

เอกสารประกอบการสอน

การเขียนโปรแกรมเบื้องต้น

ตอน

ฝึกงาน



005.1
พ 254๑

หริตา ชูติหารา

ภาควิชาคณิตศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ กิตยาเขตประจักษ์นิต

ผังงาน

- 9 ต.ย. 2535



๑ อัดสำเนา

ภาควิชาคณิตศาสตร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ วิทยาเขตประสานมิตร

สารบัญ

บทที่ 0	ผังงาน	หน้า
0.1	ความหมายประเภท และสัญลักษณ์	2
0.1.1	ผังงานระบบ	4
0.1.2	ผังงานชุดคำสั่ง	5
0.2	ตัวแปร และประเภทของตัวแปร	6
0.2.1	ตัวแปรจำนวน	7
0.2.2	ตัวแปรอักขระ	8
0.2.3	ตัวแปรตรรก	8
0.3	การให้ค่าตัวแปร	8
0.3.1	การให้ค่าตัวแปรโดยการรับจากสื่อ	8
0.3.2	การให้ค่าตัวแปรโดยใช้คำสั่งกำหนดค่า	9
0.4	โครงสร้างหลักสำหรับการประมวลผล	14
0.4.1	โครงสร้างเรียงลำดับ	14
0.4.2	โครงสร้างการตัดสินใจ	15
0.4.3	โครงสร้างการทำงานซ้ำในวงจร	20
0.5	การออกแบบจากบนลงล่าง	25

บทที่ ๐
ผังงาน
(Flow chart)

เมื่อผู้ใช้เครื่องคอมพิวเตอร์ ต้องการจะให้คอมพิวเตอร์ทำงานอย่างใดอย่างหนึ่งตามจุดประสงค์ของตัวเอง ต้องเขียนโปรแกรมขึ้นมาเพื่อให้เครื่องทำงานตามความต้องการ เนื่องจากเครื่องคอมพิวเตอร์ไม่สามารถคิดเองได้ เพราะฉะนั้นผู้เขียนโปรแกรมจะต้องวางแผนการเขียนโปรแกรมเพื่อใช้แก้ปัญหาและป้องกันไม่ให้เกิดข้อผิดพลาดในการใช้งาน

ขั้นตอนการพัฒนาโปรแกรม

- การวิเคราะห์ปัญหา
- การพัฒนาวิธีในการแก้ปัญหา
- การเขียนผังงาน
- การเลือกภาษาที่ใช้ และเขียนคำสั่งตามผังโปรแกรม
- การทดสอบโปรแกรม
- การแก้ไขโปรแกรม
- การทำเอกสารประกอบ

การวิเคราะห์ปัญหา ก่อนอื่นต้องวิเคราะห์ปัญหาเพื่อจะนำไปใช้กับคอมพิวเตอร์ก่อน ซึ่งอันดับแรกต้องเข้าใจปัญหาก่อนและต้องรู้ว่าผู้ใช้ต้องการอะไร ความมีอะไรเป็นข้อมูลเข้า

การพัฒนาวิธีในการแก้ปัญหา เมื่อทราบว่าปัญหาคืออะไรแล้ว ต่อไปก็ต้องหาอัลกอริทึม ซึ่งคือลำดับขั้นตอนที่จะใช้แก้ปัญหานั้น ๆ ทางคอมพิวเตอร์

การเขียนผังงาน เมื่อได้การแจกแจงแต่ละขั้นตอนของการแก้ปัญหาแล้วก็นำมาเขียนผังงานเพื่อให้ผู้เขียนโปรแกรมหรือผู้ใช้โปรแกรมได้เข้าใจถึงขั้นตอนในการทำงานของโปรแกรมได้

การเลือกภาษาที่ใช้และเขียนคำสั่งตามผังงาน หลังจากนั้นก็เลือกภาษาที่จะใช้เขียนโปรแกรม ซึ่งควรจะเป็นภาษาที่เหมาะสมกับการแก้ปัญหานั้น ๆ แล้วเขียนโปรแกรมตามผังงาน

การทดสอบโปรแกรม ทำการสอบโปรแกรมว่าโปรแกรมทำงานได้สมบูรณ์หรือไม่ เมื่อใส่ข้อมูลเข้าไป จะได้ผลลัพธ์ตามที่ผู้ใช้ต้องการหรือไม่

การแก้ไขโปรแกรม ถ้าข้อมูลออกหรือโปรแกรมยังมีข้อผิดพลาดอยู่โปรแกรมเมอร์ก็ต้องแก้ไขโปรแกรม จนกว่าจะได้ผลลัพธ์สมบูรณ์

การทำเอกสารประกอบ เมื่อโปรแกรมเสร็จสมบูรณ์สามารถแก้ปัญหานั้น ๆ ได้ โปรแกรมเมอร์ก็จัดทำเอกสารประกอบการใช้โปรแกรมนี้ เช่น มีโปรแกรมข้อมูลเข้าเป็นอย่างไร ข้อมูลออกที่จะได้คืออะไร การเรียกใช้โปรแกรมก็ทำอย่างไร

เมื่อโปรแกรมเมอร์ต้องการจะพัฒนาโปรแกรมทางคอมพิวเตอร์ มีเครื่องมือและเทคนิค

ต่าง ๆ ที่จะช่วยบอกขั้นตอนการทำงานให้ดูเข้าใจได้ง่าย เช่น ผังงาน pseudocode, decision table ฯลฯ และในบทนี้เราจะกล่าวถึงผังงานซึ่งมีรายละเอียดดังนี้

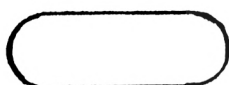
0.1 ความหมาย ประเภท และสัญลักษณ์

ผังงานเป็นแผนผังที่อธิบายขั้นตอนการประมวลผลตั้งแต่เริ่มการทำงานจนกระทั่งสิ้นสุดการทำงานนั้น ๆ โดยเขียนอธิบายรายละเอียดได้ตามที่ต้องการ

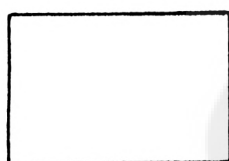
สัญลักษณ์ที่ใช้ในการเขียนผังงานมีดังนี้

สัญลักษณ์

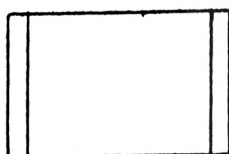
ความหมาย



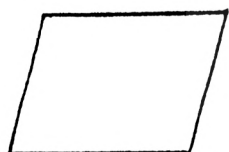
เริ่มต้น/หยุด (start/stop)



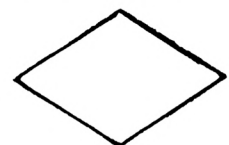
ประมวลผล (process)



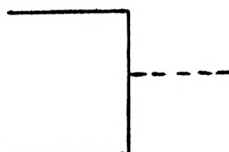
ชุดคำสั่งย่อย, ชุดคำสั่งประจำย่อย (subprogram, subroutine)



ข้อมูลเข้า/ข้อมูลออก (input/output)



การตัดสินใจ (decision)

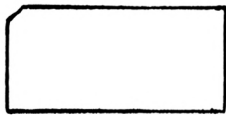


หมายเหตุ (comment)

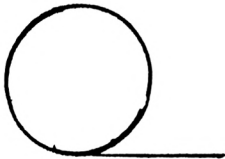
อธิบายการทำงานของคำสั่งนั้น



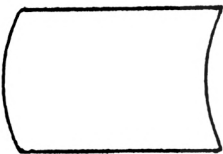
เอกสาร (document)



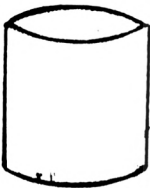
บัตร (card)



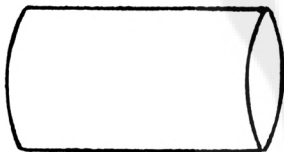
เทปบันทึก (tape)



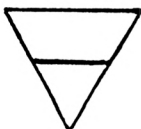
หน่วยเก็บข้อมูล (on-line storage)



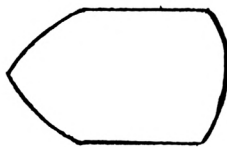
จานบันทึก (disk)



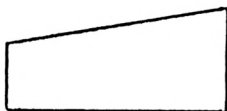
หน่วยความจำแบบทรงกระบอก (magnetic drum)



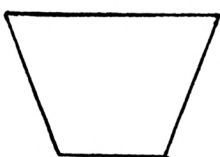
หน่วยเก็บข้อมูลไม่เชื่อมต่อ (off-line storage)



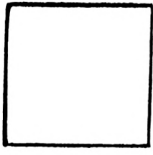
จอภาพ (monitor)



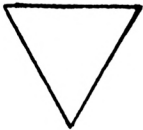
manual input



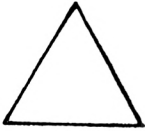
manual off-line operation



auxillary off-line operation



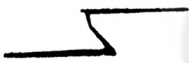
ผสาน (merge)



สกัด (extract)



การเรียงลำดับ (sort)



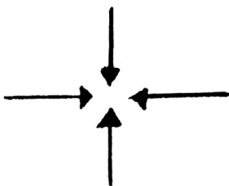
communication link



สัญลักษณ์เชื่อมต่อ (connector)



ต่อหน้าใหม่ (next page)



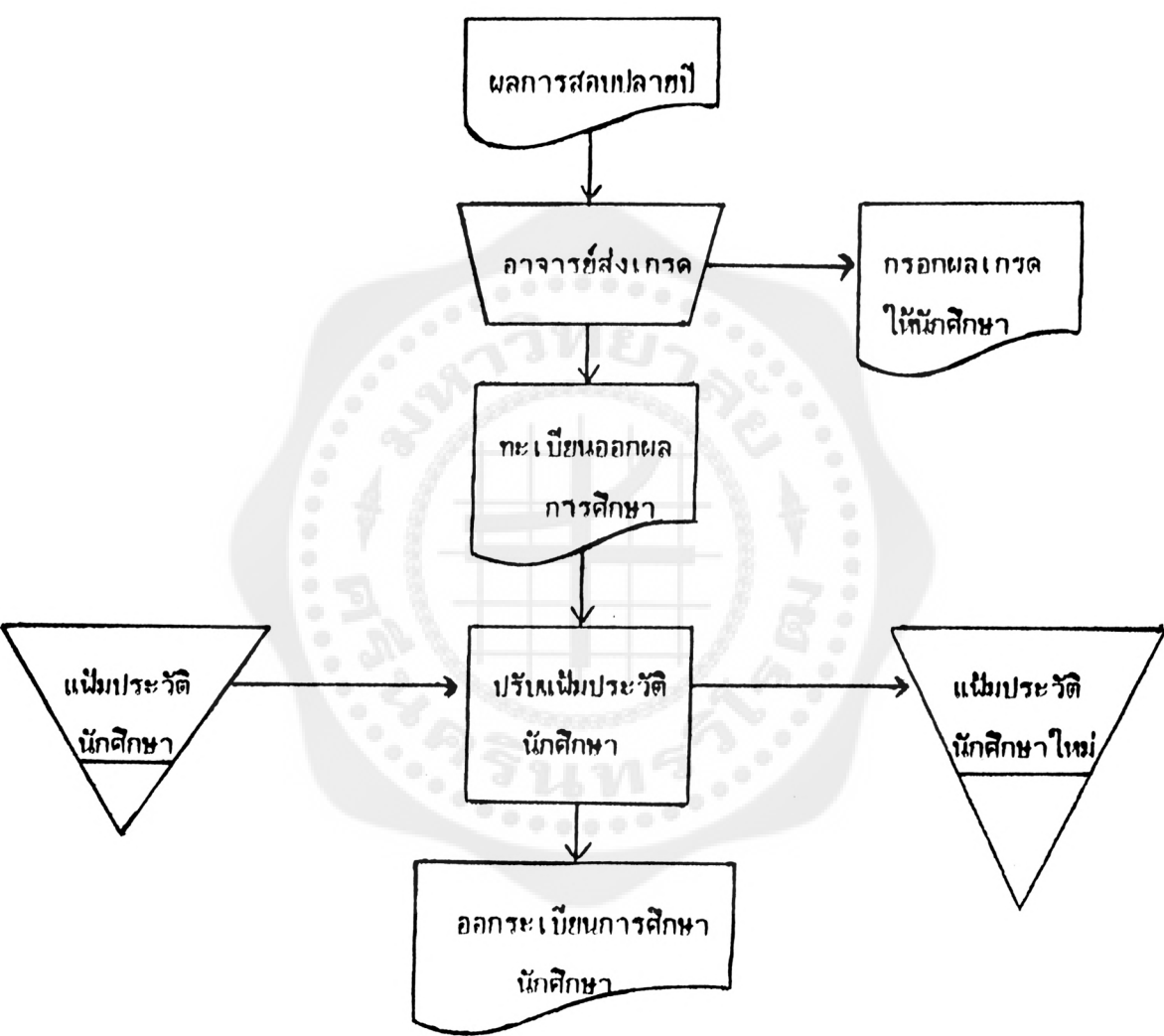
ทิศทาง (direction)

ผังงานแบ่งเป็น 2 ประเภท คือ

0.1.1 ผังงานระบบ (system flowchart) เป็นแผนผังแสดงการประมวลผลข้อมูลทั้งระบบ ตั้งแต่ข้อมูลเริ่มเข้าและออกจากระบบ นักวิเคราะห์ระบบใช้ ผังงานระบบ ออกแบบและ

อธิบายการทำงานของระบบ

ตัวอย่าง 0.1.1 แสดงการใช้ผังงานระบบของระบบเกรด

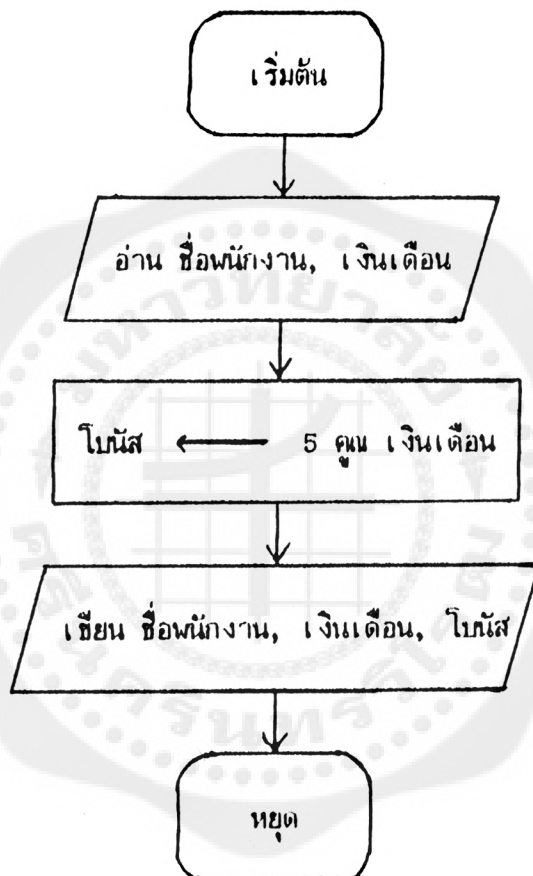


จากผลการสอบครั้งสุดท้ายของเทอม เมื่ออาจารย์ตัดเกรดส่งไปให้ฝ่ายทะเบียน และในขณะเดียวกันก็กรอกลงในใบส่งผลให้นักศึกษา เมื่อทะเบียนได้รับใบส่งเกรดก็จะนำไปปรับปรุงในแฟ้มข้อมูลที่เก็บประวัติของนักศึกษา เมื่อปรับปรุงเสร็จจะได้แฟ้มข้อมูลใหม่ แล้วจึงออกกระเบื้องการศึกษาให้นักศึกษา

0.1.2 ผังงานชุดคำสั่ง (program flowchart) เป็นลำดับของคำสั่งต่าง ๆ ในการ

พัฒนาโปรแกรมเพื่อแก้ปัญหาที่นั้น ๆ โดยเฉพาะ

ตัวอย่าง 0.1.2 บริษัทหนึ่งจ่ายโบนัสให้พนักงานต่อปี โดยสิ้นปีจะจ่ายให้เป็น 5 เท่าของเงินเดือน จะเขียนเป็นผังงานชุดคำสั่งได้ดังนี้



เมื่อเริ่มต้นจะอ่านชื่อพนักงาน และเงินเดือนเข้ามา หลังจากหาคำนวณหาโบนัส โดยนำ 5 คูณกับเงินเดือนได้เป็นโบนัส แล้วจึงพิมพ์ชื่อพนักงาน เงินเดือน และโบนัสออกมาแล้วจึงหยุด

0.2 ตัวแปร และประเภทของตัวแปร

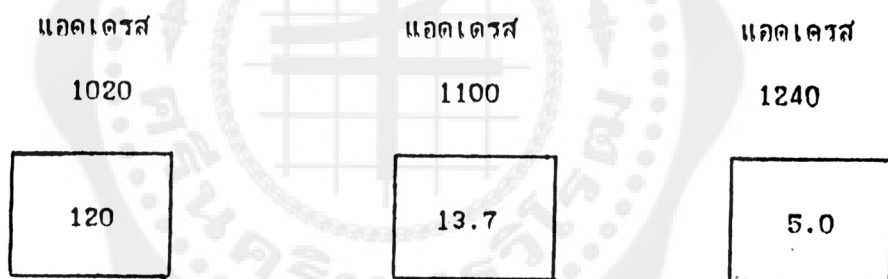
ค่าต่าง ๆ ที่ใส่เข้าไปเป็นข้อมูลจะถูกเก็บไว้ที่ใดที่หนึ่งในหน่วยความจำหลัก ซึ่งเราสามารถอ้างอิงถึงข้อมูลตัวนั้น โดยตั้งชื่อขึ้นมาแทนค่าที่นั้นซึ่งเราเรียกว่า ตัวแปร ดังนั้นตัวแปรจึงเป็นชื่อ ๑ ที่หนึ่งที่เราตั้งขึ้นมาเพื่อใช้อ้างอิงถึงข้อมูลตัวนั้น

เมื่อมีการตั้งชื่อตัวแปรขึ้นมา เครื่องคอมพิวเตอร์จะมีการจับจองพื้นที่ในหน่วยความจำหลัก ส่วนใดส่วนหนึ่งเพื่อเก็บค่าคงที่นั้นซึ่งระบุเป็นแอดเดรส (address) เพราะฉะนั้นแอดเดรสก็คือ ตำแหน่งที่ใช้ระบุบอกพื้นที่ในหน่วยความจำหลัก แทนด้วยตัวเลข

เช่น เราให้ตัวแปร X, HGT และ BASE เก็บค่าคงที่ต่อไปนี้

<u>ตัวแปร</u>	<u>ค่าคงที่</u>	<u>แอดเดรสในหน่วยความจำหลัก</u>
X	120	1020
HGT	13.7	1100
BASE	5.0	1240

ซึ่งค่าคงที่ 120 เมื่อนำไปเก็บในหน่วยความจำหลัก จะถูกเก็บที่แอดเดรส 1020 ส่วน 13.7 และ 5.0 ก็ถูกเก็บที่แอดเดรส 1100 และ 1240 ตามลำดับ ดังนี้



รูป 0.1 แสดงตัวแปร ค่าคงที่ และแอดเดรส

เมื่อโปรแกรมเมอร์ต้องการอ้างอิงถึงค่าคงที่ 120 , 13.7 และ 5.0 ที่อยู่ในหน่วยความจำหลักนั้นต้องเรียกชื่อตัวแปรคือ X, HGT และ BASE

เราสามารถแบ่งตัวแปรออกได้เป็น 3 ประเภท คือ

0.2.1 ตัวแปรจำนวน ใช้เก็บค่าคงที่ที่เป็นตัวเลข สามารถที่จะนำไปใช้ในการคำนวณได้ เช่น

<u>ตัวแปร</u>	<u>ค่าคงที่</u>
ABC	3.14156
FISH	785
TEN	10.0

0.2.2 ตัวแปรอักขระ ใช้เก็บค่าคงที่ที่เป็นข้อความ ซึ่งประกอบด้วยตัวเลข ตัวอักษร หรือสัญลักษณ์พิเศษ ไม่สามารถนำไปคำนวณได้ เช่น

<u>ตัวแปร</u>	<u>ค่าคงที่</u>
PLACE	"UNIVERSITY"
ADDRESS	" 148 SUKUMVIT ROAD"

0.2.3 ตัวแปรตรรก ใช้เก็บค่าคงที่ตรรกะ ซึ่งมี 2 อย่างคือ TRUE และ FALSE เช่น

<u>ตัวแปร</u>	<u>ค่าคงที่</u>
BOOL	TRUE
GREATER	FALSE

0.3 การให้ค่าตัวแปร

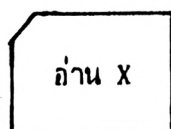
การให้ค่าตัวแปรมี 2 แบบ คือ การรับค่าจากสื่อ และโดยใช้คำสั่งกำหนดค่า

0.3.1 การให้ค่าตัวแปรโดยการรับค่าจากสื่อ

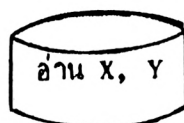
สื่อที่ใช้ในการรับข้อมูลเข้า เช่น การ์ด เทป จานบันทึก หรือหน้าจอ โดยใช้คำสั่งอ่าน (read)

รูปแบบ อ่าน ตัวแปร

จะเน้นการกำหนดค่าคงที่ที่ใส่เข้ามาทางสื่อที่กำหนดให้ตัวแปร เช่น

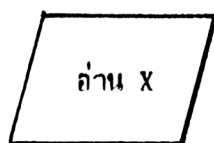


อ่านข้อมูลเข้าไปในตัวแปร X ทางบัตร



อ่านข้อมูลเข้าไปถึงในตัวแปร X และ Y ทางจานบันทึก

แต่ถ้าผู้เขียนผังงานยังไม่ทราบว่าจะให้ข้อมูลเข้าทางสื่อใดแล้วเวลาเขียนโปรแกรมจริง ๆ จึงค่อยระบุสื่อได้



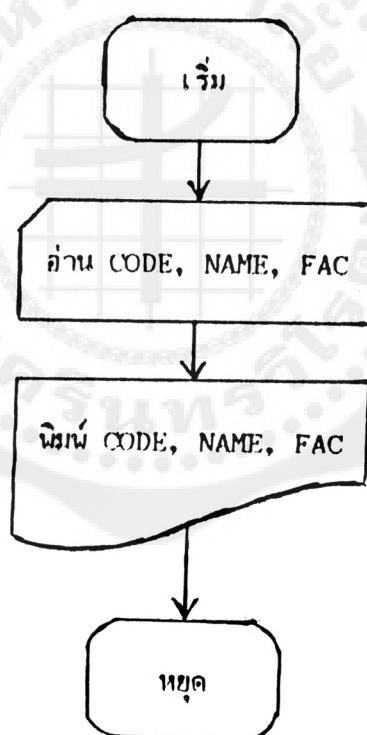
อ่านข้อมูลเข้าไปเก็บในตัวแปร x โดยระบุชื่อ

ตัวอย่าง 0.3.1 อ่าน รหัส ที่นักเรียนศึกษา คณะ เข้ามาทางบัตร แล้วพิมพ์ผลออกมาดูทางเครื่องพิมพ์

ให้ CODE แทนรหัสการศึกษา

NAME แทนชื่อการศึกษา

FAC แทนคณะ



0.3.2 การให้ค่าตัวแปรโดยคำสั่งกำหนดค่า

เมื่อมีการประมวลผลตัวแปรต่าง ๆ จะมีการเปลี่ยนแปลงค่าได้มากกว่า 1 ครั้ง เมื่อมีการเปลี่ยนแปลงค่าของตัวแปร ก็คือต้องเอาค่าใหม่ไปแทนที่ค่าเก่า ซึ่งวิธีแบบการใช้เป็น

รูปแบบ ตัวแปร ← ค่าที่กำหนด

นั่นก็คือ เอาค่าทางขวามือของเครื่องหมายลูกศรไปเก็บไว้ที่ตัวแปรทางซ้ายมือของเครื่องหมายลูกศร เช่น $X \leftarrow 1.5$ นำค่าคงที่ 1.5 ไปให้ตัวแปร X การนำค่าทางขวามือของเครื่องหมายลูกศรไปกำหนดให้ตัวแปรทางซ้ายมือ

การกำหนดค่าเป็นการนำค่าคงที่ไปกำหนดให้ตัวแปรทางซ้ายมือ ซึ่งมีรูปแบบเป็น

รูปแบบ ตัวแปร ← ค่าคงที่

ตัวอย่าง 0.3.2 $PI \leftarrow 3.1428571$ กำหนดให้ PI มีค่าเป็น 3.1428571
 $DEGREE \leftarrow 360.0$ กำหนดให้ DEGREE มีค่าเป็น 360.0
 $GRADE \leftarrow "B"$ กำหนดให้ GRADE มีข้อความเป็น "B"

การสำเนาค่าของตัวแปรทางขวามือไปกำหนดค่าให้ตัวแปรทางซ้ายมือ ซึ่งจะทำให้ตัวแปรทั้ง 2 ตัว มีค่าเท่ากัน

รูปแบบ ตัวแปร ← ตัวแปร

ตัวอย่าง 0.3.3

$X \leftarrow Y$ ถ้า Y ถูกกำหนดค่าให้เป็น 215.0 มาก่อน เมื่อพบคำสั่ง $X \leftarrow Y$ ก็จะนำค่าของ Y ซึ่งคือ 215.0 ไปกำหนดให้ X จะได้ว่า X มีค่าเป็น 215.0 เท่ากับ Y

<u>ก่อน</u>		<u>หลัง</u>	
ตัวแปร	คงที่	ตัวแปร	คงที่
X	0.0	X	215.0
Y	215.0	Y	215.0

A ← TEMP
HOUSE ← TEMP
นำค่า TEMP ไปกำหนดให้ A และ HOUSE

ก่อน		หลัง	
ตัวแปร	ค่าคงที่	ตัวแปร	ค่าคงที่
A	0	A	1718.0
TEMP	1718.0	TEMP	1718.0
HOUSE	46	HOUSE	1718.0

ก่อนอื่นขออธิบายคำว่า นิพจน์ (expression)
นิพจน์ คือ การนำตัวแปรหรือค่าคงที่มาดำเนินการกันแล้วได้ผลลัพธ์ออกมาหนึ่งค่า

รูปแบบ operand 1 [operator] operand 2

โดยที่ operand 1, operand 2 เป็น ค่าคงที่หรือตัวแปร
operator เป็น ตัวดำเนินการทางคณิตศาสตร์ ซึ่งมีดังนี้

ตัวดำเนินการ	สัญลักษณ์
บวก	+
ลบ	-
คูณ	*
หาร	/
ยกกำลัง	^

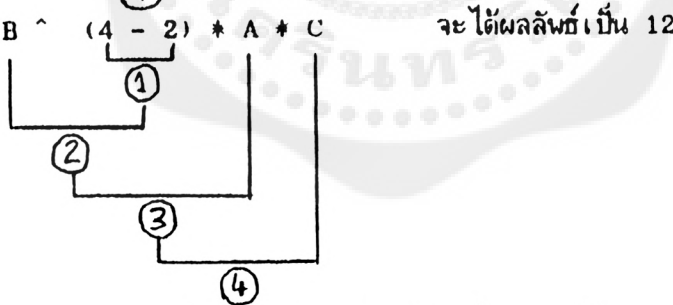
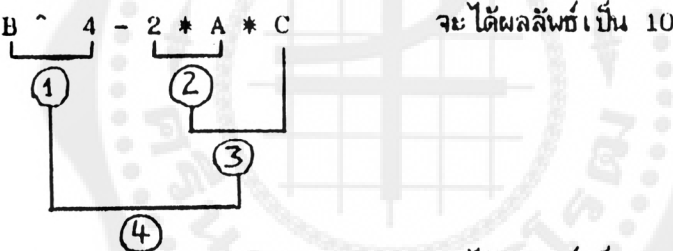
เช่น X * 1.5	นำค่าของ X คูณกับ 1.5
A * B + C	นำค่าของ A คูณกับ B แล้วบวกกับ C

เมื่อมีตัวดำเนินการหลายตัวจะเรียงลำดับการทำงานจากก่อนไปหลังดังนี้

ลำดับที่	สัญลักษณ์
1	() ,
2	^
3	* /
4	+ -

หมายเหตุ ถ้าลำดับเท่ากันจะทำจากซ้ายไปขวา

ตัวอย่าง 0.3.4 ถ้ากำหนดให้ B มีค่าเป็น 2, A มีค่าเป็น 1, C มีค่าเป็น 3 แล้ว



ถ้า U มีค่าเป็น 8 และ V มีค่าเป็น 2 แล้ว



การแทนที่เป็นการนำค่าสุดท้ายที่คำนวณได้จากทางขวามือไปแทนที่ค่าตัวแปรทางซ้ายมือ

รูปแบบ ตัวแปร $\longleftarrow$ นิพจน์

เช่น $P \longleftarrow A * B + C/3$

<u>ก่อน</u>		<u>หลัง</u>	
ตัวแปร	ค่าคงที่	ตัวแปร	ค่าคงที่
P	0	P	15
A	7	A	7
B	2	B	2
C	3	C	3

ถ้าต้องการเพิ่มค่าตัวแปรไปอีก I จะมีรูปแบบเป็น

รูปแบบ ตัวแปร $\longleftarrow$ ตัวแปร (เดิม) + I

โดยที่ I เป็น ค่าคงที่ ตัวแปร หรือ นิพจน์

เช่น	$COUNT \longleftarrow COUNT + 1$	เพิ่มค่า count ไปอีก 1
I	$\longleftarrow I + 5$	เพิ่มค่า I ไปอีก 5
X	$\longleftarrow X + J$	เพิ่มค่า X ไปอีก J
Y	$\longleftarrow Y - 2$	ลดค่า Y ไปอีก 2
L	$\longleftarrow L + (2 * K)$	เพิ่มค่า L ไปอีก 2 เท่าของ K

<u>ก่อน</u>		<u>หลัง</u>	
ตัวแปร	ค่าคงที่	ตัวแปร	ค่าคงที่
COUNT	2	COUNT	3
I	10	I	15
X	7	X	10
J	3	J	3
Y	80	Y	78
K	4	K	4
L	20	L	28

- หมายเหตุ
1. ทางซ้ายมือของเครื่องหมาย ← ต้องเป็นตัวแปรเท่านั้น
 2. ใช้เครื่องหมาย = แทน ← ได้

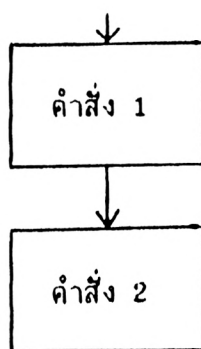
0.4 โครงสร้างหลักสำหรับการประมวลผล

โครงสร้างหลักพื้นฐานในการเขียนผังงานประกอบด้วย 3 โครงสร้างใหญ่ ๆ คือ

- โครงสร้างเรียงลำดับ (sequence structure)
- โครงสร้างการตัดสินใจ (decision structure)
- โครงสร้างการทำงานซ้ำในวงจร (loop structure)

0.4.1 โครงสร้างเรียงลำดับ

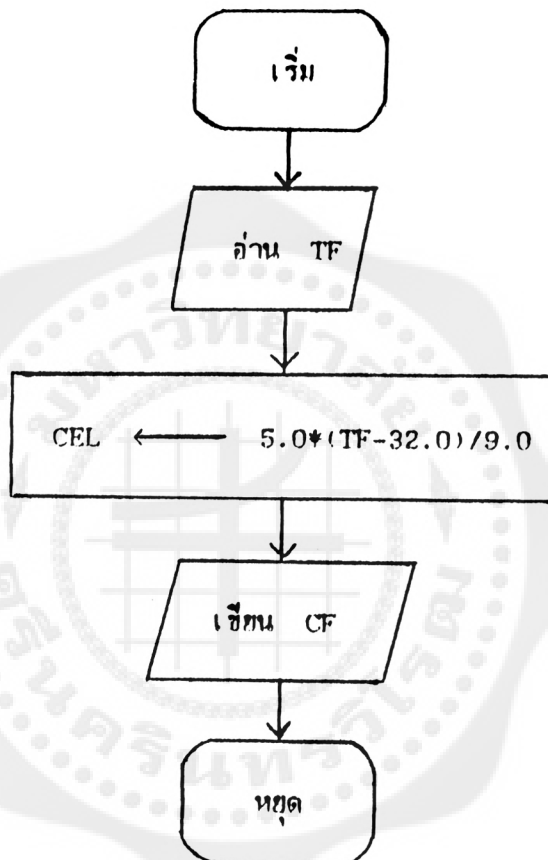
โครงสร้างเรียงลำดับ เป็นการทำงานจากบนลงล่างเรียงตามลำดับของคำสั่ง



ตัวอย่าง 0.4.1 การเปลี่ยนอุณหภูมิของสภาพาเรนไฮต์ เป็นองศาเซลเซียส

กำหนดให้ตัวแปร TF แทน อุณหภูมิของสภาพาเรนไฮต์

ตัวแปร CEL แทน อุณหภูมิของศาเซลเซียส



เริ่มต้นก็จะอ่านข้อมูล TF เข้าไปเป็นอุณหภูมิของสภาพาเรนไฮต์ แล้วคำนวณหาอุณหภูมิของศาเซลเซียส

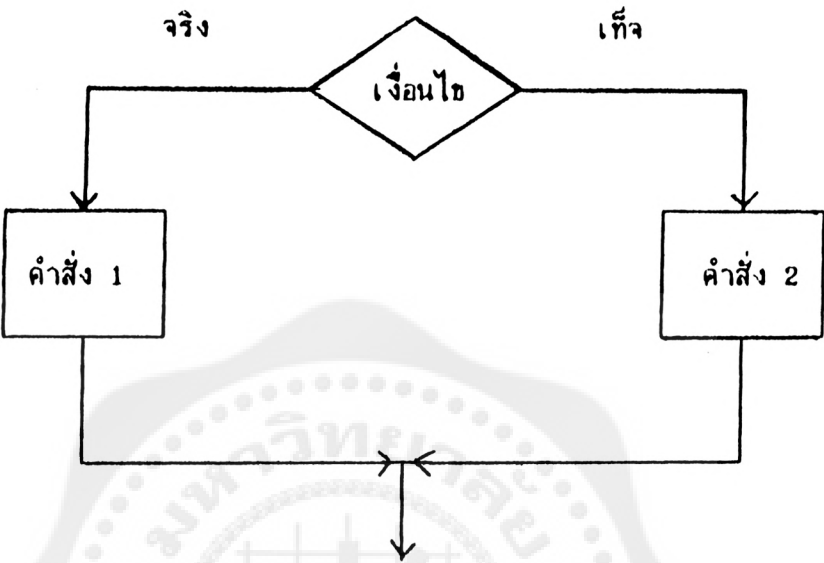
โดยใช้สูตร $5.0 * (TF - 32) / 9.0$ เมื่อได้ผลลัพธ์ก็จะแสดงผลออกมา

0.4.2 โครงสร้างการตัดสินใจ

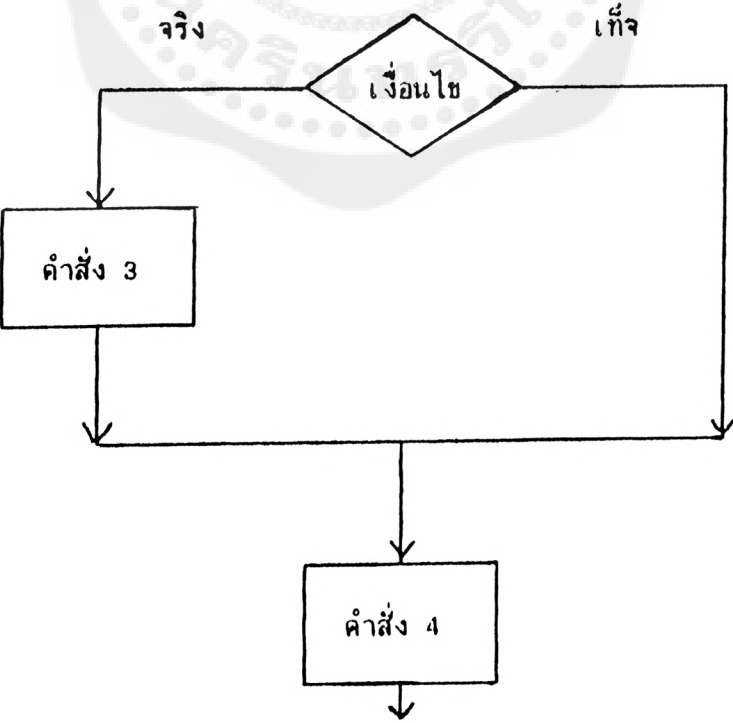
เมื่อมีทางเดินมากกว่า 2 ทาง ต้องตัดสินใจแล้วว่าจะเลือกเส้นทางใด โดยตั้งเงื่อนไขขึ้นมาแล้วตรวจสอบเงื่อนไขที่เกิดขึ้นว่าเป็นจริงหรือเท็จ

ก. มีทางเลือก 2 ทาง

ถ้าเงื่อนไขเป็นจริงไปทำคำสั่ง คำสั่ง 1 ถ้าเงื่อนไขเป็นเท็จไปทำ คำสั่ง 2

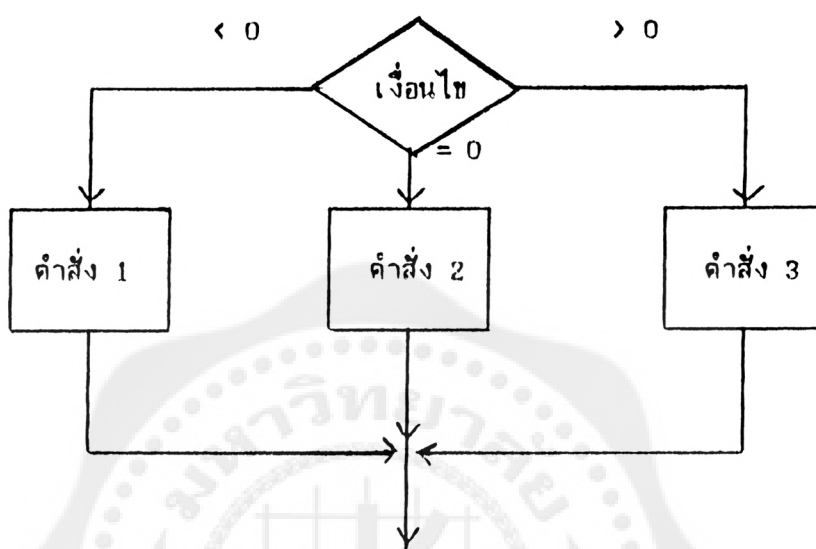


หรือ ถ้าเงื่อนไขจริงไปทำ คำสั่ง 3 และ คำสั่ง 4 ถ้าเงื่อนไขเป็นเท็จไปทำ คำสั่ง 4



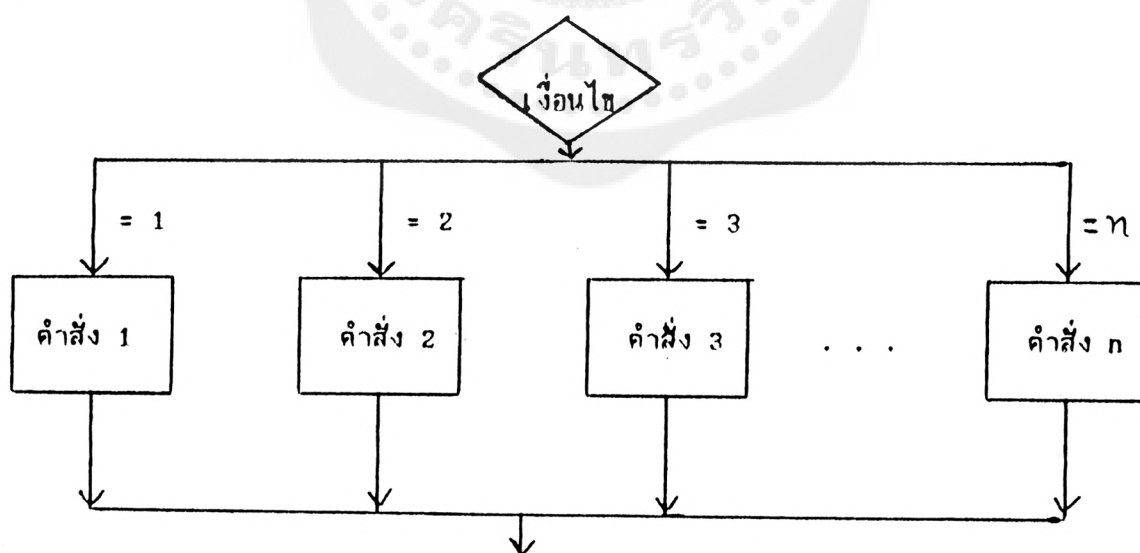
ข. มีทางเลือก 3 ทาง

ถ้าเงื่อนไข < 0 ไปทำ คำสั่ง 1 ถ้าเงื่อนไข $= 0$ ไปทำ คำสั่ง 2 และถ้าเงื่อนไข > 0 ไปทำ คำสั่ง 3



ค. มีทางเลือก n ทาง

ถ้าเงื่อนไข $= 1$ ไปทำ คำสั่ง 1, ถ้าเงื่อนไข $= 2$ ไปทำ คำสั่ง 2, ถ้าเงื่อนไข $= 3$ ไปทำ คำสั่ง 3, ..., ถ้าเงื่อนไข $= n$ ไปทำ คำสั่ง n



ตัวอย่าง 0.4.2 เมื่อต้องการปรับปรุงยอดของเงินฝากหรือถอนโดยให้

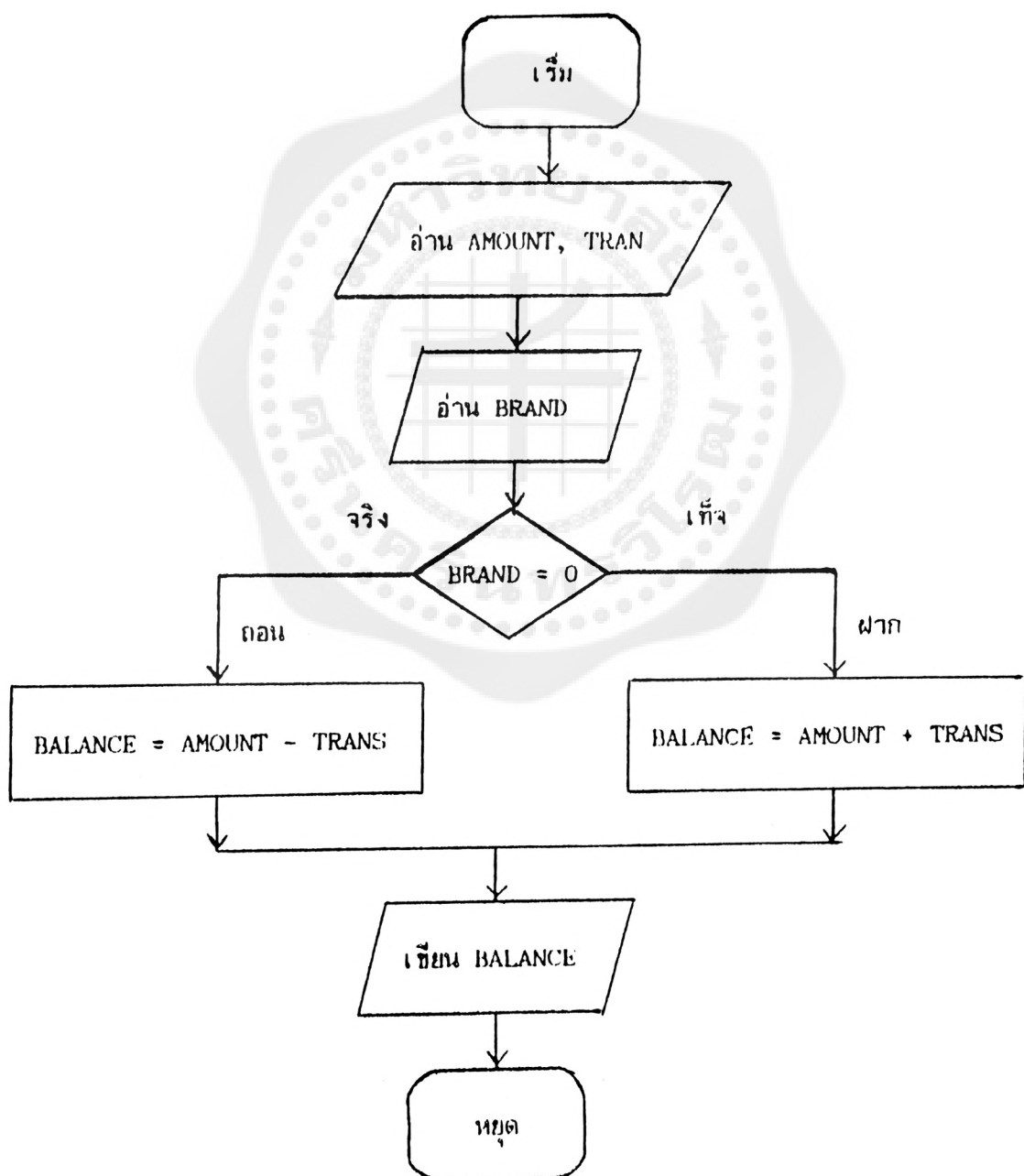
AMOUNT เป็น เงินต้น

TRANS เป็น เงินที่จะฝากหรือถอน

BALANCE เป็น ยอดเงินคงเหลือ

แล้วทำให้ BRAND เป็นตัวกำหนดว่าจะฝากหรือถอน โดย

ถ้า BRAND = 1 ฝาก BRAND = 0 ถอน

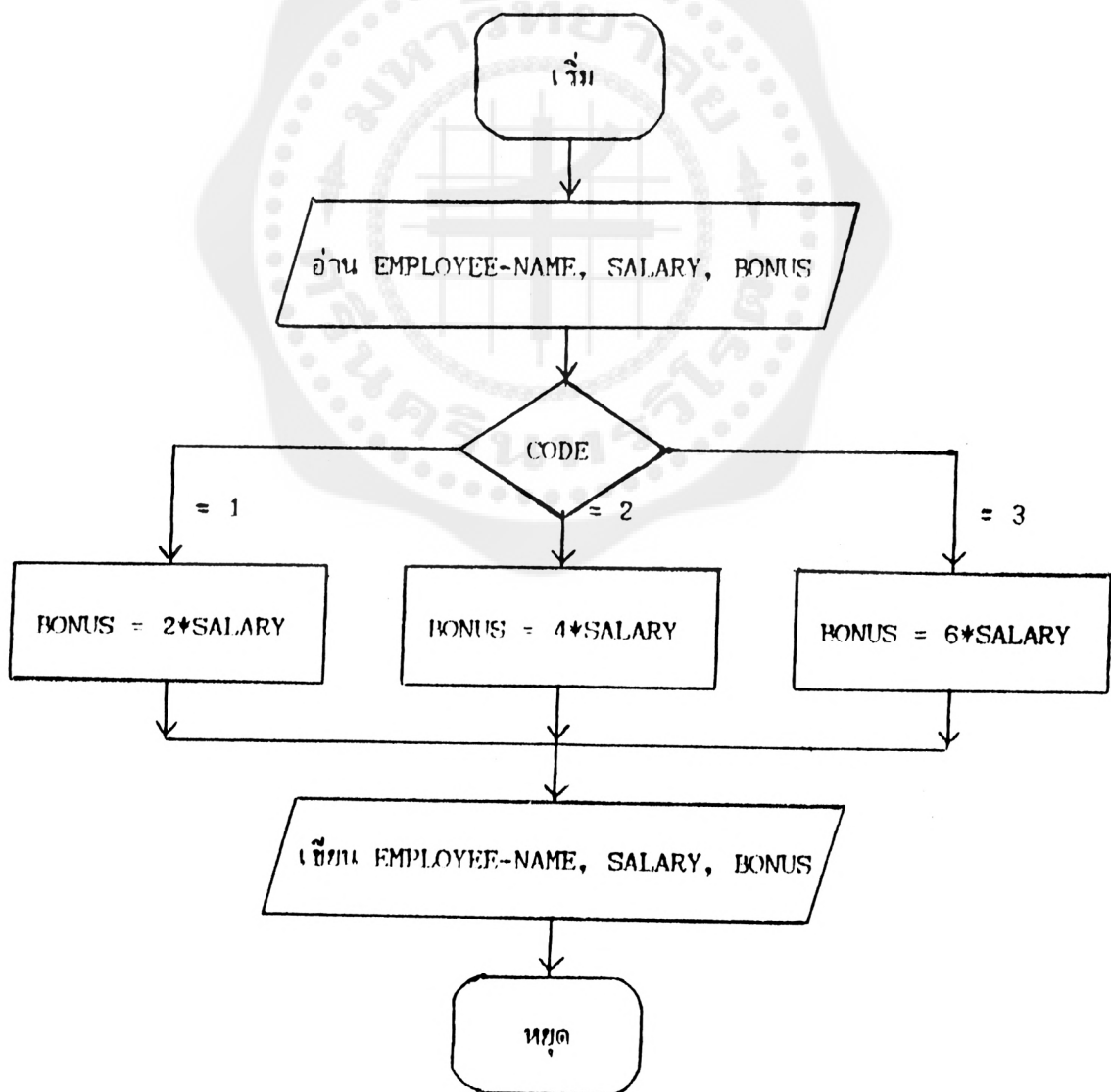


ตัวอย่าง 0.4.3 บริษัทหนึ่งแบ่งเงินโบนัสให้พนักงานต่อไปนี้ โดยแบ่งพนักงานเป็น 3 กลุ่ม

- | | | |
|------------|----------------------|------------------|
| กลุ่มที่ 1 | ระดับลูกจ้างชั่วคราว | ให้โบนัส 2 เดือน |
| กลุ่มที่ 2 | พนักงานประจำ | ให้โบนัส 4 เดือน |
| กลุ่มที่ 3 | ระดับผู้บริหาร | ให้โบนัส 6 เดือน |

กำหนดให้ CODE แทนกลุ่มของพนักงาน มีค่าเป็น 1, 2 และ 3

EMPLOYEE-NAME	แทน	ชื่อพนักงาน
SALARY	แทน	เงินเดือน
BONUS	แทน	โบนัส

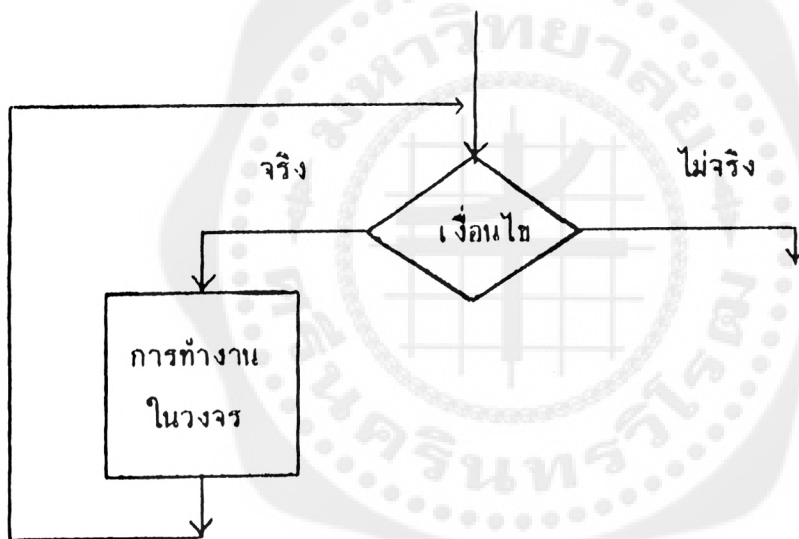


0.4.3 โครงสร้างการทำงานซ้ำในวงจร

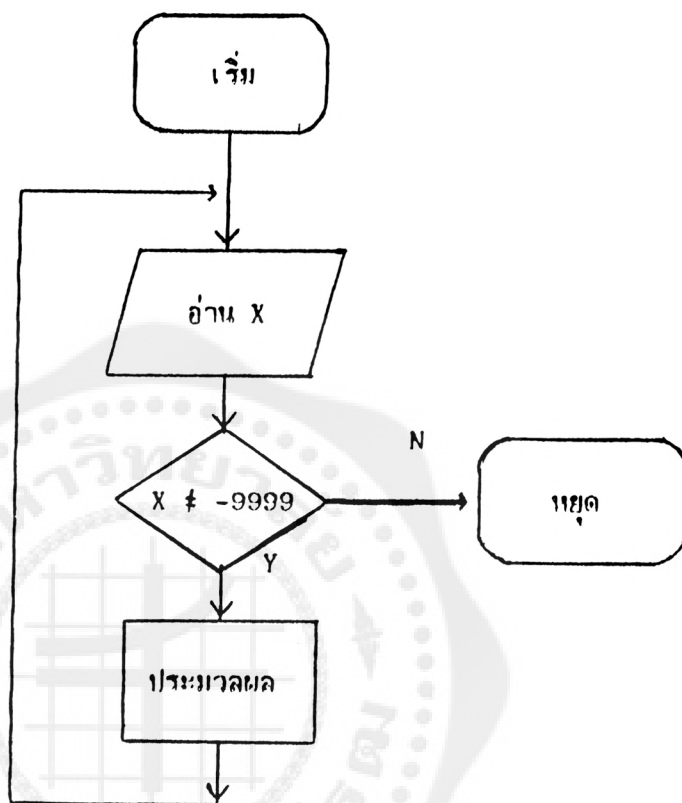
โครงสร้างการทำงานซ้ำในวงจรนั้นจะมีการกลับไปทำงานที่คำสั่งนั้น ซ้ำ ๆ ซาก ๆ หลายครั้งซึ่งเราเรียกว่า มีการวนซ้ำ อย่างน้อยจะต้องทราบว่าเมื่อไรจึงจะออกจากการวนซ้ำ ดังนั้นเราจึงต้องมีการตรวจสอบเงื่อนไข ซึ่งมีโครงสร้างหลัก 2 แบบ คือ

- แบบ While do และ
- แบบ Repeat until

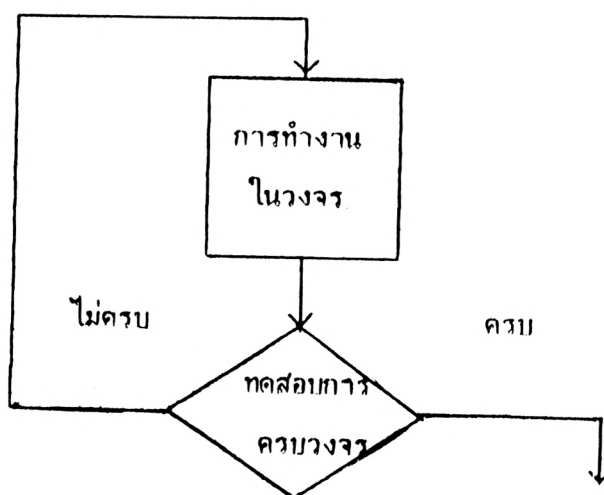
ก. แบบ While do จะมีการตรวจสอบเงื่อนไขก่อนเข้าไปประมวลผล ถ้าเป็นจริงจะเข้าวนซ้ำ ถ้าเป็นเท็จจะออกจากการวนซ้ำ



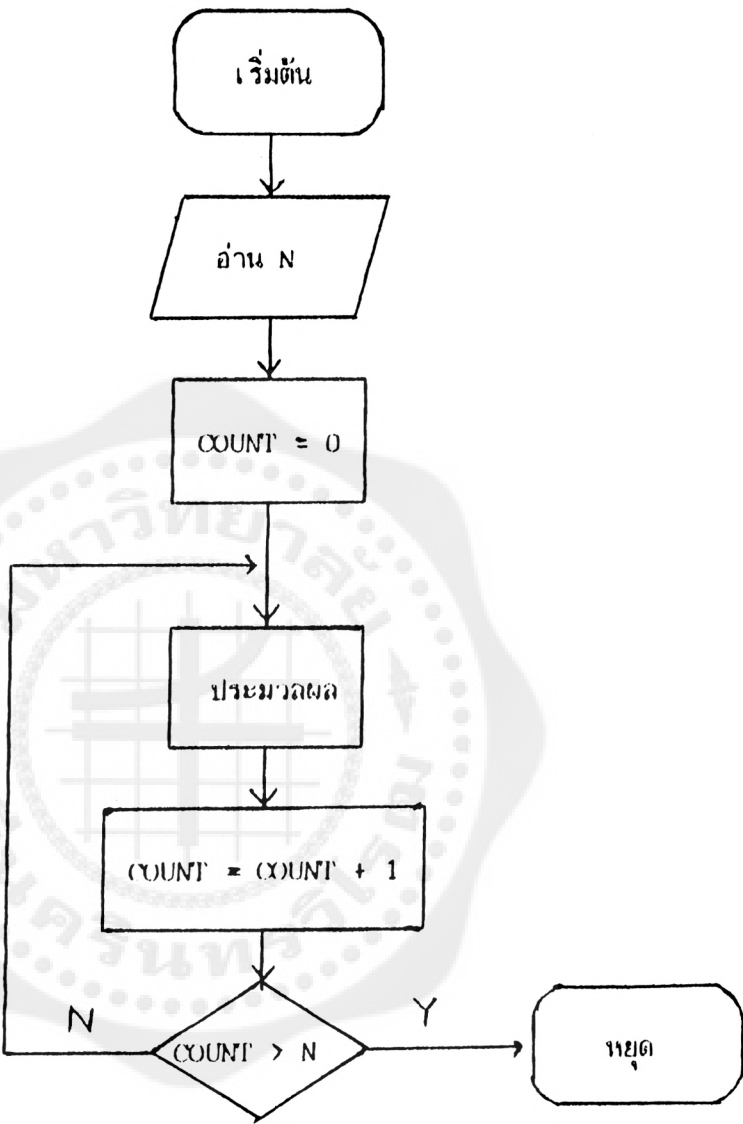
ตัวอย่าง 0.4.4 การทำงานซ้ำในวงจรแบบตรวจสอบเงื่อนไขก่อน



ท. แบบ Repeat, until ตรวจสอบเงื่อนไขหลังจากประมวลผลเสร็จแล้ว ถ้าครบ
วงจรจะออกจากการวนซ้ำ ถ้าไม่ครบวงจรจะวนซ้ำลูป



ตัวอย่าง 0.4.5 การทำงานซ้ำในวงจรแบบตรวจสอบเงื่อนไขเมื่อประมวลผลเสร็จแล้ว



ตัวอย่าง 0.4.8 ให้หาผลบวกของเลข $2 + 4 + 6 + 8 + 10 + \dots + 20$

กำหนดให้ SUM เป็นผลรวมของ $2 + 4 + \dots + 20$ กำหนดค่าแรก $SUM = 0$

K เก็บค่า $2, 4, 6, \dots, 20$

เมื่อ $K = 2$ $SUM = 0 + 2$

$K = 4$ $SUM = 2 + 4$

$K = 6$ $SUM = 6 + 6$

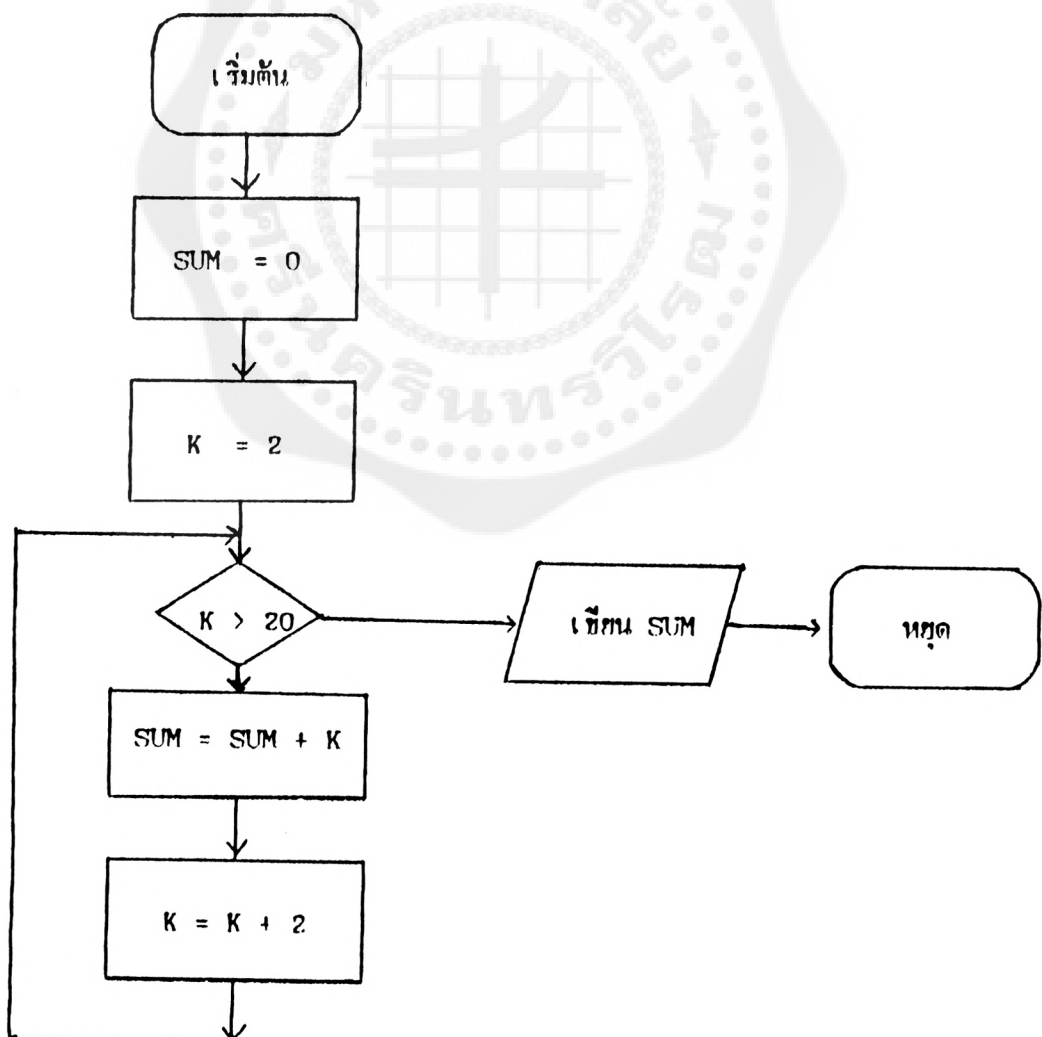
$K = 8$ $SUM = 12 + 8$

ค่า K ถูกเพิ่มทีละ 2

ค่า SUM ถูกเพิ่มทีละ K

นั่นก็คือ $K = K + 2$

นั่นก็คือ $SUM = SUM + K$



ตัวอย่าง 0.4.7 หาคะแนนเฉลี่ยของคะแนนสอบ mid-term ของนักศึกษารวมทั้งหาค่าเบี่ยงเบนมาตรฐาน

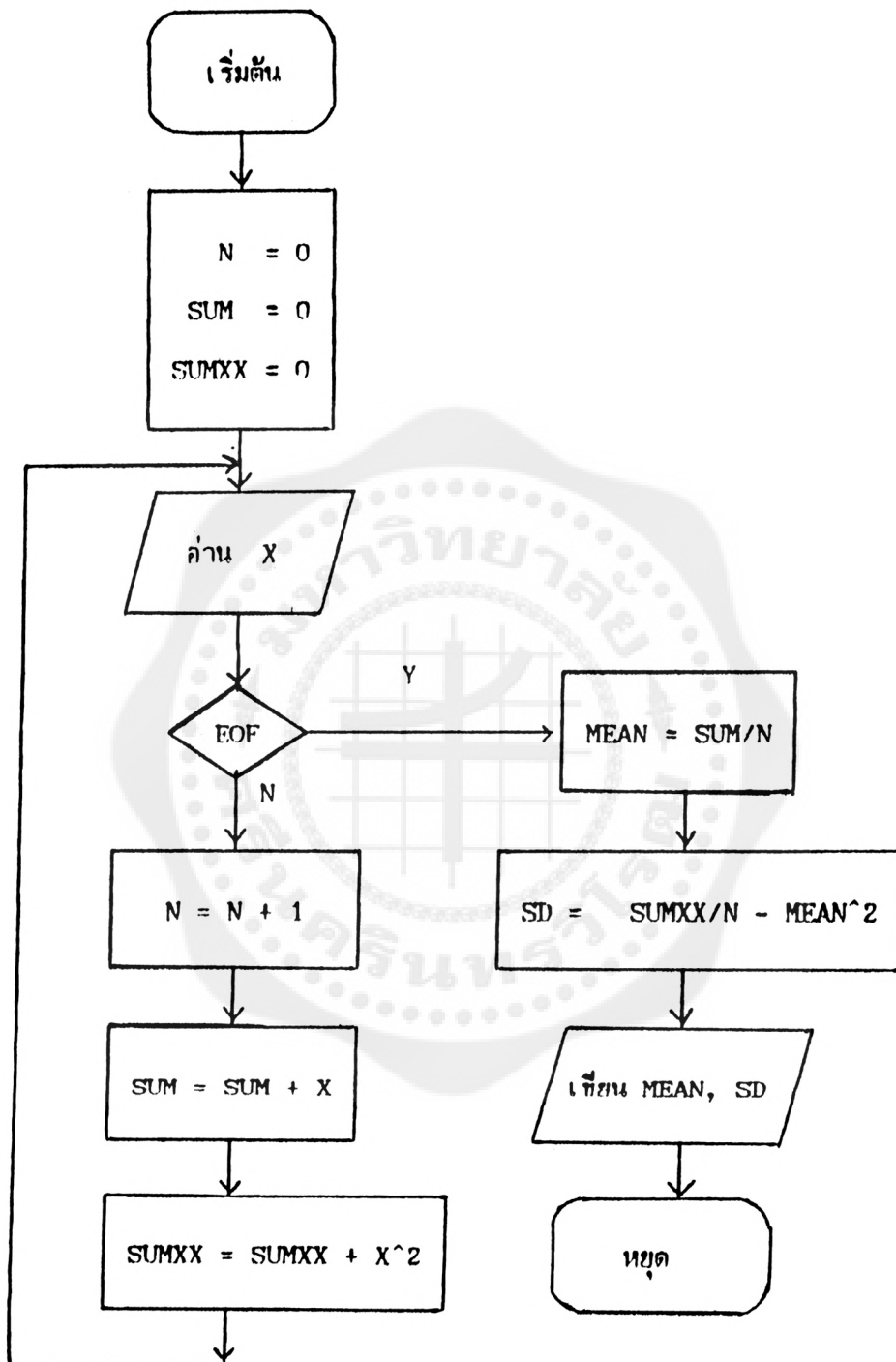
โดยกำหนดสูตร
$$\bar{X} = \sum_{i=1}^N X_i / N$$

$$SD = \sqrt{\sum_{i=1}^N X_i^2 / N - \bar{X}^2}$$

ในกรณีนี้ไม่ทราบว่ามีห้องหนึ่งมีนักศึกษากี่คน ดังนั้นเราจะต้องอ่านจนกว่าหมดแฟ้ม โดยตรวจสอบว่าหมดแฟ้ม (EOF : End Of File) หรือยัง ถ้าหมดแฟ้มก็สามารถหา $\bar{X}$ และ SD ได้ โดยในขณะที่อ่านต้องนับจำนวนนักศึกษาไปด้วย

กำหนดให้

- X แทนคะแนนของนักศึกษาแต่ละคน
- N แทนจำนวนนักศึกษาทั้งหมด
- SUM แทนคะแนนรวมของนักศึกษาทั้งหมด
- SUMXX แทนคะแนนรวมของนักศึกษายกกำลัง 2
- MEAN แทนคะแนนเฉลี่ยของนักศึกษา
- SD แทนค่าเบี่ยงเบนมาตรฐาน



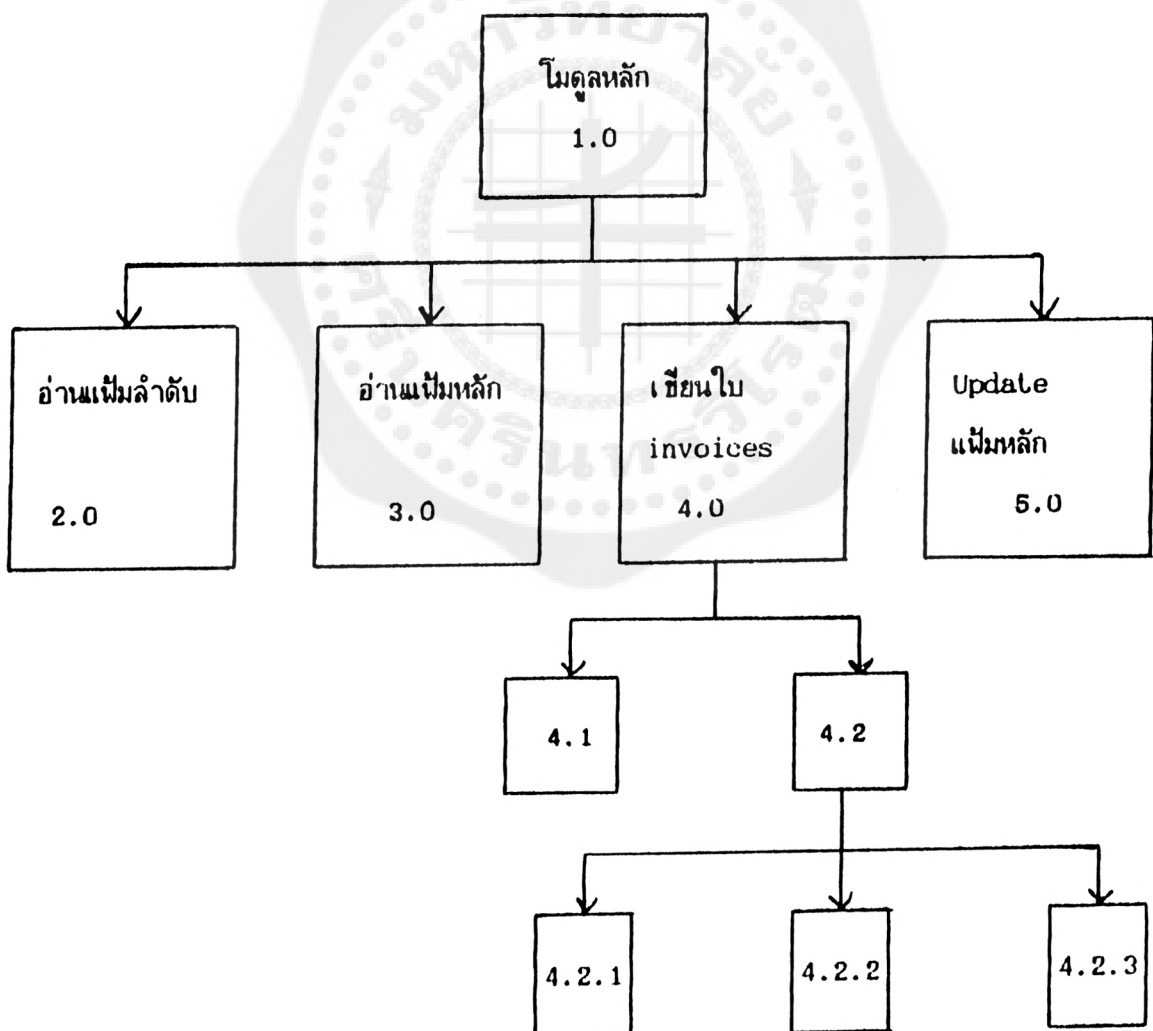
0.5 การออกแบบจากบนลงล่าง

ในการเขียนโปรแกรมใหญ่ ๆ เราจะแบ่งเป็นส่วน ๆ แยกไป โดยจัดส่วนที่เหมือนกันไว้ด้วยกัน แต่ละส่วนนี้เราเรียกว่า โมดูล (module) โดยจะเขียนผังงานจากโมดูลหลักแยกไปเป็น

โมดูลย่อย แล้วไลต์จากโมดูลหลักไปโมดูลย่อย เป็นการออกแบบจากบนลงล่าง ซึ่งจะช่วยให้เขียนได้ง่ายและแก้ไขได้ง่ายขึ้น

ดังนั้นเราจึงใช้การออกแบบจากบนลงล่างในการพัฒนาโมดูล โดยจะเริ่มเขียนโมดูลหลักก่อน และระดับต่อไปก็เขียนโมดูลย่อยซ้อนไปเรื่อย ๆ ในการนำไปเขียนโปรแกรมจะสามารถทดสอบและแก้ไขเป็นส่วน ๆ ได้

ตัวอย่าง 0.5.1 เมื่อร้านค้าแห่งหนึ่งต้องการเขียนโปรแกรมคิดคำนวณ จำนวนสินค้าคงเหลือ และราคาได้ออกแบบข้อมูลต่าง ๆ จากบนลงล่างดังนี้



กำหนดให้

แฟ้มหลัก เป็นไฟล์ที่เก็บรายการสินค้าทั้งหมดของร้าน เช่น รหัสสินค้า ชื่อสินค้า
จำนวน

สินค้าคงเหลือ ราคาต่อหน่วย ฯลฯ

แฟ้มลำดับ เป็นไฟล์ที่ลูกค้าสั่งซื้อสินค้ามา เช่น รหัสสินค้า จำนวนที่ต้องการ

คำอธิบายโมดูลต่าง ๆ

โมดูล 1.0 : โมดูลหลักสำหรับคิดคำนวณการสั่งซื้อสินค้า

โมดูล 2.0 : อ่านแฟ้มลำดับ

โมดูล 3.0 : อ่านแฟ้มหลัก เพื่อค้นหารหัสสินค้าที่ลูกค้าสั่ง

โมดูล 4.0 : เขียนใบเสร็จ และออกรายงานต่าง ๆ

โมดูล 4.1 : -----

โมดูล 4.2 : -----

โมดูล 4.2.1 : -----

โมดูล 4.2.2 : -----

โมดูล 4.2.3 : -----

โมดูล 5.0 : ปรับปรุงรายการสินค้าทั้งหมดที่ถูกลบออกไปแล้วในแฟ้มหลัก รวมทั้งขอรวมด้วย
ต่อไปก็สามารถเขียนผังงานตามโมดูลได้

แบบฝึกหัดบทที่ 0

1. ให้เขียนผังงานคำนวณหาค่า Z จาก

$$Z = \frac{(U / V) + (X / Y)}{2}$$

เมื่ออ่านค่า U, V, X และ Y เข้ามา 10 จำนวน

2. ให้เขียนผังงานคำนวณหาค่า P

$$P = \begin{cases} \log(t^2 - a) & \text{เมื่อ } t^2 > a \\ \log(t^2) & \text{เมื่อ } t^2 \leq a \end{cases}$$

3. ให้หาปริมาตรและพื้นที่ของวงกลมเมื่อทราบรัศมี

$$V = 4\pi r^3 / 3$$

$$A = 4\pi r^2$$

เมื่อ V คือ ปริมาตร

A คือ พื้นที่

r คือ รัศมีของวงกลม

4. จากสมการ quadratic equation

$$ax^2 + bx + c = 0$$

a, b และ c เป็นค่าคงที่ที่เป็นสัมประสิทธิ์ของ x^2 , x และ 1

เมื่อทราบว่า a, b, c เราสามารถหาค่า X ได้จากสูตร

$$X_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

$$X_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$

ให้เขียนผังงานเพื่อคำนวณ X_1 และ X_2 ที่เป็นรากของสมการ quadratic แต่ถ้า $b^2 - 4ac < 0$ เราไม่สามารถหาค่า X_1 และ X_2 ออกมาได้ เพราะว่าเป็นค่าจินตภาพ (imaginary part) ให้พิมพ์คำว่า "Can't find root" แล้วกลับไปอ่านค่าใหม่เข้ามาจนกว่าจะหมดแผ่น

5. เมื่อทราบด้านของสามเหลี่ยม 3 ด้าน สามารถหาพื้นที่ของสามเหลี่ยมได้จากสูตร

$$A = \sqrt{S (s - a) (s - b) (s - c)}$$

โดย a, b, c เป็นด้านของสามเหลี่ยมทั้ง 3

A = เป็นพื้นที่ของสามเหลี่ยม

$$S = \frac{a + b + c}{2}$$

ให้เขียนผังงานหาพื้นที่ของสามเหลี่ยมเมื่อทราบด้าน a, b, c ดังนี้

a	b	c
11.88	8.06	12.75
5.55	4.54	7.56
10.00	10.00	10.00
12.75	9.89	11.42
12.00	8.00	12.00
20.42	27.42	31.59
7.17	2.97	6.66
173.67	87.38	139.01

รวมทั้งหารัศมีของวงกลมที่บรรจุในสามเหลี่ยม และนอกสามเหลี่ยม โดยที่ถ้า r_c เป็นรัศมีของสามเหลี่ยมที่บรรจุในวงกลม แล้ว $r_c = abc / (4A)$

ถ้า r_i เป็นรัศมีของสามเหลี่ยมที่บรรจุนอกวงกลม แล้ว $r_i = A/S$

6. ถ้าต้องการคิติดอกเบี้ยเงินฝาก โดย i เป็นอัตราดอกเบี้ย 10% ต่อปี และ P เป็นเงินต้น 500 บาท ถ้าฝาก 10 ปี จะได้ยอดเงินฝากทั้งหมดเป็น

$$A = P(1 + i)^n \text{ โดย } n \text{ เป็นระยะเวลาที่คิติดอกเบี้ย}$$

ถ้ากำหนดให้คิติดอกเบี้ยเมื่อครบปีครึ่งหนึ่ง แล้วจะได้ยอดเงินฝากเท่าไร

แต่ถ้าเราคิติดอกเบี้ย 3 เดือนครั้ง จะคิติดอกเงินฝากได้เป็น

$$A = P(1 + i/4)^{4n} \text{ โดย } n = 3/12 \text{ เดือน}$$

อยากรทราบว่าการคิติดอกเงินฝากวิธีหลังนี้จะเท่ากับวิธีแรกหรือไม่ ให้เขียนผังงานคำนวณออกมาดู

บรรณานุกรม

- Byron S. Gottfried. Theory and Problems of Programming with BASIC,
McGraw Hill Book Company, 1975
- Daniel D. Benice. Introduction to Computer and Data Processing,
Prentice-Hall Company, 1980.



