

บทที่ 7

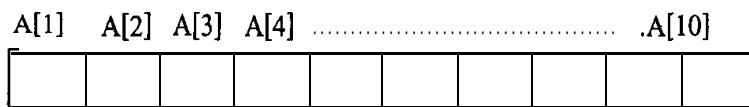
โครงสร้างข้อมูลแบบซับซ้อน (Complex Data Types)

บทนำ การจัดการในพื้นที่ส่วนที่เป็นสมองของคอมพิวเตอร์ เพื่อจัดเก็บข้อมูลนั้น เราไม่สามารถจะใช้ตัวแบบของข้อมูลในลักษณะที่เป็นโครงสร้างข้อมูลแบบง่ายในการจัดการข้อมูลบางลักษณะให้สามารถดำเนินการได้อย่างมีประสิทธิภาพ ดังนั้นจึงมีการสร้างโครงสร้างข้อมูลประเภทซับซ้อนขึ้นเพื่อสนองตอบในการแก้ปัญหาที่ยุ่ยากให้สามารถดำเนินการได้ง่ายเข้า

ข้อมูลที่เป็นโครงสร้างแบบง่ายไม่ว่าจะเป็นชนิดใดที่กล่าวมาอันประกอบด้วย ข้อมูลแบบจำนวนเต็ม ข้อมูลประเภทเลขจำนวนจริง ข้อมูลเชิงอักขระ หรือข้อมูลแบบตรรกะ นั้นเราอาจจะเรียกว่าเป็นข้อมูลแบบเดี่ยวก็ได้ ด้วยสาเหตุที่ว่าข้อมูลประเภทนี้จะอยู่กันเป็นอิสระถือเสมือนเป็นเซลล์ๆ แยกออกจากกัน แต่ละตัวแปรจะไม่มีสัมพันธ์ภาพกันในแง่ของพื้นที่ ปัญหาของการจัดการระบบงานบางอย่างไม่อาจจะอาศัยตัวแปรเหล่านี้ในการจัดการข้อมูลได้ ตัวอย่างเช่น จำนวนนักเรียนที่ลงทะเบียนวิชา CS456 นั้นจะมีการเก็บคะแนนของแต่ละคนแบ่งออกเป็น 3 ส่วนอันประกอบด้วย คะแนนการบ้าน คะแนนสอบกลางภาค และคะแนนสอบปลายปี สมมุติว่าเรามีนักเรียนที่ลงทะเบียนวิชานี้ทั้งสิ้น 200 คน เราจะจัดการกับข้อมูลเหล่านี้ได้อย่างไรในอันที่จะนำคะแนนสอบรวม (คะแนนรวม = คะแนนการบ้าน + คะแนนสอบกลางภาค + และคะแนนสอบปลายปี) จากตัวอย่างที่เสนอมานี้ เราคงไม่สามารถที่จะสร้างตัวแปรประเภทโครงสร้างแบบง่ายมาดำเนินงานนี้ได้เพราะคงจะต้องทำการจับจองจำนวนตัวแปรเดี่ยวๆทั้งสิ้นถึงจำนวน 200×4 เซลล์เพื่อจะนำไปเก็บข้อมูลของนักเรียน 200 คนได้ โดยที่แต่ละคนก็จะมีรายการข้อมูลอันประกอบด้วย คะแนนรวม , คะแนนการบ้าน , คะแนนสอบกลางภาค , และคะแนนสอบปลายปี ดังนั้นจึงมีการพัฒนาสร้างตัวแปรที่มีคุณลักษณะพิเศษขึ้นมาใช้งาน เช่นการสร้างตัวแปรกลุ่มประเภทต่างๆ

โครงสร้างข้อมูลแบบอะเรย์ (The Array Data Structure) โครงสร้างข้อมูลแบบอะเรย์นั้นจัดว่าเป็นตัวแปรแบบกลุ่มมีการกำหนดพื้นที่ให้ข้อมูลประเภทเดียวกันอยู่ด้วยกัน โดยที่พื้นที่ที่จัดเก็บข้อมูลของกลุ่มนั้นจะถูกจัดสรรให้อยู่ต่อเนื่องกัน ตัวอย่างที่ใกล้เคียงกันก็คือเซตของตัวเลขในเรื่องเซตทางคณิตศาสตร์ ซึ่ง จะเห็นว่า สมาชิกของเซตนั้นจะอยู่เรียงลำดับกันไป แต่เงื่อนไขของเซตที่จะเป็นลักษณะการจัดการข้อมูลแบบอะเรย์ได้นั้นจะต้องมีข้อตกลงว่า **ทุกๆสมาชิก (Element)** นั้นจะต้องเป็น **ประเภท (Type) และขนาด (Size) เดียวกันด้วย**

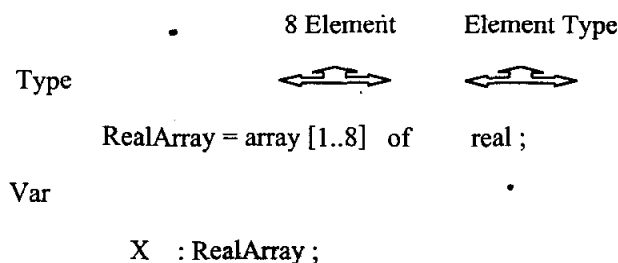
โครงสร้างการจัดการและการจัดสรรพื้นที่สำหรับตัวแปรที่เป็นอะเรย์



เช่นเดียวกับการจัดสรรพื้นที่สำหรับข้อมูลแบบง่ายที่จะต้องมีการประกาศการจัดสรรพื้นที่ ดังนั้นอะเรย์ก็จำเป็นที่จะต้องมีการของจัดสรรพื้นที่เช่นกัน การใช้ตัวแปรแบบอะเรย์จัดว่าเป็นมาตรฐานของการสร้างตัวแปรที่มีปรากฏในทุกภาษาคอมพิวเตอร์ เพียงแต่ว่าบางภาษาอาจจะยอมให้มีข้อมูลแบบอะเรย์ในลักษณะมิติเดียวเท่านั้น เช่นภาษา RPG ในขณะที่ภาษาคอมพิวเตอร์ทั่วไปยอมให้มีการกำหนดอะเรย์ประเภทหลายมิติ (Multi Dimension) ได้ด้วย เช่นภาษา Pascal , BASIC , Fortran , C เป็นต้น

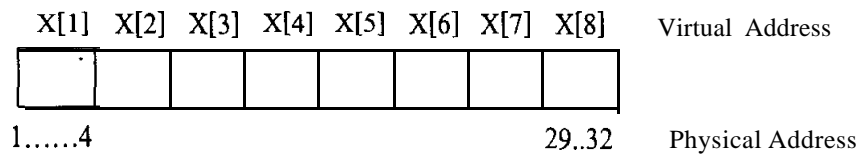
การประกาศตัวแปรแบบอะเรย์ (Declaring Arrays) แต่ละภาษาก็จะมีการกำหนดไวยากรณ์ (Syntax) ของการประกาศจับจองขนาดพื้นที่ของอะเรย์แตกต่างกันไป ดังตัวอย่างการใช้คำสั่งที่ปรากฏในภาษาคอมพิวเตอร์ต่างๆ

ตัวอย่างการใช้ภาษา Pascal กำหนดอะเรย์มิติเดียวและหลายมิติ

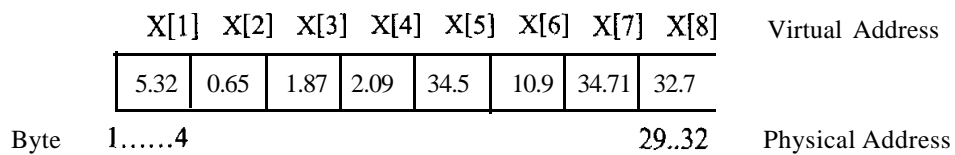


ผลที่เกิดขึ้นจากการจัดสรรพื้นที่ของคอมไพเลอร์ตามคำสั่งนี้ก็คือ จะมีการจัดสรรพื้นที่ให้ทั้งหมด 8 เซลล์แต่ละเซลล์จะอยู่เรียงติดต่อกันไป ในกรณีที่ระบบนั้นกำหนดให้ตัวแปรจำนวนจริงมีขนาดเท่ากับ 4 Bytes ก็จะได้พื้นที่ขนาด $8 \times 4 = 32$ Bytes ให้กับตัวแปรกลุ่ม X โดยที่การอ้างอิงถึงเซลล์แต่ละเพื่อจะนำข้อมูลไปจัดเก็บหรือจะดึงออกมาใช้งานนั้นจะอ้างอิงโดยการระบุถึงชื่อของตัวแปรและกำกับด้วยตำแหน่งที่อ้างอิง เช่น X[1] จะหมายถึงตำแหน่งที่ 1 และ X[4] จะหมายถึงตำแหน่งที่ 4 ของอะเรย์ X เป็นต้น

โครงสร้างของข้อมูลที่จัดสรรแล้ว

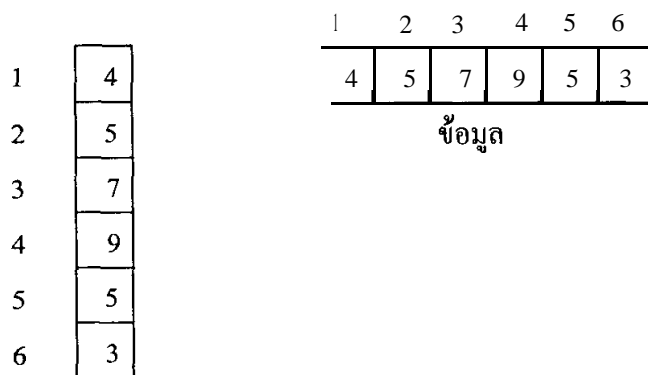


Virtual Address ก็คือตำแหน่งเสมือนที่อ้างอิงในการใช้งาน ทำให้ง่ายในการติดต่อ โดยที่จะต้องมีการนำ Virtual Address หรือที่เรียกว่า X[1] ,...,X[8] ไปค้นหา Actual Address ต่อไปในที่นี้ Actual Address ก็คือตำแหน่งจริงของ Byte ที่ได้รับการจัดสรรเราอาจจะเรียกตำแหน่งจริงๆนี้ภายใน Memory ว่าเป็น Physical Address



ภายหลังการจัดเก็บข้อมูลเข้าไปในอะเรย์ที่สร้างแล้วเราสามารถจะเรียกข้อมูลเหล่านี้มาดำเนินการได้โดยการระบุเป็น ชื่อของ ตัวแปรและตามด้วยดัชนีสืบค้นดังนี้คือ X[I] ; I = 1,...,8

เนื่องจากการจัดการข้อมูลของอะเรย์จะมีลักษณะเป็นการจัดการที่ต่อเนื่องกันดังนั้นเราจึงสามารถใช้ประโยชน์จากคุณสมบัตินี้ได้หลายประการ เช่นการเรียงลำดับ (Sorting) การค้นหา (Searching) ตลอดจนการดำเนินงานอย่างอื่นที่จะนำไปประยุกต์ต่อไปด้วยคุณสมบัติที่ปรากฏนี้ เราอาจจะมองวิธีการจัดการกับอะเรย์แบบมิติเดียวได้ใน 2 รูปแบบคือมองเป็น Column Vector หรือ Row Vector



การกำหนดตำแหน่งของ Index ในอะเรย์นั้นอาจจะดำเนินการโดยการกำหนดค่าเริ่มต้นเป็นเลขอะไรก็ได้เพื่อความเหมาะสมในการใช้งาน เช่นเราจะทำการเก็บข้อมูลยอดขายสินค้าในบริษัทตั้งแต่ปีพศ.2530 - ปีพศ.2540 เราสามารถกำหนดอะเรย์ได้ดังนี้คือ

STOCK [2530] , STOCK[2531] , , , STOCK[2540]

การกำหนดเช่นนี้นั้น ในทางปฏิบัติที่จะอ้างไปใช้งานจะเข้าใจง่าย เพราะเพียงแค่กำหนด YEAR คือปีที่ต้องการเช่น ปีพศ. 2530,2531, ..., 2540 ถ้าเราต้องการมูลค่าของสินค้าปีที่ต้องการได้โดยการระบุว่า STOCK [YEAR] ได้ตามที่ต้องการ

การประกาศกำหนดอะเรย์ในแต่ละภาษานั้นจะมีลักษณะคล้ายคลึงกันยกเว้นในรายละเอียดบางอย่างอาจจะแตกต่างกันบ้าง ตัวอย่างเช่น

ภาษาเบสิก ใช้คำสั่งกำหนดดังนี้ DIM X[10] ผลก็คือจะได้อะเรย์ที่มีเซลล์แต่ละเซลล์เป็นเลขจำนวนจริง (Real Number) โดยที่อะเรย์นั้นจะเริ่มทำการจับจองตั้งแต่ X[0] , X[1] , ..., X[10] ซึ่งมีจำนวนทั้งสิ้น 11 เซล ในกรณีที่ต้องการให้จองเพียง 10 เซลโดยมีค่าเริ่มต้นตั้งแต่ X[i] , ..., X[10] เราจะต้องระบุคำสั่ง OPTION BASE 1 ก่อนหน้าที่จะประกาศของอะเรย์ X เนื่องจากว่าในภาษาเบสิกนั้น จะมีการกำหนดคุณลักษณะปริยาย (Default) เป็น OPTION BASE 0 ให้กับระบบการจัดสรรพื้นที่ของอะเรย์

ภาษาซี การใช้คำสั่งกำหนดพื้นที่ เช่น float x[10]; จะส่งผลให้มีการจองพื้นที่ให้ทั้งหมด 10 เซล โดยที่เซลล์เริ่มต้นคือ x[0] , ..., x[9] โดยที่แต่ละเซลล์จะเก็บจำนวนที่เป็นจำนวนจริงได้

ผลของการกำหนดพื้นที่ และ ดัชนีสืบค้นของอะเรย์ในแต่ละภาษานั้นจะโยงไปถึงการอ้างถึงตำแหน่งของอะเรย์ในการทำงาน เช่นถ้าใช้ภาษาซีในการทำงาน เมื่อประกาศกำหนดพื้นที่ float x[10]; เวลาใช้งานจะอ้างถึง x [10] ไม่ได้ เราจะสามารถอ้างได้ ตั้งแต่ x[0] , ..., x[9] ได้เท่านั้น

x[0]	x[1]	x[2]	x[3]	x[4]	x[5]	x[6]	x[7]	x[8]	x[9]
5.32	0.65	1.87	2.09	34.5	10.9	34.71	32.7	43.2	4.07

Byte 1.. .4 Physical Address

ภาพการจัดการจัดสรรพื้นที่บนภาษาซี ในการกำหนดอะเรย์ x [10]

ตัวอย่างการจองอะเรย์ชนิดต่างๆ โดยใช้ภาษาปาสคาล

Type

IntArray = array [1..10] of Integer ;
 LongintArray = array [1..10] of Longint ;
 RealArray = array [1..10] of Real ;
 BoolArray = array [1..10] of Boolean ;
 Str8ch = array [1..8] of char ;

Var

X : IntArray ;
 Y : LongintArray ;
 Z : RealArray ;
 W : BoolArray ;
 L : Array [1..10] of Str8ch ;

จากตัวอย่างการประกาศอะเรย์ที่ปรากฏนี้จะเห็นได้ว่าอะเรย์ L นั้นอาจจะค่อนข้างซับซ้อนเพราะว่าจะเป็นอะเรย์ซ้อนอะเรย์ดัง โครงสร้างที่ปรากฏดังนี้

L[1,1]L[1,2].....L[1,8]

L[1]	T	H	A	I	L	A	N	D
L[2]								
L[3]								
L[4]								
L[5]								
L[6]								
L[7]								
L[8]								

การระบุตำแหน่งในการเก็บข้อมูลนั้นจะต้องระบุด้วยการใช้ดัชนี 2 ตัวเพื่อระบุตำแหน่งคือ L[I,J] ; I = 1,2,...,8 ; J = 1,2,...,8 โดยที่เซลล์ที่ระบุนั้นสามารถเก็บอักขระได้เพียง 1 อักขระเพราะมีพื้นที่เพียง 1Byte เท่านั้น ในกรณีของภาษาปาสคาลนั้นบางรุ่นของภาษานี้จะ

กำหนดรูปแบบการใช้ข้อมูลลักษณะนี้ให้ง่ายเข้า โดยการกำหนด ข้อมูลที่มีโครงสร้างเป็น String ให้ใช้ โดยที่ความหมายของ String ก็คือชุดของอักขระนั่นเอง ดังนั้นด้วยการยอมให้มีการกำหนด โครงสร้างข้อมูลแบบ String จึงทำให้เราอาจจะใช้คำสั่งนี้ในการสร้าง อะเรย์ L

Type

```
Str8ch = String[8];
```

Var

```
L : Array [1..8] of Str8ch;
```

การใช้คำสั่งนี้ในการจัดสรรพื้นที่ให้อะเรย์ทำให้การจัดการพื้นที่ดังกล่าวง่ายขึ้น เช่นถ้าเรา ระบุว่า L[1] ก็จะหมายถึงพื้นที่ที่สามารถเก็บข้อมูลเป็นสายอักขระยาวได้สูงสุดคือ 8 ตัวอักขระ เป็นต้น แต่ก็ให้ระลึกไว้ว่าการที่เราดำเนินการเช่นนี้จะมีข้อเสียเพราะ โครงสร้างข้อมูลแบบ String ไม่ใช่คุณสมบัติที่เป็นมาตรฐานของภาษาปาสคาล

เพื่อให้ผู้อ่านเข้าใจดียิ่งขึ้นในการจัดเก็บข้อมูลให้ดูตัวอย่างการสร้าง และการจัดเก็บข้อมูล แบบอะเรย์ในโครงสร้างต่างๆจากตารางตัวอย่างคำสั่งของภาษาปาสคาลต่อไปนี้

Array Type	Subscript Type	Application
<pre>Type NameArray = array[1..10] of char ; Var Name : NameArray ;</pre>	Integer Subrange	<pre>Name[1] := 'A'</pre> Store a person's name using 10 character
<pre>Type Temp = array[-10..10] of real ; Var Fahreheit : Temps ;</pre>	Integer Subrange	<pre>Fahrenheit[0] := 32.0</pre> store Fahrenheit temperatures equivalent to -10 through 10 degrees Celsius
<pre>Type Counters = array['A'..'Z'] of integer ; Var LetterCounter :Counters ;</pre>	Character Subrange	<pre>LetterCount['A'] := 0;</pre> stores the numbers of times each uppercase letter occurs
<pre>Type Flays = array['A'..'Z'] of boolean ; Var LetterFound : Flays ;</pre>	Character Subrange	<pre>LetterFound['X'] := False</pre> ,stores a set of Boolean values indicating which letters occur and which do not
<pre>Type BoolCounts = array[Boolean] of integer ; Var Answer : BoolCounts ;</pre>	Boolean	<pre>Answers[True] := 15 ;</pre> stores the number of true answers and false answers to a quiz

ตัวอย่างการนำโครงสร้างของอะเรย์รูปแบบต่างๆ ไปประยุกต์ในโรงงานแห่งหนึ่งในการจัดเก็บข้อมูลของพนักงาน

```

const
    NumEmp = 8 ;
Type
    EmpRange = 1 ..NumEmp ;
    EmpArray = array[EmpRange] of Boolean ;
    Day =[Sunday,Monday,Tuesday,Wednesday,Thursday,Friday,Saturday];
    DayArray = array [Day] of real ;
Var
    Vacation    : EmpArray ;
    PlantHours : DayArray ;

```

จากวิธีการประกาศตัวแปรที่ปรากฏข้างบนนี้ ส่งผลให้ มีการสร้าง Type ให้กับ EmpArray มีขนาด Subrang คือ 1.. NumEmp (1..8) ดังนั้นเมื่อมีการจองพื้นที่ให้กับตัวแปร Vacation จำนวน 8 เซล โดยที่แต่ละเซลล์จะมีความสามารถในการจัดเก็บ Boolean Value (ซึ่งมีสัญญาณเป็น True หรือ False เท่านั้น) เจตนาของการเก็บข้อมูลในงานนี้ก็คือถ้าเซลล์ใดของอะเรย์ Vacation [I] มีค่าเป็น True ก็หมายความว่า พนักงานคนที่ I นั้นจะเป็นวันหยุดของเขา สมมุติว่าพนักงานคนที่ 1,3,5 และ 7 เป็นวันหยุดของเขาส่วนพนักงานอื่นๆยังทำงานอยู่ดังนั้นข้อมูลที่ปรากฏในอะเรย์จะปรากฏดังนี้

Vacation	
[1]	True
[2]	False
[3]	True
[4]	False
[5]	True
[6]	False
[7]	True
[8]	False

ส่วนอะเรย์ของ DayArray นั้นจะมีดัชนีแบบพิเศษคือเป็น Enumerate Type ของ Day คือเป็น Sunday ,Monday,...,Saturday ดังนั้นการจัดสรรพื้นที่สำหรับ PlantHours ก็คือแต่ละเซลล์ของ PlantHours จะสามารถเก็บข้อมูลที่เป็นประเภทจำนวนจริง

PlantHours

[Sunday]	0.0
[Monday]	8.0
[Tuesday]	16.0
[Wednesday]	24.0
[Thursday]	8.0
[Friday]	16.0
[Saturday]	0.0

จากอะเรย์ของ PlantHours นี้จะแสดงให้เห็นว่า โรงงานแห่งนี้จะปิดกิจการในวัน Sunday และ Saturday ส่วนวัน Monday , Thursday จะทำงานเพียงกะเดียวคือ 8 ชั่วโมง ในขณะที่วัน Tuesday ,Friday จะทำงาน 2 กะคือ 16 ชั่วโมง ส่วนวัน Wednesday จะทำงาน 3 กะคือ 24 ชั่วโมง ตัวอย่างคำสั่งในการจัดสรรพื้นที่นี้ สามารถที่จะปรับได้โดยการตัด const ทิ้งไปดังนั้นคำสั่งจะเขียนได้ดังนี้

```
Type EmpArray = array[1..8] of Boolean ;
```

การดำเนินการกับข้อมูลในอะเรย์

โดยปกติแล้วธรรมชาติของการใช้งานกับเซลล์ในอะเรย์นั้นมักจะกระทำแบบเรียงลำดับ (Sequential Access) โดยเริ่มต้นจากเซลล์แรกของอะเรย์จนถึงเซลล์สุดท้าย เรียงลำดับกันไป โดยที่จะควบคุมการเดินทางในแต่ละตำแหน่งนั้นมักใช้ควบคู่กับการใช้ For Loop ในการควบคุมเสมอ ตัวอย่างต่อไปนี้นี้จะแสดงวิธีการรับข้อมูลจากแป้นพิมพ์เข้าไปเก็บในอะเรย์ที่ตำแหน่งเริ่มต้นจนถึงตำแหน่งสุดท้าย

```

Program Demo1 ;
Type
    IndexRange = 1..10;
    IntArray = array [IndexRange] of Integer ;
Var
    Cube : IntArray ;
    I : Integer ;
Begin
    ,
    For I := 1 to 10 do
        Cube[I] := I * I * I ;

End.

```

ผลของการปฏิบัติงานของโปรแกรมนี้จะทำให้ข้อมูลที่ปรากฏในแต่ละเซลล์ของอะเรย์ Cube

Array Cube

[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]
1	8	27	64	125	216	343	512	729	1000

ตัวอย่างที่ 7.1 จะแสดงการหาคะแนนเฉลี่ย (Mean) และคะแนนส่วนเบี่ยงเบนมาตรฐาน (SD) ของคะแนนนักเรียนที่เข้าสอบในวิชา CT000 จำนวน 30 คน โดยมีการตัดเกรดอิงกลุ่มดังนี้คือ

ถ้าคะแนนสอบน้อยกว่าค่าของ $Mean - 2.0 * SD$ จะได้เกรด 'F'

ถ้าคะแนนสอบอยู่ในช่วงของ $Mean - 2.0 * SD$ ถึง $Mean + 2.0 * SD$ จะได้เกรด 'P'

ถ้าคะแนนสอบมากกว่าหรือเท่ากับค่าของ $Mean + 2.0 * SD$ จะได้เกรด 'G'

จากความต้องการของปัญหานี้เราจะออกแบบโครงสร้างข้อมูลและขั้นตอนในการปฏิบัติงานได้ดังนี้

(การคำนวณค่า Mean และ SD จะใช้สูตรดังนี้ $\text{Mean} = \frac{\sum_{i=1}^N X_i}{N}$)

และ $\text{SD} = \sqrt{\frac{\sum_{i=1}^N (X_i - \bar{X})^2}{N-1}}$ แต่เนื่องจากสูตรที่เขียนมานี้จะเหมาะกับการทำงานด้วยการคำนวณโดยมนุษย์มากกว่าที่จะใช้โปรแกรม ด้วยเหตุผลที่ว่าถ้าใช้สูตรนี้ในการประมวลผลเราจะต้องดำเนินการคำนวณค่าของ \bar{X} มาก่อนหลังจากนั้นจึงนำค่า \bar{X} ที่ได้มาหา

$$\frac{\sum_{i=1}^N (X_i - \bar{X})^2}{N-1}$$

จากนั้นจึงนำค่าที่ได้ไปหารค่ารากที่สอง ตามแนวทางนี้เราจะต้องวนทำงานใน Loop ถึง 2 ครั้ง โดยที่ครั้งแรกจะวน Loop เพื่อหาค่าของ $\sum_{i=1}^N X_i$ เพื่อนำไปหาค่า \bar{X} ต่อไป ส่วน Loop ที่สองเพื่อจะหา

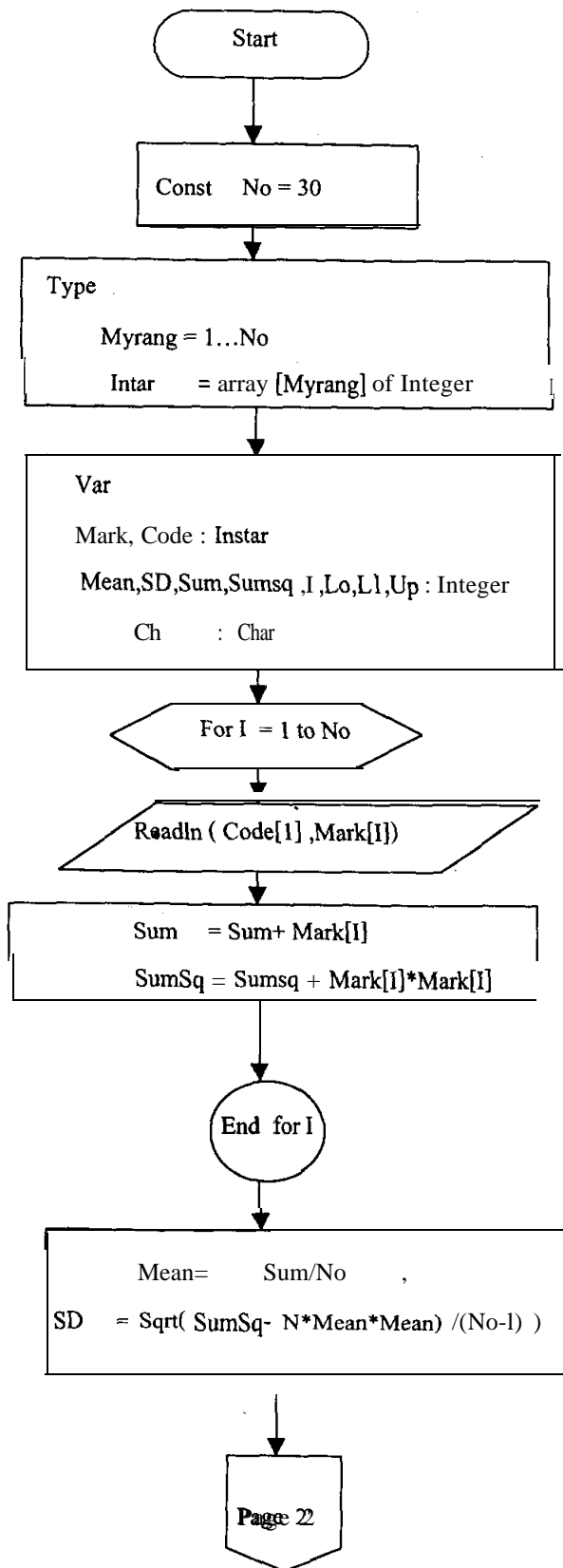
$$\sum_{i=1}^N (X_i - \bar{X})^2$$

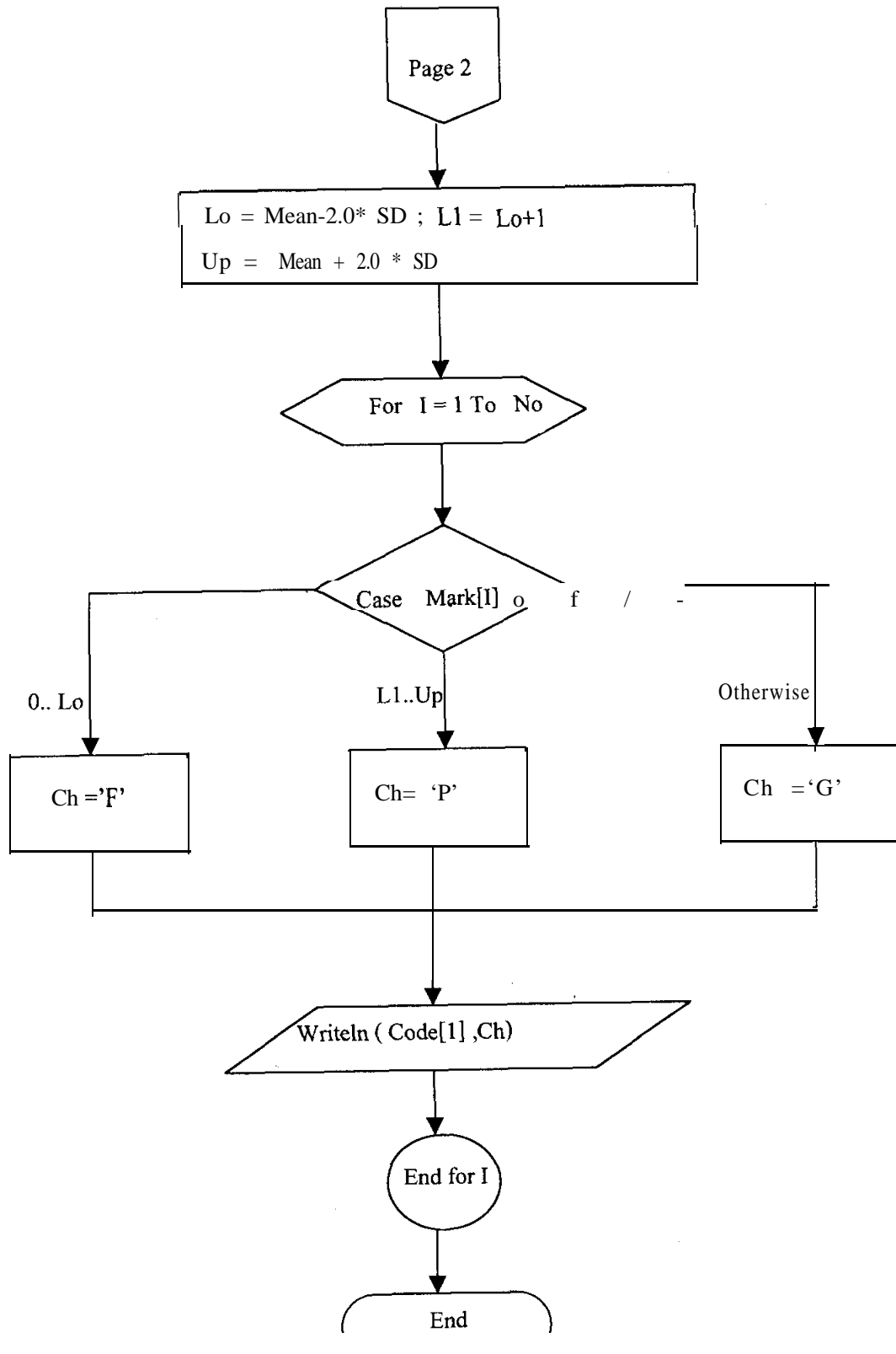
ซึ่งโดยแนวทางนี้จะทำให้เสียเวลาในการวน Loop ถึงสองครั้ง ดังนั้นเราจะใช้สูตรในการคำนวณค่า SD ใหม่ดังนี้คือ

$$\frac{\sum X_i^2 - N\bar{X}^2}{N-1}$$

ซึ่งเป็นสูตรที่เหมาะสมกับการคำนวณด้วย Programming Method เพราะในการทำงานวนเพื่อคำนวณค่า $\sum_{i=1}^N X_i$ เราจะสั่งให้ดำเนินการหา $\sum_{i=1}^N X_i^2$ พร้อมกันไปด้วย ดังนั้นการทำงานประมวลผลงานนี้จะลดเวลาลงไปด้วย

การดำเนินงานของโปรแกรมแนวทางที่ออกแบบได้จะปรากฏโดยใช้ผังโปรแกรมต่อไปนี้





ตัวอย่างที่ 7.2 กำหนดความต้องการดังนี้ ต้องการสร้างสารสนเทศเพื่อการตัดสินใจในส่วนของการจัดการของแผนกขาย โดยที่ข้อมูลที่เก็บรวบรวมเพื่อสร้างสารสนเทศจะประกอบด้วยข้อมูลของลูกค้าที่บริโภคสินค้าของบริษัทอื่นประกอบด้วย อายุ และเพศ สารสนเทศที่ต้องการเพื่อสนับสนุนในการบริหารการขายปรากฏดังนี้คือ รายงานฉบับที่ 1 มีรูปแบบดังนี้

อายุ	จำนวนคน
< 15	XXX
16-25	XXX
26-35	x x x
36- 45	x x x
> =46	x x x

และงานฉบับที่ 2 มีรูปแบบดังนี้

อายุ	เพศ ชาย	เพศหญิง
< 15	x x x	x x x
16-25	XXX	XXX
26-35	XXX	XXX.
36- 45	XXX	XXX
> =46	x x x	x x x

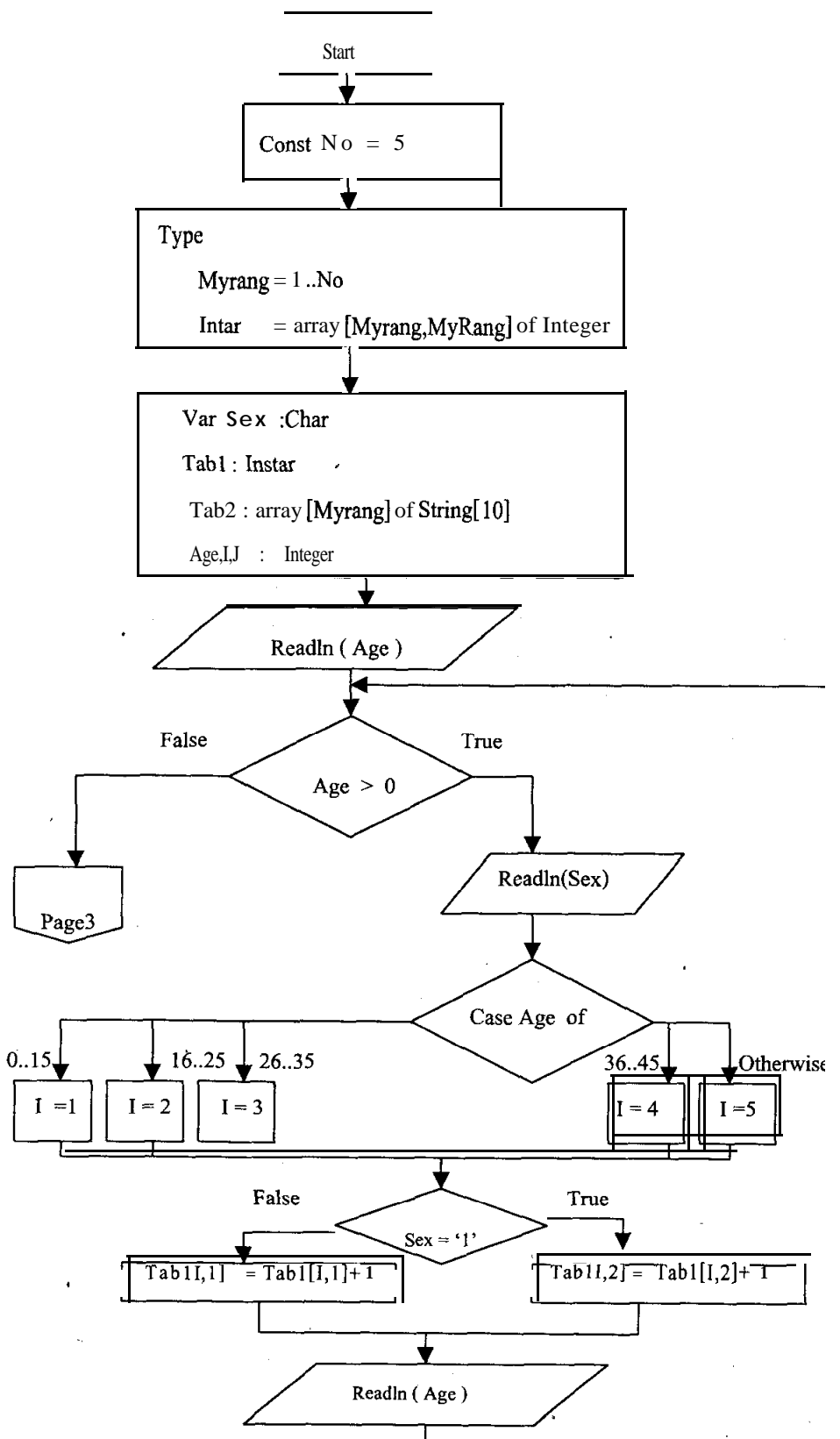
การออกแบบโครงสร้างข้อมูลเพื่อที่จะสามารถสร้างรายงานทั้ง 2 ฉบับนี้ก็คือ เราจะยึดเอาตารางจากรายงานฉบับที่ 2 เป็นแนวทางในการเก็บข้อมูลที่จะได้จากการรับข้อมูลแล้วทำการคัดเลือกใส่ในอะเรย์ที่กำหนดให้ พิจารณาจากอะเรย์ของตารางที่ 2 เราจะกำหนดอะเรย์ ขนาด 2 มิติในการจัดเก็บข้อมูลแทนที่จะเป็นอะเรย์มิติเดียวเพราะข้อมูลในแต่ละเซลล์จะมีลักษณะเดียวกันทั้งประเภทและขนาดคือเป็นข้อมูลของเลขจำนวนเต็ม การจัดการในโครงสร้างของอะเรย์ 2 มิติทำให้การประมวลผลข้อมูลก่อนจะง่าย ภายหลังเมื่อประมวลผลได้อะเรย์ดังกล่าวนี้แล้วเราสามารถที่จะนำข้อมูลที่แยกเป็น 2 เพศในแต่ละช่วงอายุมารวมแสดงผลได้

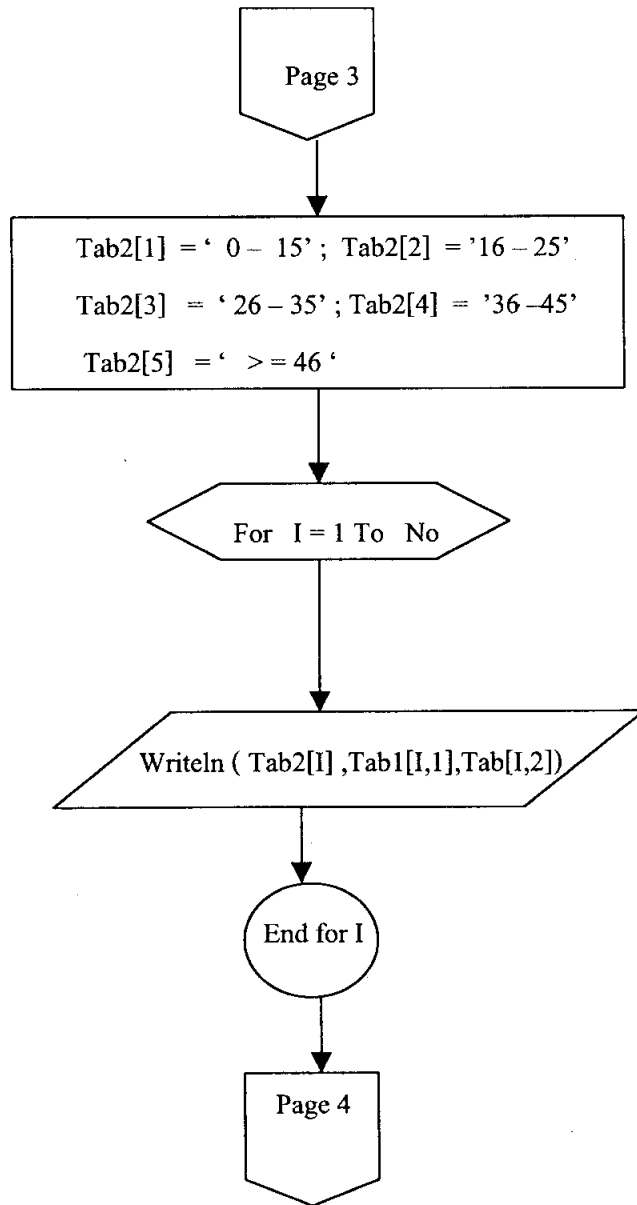
ข้อมูลของลูกค้าแต่ละคนนั้นเราไม่จำเป็นจะต้องจัดเก็บไว้เพราะผลลัพธ์ที่ได้สามารถดำเนินการโดยไม่ต้องเก็บข้อมูลแต่ละคนไว้ในอะเรย์ ซึ่งจะส่งผลให้ต้องใช้อะเรย์ขนาดใหญ่เนื่องจาก ถ้าเราจะต้องใช้อะเรย์ขนาดใหญ่มาก นอกจากนั้นเรายังมีข้อจำกัดที่ว่าปัญหานี้เราไม่ทราบว่ามีลูกค้ากี่คนเรา

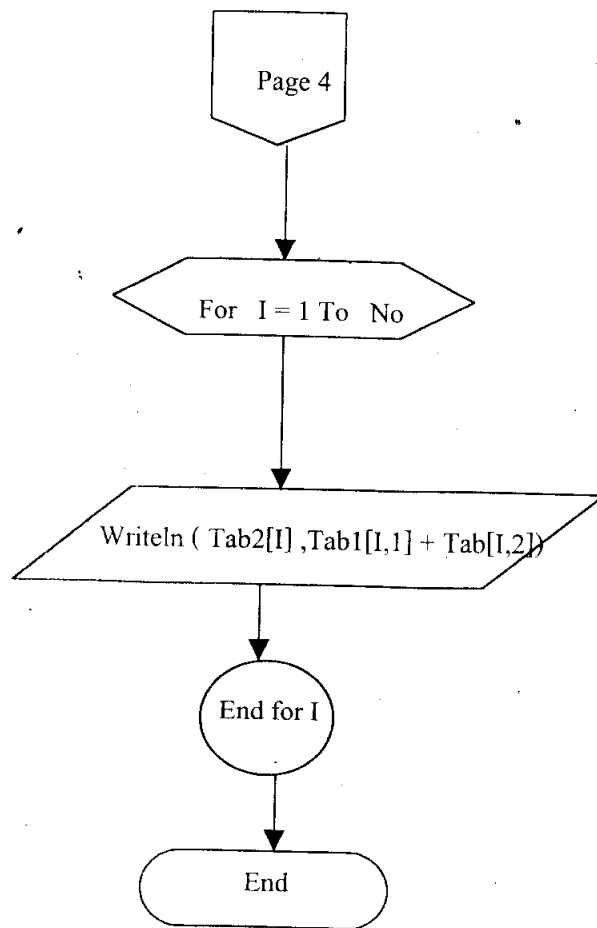
จึงไม่สามารถจับจองพื้นที่ได้เพราะการจองพื้นที่ให้ข้อมูลที่เป็นอะเรย์นั้นจะต้องดำเนินการในส่วนการ Compile ของโปรแกรม เราจึงเรียกว่า อะเรย์เป็น ตัวแปรแบบสถิต (Static Variable) เมื่อเป็นเช่นนี้ในทางปฏิบัติถ้าจำเป็นจะต้องกำหนดพื้นที่อะเรย์โดยไม่ทราบขนาดที่แท้จริง เราก็สามารถดำเนินการได้โดยจองเผื่อไว้ซึ่งจะมีข้อเสียในกรณีที่ใช้ไม่หมดก็จะถูกครอบครองไว้เป็นการเสียพื้นที่ และในขณะเดียวกันถ้ามีข้อมูลมากกว่าที่จองไว้ก็ไม่สามารถนำไปเก็บได้ทำให้ระบบล้มเหลวในการที่จะนำไปประมวลผลต่อไป ดังนั้นในกรณีที่เรารู้แล้วว่าไม่จำเป็นที่จะต้องทำการเก็บข้อมูลไว้ในอะเรย์ เราก็ไม่จำเป็นจะต้องจองอะเรย์ ดังเช่นการออกแบบโปรแกรมในตัวอย่างที่ 2 นี้

การออกแบบโครงสร้างตัวแปร การจัดสรรพื้นที่อะเรย์แบบ 2 มิติ มีขนาดเท่ากับ 5 แถวๆละ 2 สดมภ์เพื่อไว้รับจำนวนลูก้าตามเงื่อนไขที่ปรากฏ ส่วนข้อมูลที่จะพิมพ์ประกอบในตารางคือ กลุ่มในแต่ละช่วงอายุเราจะเก็บไว้ในอะเรย์มิติเดียวที่มีโครงสร้างของแต่ละเซลล์เป็น String นอกจากนี้จะมีตัวแปรแบบง่าย เช่น Age ไว้รับอายุของลูก้าแต่ละคน Sex ไว้รับเพศของลูก้า โดยที่จะให้รหัสเพื่อง่ายและประหยัดในการป้อนข้อมูลคือ FEMALE แทนที่ด้วยอักขระ '1' ส่วน MALE จะแทนด้วยรหัสอักขระ '2' สาเหตุที่ไม่ใช้ รหัสอักขระด้วย 'F' หรือ 'M' ก็เพราะว่าอาจจะเป็นไปได้ที่ผู้ป้อนข้อมูลป้อนรหัสเป็นอักขระ 'f' หรือ 'm' แทน ซึ่ง รหัส 'F' กับ 'f' เป็นรหัส Ascii Code ที่แตกต่างกัน เช่นเดียวกับ 'M' และ 'm' ก็เช่นเดียวกัน การป้อนข้อมูลสำหรับอักขระนั้น โอกาสที่จะป้อนผิดระหว่างตัวอักษรที่เป็น Uppercas ('A',...'Z') กับ Lowercase ('a',...'z') จะเป็นไปได้ก่อนจะสูง เราจึงควรเลือกรหัสที่เป็นอักขระที่เป็นตัวเลขแทนเพื่อจะลดความผิดพลาดดังกล่าว

จากแนวคิดในการออกแบบโครงสร้างข้อมูลและ ขั้นตอนการดำเนินงานที่กล่าวมานี้ เราจะนำมาออกแบบโปรแกรม โดยการใช้เครื่องมือคือผังโปรแกรม





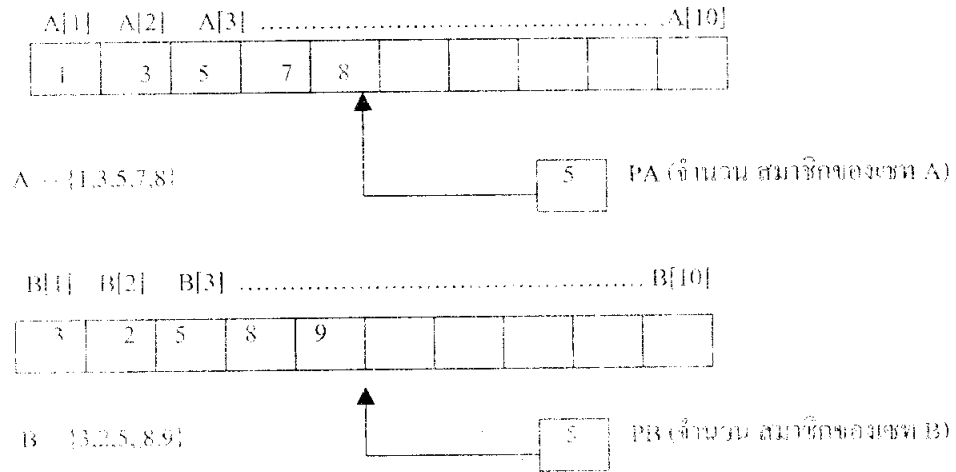


อะเรย์มิติเดียวและการนำไปใช้งานเชิงคณิตศาสตร์

อะเรย์มิติเดียวกับเซตในทางคณิตศาสตร์จะมีลักษณะคล้ายกันมาก ดังนั้นการนำเซตไปประยุกต์ใช้งานก็สามารถดำเนินงานในรูปแบบของอะเรย์มิติเดียว พิจารณาร่วมกับตารางดำเนินงานของเซตโดยการดำเนินงานด้วยโปรแกรมดังนี้คือ

ตัวอย่างที่ 7.3 กำหนดให้เซต $A = \{1,3,5,7,8\}$ และเซต $B = \{3,2,5,8,9\}$ จงลำดับขั้นตอนการทำงานในการหาเซต $D = A \cup B$ และเซต $C = A \cap B$ โดยที่เซต A และ B มีจำนวนสมาชิกในเซตสูงสุดไม่เกิน 10 สมาชิก

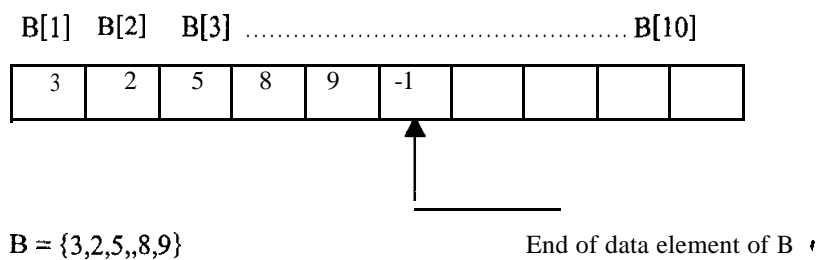
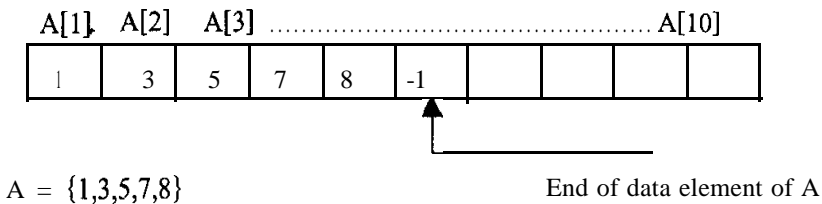
ทางเลือกที่ 1 ในการแก้ปัญหาปัญหานี้ก็คือการกำหนดอะเรย์ที่จะเก็บข้อมูลให้กับอะเรย์ A , B , C และ D



จำนวนสมาชิกในอะเรย์ A และ B นั้นจะตั้งข้อกำหนดสูงสุดไว้ว่าสูงสุดที่ได้คือ 10 สมาชิก โดยที่สมาชิกทุกตัวเป็นเลขจำนวนเต็มเท่านั้น ดังนั้นการกำหนดขนาดของอะเรย์ก็คือ A,B,C จะเป็นอะเรย์มิติเดียวที่มีขนาด เท่า 10 เซล ส่วนอะเรย์ D ก็จะเป็นอะเรย์ขนาด 20 เซล การที่กำหนด C ไว้แบบนี้ก็เพราะกรณีที่สูงสุดที่รับไปได้ก็คือสมาชิกทุกตัวใน A และ B นี้ มีสมาชิกที่ซ้ำกันทุกตัว ในขณะที่ D ไม่สูงสุดคือ 20 เซลถ้าหากว่าทั้ง A และ B ไม่มีสมาชิกตัวใดซ้ำกันเลย ประเด็นในกรณีที่สมาชิกใน A และ B มีไม่ครบ 10 สมาชิกเราก็จะตั้งตัวแปร PA ,PB, PC และ PD ไว้เป็นเพียงที่เก็บค่าของแต่ละอะเรย์ โดยสมมติไว้ ดังนั้น PA ก็จะทำหน้าที่เสมือน Pointer สำหรับ อะเรย์ A เช่นเดียวกับ อะเรย์ B,C,D ก็จะมี PB , PC , PD เป็น Pointer

ทางเลือกที่ 2 การกำหนดพื้นที่สำหรับเก็บข้อมูลที่เป็นสมาชิกของเซต A ,B,C และ D ก็จะดำเนินการลักษณะเช่นเดียวกับทางเลือกที่ 1 แต่จะไม่กำหนดพื้นที่ที่ทำหน้าที่เสมือน Pointer ให้กับอะเรย์ A ,B ,C และ D โดยที่เราจะใส่ข้อมูลพิเศษทำหน้าที่เป็นเสมือน End of data element บนดั่งนี้ ซึ่งอะเรย์เก็บข้อมูลพิเศษนี้ก็เพื่อให้พิจารณาจากข้อมูลของอะเรย์ A และ B ว่า สมาชิกแต่ละตัวไม่

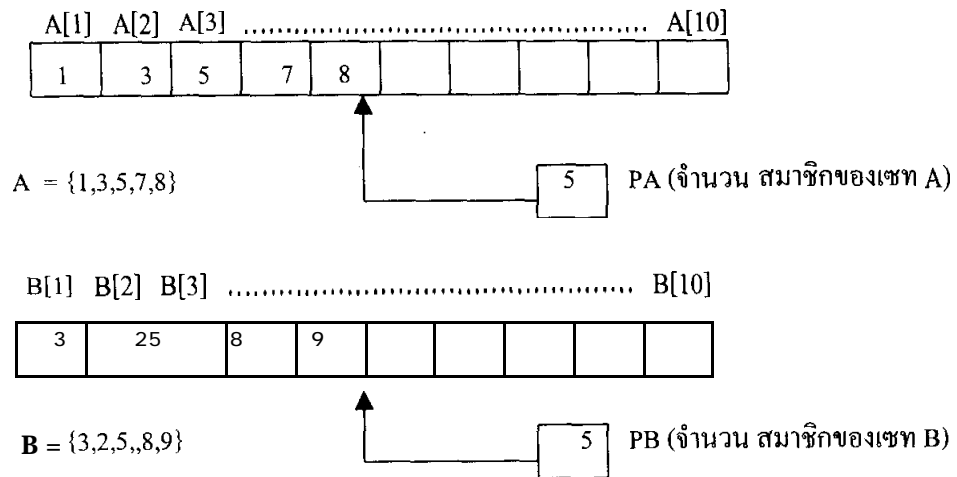
สามารถเป็นค่าใดก็ได้บ้าง เช่นถ้าโจทย์กำหนดว่าสมาชิกแต่ละตัวเป็นได้เฉพาะเลขจำนวนเต็มบวก เราสามารถกำหนดให้ข้อมูลพิเศษ คือ เลขติดลบ ได้ ดังนั้นในปัญหาโจทย์ข้อนี้เราอาจจะกำหนดให้เป็น -1 ทำหน้าที่เป็น End of data element ลักษณะการจัดการข้อมูลแบบนี้จะคล้ายกับภาษา C ซึ่งจะสร้าง End of String คือ '\0' ซึ่งการกำหนดรูปแบบนี้ส่งผลให้การ Pass data ให้กับ โปรแกรมย่อยๆ กระทำได้ง่ายเพราะไม่ต้องบอกความยาวของ String โดยโปรแกรมที่รับเอา String นั้นไปประมวลผล จะตรวจสอบ (Detect) ความยาวด้วย รหัสที่ตรงกับ '\0' ดังนั้นในงานของเราสำหรับการจัดเก็บตาม โครงสร้างนี้ก็คือใช้ -1 เป็นการตรวจสอบขนาดของข้อมูล รูปแบบในการจัดเก็บจะปรากฏตามรูป



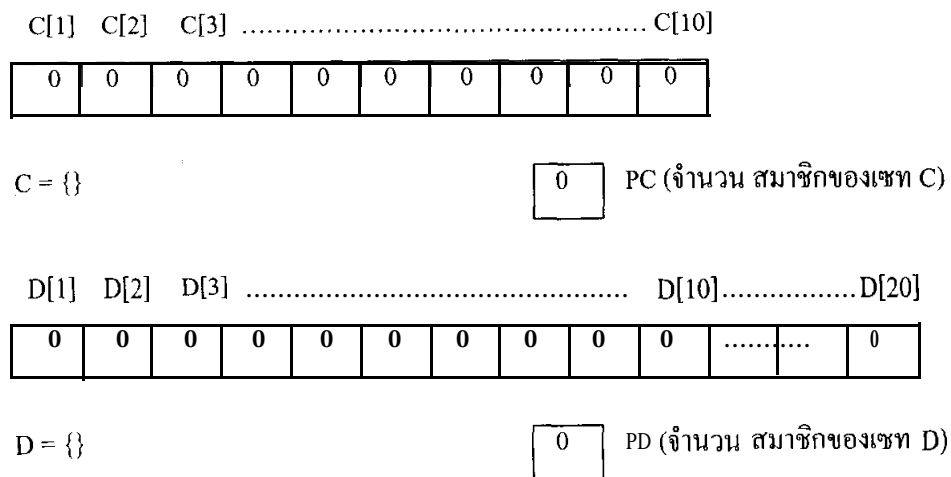
โครงสร้างในการจัดเก็บข้อมูลวิธีที่ 2 นี้ จะเห็นว่ามีข้อคืออยู่ส่วนน้อยคือสามารถประหยัดพื้นที่ไปได้ส่วนหนึ่งที่ทำหน้าที่เป็นเสมือน Pointer แต่ผลเสียที่ตามมาก็คือค่อนข้างจะยุ่งยากมากซึ่งพอสรุปได้ดังนี้คือ

- การออกแบบมีข้อจำกัดมาก ถ้าเลือกเลขใดๆหนึ่งค่ามาทำหน้าที่ End of data element และเกิดในสถานการณ์การใช้งานจริงมีข้อมูลที่มีลักษณะเป็น End of data element เกิดขึ้นมางานการประมวลผลนั้นจะล้มเหลว
- ในกรณีที่เรามีสมาชิกของอะเรย์ A หรือ B เป็นจำนวน 10 จำนวนเท่ากับขนาดของอะเรย์ที่จองไว้แล้ว ก็จะส่งผลให้ผู้ออกแบบต้องทำการกำหนดเงื่อนไขผนวกเข้าไปในการตรวจสอบด้วย ทำให้ยุ่งยากมากขึ้นซึ่งอาจจะทำให้การทำงานผิดพลาดในกรณีที่สมาชิกในอะเรย์ A หรือ B มี 10 สมาชิก (นั่นหมายความว่ากรณีนี้จะไม่มีค่า -1 ปรากฏในอะเรย์เลย)

บทสรุป จากทางเลือกในการจัดการกับข้อมูลของอะเรย์ A ,B,C และ D เราจะเลือกการสร้างอะเรย์ A ,B,C และ D โดยมีการกำหนดตัวแปรที่ทำหน้าที่เป็นเสมือน Pointer ของแต่ละอะเรย์ด้วย



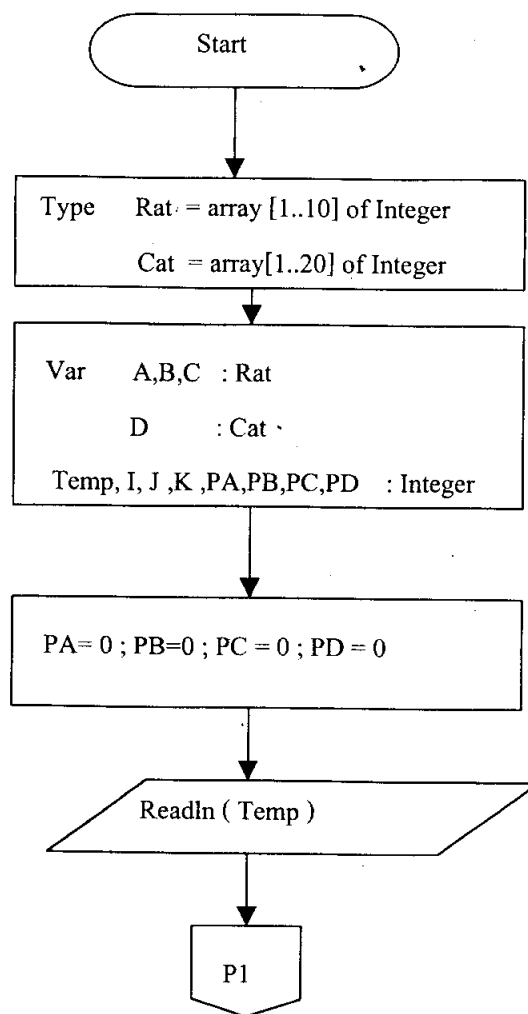
ลำดับต่อไปในการวิเคราะห์ของโปรแกรมก็คือสร้างขั้นตอนในการดำเนินงานเพื่อหาอะเรย์ $D = A \cup B$ และเซต $C = A \cap B$ ภายหลังเมื่อเรารับข้อมูลเข้าไปจัดเก็บในอะเรย์ A และ B แล้ว จะเห็นได้ว่า อะเรย์ C และ D จะมีข้อมูลปรากฏดังภาพนี้

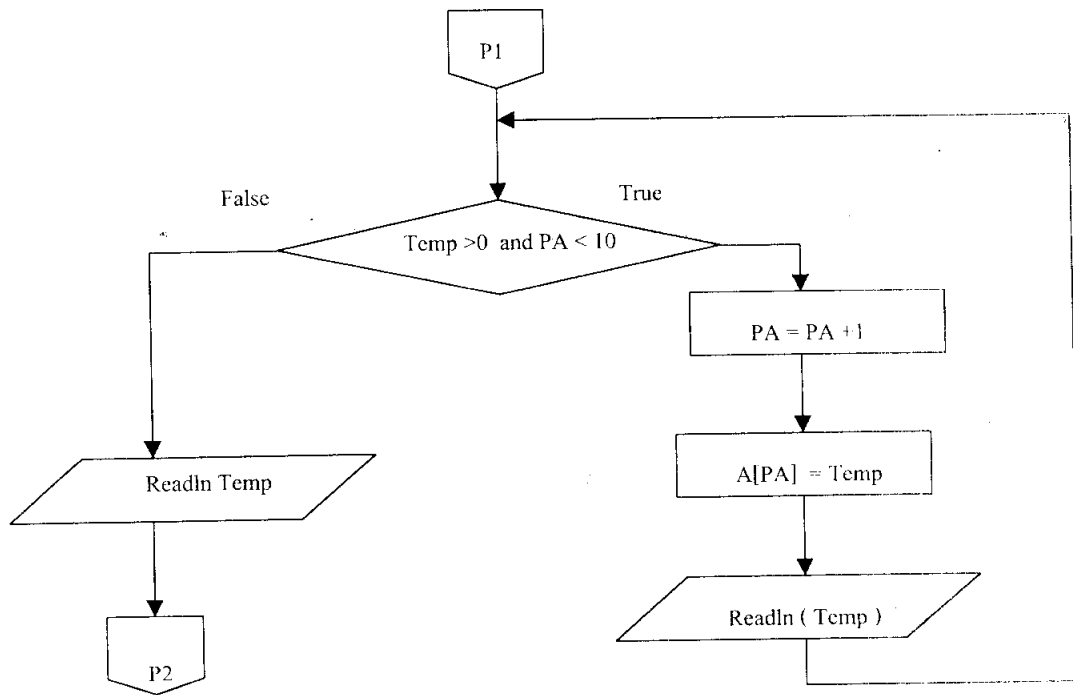


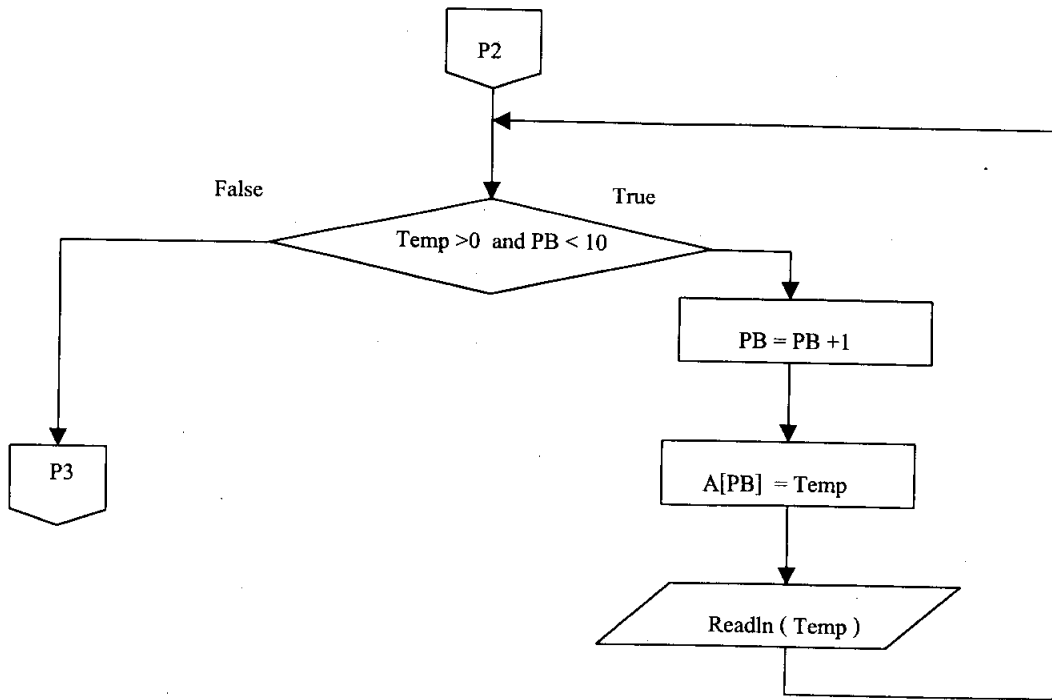
ขั้นตอนในการทำงานต่อไปก็คือให้นำสมาชิกแต่ละตัว (ทำกับทุกสมาชิก) ในที่นี้คือ $A[I]; I=1, \dots, PA$ โดยนำ $A[I]$ มาเปรียบเทียบกับสมาชิกในอะเรย์ B ว่ามีตัวใดใน $B[J]; J=1, \dots, PB$ ซ้ำกับสมาชิกของ A หรือไม่ ซึ่งการตรวจสอบกับ B นั้นจะประกอบด้วย 2 เงื่อนไข คือถ้าซ้ำออกจาก การตรวจสอบ แต่ถ้าไม่ซ้ำจะต้องตรวจสอบไปเรื่อย แต่ต้องไม่เกินจำนวนสมาชิกที่ B มีอยู่ในที่นี้

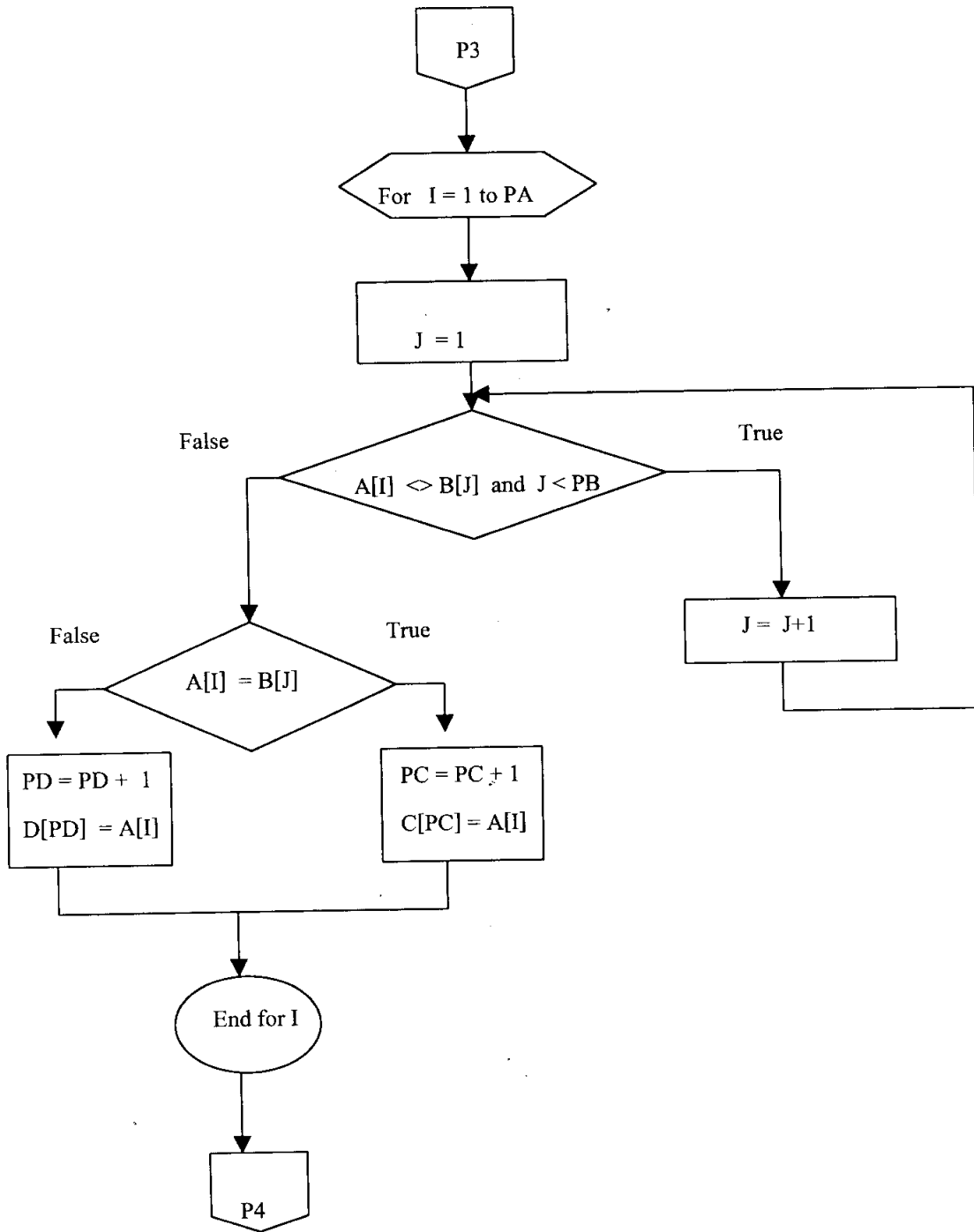
จำนวนสมาชิกของ B สามารถใช้ตัวแปร PB มาช่วยควบคุมได้ ผลจากการตรวจสอบนั้นจะได้สมาชิก
มาใส่ในอะเรย์ C ถ้ามีสมาชิกใดที่ปรากฏทั้งในอะเรย์ A และ B

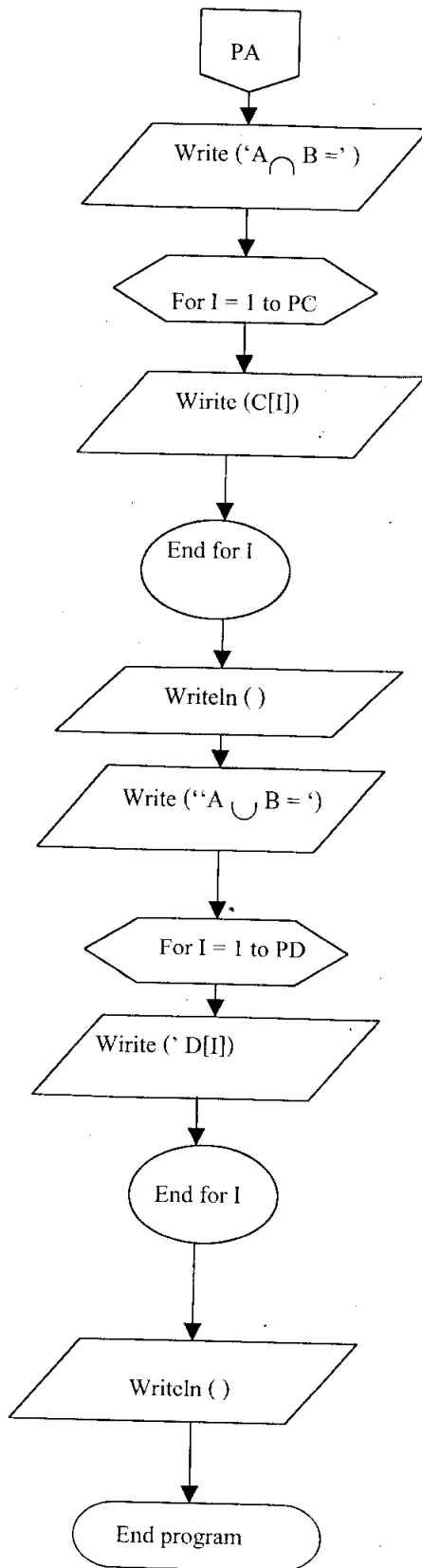
ผังโปรแกรมของการทำงานในการประมวลผลตามที่ต้องการภายใต้ข้อตกลงกำหนดให้ A ,B
เป็น อะเรย์ของ Integer มีไม่เกิน 10 สมาชิก



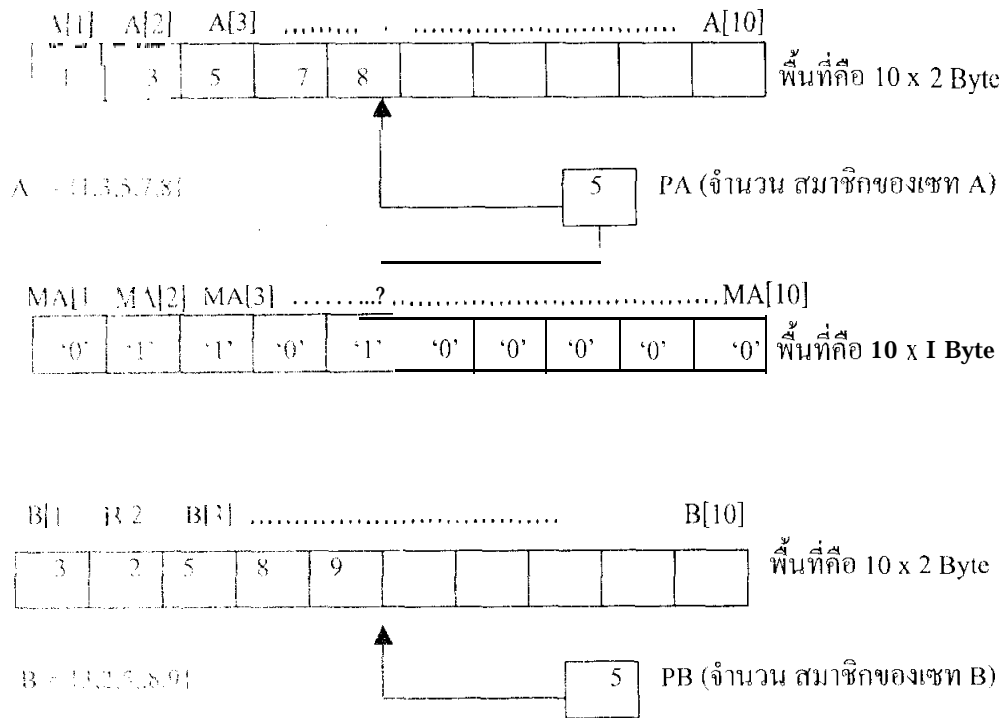




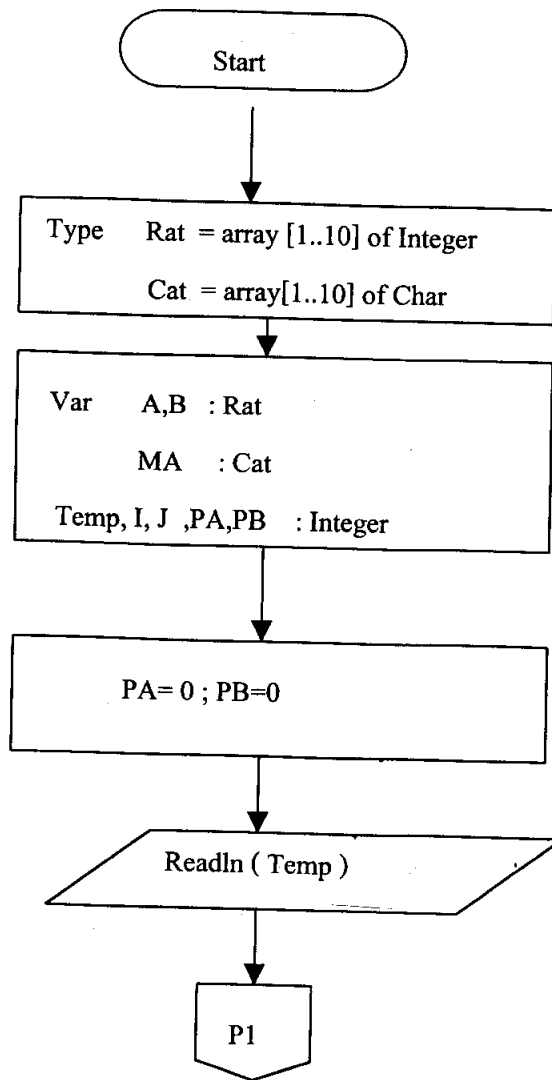


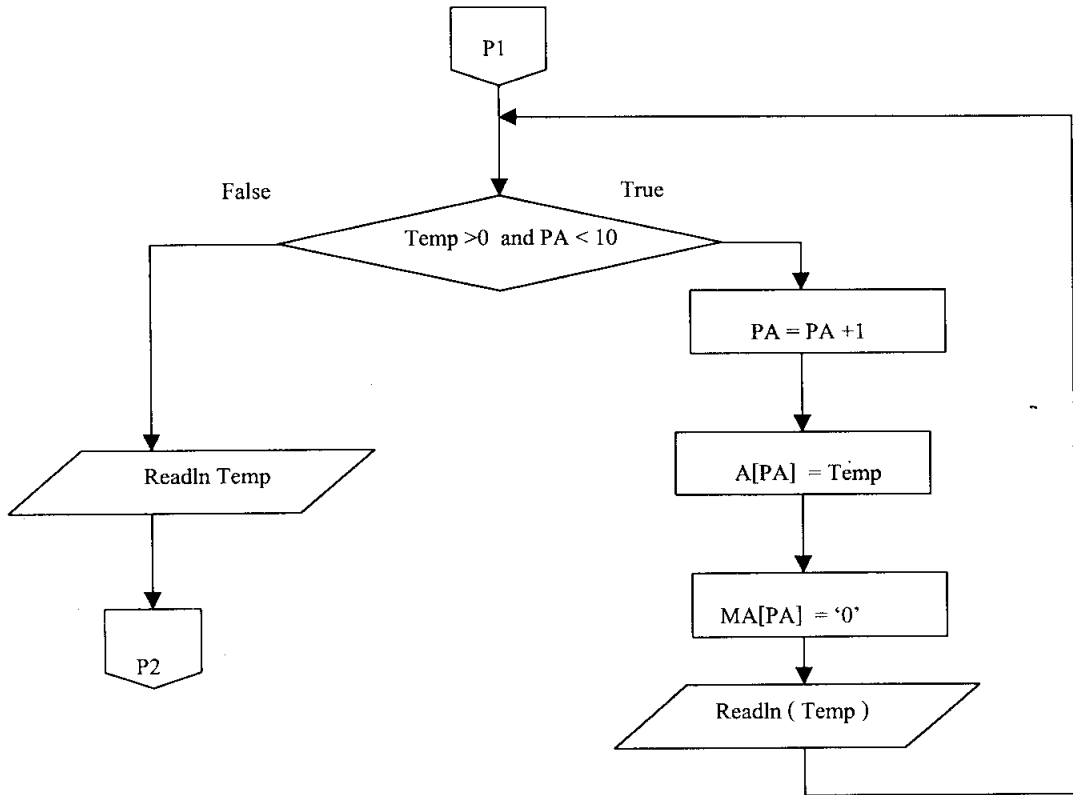


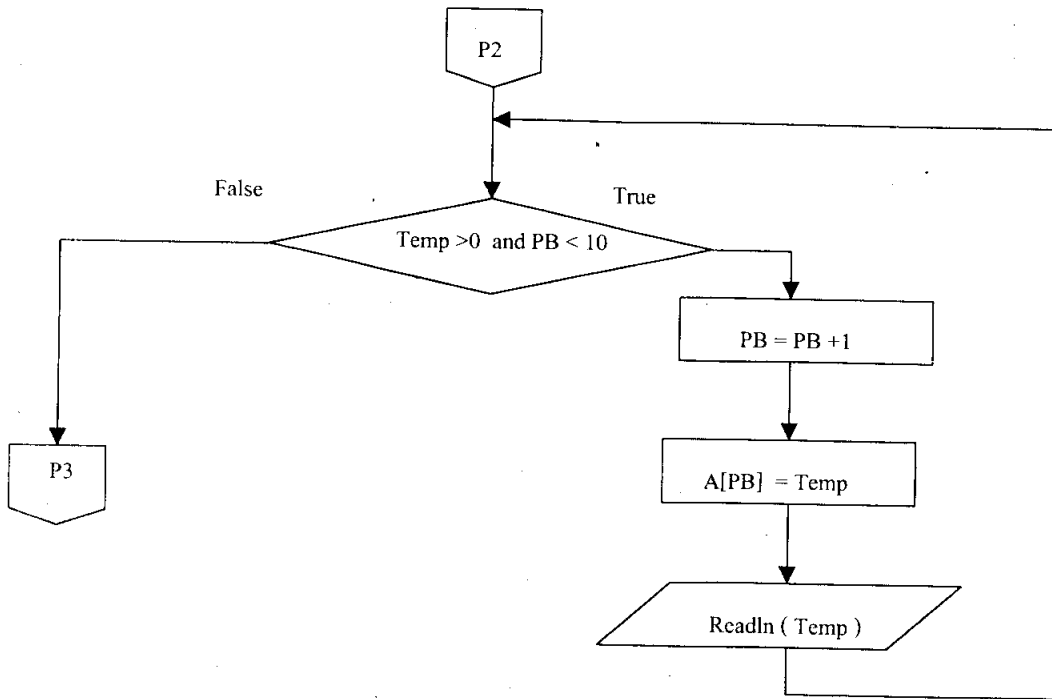
ทางเลือกที่ 3 จะดำเนินการคล้ายกับทางเลือกที่ 2 แต่จะแตกต่างกันตรงที่ว่า แทนที่จะกำหนดให้มี
 อะเรย์ C และ D เพื่อเก็บสมาชิกของ $D = A \cup B$ และเซต $C = A \cap B$ เราจะสร้างอะเรย์ขึ้นมา
 เพียงอะเรย์เดียวเพื่อทำหน้าที่ในการเก็บข้อมูลซึ่งแสดงความสัมพันธ์ระหว่างอะเรย์ A และ B ในที่นี้
 จะเรียกว่า อะเรย์ MA โดยที่ MA จะเป็น Array [1..10] of char ซึ่งจะทำหน้าที่ในการเก็บข้อมูลที่เป็นมี
 มูลค่าเป็น '0' หรือ '1' โดยที่ถ้าข้อมูลมีค่าเป็น '1' ก็หมายถึงว่าตำแหน่งของ A[I] ; I = 1,..., PA นั้นๆ
 จะเป็นข้อมูลที่ซ้ำกับสมาชิกในอะเรย์ B (ในตำแหน่งใดบนอะเรย์ของ B ก็ได้) ให้อูจากรูปผลลัพธ์ที่
 เกิดจากกรณีนี้มาตามความต้องการที่กำหนด

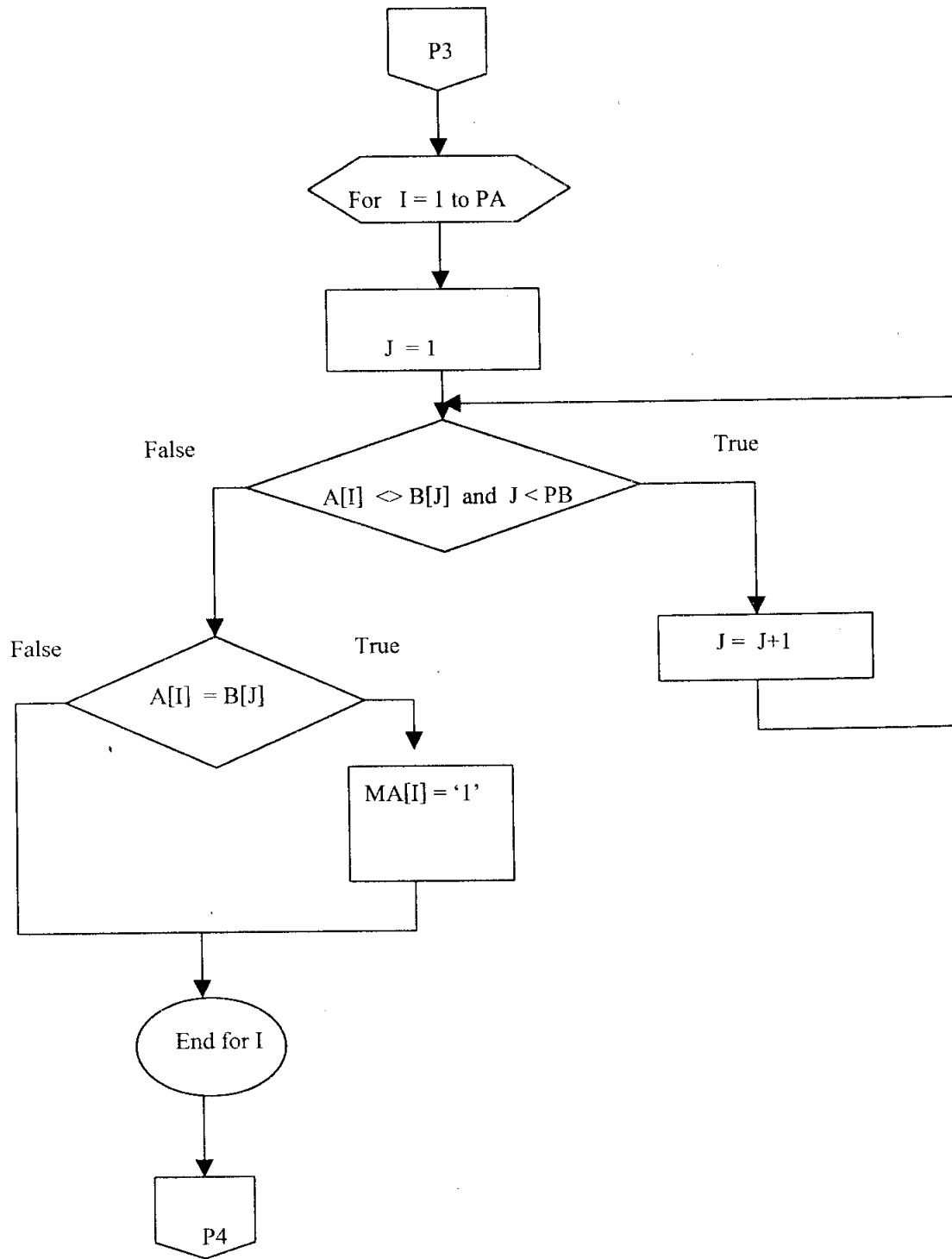


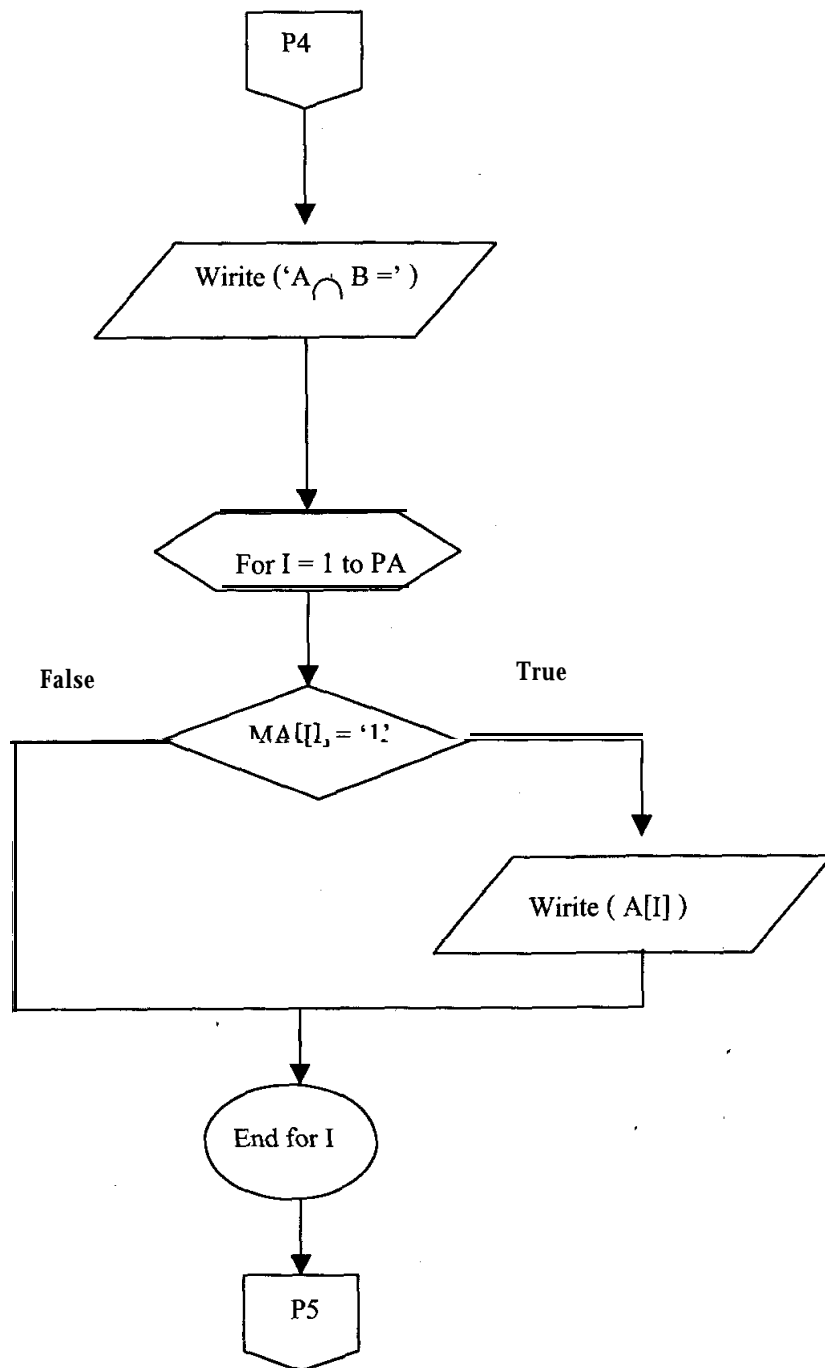
ดังนั้นถ้าเราดำเนินการ ได้ตั้งรูปที่ปรากฏของอะเรย์ MA ก็จะสามารถหาสมาชิกของ ของ
 $D = A \cup B$ และเซต $C = A \cap B$ ได้โดยไม่ต้องสร้างอะเรย์ C, D ให้เปลืองพื้นที่ดังการดำเนิน
 งานตามทางเลือกทางที่ 2 เพราะอะเรย์ MA จะทดแทนอะเรย์ C, D ได้ครบถ้วนตามความต้องการ ดัง
 นั้นถ้าเราเลือกทางนี้การดำเนินงานจะปรากฏตามผังโปรแกรมต่อไปนี้

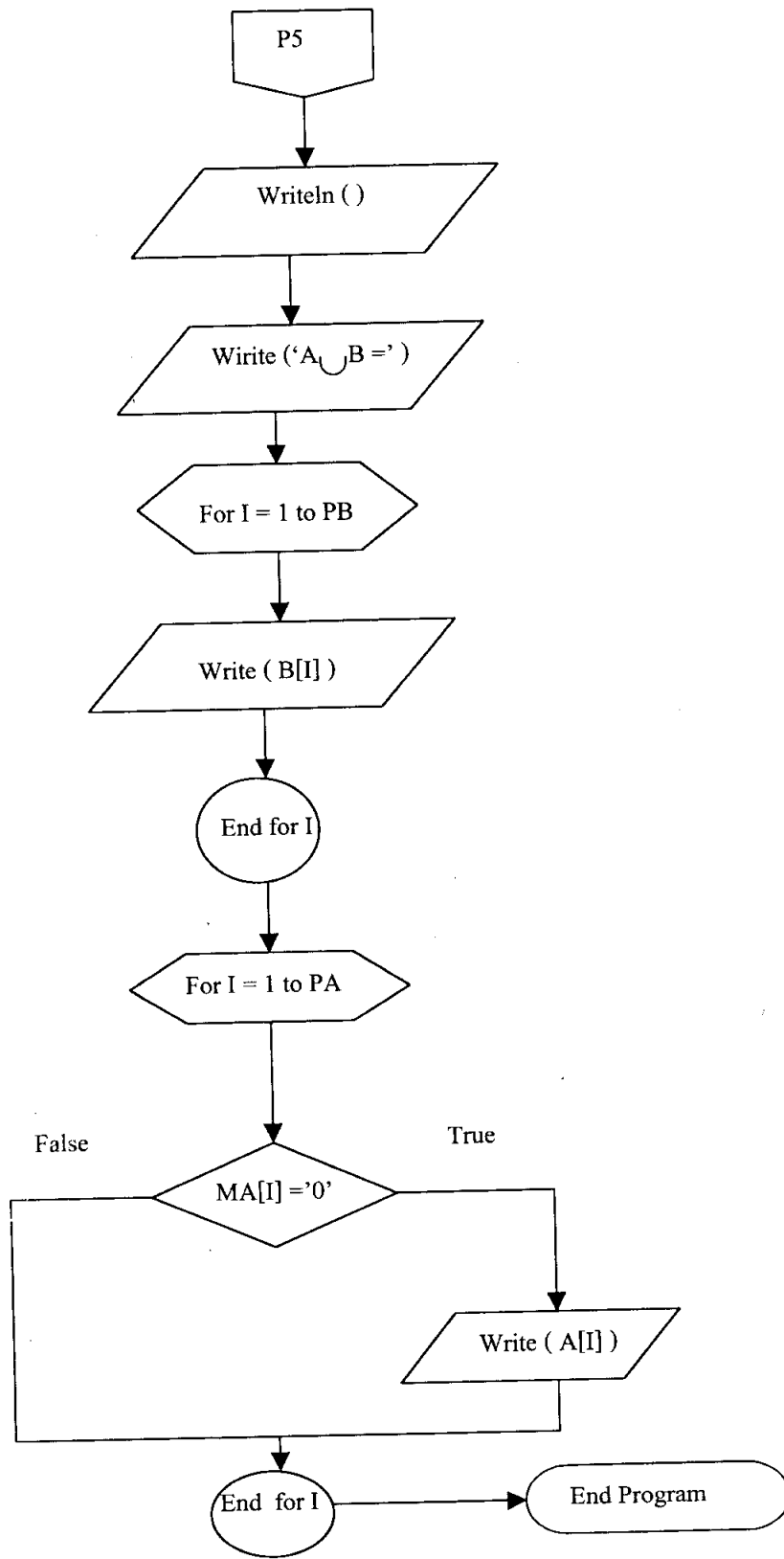












ตัวอย่างที่ 7.4 กำหนดให้คะแนนสอบของวิชาหนึ่งที่มีคะแนนต่ำสุดคือ 0 และคะแนนสูงสุดคือ 100 มีนักศึกษาเข้าสอบจำนวนหนึ่ง ภายหลังจากสอบคะแนนของนักศึกษาแต่ละคนที่เข้าสอบจะป้อนเข้าเครื่องคอมพิวเตอร์ ให้ออกแบบโปรแกรมในการสร้างสารสนเทศเพื่อนำไปตัดสินใจในการตัดสินผลการสอบ โดยที่สารสนเทศที่อาจารย์ต้องการจะประกอบด้วย คะแนนเฉลี่ย คะแนนที่ต่ำที่สุด คะแนนที่สูงที่สุด คะแนนมัธยฐาน และคะแนนฐานนิยม

แนวทางวิเคราะห์ความต้องการและข้อจำกัดของปัญหาจะมีทางเลือกในการดำเนินงานได้หลายๆแบบ ทางเลือกที่จะดำเนินการต่อไปนี้จะยังเป็นอีกหนึ่งทางเลือก ให้นักศึกษาอ่านและวิเคราะห์ถึงข้อดีข้อเสีย แล้วนำไปออกแบบโดยทางเลือกอื่นที่ดีกว่านี้ โดยการพิจารณาจากคุณสมบัติของโปรแกรมที่ดีที่อธิบายมาแล้วในบทที่ 4

การกำหนดโครงสร้างข้อมูลจะสร้างอะเรย์มิติเดียวที่ประกอบด้วยเซลล์ที่เป็น Integer โดยเริ่มจากลำดับที่ 0 ... 100 เจตนาจะใช้ ดัชนีของอะเรย์ หมายถึงคะแนนสอบที่เป็นไปได้ทั้งหมด ตั้งแต่คะแนน 0 ... 100 ส่วนข้อมูลในแต่ละเซลล์จะเก็บความถี่ของคนที่สอบได้คะแนนต่างๆกันไป ให้อุภาพประกอบความต้องการต่อไปนี้

A[0]	A[1]	A[2]	...	—	.A[100]
0	3	1	3	0	1

พื้นที่คือ 101 x 2 Bytes

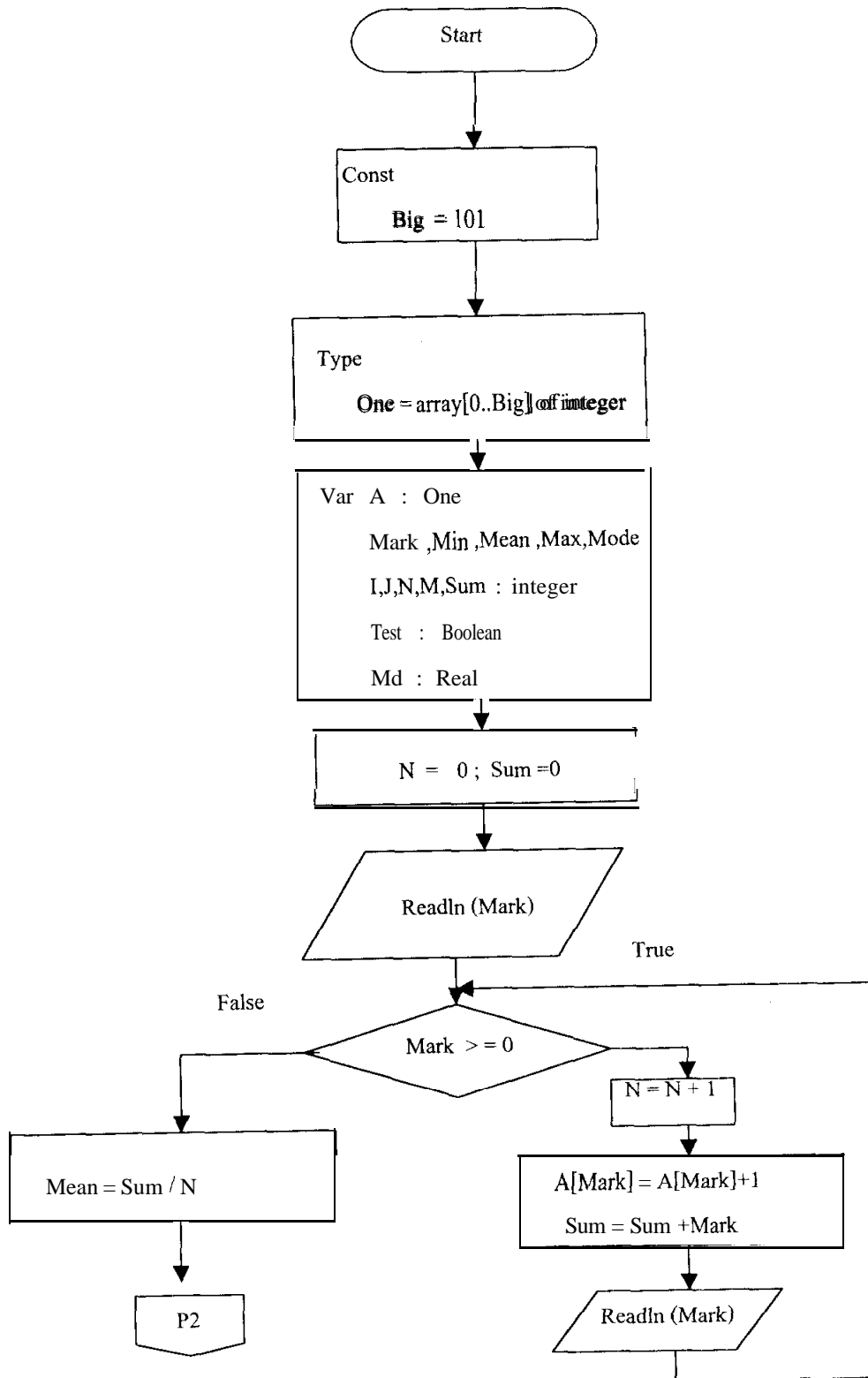
อะเรย์ A จะมีความหมายดังนี้คือ A[I] ; I = 0,1,...,100 หมายถึงเซลล์ที่จะเก็บข้อมูลที่เป็นความถี่ของคะแนน ณ.คะแนนที่ต่างๆกัน จากตัวอย่าง A[0] = 0 จะมีความหมายว่านักเรียนที่ได้คะแนน 0 จะมีจำนวน 0 คน ส่วน A[1] จะหมายถึง คะแนน 1 คะแนน จะมีเด็กนักเรียนสอบได้อยู่ 3 คน ดังนี้ เป็นต้น

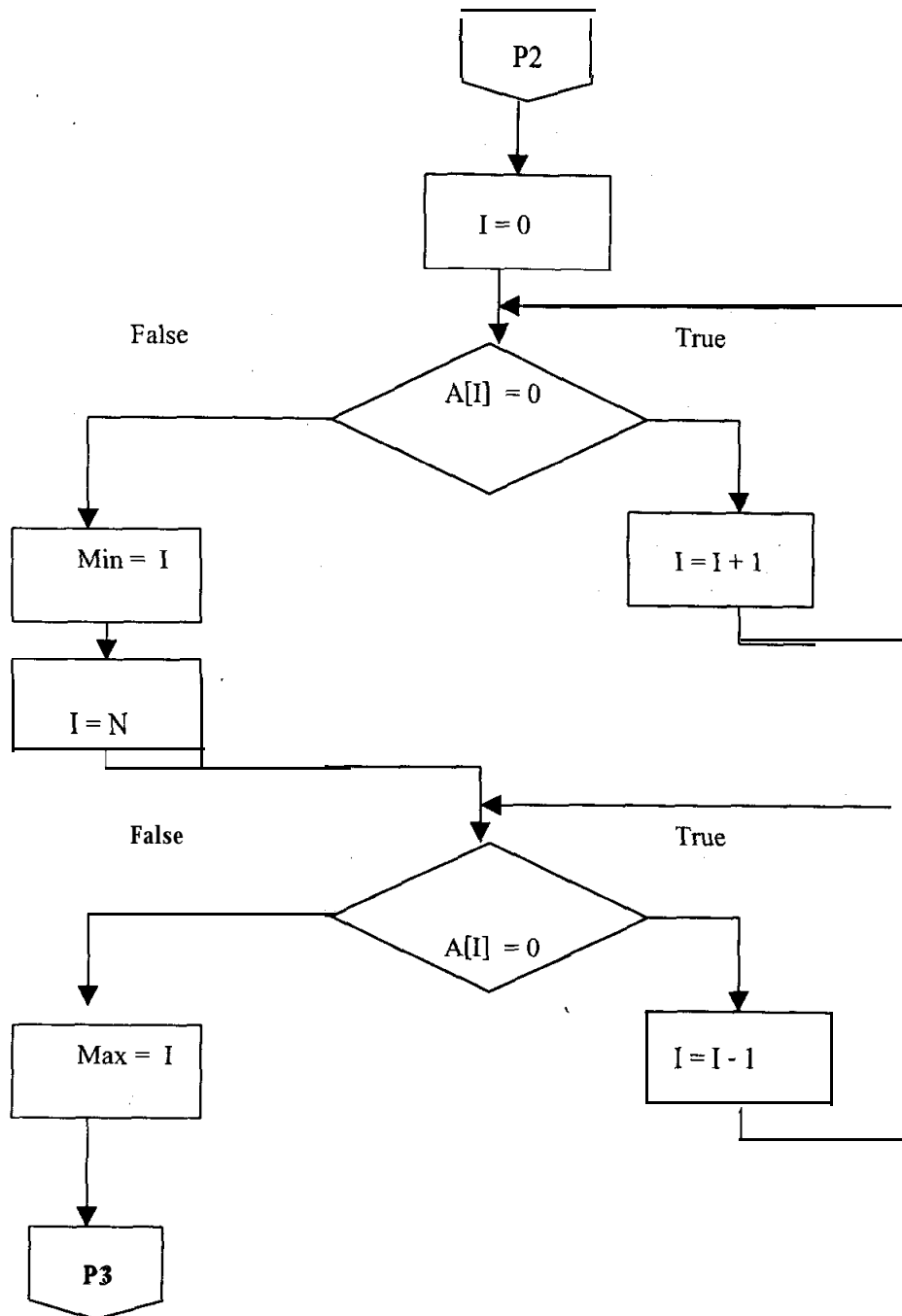
ลำดับขั้นตอนการทำงานเพื่อหาค่าสารสนเทศต่อไปนี้จะดำเนินการดังต่อไปนี้คือ

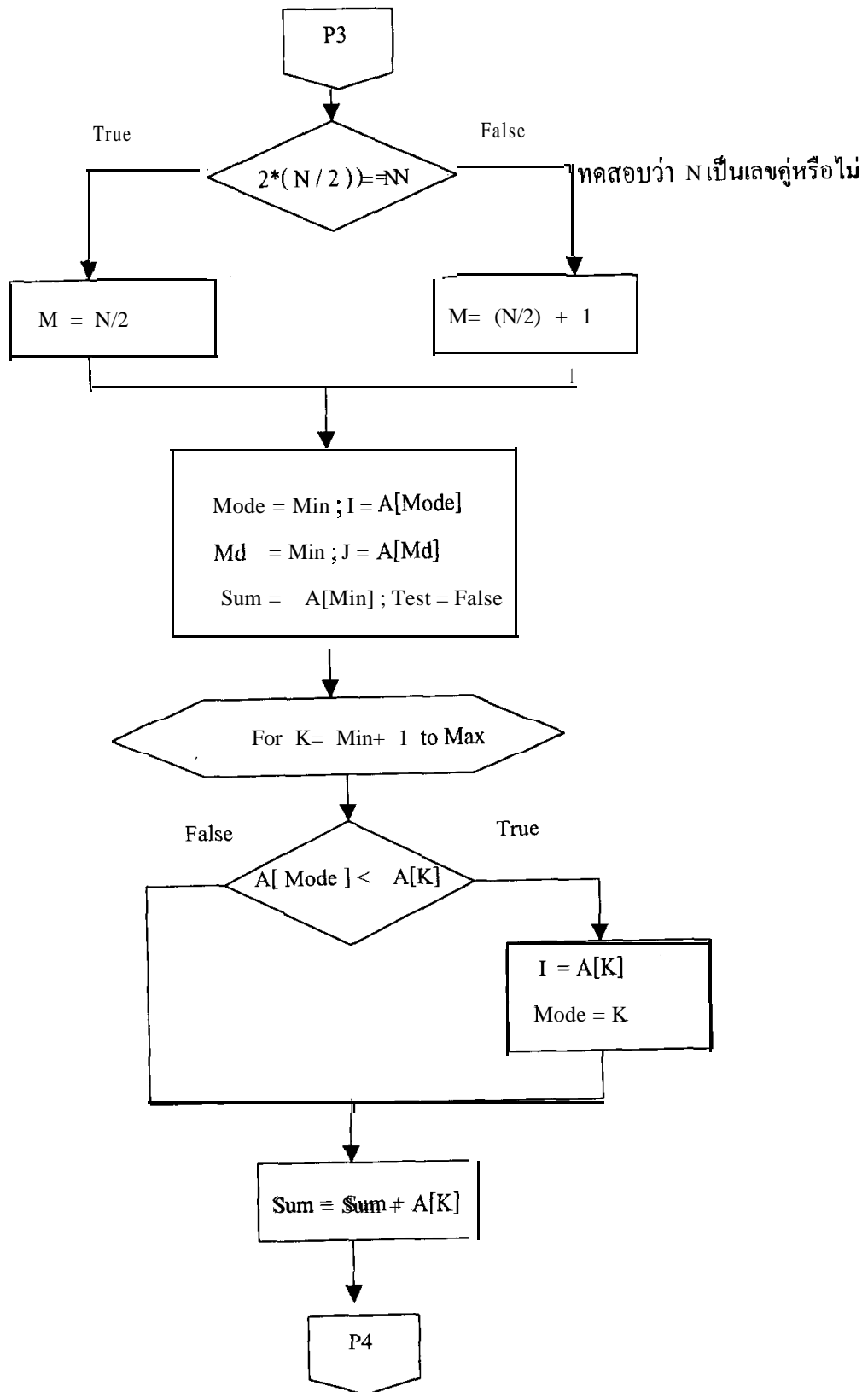
- ค่าเฉลี่ย ดำเนินการโดยการนำข้อมูลคือคะแนนที่ป้อนเข้าไปเก็บสะสมในตัวแปร SUM
- ค่าคะแนนที่ต่ำที่สุดและคะแนนสูงสุดจะดำเนินการโดยค้นหาจากอะเรย์ A
- ค่าคะแนนมัธยฐาน จะดำเนินการโดยการ คำนวณหาจำนวนครึ่งหนึ่งของจำนวนร้อยละของนักเรียนทั้งหมด (คะแนนมัธยฐานหมายถึง คะแนนที่ Percent 50)
- ค่าคะแนนฐานนิยมหมายถึง คะแนนที่จำนวนนักเรียนสอบได้มากที่สุด

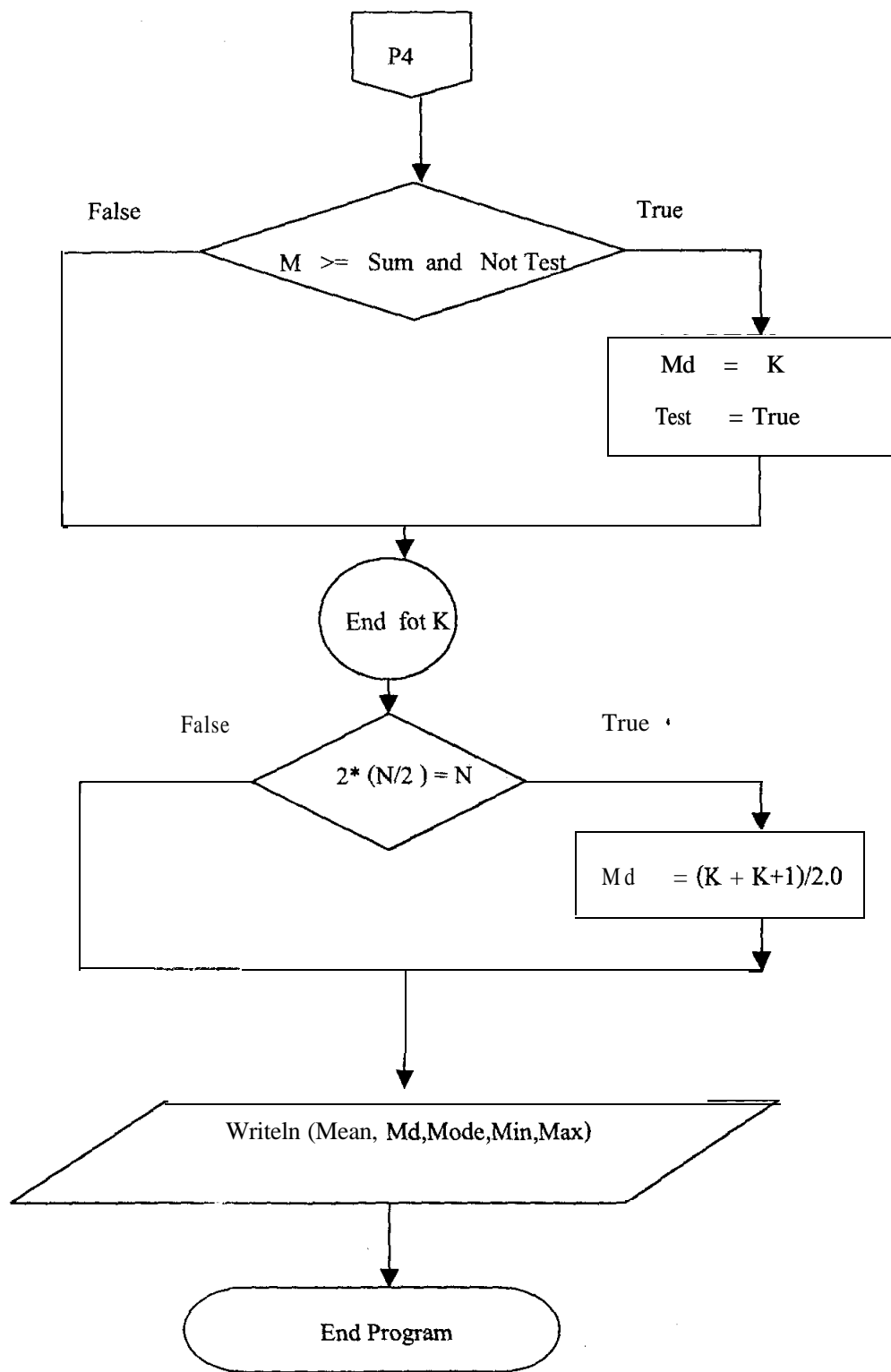
ตัวแปรเพิ่มเติมที่จะดำเนินงานคือ Mean , Md , Mode ,Min ,Max สมมุติให้เป็น Integer

ผังโปรแกรมในการดำเนินงานของปัญหาจากตัวอย่างที่ 2 จะเป็นดังนี้









จากผังโปรแกรมที่เขียนมาเนี่ยยังมีจุดบกพร่องอยู่บางจุดที่ผู้เขียน โปรแกรมจะต้องปรับแก้ใน รายละเอียด เช่นในกรณีของค่ามัธยฐานนั้นจะได้ค่าที่ค่อนข้างจะหยาบซึ่งจำเป็นจะต้องใช้สูตรในการ ปรับ เช่นเดียวกับค่าของ ฐานนิยม (Mode) แต่ถ้าจะไม่สนใจรายละเอียดมากนักค่าสถิติที่ได้จากผัง โปรแกรมนี้ก็สามารถจะใช้เป็นสารสนเทศในการตัดสินใจได้สำหรับการตัดเกรดผลการสอบต่อไป

<

คำถามท้ายบท

- กำหนดให้เลขจำนวนหนึ่งที่ได้รับจากแปดพิมพ์เป็นเลขจำนวนเต็มบวกที่มีค่าไม่เกิน 500 จงออกแบบโปรแกรม เพื่อดำเนินการดังนี้คือ ภายหลังที่รับข้อมูลมา ให้แสดงผลว่าเลขจำนวนนั้น มีตัวประกอบร่วมอะไรบ้าง ตัวอย่างต่อไปนี้จะแสดงผลที่ได้จากการทำงานคือ
กรณีที่ 1 $24 = 2 * 2 * 2 * 3$
กรณีที่ 2 $15 = 3 * 5$
กรณีที่ 3 $7 = \text{'Prime Number'}$
- กำหนดให้ Apartment แห่งหนึ่งมีห้องเช่าอยู่ 20 ห้อง เจ้าของ Apartment จะเก็บข้อมูลของผู้เช่าประกอบด้วย ชื่อ นามสกุล และจำนวนการใช้โทรศัพท์ในแต่ละเดือน โดยที่การใช้โทรศัพท์ของผู้ใช้แต่ละห้องจะแบ่งเป็น 2 ประเภทคือการใช้โทรศัพท์ภายใน Apartment จะไม่เสียเงินค่าบริการ ส่วนการใช้โทรศัพท์นอก Apartment จะต้องเสียเงินครั้งละ 4 บาท รายละเอียดนอกจากข้อตกลงนี้ให้กำหนดเองตามสภาพความเป็นจริง สิ่งที่ต้องการของระบบงานนี้จะประกอบด้วย
 - ให้แสดงรายงานค่าใช้จ่ายโทรศัพท์ของผู้ใช้แต่ละคนตอนสิ้นเดือน
 - ให้รายงานว่าผู้ใช้โทรศัพท์รายใดใช้โทรศัพท์ ภายในสูงสุด และผู้ใช้รายใดใช้โทรศัพท์ภายนอกสูงสุด
 - ให้รายงานว่าผู้ใช้โทรศัพท์รายใดใช้โทรศัพท์ ภายนอกสูงกว่า 50 ครั้งขึ้นไป
- สำนักงานสถิติแห่งชาติใช้วงล้อหมุนออกรางวัลสลากกินแบ่ง โคนบนว่าหลักสี่ที่หมุนนั้นเป็นหลักที่ไม่เที่ยงตรง จึงทำการทดสอบการหมุนของหลักสี่ที่หมุน ซึ่งมีทั้งหมด 7 หลัก แต่ละหลักจะหมุน 20 ครั้ง ให้นักศึกษาสรุปผลจากการทดลอง เพื่อให้สารสนเทศกับผู้อำนวยความสะดวกในการนำไปตัดสินใจว่า วงล้อหลักใดที่ไม่เที่ยงตรงจริงจะได้ยกเลิกการใช้งานและหาของใหม่มาแทน
- การสอบครั้งหนึ่งเป็นข้อสอบปรนัย 100 ข้อ โดยที่ถ้าใครตอบผิดจะติดลบเพื่อป้องกันการเดา ภายหลังจากการสอบเสร็จสิ้นลง นักศึกษาแต่ละคนจะมีคะแนนรวม บางคนได้ 0 บางคนสอบได้คะแนนติดลบ เช่น -6 คะแนน เป็นต้น จากข้อกำหนดที่กล่าวมานี้ ให้ออกแบบโครงสร้างข้อมูล และขั้นตอนการดำเนินงาน เพื่อสร้างสารสนเทศดังนี้คือ คะแนนเฉลี่ย คะแนนสูงสุด คะแนนต่ำสุด คะแนนมัธยฐาน
- บริษัทแห่งหนึ่งมีการจ้างคนงานเป็นแรงงานรายวัน โดยที่ค่าจ้างคนงานแต่ละคนนั้นจะเท่ากับ 147 บาท การจ่ายค่าจ้างจะจ่ายทุกๆ 5 วัน โดยที่ พนักงานแต่ละคนจะมีข้อมูลเพื่อ

เขียนการทำงานอันประกอบด้วย ชื่อของพนักงาน จำนวนวันที่ทำงาน และเงินรวมที่จะจ่ายในสัปดาห์นั้น ข้อมูลในการจ่ายเงินจะต้องมีทั้งยอดรวม และรายละเอียดว่าจะจ่ายธนบัตรแต่ละประเภทกี่ใบ ตัวอย่างเช่น

นาย ก ทำงาน 4 วัน ได้เงินค่าจ้างรวมเท่ากับ 688 บาท	
ธนบัตรใบละหนึ่งพันบาท จำนวน	0 ใบ
ธนบัตรใบละห้าร้อยบาท จำนวน	1 ใบ
ธนบัตรใบละหนึ่งร้อยบาท จำนวน	1 ใบ
ธนบัตรใบละห้าสิบบาท จำนวน	1 ใบ
ธนบัตรใบละยี่สิบบาท จำนวน	1 ใบ
ธนบัตรใบละสิบบาท จำนวน	1 ใบ
เหรียญใบละห้าบาท จำนวน	1 ใบ
เหรียญใบละหนึ่งบาท จำนวน	3 ใบ

จงออกแบบโครงสร้างข้อมูลและวิธีการดำเนินงานให้ได้ผลตามที่ต้องการตามตัวอย่างที่แสดงมานี้

- กำหนดให้ A เป็นเซตของเลขจำนวนเต็มที่ได้รับจากเป็นพิมพ์ โดยที่สมาชิกที่สูงสุดที่เป็นไปได้คือ 20 สมาชิก ให้ทำการสร้างอะเรย์ที่เป็นการบ่งบอกคุณลักษณะของสมาชิกแต่ละตัวว่าเป็นเลขคู่หรือเลขคี่ โดยการกำหนดว่าถ้าตำแหน่งใดเป็นเลขคู่ให้เก็บสัญญาณ True ไว้ที่ อะเรย์ TA ดูจากตัวอย่างต่อไปนี้เพื่อประกอบการออกแบบโปรแกรมตามความต้องการดังกล่าว

ตัวอย่าง เช่น เซต $A = \{2,4,6,7,9,10,45\}$ ดังนั้น

$MA = \{\text{True}, \text{True}, \text{True}, \text{False}, \text{False}, \text{True}, \text{False}\}$

- ให้ออกแบบโปรแกรมเพื่อทำการแสดงผลให้เหมือนเข็มนาฬิกา แสดงผลเฉพาะเข็มสั้นที่ชี้ จำนวนชั่วโมง (กำหนดว่านาฬิกาแสดงผลได้ตั้งแต่ 1,..., 12 เท่านั้น) โดยที่ให้นาฬิกาเรือนนี้เดินเพียง 2 วัน
- จงออกแบบโปรแกรมในการตรวจสอบหาค่า Pime Number ที่มีค่าไม่เกิน 500