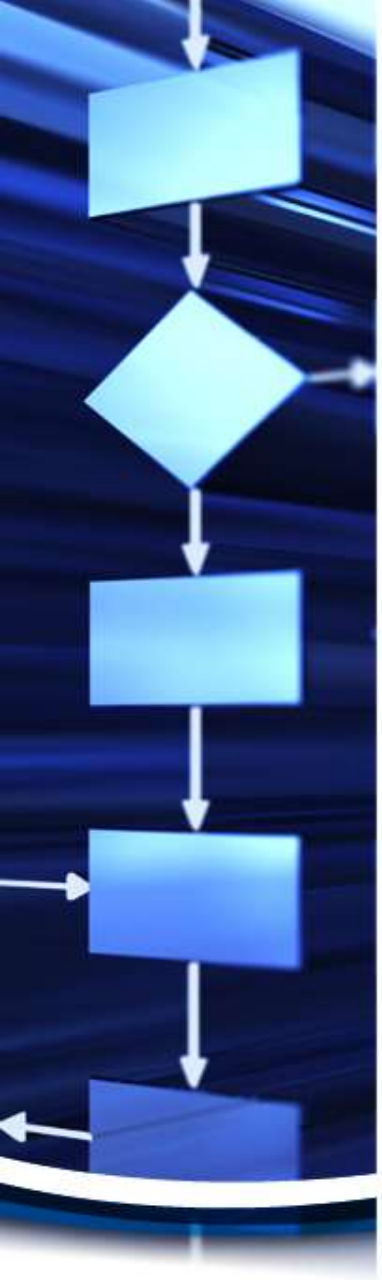
FLOWCHART SYMBOLS

Symbol	Name / Used for	Description
	Start / End (Terminator Symbol)	The terminator symbol marks the starting or ending point of the system. It usually contains the word “START” or “END”
	Process Box	A box can represent a single step . Such as SUM=A+B.
	Input / Output	It represents material or information entering or leaving the system, such as Read A, Read B.
	**Decision Box**	A Decision or branching point. Lines representing different decisions emerge from different points of the diamond.
	**Flow Lines**	A flow line indicate the sequence of steps and the direction of flow.



FLOWCHART SYMBOLS

Symbol	Name / Used for	Description
	**ON Page Connector Symbol**	The terminator symbol marks the starting or ending point of the system. It usually contains the word “START” or “END”
	OFF Page Connector Symbol	symbol connects separate elements across multiple pages. It is frequently used within complex charts.
	Pre – defined Process Symbol	It indicates a complicated process or operation that is well-known or defined elsewhere.
	**The stored data symbol**	The stored data symbol represents data on a storage device.

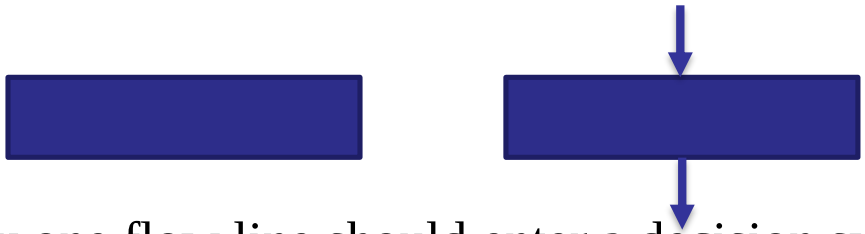


GUIDELINES DRAW A FLOWCHARTS

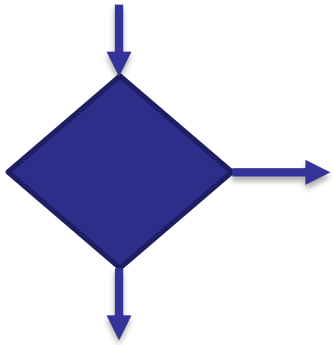
1. The A flowchart should be **clear, neat and easy to follow**.

2. The usual direction of the **flow of a procedure** or system(arrow) is from "**Left to Right**" or " **Top to Bottom**".

3. Only one flow line should come out from a process symbol.



4. Only one flow line should enter a decision symbol.

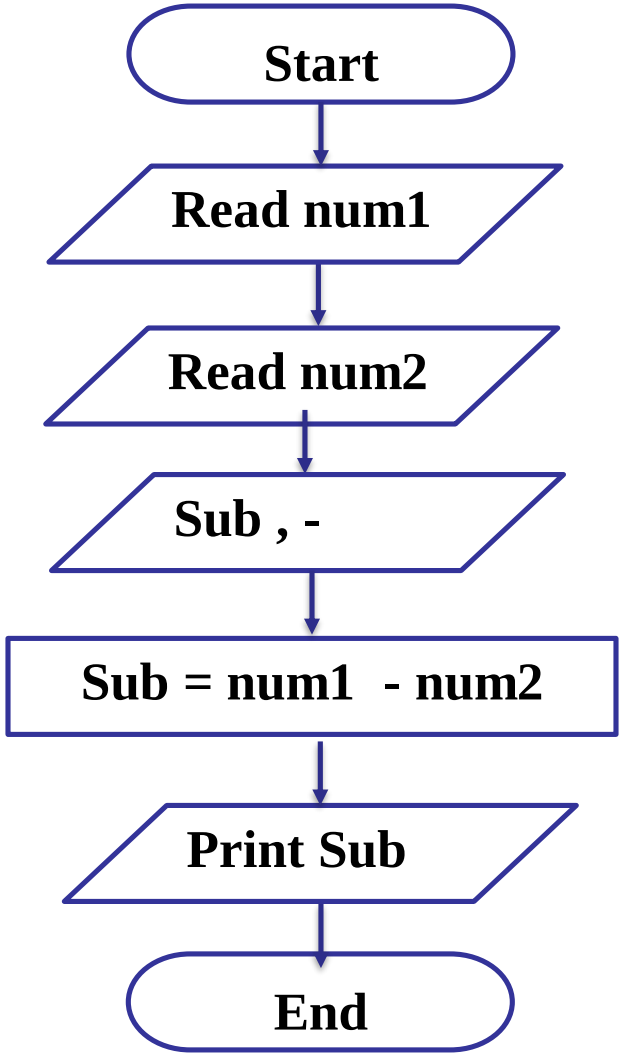


5. Ensure that the flowchart has a logical start and finish.

SIMPLE FLOWCHART

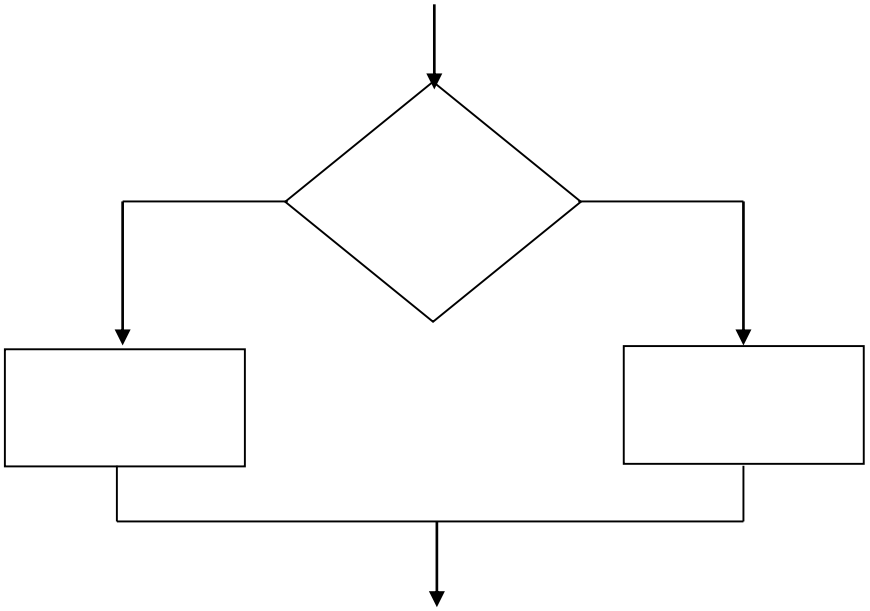
❖ How to draw a simple flowchart :

Example 1: Flowchart to find the sum of two numbers.



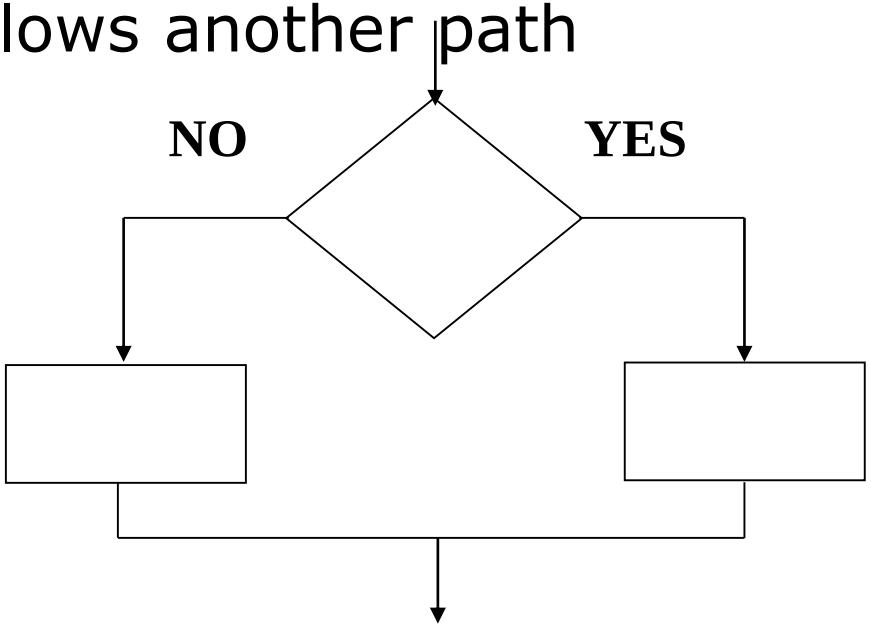
Decision Structure

- One of two possible actions is taken, depending on a condition.



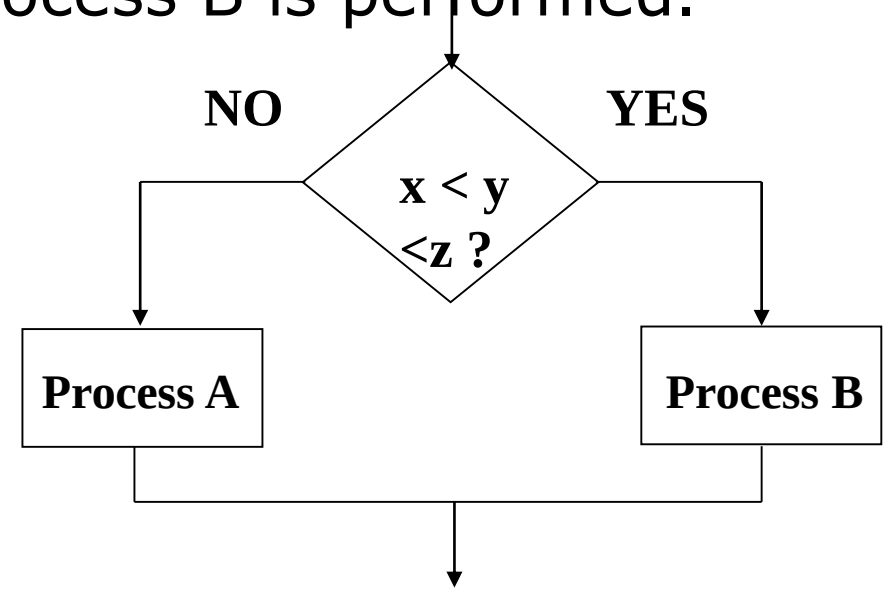
Decision Structure

- A new symbol, the diamond, indicates a yes/no question. If the answer to the question is yes, the flow follows one path. If the answer is no, the flow follows another path



Decision Structure

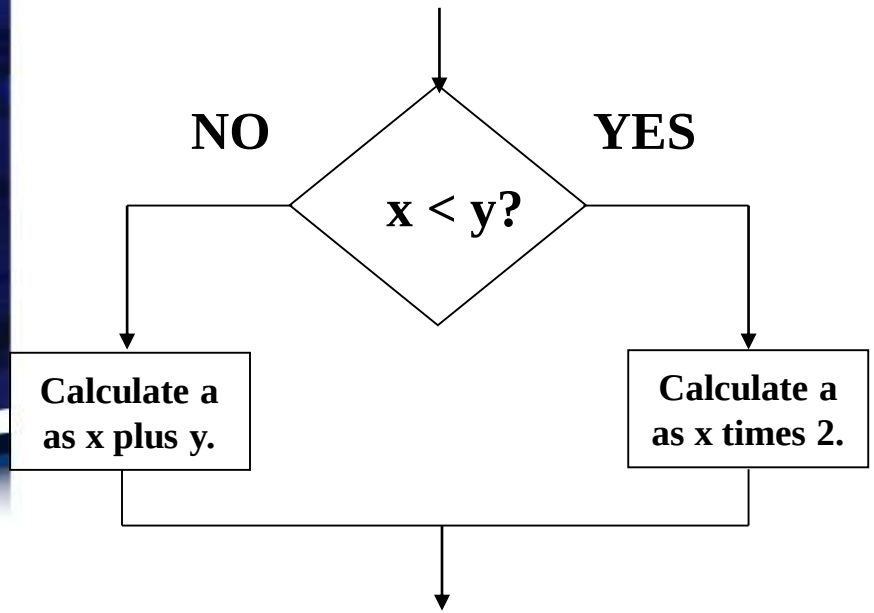
- In the flowchart segment below, the question "is x < y?" is asked. If the answer is no, then process A is performed. If the answer is yes, then process B is performed.



Decision Structure

- The flowchart segment below shows how a decision structure is expressed in C as an if/else statement.

Flowchart



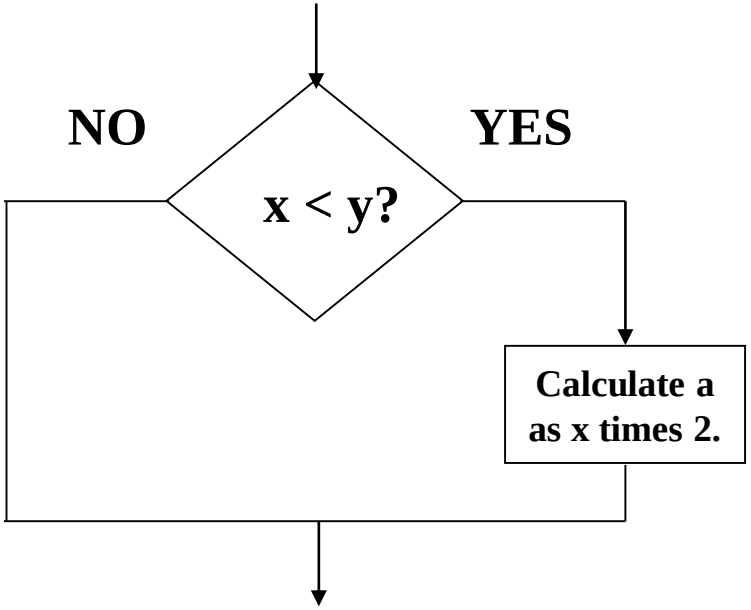
C Code

```
if (x < y)
        a = x * 2;
else
        a = x + y;
```

Decision Structure

- The flowchart segment below shows a decision structure with only one action to perform. It is expressed as an if statement in C code.

Flowchart

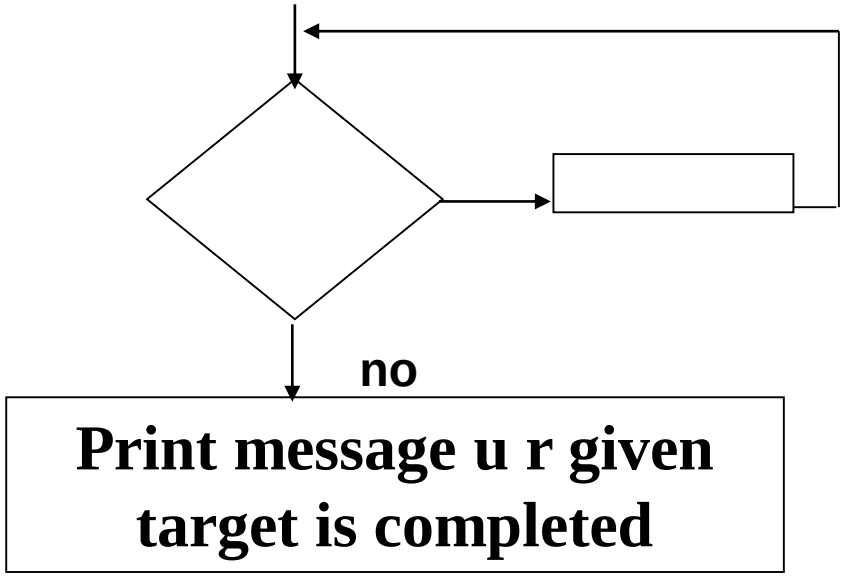


C Code

```
if (x < y)
        a = x * 2;
```

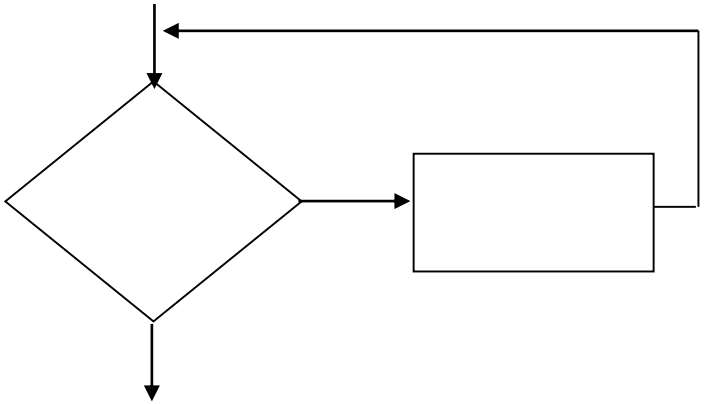
Repetition(Looping) Structure

- A repetition structure represents part of the program that repeats. This type of structure is commonly known as a loop.



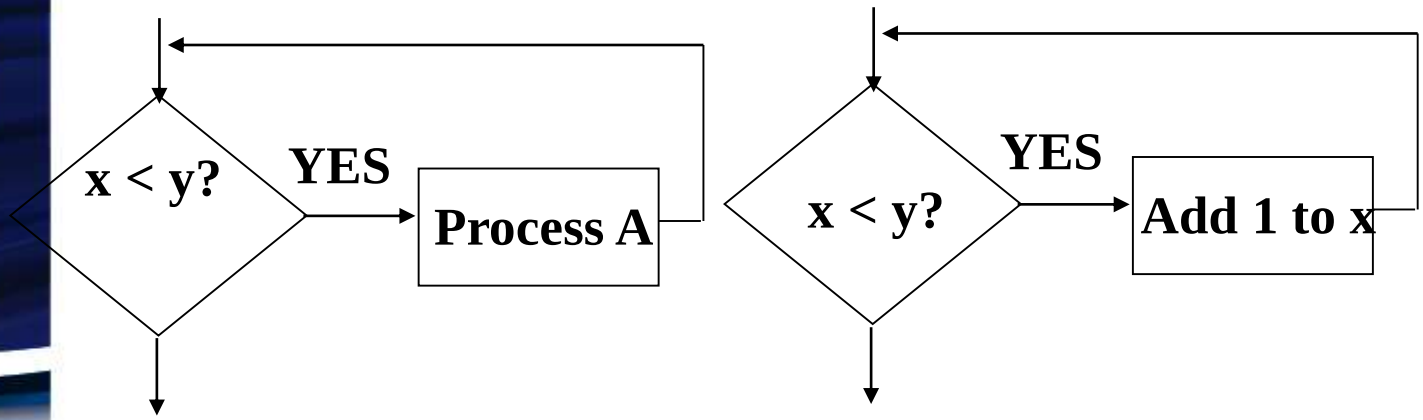
Repetition(Looping) Structure

- Notice the use of the diamond symbol. A loop tests a condition, and if the condition exists, it performs an action. Then it tests the condition again. If the condition still exists, the action is repeated. This continues until the condition no longer exists.



Repetition(Looping) Structure

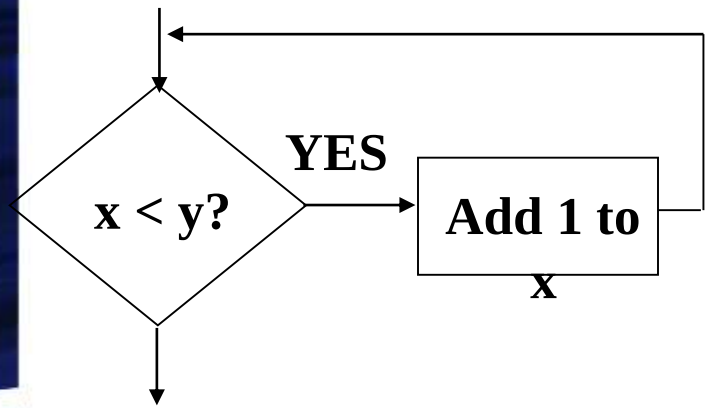
- In the flowchart segment, the question “is x < y?” is asked. If the answer is yes, then Process A is performed. The question “is x < y?” is asked again. Process A is repeated as long as x is less than y. When x is no longer less than y, the repetition stops and the structure is exited.



Repetition(Looping) Structure

- The flowchart segment below shows a repetition structure expressed in C as a while loop.

Flowchart

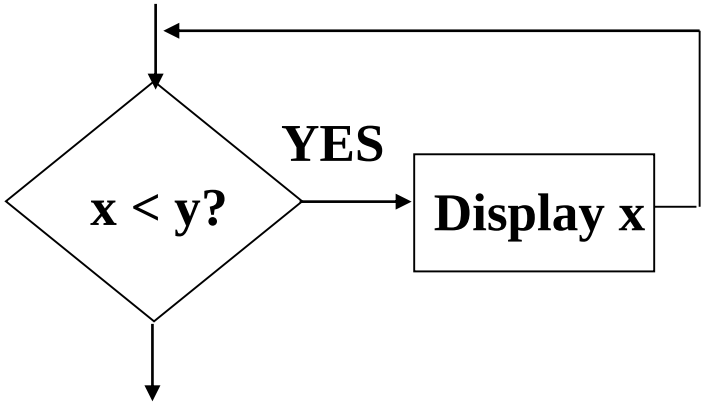


C Code

```
if (x < y)
        x++;
Else
Print " Thank u visit once again"
```

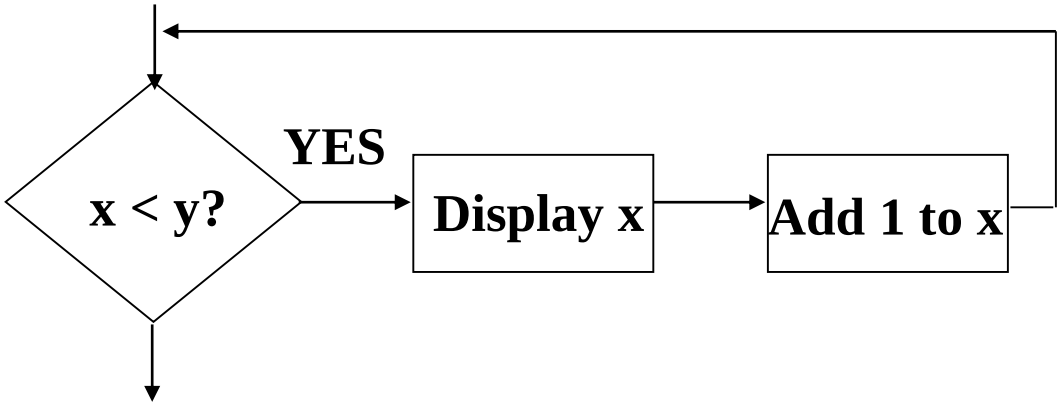
Controlling a Repetition Structure

- The action performed by a repetition structure must eventually cause the loop to terminate. Otherwise, an infinite loop is created.
- In this flowchart segment, x is never changed. Once the loop starts, it will never end.
- QUESTION: How can this flowchart be modified so it is no longer an infinite loop?



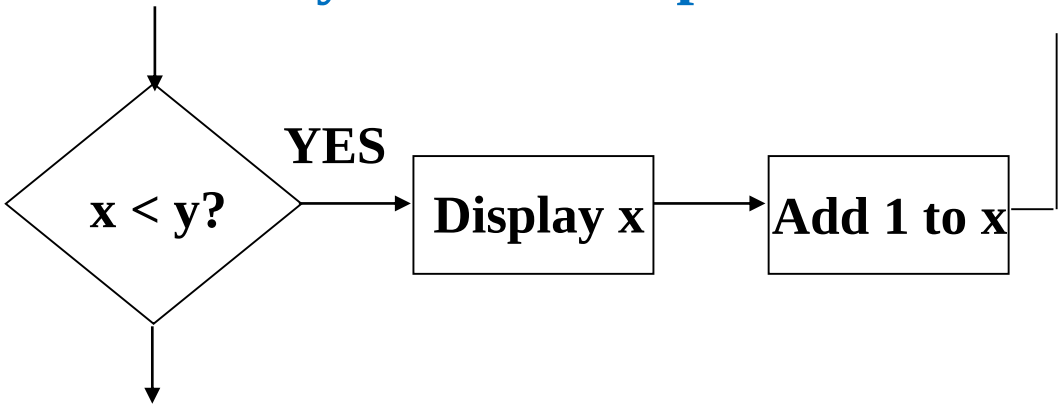
Controlling a Repetition Structure

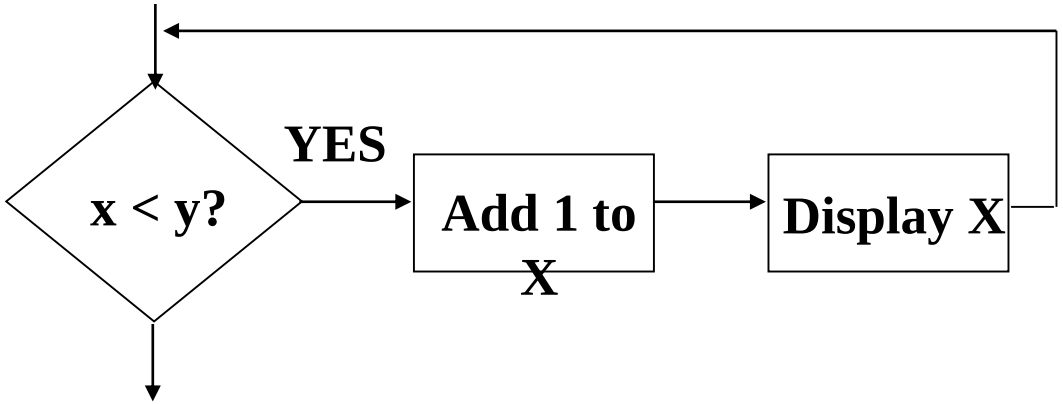
- ANSWER: By adding an action within the repetition that changes the value of x.



A Pre-Test Repetition Structure

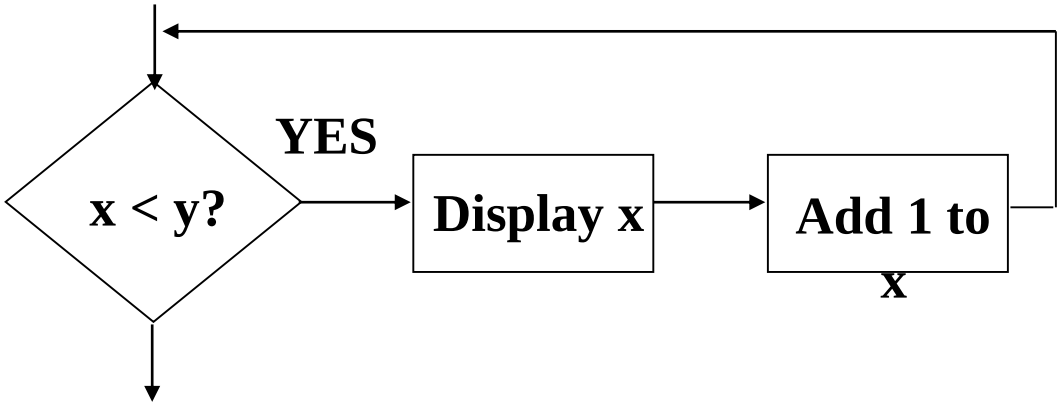
- This type of structure is known as a pre-test repetition structure. **The condition is tested *BEFORE* any actions are performed.**





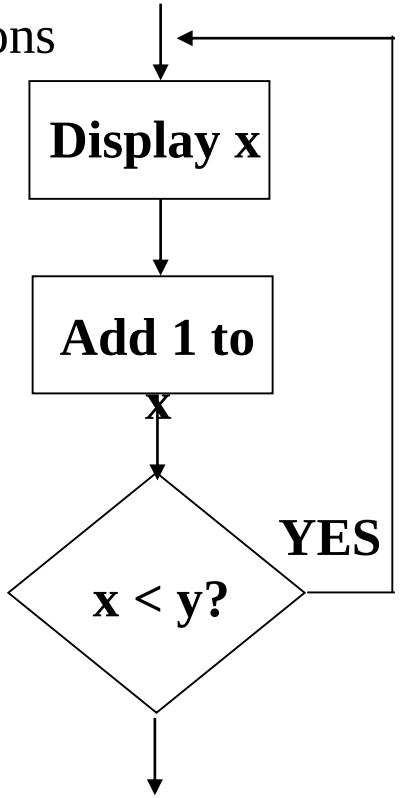
A Pre-Test Repetition Structure

- In a pre-test repetition structure, if the condition does not exist, the loop will never begin.



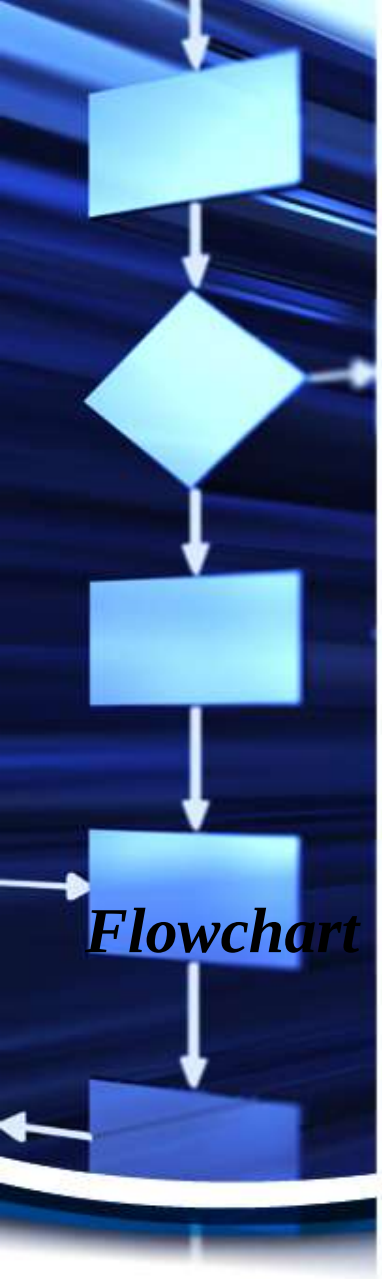
A Post-Test Repetition Structure

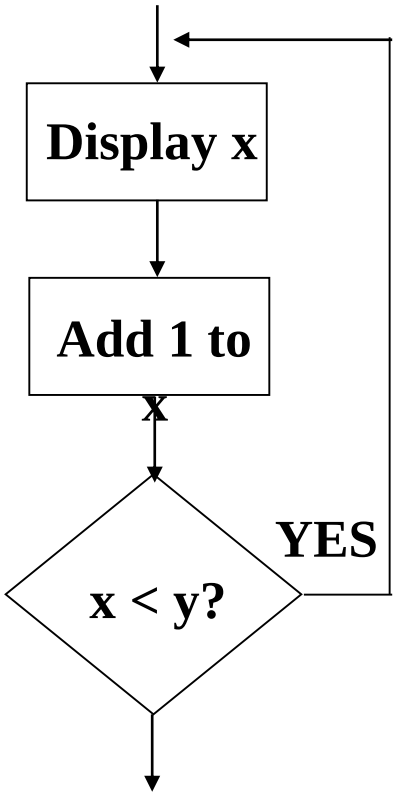
- This flowchart segment shows a post-test repetition structure.
- The condition is tested *AFTER* the actions are performed.
- A post-test repetition structure always performs its actions at least once.



A Post-Test Repetition Structure

- The flowchart segment below shows a post-test repetition structure expressed in C as a do-while loop.

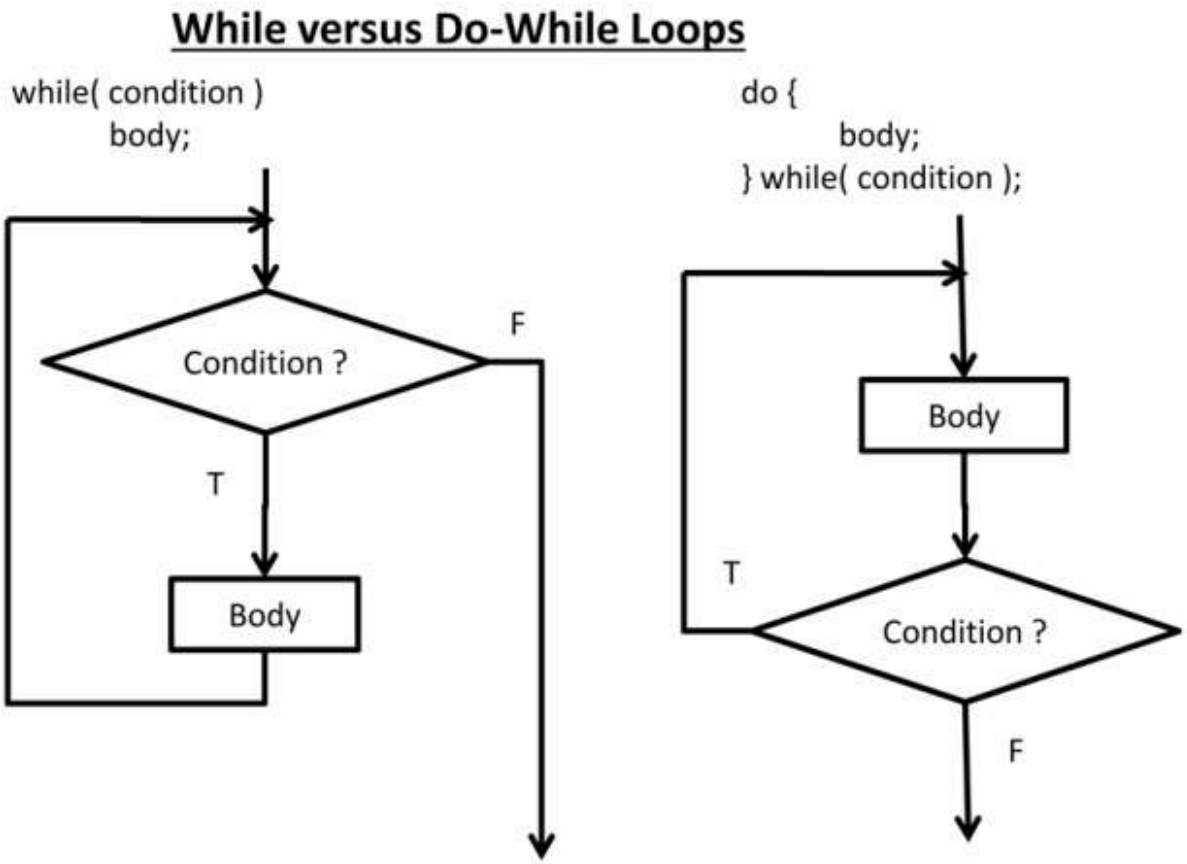




C Code

```
do
{
        printf("%d",x);
        x++;
} while (x < y);
```

Post-Test vs Pre-Test Repetition Structure



Converting Algorithm to a Flowchart

Algorithm and Flowchart to find the sum of two numbers.

Algorithm:

Step 1 : Start

Step 2: [Initialize Counter]

Input num1, num2 and Sum

Step 3: [Process Counter]

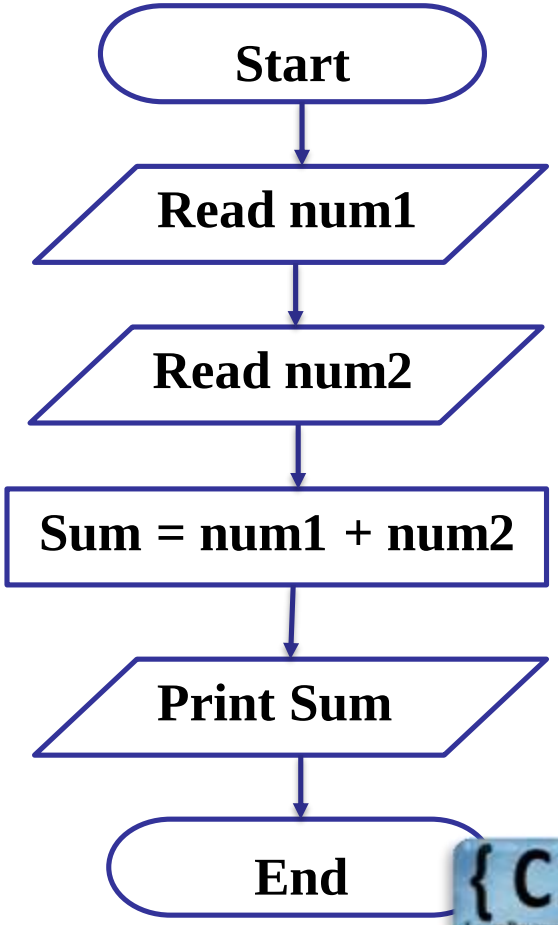
Sum= num1 + num2

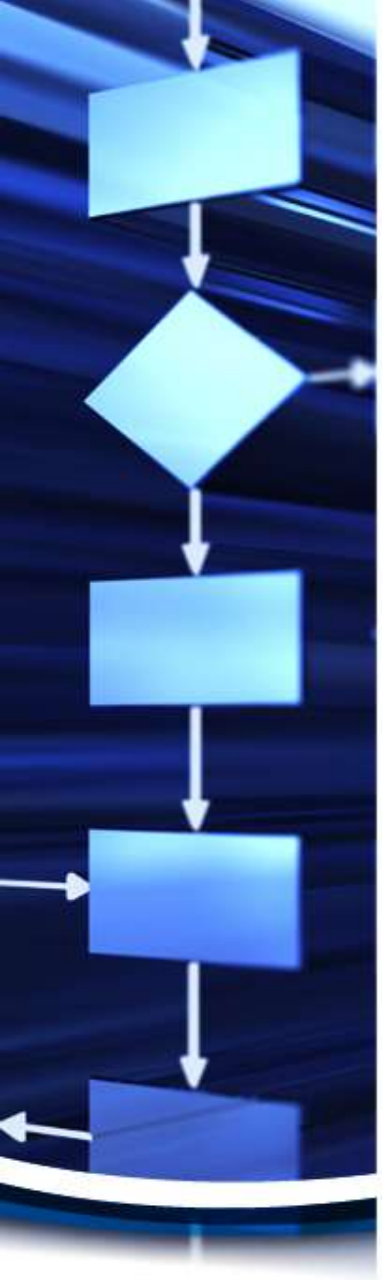
Step 4: [Output]

Print Sum

Step 5: End.

Flowchart:





Converting Algorithm to a Flowchart

Decision Making:

To find greatest number between two numbers.

Algorithm:

Step 1 : Start

Step 2: [Initialize Counter]

Input num1, num2

Step 3: [Process Counter]

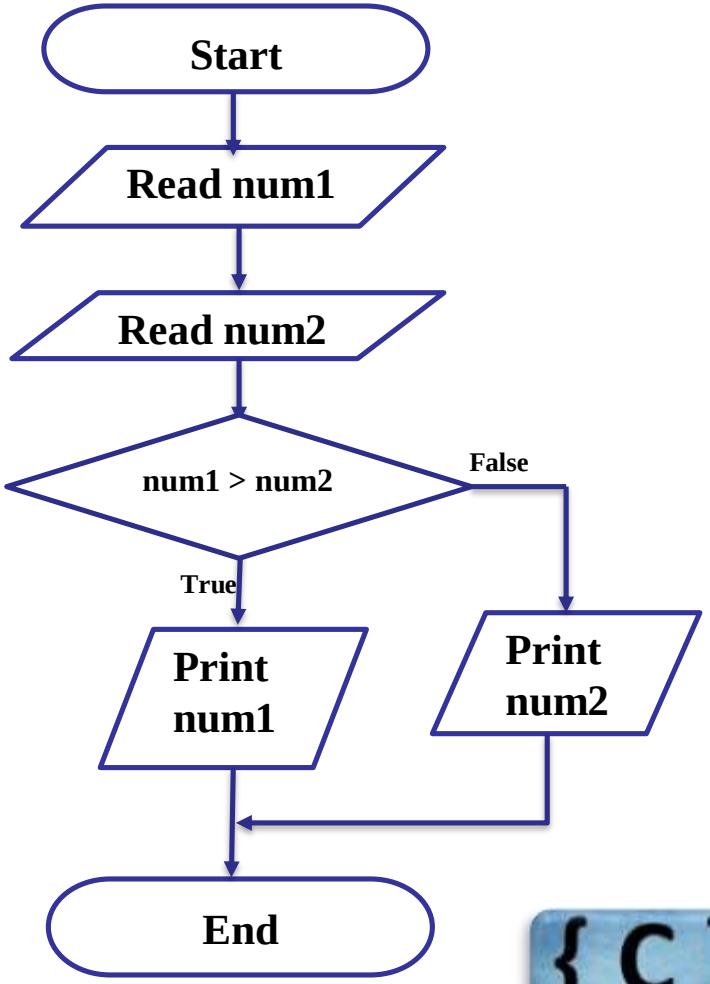
num1 > num2

Step 4: [Output]

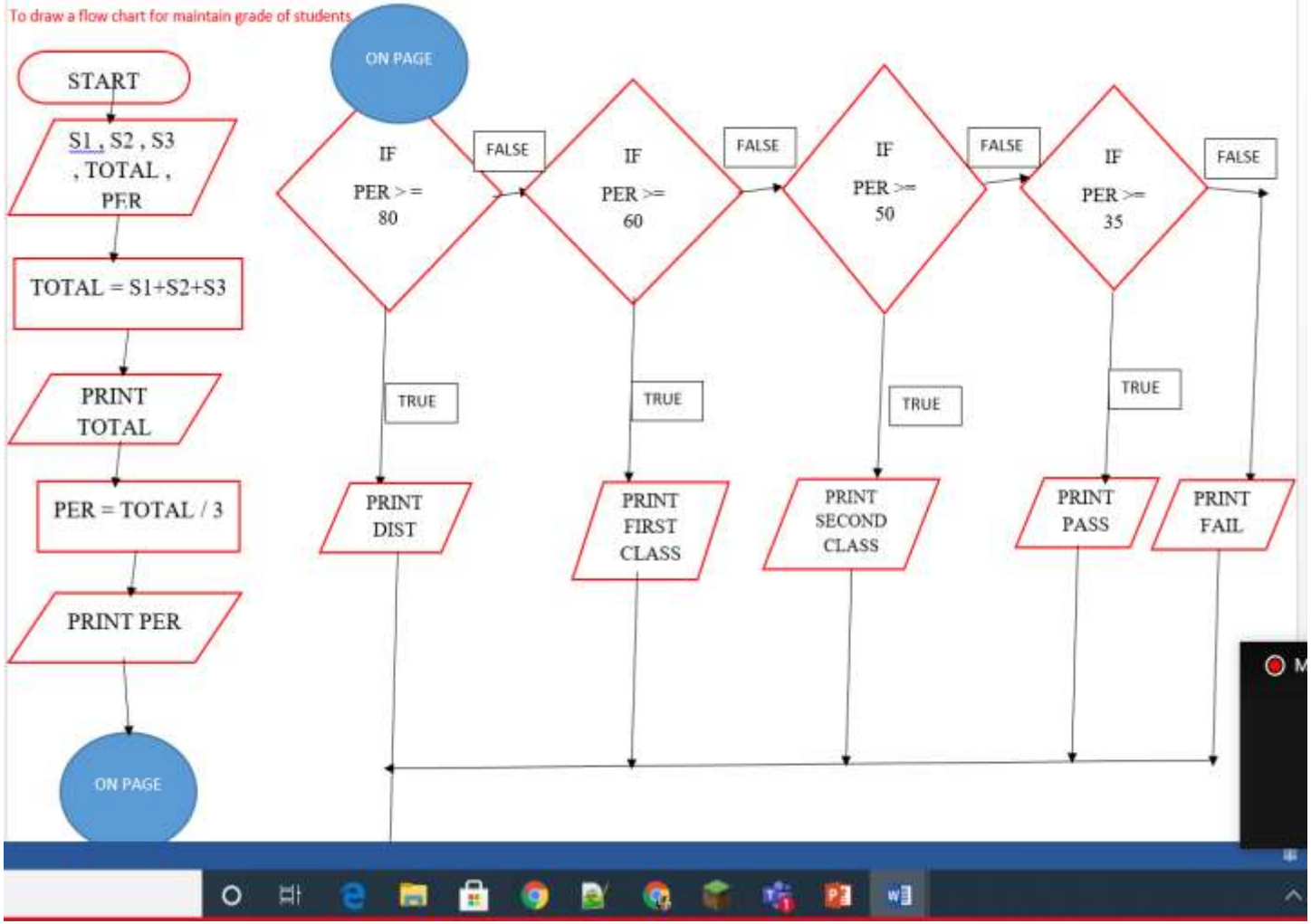
Print num1 [if success]

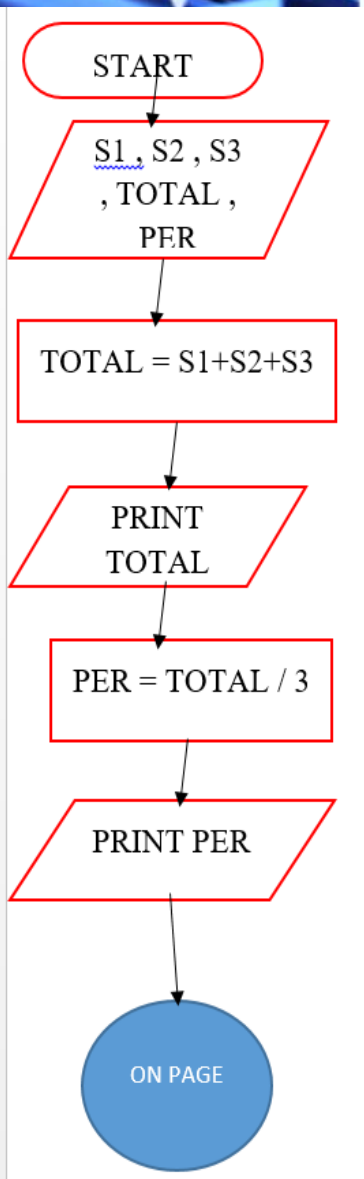
Step 5: End.

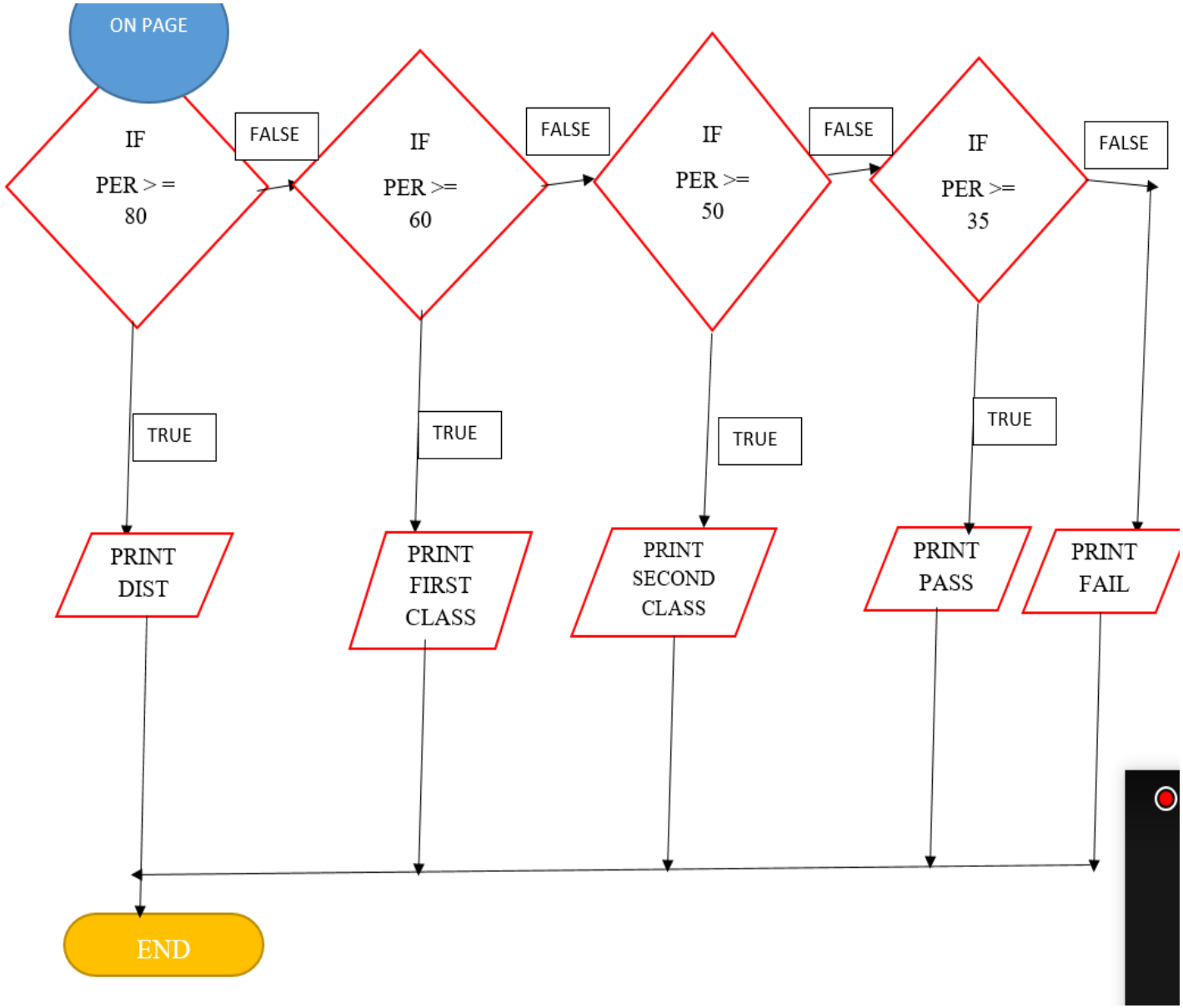
Flowchart:



To draw a flow chart for maintain grade of students




START
S1 , S2 , S3 , TOTAL , PER
TOTAL = S1+S2+S3
PRINT TOTAL
PER = TOTAL / 3
PRINT PER
ON PAGE


ON PAGE
IF PER > = 80
FALSE
IF PER >= 60
FALSE
IF PER >= 50
FALSE
IF PER >= 35
FALSE
TRUE
TRUE
TRUE
TRUE
PRINT DIST
PRINT FIRST CLASS
PRINT SECOND CLASS
PRINT PASS
PRINT FAIL
END

Converting Algorithm to a Flowchart

Looping: **[** continually repeated until specific condition is fulfilled]

Generate the multiplication table of 10 to understand **loop.**

Algorithm:

Step 1 : Start

Step 2: [Initialize Counter]

Get the number N

Step 3: Let 'Count' be equal to 1

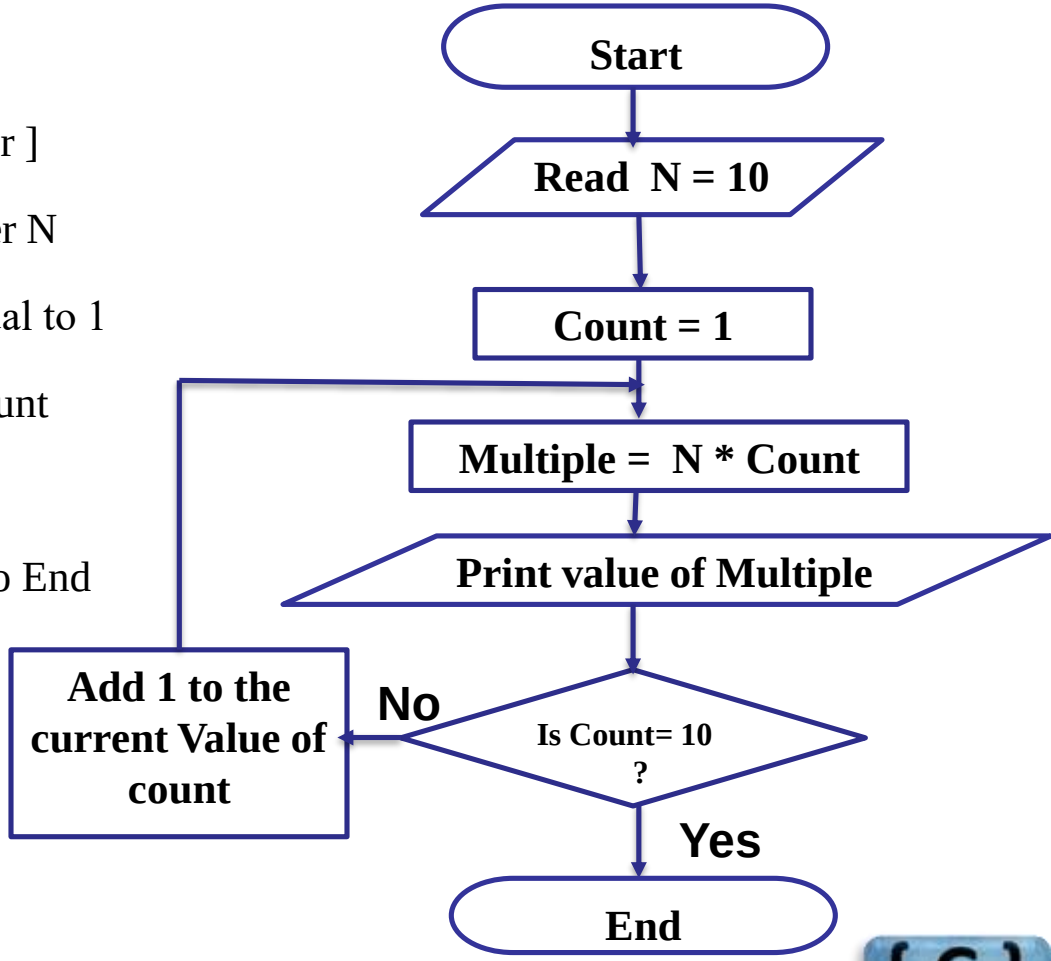
Step 4: Multiple = N * Count

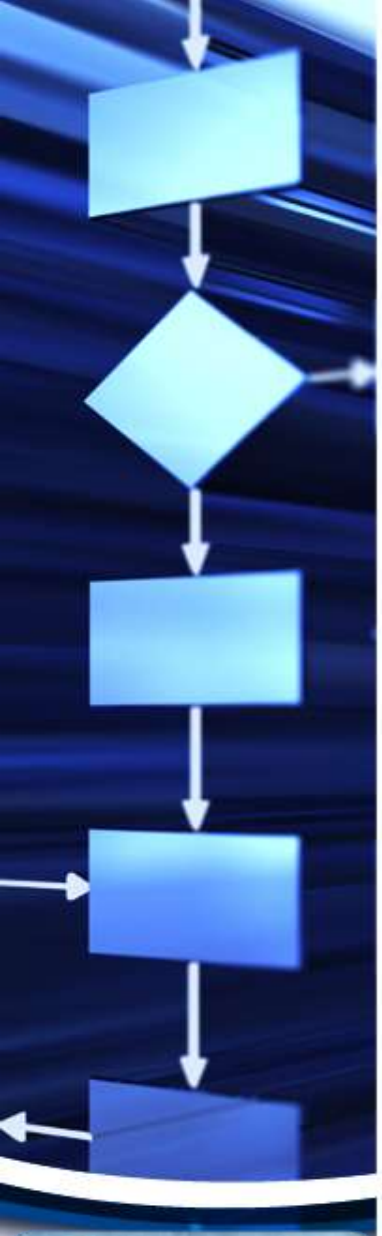
Step 5: Display Multiple

Step 6: If Count is 10 go to End

Step 7: End

Flowchart:





{ C }
LearnProgramCode