

บทที่ 5 การพัฒนาโปรแกรม



รวบรวมโดย อ.สิทธิโชค ทรัพย์ไพบุลย์กิจ

2. การออกแบบโปรแกรม (Program Design) หรือการออกแบบขั้นตอนวิธี (Algorithm Design)

นำผลการวิเคราะห์จากขั้นตอนที่ 1 การวิเคราะห์ปัญหา มาแสดงรายละเอียดได้ 3 รูปแบบ คือ

2.1. ผังงาน (Flowchart) คือ การใช้รูปภาพหรือสัญลักษณ์ แทนรายละเอียดขั้นตอนการแก้ปัญหาอย่างเป็นเหตุเป็นผล

2.2. รหัสเทียม (Pseudo code) คือการใช้ภาษาเขียนแทนรายละเอียดขั้นตอนการแก้ปัญหาอย่างเป็นเหตุเป็นผล

2.3 ขั้นตอนวิธี (Algorithm) ลำดับการทำงานที่เป็นขั้นตอนที่ชัดเจนเพื่อใช้ในการแก้ปัญหา

1.Programming Methodology

ขั้นตอนการออกแบบโปรแกรม

- การแก้ปัญหา (Problem Solving)
 - 1 การวิเคราะห์ปัญหา (Problem Analysis)
 - ระบุปัญหา และทำการวิเคราะห์ปัญหา
 - 2 การออกแบบโปรแกรม (Program Design)
 - การออกแบบโปรแกรมเพื่อแก้ปัญหา หรือการออกแบบขั้นตอนวิธี (Algorithm Design)
 - 3 การเขียนโปรแกรม (Program Coding)
 - เลือกภาษาในการเขียนโปรแกรม
 - เขียนโปรแกรมจากขั้นตอนวิธีที่ออกแบบไว้
 4. ทดสอบโปรแกรม และแก้ไขข้อผิดพลาด (Program Testing and Debugging)

2.1 การเขียนผังงาน (Flowchart)

แผนภาพแสดงลำดับขั้นตอนการทำงาน เป็นเครื่องมือที่ใช้ในการวางแผนขั้นแรก โดยใช้สัญลักษณ์ต่าง ๆ ในการเขียนผังงาน เพื่อจัดลำดับขั้นตอนในการแก้ปัญหา ซึ่งช่วยในการเขียนโปรแกรม เป็นวิธีที่นิยมใช้เพราะทำให้เห็นภาพในการทำงานของโปรแกรมง่ายกว่าใช้ข้อความ หากมีข้อผิดพลาด สามารถดูจากผังงานจะทำให้การแก้ไขหรือปรับปรุงโปรแกรมทำได้ง่ายขึ้น

2.2 การเขียนรหัสเทียม (Pseudo code)

เป็นการเขียนบรรยายเพื่ออธิบายขั้นตอนการแก้ปัญหาเช่นเดียวกับผังโปรแกรม สามารถใช้ภาษาอังกฤษ หรือภาษาอื่นได้ สามารถเปลี่ยนเป็นภาษาคอมพิวเตอร์ได้ง่าย มีลักษณะเป็นโครงสร้างคล้ายกับผังโปรแกรม

2.3 ขั้นตอนวิธี(Algorithms)

ลำดับขั้นตอนวิธีการทำงานที่ชัดเจนอย่างเป็นเหตุเป็นผล ไม่กำกวม ซึ่งผ่านการวิเคราะห์ และแยกแยะ สำหรับอธิบายการแก้ปัญหา ซึ่งถ้าปฏิบัติตามอย่างถูกต้องแล้ว จะทำให้งานหรือการประมวลผลที่ต้องการนั้นสำเร็จ ใช้ภาษาใดก็ได้ในการนำเสนอขั้นตอนวิธี

ความแตกต่างของขั้นตอนวิธี และรหัสเทียม

รหัสเทียมจะอธิบาย หรือแปลงขั้นตอนวิธี ให้สามารถนำไปเป็นแนวทางในการเขียนโปรแกรมได้จริง โดยทั่ว ๆ ไปจะใช้ภาษาอังกฤษ และสัญลักษณ์ที่เป็นสากล

3. การเขียนโปรแกรม

ขั้นตอนนี้เป็นการนำส่วนออกแบบโปรแกรมที่ออกแบบไว้แล้ว จากขั้นตอนที่ 2 (ผังงาน หรือรหัสเทียม) มาเขียนเป็นภาษาคอมพิวเตอร์ เช่น C C++ Java เป็นต้น

4. การทดสอบโปรแกรม

แบ่งการตรวจสอบออกเป็น 3 ส่วนคือ

- 4.1. ความผิดพลาดทางไวยากรณ์ (Syntax Error)
- 4.2. ความผิดพลาดในการดำเนินงาน (Runtime Error)
- 4.3. ความผิดพลาดทางตรรกะ (Logical Error)

4.1 ความผิดพลาดทางไวยากรณ์ (Syntax Error)

เป็นข้อผิดพลาดที่เกิดเนื่องจากเขียนคำสั่งของภาษาไม่ถูกต้อง ไม่ถูกต้องตามหลักไวยากรณ์ ข้อผิดพลาดนี้โปรแกรมจะหยุดการทำงานและพบได้ในช่วงแปลโปรแกรม เช่น จาก คำสั่ง ตัวอย่าง

4.2 ความผิดพลาดในการดำเนินงาน (Runtime Error)

เกิดจากวิธีการเขียนโปรแกรม เช่น การหารด้วยศูนย์ ชนิดข้อมูลไม่ถูกต้อง ข้อมูลที่ได้เกินขีดจำกัดบน หรือน้อยกว่าขีดจำกัดล่าง ข้อผิดพลาดนี้โปรแกรมจะหยุดการปฏิบัติการ และพบได้ในช่วงการปฏิบัติงานโปรแกรม

4.3 ความผิดพลาดทางตรรกะ (Logical Error)

เกิดเนื่องจากผู้เขียน เขียนผิดขั้นตอน ผู้เขียน ตีความหมายของปัญหาผิด ทำให้ได้ผลลัพธ์ไม่ถูกต้อง เช่น ต้องการ $2+4$ แต่พิมพ์ผิดเป็น $2*4$ ทำให้ได้คำตอบที่ไม่ถูกต้อง ข้อผิดพลาดนี้โปรแกรมจะไม่หยุดการทำงาน จึงต้องทำการทดสอบการทำงานของโปรแกรมว่าถูกต้องเป็นไปตามที่ต้องการหรือไม่

2.1 การเขียนผังโปรแกรม (Flowchart)

กระบวนการพื้นฐาน มี 3 กระบวนการได้แก่

1. การรับข้อมูลเข้า (Input)
2. การประมวลผล (Process)
3. การแสดงผลลัพธ์ (Output)



เครื่องหมายสัญลักษณ์ที่ใช้ในการเขียนผังโปรแกรม

สัญลักษณ์	ความหมาย	ตัวอย่าง
	การเริ่มต้นหรือ สิ้นสุด	Start Stop
	กระบวนการทั่ว ๆ ไป เช่นการคำนวณ	sum = sum + x
	การเปรียบเทียบเงื่อนไข	N > 20 yes no
	การนำข้อมูลเข้า หรือ แสดงผลข้อมูลทาง อุปกรณ์แสดงผล	Read X

เครื่องหมายสัญลักษณ์ที่ใช้ในการเขียนผังโปรแกรม

สัญลักษณ์	ความหมาย	ตัวอย่าง
	การแสดงผลทาง เครื่องพิมพ์	Print N
	กระบวนการที่ได้นิยามไว้	Sort
	หน่วยความจำสำรอง	Master File

เครื่องหมายสัญลักษณ์ที่ใช้ในการเขียนผังโปรแกรม

สัญลักษณ์	ความหมาย	ตัวอย่าง
	ตัวเชื่อมต่อ	A A
	การเชื่อมต่อไปหน้าอื่น	1 1
	ลูกศรชี้ทิศทาง และ ลำดับการทำงาน	คำนวณพื้นที่

2.2 รหัสเทียม (Pseudo code)

รหัสเทียม หมายถึง

- คำอธิบายที่มีโครงสร้างและรูปแบบชัดเจน
 - แบบลำดับ แบบเลือก แบบวนซ้ำ
- ไม่ขึ้นกับภาษาคอมพิวเตอร์ใดภาษาหนึ่งโดยเฉพาะ
- สามารถเปลี่ยนไปเป็นภาษาคอมพิวเตอร์ได้ง่าย
- นิยมใช้ภาษาอังกฤษ
- ต้องบอกจุดเริ่มต้นและสิ้นสุด
 - เริ่มต้น คือ Start
 - สิ้นสุด คือ Stop

การเขียนผังโปรแกรม และรหัสเทียมแบบโครงสร้าง

แบ่งออกเป็น 3 โครงสร้างใหญ่ ๆ ดังนี้

A.1. โครงสร้างแบบเรียงลำดับ (Sequential Structure)

A.2. โครงสร้างแบบมีเงื่อนไข (Decision / Selection Structure)

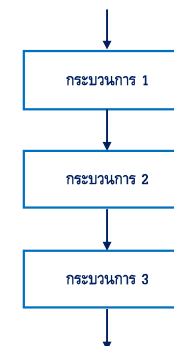
A.3. โครงสร้างแบบวนซ้ำ (Repetition Structure)

A.1 โครงสร้างแบบเรียงลำดับ (Sequential Structure)

- A.1.1 ผังโปรแกรมของโครงสร้างแบบเรียงลำดับ
- A.1.2 รหัสเทียมของโครงสร้างแบบเรียงลำดับ

A.1.1 ผังโปรแกรมของโครงสร้างแบบเรียงลำดับ

เป็นการจัดลำดับคำสั่งตามลำดับก่อนหลังสามารถเขียนเป็นรูปแบบโครงสร้าง



A.1.2 รหัสเทียมของโครงสร้างแบบเรียงลำดับ (Sequential Structure)

เขียนคำสั่งตามลำดับก่อนหลัง เช่น

DO

Process 1

Process 2

END

ตัวอย่างโจทย์โครงสร้างแบบเรียงลำดับ

จงเขียนผังโปรแกรมและรหัสเทียมสำหรับรับข้อมูลจำนวน ชั่วโมงการทำงานใน 1 สัปดาห์ และอัตราเงินได้ต่อชั่วโมงเพื่อ คำนวณรายได้อต่อสัปดาห์ และเงินได้หลังหักภาษี กำหนด อัตราภาษีเท่ากับ 10% ของเงินได้อต่อสัปดาห์

การวิเคราะห์โจทย์

วัตถุประสงค์หลัก เงินได้อต่อสัปดาห์ และเงินได้หลังหักภาษี

ผลลัพธ์ที่แสดง “Gross pay is เงินได้อต่อสัปดาห์”

“Net pay is เงินได้หลังหักภาษี”

การวิเคราะห์และการกำหนดคุณสมบัติของปัญหา

โดยทำการวิเคราะห์รายละเอียดต่าง ๆ ดังนี้

1. วัตถุประสงค์หลัก (Main purpose)

2. ผลลัพธ์ที่แสดง (Output requirement)

วัตถุประสงค์ของการแก้ปัญหา กำหนดรูปแบบการนำเสนอผลลัพธ์

3. ข้อมูลที่ต้องนำเข้า (Input Requirement)

ปัญหาบอกข้อมูลอะไรบ้าง กำหนดรูปแบบของข้อมูลที่ได้รับเข้า

4. ตัวแปรที่ใช้ (Variable)

5. วิธีการประมวลผล (Process)

กำหนดค่า กำหนดสูตรหรือวิธีการ กำหนดเงื่อนไขของการคำนวณ (ถ้ามี) เพื่อป้องกันไม่ให้โปรแกรมทำงานผิดพลาด เช่น กรณีการหารด้วยศูนย์ การวนซ้ำไม่รู้จบ

การวิเคราะห์โจทย์(ต่อ)

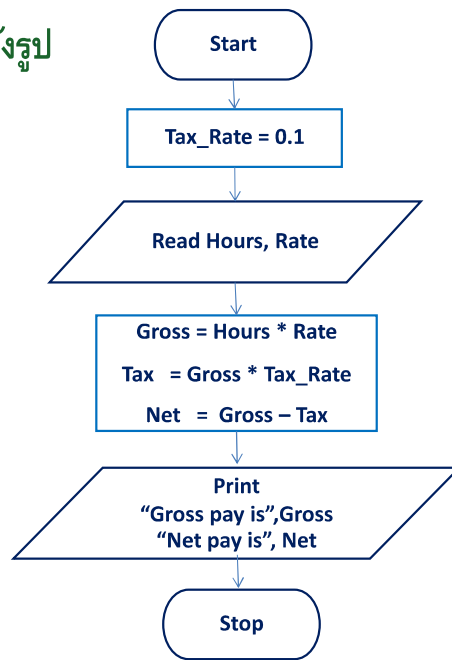
ข้อมูลเข้า	จำนวนชั่วโมงการทำงาน และ อัตราเงินได้ต่อชั่วโมง
ตัวแปรที่ใช้	Hours แทนจำนวนชั่วโมงการทำงานใน 1 สัปดาห์
	Rate แทนอัตราเงินได้ต่อชั่วโมง
	Tax_Rate แทนอัตราภาษีที่หัก
	Tax แทนเงินภาษีที่ต้องจ่าย
	Gross เงินได้ก่อนหักภาษี
	Net เงินได้หลังหักภาษี

การประมวลผล $Gross = Hours * Rate$

$Tax = Gross * Tax_Rate$

$Net = Gross - Tax$

ผังโปรแกรม ดังรูป



รหัสเทียม ดังต่อไปนี้

Start

Tax_Rate = 0.1

Read Hours, Rate

Gross = Hours * Rate

Tax = Gross * Tax_Rate

Net = Gross - Tax

Print "Gross pay is ", Gross, "Net pay is", Net

Stop

ตัวอย่างข้อมูลเข้า และผลลัพธ์

ข้อมูลเข้า	ผลลัพธ์
Hours = 100 Rate = 30	Gross pay is 3000 Net pay is 2700
Hours = 20 Rate = 27.5	Gross pay is 550 Net pay is 495

แบบฝึกหัด

จงเขียน Flowchart และ Pseudo code สำหรับวิเคราะห์ปัญหาการทอนเหรียญ 1,5 หรือ 10 โดยได้เงินทอนมา M บาท ดังตัวอย่างแสดงด้านล่าง

ตัวอย่าง

เงินทอน 17 บาท (กรณีที่ M = 17)

- เงินทอน เหรียญ 10 บาท 1 เหรียญ
- เงินทอน เหรียญ 5 บาท 1 เหรียญ 1 เหรียญ
- เงินทอน เหรียญ 1 บาท 2 เหรียญ

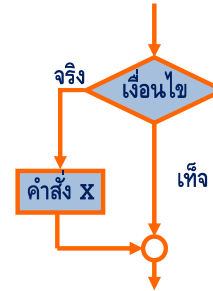
A.2 โครงสร้างแบบมีเงื่อนไข

เป็นโครงสร้างที่มีทางเลือกแบ่งเป็น 2 ประเภท

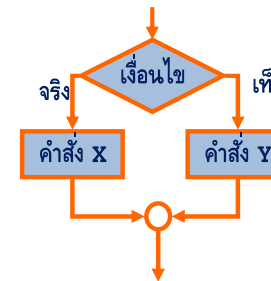
- A.2.1 โครงสร้างแบบมีเงื่อนไขให้เลือกกระทำโดยใช้ IF (Decision/Selection Structure)
 - A.2.1.1 ฟังก์ชันโปรแกรมของโครงสร้างแบบมีเงื่อนไขให้เลือกกระทำ 3 แบบ
 - A.2.1.2 รหัสเทียมของโครงสร้างแบบมีเงื่อนไขให้เลือกกระทำ 3 แบบ
- A.2.2 โครงสร้างแบบมีเงื่อนไขให้เลือกกระทำหลายทาง (CASE structure)
 - A.2.2.1 ฟังก์ชันโปรแกรมของโครงสร้างแบบมีเงื่อนไขให้เลือกกระทำหลายทาง
 - A.2.2.2 รหัสเทียมของโครงสร้างแบบมีเงื่อนไขให้เลือกกระทำหลายทาง

A.2.1.1 ฟังก์ชันโปรแกรมแบบมีเงื่อนไขให้เลือกกระทำโดยใช้ IF (Decision/Selection Structure)

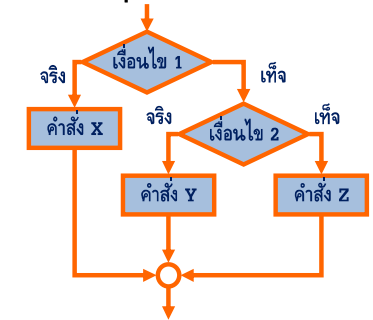
IF THEN



IF THEN ELSE



IF ELSE IF ELSE, Nested IF, Multiple IF



A.2.1.2a รหัสเทียมของโครงสร้างแบบมีเงื่อนไขแบบเลือกกระทำแบบ IF THEN

ต้องมีการทดสอบเงื่อนไขก่อน แล้วเลือกกระทำ ถ้าเงื่อนไขเป็นจริงจะทำ Process 1 ถ้าไม่จริงไม่ทำอะไร เช่น

IF condition THEN

Process 1

ENDIF

A.2.1.2b รหัสเทียมของโครงสร้างแบบมีเงื่อนไขแบบเลือกกระทำแบบ IF THEN ELSE

ต้องมีการทดสอบเงื่อนไขก่อน แล้วเลือกกระทำ ถ้าเงื่อนไขเป็นจริงจะทำ Process 1 ถ้าเงื่อนไขเป็นเท็จจะทำ Process 2 เช่น

IF condition THEN

Process 1

ELSE

Process 2

ENDIF

A.2.1.2c1 รหัสเทียมของโครงสร้างแบบมีเงื่อนไขแบบเลือก

กระทำแบบ IF ELSE IF ELSE

ต้องมีการทดสอบเงื่อนไขก่อน แล้วเลือกกระทำ ถ้าเงื่อนไขเป็นจริงจะทำ Process 1 ถ้าเงื่อนไขเป็นเท็จจะทดสอบเงื่อนไขถัดไป ถ้าจริงทำ Process 2 ถ้าไม่จริงเช็คเงื่อนไขอื่นอีกเรื่อย ๆ จนกว่าจะเจอเงื่อนไขที่จริง และทำ Process นั้นๆ หรือถ้าไม่เจอก็ทำ Process X ในเงื่อนไขสุดท้าย (ELSE) ก่อนจบการทำงาน เช่น

```
IF condition THEN
    Process 1
ELSE IF condition THEN
    Process 2
...
ELSE
    Process X
ENDIF
```

A.2.1.2c2 รหัสเทียมของโครงสร้างแบบมีเงื่อนไขแบบเลือก

กระทำแบบ Nested IF หรือ Multiple IF

```
IF condition THEN
    Process 1
    IF condition THEN
        Process 2
    ELSE
        Process 3
    ENDIF
ELSE
    Process 4
ENDIF

IF condition THEN
    Process 1
END IF
IF condition THEN
    Process 2
END IF
IF condition THEN
    Process 3
ELSE
    Process 4
END IF
```

ตัวอย่างโครงสร้างแบบมีเงื่อนไขให้เลือกกระทำโดยใช้ IF

จงเขียนผังโปรแกรมเพื่อรับเลขจำนวนเต็ม 2 จำนวน แล้วเปรียบเทียบว่าจำนวนไหนมากกว่า ถ้าเท่ากันให้บอกว่าเท่ากัน

การวิเคราะห์โจทย์

วัตถุประสงค์หลัก เปรียบเทียบเลข 2 จำนวน

ผลลัพธ์ที่แสดง The greater number is เลขจำนวนที่มากกว่า
Both numbers are equal

การวิเคราะห์โจทย์(ต่อ)

ข้อมูลเข้า เลขจำนวนเต็ม 2 จำนวน

ตัวแปรที่ใช้ num1 แทนเลขจำนวนที่ 1

num 2 แทนเลขจำนวนที่ 2

การประมวลผล ถ้า num1 > num2 พิมพ์

“The greater number is”, num1

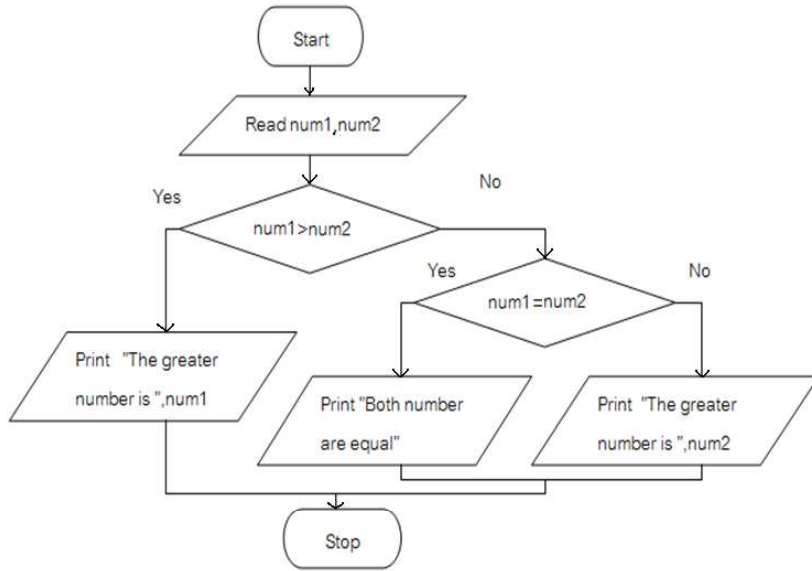
ถ้า num1 < num2 พิมพ์

“The greater number is”, num2

ถ้า num1 = num2 พิมพ์

“Both numbers are equal”

ผังโปรแกรม ดังรูป



รหัสเทียม ดังต่อไปนี้

START

READ num1,num2

IF num1 > num2 THEN

PRINT "The greater number is", num1

ELSE

IF num1 = num2

PRINT "Both number are equal"

ELSE

PRINT "The greater number is", num2

ENDIF

END IF

STOP

ตัวอย่างข้อมูลเข้า และผลลัพธ์

ข้อมูลเข้า	ผลลัพธ์
num1 = 5 num2 = 12	The greater number is 12
num1 = 20 num2 = 2	The greater number is 20
num1 = 134 num2 = 134	Both number are equal

แบบฝึกหัด

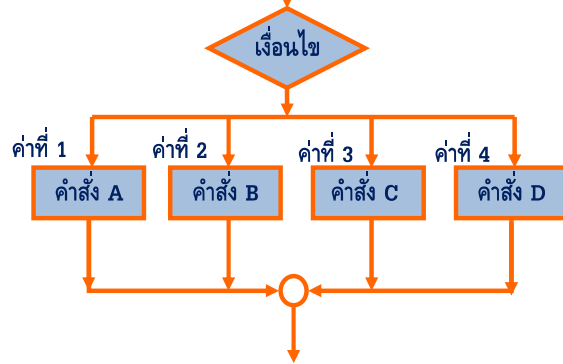
- ให้ทำการเปรียบเทียบตัวเลข 3 ค่า คือ ตัวเลข **A, B, C** แล้วให้แสดงค่าออกมาว่าตัวไหนมีค่ามากที่สุด ตัวไหนมีค่าน้อยที่สุด ตัวไหนเป็นค่ากลาง
- จงออกแบบโปรแกรมตัดเกรดของคะแนนนักศึกษา **1** คน และแสดงผลเกรด โดยมีเงื่อนไขดังนี้

คะแนน	เกรด
0-50	F
51-60	D
61-70	C
71-80	B
81-100	A

— ทำการวิเคราะห์โจทย์ เขียน Flowchart และ Pseudo code

A.2.2.1 ผังโปรแกรมของโครงสร้างแบบมีเงื่อนไขให้เลือกกระทำหลายทาง (CASE structure)

ถ้าเงื่อนไขไหนเป็นจริงจะทำตามขบวนการหนึ่งในหลาย ๆ ทางเลือก สามารถเขียนเป็นโครงสร้างดังรูป



A.2.2.2 รหัสเทียมแบบมีทางเลือกกระทำหลายทาง (CASE structure)

- CASE expression OF variable
case 1 : sequence 1
case 2 : sequence 2
...
case n : sequence n
OTHER:
default sequence
ENDCASE

ตัวอย่างโครงสร้างแบบมีเงื่อนไขให้เลือกกระทำหลายทาง

จงเขียนผังโปรแกรมเพื่อหาชั้นปีของนักศึกษา

การวิเคราะห์โจทย์

วัตถุประสงค์หลัก ชั้นปีของนักศึกษา

ผลลัพธ์ที่แสดง ชั้นปี คือ

ข้อมูลเข้า ตัวเลขปีของนักศึกษา

ตัวแปรที่ใช้ Y แทนปีของนักศึกษา

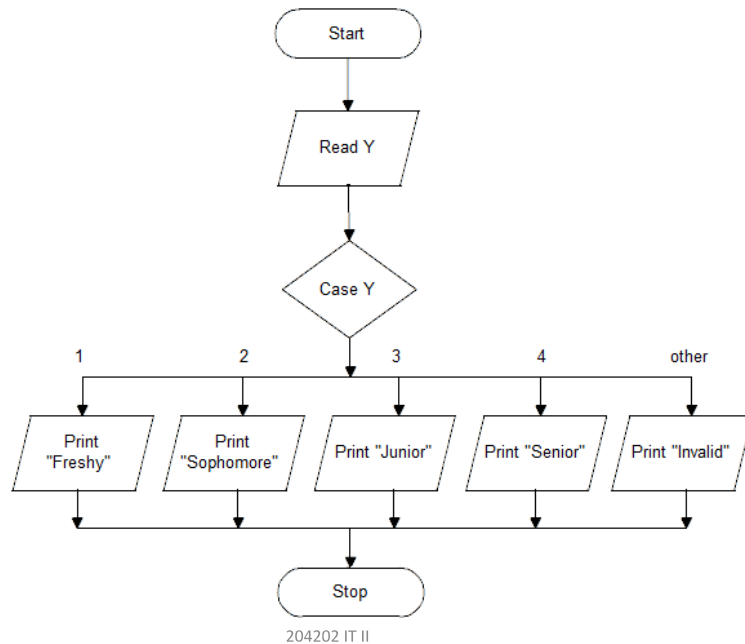
การวิเคราะห์โจทย์(ต่อ)

การประมวลผล

CASE Y

- 1 พิมพ์ "Freshy"
 - 2 พิมพ์ "Sophomore"
 - 3 พิมพ์ "Junior"
 - 4 พิมพ์ "Senior"
- อื่น ๆ พิมพ์ Invalid

ผังโปรแกรม ดังรูป



รหัสเทียม ดังต่อไปนี้

```
START
READ Y
CASE OF Y
  case "1" :
    PRINT "Freshy"
  case "2" :
    PRINT "Sophomore"
  case "3" :
    PRINT "Junior"
  case "4" :
    PRINT "Senior"
  Other
  default : PRINT "Invalid"
ENDCASE
STOP
```

ตัวอย่างข้อมูลเข้า และผลลัพธ์

ข้อมูลเข้า	ผลลัพธ์
1	Freshy
4	Senior
6	Invalid
a	Invalid

แบบฝึกหัด

- แปลงการออกแบบโปรแกรมตัดเกรดให้เป็นจาก IF ,ELSE IF, ELSE ให้เป็นแบบ CASE
- จงออกแบบโปรแกรมในการให้ผู้ใช้เลือกชนิดสินค้า เพียง 1 ชนิด และได้จำนวนสินค้าชนิดนั้นที่จะซื้อ เพื่อแสดงมูลค่ารวมของสินค้าที่ผู้ใช้เลือก ดังนี้ (โดยตัวแปรที่รับค่าจำนวนสินค้า คือ N)
 - 1. Note book เครื่อง ละ 15000 บาท
 - 2. PC ชุด ละ 20000 บาท
 - 3. หูฟัง ชุดละ 300 บาท
 - 4. กระดาษ รีมละ 250 บาท
 - หากเป็นสินค้าชนิดอื่น ให้แสดงข้อความว่า ไม่มีขาย