CHAPTER 2

PROBLEM SOLVING

- This chapter will cover the following topics:
  - Problem Solving Concepts for the Computer
  - Pre-Programming Phase
  - Programming Or Implementation Phase

What Problem Can Be Solved By Computer

- When the solution can be produced by a set of step-by-step procedures or actions.
- This step-by-step action is called an *algorithm*.
- The algorithm will process some inputs and produced output.
- Solving problem by computer undergo two phases:
- Phase 1:
    - Organizing the problem or pre-programming phase.
- Phase 2:
    - Programming phase.

PRE-PROGRAMMING PHASE

- This phase requires five steps:
  - Analyzing the problem.
  - Developing the Hierarchy Input Process Output (HIPO) chart or Interactivity Chart (IC).
  - Developing the Input-Process-Output (IPO) Chart.
  - Drawing the Program flowcharts.
  - Writing the algorithms.

PRE-PROGRAMMING PHASE

- **Analyzing The Problem**
  - Understand and analyze the problem to determine whether it can be solved by a computer.
  - Analyze the requirements of the problem.
  - Identify the following:
    - Data requirement.
    - Processing requirement or procedures that will be needed to solve the problem.
    - The output.

PRE-PROGRAMMING PHASE

- All These requirements can be presented in a Problem Analysis Chart (PAC)

Data	Processing	Output
given in the problem or provided by the user	List of processing required or procedures.	Output requirement.

PRE-PROGRAMMING PHASE

- Example: **Payroll Problem**
- Calculate the salary of an employee who works by hourly basis. The formula to be used is
- Salary = Hour works * Pay rate

Data	Processing	Output
Hours work, Pay rate	Salary = Hours work * payrate	Salary

Problem 1

Write a Problem Analysis Chart (PAC) to convert the distance in miles to kilometers

where 1.609 kilometers per mile.

Data	Processing	Output
Distance in miles	Kilometers = 1.609 x miles	Distance in kilometers

Problem 2

Write a Problem Analysis Chart (PAC) to find an area of a circle where area = pi * radius * radius

Data	Processing	Output
radius	area = 3.14 x radius x radius	area

Problem 3

Write a Problem Analysis Chart (PAC) to compute and display the temperature
inside the earth in Celsius and Fahrenheit. The relevant formulae are

Celsius = 10 x (depth) + 20

Fahrenheit = 1.8 x (Celsius) + 32

Data	Processing	Output
depth	celsius = 10 x (depth) + 20 fahrenheit = 1.8 x (celsius) + 32	Display celsius, Display fahrenheit

Problem 4

Write a problem analysis chart (PAC) that asks a user to enter the distance of a trip in miles, the miles per gallon estimate for the user's car, and the average cost of a gallon of gas. Calculate and display the number of gallons of gas needed and the estimated cost of the trip.

Data	Processing	Output
distance, miles per gallon, cost per gallon	gas needed = distance / miles per gallon. estimated cost = cost per gallon x gas needed	Display gas needed Display estimated cost

PRE-PROGRAMMING PHASE

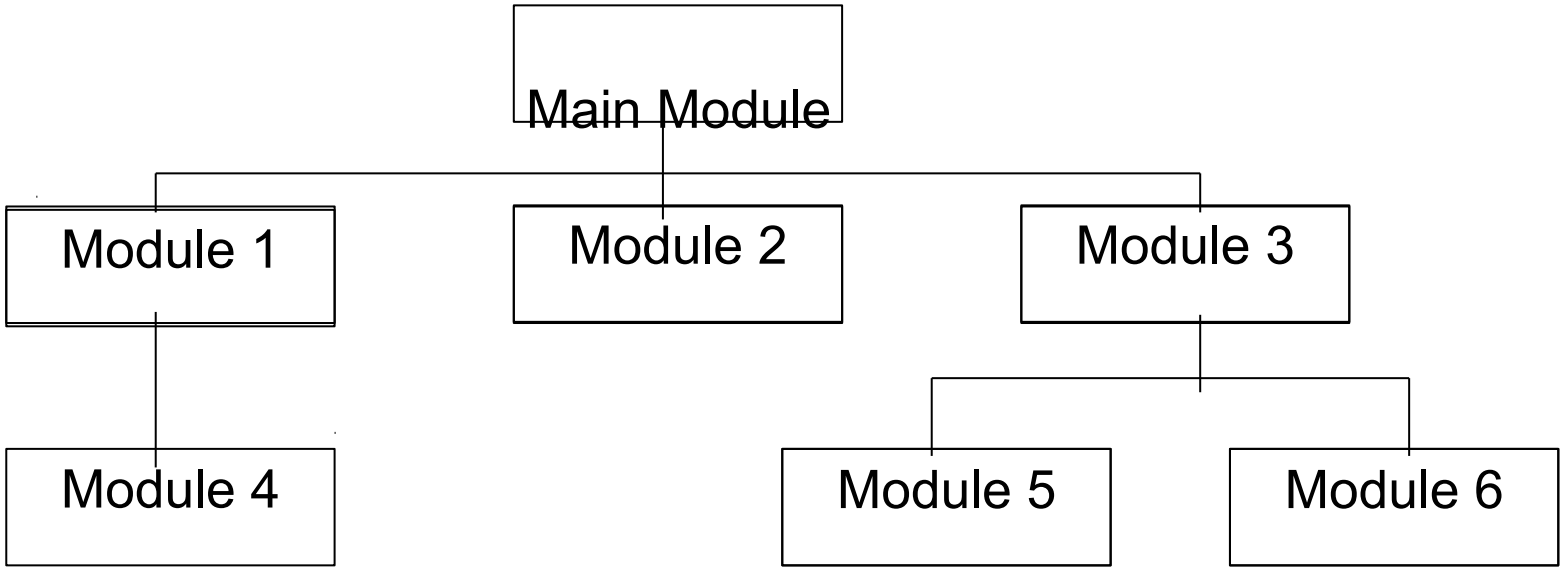
- **Developing the Hierarchy Input Process Output (HIPO) or Interactivity Chart**
  - The problem is normally big and complex.
  - Thus, requires big program.
  - Thus, the processing can be divided into subtasks called modules.
  - Each module accomplishes one function.
  - These modules are connected to each other to show the interaction of processing between the modules.

PRE-PROGRAMMING PHASE

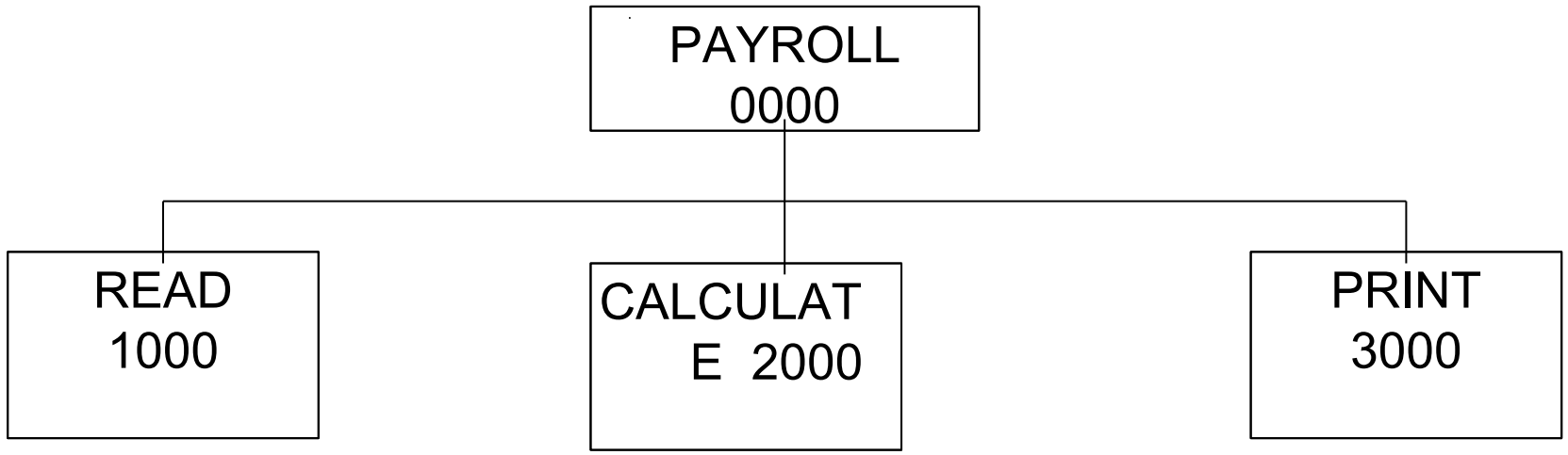
- Main/control module controls the flow all other modules.
- The IC is developed using top-down-method: top to down left to right order (also refer to order of processing).
- Modules are numbered, marked for duplication, repetition or decision.

PRE-PROGRAMMING PHASE

- The interaction will form a hierarchy, called Hierarchy Input Process Output Chart (HIPO) or Interactivity Chart (IC). Programming which use this approach (problem is divided into subtasks) is called *Structured Programming*.



PRE-PROGRAMMING PHASE

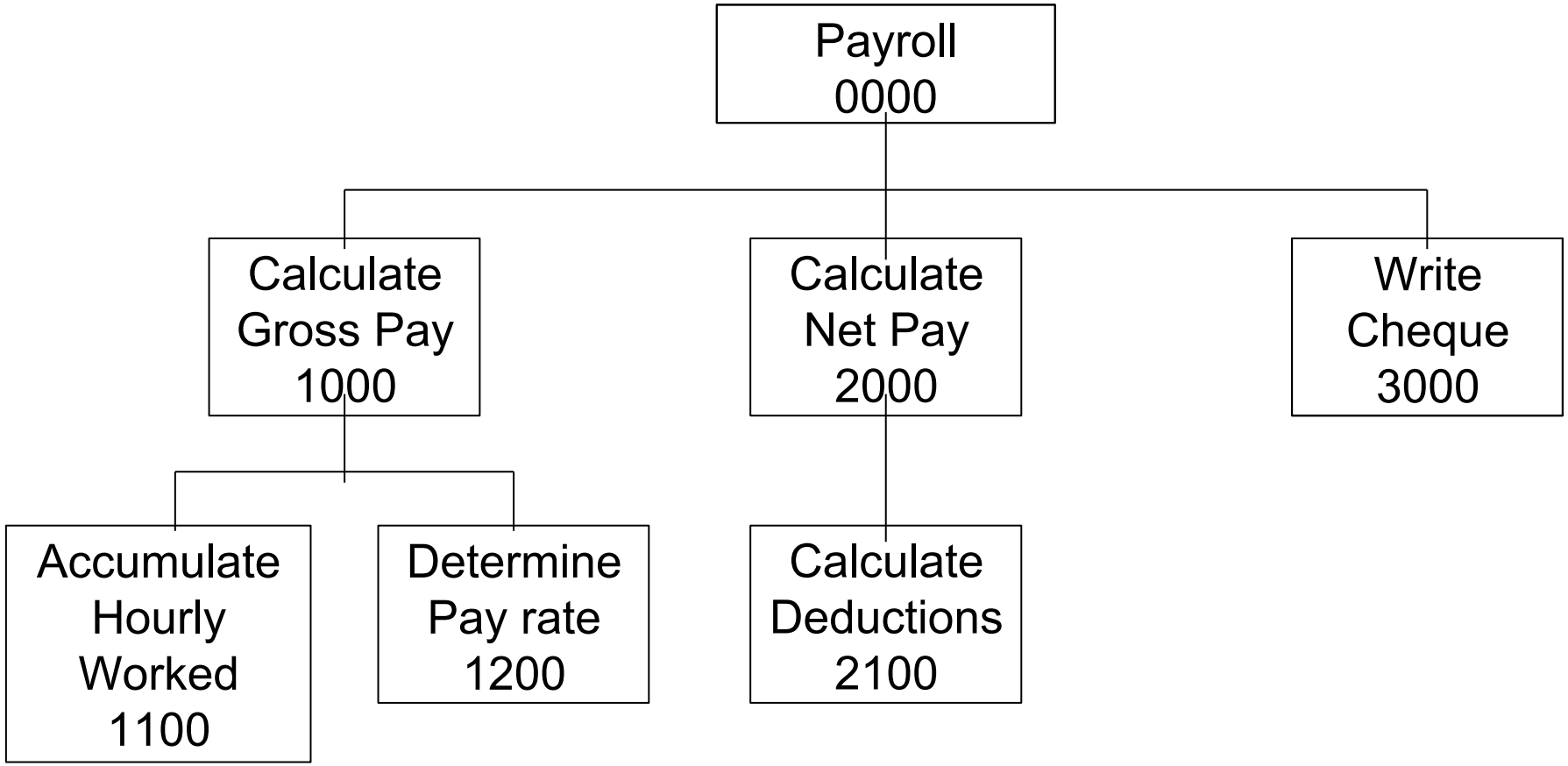


PRE-PROGRAMMING PHASE

Example 2.2: Extended Payroll Problem

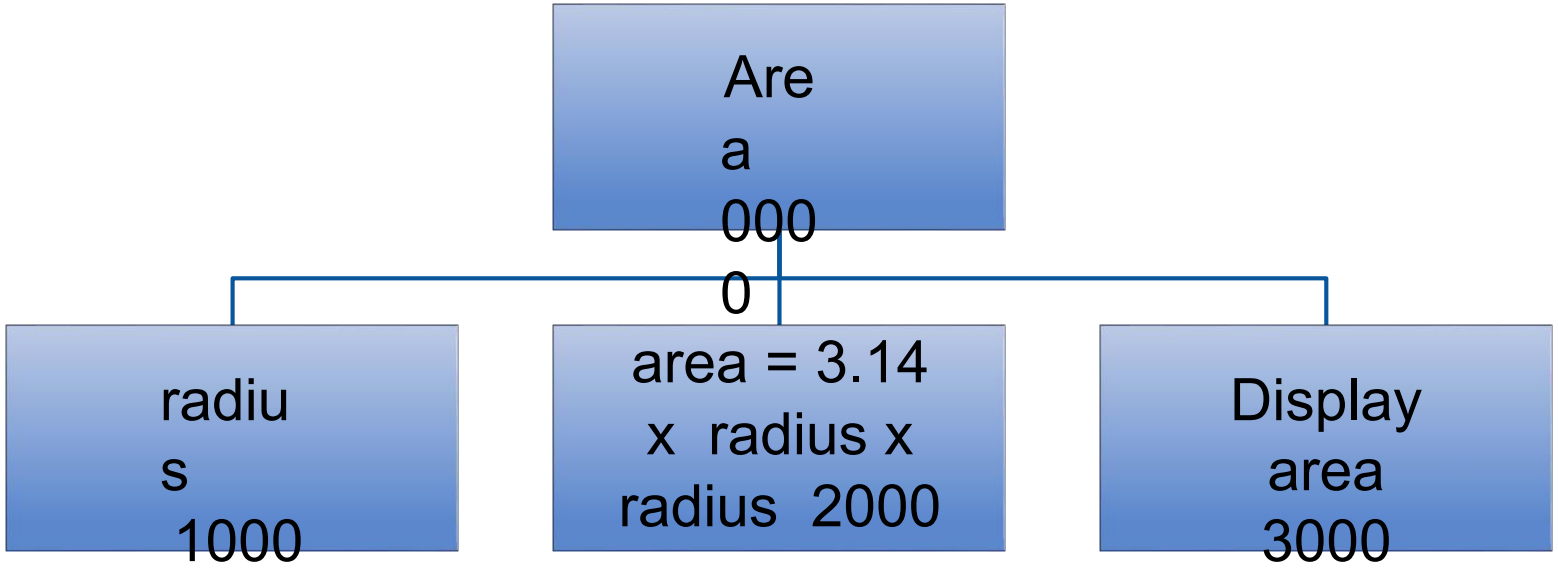
You are required to write a program to calculate both the gross pay and the net pay of every employee of your company. To determine the gross pay, you have to multiply the accumulated total hours worked by the employee, by the appropriate pay rate. The program should print the cheque that tells the total net pay. The net pay is calculated by subtracting the gross pay with any deductions that may be incurred by the employee.

PRE-PROGRAMMING PHASE



Problem 2

Write a Hierarchy Input Process Output (HIPO) to find an area of a
circle where area = pi * radius * radius

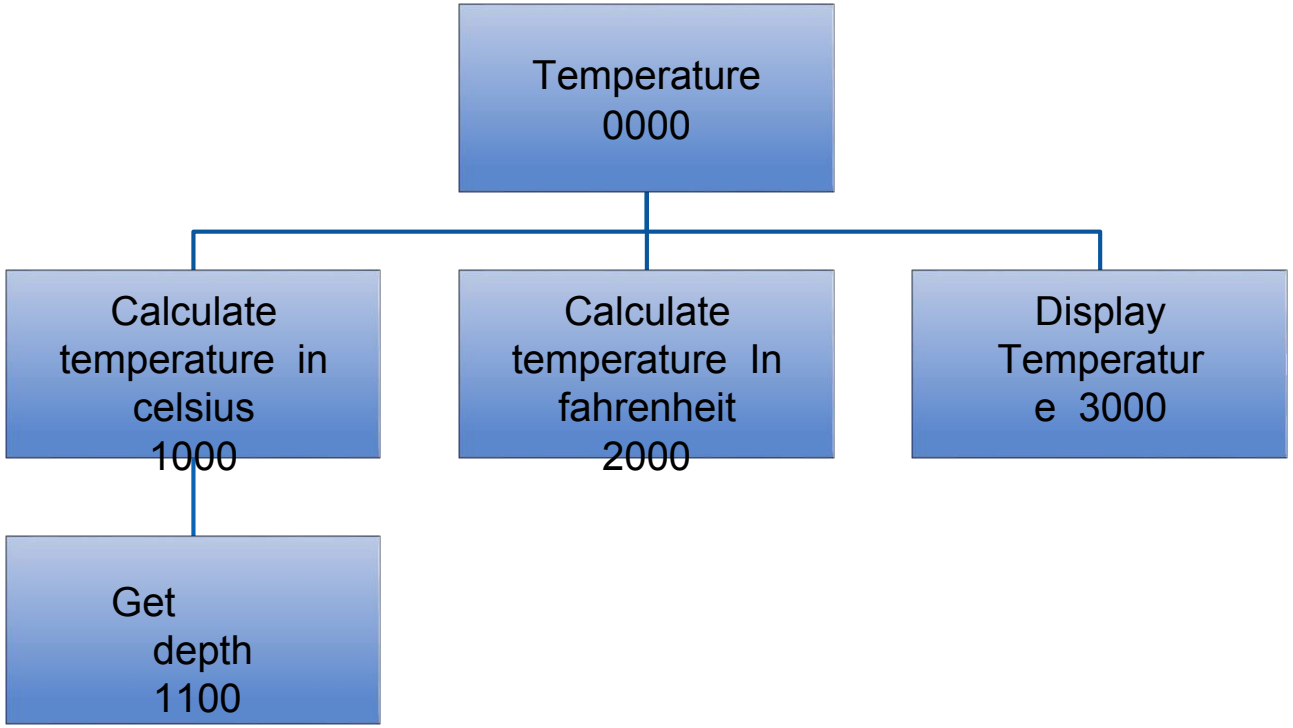


Problem 3

Write a Hierarchy Input Process Output (HIPO) to compute and display the temperature inside the earth in Celsius and Fahrenheit. The relevant formulas are

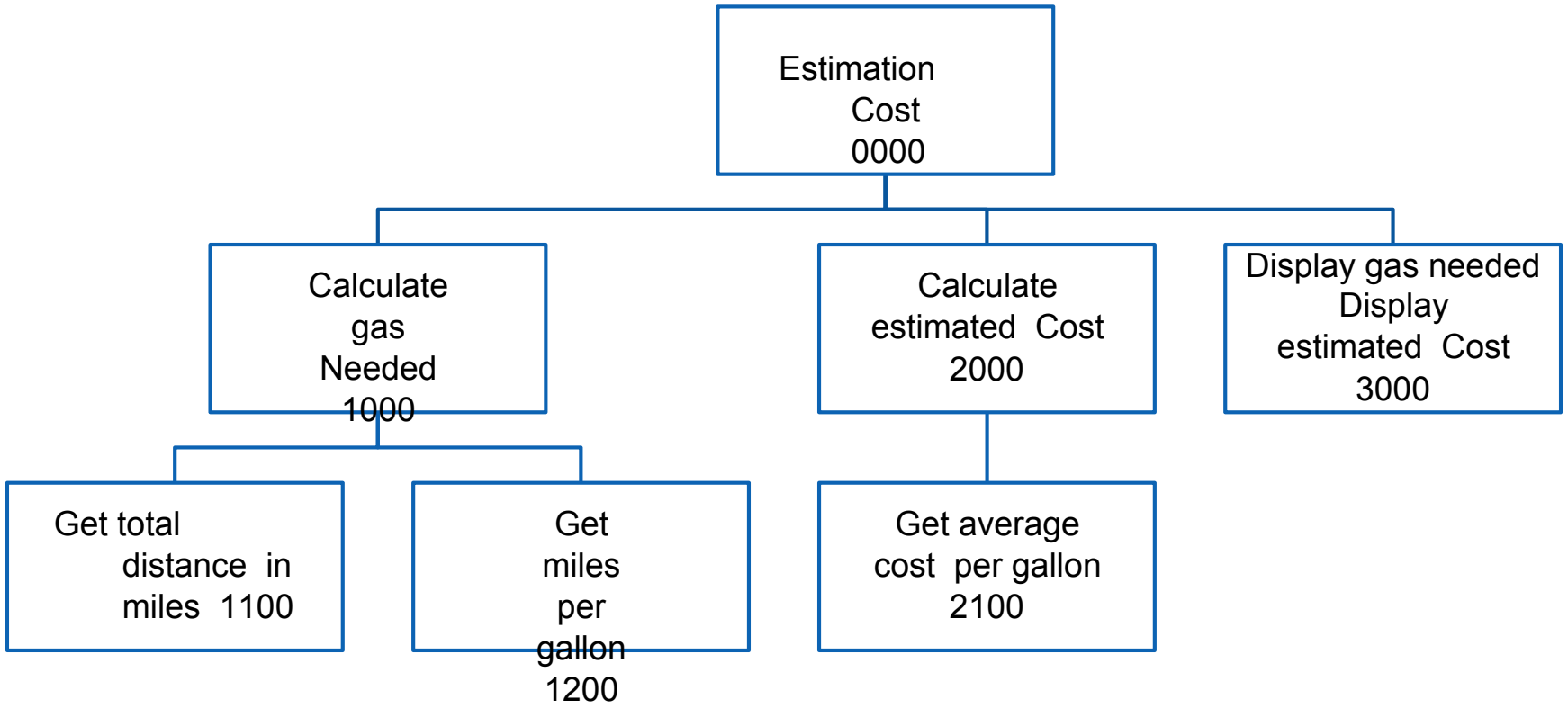
Celsius = 10 x (depth) + 20

Fahrenheit = 1.8 x (Celsius) + 32



Problem 4

Write a Hierarchy Input Process Output (HIPO) that asks a user to enter the distance of a trip in miles, the miles per gallon estimate for the user's car, and the average cost of a gallon of gas. Calculate and display the number of gallons of gas needed and the estimated cost of the trip.


Estimation
Cost
0000
Calculate
gas
Needed
1000
Calculate
estimated Cost
2000
Display gas needed
Display
estimated Cost
3000
Get total
distance in
miles 1100
Get
miles
per
gallon
1200
Get average
cost per gallon
2100

PRE-PROGRAMMING PHASE

- **Developing the Input Process Output (IPO) Chart**
  - Extends and organizes the information in the Problem Analysis Chart.
  - It shows in more detail what data items are input, what are the processing or modules on that data, and what will be the result or output.
  - It combines information from PAC and HIPO Chart.

PRE-PROGRAMMING PHASE

Input	Processing	Module	Output
-Hours Worked - Pay Rate -Deduction	-Enter Hourly Worked -Enter Pay Rate -Calculate Gross Pay -Enter Deductions -Calculate Net Pay -Print Cheque -End	1100 1200 1000 2100 2000 3000 0000	-Net pay

Problem 2

Write a Input Process Output (IPO) to find an area of a circle

where area = pi * radius * radius

Input	Processing	Module	Output
- radius	- Enter radius - area = 3.14 x radius x radius -Display area -end	1000 2000 3000 0000	- Area of a circle

Problem 3

Write an Input Process Output (IPO) that asks a user to enter the distance of a trip in miles, the miles per gallon estimate for the user's car, and the average cost of a gallon of gas. Calculate and display the number of gallons of gas needed and the estimated cost of the trip.

Input	Processing	Module	Output
-Distance in miles - Miles per gallon -Cost gas per gallon	- Enter distance - Enter miles per gallon -Calculate total gas needed -Enter cost gas per gallon - Calculate estimated cost -Display total gas and estimated cost - End	1100 1200 1000 2100 2000 3000 0000	-Total gas needed - Estimated cost

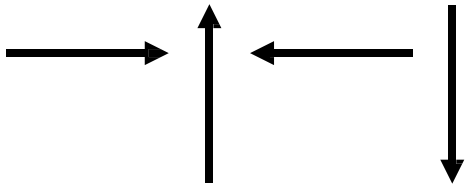
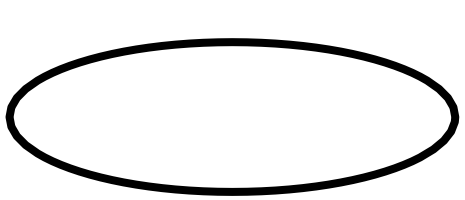
PRE-PROGRAMMING PHASE

- **Drawing the Program Flowcharts**
  - Flowchart is the graphic representations of the individual steps or actions to implement a particular module.
  - The flowchart can be likened to the blueprint of a building. An architect draws a blueprint before beginning construction on a building, so the programmer draws a flowchart before writing a program.
  - Flowchart is independent of any programming language.

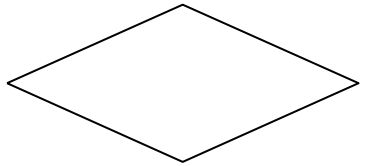
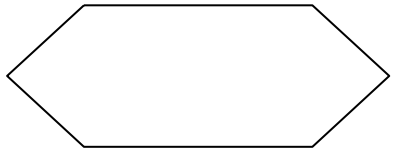
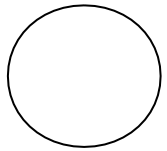
PRE-PROGRAMMING PHASE

- Flowchart is the logical design of a program.
- It is the basis from which the actual program code is developed.
- Flowchart serves as documentation for computer program.
- The flowchart must be drawn according to definite rules and utilizes standard symbols adopted internationally.
- The International Organization for Standardization (IOS) was the symbols shown below (You can draw the symbols using ready-made flowcharting template):

PRE-PROGRAMMING PHASE

Symbol	Function
	Show the direction of data flowor logical solution.
	Indicate the beginning and ending of a set of actions or instructions (logical flow) of a module or program.
	Indicate a process, such as calculations, opening and closing files.

PRE-PROGRAMMING PHASE

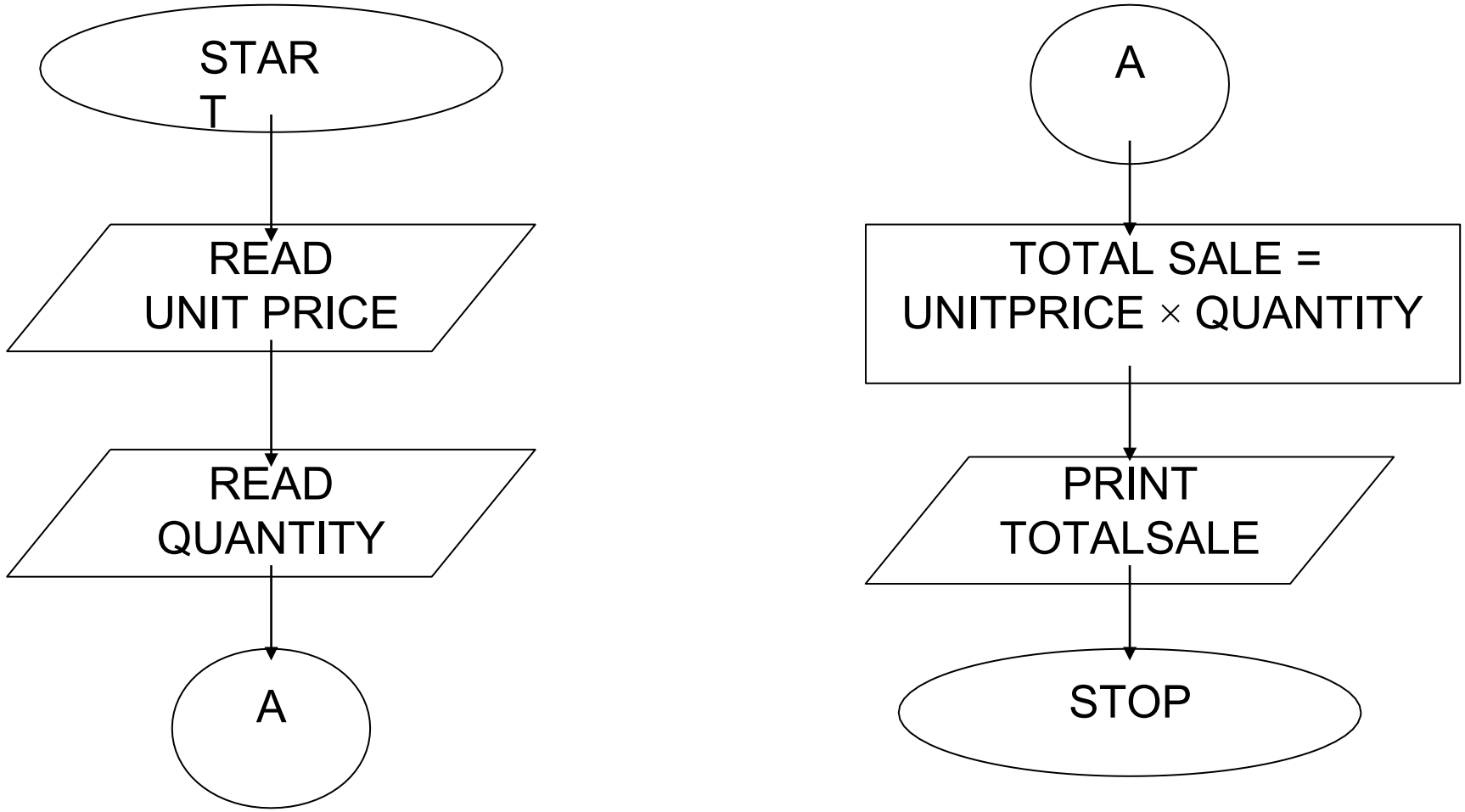
	Indicate input to the program and output from the program.
	Use for making decision. Either True or False based on certain condition.
	Use for doing a repetition or looping of certain steps.
	Connection of flowchart on the same page.
	Connection of flowchart from page to page.

PRE-PROGRAMMING PHASE

Example 2.3 : Sale Problem

- Draw a flowchart for a problem that to read two numbers. The first number represents the unit price of a product and the second number represents the quantity of the product sold. Calculate and print the total sale.
- Solution: Stepwise Analysis of the Sale Problem
    - Start of processing
    - Read the unit price
    - Read the quantity
    - Calculate total sale
    - Print total sale
    - Stop the processing

PRE-PROGRAMMING PHASE

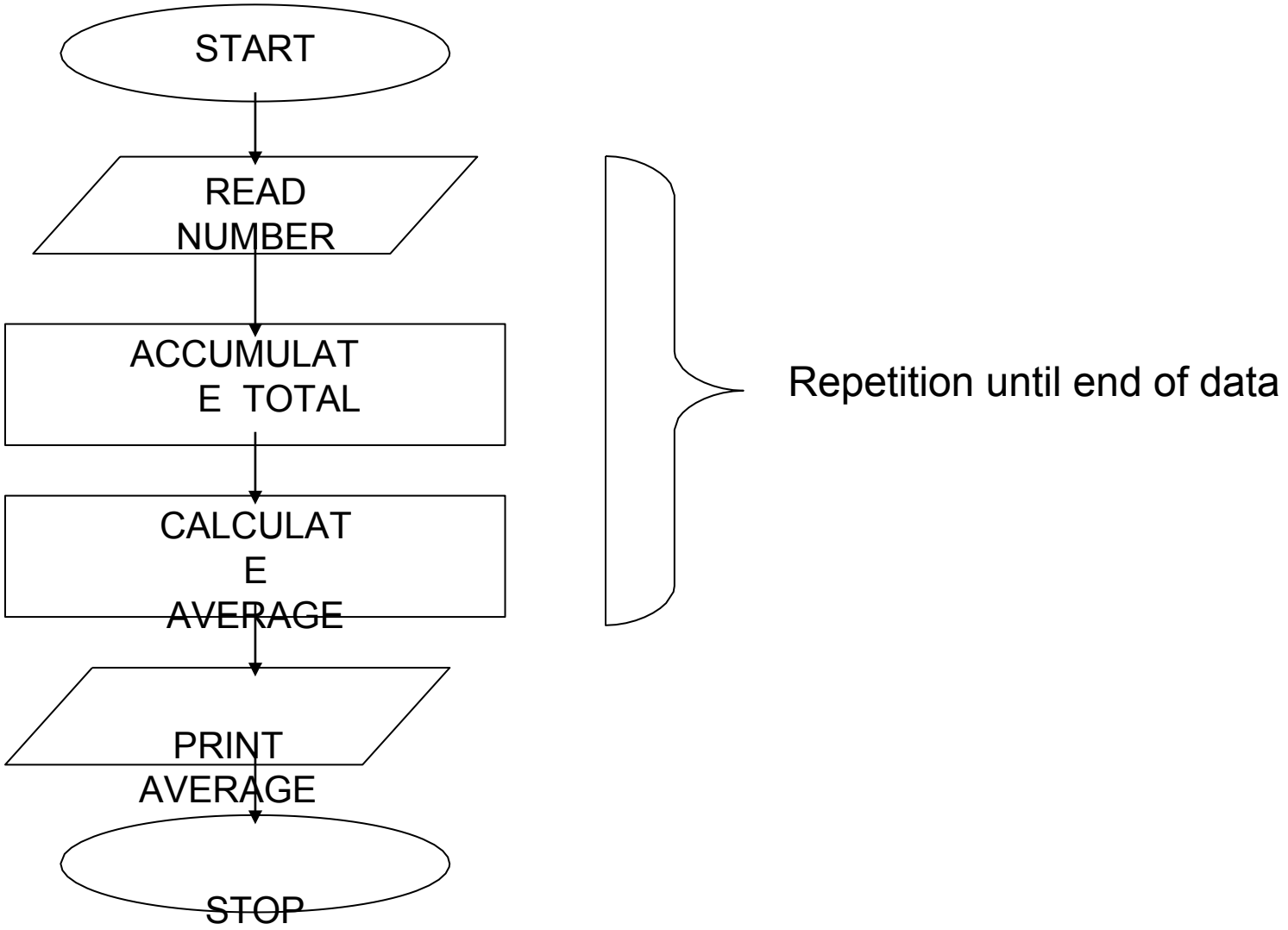


PRE-PROGRAMMING PHASE

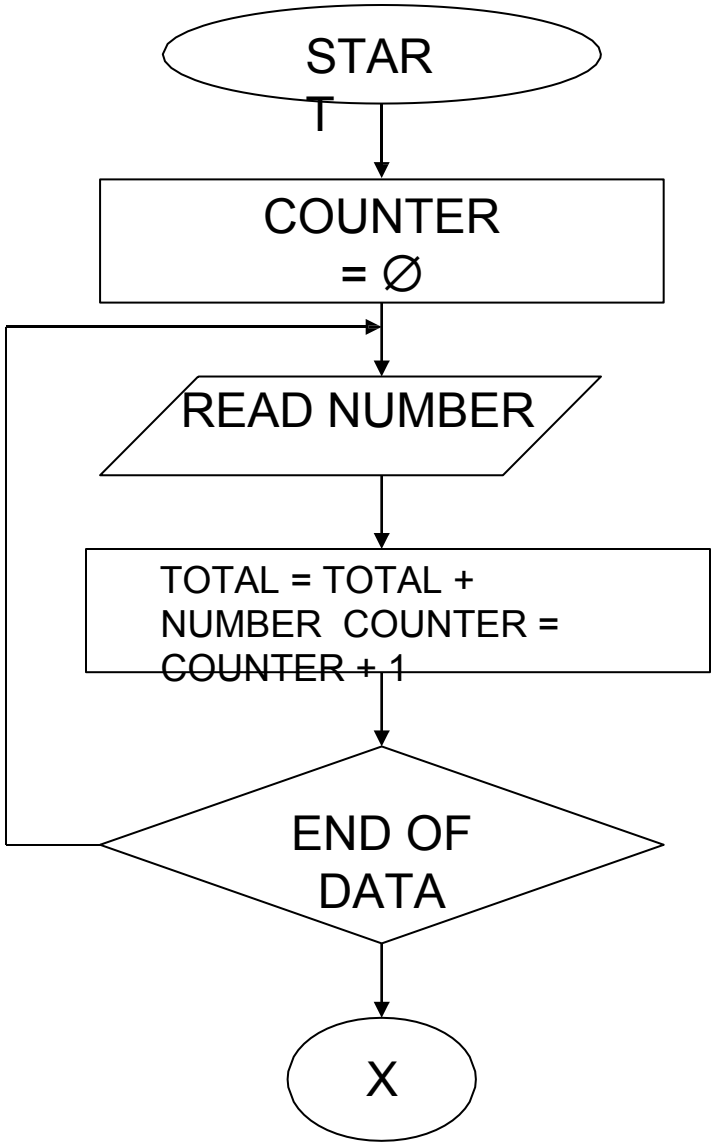
Finding Average Problem

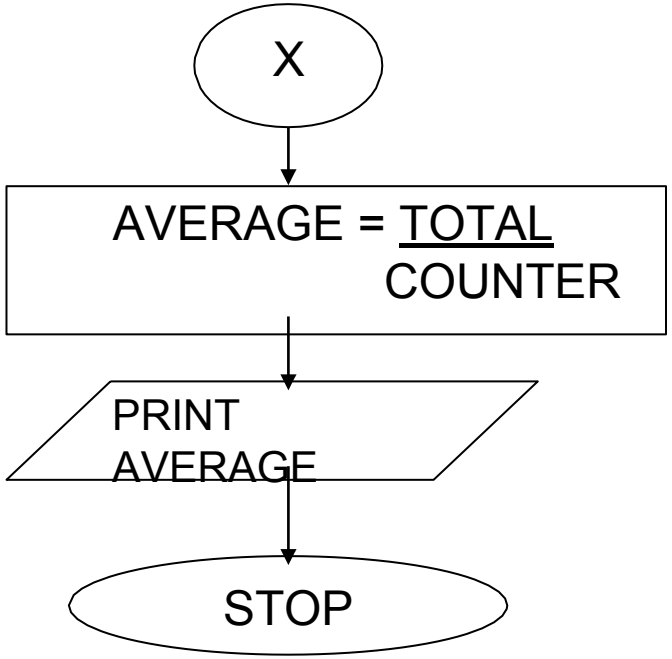
- Read a sequence of number, find the average of the number and print the average.
- Solution: Stepwise Analysis of Average Problem
    - Start the processing
    - Read a number
    - Add the number
    - Repeat reading until last data
    - Calculate the average
    - Print the average
    - Stop the processing

PRE-PROGRAMMING PHASE



PRE-PROGRAMMING PHASE





PRE-PROGRAMMING PHASE

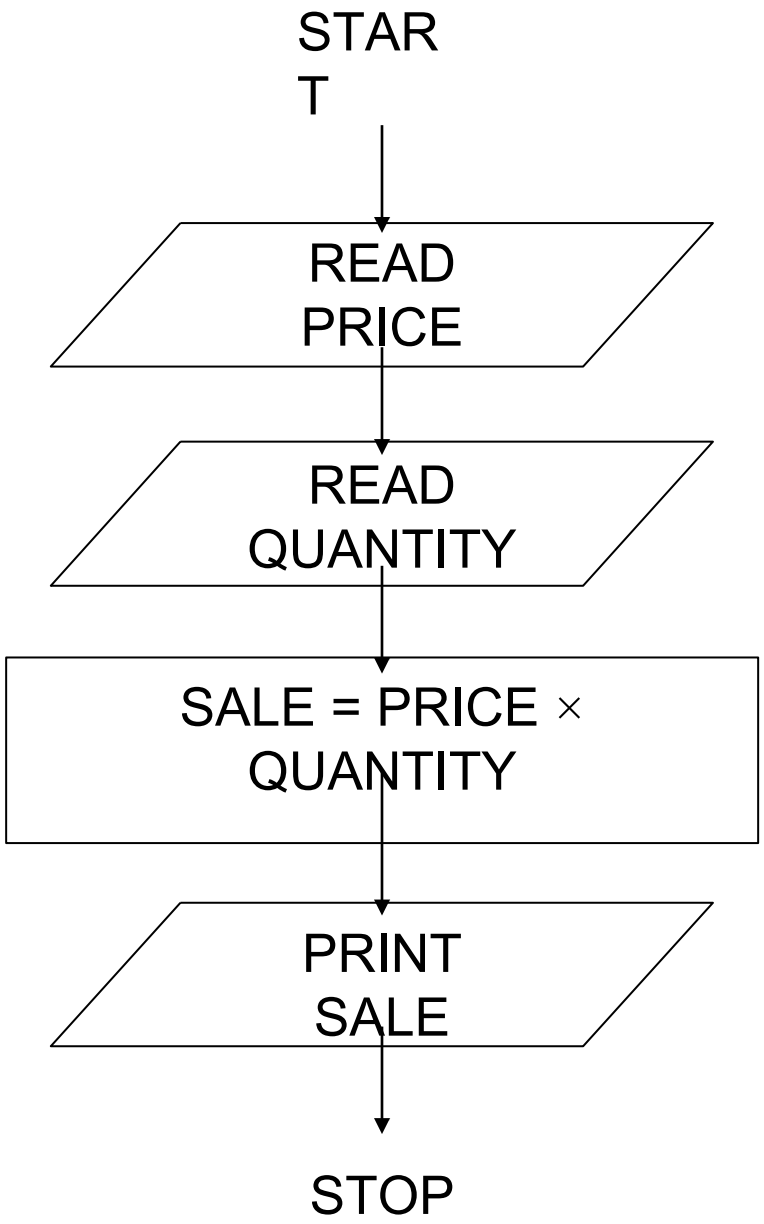
TASK:

Draw a flow chart to print the sum of all even numbers between 1 to n.

PRE-PROGRAMMING PHASE

- **Writing the Algorithm (Pseudocode)**

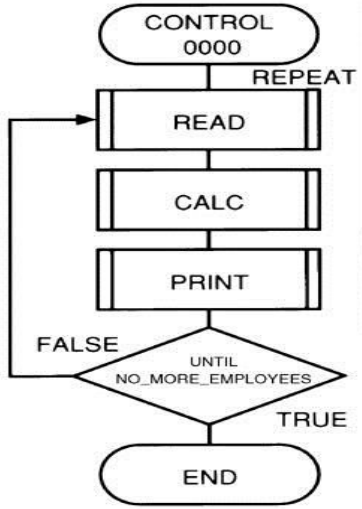
- Pseudocode means an imitation computer code.
- It is used in place of symbols or a flowchart to describe the logic of a program. Thus, it is a set of instructions (descriptive form) to describe the logic of a program.
- Pseudocode is close to the actual programming language.
- Using the Pseudocode, the programmer can start to write the actual code.

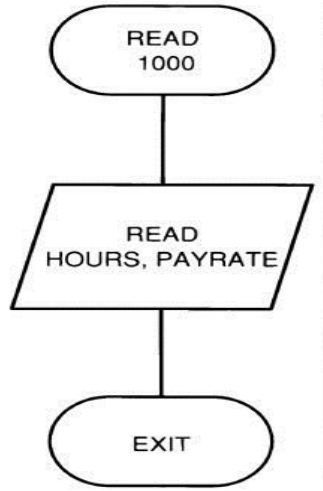


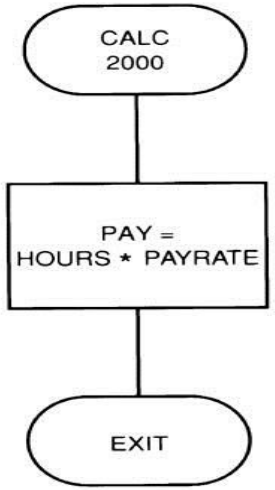
Algorithm:

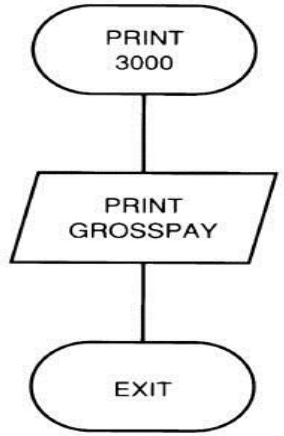
Start
Read price, quantity
Sale = price x quantity
Print Sale
End

Payroll System

Algorithm	Flowchart
Control Module 0000 1. REPEAT PROCESS READ PROCESS CALC PROCESS PRINT UNTIL NO_MORE_ EMPLOYEES 2. END	

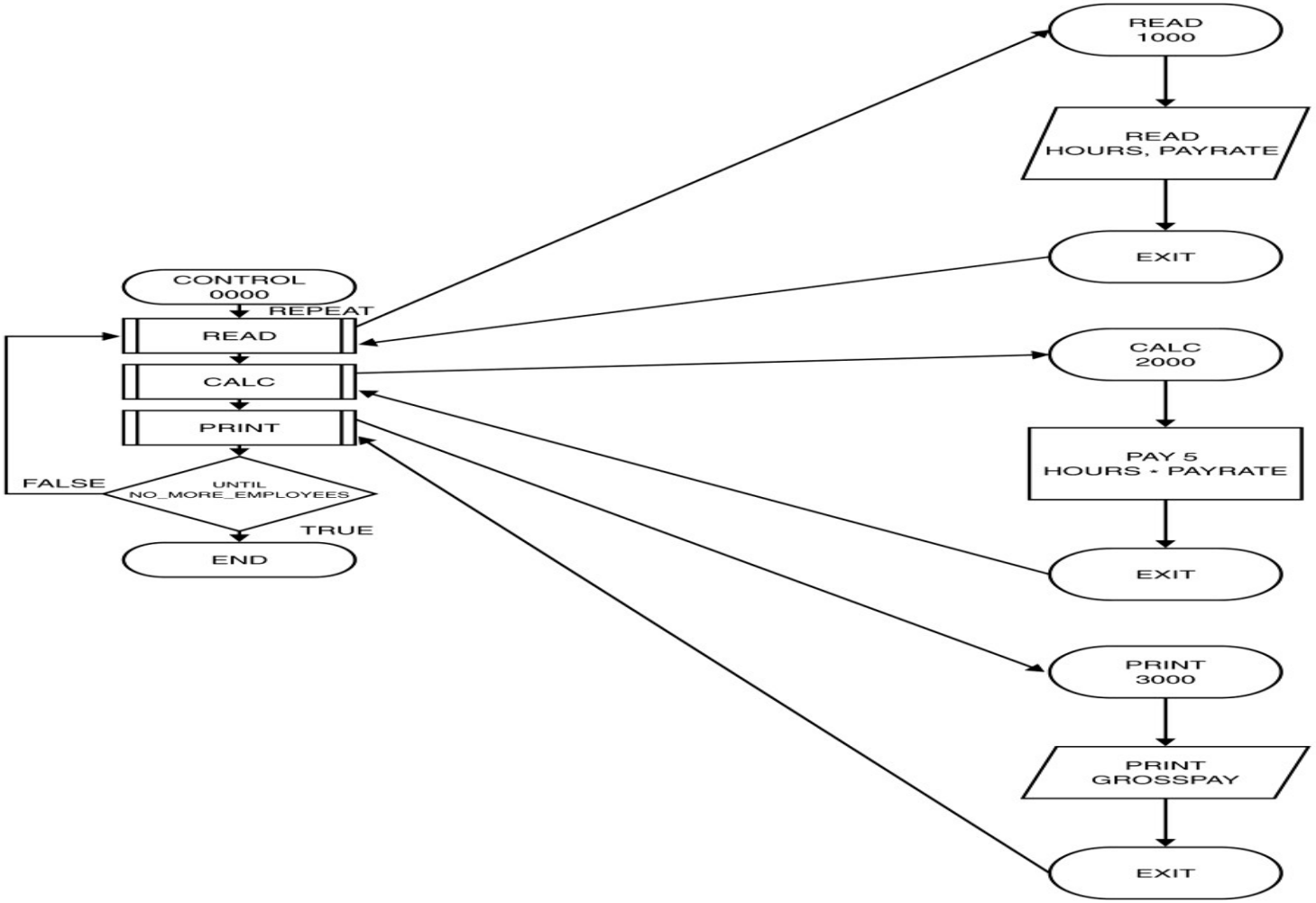
Algorithm	Flowchart
Read Module 1000 1. READ HOURS, PAYRATE 2. EXIT	

Algorithm	Flowchart
CALC Module 2000 1. GROSS PAY = HOURS WORKED * PAYRATE 2. EXIT	

Algorithm	Flowchart
Print Module 3000 1. PRINT PAY 2. EXIT	

Example: Flowchart & Algorithm

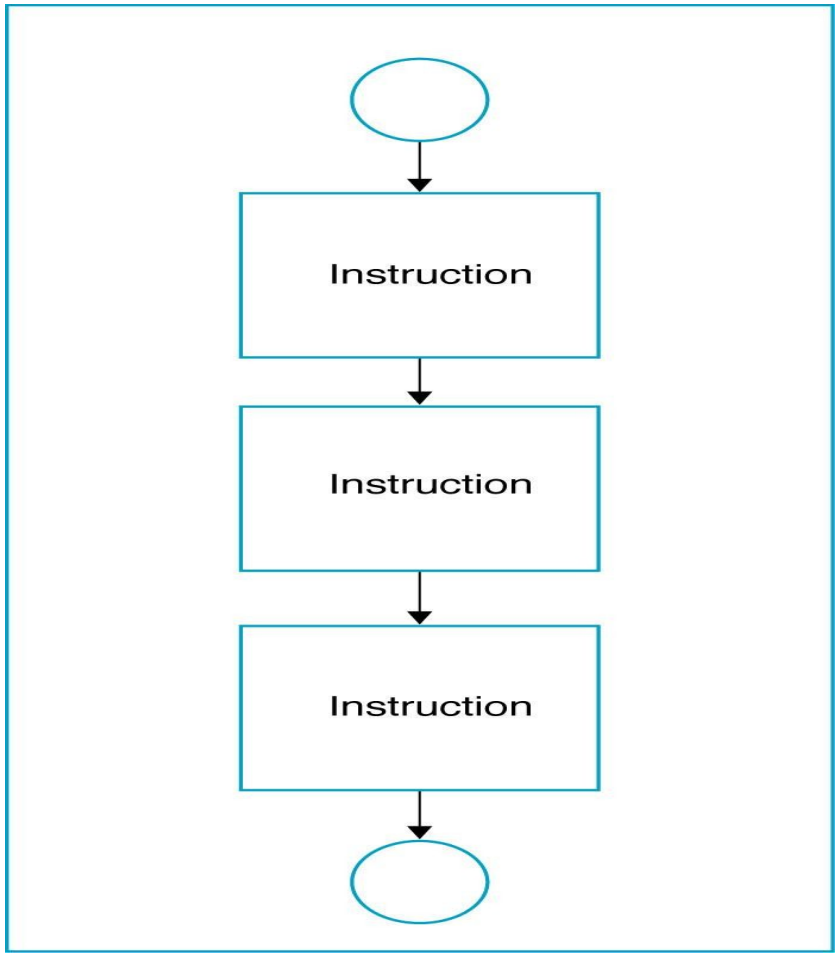
Order of Execution of Instructions : Payroll System



Structuring a Program

- Develop efficient computer solution to problems:
  1. Use Modules
  2. Use four logic structures
     a. Sequential structure
        - Executes instructions one after another in a sequence.
     b. Decision structure
        - Branches to execute one of two possible sets of instructions.
     c. Loop structure
        - Executes set of instruction many times.
     d. Case structure
        - Executes one set of instructions out of several sets.
  3. Eliminate rewriting of identical process by using modules.
  4. Use techniques to improve readability including four logic structure, proper naming of variables, internal documentation and proper indentation.

a. Sequential Logic Structure



b. Decision Logic Structure

- Implements using the IF/THEN/ELSE instruction.
- Tells the computer that IF a condition is true, THEN execute a set of instructions, or ELSE execute another set of instructions
- ELSE part is optional, as there is not always a set of instructions if the conditions are false.

- Algorithm:

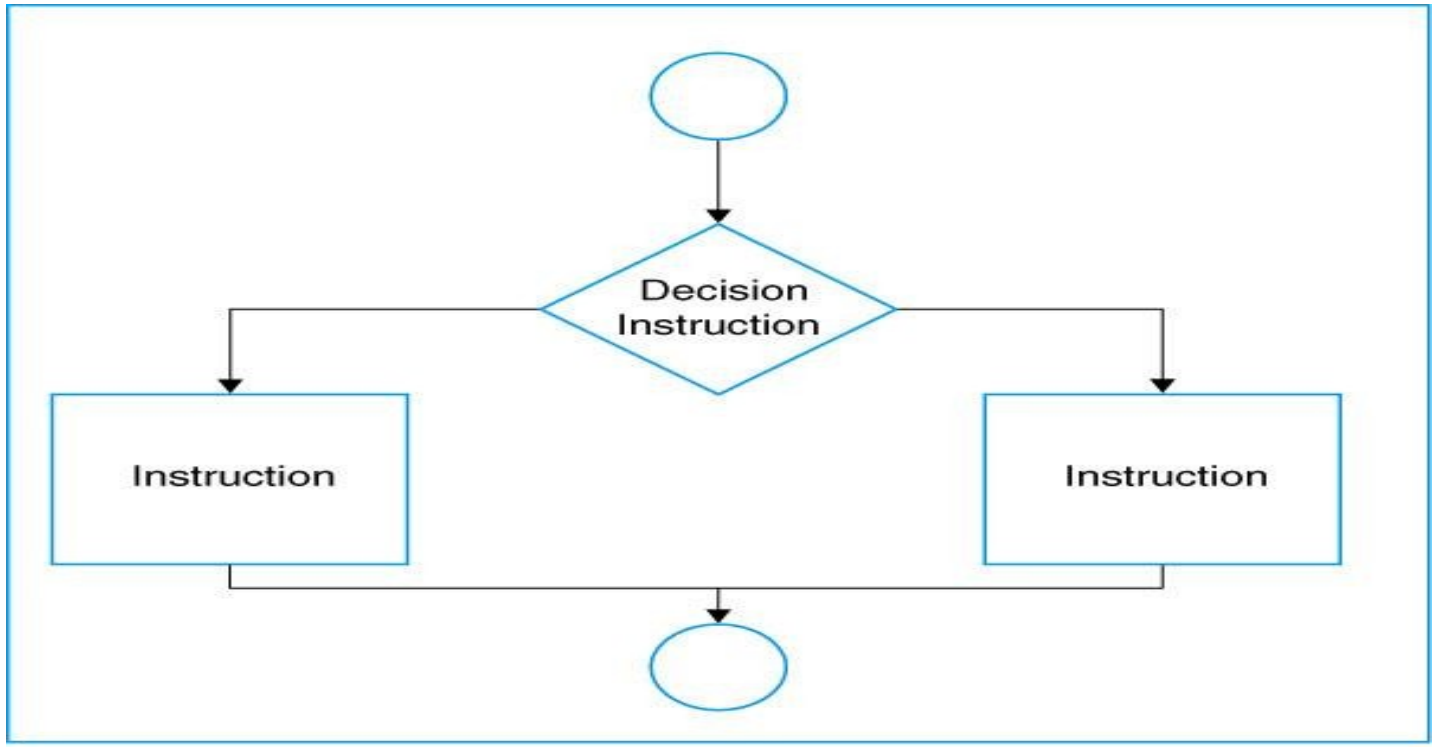
IF <condition(s)> THEN

<TRUE instruction(s)>

ELSE

<FALSE instruction(s)

Decision Logic Structure



Examples of conditional expressions

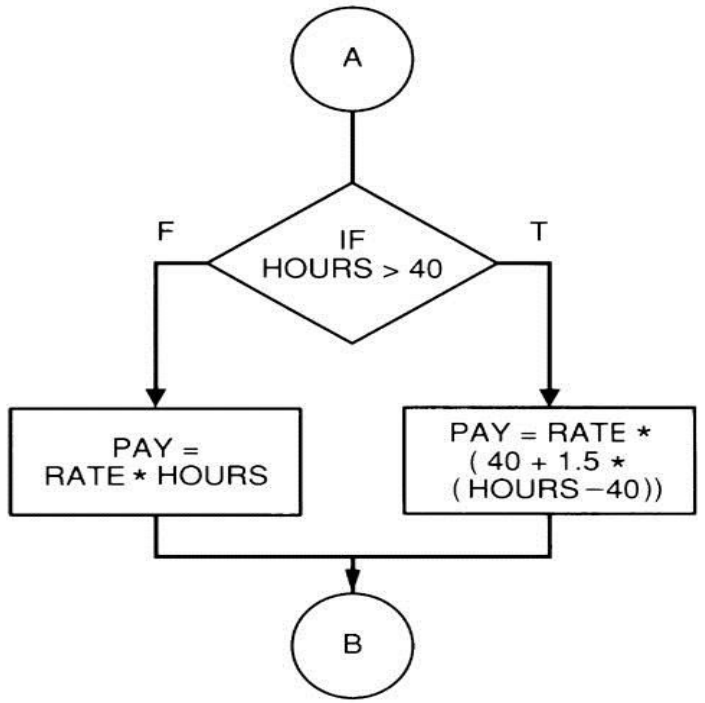
- A < B (A and B are the same data type – either numeric, character, or string)
- X + 5 >= Z (X and Z are numeric data)
- E < 5 or F > 10 (E and F are numeric data)

Example

Assume your are calculating pay at an hourly rate, and overtime pay(over 40 hours) at 1.5 times the hourly rate.

- IF the hours are greater than 40, THEN the pay is calculated for overtime, or ELSE the pay is calculated in the usual way.

Example Decision Structure

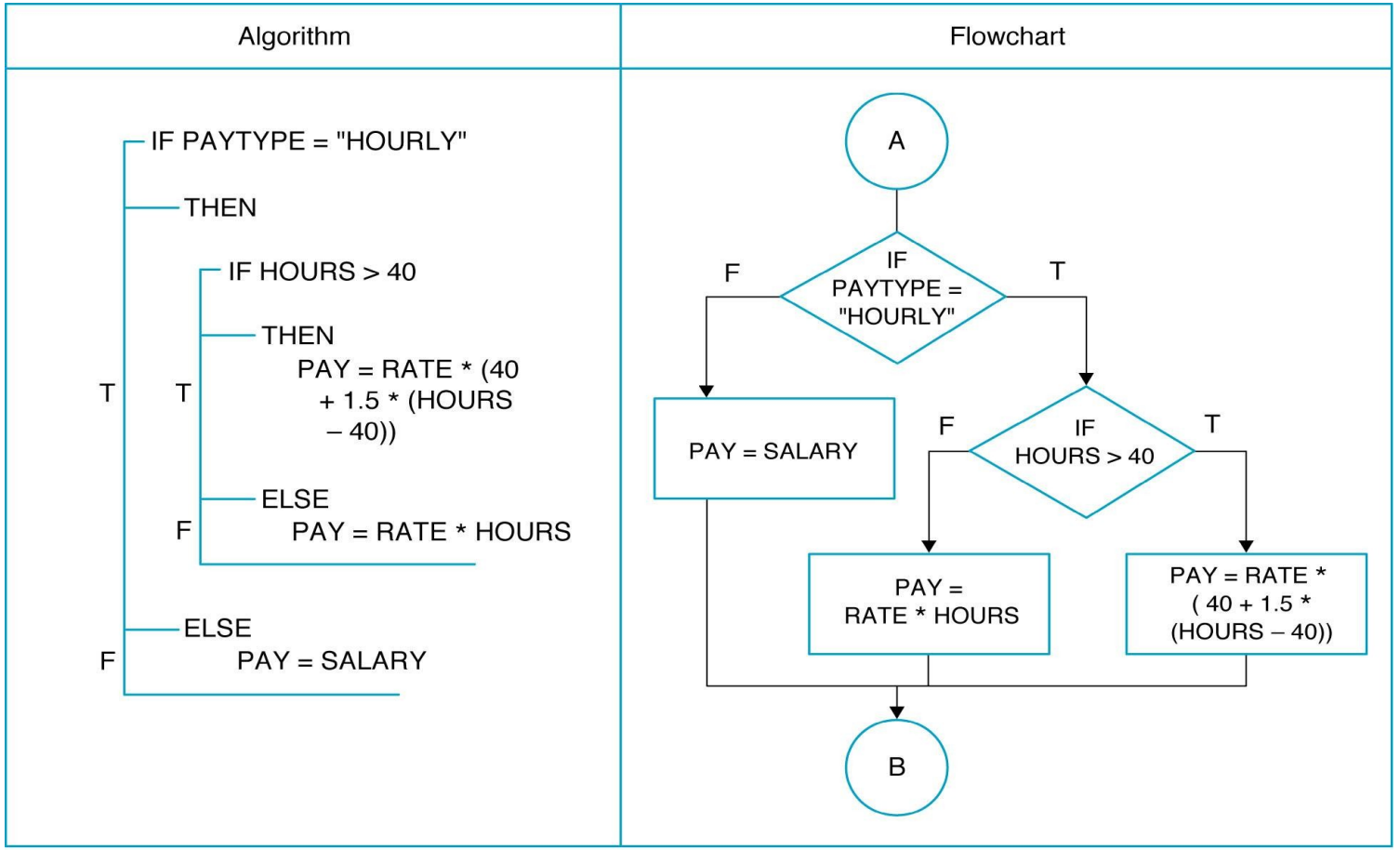
Algorithm	Flowchart
IF HOURS > 40 THEN T PAY = RATE * (40 + 1.5 * (HOURS − 40)) ELSE F PAY = RATE * HOURS	

Note: For all flowcharts with decision blocks, T = TRUE and F = FALSE

NESTED IF/THEN/ELSE INSTRUCTIONS

- Multiple decisions.
- Instructions are sets of instruction in which each level of a decision is embedded in a level before it.

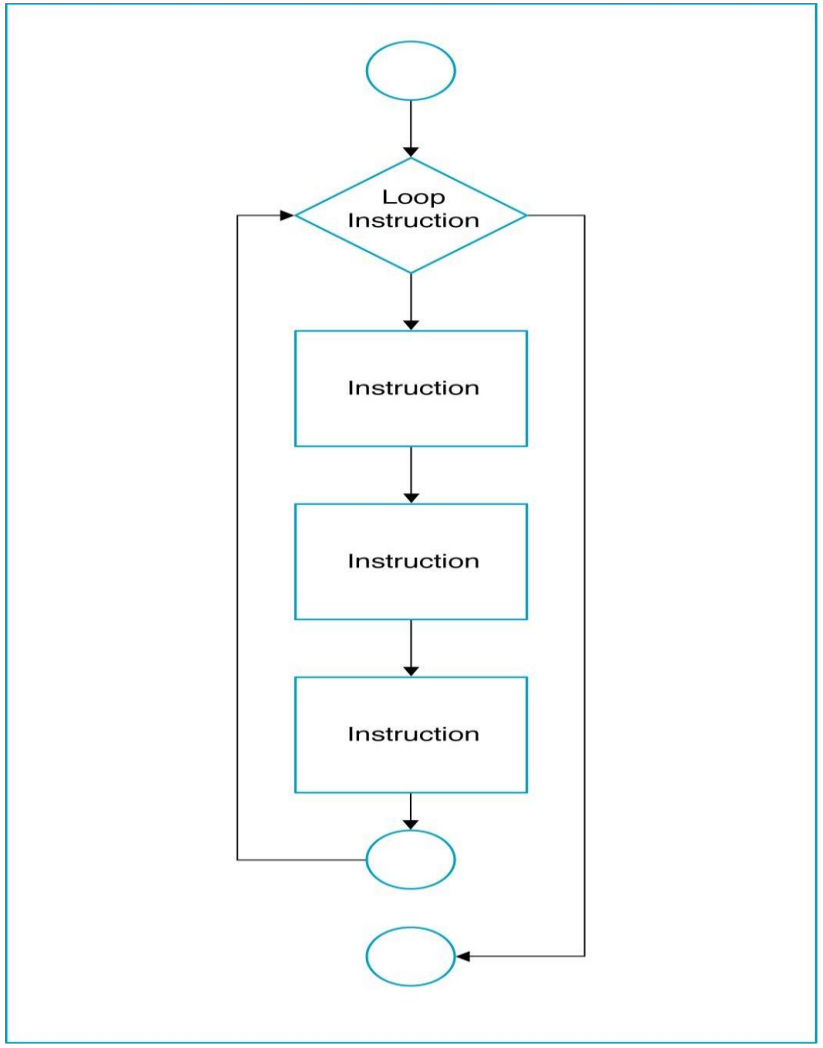
NESTED IF/THEN/ELSE INSTRUCTIONS



c. Loop Logic Structure

- Repeat structure
- To solve the problem that doing the same task over and over for different sets of data
- Types of loop:
  - WHILE loop
  - Do WHILE loop

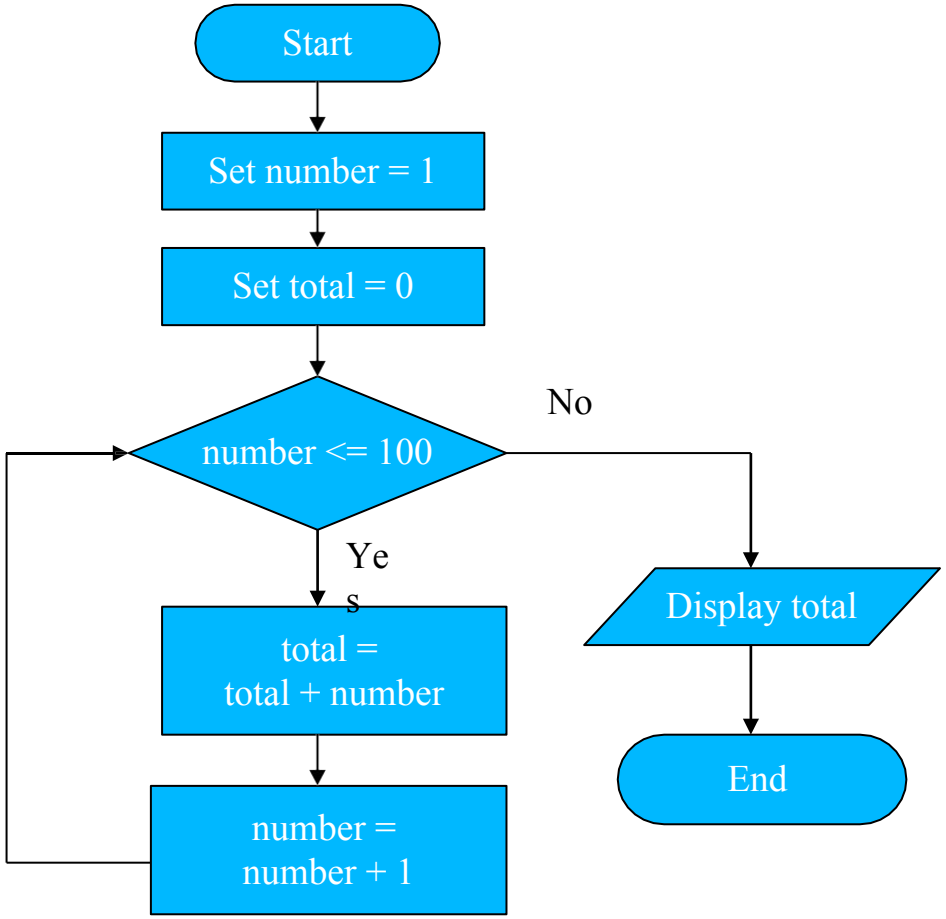
Loop Logic Structure



WHILE loop

- Do the loop body if the condition is true.
- Example: Get the sum of 1, 2, 3, …, 100.
- Algorithm:
    - Set the number = 1
    - Set the total = 0
    - While (number <= 100)
        - total = total + number
        - number = number + 1
    - End While
    - Display total

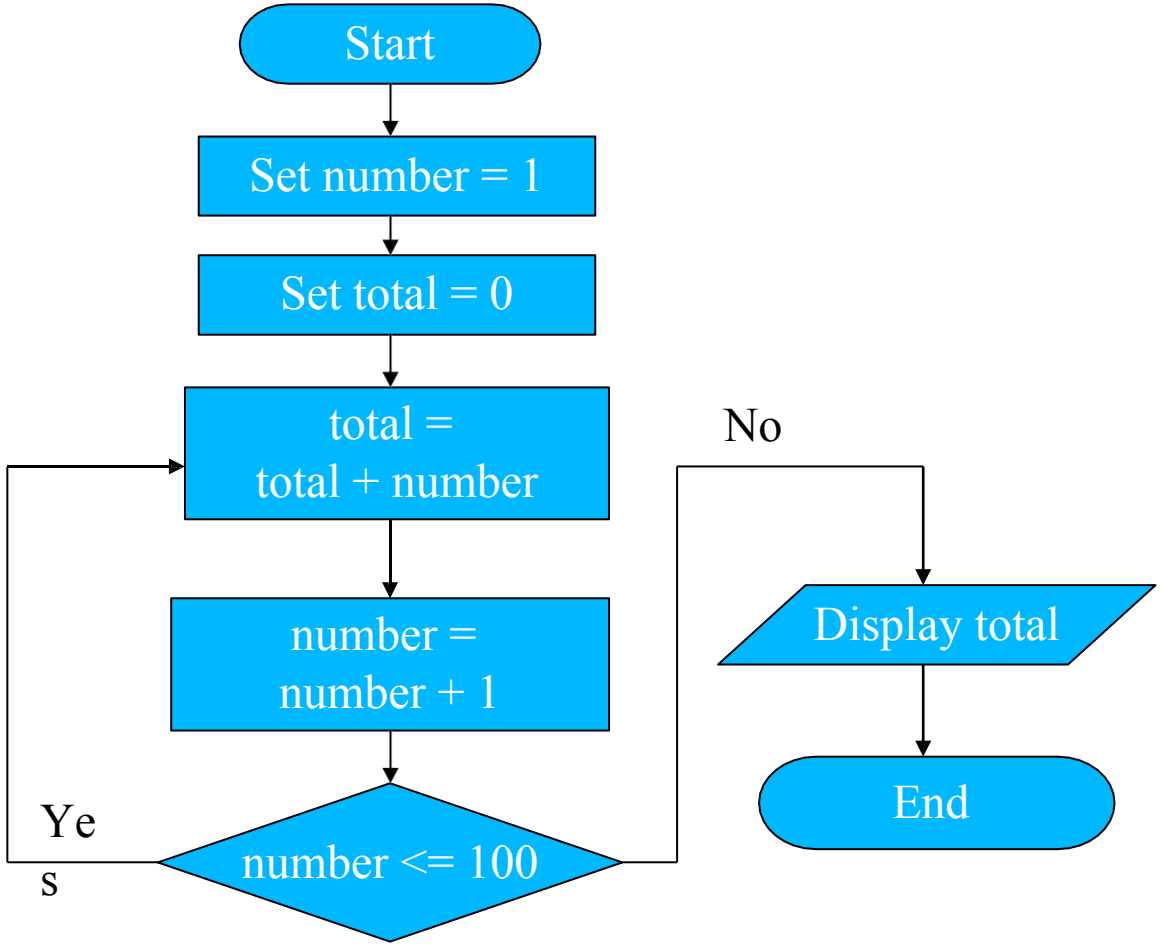
WHILE loop


Start
Set number = 1
Set total = 0
number <= 100
No
Ye
s
total =
total + number
number =
number + 1
Display total
End

DO WHILE Loop

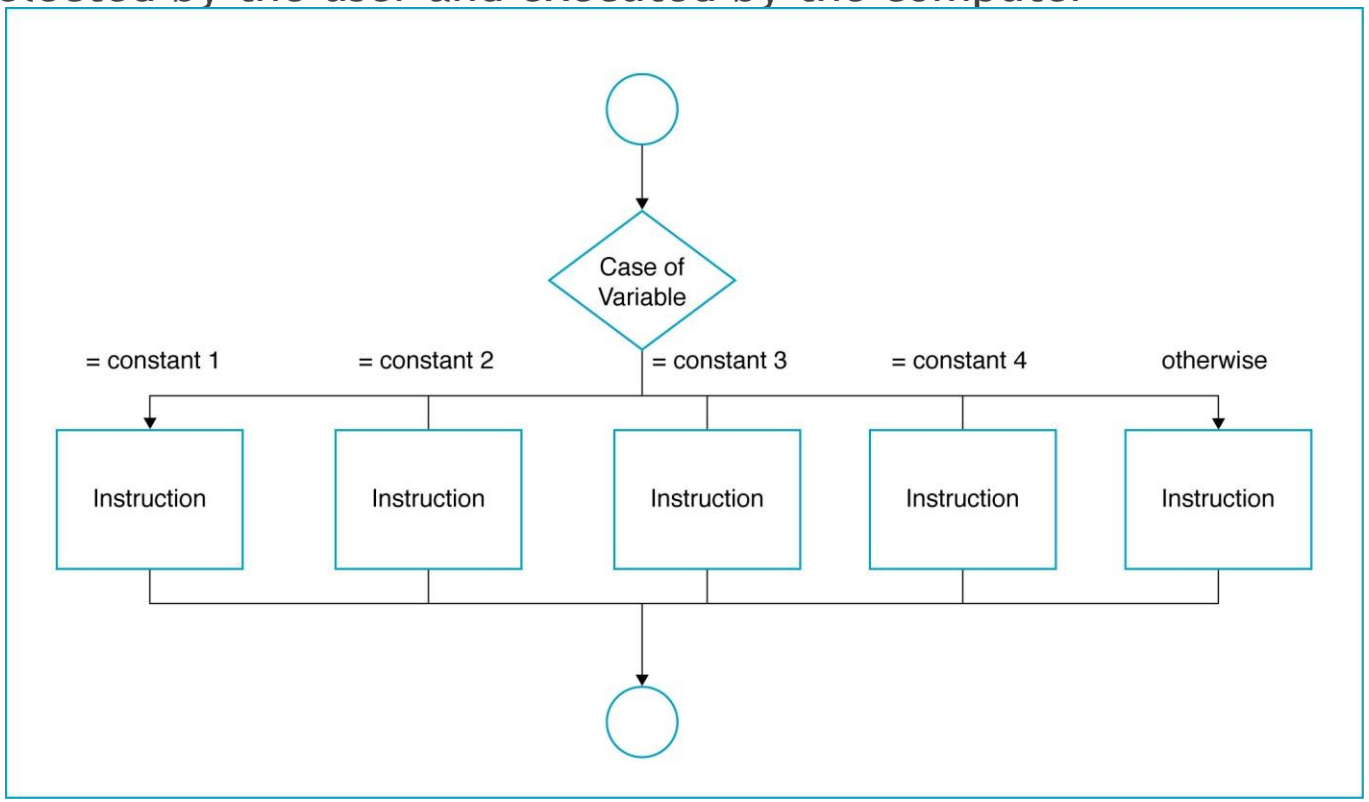
- The body of the loop will process first before check the condition.
- Example: Get the sum of 1, 2, 3, …100.

DO WHILE Loop



The Case Logic Structure

- Made up of several or many sets of instructions, only one of which will be selected by the user and executed by the computer



Case Logic Structure

- Example: A company has four different medical plans. The programmer has given each plan a code corresponding to the beginning initial of the company: Plan 1 = F, Plan 2 = B, Plan 3 = K, Plan 4 = E.

  The company pays for all of Plan 1. The individual has to pay for part of the others. The payroll deduction for Plan 2 = 4.65, for Plan 3 = 7.85, and for Plan 4 =

  5.50. Any other codes are considered in error.

  Write the algorithm and draw the flowchart for a module to determine the payroll deduction.

Example of Case Logic Structure

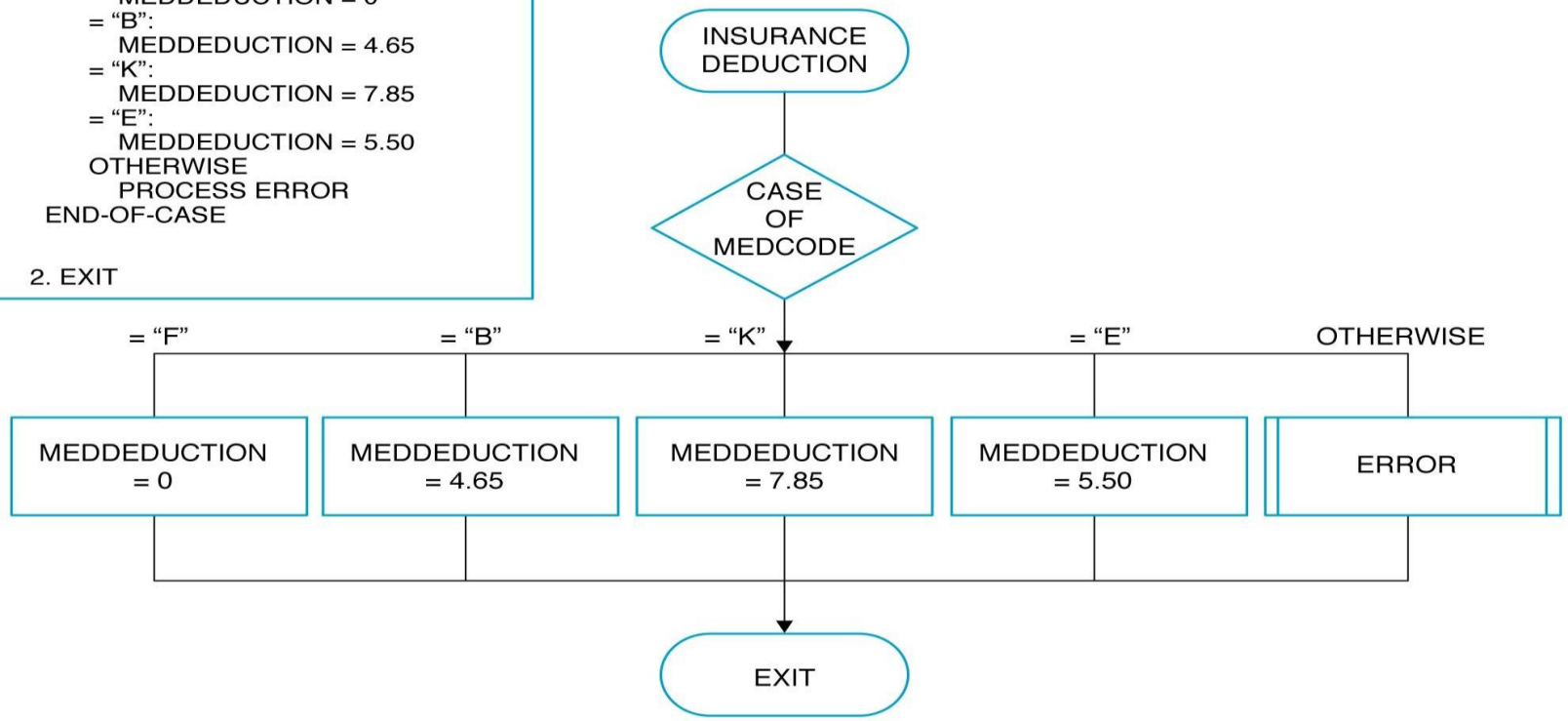
Algorithm

INSURANCE DEDUCTION

1. CASE OF MEDCODE
   = “F”:
      MEDDEDUCTION = 0
   = “B”:
      MEDDEDUCTION = 4.65
   = “K”:
      MEDDEDUCTION = 7.85
   = “E”:
      MEDDEDUCTION = 5.50
   OTHERWISE
      PROCESS ERROR
   END-OF-CASE

2. EXIT

Flowchart



TASK

Write Algorithm & Flowchart to print multiplication Table of a number.