Simple Program Design

"Appendix 1: Flowcharts"

Chapter Outline

Flowcharts

- The Three Basic Control Structures
  - 1 Sequence
  - 2 Selection
  - 3 Repetition
- Simple Algorithms That Use The Sequence Control Structure
  - Example 3.1 Add Three numbers
  - A defining diagram
  - B Solution algorithm
  - Example 3.2 Find average temperature
  - A Defining diagram
  - B Solution algorithm
  - Example 3.3 Calculate mowing time
  - A Defining diagram
  - B Solution algorithm
- Flowcharts and the Selection Control Structure
  - Simple IF statement
  - Null ELSE statement
  - Combined IF statement
  - Nested IF statement
- Simple Alogrithms That Use the Selection Control Structure
  - Example 4.1 Read three characters
  - A Defining diagram
  - B Solution algorithm
  - Example 4.2 Process customer record
  - A Defining diagram
  - B Solution algorithm
  - Example 4.3 Calculate employee's pay

A Defining diagram
B Solution algorithm

The CASE Structure Expressed as a Flowchart

Example 4.4 Process customer record
A Defining diagram
B Solution Algorithm

Flowchart and the Repetition Control Structure
Simple Algorithms That Use the Repetition Control Structure

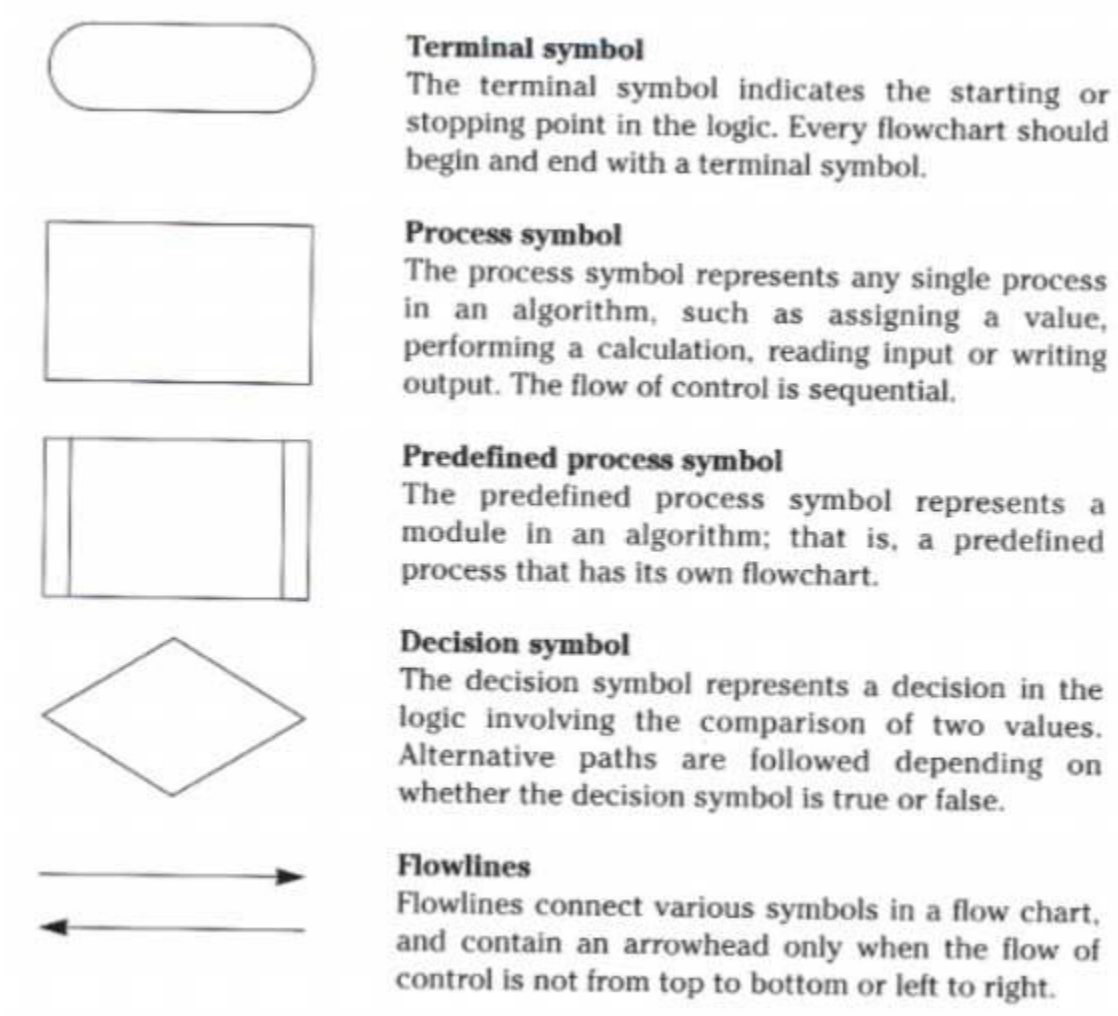
Example 5.1 Fahrenheit-Celsius conversion
A Defining diagram
B Solution algorithm
Example 5.2 Print examination scores
A Defining diagram
B Solution alogorithm
Example 5.3 Process student enrolments
A Defining diagram
B Solution algorithm
Example 5.4 Process inventory items
A Defining diagram
B Solution algorithm

Flowcharts and Modules

Example 8.1 Read three characters
A Defining diagram
B Hierarchy chart
C Solution algorithm using a predefined process symbol

Flowcharts

Flowcharts are a visual outlining tool. They can be used to represent an algorithm. The flowing five symbols are enough to flowchart an algorithm.



Terminal symbol
The terminal symbol indicates the starting or stopping point in the logic. Every flowchart should begin and end with a terminal symbol.

Process symbol
The process symbol represents any single process in an algorithm, such as assigning a value, performing a calculation, reading input or writing output. The flow of control is sequential.

Predefined process symbol
The predefined process symbol represents a module in an algorithm; that is, a predefined process that has its own flowchart.

Decision symbol
The decision symbol represents a decision in the logic involving the comparison of two values. Alternative paths are followed depending on whether the decision symbol is true or false.

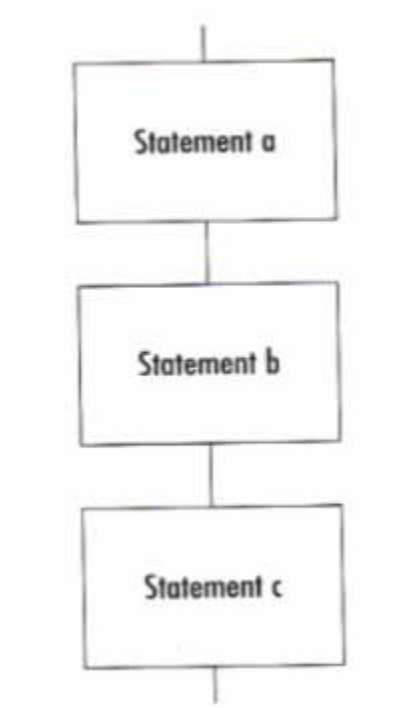
Flowlines
Flowlines connect various symbols in a flow chart, and contain an arrowhead only when the flow of control is not from top to bottom or left to right.

The Three Basic Control Structures

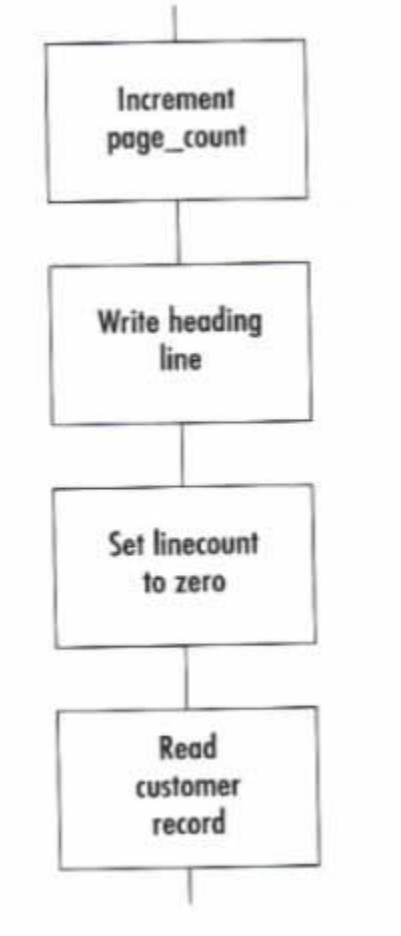
Our programs are made up of the three basic constructures of: sequence, selection, and repetition.

1 Sequence

The sequence control structure is defined as the straight forward execution of one procesing step after another. Here is the general form of a sequence.

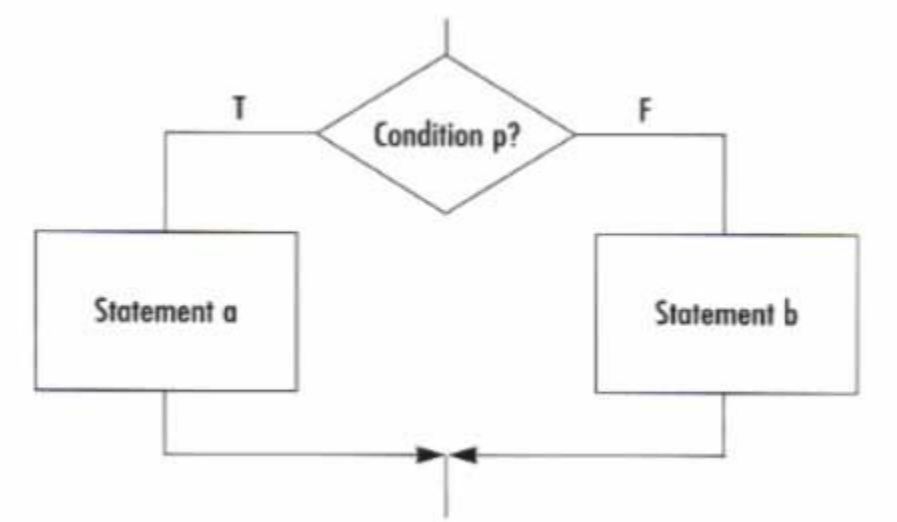


Here is a typical sequence of statements.

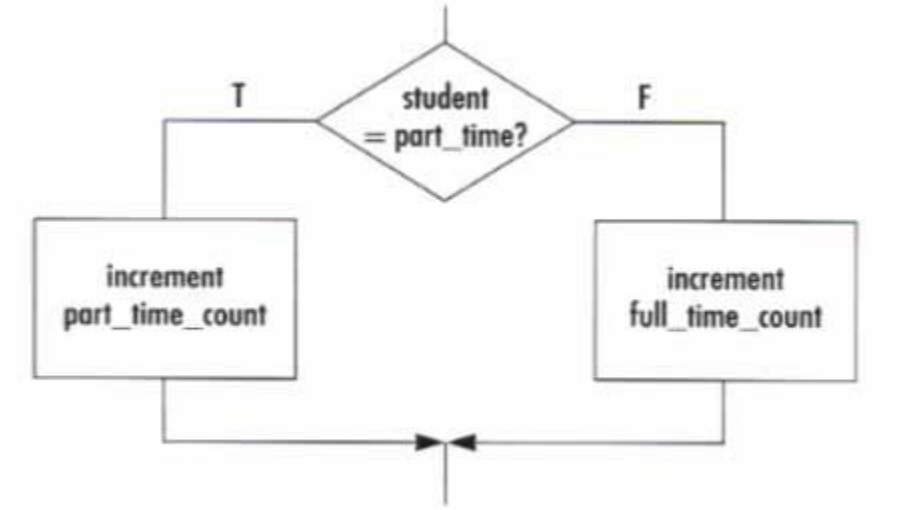


2 Selection

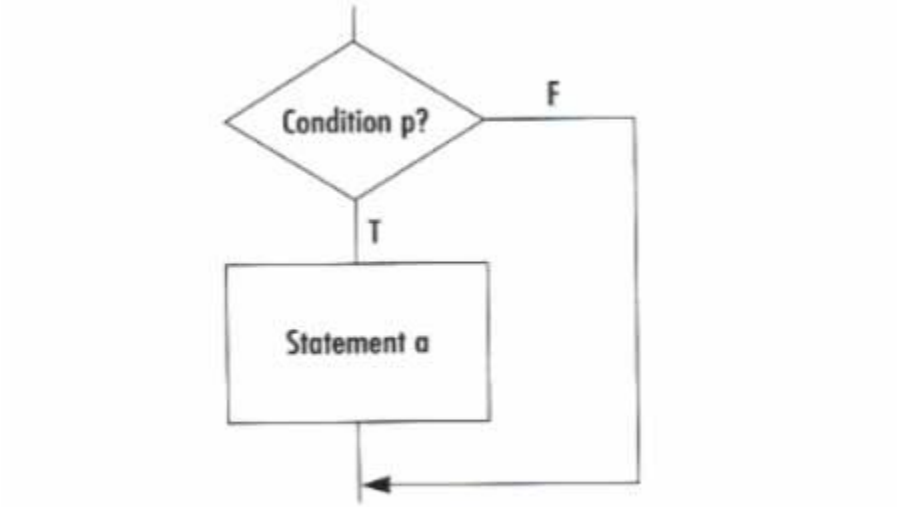
The selection control structure can be defined as the presentation of a condition, and the choice between two actions depending on whether the condition is true or false. Here is the general form of the selection construct.



Here is a typical selection statement.

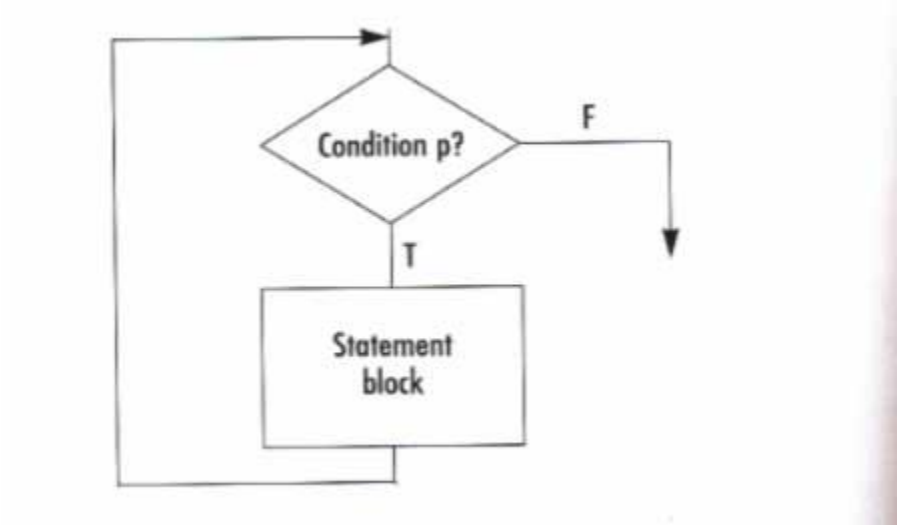


Here is a selection with a empty false side.

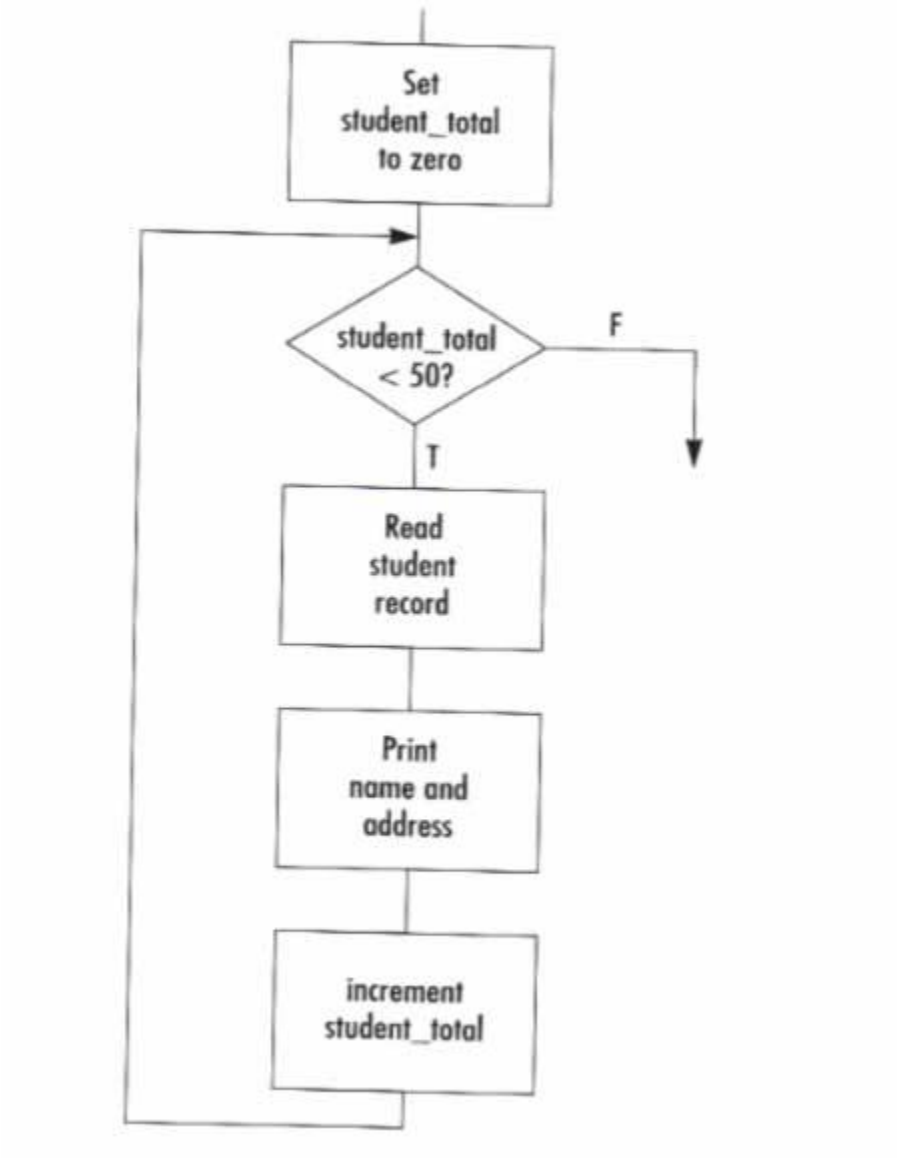


3 Repetition

The repetition control structure can be defined as the presentation of a set of instructions to be performed repeatly, as long as a condition is true. Here is the general form of repetition.



Here is a repetition control structure with typical statements.



Simple Algorithms That Use the Sequence Control Structure

Here are the examples from chapter 3 using flowcharts.

Example 3.1 Add three numbers

A program is required to read three numbers, add them together and print their total.

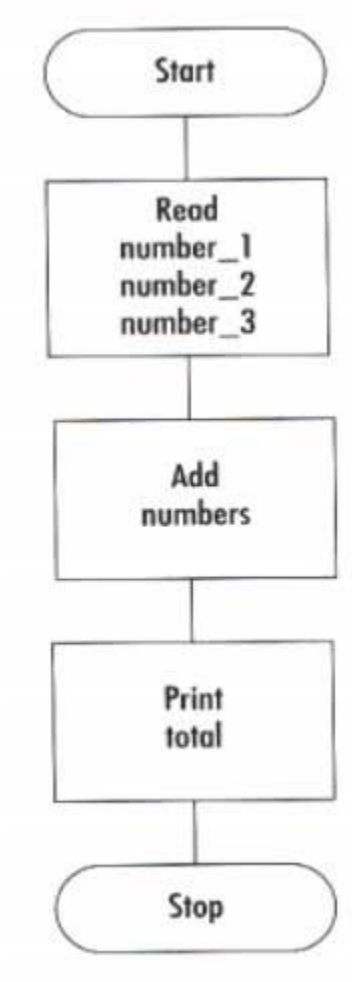
A Defining diagram

Here is the IPO (Input, Process, Output) chart.

Input	Processing	Output
number_1 number_2 number_3	Read three numbers Add numbers together Print total number	total

B Solution algorithm

Here is the algorithm as a flowchart.



Example 3.2 Find average temperature

A program is to get the maximum and minimum temperatures for a day and calculate and display the average temperature.

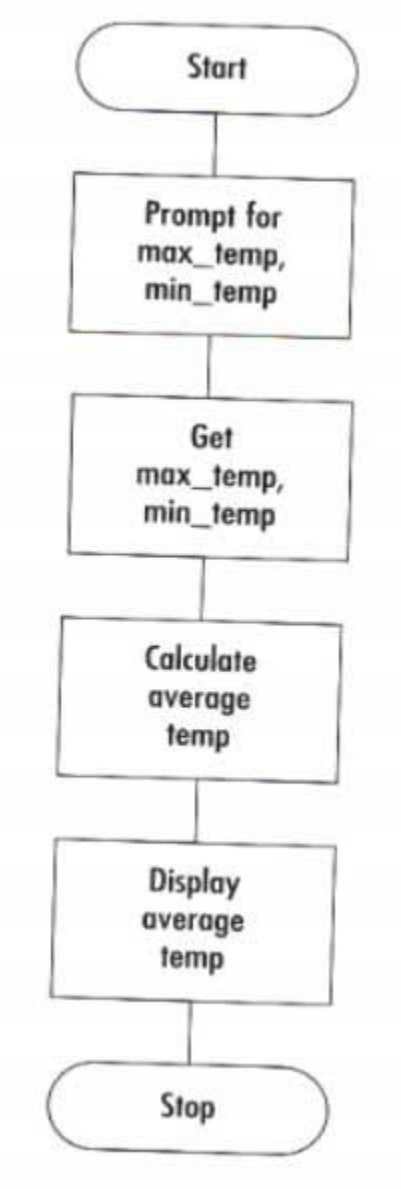
A Defining diagram

Here is the IPO chart.

Input	Processing	Output
max_temp min_temp	Prompt for temperatures Get max, min temperatures Calculate average temperature Display average temperature	avg_temp

B Solution algorithm

Here is the algorithm as a flowchart.



Example 3.3 Calculate mowing time

A program is required to read in the length and width of a lot, and the length and width of the rectangular house that has been built on the lot. Then it computes and displys the mowing time the lot.

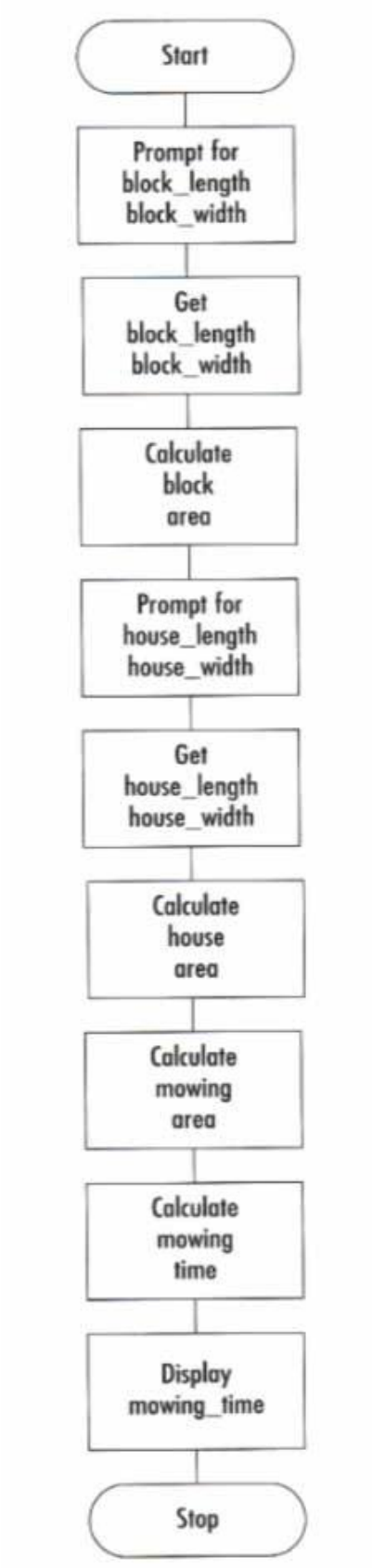
A Defining diagram

Here is the IPO chart.

Input	Processing	Output
block_length block_width house_length house_width	Prompt for block measurements Get block measurements Prompt for house measurements Get house measurements Calculate mowing area Calculate mowing time	mowing_time

B Solution algorithm

Here is the algorithm as a flowchart.

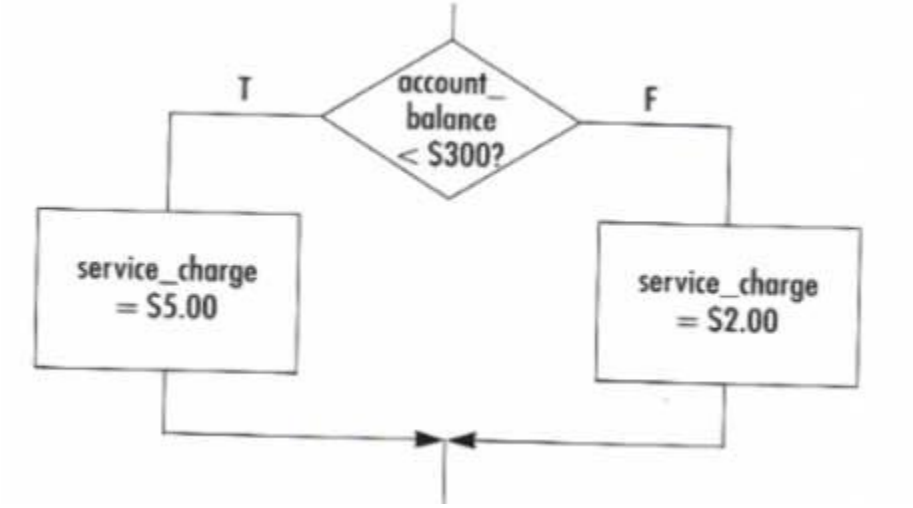

Start
Prompt for block_length block_width
Get block_length block_width
Calculate block area
Prompt for house_length house_width
Get house_length house_width
Calculate house area
Calculate mowing area
Calculate mowing time
Display mowing_time
Stop

Flowcharts and the Selection Control Structure

Here are the examples from chapter 4 using flowcharts.

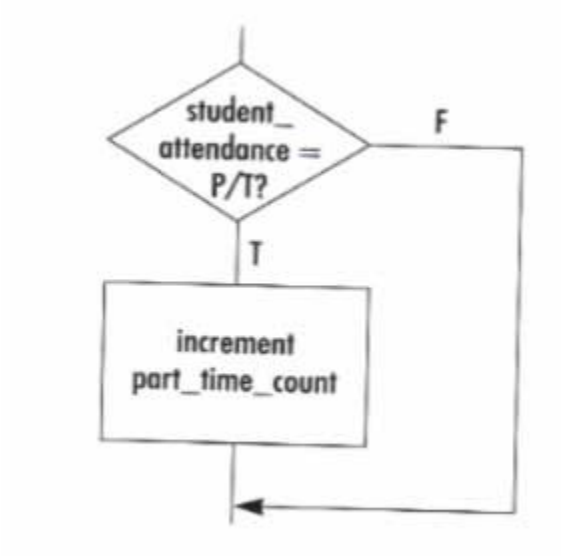
Simple IF statement

Simple selection occurs when a choice is made between two alternative paths.



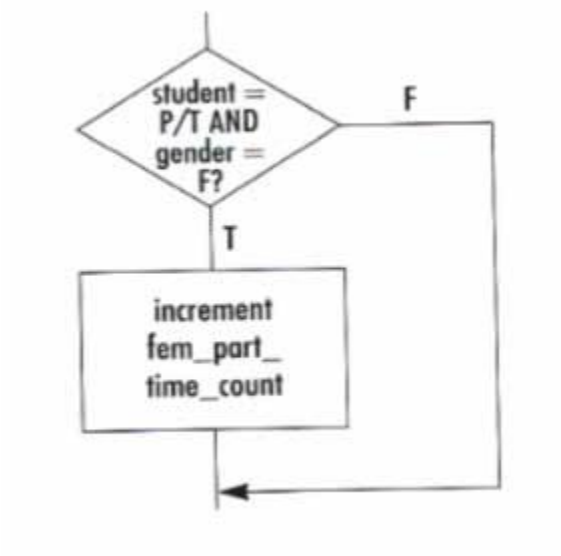
Null ELSE statement

The null ELSE structure is a variation of the simple IF structure with the false side being empty.



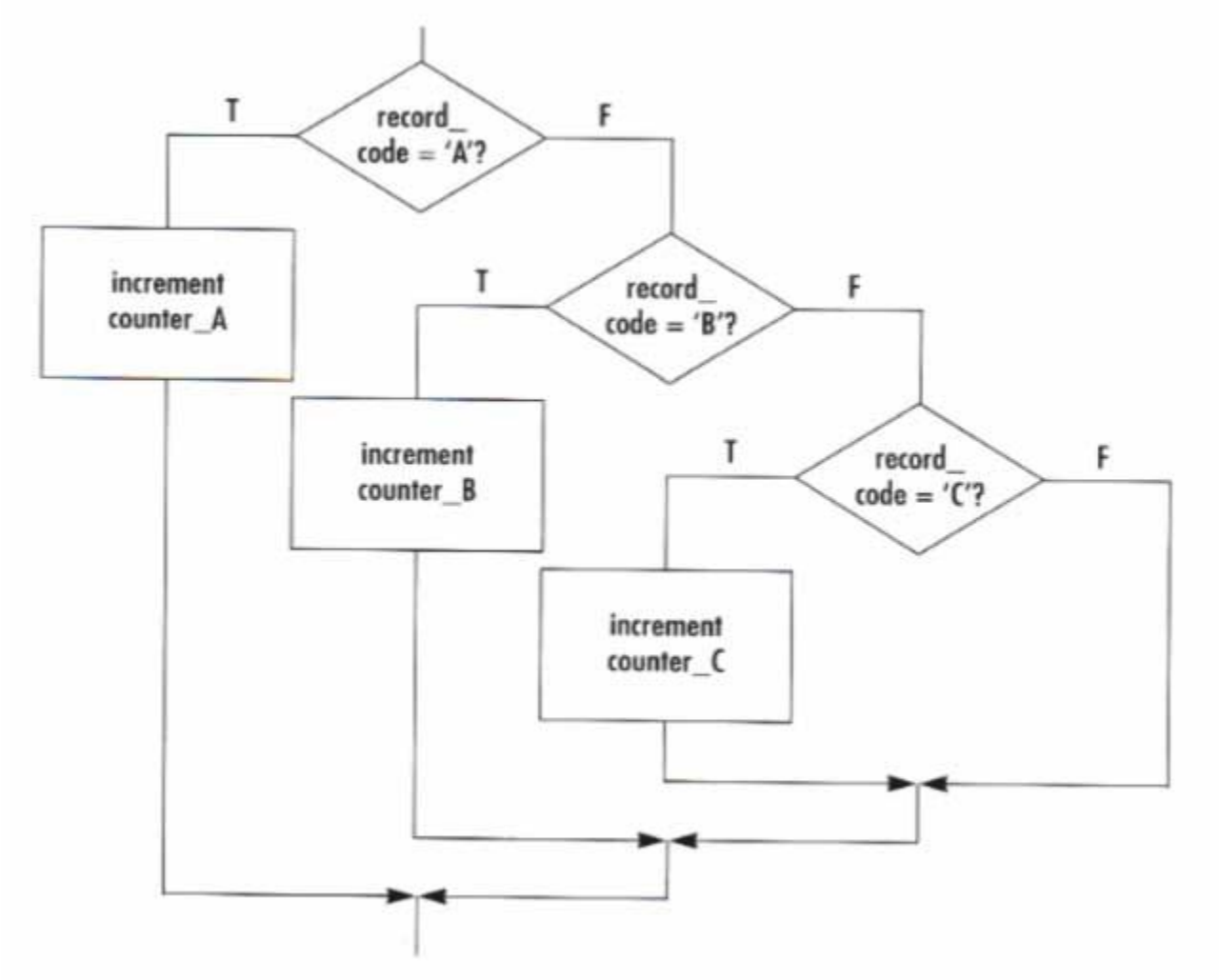
Combined IF statement

A combined IF statement is one that contains multiple conditions in the decision symbol.



Nested IF statement

A nested IF statement has another IF in either the true or false branch.



Simple Algorithms That Use the Selection Control Structure

Finally here are the examples from chapter 4 rewritten as flowcharts.

Example 4.1 Read three characters

An algorithm prompts the operator for three characters, gets the characters, sorts them and ouputs them in ascending order.

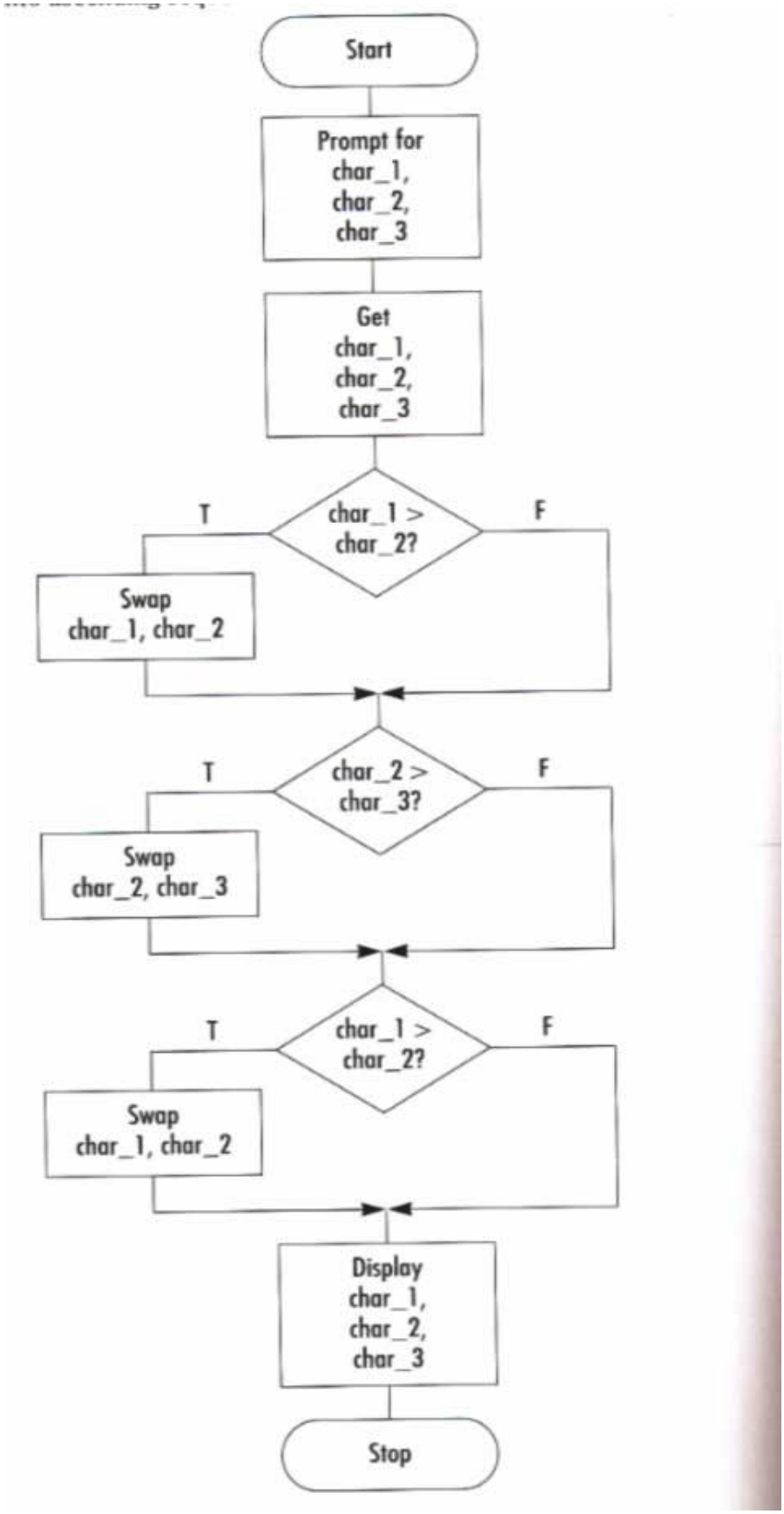
A Defining diagram

Here is the IPO chart.

Input	Processing	Output
char_1 char_2 char_3	Prompt for characters Accept three characters Sort three characters Output three characters	char_1 char_2 char_3

B Solution algorithm

Here is the algorithm as a flowchart.


Start
Prompt for char_1, char_2, char_3
Get char_1, char_2, char_3
char_1 > char_2?
T
F
Swap char_1, char_2
char_2 > char_3?
T
F
Swap char_2, char_3
char_1 > char_2?
T
F
Swap char_1, char_2
Display char_1, char_2, char_3
Stop

Example 4.2 Process customer record

A program reads a customer's name, purchase amount, and a tax code. The tax code is validated, the sales tax is computed along with the toal. very thing is printed out.

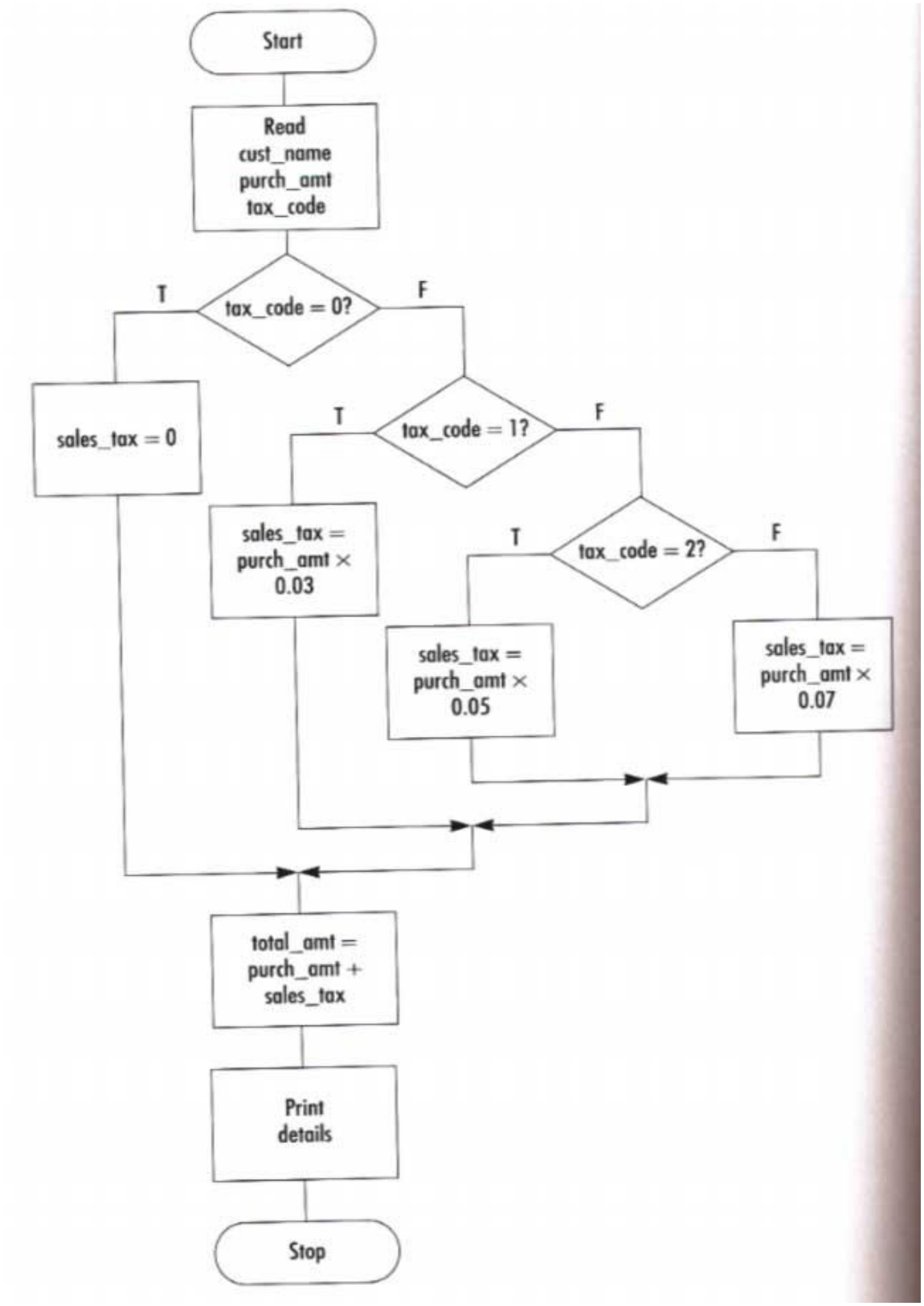
A Defining diagram

Here is the IPO chart.

Input	Processing	Output
cust_name purch_amt tax_code	Read customer details Compute sales tax Compute total amount Print customer details	cust_name purch_amt sales_tax total_amt

B Solution algorithm

Here is the algorithm as a flowchart.



Example 4.3 Calculate employee's pay

A program is to read payroll records, compute an employee's pay, and print out everything.

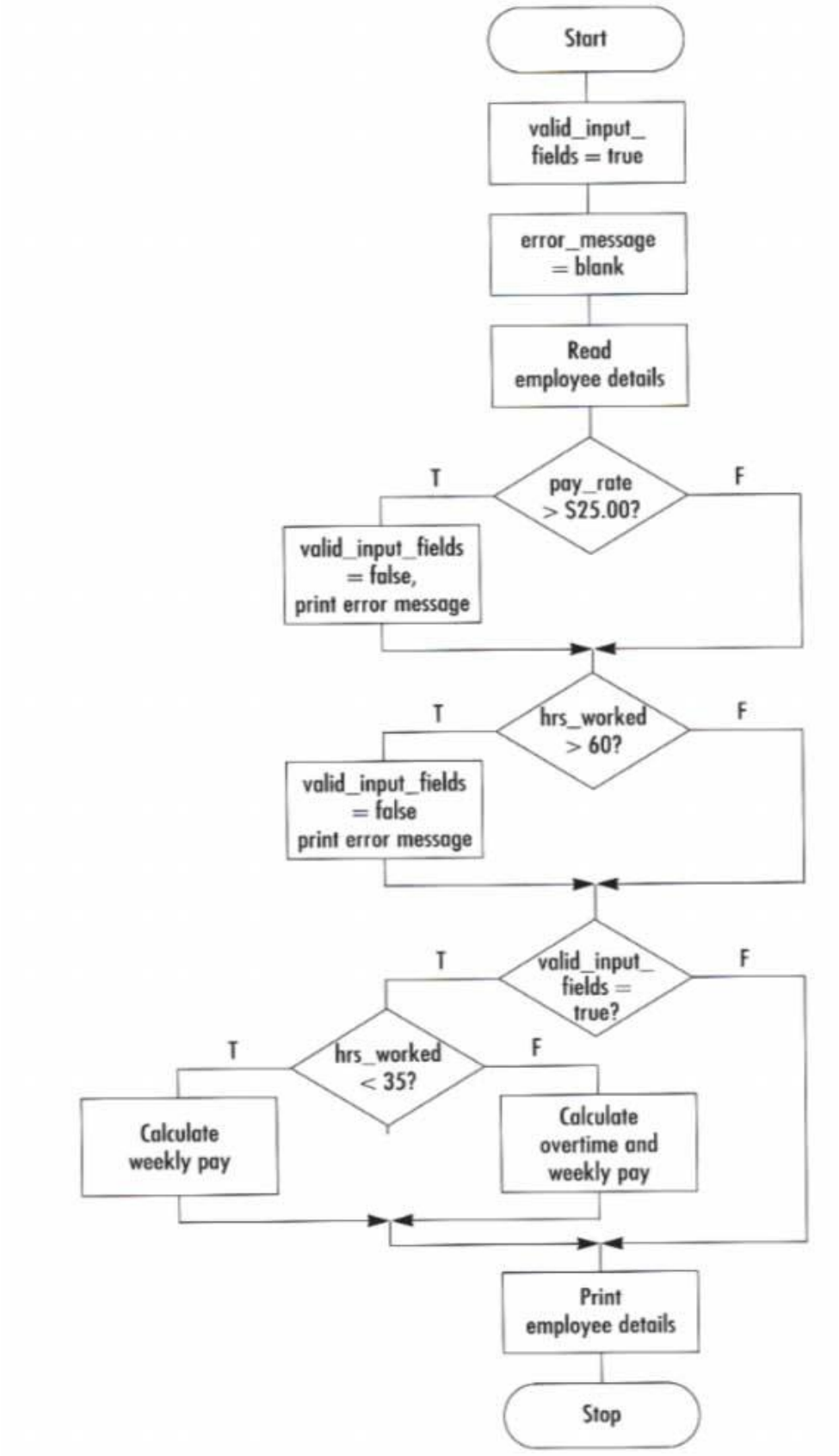
A Defining diagram

Here is the IPO chart for the program.

Input	Processing	Output
emp_no pay_rate hrs_worked	Read employee details Validate input fields Calculate employee pay Print employee details	emp_no pay_rate hrs_worked emp_weekly_pay error_message

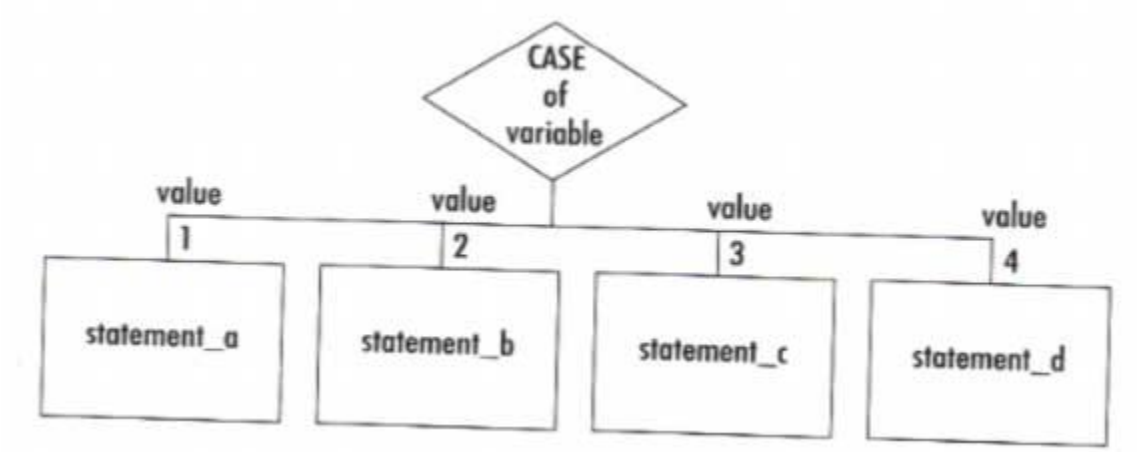
B Solution algorithm

Here is the solution algorithm as a flowchart.


Start
valid_input_ fields = true
error_message = blank
Read employee details
pay_rate > $25.00?
T
F
valid_input_fields = false, print error message
hrs_worked > 60?
T
F
valid_input_fields = false print error message
valid_input_ fields = true?
T
F
hrs_worked < 35?
T
F
Calculate weekly pay
Calculate overtime and weekly pay
Print employee details
Stop

The CASE Structure Expressed as a Flowchart

The CASE statement is great for a test that has many acceptable answers. Here is one way to represent a CASE statement in a flowchart.



Example 4.4 Process customer record

A program is to read customer records, validate the tax code, compute the sales tax and total and print out everything but the tax code.

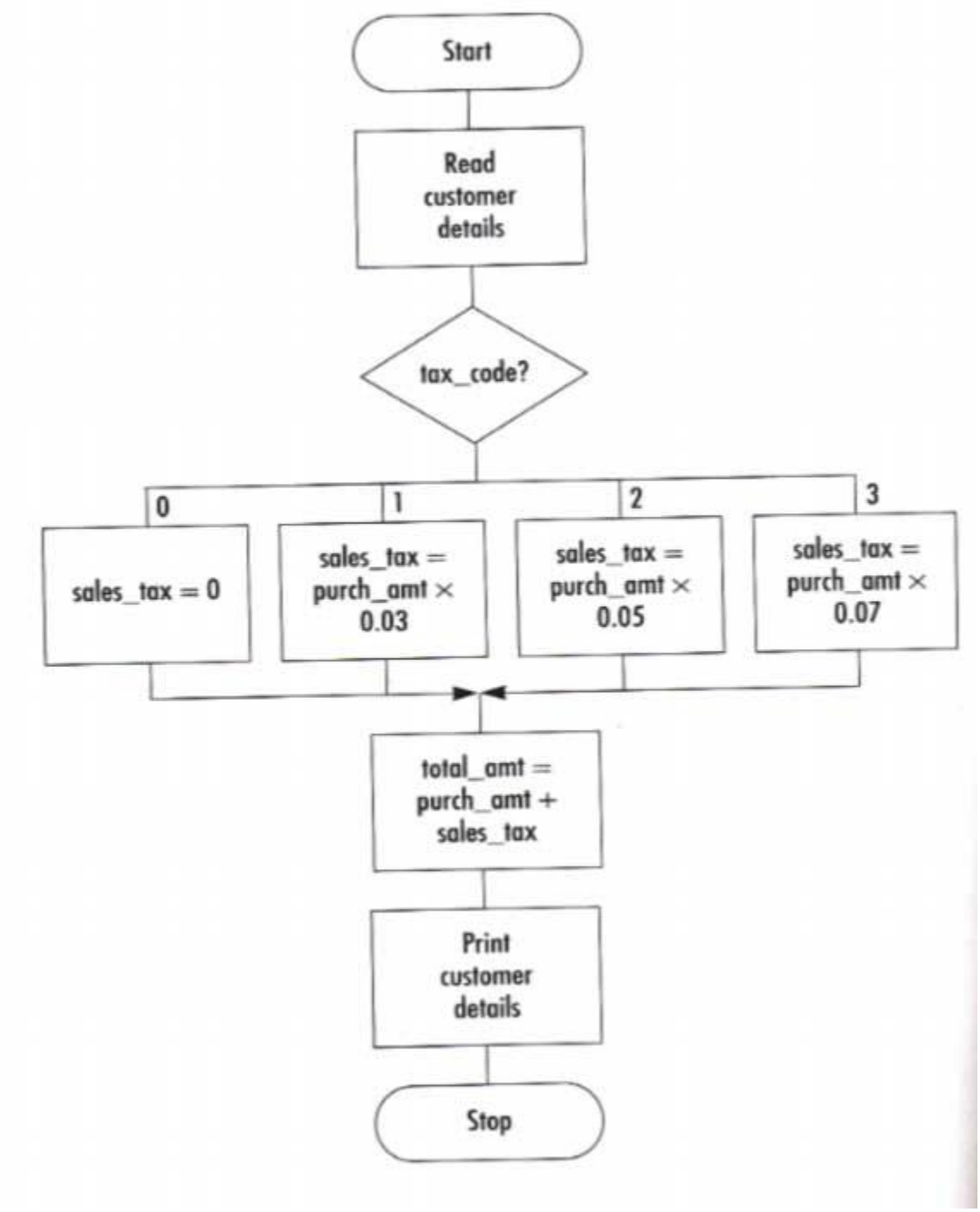
A Defining diagram

Here is the IPO chart.

Input	Processing	Output
cust_name purch_amt tax_code	Read customer details Compute sales tax Compute total amount Print customer details	cust_name purch_amt sales_tax total_amt

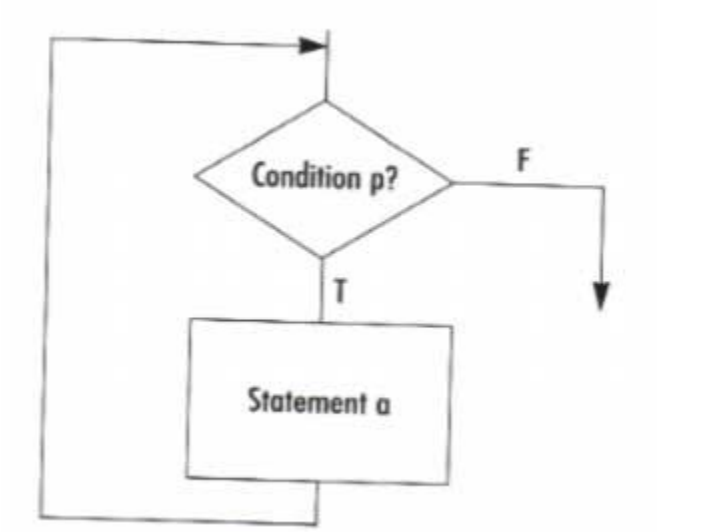
B Solution algorithm

Here is the solution algorithm as a flowchart.



Flowcharts and the Repetition Control Structure

Repetition is used to repeat a group of instructions 0 or more times. It makes for economic programming. Here is the DOWHILE in a flowchart.



Simple Algorithms That Use the Repetition Control Structure

Here are some algorithms from chapter 5 rewritten as flowcharts.

Example 5.1 Fahrenheit-Celsius conversion

A program is to get 15 Fahrenheit temperatures and convert them to Celsius and display them with a "all temperatures processed" message.

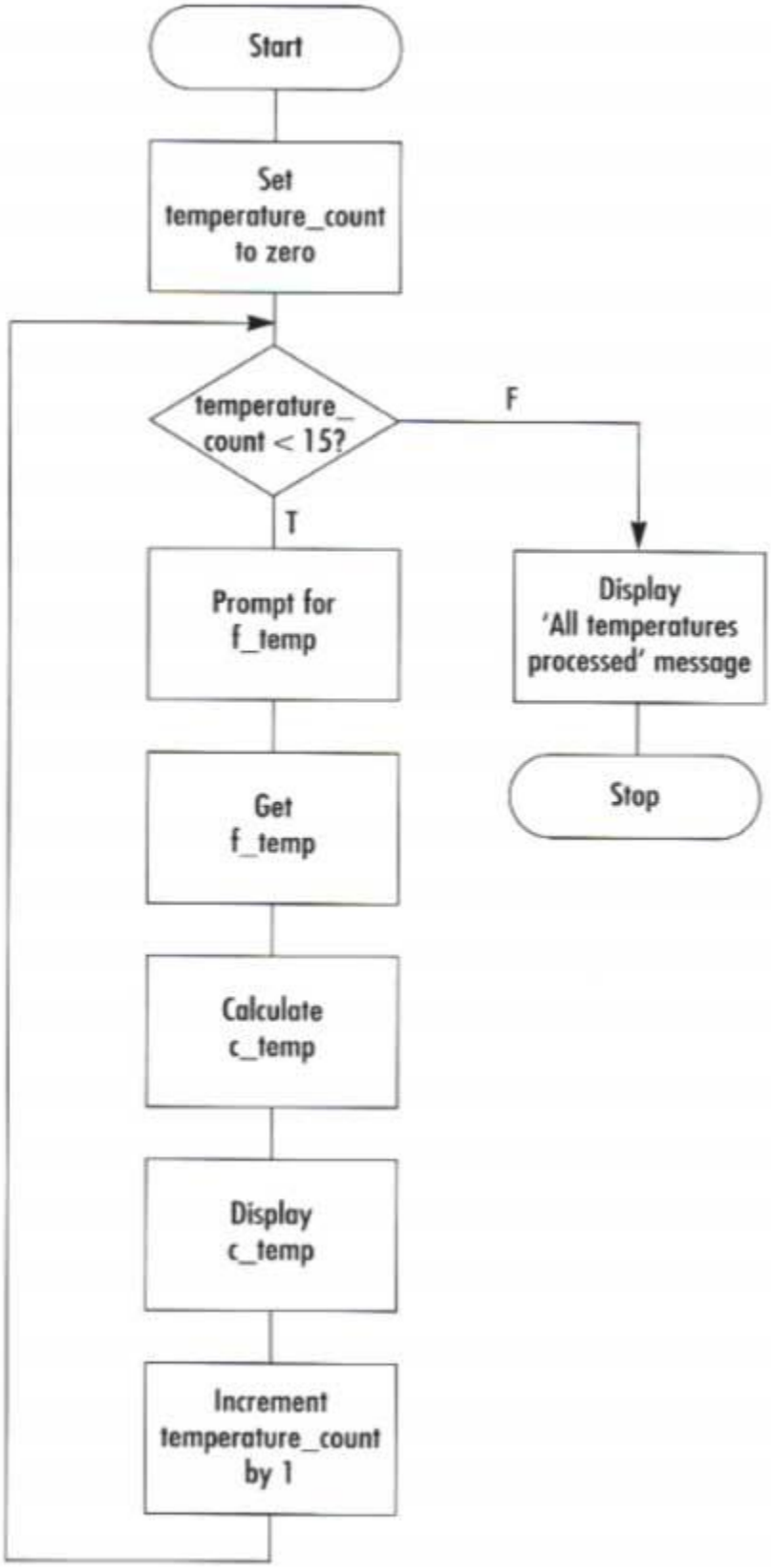
A Defining diagram

Here is the IPO chart.

Input	Processing	Output
f_temp (15 temperatures)	Get Fahrenheit temperatures Convert temperatures Display Celsius temperatures Display screen message	c_temp (15 temperatures)

B Solution algorithm

Here is the solution algorithm rewritten as a flowchart.


Start
Set temperature_count to zero
temperature_ count < 15?
F
T
Prompt for f_temp
Get f_temp
Calculate c_temp
Display c_temp
Increment temperature_count by 1
Display 'All temperatures processed' message
Stop

Example 5.2 Print examination scores

A program is to read and print a series of student's names and exams scores. Then it is to print the average score..

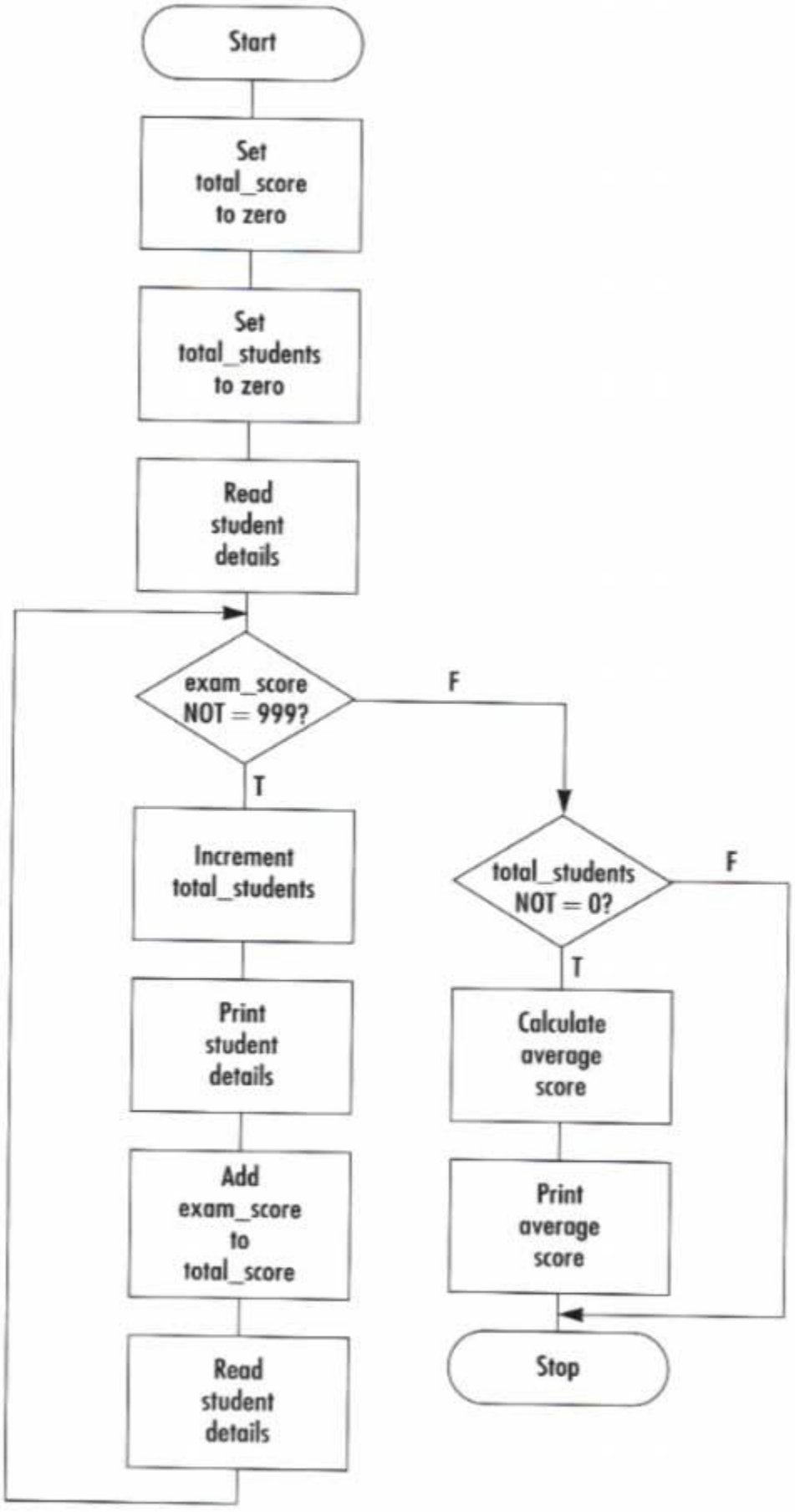
A Defining diagram

Here is the IPO chart.

Input	Processing	Output
name exam_score	Read student details Print student details Compute average score Print average_score	name exam_score average_score

B Solution algorithm

Here is the Solution algorithm as a flowchart.


Start
Set total_score to zero
Set total_students to zero
Read student details
exam_score NOT = 999?
F
T
Increment total_students
Print student details
Add exam_score to total_score
Read student details
total_students NOT = 0?
F
T
Calculate average score
Print average score
Stop

Example 5.3 Process student enrolments

A program is to read a student records file and print out only the records of students taking Programming I course.

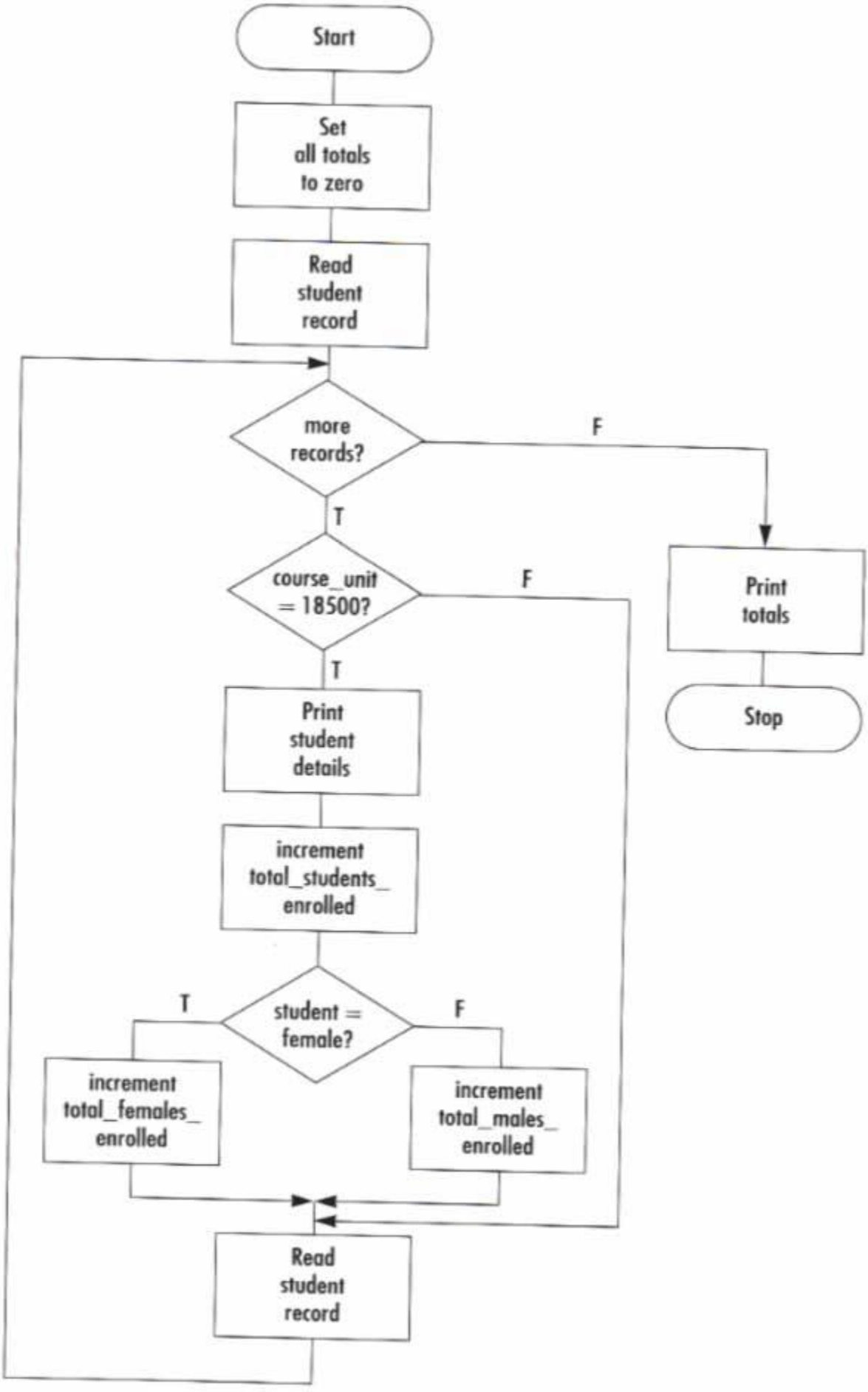
A Defining diagram

Here is the IPO chart.

Input	Processing	Output
student_record • student_no • name • address • postcode • gender • course_unit	Read student records Select student records Print selected records Compute total females enrolled Compute total males enrolled Compute total student enrolled Print totals	selected student records totals

B Solution algorithm

Here is the Solution algorithm as a flowchart.


Start
Set all totals to zero
Read student record
more records?
F
T
course_unit = 18500?
F
T
Print totals
Stop
Print student details
increment total_students_ enrolled
student = female?
T
F
increment total_females_ enrolled
increment total_males_ enrolled
Read student record

Example 5.4 Process inventory items

The program is to read an inventory file and produce a low stock items report.

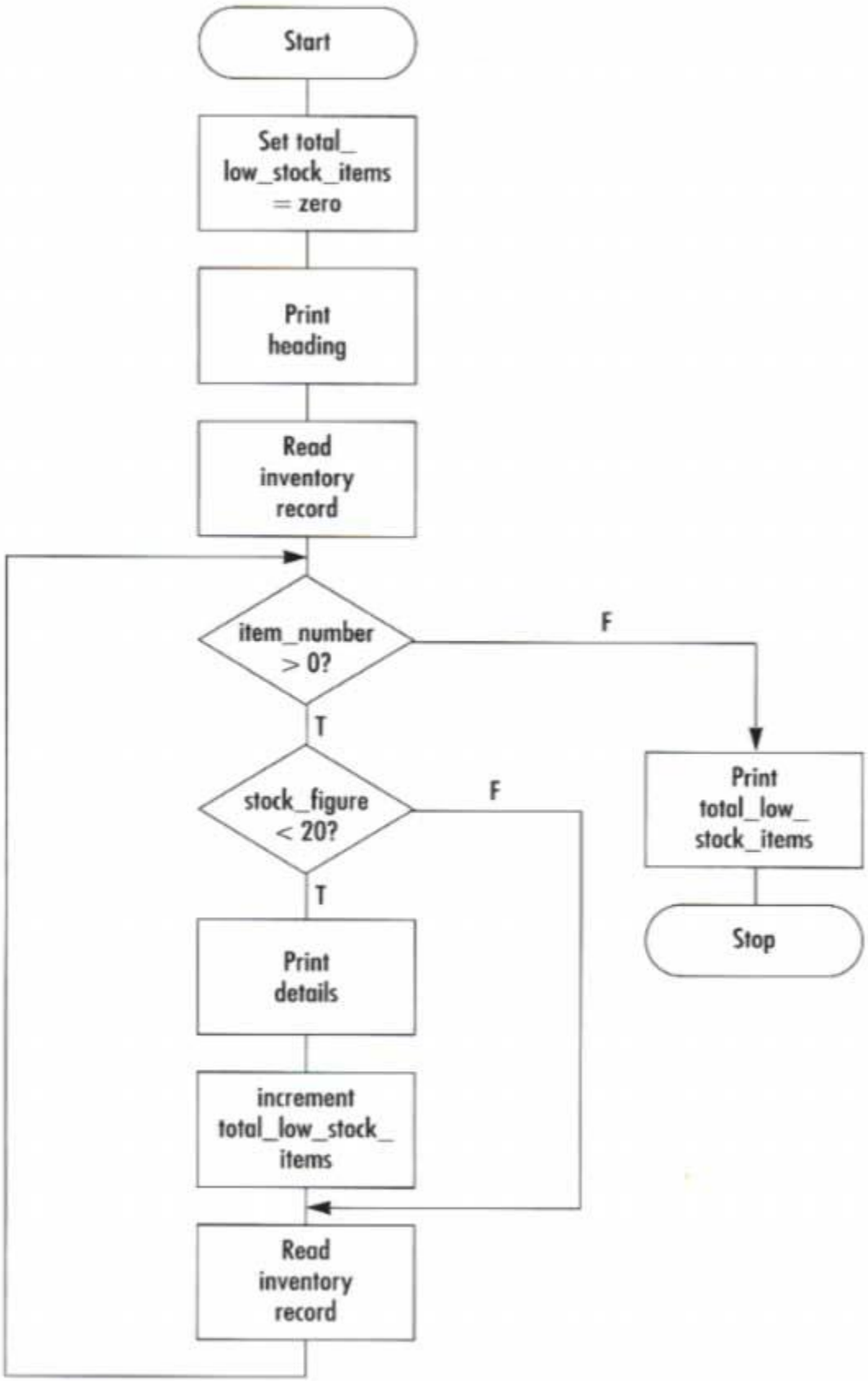
A Defining diagram

Here is the IPO chart

Input	Processing	Output
inventory record • item_number • item_description • stock_figure	Read inventory records Select low stock items Print low stock records Print total low stock items	heading selected records • item_number • item_description • stock_figure total_low_stock_items

B Solution algorithm

Here is the Solution algorithm as a flowchart.


Start
Set total_ low_stock_items = zero
Print heading
Read inventory record
item_number > 0?
F
T
stock_figure < 20?
F
T
Print details
increment total_low_stock_ items
Read inventory record
Print total_low_ stock_items
Stop

Flowcharts and Modules

Flowcharts can use the predefined symbol to break a program up into modules

Example 8.1 Read three characters

Here is the read three characters and print them out in ascending order program.

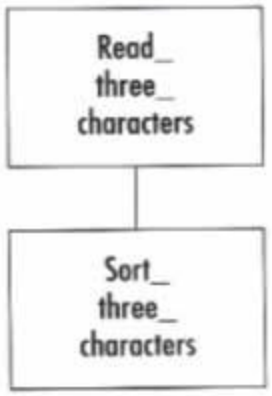
A Defining diagram

Here is the IPO chart.

Input	Processing	Output
char_1 char_2 char_3	Prompt for characters Accept three characters Sort three characters Output three characters	char_1 char_2 char_3

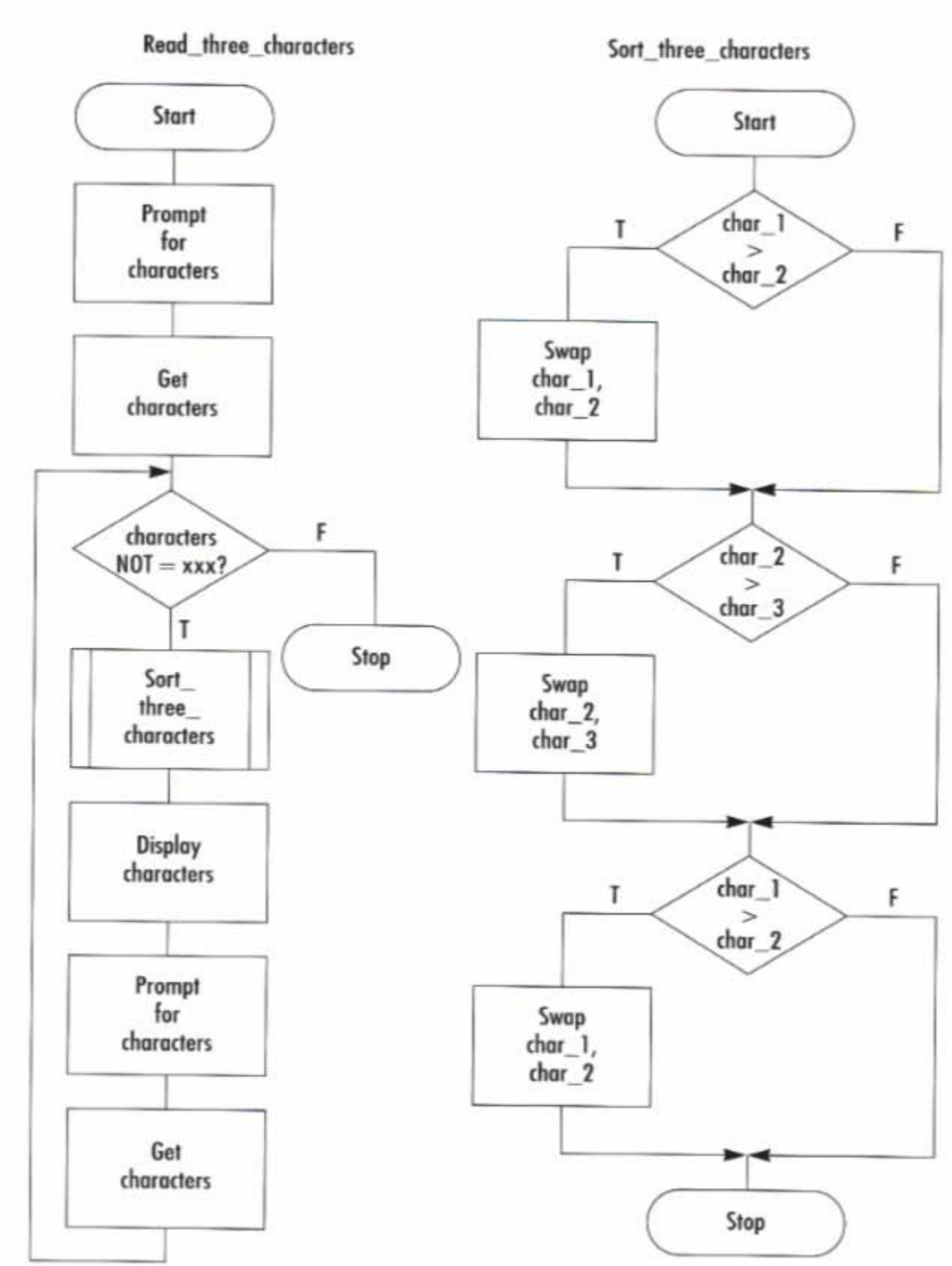
B Hierarchy chart

Here is a possible hierarchy chart.



C Solution algorithm using a predefined process symbol

Here is the Solution algorithm using flowchats.


Read_three_characters
Start
Prompt for characters
Get characters
characters NOT = xxx?
F
T
Stop
Sort_ three_ characters
Display characters
Prompt for characters
Get characters
Sort_three_characters
Start
char_1 > char_2
T
F
Swap char_1, char_2
char_2 > char_3
T
F
Swap char_2, char_3
char_1 > char_2
T
F
Swap char_1, char_2
Stop