

Introduction to Flowcharting




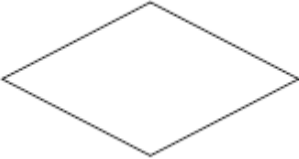


The Flowchart

A Flowchart

- shows logic of an algorithm
- emphasizes individual steps and their interconnections
- e.g. control flow from one action to the next
- Used for analyzing, designing, documenting or managing a process or program.

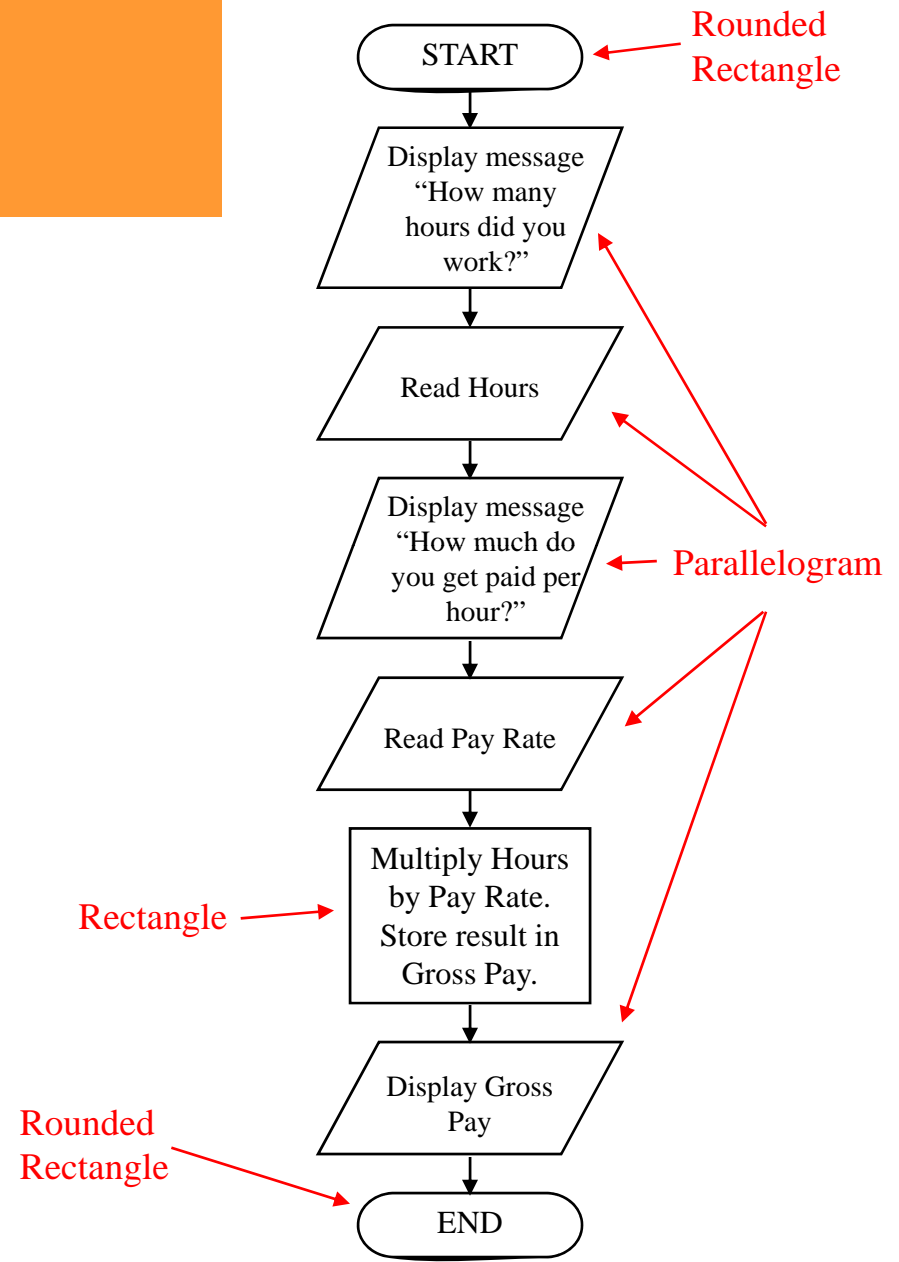
Flowchart Symbols

Basic

Name	Symbol	Use in Flowchart
Rounded Rectangle		Denotes the beginning or end of the program
Parallelogram		Denotes an Input Operation
Rectangle		Denotes a process to be carried out e.g. addition, subtraction, division etc.
Diamond		Denotes a decision (or branch) to be made. The program should continue along one of two routes. (e.g. IF/THEN/ELSE)
Repetition /looping		Denotes repetition of same thing. (e.g looping)
Flow line		Denotes the direction of logic flow in the program

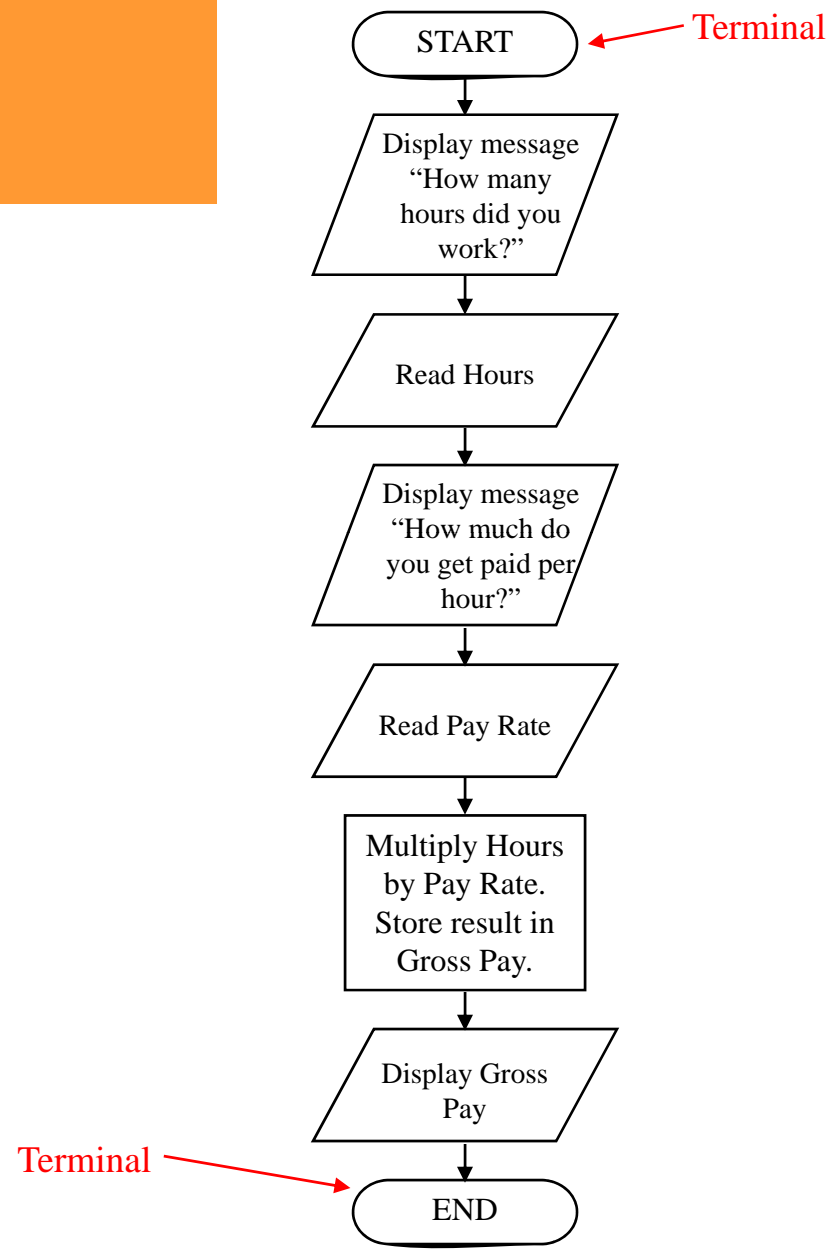
Basic Flowchart Symbols

- Notice there are three types of symbols in this flowchart:
 - rounded rectangles (**Start/End**)
 - Parallelograms (**Input**)
 - a rectangle (**Process**)
- Each symbol represents a different type of operation.



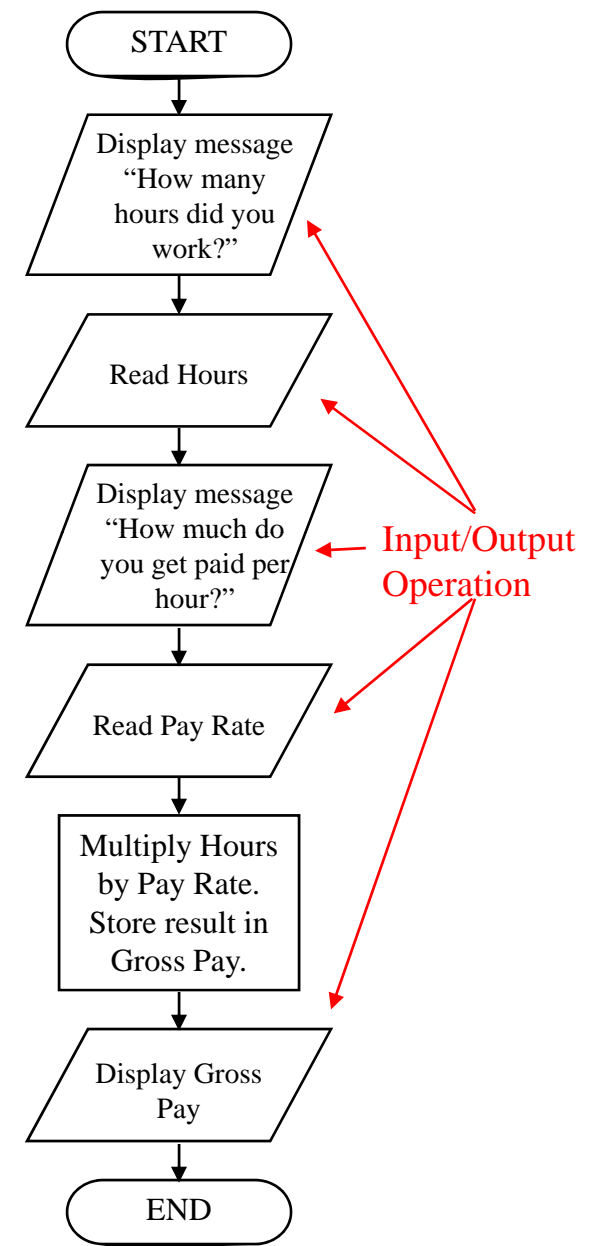
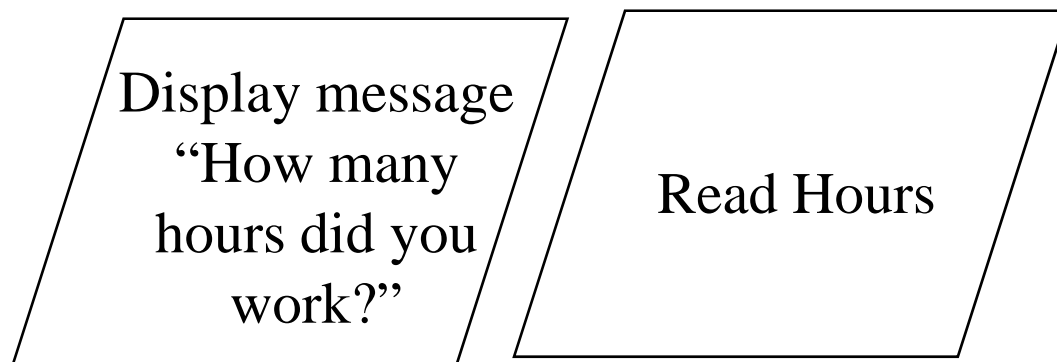
Basic Flowchart Symbols

- Terminals
 - represented by rounded rectangles
 - indicate a starting or ending point



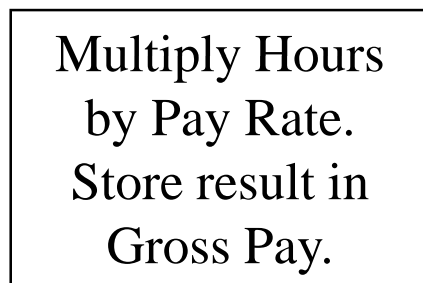
Basic Flowchart Symbols

- Input/Output Operations
 - represented by parallelograms
 - indicate an input or output operation

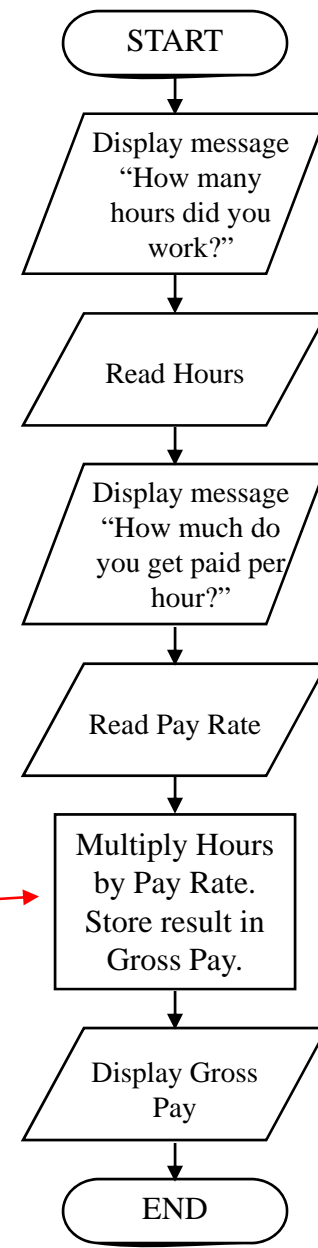


Basic Flowchart Symbols

- Processes
 - represented by rectangles
 - indicates a process such as a mathematical computation or variable assignment



Process →



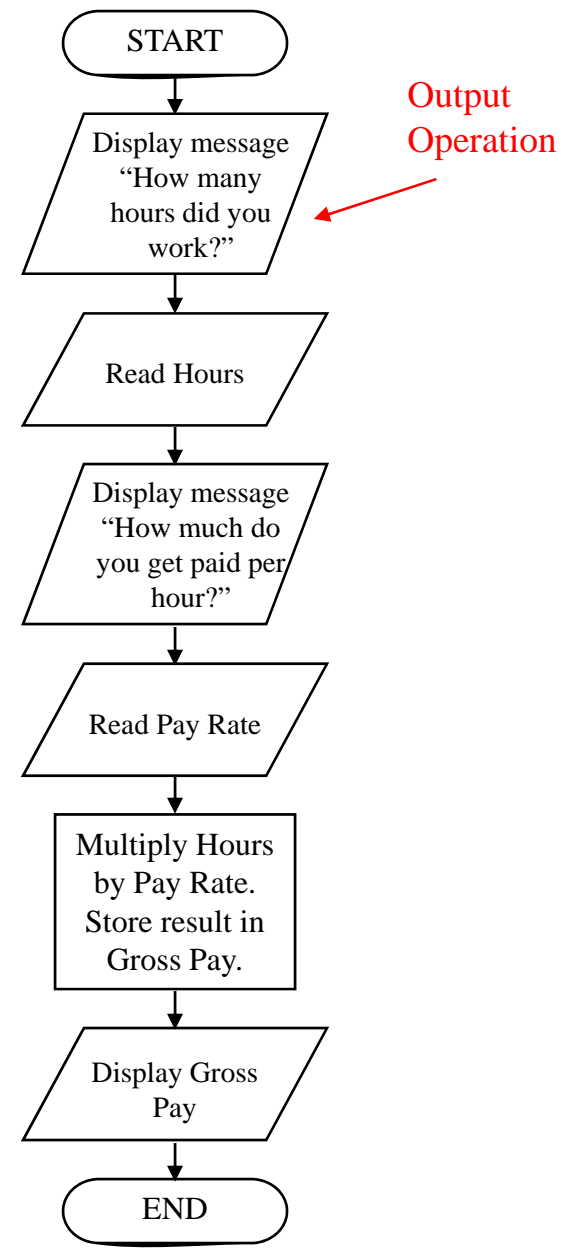
Stepping Through the Flowchart

Variable Contents:

Hours: ?

Pay Rate: ?

Gross Pay: ?



Stepping Through the Flowchart

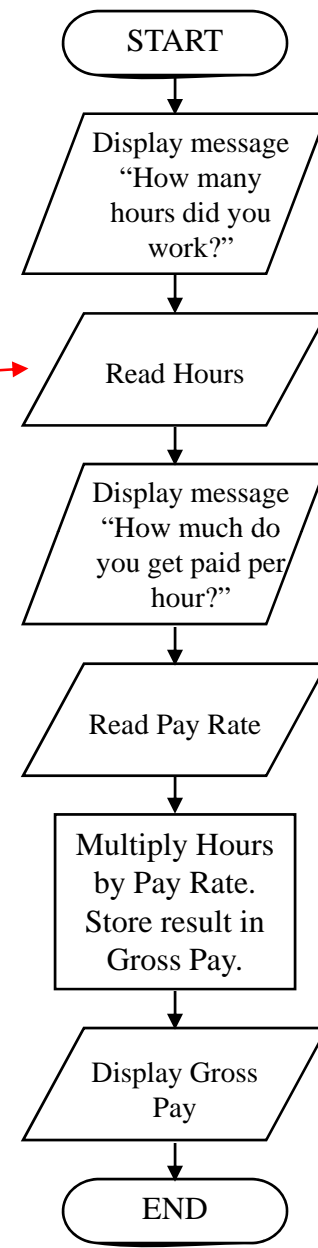
Variable Contents:

Hours: 40

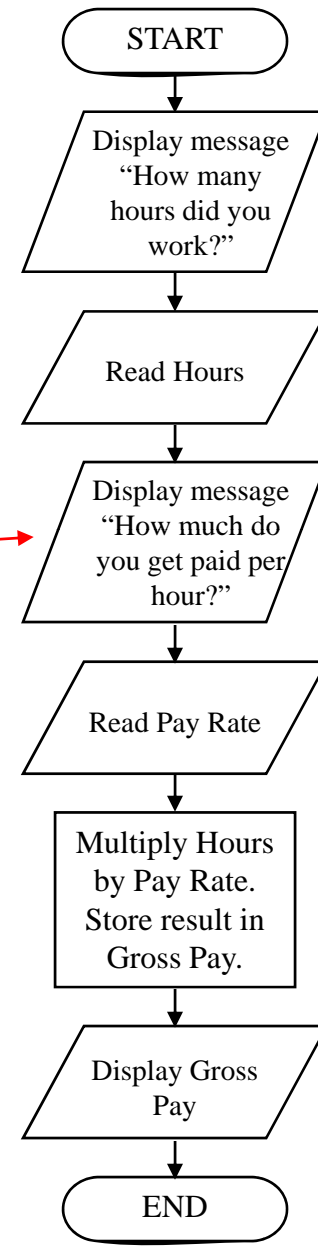
Pay Rate: ?

Gross Pay: ?

Input
Operation
(User types
40)



Stepping Through the Flowchart



Output
Operation

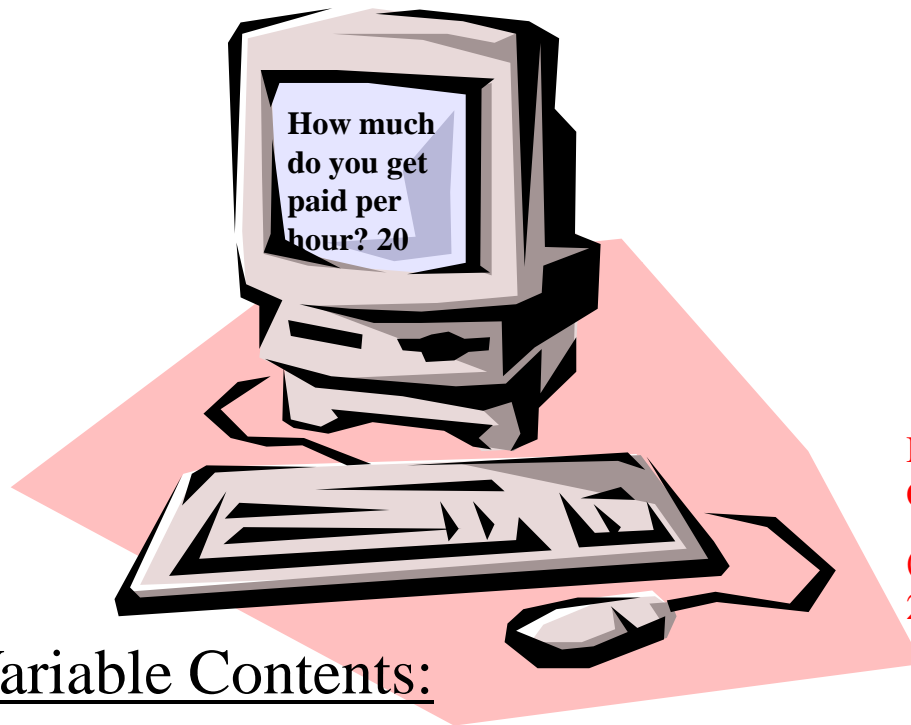
Variable Contents:

Hours: 40

Pay Rate: ?

Gross Pay: ?

Stepping Through the Flowchart



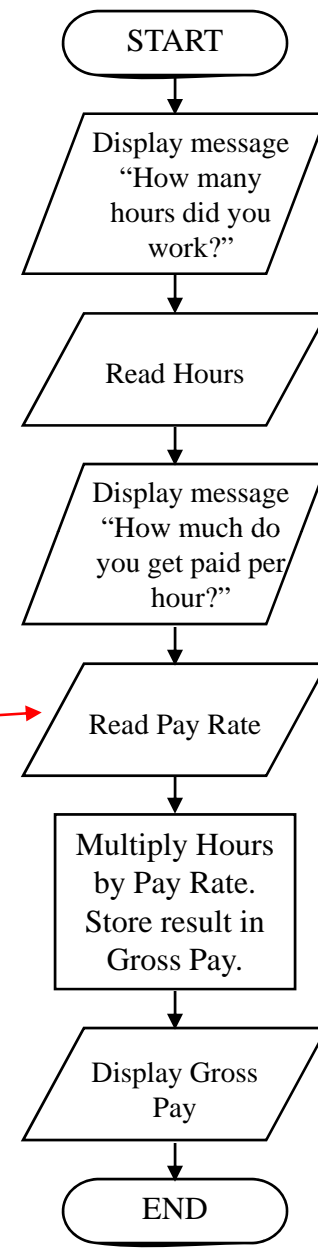
Variable Contents:

Hours: 40

Pay Rate: 20

Gross Pay: ?

Input
Operation
(User types
20)



Stepping Through the Flowchart

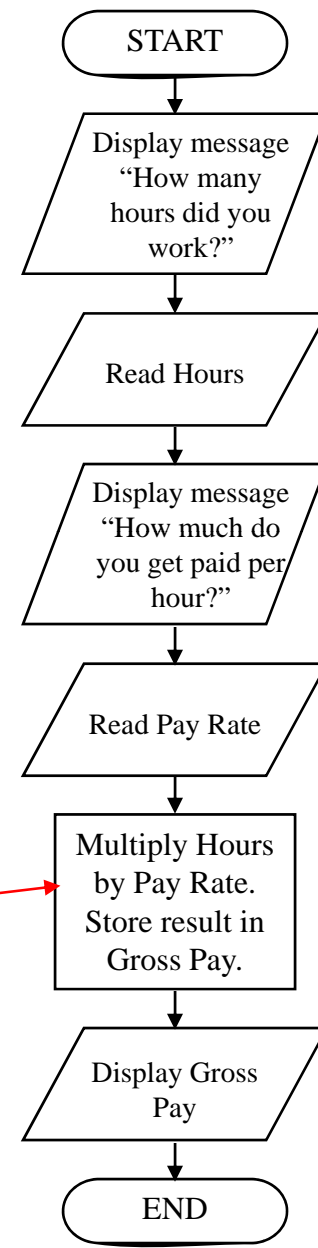
Variable Contents:

Hours: 40

Pay Rate: 20

Gross Pay: 800

Process: The product of 40 times 20 is stored in Gross Pay



Stepping Through the Flowchart

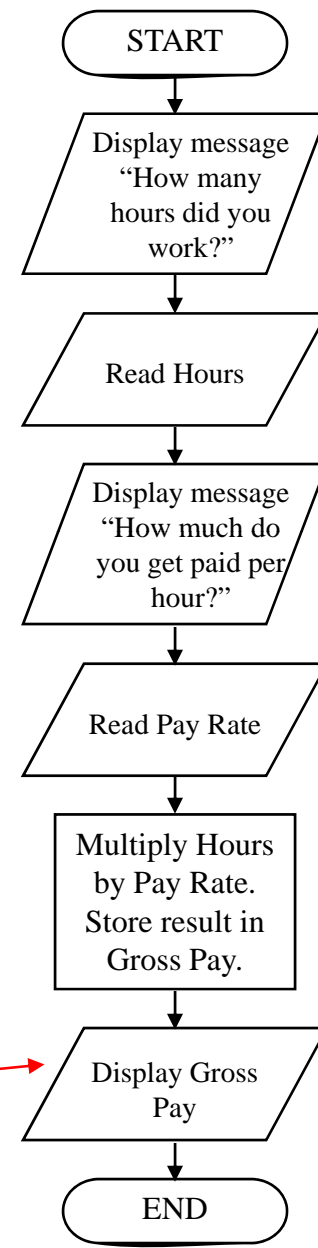
Variable Contents:

Hours: 40

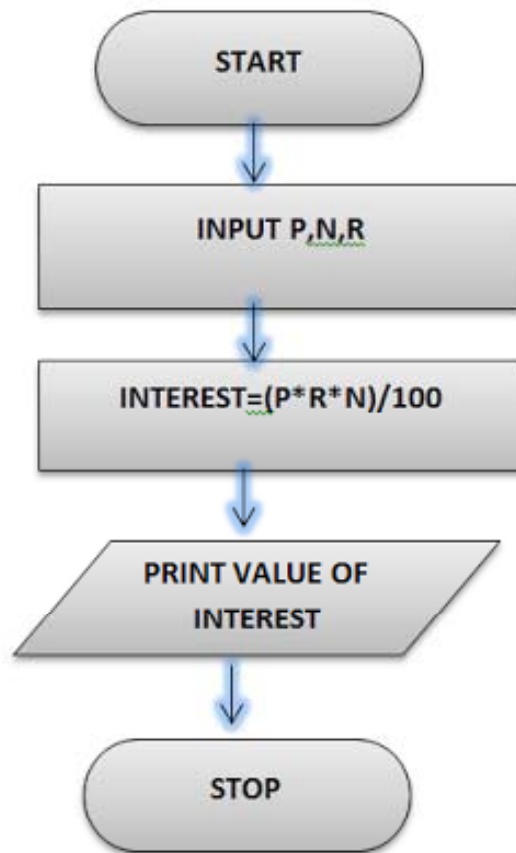
Pay Rate: 20

Gross Pay: 800

Output
Operation



Calculate simple interest



Solution

8

Draw a flowchart for a program that calculates and print the area and the perimeter of a rectangle.

□ **Input**

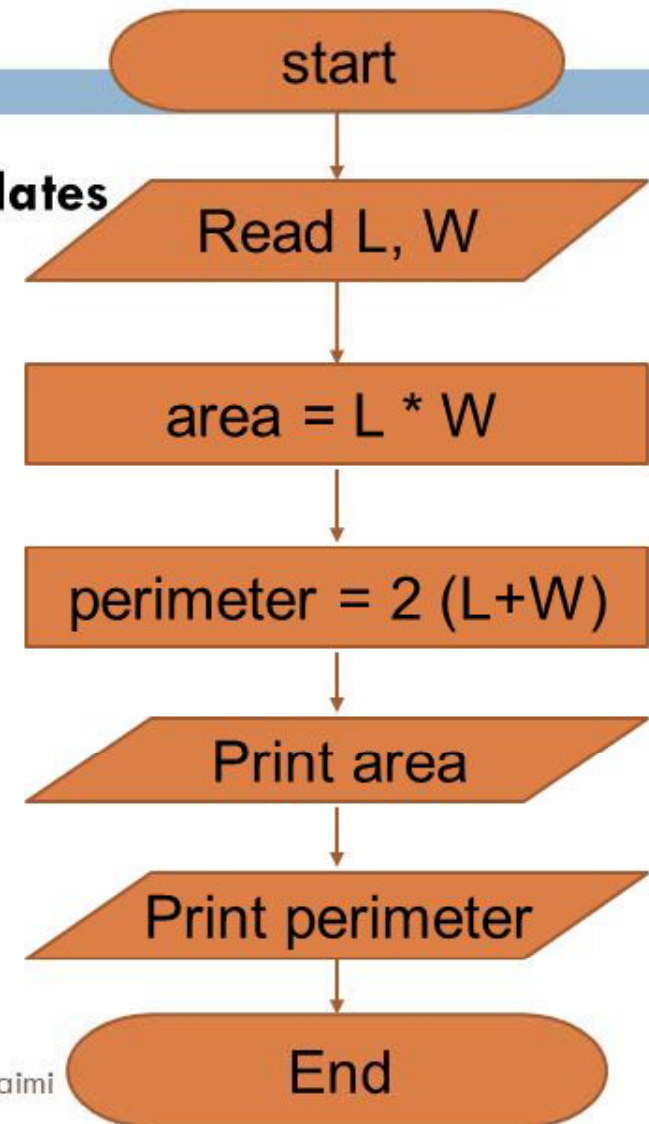
- Length
- width

□ **Processing**

- $\text{Area} = \text{length} * \text{width}$
- $\text{Perimeter} = 2 * (\text{length} + \text{width})$

□ **Output**

- Area
- Perimeter

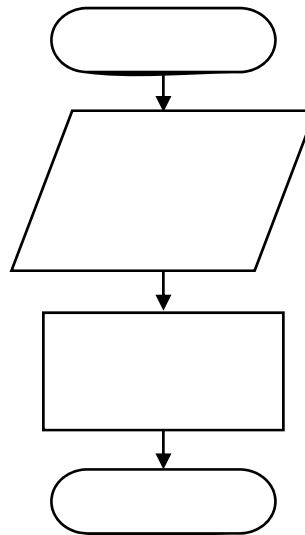


Four Flowchart Structures

- Sequence
- Decision
- Repetition
- Case

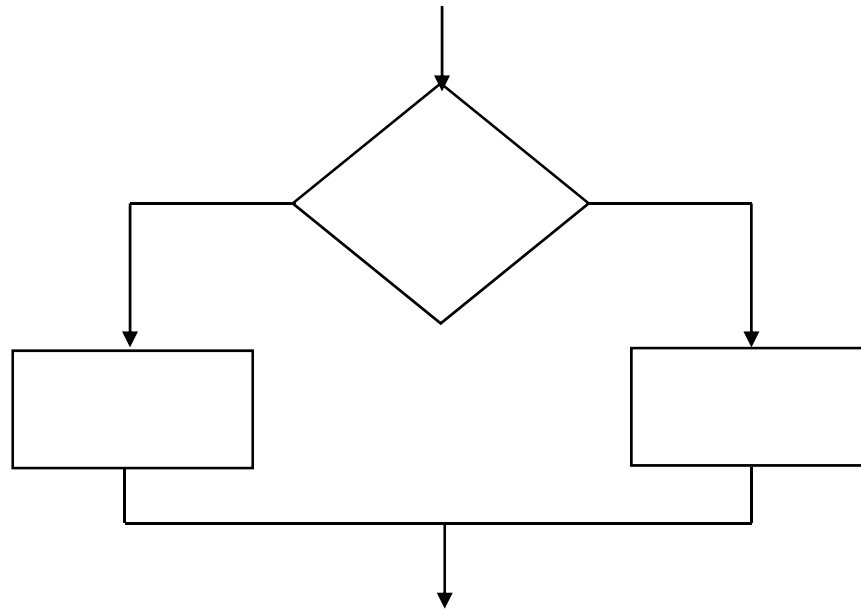
Sequence Structure

- a series of actions are performed in sequence
- The pay-calculating example was a sequence flowchart.



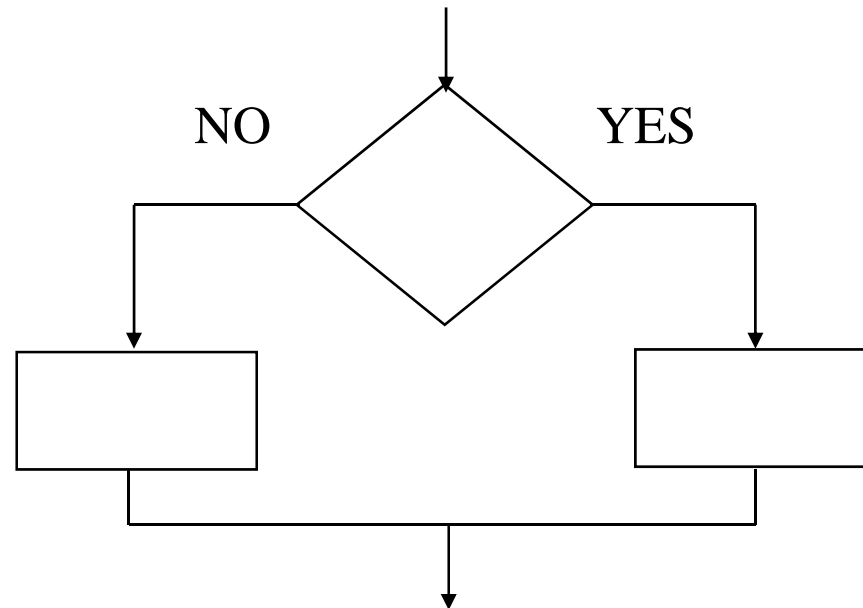
Decision Structure

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



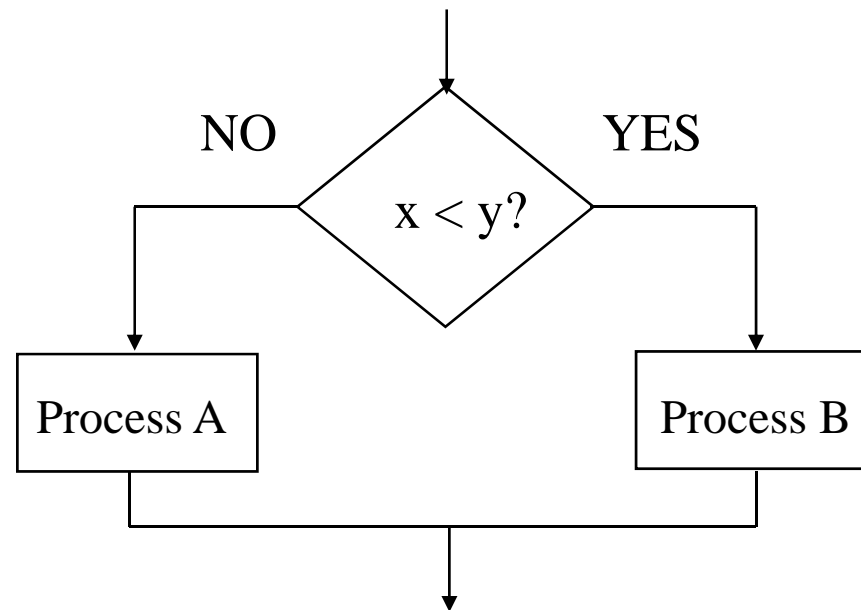
Decision Structure

- 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



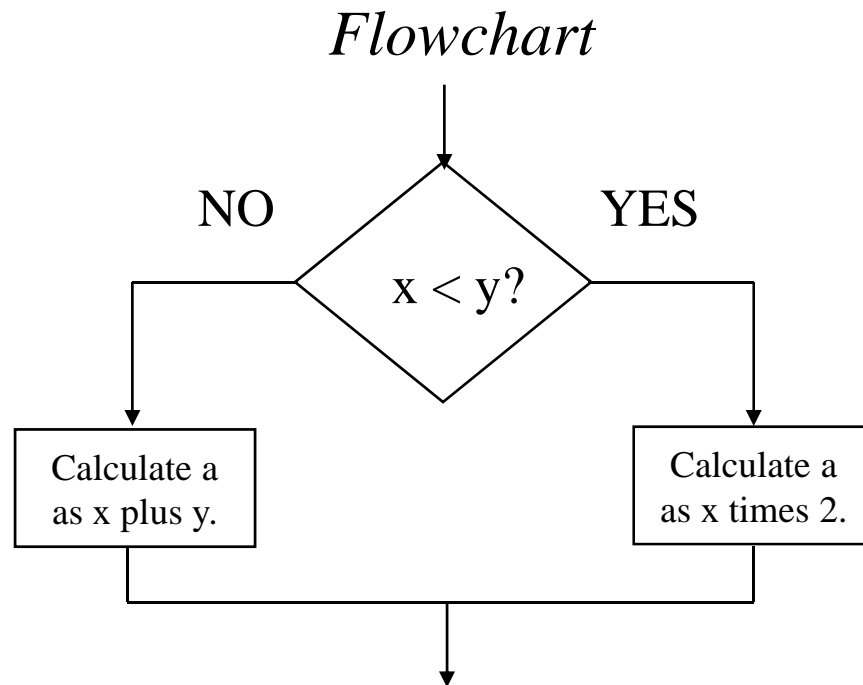
Decision Structure

- 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.



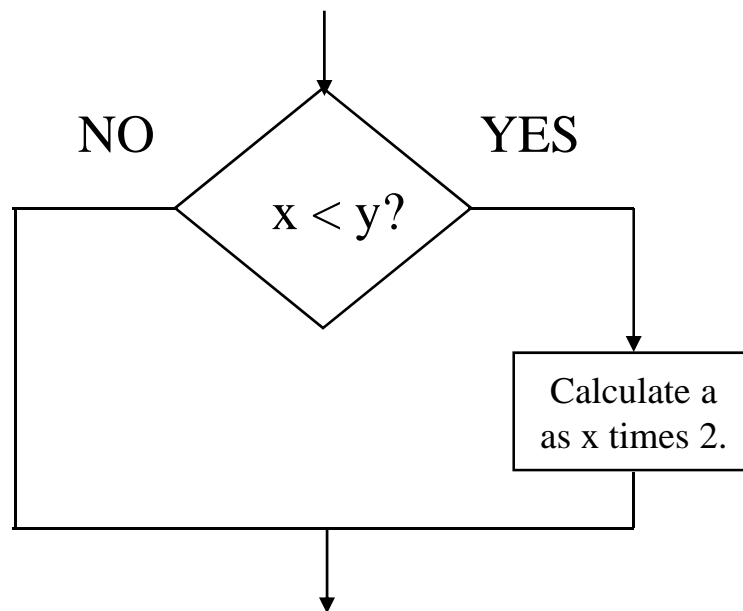
C++ Code

```
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.

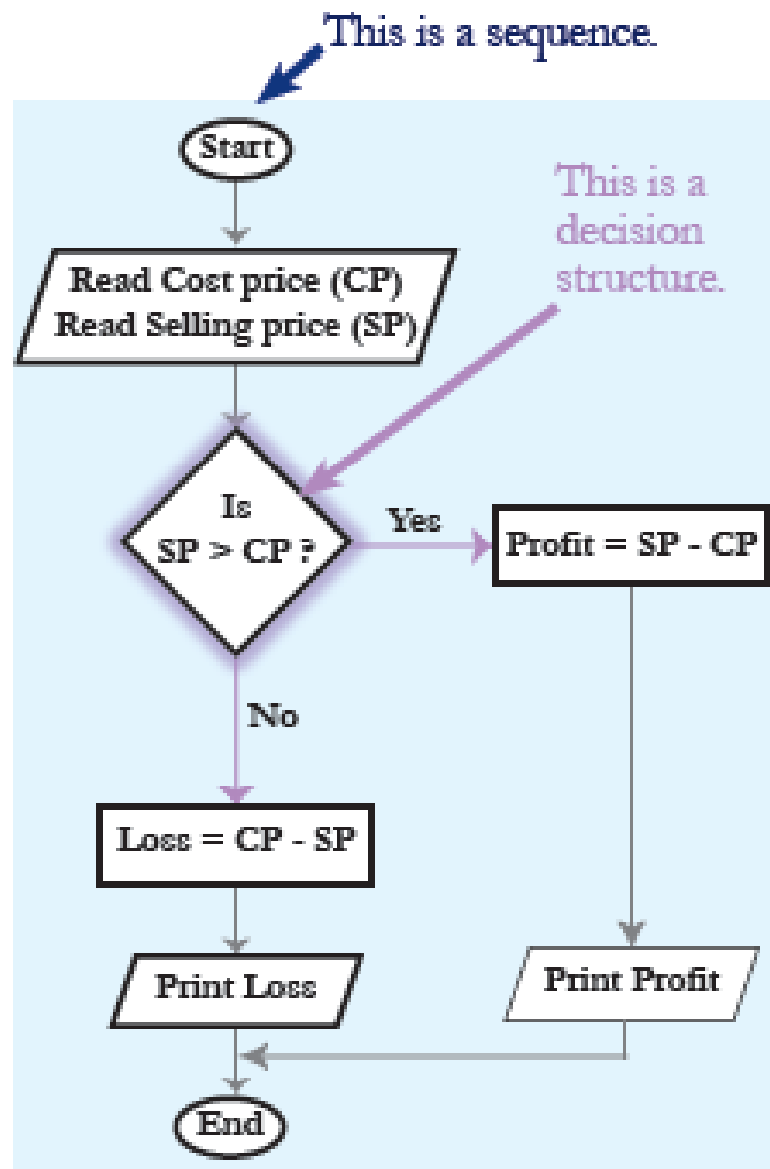
Flowchart



C++ Code

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

Flowchart - How to find profit or loss.



Finding profit or loss when
CP = 325 and SP = 458

Start

Read CP=325

Read SP= 458

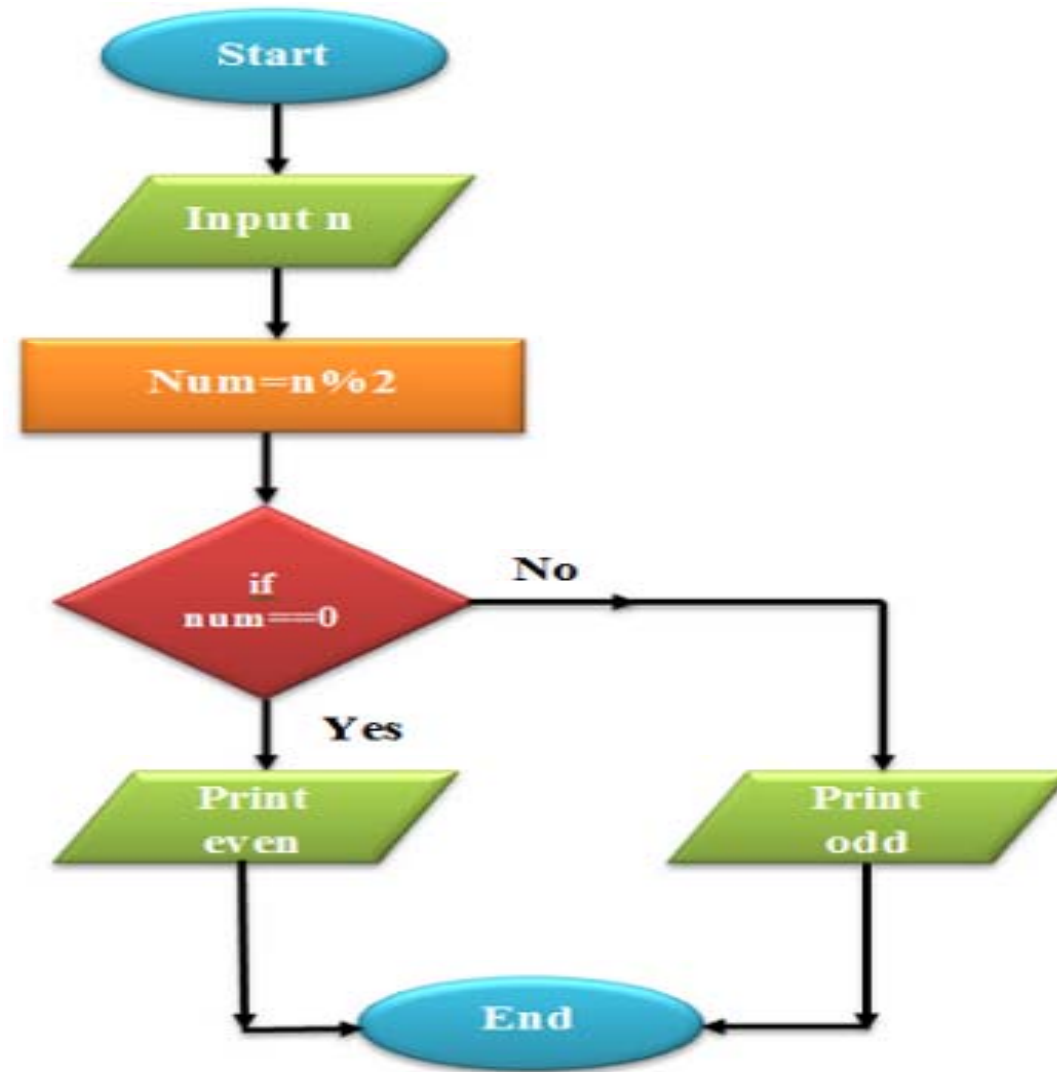
Condition:

Is 458 > 325? — Profit= 458-325

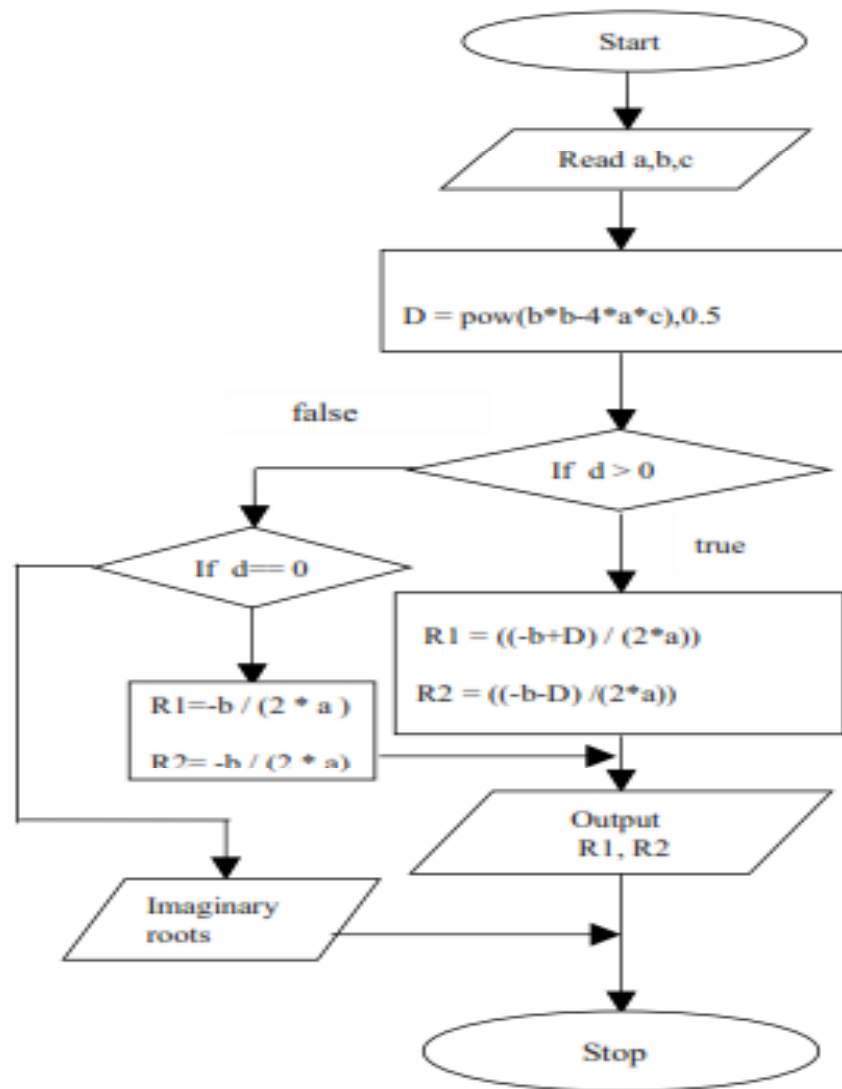
Profit= Rs. 133

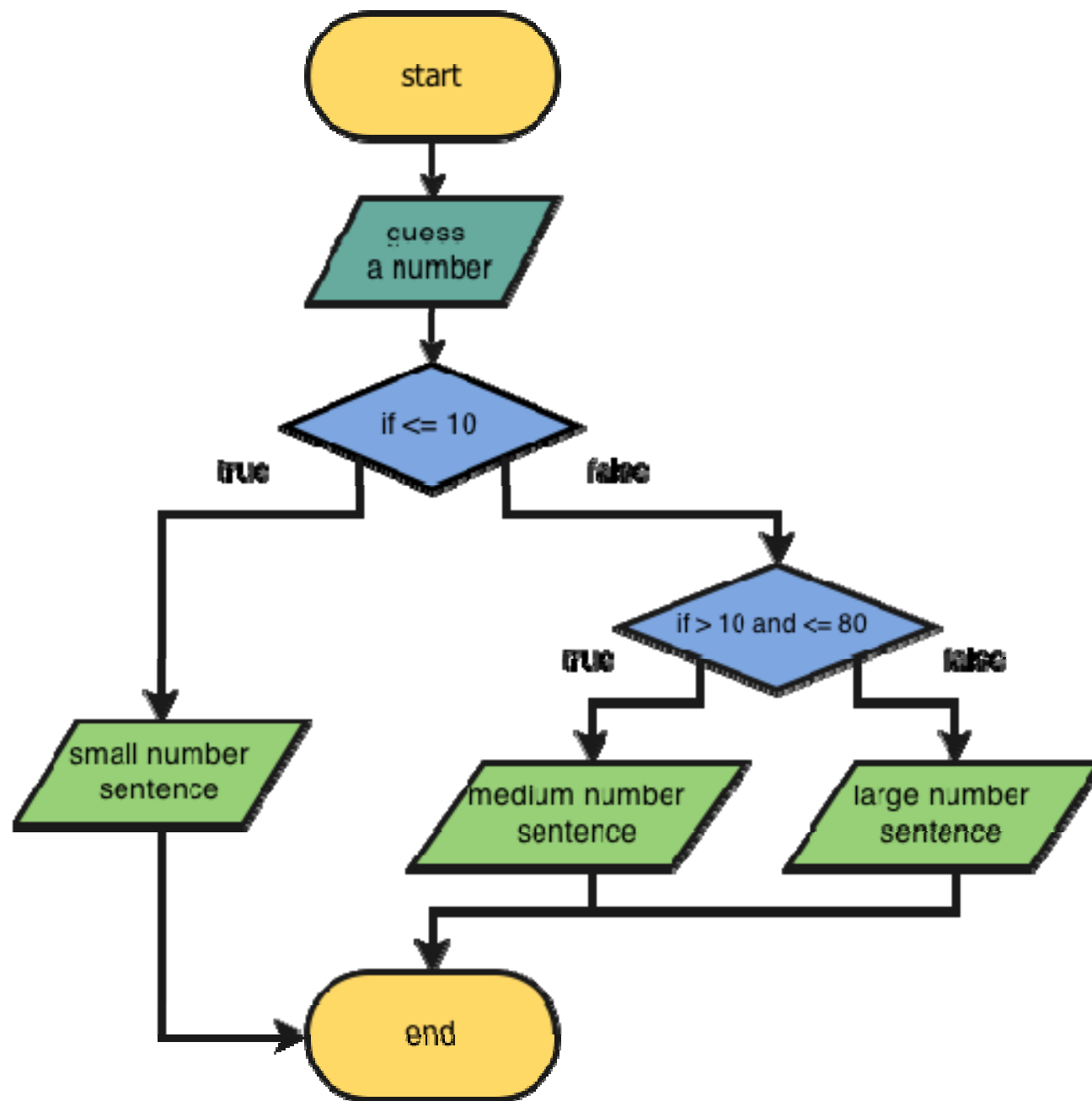
End

Even or odd

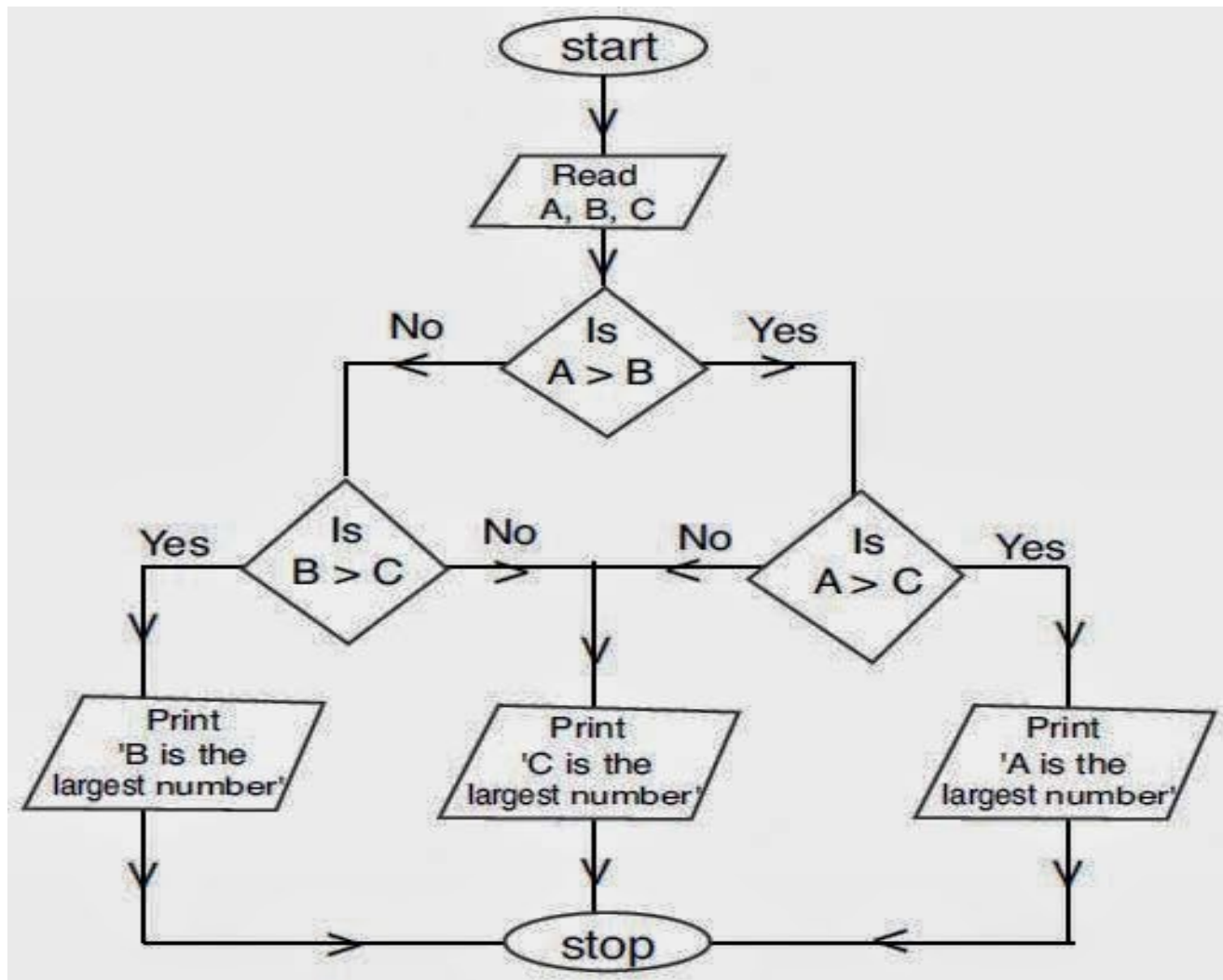


Roots of a quadratic equation

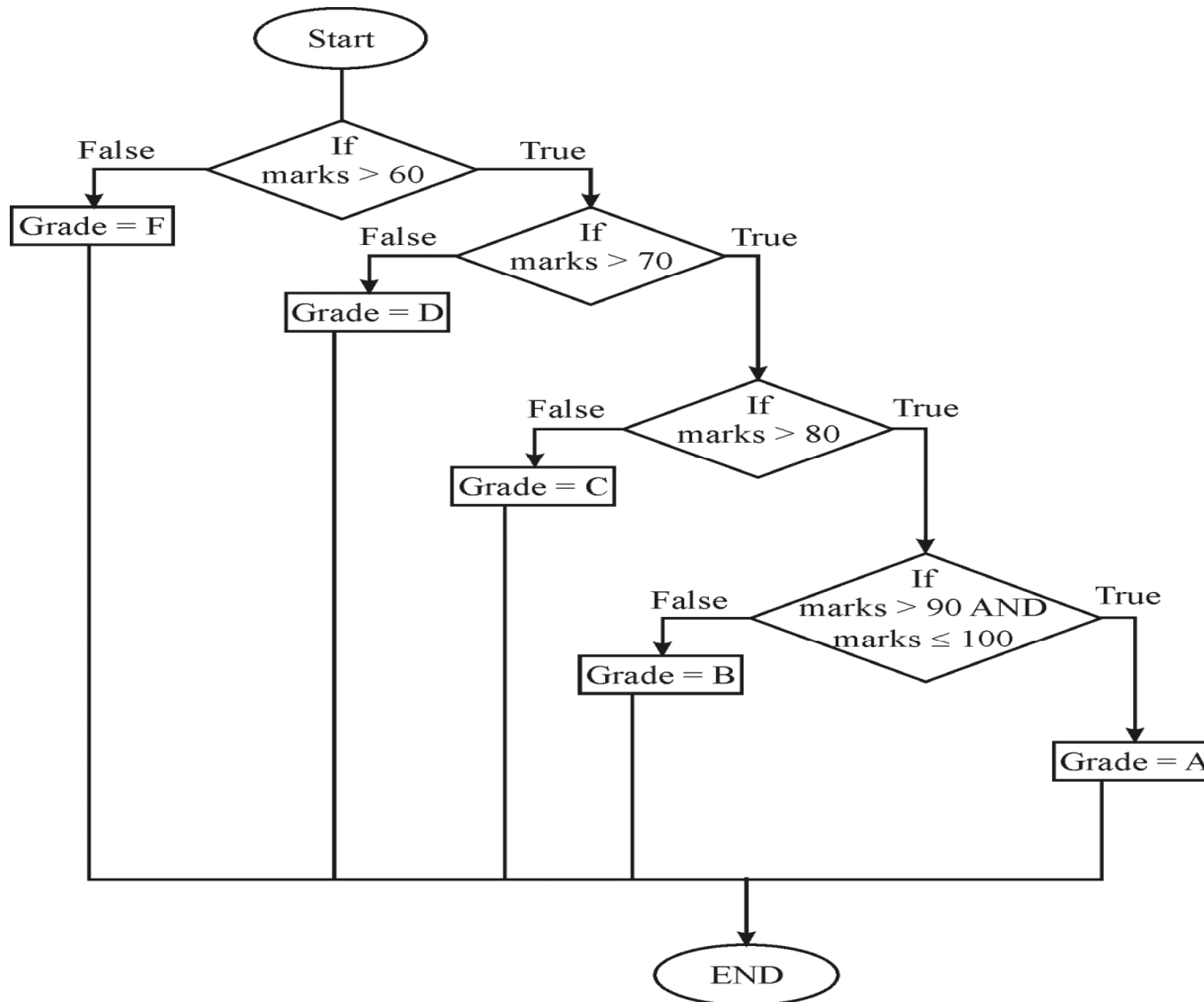




Largest of three numbers



Find grades of students



Example : Do it yourself

Bonus Schedule	
OVERTIME – (2/3)*ABSENT	Bonus Paid
>40 hours	\$50
>30 but \leq 40 hours	\$40
>20 but \leq 30 hours	\$30
>10 but \leq 20 hours	\$20
\leq 10 hours	\$10

Algorithms

Step 1: *Input* NAME,OVERTIME,ABSENT

Step 2: *if* (OVERTIME–(2/3)*ABSENT > 40) *then*

PAYMENT \leftarrow 50

else if (OVERTIME–(2/3)*ABSENT > 30) *then*

PAYMENT \leftarrow 40

else if (OVERTIME–(2/3)*ABSENT > 20) *then*

PAYMENT \leftarrow 30

else if (OVERTIME–(2/3)*ABSENT > 10) *then*

PAYMENT \leftarrow 20

else

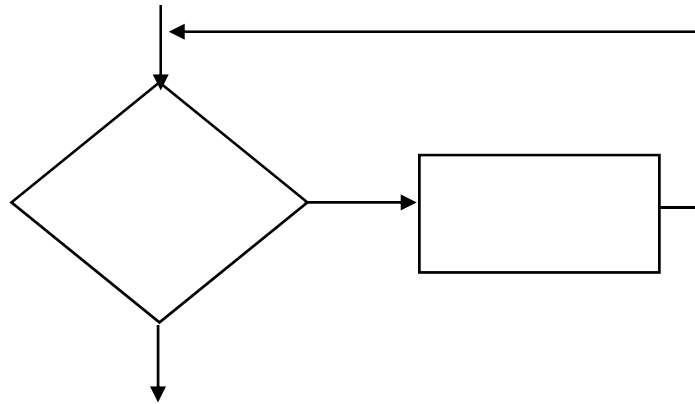
PAYMENT \leftarrow 10

endif

Step 3: *Print* “Bonus for”, NAME “is \$”, PAYMENT

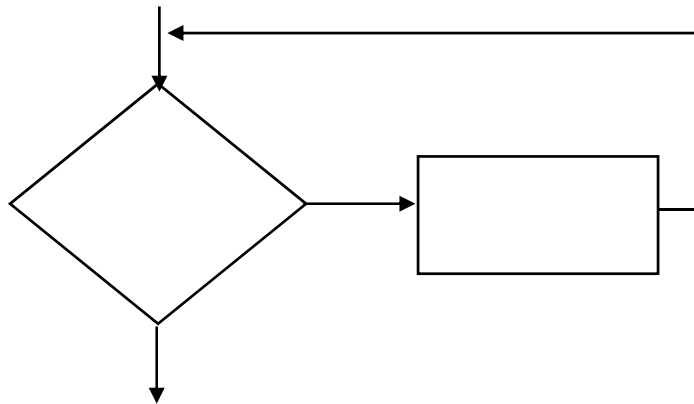
Repetition Structure

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



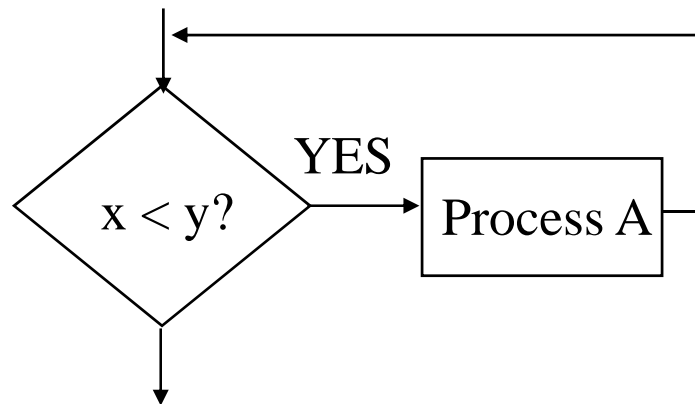
Repetition Structure

- 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 Structure

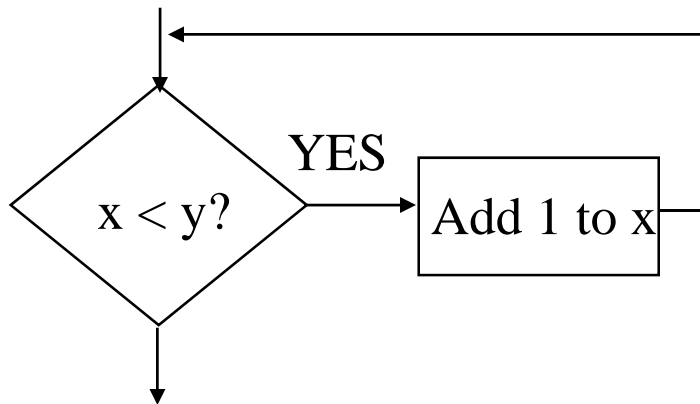
- In the flowchart segment, the question “is $x < y$?” is asked. If the answer is yes, then Process A is performed. The question “is $x < y$?” is asked again. Process A is repeated as long as x is less than y . When x is no longer less than y , the repetition stops and the structure is exited.



Repetition Structure

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

Flowchart

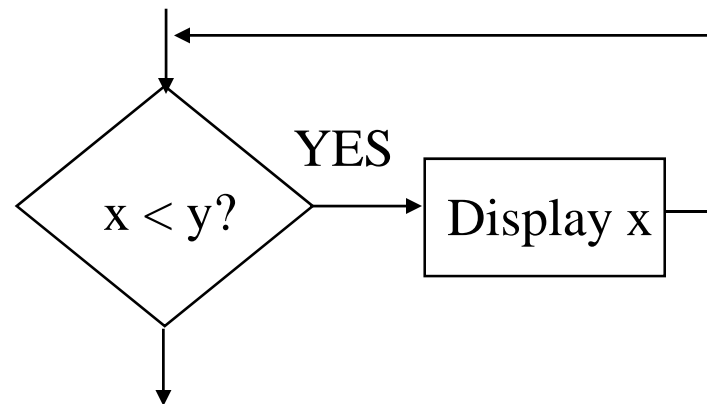


C++ Code

```
while (x < y)
    x++;
```

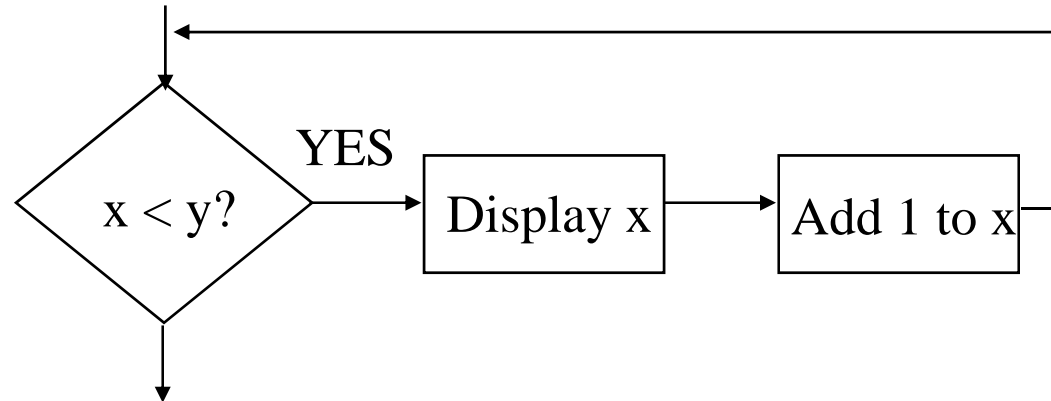
Controlling a Repetition Structure

- 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?



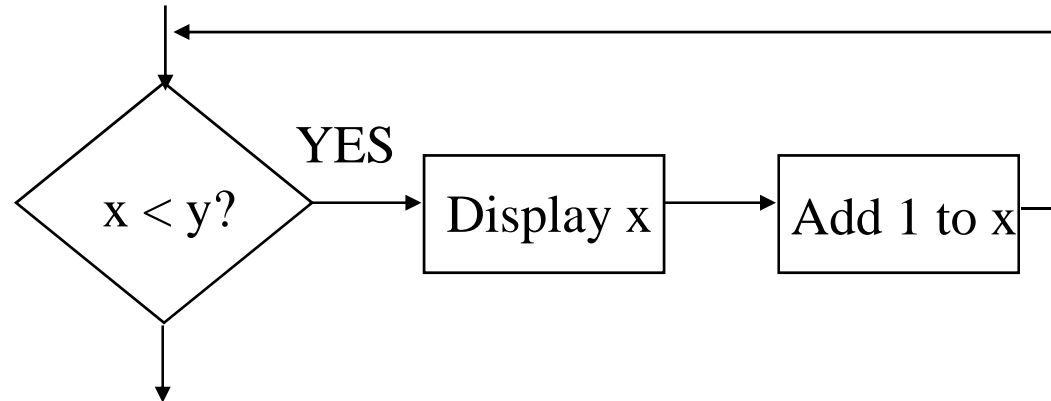
Controlling a Repetition Structure

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



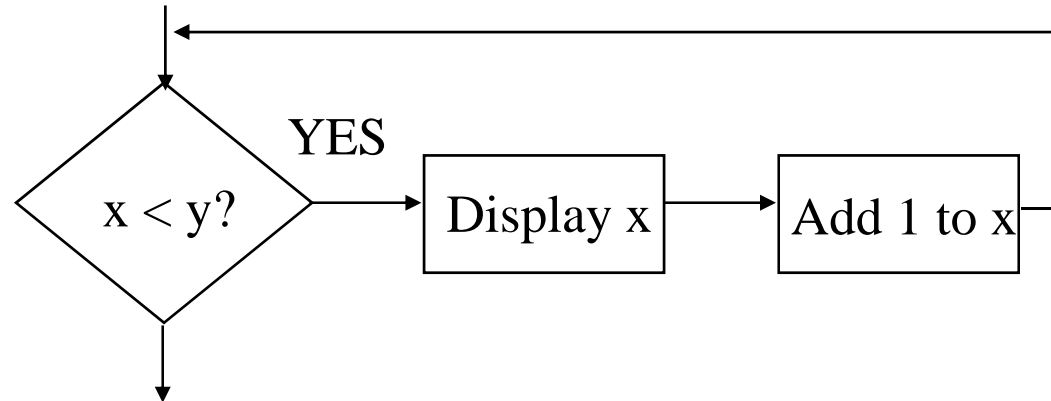
A Pre-Test Repetition Structure

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



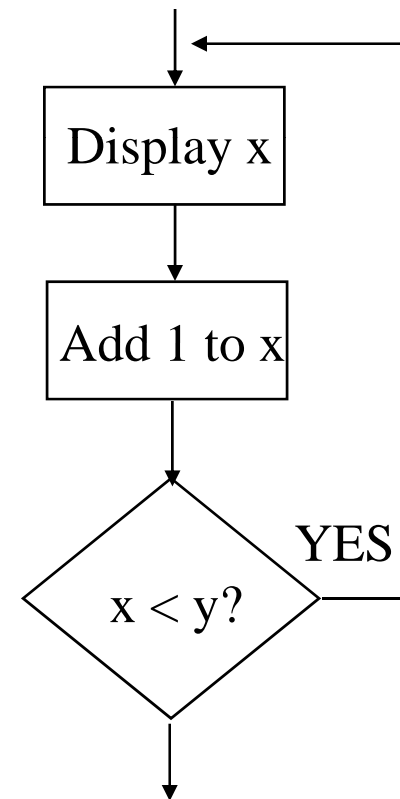
A Pre-Test Repetition Structure

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



A Post-Test Repetition Structure

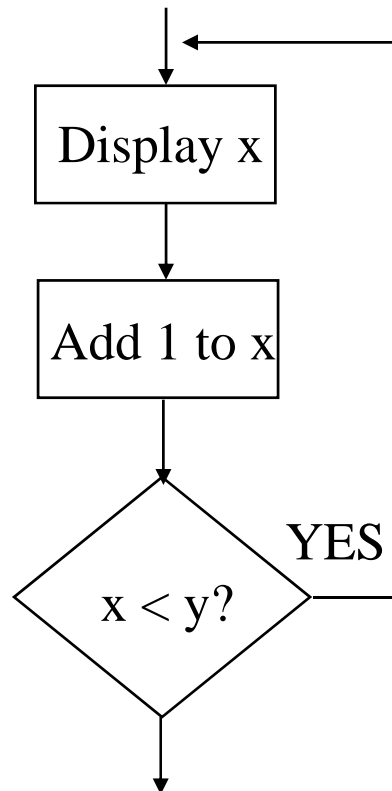
- This flowchart segment shows a post-test repetition structure.
- The condition is tested *AFTER* the actions are performed.
- A post-test repetition structure always performs its actions at least once.



A Post-Test Repetition Structure

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

Flowchart



C++ Code

```
do
{
    cout << x << endl;
    x++;
} while (x < y);
```


factorial

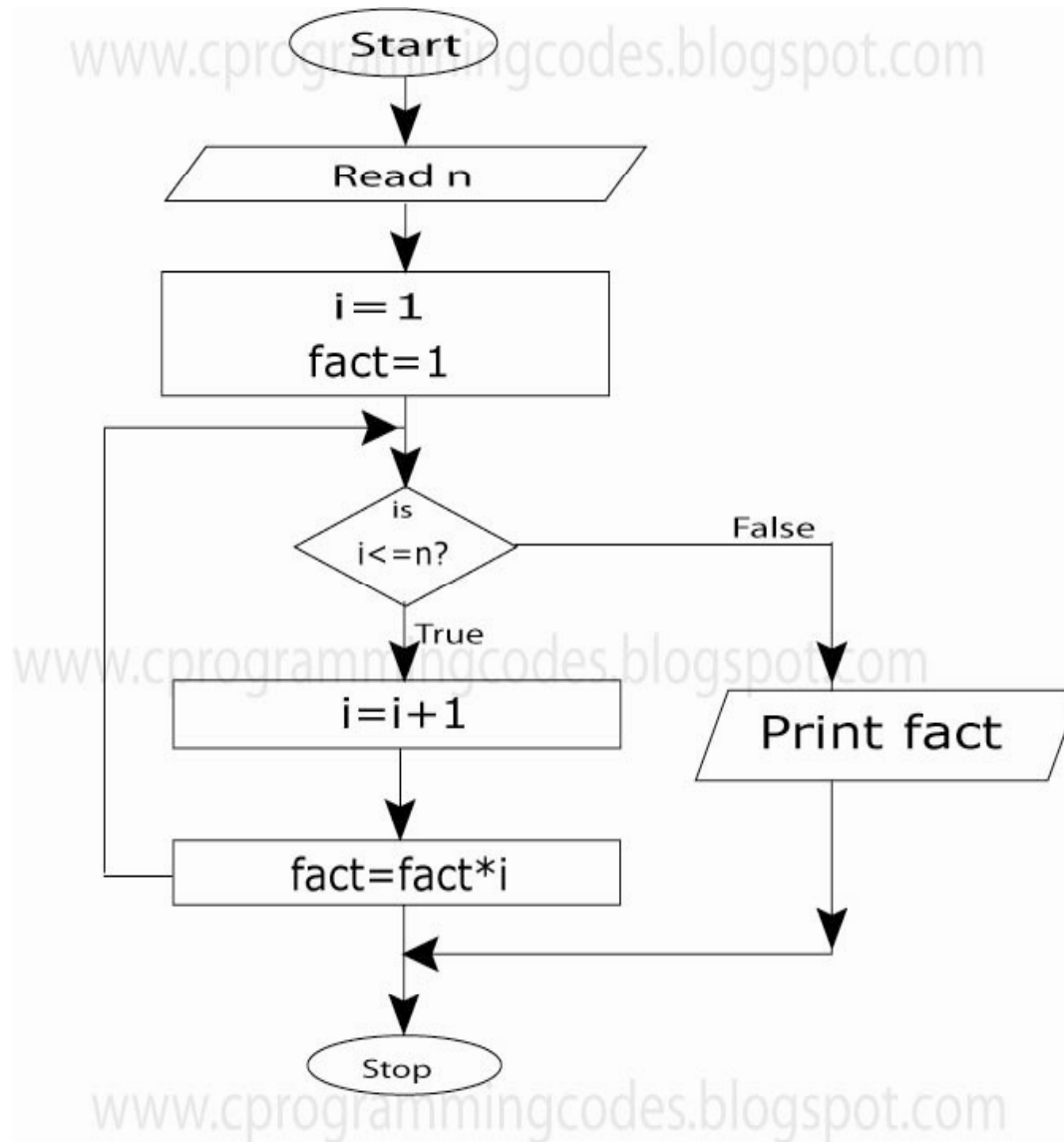
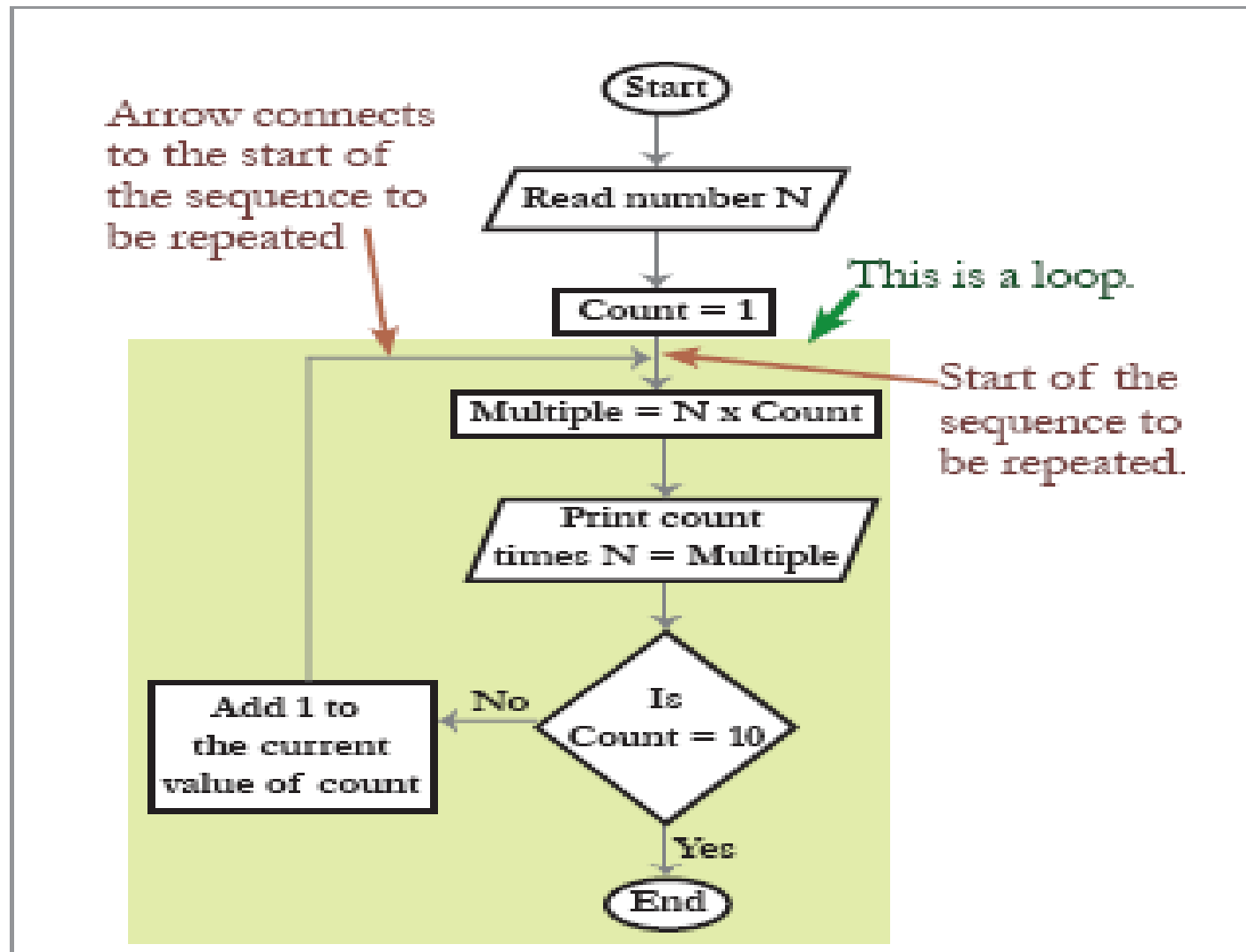
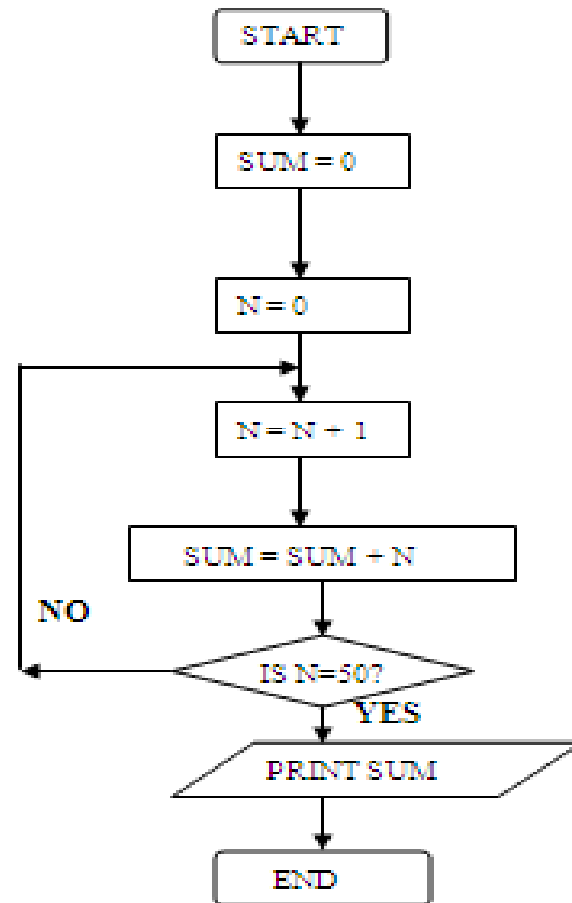


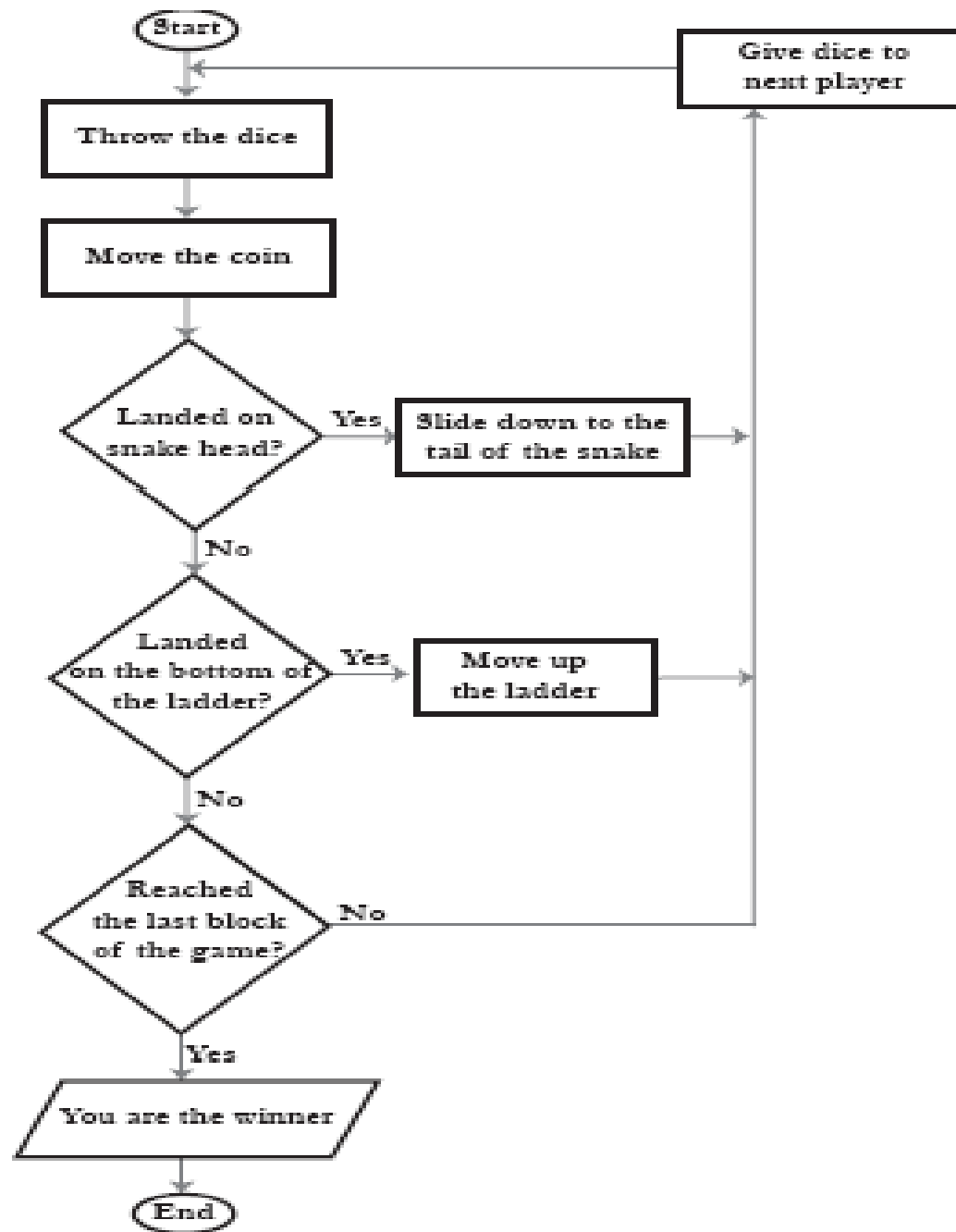
Table of a number

Flowchart of multiplication table for a given number N



Print sum of first 50 natural numbers



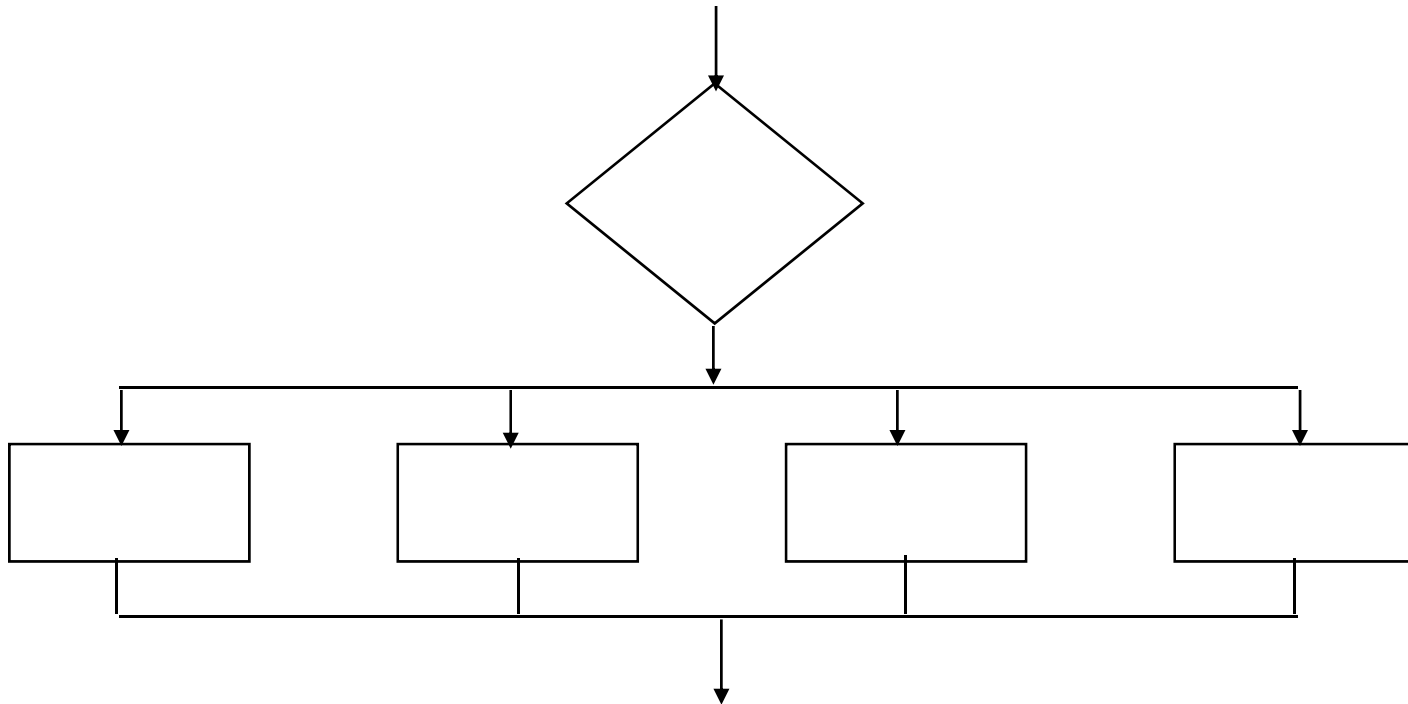


Practice

- Draw a flowchart to read a number N and print all its divisors.
- Draw a flowchart for computing the sum of the digits of any given number

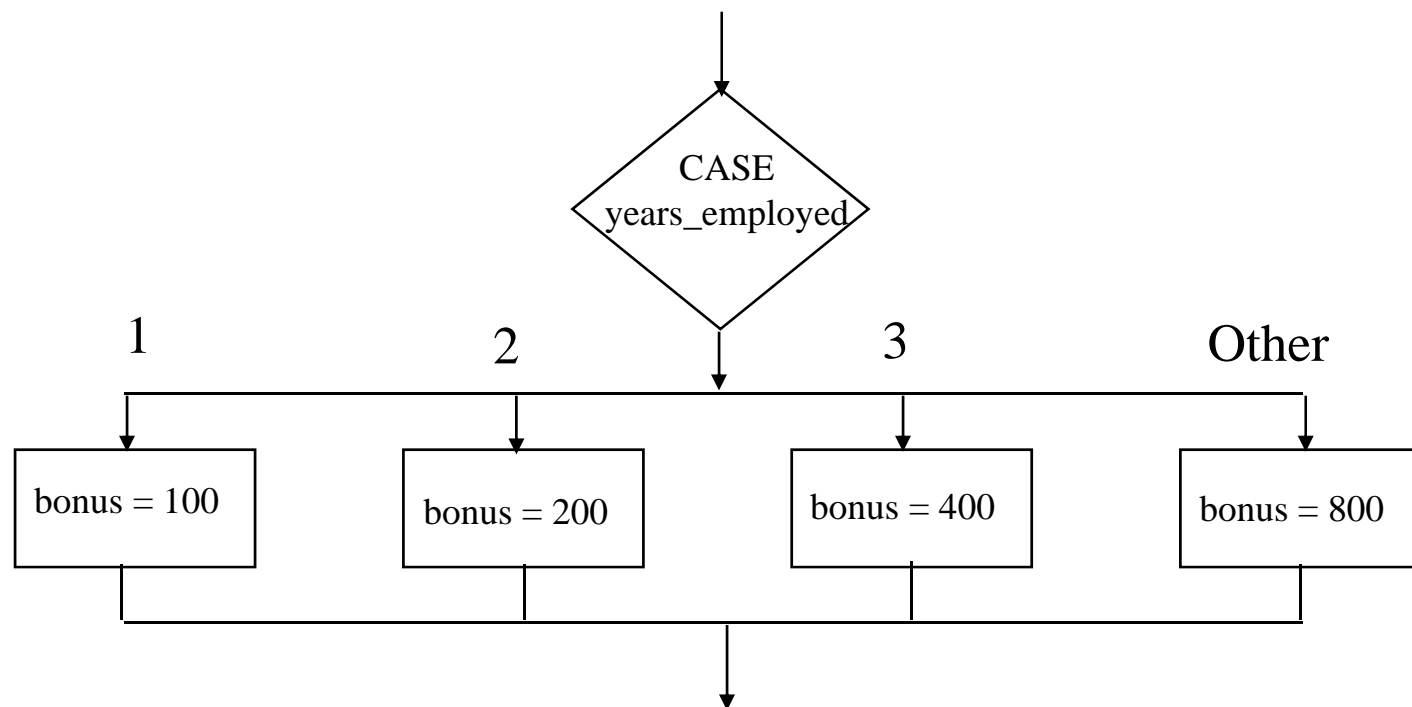
Case Structure

- One of several possible actions is taken, depending on the contents of a variable.

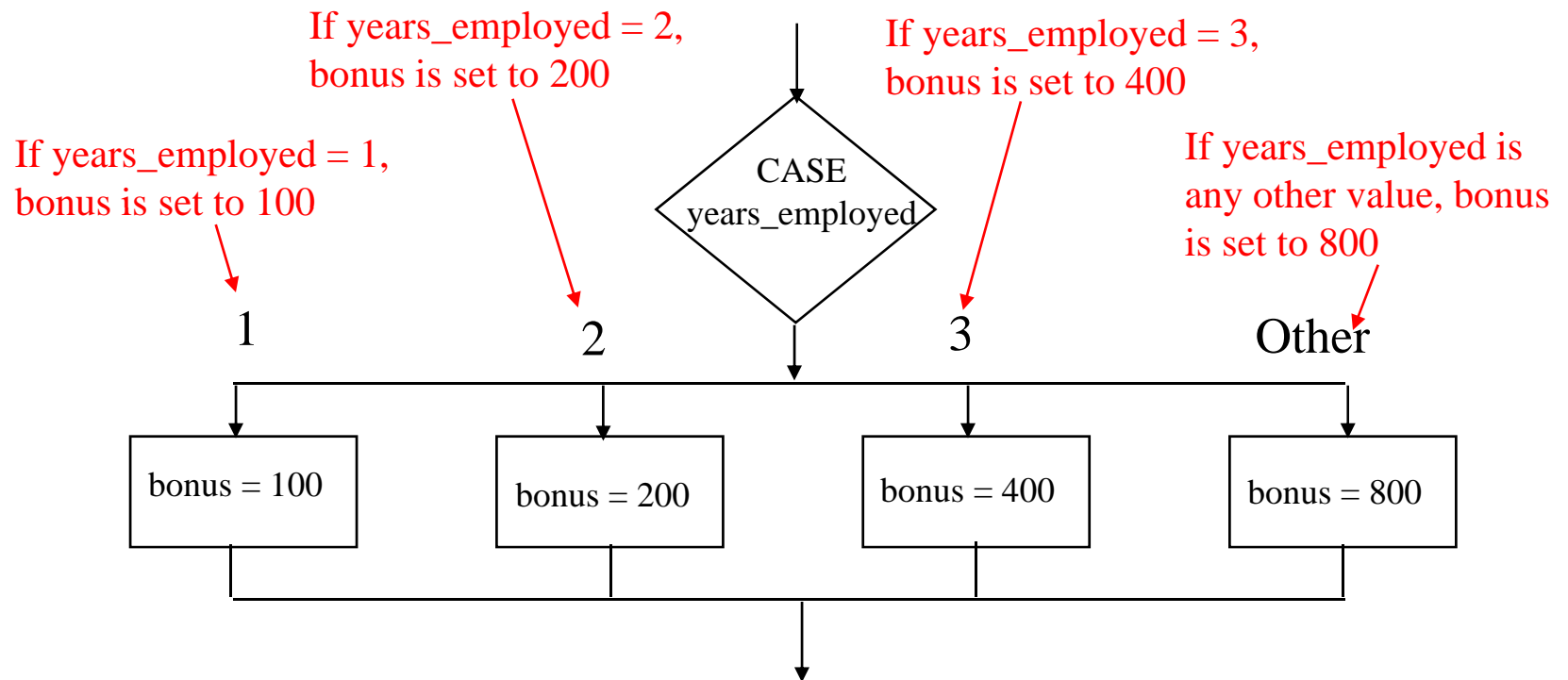


Case Structure

- The structure below indicates actions to perform depending on the value in `years_employed`.

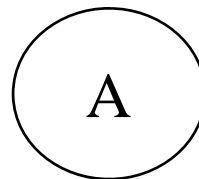


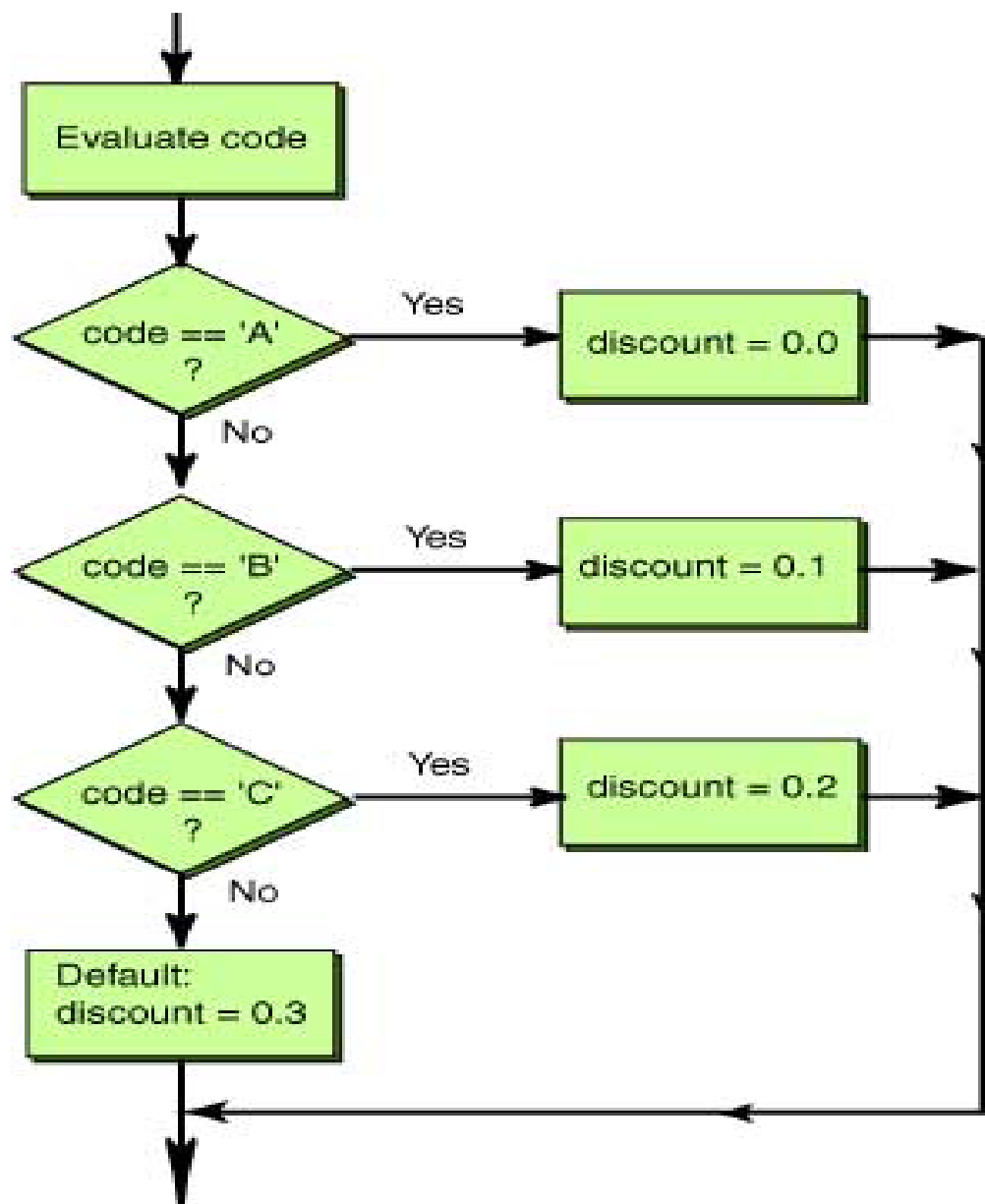
Case Structure



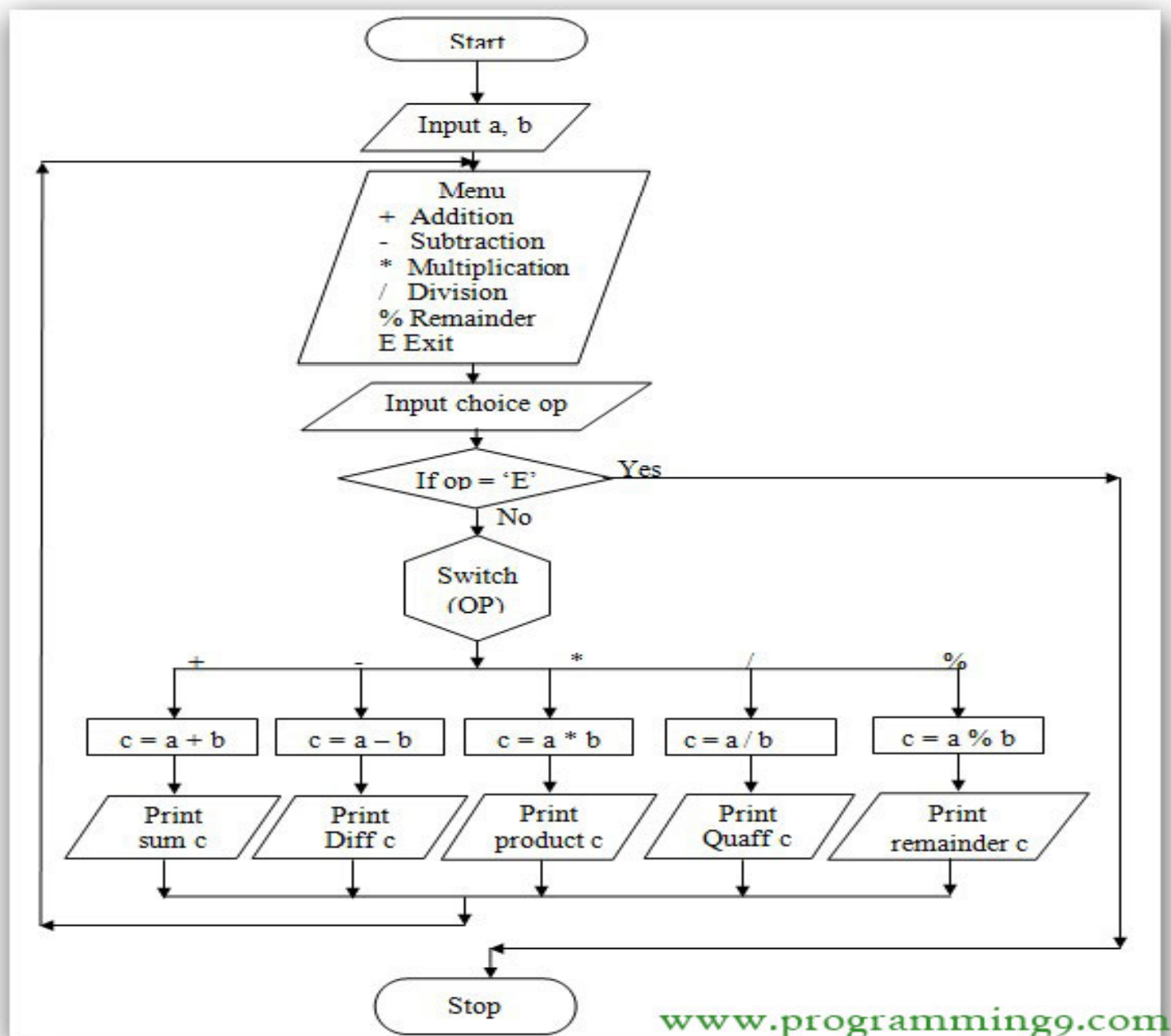
Connectors

- Sometimes a flowchart will not fit on one page.
- A connector (represented by a small circle) allows you to connect two flowchart segments.





Menu driven program of a calculator



Connectors

- The “A” connector indicates that the second flowchart segment begins where the first segment ends.

