

## แบบฝึกหัด

1. ข้อมูลส่วนตัวของลูกจ้างแต่ละคนในบริษัทการค้าแห่งหนึ่ง บันทึกไว้ในบัตร 80 คอลัมน์ โดยมีรูปแบบดังนี้

### คอลัมน์

1	รหัสประจำแผนก
2-9	หมายเลขประจำตัว
10	ว่าง
11-35	ชื่อ นามสกุล
36	สถานภาพสมรส (1 = โสด, 2 = แต่งงาน, 3 = หย่าร้าง, 4 = ม้าย)
37	เพศ (1 = ชาย, 2 = หญิง)
38-39	วัน เดือน } ที่เกิด ปี
40-41	
42-43	
44-45	จำนวนบุตร
46-55	รายได้ต่อเดือนทศนิยม 2 ตำแหน่ง
56-80	ว่าง

a) ให้นักศึกษาเขียน file และ record description entires จากข้อมูลข้างต้นให้ถูกต้อง

b) ให้สมมติข้อมูลตัวอย่างตามรูปแบบดังกล่าวจำนวน 5 เรคคอร์ด

2. a) จงบอกจำนวนตัวอักษรในเนื้อที่หน่วยความจำที่นิยามข้างล่างนี้

05 A.

10 A-1 PIC XX.

10 A-2 PIC 999.

03 B REDEFINES A.

10 B-1 PIC 9.

10 B-2 PIC X.

10 B-3 PIC XXX.

05 C REDEFINES B PIC X(5).

h) จงวาดรูปแสดงรายละเอียดของเนื้อที่หน่วยความจำที่นิยามข้างล่างนี้

01 REC-A.

05 A1 PIC XX.

05 A2 PIC 9999.

05 A3 PIC 9.

05 A4 PIC XXX.

01 REC-B REDEFINES REC.A.

05 B1 PIC X(5).

05 B2 PIC 9(5).

c) จงวาดรูปแสดงรายละเอียดของ record description entries ข้างล่างนี้

01 SAMPLE.

05 A.

10 B PIC X.

10 c PIC x.

05 D.

10 E PIC X.

10 F.

15 G PIC X.

15 H PIC X.

01 I.

15 J PIC x .

15 K PIC X.

จากนั้นให้บอกความหมายของสองคำสั่งข้างล่างนี้

66 GROUP-CG RENAMES C THRU G.

66 BH RENAMES A THRU F.

3. a) จงเขียน data description entries จากรูปข้างล่างนี้ซึ่งสมมติว่าเป็นข้อมูลที่เราจะเจาะในบัตร 80 คอลัมน์หนึ่งใบ

STUDENT-RECORD								
ENROLLMENT-DATE			STUDENT-NO	NAME-ADDRESS				
E-MONTH	E-DAY	E-YEAR		STREET	CITY	STATE	ZIP	
2 digits	2 digits	2 digits	8 digits.	15 char.	10 char.	2 char.	5 digits	spaces

b) เรคคอร์ดในข้อ (a) แบ่งออกเป็นกี่ระดับ

c) และมีจำนวนข้อมูลเดี่ยว (elementary data item) กี่ตัว, อะไรบ้าง

4. รูปภาพที่แสดงข้างล่างนี้เป็นหนึ่งเรคคอร์ดชื่อ BIGFIELD ตัวเลข 1 ถึง 13 แสดงตำแหน่งของตัวอักษร, เรคคอร์ดนี้ประกอบด้วยตัวอักษรทั้งหมด 13 ตัว ต้องการนำข้อมูลในตำแหน่งต่าง ๆ มาใช้ โดยที่โครงสร้างปัจจุบันของเรคคอร์ดนี้ยังคงเป็นเช่นเดิม จะมีวิธีเขียนคำสั่งอย่างไร

BIGFIELD												
GROUP-A							GROUP-B					
AA		AB			AC		BA		BB		BC	
1	2	3	4	5	6	7	8	9	10	11	12	13

a) อ้างถึง 1, 2, 3, 4 โดยใช้ชื่อเดี่ยว

b) อ้างถึง 5, 6, 7 โดยใช้ชื่อเดี่ยว

c) อ้างถึง 8, 9, 10 โดยใช้ชื่อเดี่ยว

d) อ้างถึง 11, 12, 13 โดยใช้ชื่อเดี่ยว

e) อ้างถึง 1, 2, 3, 4, 5, 6, 7 โดยใช้ชื่อเดี่ยว

f) อ้างถึง 8, 9, 10, 11, 12, 13 โดยใช้ชื่อเดี่ยว

g) อ้างถึง 3, 4, 5, 6, 7, 8, 9, 10 โดยใช้ชื่อเดี่ยว

5. โรงแรมเซอรอตันเป็นระบบโรงแรมซึ่งมีผู้บริหารชุดเดียวกัน ตั้งอยู่ใน 50 ประเทศ ในแต่ละประเทศแบ่งออกเป็น 30 เขต และในแต่ละเขตมีอยู่ 10 โรงแรมด้วยกัน แต่ละโรงแรมมีห้องพักอยู่สองชนิดคือ ชนิดเตียงเดี่ยวและเตียงคู่ จงเขียนคำสั่งสร้างตารางหนึ่งตารางใน Working-storage section เพื่อเก็บข้อมูลทั้งหมดของระบบโรงแรมนี้

6. โรงงานอุตสาหกรรมแห่งหนึ่งแบ่งงานออกเป็น 7 แผนก และอัตราค่าจ้างแบ่งออกเป็น 3 ประเภท คืองานกลางวัน, งานกลางคืน และงานวันหยุด สำหรับอัตราค่าจ้างขึ้นอยู่กับหมายเลขงาน (แผนก) และช่วงเวลาการทำงานดังนี้

JOB-NO	DAY-RATE	NIGHT-RATE	WEEKEND-RATE
01	7.25	8.30	8.15
02	6.15	7.05	6.95
04	7.45	8.50	8.25
05	8.55	9.75	9.50
0h	7.50	X.65	8.45
08	9.25	10.50	10.25
10	8.95	10.00	9.75

จงเขียนคำสั่งสร้างตารางในหน่วยความจำของเครื่องคอมพิวเตอร์เพื่อให้เก็บตัวเลขทั้งหมดข้างต้น

7. a) จากคำสั่งข้างล่างนี้ จงวาดรูปแสดงลักษณะของข้อมูลภายในหน่วยความจำของคอมพิวเตอร์

01 ENROLLMENT-TABLE.

02 UNIVERSITY-NUMBER OCCURS 3 TIMES.

03 YEAR OCCURS 5 TIMES.

04 SEX OCCURS 2 TIMES PICTURE 9(5).

b) นักศึกษาชายชั้นปีที่ 3 ศึกษาในมหาวิทยาลัยหมายเลข 2 จะแทนด้วย data-name อะไร ?

c) แผนกทะเบียนของมหาวิทยาลัยต้องการทราบจำนวนนักศึกษาทั้งหมดที่ลงทะเบียนเรียนภาคฤดูร้อน ปีการศึกษา 2528 ว่ามีจำนวนกี่คน ถ้าเตรียมเนื้อที่เพื่อเก็บข้อมูลดังนี้

77 TOTAL-ENROLLMENT PICTURE 9(6)

01 ENROLLMENT-TABLE.

02 FACULTY OCCURS 7 TIMES

03 CLASS-LEVEL OCCURS 4 TIMES.

04 STUDENT OCCURS 2 TIMES PICTURE 9(4)

จงเขียนคำสั่งเพื่อหาผลรวมทั้งหมดข้างต้น (ทั้งนักศึกษาชาย-หญิง, ทุกชั้นปีและทุกคณะ)

8. จากคำสั่งข้างล่างนี้ ให้วาดรูปแสดงรายละเอียดของเนื้อที่ชื่อ NAME-Z และ เนื้อที่ & NAME-1 ที่มีการจัดระเบียบใหม่ และให้อธิบายความหมายด้วย

02 NAME-2.

03 SALARY PICTURE XXX.

03 SO-SEC-NO PICTURE X(9).

03 MONTH PICTURE XX.

02 NAME-1 REDEFINES NAME-2.

03 MAN-NO PICTURE X(6).

03 WAGE PICTURE 999V999.

03 YEAR PICTURE XX.

9. กำหนดให้

01 SAMPLE-ITEM.

05 A PICTURE 9(4)V99.

05 B REDEFINES A.

10 C PICTURE 9(2)V9.

10 D PICTURE 9(3).

หลังจากที่เครื่อง execute คำสั่ง MOVE 12.34 TO A. จงบอกมูลค่าของ C และ D

10. จงวาดรูปแสดงเนื้อที่ในหน่วยความจำของคอมพิวเตอร์ เมื่อนิยามคำสั่งข้างล่างนี้

01 FREIGHT-CHARGES-TABLE.

05 WEIGHT-RANGE OCCURS 3 TIMES.

10 WEIGHT-LIMIT PICTURE S9(3).

10 DESTINATION OCCURS 3 TIMES.

15 FREIGHT-CHARGE PICTURE S9(3)V99.

11. จงเขียนคำสั่งใน Working-storage section สร้างตารางเก็บปริมาณขายและอัตราส่วนลดไว้ในหน่วยความจำของคอมพิวเตอร์ ดังนี้ ถ้าปริมาณขายไม่เกิน 10 ไม่มีส่วนลด,

ปริมาณขายตั้งแต่ 11 ถึง 25 มีส่วนลด 20%, ปริมาณขาย 26 ถึง 100 มีส่วนลด 25% และ  
 ต่อ ๆ ไป ตามตัวเลขข้างล่างนี้

ปริมาณขาย	อัตราส่วนลด
10	.000
25	.020
100	.025
250	.040
500	.080
1000	.105
2000	.130

### SIZE clause

มีรูปแบบดังนี้

**SIZE IS integer-1** { CHARACTERS  
 DIGITS }

clause นี้ใช้สำหรับกำหนดจำนวนตำแหน่งตัวอักษรของ item นั้นภายในหน่วยความจำ, integer-1 หมายถึงจำนวนตัวอักษรของ alphabetic หรือ alphanumeric item หรือ จำนวนตัวเลข สำหรับ numeric item ส่วนคำว่า CHARACTERS และ DIGITS เป็น optional words จะไม่เขียนก็ได้

Numeric item. ต้องเป็น ข้อมูลเดี่ยว และมีความยาวสูงสุดไม่เกิน 12 ตัว, ขนาดสูงสุดของฟิลด์เท่ากับ 4098 characters ถ้าเป็น alphanumeric ให้นับเครื่องหมายและจุดทศนิยมด้วย แต่ถ้าเป็นจุดทศนิยมสมมติไม่ต้องนับ

ถ้า entry นั้นเป็น ข้อมูลเดี่ยว ต้องมี SIZE clause หรือ PICTURE clause แต่ถ้าเป็น ข้อมูลกลุ่ม อาจจะไม่มีทั้งสอง clauses นี้ แต่ถ้ามี SIZE clause ขนาดของมันต้องเท่ากับผลรวมของขนาดของข้อมูลเดี่ยว ทั้งหมดที่ประกอบขึ้นเป็นกลุ่ม (group) นั้น, ถ้า entry ตัวหนึ่ง, บรรยาย ทั้ง SIZE และ PICTURE clause, ต้องมีความหมายตรงกัน (they must agree)

ตัวอย่าง

03 A PICTURE 999 SIZE 3.

03 B PICTURE XX SIZE 2 CHARACTERS.

**CLASS clause**

มีรูปแบบดังนี้

	<u>NUMERIC</u>	
CLASS IS	<u>ALPHABETIC</u>	
	<u>ALPHANUMERIC</u>	
	I <u>AN</u>	

จากที่เราทราบแล้วว่า data items แบ่งออกได้เป็นสามชนิดคือ numeric, alphabetic และ alphanumeric items

NUMERIC item นั้นประกอบด้วยเลข 0 ถึง 9 เท่านั้น, อาจจะมีจุดทศนิยมที่สมมติขึ้น อาจจะมีเครื่องหมายบวกหรือลบกำกับก็ได้ แต่จะมีเฉพาะตัวเลขเท่านั้นที่อยู่ในตำแหน่งของ ตัวอักษรในหน่วยความจำ

ALPHABETIC item นั้น อาจจะใช้สลับที่กับ alphanumeric items ได้

ALPHANUMERIC item นั้นประกอบด้วยตัวอักษรอะไรก็ได้จาก COBOL characters AN เป็นคำย่อของ ALPHANUMERIC

CLASS clause สำหรับ item แต่ละตัว, จะเขียนที่ระดับใดก็ได้, สำหรับ CLASS clause ของข้อมูลที่เป็นระดับกลุ่มจะมีผล (applied) ถึงข้อมูลเดี่ยวแต่ละตัวในกลุ่มนั้นด้วย และจะต้องไม่ขัดแย้งกับ CLASS ใดๆ ที่กำหนดในข้อมูลเดี่ยวเหล่านั้น, ALPHABETIC หรือ NUMERIC items ภายใน ALPHANUMERIC group หนึ่งๆ ไม่ถือว่าขัดแย้งกัน ระหว่างคำว่า ALPHABETIC และ ALPHANUMERIC ในความหมายของคอมไพเลอร์ถือว่าไม่แตกต่างกัน

entry แต่ละตัวซึ่งมี PICTURE clause ไม่จำเป็นต้องมี CLASS clause

แต่ในกรณีที่มีทั้งสองอย่าง จะต้องเข้ากันได้ และคำว่า CLASS อาจจะไม่เขียนไว้ก็ได้

**ตัวอย่าง**

03 C PIC 9(3)V9(2) NUMERIC.

USAGE clause

มีรูปแบบดังนี้

USAGE IS	{	<u>COMP</u> <u>COMPUTATIONAL</u> <u>COMP-1</u> <u>COMPUTATIONAL-1</u> <u>DISPLAY</u>	}
----------	---	--	---

clause นี้ใช้กับ item แต่ละตัวที่ระดับใดก็ได้ ถ้าใช้กับข้อมูลกลุ่ม clause นี้ต้องไม่ขัดแย้งกับ entry ที่อยู่ในระดับต่ำกว่ามัน

USAGE เป็น optional word จะไม่เขียนไว้ก็ได้, ถ้าไม่เขียน clause นี้เครื่องจะถือว่า มี USAGE เป็น DISPLAY\*

item ซึ่งใช้ COMPUTATIONAL ต้องเป็นตัวเลขส่วน CLASS clause ที่ใช้คู่กันอาจจะไม่เขียนก็ได้, numeric item ที่กำหนดโดย COMPUTATIONAL อาจจะใช้สำหรับแสดงผล (display) ได้

COMPUTATIONAL-I ใช้บรรยายลักษณะข้อมูลเดี่ยวแต่ละตัวใน working-storage ที่เป็น subscript item เท่านั้น, option นี้อาจใช้กับข้อมูลอิสระได้ (item ที่ใช้เลขบอกระดับ 77) มูลค่าของ item นี้จะเก็บในหน่วยความจำในลักษณะเลขฐานสอง และใช้เนื้อที่หนึ่ง word เนื่องจาก item นี้ถูกใช้เป็น subscripting มูลค่าของมันจึงมากกว่า  $2^{15}-1$  ไม่ได้ และจะมีความหมายเฉพาะถ้าใช้คู่กับ VALUE clause

item ที่บรรยายด้วย COMPUTATIONAL-I อาจย้าย (move) ไปยัง numeric item ซึ่งไม่ได้บรรยายด้วย COMPUTATIONAL-I ได้ ในกรณีนี้มูลค่าของ item นั้นจะถูกเปลี่ยนเป็นรูปแบบ B C D ในทางตรงกันข้ามเราก็อาจย้ายข้อมูลจากที่ไม่ได้ใช้ COMPUTATIONAL-I clause ไปยังเนื้อที่ซึ่งใช้ COMPUTATIONAL-I clause ก็ได้ในกรณีนี้ก็คือการเปลี่ยนรูปมูลค่าของ item นั้นจาก BCD format ให้เป็น Binary format นั่นเอง

COMPUTATIONAL-I items สองจำนวนอาจจะเปรียบเทียบกันด้วย a simple relations operator ได้ หรือ item ตัวหนึ่งเป็น COMPUTATIONAL-I เปรียบเทียบกับ item อีกตัวหนึ่งซึ่งเป็น numeric literal ได้, เราอาจจะตรวจสอบว่า COMPUTATIONAL-I item มีค่าเป็นบวก, ลบ หรือศูนย์ ได้, items เหล่านี้จะอยู่ในคำสั่งบวกหรือลบได้ก็ต่อเมื่อตัวกระทำในขณะนั้นเป็น COMPUTATIONAL-I items ทุกตัว, item นี้ แต่ละตัวอาจจะมีค่าเพิ่มขึ้น หรือลดลงก็ได้

ตัวอย่าง

WORKING-STORAGE SECTION.

77 I PICTURE 99 VALUE ZEROES USAGE COMPUTATIONAL-I.

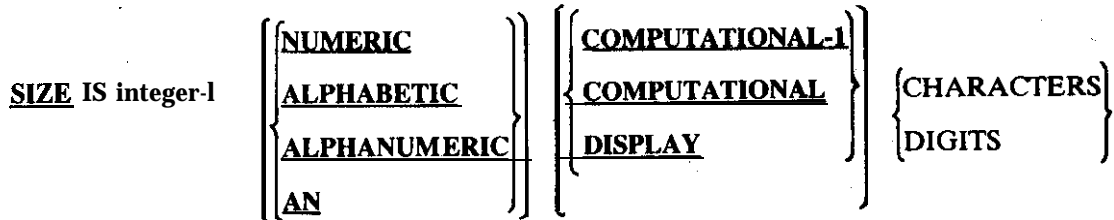
77 J PICTURE 99 VALUE ZEROES USAGE COMPUTATIONAL-I.

---

\* ให้แสดงผลด้วย



**Combining SIZE, CLASS และ USAGE**  
มีรูปแบบดังนี้



ในที่นี้ CLASS และ USAGE เป็น optional COBOL words ถ้า SIZE clause ตามด้วย CLASS และ USAGE clauses ไม่มี optional words ผลลัพธ์จะเป็น single clause ที่ประกอบด้วย รายละเอียดของทั้งสามชุด

การรวมกันชนิดนี้ อาจจะเป็น SIZE กับ CLASS, SIZE กับ USAGE หรือ SIZE, CLASS และ USAGE ทั้งสามอย่างเข้าด้วยกัน กฎเกณฑ์สำหรับการรวมกันนี้เหมือนกับกฎเกณฑ์เมื่อแยกกันนั่นเอง ส่วนลำดับที่จะเรียงอันไหนก่อนอันไหนหลังก็ได้

**ตัวอย่าง 1**

clause	ภายในหน่วยความจำ
SIZE IS 13 ALPHANUMERIC DISPLAY CHARACTERS	ERROR-MESSAGE
SIZE IS 3 NUMERIC COMPUTATIONAL DIGITS	123
SIZE IS 6 NUMERIC COMPUTATIONAL DIGITS	169243
SIZE IS 9ALPHANUMERIC DISPLAY CHARACTERS	STOCK-456
SIZE IS 7 AN DISPLAY CHARACTERS	- \$**4.80

**ตัวอย่าง 2**

- 03 A.
- 04 A-1 PIC X(3).
- 04 A-2 PIC A(4).
- 03 B REDEFINES A.
- 04 B-1 PIC 9(2) USAGE COMP.
- 04 B-2 PIC X(4).
- 04 B-3 PIC X(1).

### ตัวอย่าง 3

77 J PICTURE 99 VALUE ZEROS COMP-1.

77 J PICTURE 99 VALUE ZEROS COMP-1.

77 K PICTURE 99 VALUE ZEROS COMP-1.

77 TEMP PICTURE X(80) VALUE SPACES.

#### **BLANK WHEN ZERO clause**

clause นี้เป็นการกำหนดให้ item 1 ตัว สำหรับทุกตัวอักษรที่มีค่าเป็นศูนย์ให้แทนด้วยตัวอักษร blank

มีรูปแบบดังนี้

#### **BLANK WHEN ZERO**

เมื่อใช้ **BLANK WHEN ZERO** ถ้า item ตัวนั้นมีค่าเป็นศูนย์ มันจะมีแต่ตัวอักษร blanks เท่านั้น

**BLANK WHEN ZERO** clause กำหนดได้เฉพาะกับข้อมูลเดี่ยว (elementary item) เท่านั้น และต้องเป็น numeric edited item หรือ numeric item เมื่อใช้ clause นี้ กับ item ที่มี picture เป็น numeric หลังจากนั้น item นี้จะเป็น numeric edited

clause นี้ จะไปใช้กับเลขบอกระดับ 66 หรือเลขบอกระดับ 88 ไม่ได้

### ตัวอย่าง

03 TAB PIC 9(6) BLANK WHEN ZERO.

ถ้า TAB มีค่าเป็น 

0	1	0	2	3	0
---	---	---	---	---	---

 จะได้ TAB มีค่าเป็น 

1	2	3
---	---	---

#### **JUSTIFIED clause**

clause นี้ใช้สำหรับกำหนดตำแหน่งของข้อมูลภายใน receiving alphabetic item หรือ alphanumeric data item

มีรูปแบบดังนี้

{	<b>JUSTIFIED</b>	}	RIGHT
{	<b>JUST</b>	}	

โดยปกติ กฎสำหรับตำแหน่งของข้อมูลภายใน receiving alphanumeric item หรือ alphabetic item คือ

ข้อมูลถูกจัดใน receiving item เริ่มจากตำแหน่งของตัวอักขระซ้ายมือสุดภายในฟิลด์ของ receiving ส่วนตำแหน่งของตัวอักขระทางขวามือที่ไม่ใช่เครื่องจะใส่ blank ให้ ถ้ามีการตัดส่วนเกินทิ้ง เครื่องจะตัดทางขวามือ

ส่วน JUSTIFIED clause มีผลทำให้ ตำแหน่งของข้อมูลใน receiving field เป็นดังนี้

1. เมื่อข้อมูลของ receiving มีการกำหนด JUSTIFIED clause และข้อมูลที่ส่งมามีขนาดมากกว่าเนื้อที่ของ receiving เครื่องจะตัดตัวอักขระทางซ้ายมือทิ้ง

2. เมื่อข้อมูลของ receiving มีการกำหนดด้วย JUSTIFIED clause และมีขนาดใหญ่กว่าข้อมูลที่ส่งมา ข้อมูลนั้นจะถูกจัดในตำแหน่งของตัวอักขระขวามือสุดในเนื้อที่นั้นก่อนเนื้อที่ซึ่งไม่มีตัวอักขระทางซ้ายมือเครื่องจะใส่ blank ใหม่

JUSTIFIED clause ใช้ได้เฉพาะกับข้อมูลเดียว และจะเอาไปใช้กับเลขบอกระดับ 66 และ เลขบอกระดับ 88 ไม่ได้

ตัวอย่าง

03 ADDRESS PICTURE X(30) JUSTIFIED RIGHT.

**แบบฝึกหัด**

เมื่อคอมพิวเตอร์ execute คำสั่ง MOVE 'ABCDE' TO SAMPLE-DATA. จงบอกผลลัพธ์ของ SAMPLE-DATA เมื่อนิยามด้วย picture clause ต่อไปนี้

- PICTURE X(3)
- PICTURE X(3) JUSTIFIED RIGHT
- PICTURE XX/XX
- PICTURE X(8) JUSTIFIED RIGHT
- PICTURE X(8)

### Working-storage section

ขณะที่มีการ execute object program, ผลลัพธ์ที่เกิดขึ้นในระหว่างนั้น และข้อเท็จจริงอื่น ๆ ที่จำเป็นต้องเก็บไว้ก่อนในหน่วยความจำ เพื่อทำงานในขั้นต่อไป หรือก่อนจะนำออกมาจากหน่วยความจำ (output) เนื้อที่เก็บเหล่านี้ คือ working-storage items โดยแต่ละ item ต้องบรรยายรายละเอียดใน record description entry

item ใน section นี้แบ่งออกเป็นสองชนิด คือ ข้อมูลอิสระ (independent item) และ ข้อมูลกลุ่ม (group item) โดยต้องเขียน section header ก่อนที่มาร์จิ้น A แล้วตามด้วย entries ของข้อมูลอิสระ และ entries ของข้อมูลกลุ่มตามลำดับ โดยมีรูปแบบเหมือนกับที่กล่าวมาแล้ว ใน file section แต่คอมพิวเตอร์บางเครื่องเช่น VAX จะเขียนข้อมูลกลุ่มและข้อมูลอิสระอันไหนก่อนหลังก็ได้

### ตัวอย่าง 1

```
WORKING-STORAGE SECTION.
```

```
77 TABLE-KEY PICTURE S99.
```

```
01 HEADING-1
```

```
01 HEADING-2.
```

### ตัวอย่าง 2

```
WORKING-STORAGE SECTION.
```

```
01 DAYS.
```

```
05 FILLER PICTURE X(9) VALUE "MONDAY".
```

```
05 FILLER PICTURE X(9) VALUE "TUESDAY".
```

```
05 FILLER PICTURE X(9) VALUE "WEDNESDAY".
```

```
05 FILLER PICTURE X(9) VALUE "THURSDAY".
```

```
05 FILLER PICTURE X(9) VALUE "FRIDAY".
```

```
05 FILLER PICTURE X(9) VALUE "SATURDAY".
```

```
05 FILLER PICTURE X(9) VALUE "SUNDAY".
```

```
01 DAYS-TABLE REDEFINES DAYS.
```

```
05 DAY-NAME PICTURE X(9) OCCURS 7.
```

จากคำสั่งทั้งหมดข้างต้นมีความหมายดังนี้

DAYS เป็น item ใช้เนื้อที่ 63 ตำแหน่ง แบ่งออกเป็น 7 ฟیلด์ แต่ละฟیلด์ใช้เนื้อที่ 9 คอลัมน์ มีค่า (content) เป็นชื่อของแต่ละวันในหนึ่งสัปดาห์ ส่วน DAYS-TABLE มีการ redefines เนื้อที่ข้างต้น และแบ่งออกเป็น 7 ฟیلด์ แต่ละฟیلด์ใช้เนื้อที่ 9 คอลัมน์ ดังนี้

คำสั่ง            MOVE DAY-NAME (3) TO PRINT-NAME.

หมายถึง        MOVE "WEDNESDAY" TO PRINT-NAME.

และคำสั่ง       MOVE 7 TO SUB.

                      MOVE DAY-NAME (SUB) TO PRINT-NAME.

หมายถึง        MOVE "SUNDAY" TO PRINT-NAME.

#### **Constant section**

ชื่อที่กำหนดให้มีมูลค่าเป็นค่าคงที่ และจะต้องไม่มีการเปลี่ยนแปลงค่าระหว่างที่มีการ execute object program ให้บรรยายรายละเอียดใน section นี้

item ใน section นี้แบ่งออกเป็นสองชนิด คือ ข้อมูลอิสระและข้อมูลกลุ่มสำหรับความหมาย รูปแบบและการใช้เหมือนกับ item ใน working-storage section ทุกอย่าง

แต่ส่วนใหญ่แล้วเขาไม่นิยมเขียน constant section เพราะว่ามันสามารถเขียนไว้ใน working-storage section ได้อยู่แล้ว และข้อจำกัดของกฎเกณฑ์มีน้อยกว่า

## แบบฝึกหัด

### 1 . ถูกหรือผิด

- a) Each COBOL program must contain five divisions.
- b) The first division is the ENVIRONMENT DIVISION.
- c) The DATA DIVISION immediately follows the ENVIRONMENT DIVISION.
- d) At present the shortest division will be the PROCEDURE DIVISION.
- e) Only a division header is required for the ENVIRONMENT DIVISION.
- f) The IDENTIFICATION DIVISION must contain one entry other than the division header.
- g) In COBOL as in English, a period is followed by a space.
- h) The program name contained in the PROGRAM-ID paragraph may have a maximum length of sixteen characters
- i) A program which uses six distinct data-names in its PROCEDURE DIVISION must contain, at present, six corresponding 01 entries in the DATA DIVISION.
- j) At present, 01 entries in the DATA DIVISION are found in the WORKING-STORAGE SECTION.
- k) The assumed decimal position in a PICTURE is indicated by a D.
- l) V is a code letter that means numeric data.
- m) X is a code letter which means alphameric data
- n) Y is a code symbol that means alphabetic data.
- o) PICTURE specifications may frequently be abbreviated.

- 2 กำหนดให้ NAME-1, NAME-2, NAME-3 และ NAME-4 เป็น independent data items ถามว่า items ทั้งหมดข้างต้นซึ่งเรา defined ไว้ข้างล่างนี้ถูกต้องหรือไม่? และถ้าผิดให้บอกเหตุผลด้วย

กอดัมน์

8 9 10 11 12

a)	7	7		N	A	M	E	-	1			
b)		7	7		N	A	M	E	-	2		
c)			7	7		N	A	M	E	-	3	
d)				7	7		N	A	M	E	-	4
e)	7	7			N	A	M	E	-	4		
f)	0	1		N	A	M	E	-	4			
g)	0	1			N	A	M	E	-	4		

3. a) จงบอกที่ผิดและเหตุผลในคำสั่งข้างล่างนี้

00 GROUP-FIELD.

04 W-PART PICTURE 99.

04 W-TYPE.

04 W-PART PICTURE A.

50 W-NAME PICTURE X(20)

b) คำต่อไปนี้ มีชื่อเรียกเฉพาะว่าอะไร ?

SPACES, ZEROS, QUOTES

c) data items ต่อไปนี้ อะไรเป็นข้อมูลกลุ่ม และอะไรเป็นข้อมูลเดี่ยว

01 DATA-RECORD PICTURE X(10).

03 MONTH-TOTAL.

01 MASTEIR-RECORD.

4. จงเขียนส่วนหนึ่งของโปรแกรมใน Working-storage section สร้างตารางซึ่งประกอบด้วย เดือนทั้ง 12 เดือนภายใน 1 ปี เพื่อที่ไว้ใน Procedure division เมื่อเราเรียกว่า data-name

MONTH (1)                      จะได้มูลค่า                      "JANUARY"

MONTH (2)                      จะได้มูลค่า                      "FEBRUARY"

MONTH (3)                      จะได้มูลค่า                      "MARCH"

MONTH (12)                      จะได้มูลค่า                      "DECEMBER" ตามลำดับ

5. รายละเอียดของ record description entry ชุดหนึ่งเรากำหนดไว้ดังนี้

01 SALE-RECORD

02 FIELD-A

03 FIELD-B

04 FIELD-C

04 FIELD-D

03 FIELD-E

02 FIELD-F

จงตอบคำถามต่อไปนี้ว่าถูกต้องหรือไม่พร้อมทั้งเหตุผล

- a) FIELD-C และ FIELD-D ต้องมี picture clause แต่ FIELD-E ไม่ต้องมี
- b) FIELD-C เป็นข้อมูลอิสระ
- c) มวลค่าของ FIELD-A, FIELD-B และ FIELD-C ทั้งหมดนี้ต้องเริ่มที่ตำแหน่ง 1 ของ SALE-RECORD
- d) ถ้า FIELD-A จบที่ตำแหน่งที่ 20 แล้ว FIELD-F ต้องเริ่มที่ตำแหน่งที่ 21
- e) ขนาดของ FIELD-D ต้องน้อยกว่าขนาดของ FIELD-B
- f) ขนาดของ FIELD-E ต้องใหญ่กว่าขนาดของ FIELD-B
- g) FIELD-F ต้องมี picture clause

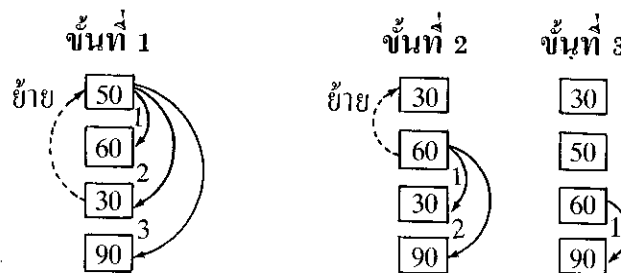


## การเรียงลำดับข้อมูล (Sorting)

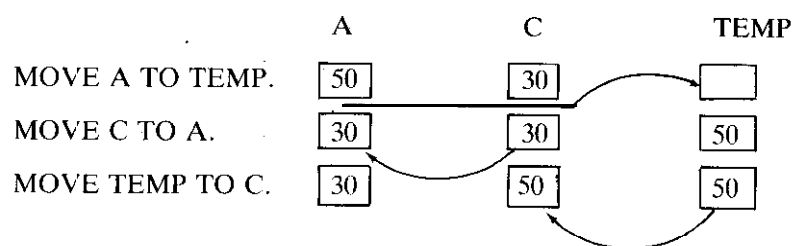
สำหรับกรณีที่ต้องการเรียงข้อมูลจากตัวเลขน้อยไปยังตัวเลขมาก (Ascending order หรือ Increasing order) สมมติให้อินพุตเป็นบัตร 4 ใบ บนที่กคะแนนสอบไล่ของนักศึกษา ดังนี้

บัตรใบที่ 1	คะแนน 50	ให้ชื่อฟิลด์คือ A
บัตรใบที่ 2	คะแนน 60	ให้ชื่อฟิลด์คือ B
บัตรใบที่ 3	คะแนน 30	ให้ชื่อฟิลด์คือ C
บัตรใบที่ 4	คะแนน 90	ให้ชื่อฟิลด์คือ D

ขั้นตอนในการทำงาน



คำสั่งให้สลับที่กันระหว่าง A กับ C



หมายเหตุ TEMP เป็น data-name ซึ่งเตรียมไว้เพื่อเก็บค่าของ A เมื่อผ่านคำสั่งแรก ค่าใน A ก็ยังคงมีอยู่เหมือนเดิม

รายละเอียดในการสั่งให้เรียงข้อมูลมีดังนี้

1) เปรียบเทียบค่าในเรคคอร์ดที่หนึ่งกับค่าในเรคคอร์ดที่สอง ปรากฏว่าข้อมูลเรียงกันอยู่แล้ว ให้ผ่านไปทำข้อ (2) แต่ถ้าข้อมูลยังไม่เรียงก็สั่งให้สลับที่กันก่อน

2) เปรียบเทียบค่าในเรคคอร์ดที่หนึ่งกับค่าในเรคคอร์ดที่สาม ปรากฏว่าข้อมูลไม่ได้เรียงลำดับจากน้อยไปหามากตามที่ต้องการ จึงต้องสลับที่กัน โดยการสร้างเนื้อที่ขึ้นใหม่ใน Working-storage section ใช้เลขบอกระดับ 77 ชื่อ TEMP มีขนาดเดียวกับ A

3) เปรียบเทียบค่าในเรคคอร์ดที่หนึ่งกับค่าในเรคคอร์ดที่สี่ ข้อมูลเรียงกันแล้วให้กลับไปทำข้อ (2) ใหม่ แต่คราวนี้กำหนดเรคคอร์ดที่สองเป็นหลักในการเปรียบเทียบมูลค่า (ขั้นที่ 2) แล้วทำแบบเดิมอีกต่อไปเรื่อย ๆ จนถึงขั้นสุดท้าย เรคคอร์ดที่ยึดเป็นหลักในการเปรียบเทียบคือ เรคคอร์ดรองสุดท้าย (เรคคอร์ดที่สามในขั้นที่ 3)

เมื่อเสร็จขั้นตอนดังกล่าวข้างต้น เราจะได้เรคคอร์ดเรียงตามลำดับที่ต้องการ

ตัวอย่าง จงเขียนโปรแกรม เรียงข้อมูล ซึ่งเป็นเปอร์เซ็นต์การสอบได้ ของนักศึกษาแต่ละคนบันทึกในบัตร 80 คอลัมน์ จำนวน 10 คน ให้เรียงจากน้อยไปหามาก แล้วพิมพ์ออกมาทางพรินเตอร์โดยกำหนดรูปแบบของอินพุท ดังนี้

คอลัมน์ 1-35

ชื่อนักศึกษา

คอลัมน์ 78-80

เปอร์เซ็นต์ที่สอบได้

รูปแบบของเอาต์พุท

พิมพ์

เรคคอร์ดเดิม 10 คน

เว้นสองบรรทัด

พิมพ์

เรคคอร์ดใหม่ที่เรียงลำดับแล้ว

จากน้อยไปหามาก จำนวน

10 คนเช่นกัน

DATA DIVISION.

FILE SECTION.

FD C-FILE LABEL RECORD IS OMITTED DATA RECORD IS C-REC.

01 C-REC.

03 NAME PICTURE X(35).

03 FILLER PICTURE X(42).

03 SCORE PICTURE 999.

FD P-FILE LABEL RECORD IS OMITTED DATA RECORD IS P-REC.

01 P-REC PICTURE X(136).

WORKING-STORAGE SECTION.

77 I PICTURE 99 VALUE ZEROS.

77 J PICTURE 99 VALUE ZEROS.

77 K PICTURE 99 VALUE ZEROS.

77 TEMP PICTURE X(80) VALUE SPACES.

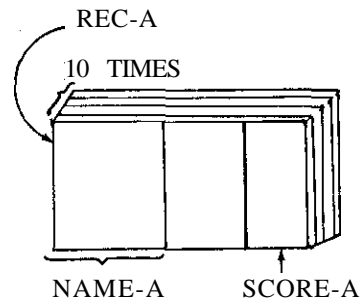
01 GROUP-A.

03 REC-A OCCURS 10 TIMES.

05 NAME-A PICTURE X(35).

05 FILLER PICTURE X(42).

05 SCORE-A PICTURE 999.



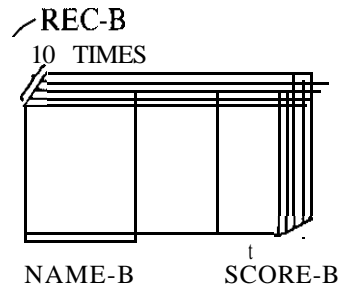
01 GROUP-B.

03 REC-B OCCURS 10 TIMES.

05 NAME-B PICTURE X(35).

05 FILLER PICTURE X(42).

05 SCORE-B PICTURE 999.



PROCEDURE DIVISION.

PARA-1.

OPEN INPUT C-FILE, OUTPUT P-FILE.

MOVE SPACES TO P-REC, GROUP-A, GROUP-B.

PARA-2.

READ C-FILE AT END GO TO PARA-3.

ADD 1 TO K.

MOVE C-REC TO REC-A (K), REC-B (K).

GO TO PARA-2.

**PARA-3.**

ADD 1 TO I.

MOVE I TO J.

**PARA-4.**

ADD 1 TO J.

IF SCORE-A (I) NOT GR SCORE-A (J) GO TO PARAJ.

MOVE REC-A (I) TO TEMP.

MOVE REC-A (J) TO REC-A (I).

MOVE TEMP TO REC-A (J).

**PARA-5.**

IF I EQ 9 GO TO **PARA-6.**

IF J EQ 10 MOVE ZEROS TO J GO TO **PARA-3.**

GO TO **PARA-4.**

**PARA-6.**

MOVE ZEROS TO I.

**PARA-7.**

ADD 1 TO I.

WRITE P-REC FROM REC-B (I).

**PARA-8.**

PERFORM **PARA-7** 9 TIMES.

**PARA-9.**

WRITE P-REC.

WRITE P-REC.

**PARA-10**

MOVE ZEROS TO I.

**PARA-11.**

ADD 1 TO I.

WRITE P-REC FROM REC-A (I).

PARA-12.

PERFORM PARA-11 9 TIMES.

CLOSE. C-FILE, P-FILE.

STOP RUN.

### ตัวอย่างโปรแกรม

คลังสินค้าแห่งหนึ่งแบ่งหน่วยขายสินค้าออกเป็นสองแผนก แต่ละแผนกจะมีหมายเลขบัญชีสินค้าอยู่ 5 รายการ เหมือนกัน หลังจากดำเนินการไป 6 เดือน ปรากฏตัวเลขจำนวนสินค้าที่จำหน่ายไปดังนี้

#### แผนก A

		เดือนที่					
		1	2	3	4	5	6
หมายเลขบัญชี	1	100	175	300	450	500	750
	2	150	150	100	75	100	150
	3	75	100	125	150	175	200
	4	200	250	300	500	750	900
	5	750	700	650	400	500	450

#### แผนก B

		เดือนที่					
		1	2	3	4	5	6
หมายเลขบัญชี	1	250	325	400	650	400	250
	2	325	250	175	100	150	200
	3	100	150	200	250	175	250
	4	225	400	350	575	700	850
	5	800	700	750	500	650	600

1) จงเขียนโปรแกรมอ่านข้อมูลซึ่งเป็นตัวเลขในแต่ละบัญชีจำนวน 10 เรคคอร์ด แล้วสร้างตารางเก็บตัวเลขเหล่านี้ สำหรับตัวเลขจากแผนก A ให้เก็บในอะเรย์ ชื่อ ACCT-1 ตัวเลขจากแผนก B ให้เก็บในอะเรย์ชื่อ ACCT-2

2) หลังจากนั้นให้สร้างอะเรย์ใหม่ชื่อ TOT เก็บตัวเลขซึ่งเป็นผลบวกของอะเรย์ทั้งสอง ชุดนี้ แล้วพิมพ์อะเรย์ TOT

3) ขั้นสุดท้ายให้หาผลบวกทั้งหมดทีละโรว์ ทีละคอลัมน์ แล้วหา grand total ผลรวมนี้ให้เก็บที่อะเรย์ 1 มิติ อะเรย์แรกชื่อ ACCOUNT มีอิลีเมนต์ชื่อ ACT เก็บตัวเลข 5 จำนวน อะเรย์ที่สองชื่อ MONTH เก็บตัวเลขในแต่ละเดือนจำนวน 8 เดือน อิลีเมนต์แต่ละตัวชื่อ MO, grand total (ผลรวมทั้งหมด) ใช้ชื่อว่า GRAND เป็นตัวเลขหนึ่งจำนวน

จากนั้นพิมพ์อะเรย์ชื่อ ACCOUNT, MONTH และค่า GRAND

ตัวอย่าง เข้าที่ทุก

	TOTAL						ACCOUNT
**	350	500	700	1100	900	1000	** 4550
**	475	400	275	175	250	350	** 1925
**	175	250	325	400	350	450	** 1950
**	425	650	650	1075	1450	1750	** 6000
**	1550	1400	1400	900	1150	1050	** 7450
MONTH	2975	3200	3350	3650	4100	4600	21875

2-OCT-1987 13:25:  
2-OCT-1987 13:25:

IDENTIFICATION DIVISION.  
PROGRAM-ID, EXAM.  
AUTHOR: CHAISANG

ENVIRONMENT DIVISION.  
CONFIGURATION SECTION.  
SOURCE-COMPUTER. VAX-11.  
OBJECT-COMPUTEK. VAX-11.

INPUT-OUTPUT SECTION.  
FILE-CONTROL.

SELECT INP-FILE ASSIGN TO "INP.DAT".  
SELECT OUT-FILE ASSIGN TO "OUTP.OUT".

DATA DIVISION.  
FILE SECTION.

FD INP-FILE LABEL RECORD IS OMITTED DATA RECORD IS IN-REC.

01	IN-REC.				
	02	DA1	PIC	999.	
	02	FILLER	PIC	X.	
	" 2	DA2	PIC	999.	
	02	FILLER	PIC	X.	
	02	DA3	PIC	999.	
	02	FILLER	PIC	X.	
	02	DA4	PIC	999.	
	02	FILLER	PIC	X.	
	02	DA5	PIC	999.	
	02	FILLFR	PIC	X.	
	02	DA6	PIC	999.	

FD OUT-FILE LABEL RECORD IS OMITTED DATA RECORD IS OUT-REC.

01	OUT-REC.				
	02	FILLER	PIC	X(132).	

WORKING-STORAGE SECTION.

01	HD1.				
	02	FILLER	PIC	X(30) VALUE SPACES.	
	02	FILLFR	PIC	X(5) VALUE TOTAL.	
01	HD2.				
	02	FILLER	PIC	X(4) VALUE SPACES.	
	02	FILLER	PIC	X(80) VALUE '1	2 3
					5 6 TOTAL.

"1	ACT1.				
	02	AAA	OCCURS	5 TIMES.	
		"5	ACCT1	OCCURS	6 TIRES PIC 999.

01	ACT2.				
	02	AA	OCCURS	5 TIMES.	
		"5	ACCT2	OCCURS	6 TIMES PIC 999.

01	TOTA.				
	02	AR	OCCURS	6 TIRES.	
		05	TUT	OCCURS	7 TINES PIC 99999.

"1	DET-LINE.				
	02	OU1	PIC	ZZ,ZZZ.	
	02	FILLER	PIC	X(3) VALUE SPACES.	
	02	OU2	PIC	ZZ,ZZZ.	
	02	FILLER	PIC	X(3) VALUE SPACES.	
	02	OU3	PIC	ZZ,ZZZ.	
	02	FILLER	PIC	X(3) VALUE SPACES.	
	02	OU4	PIC	ZZ,ZZZ.	

```

02 FILLER PIC X(3) VALUE SPACES.
02 OU PIC ZZ,ZZZ.
02 FILLER PIC X(3) VALUE SPACES.
02 OU6 PIC ZZ,ZZZ.
02 FILLER PIC X(3) VALUE SPACES.
02 ou7 PIC ZZ,ZZZ.
77 I PIC 99 VALUE 0.
77 J PIC 99 VALUE 0.
77 EOF PIC X VALUE 'N'.
PROCEDURE DIVISION.
MAIN-LINE-CONTROL.
  MOVE ZEROS TO ACT1.
  MOVE ZEROS TO ACT2.
  MOVE ZEROS TO TOTA.

  PERFORM PATH1.
  PERFORM RD-AND-STA VARYING I FROM 1 BY 1 UNTIL EOF = 'Y'.
  PERFORM CALC VARYING I FROM 1 BY 1 UNTIL I > 5.
  PERFORM CALC-GTO VARYING I FROM 1 BY 1 UNTIL I > 5.
  PERFORM PRN VARYING I FROM 1 BY 1 UNTIL I > 6.
  PERFORM CLOSING.
  STOP RUN.

PATH1.
  OPEN INPUT INP-FILE, OUTPUT OUT-FILE.
  WRITE OUT-REC FROM HD1.
  WRITE OUT-REC FROM HD2.
  READ INP-FILE AT END MOVE 'Y' TO EOF.
KD-AND-STA.
  IF I < 6
    MOVE DA1 TO ACCT1(I,1)
    MOVE DA2 TO ACCT1(I,2)
    MOVE DA3 TO ACCT1(I,3)
    MOVE DA4 TO ACCT1(I,4)
    MOVE DA5 TO ACCT1(I,5)
    MOVE DA6 TO ACCT1(I,6)
  ELSE
    COMPUTE J = I - 5
    MOVE DA1 TO ACCT2(J,1)
    MOVE DA2 TO ACCT2(J,2)
    MOVE DA3 TO ACCT2(J,3)
    MOVE DA4 TO ACCT2(J,4)
    MOVE DA5 TO ACCT2(J,5)
    MOVE DA6 TO ACCT2(J,6).
  READ INP-FILE AT END MOVE 'Y' TO EOF.

CALC.
  PERFORM CALC1 VARYING J FROM 1 BY 1 UNTIL J > 6.

CALC1.
  COMPUTE TOT(I,J) = ACCT1(I,J) + ACCT2(I,J).
  COMPUTE TOT(I,7) = TOT(I,7) + ACCT1(I,J) + ACCT2(I,J).
  COMPUTE TDT(6,J) = TOT(6,J) + ACCT1(I,J) + ACCT2(I,J).

CALC-GTO.
  COMPUTE TOT(6,7) = TOT(6,7) + TOT(I,7).

PRN.
  MOVE TOT(I,1) TO OU1.
  MOVE TOT(I,2) TO OU2.
  MOVE TOT(I,3) TO OU3.

```



```

MOVE TOT(I,4) TO OU4.
MOVE TOT(I,5) TO OU5.
MOVE TOT(I,6) TO OU6.
MOVE TOT(I,7) TO OU7.
WRITE OUT-REC FROM OET-LINE.

```

```

CLOSING.
CLOSE INP-FILE , OUT-FILE.
STOP RUN.

```

รูปแบบของ input

100	175	300	450	500	750
150	150	100	075	100	150
075	100	125	150	175	200
200	250	300	500	750	900
750	700	650	400	500	450
250	325	400	650	400	250
325	250	175	100	150	200
100	150	200	250	175	250
225	400	350	575	700	850
800	700	750	500	650	600

รูปแบบของ output

1	2	3	TOTAL 4	5	b	TOTAL
350	500	700	1,100	900	1,000	4,550
475	400	<b>275</b>	175	250	350	1,925
175	250	325	400	350	450	<b>1,950</b>
<b>425</b>	<b>650</b>	650	<b>1,075</b>	<b>1,450</b>	<b>1,750</b>	<b>6,000</b>
<b>1,550</b>	1,400	<b>1,400</b>	900	<b>1,150</b>	<b>1,050</b>	7,450
<b>2,975</b>	<b>3,200</b>	3,350	<b>3,650</b>	<b>4,100</b>	4,600	<b>21,875</b>

## แบบฝึกหัด

1. กำหนดฟิลด์ซึ่ง defined ไว้แล้ว 5 ฟิลด์ มีชื่อเรียกดังนี้ FIELD-1, FIELD-2, FIELD-3, FIELD-4 และ FIELD-5 จงเขียนคำสั่งเท่าที่จำเป็นในการหาฟิลด์ที่มีมูลค่าสูงสุดและฟิลด์ที่มีมูลค่าต่ำที่สุด ได้ผลลัพธ์เอาไปไว้ในฟิลด์ชื่อ LARGEST-VALUE และ SMALLEST-VALUE ตามลำดับ
2. มีมูลค่าอยู่ 4 จำนวนตั้งชื่อว่า A, B, C, และ D เก็บอยู่ภายในหน่วยความจำของเครื่องคอมพิวเตอร์ จงเขียน flowchart และ โปรแกรมเฉพาะส่วนที่เป็น Procedure division เรียงมูลค่าทั้งหมดนี้จากมากไปหาน้อยตามลำดับ กำหนดให้ T เป็นหน่วยความจำชั่วคราว จากนั้นพิมพ์มูลค่าทั้ง 4 ตัวนี้ซึ่งเรียงลำดับแล้วออกมาทางเทอร์มินัล
3. มีมูลค่าอยู่ 5 จำนวน ชื่อ A, B, C, D, และ E เก็บอยู่ภายในหน่วยความจำของเครื่องคอมพิวเตอร์ จงเขียน flowchart และโปรแกรมเฉพาะส่วนที่เป็น procedure division กำหนดหาตัวเลขที่มีค่าน้อยที่สุดและมากที่สุด แล้วพิมพ์ชื่อและตัวเลขที่มีค่ามากที่สุดและน้อยที่สุดออกมาทางเทอร์มินัล
4. ห้างสรรพสินค้าหัวหมากได้จัดรายการขายเครื่องดื่มประเภทน้ำอัดลมผสมสี จำนวน 4 ชนิดใน 1 อาทิตย์ที่ผ่านมาได้ตัวเลขตามตารางข้างล่างนี้

	ชนิดที่ 1	ชนิดที่ 2	ชนิดที่ 3	ชนิดที่ 4
วันอาทิตย์	67	39	20	58
วันจันทร์	50	35	22	62
วันอังคาร	58	40	20	60
วันพุธ	70	36	24	64
วันพฤหัสบดี	66	32	21	57
วันศุกร์	70	30	25	63
วันเสาร์	40	10	15	30

จงเขียน flowchart และโปรแกรม

- a) พิมพ์ยอดขายว่า เมื่ออาทิตย์ที่ผ่านมา น้ำอัดลมผสมสีแต่ละชนิดขายได้เป็นจำนวนกี่ขวด
- b) พิมพ์ยอดขายว่า ในแต่ละวันขายน้ำอัดลมได้กี่ขวด
- c) พิมพ์จำนวนน้ำอัดลมทั้งหมดที่ขายได้ในอาทิตย์นั้น

หมายเหตุ ให้พิมพ์หัวเรื่องของตารางด้วย

5. ชมรมคุ้มครองผู้บริโภคกาแฟกลุ่มหนึ่ง ได้ตั้งข้อสังเกตว่าราคาของกาแฟผงแดงขนาดบรรจุขวด ๆ ละ 283 กรัม ได้เพิ่มขึ้นมากตลอดระยะเวลาที่ผ่านมา จึงทำการสำรวจราคา กาแฟชนิดนี้โดยการสุ่มตัวอย่างร้านค้า แล้วบันทึกราคาของกาแฟทุกเดือนระหว่างปี พ.ศ. 2527, 2528 และ 2529 พบว่าการเปลี่ยนแปลงของราคากาแฟอยู่ในช่วงขวดละ 184 บาท ถึง 255 บาท ดังตารางข้อมูลที่แสดงนี้

ปี	เดือน											
	1	2	3	4	5	6	7	8	9	10	11	12
2527	187	189	189	189	185	184	190	200	190	190	190	200
2528	195	195	195	190	185	200	210	220	215	200	210	197
2529	220	225	230	235	240	245	250	255	255	245	247	240

ตัวเลขทั้งหมดนี้ได้บันทึกลงในบัตรข้อมูล ข้อมูลแต่ละปีใช้บัตร 80 คอลัมน์ 1 ใบ จงเขียน flowchart และโปรแกรมวิเคราะห์การเปลี่ยนแปลงของราคากาแฟในช่วง 3 ปีข้างต้นดังนี้

a) คำนวณเปอร์เซ็นต์การเพิ่มขึ้นของราคากาแฟแต่ละเดือนโดยใช้สูตร

$$\% \text{ การเพิ่มของราคากาแฟจากเดือนที่ } i-1 \text{ ถึงเดือนที่ } i = 100 * \left( \frac{\text{ราคากาแฟเดือนที่ } i}{\text{ราคากาแฟเดือนที่ } i-1} - 1 \right)$$

b) คำนวณราคาเฉลี่ยของกาแฟในแต่ละปี

c) คำนวณเปอร์เซ็นต์การเพิ่มขึ้นของราคากาแฟเฉลี่ยแต่ละปีจากปี 2527-2528 และจากปี 2528-2529 โดยใช้สูตรทำนองเดียวกับข้อ a)

d) พิมพ์ตารางข้อมูลที่เป็นอินพุท ตารางเปอร์เซ็นต์การเพิ่มของราคากาแฟแต่ละเดือน (จากเดือนที่ 1-2, เดือนที่ 2-3, ... เดือนที่ 11-12) ราคาเฉลี่ยของกาแฟในแต่ละปีและเปอร์เซ็นต์การเพิ่มขึ้นของราคากาแฟเฉลี่ยแต่ละปี

6. จงเขียน flowchart และโปรแกรมสร้างตารางแจกแจงความถี่ (frequency table) จากคำตอบในแบบสอบถามจำนวน 50 ชุด ซึ่งได้จากการสำรวจภาวะตลาดของห้างสรรพสินค้าแห่งหนึ่งในแบบสอบถามแต่ละชุดจะมีคำถามอยู่ 25 ข้อ และในคำถามแต่ละข้อจะมีคำตอบอยู่ 10 ตัวเลือกมีค่าเป็น 0, 1, 2, 3, 4, 5, 6, 7, 8 และ 9 ตามลำดับ

ตัวอย่าง บัตรข้อมูล

คอลลัมน์ 1-4 หมายเลขแบบสอบถาม

คอลลัมน์ 11 คำตอบของคำถามข้อ 1

คอลลัมน์ 12 คำตอบของคำถามข้อ 2

คอลลัมน์ 13 คำตอบของคำถามข้อ 3

.

.

.

คอลลัมน์ 35 คำตอบของคำถามข้อ 25

0001	0174321560011238785234015
------	---------------------------

ผลลัพธ์ให้พิมพ์ดังนี้

ตารางแจกแจงความถี่ของแบบสอบถาม

ตัวเลือก คำถาม	0	1	2	3	4	5	6	7	8	9	รวม
ข้อ 1	11	8	12	4	0	6	4	1	1	3	50
ข้อ 2											50
ข้อ 3											50
.											.
.											.
.											.
ข้อ 25											50

7. จงเขียน flowchart และโปรแกรมคำนวณจำนวนเงินทั้งหมดที่บริษัทต้องจ่ายให้แก่พนักงาน

ในแต่ละสัปดาห์ ข้อมูลของพนักงานแต่ละคนอยู่ในบัตร 80 คอลลัมน์ มีรายละเอียดดังนี้

คอลลัมน์

ความหมาย

1

ชนิดของบัตรข้อมูล

1 = REGULAR

2 = OVERTIME

คอลัมน์	ความหมาย
	3 = BONUS
	4 = OTHERS
2-6	หมายเลขประจำตัวของพนักงาน
7-11	ค่าจ้างทำงานมีจุดทศนิยม 2 ตำแหน่ง

ตัวอย่าง อินพุต			
บัตรใบที่ 1	1	1111	08542
บัตรใบที่ 2	2	1111	01465
.	1	1112	09011
.	2	1112	01200
.	1	1113	10000
	3	1113	02000
	4	1114	50000
	1	1115	08065
	1	1116	12000
	2	1116	02376
	4	1116	18011
	1	1117	00500
	1	1118	
	3	1117	

ต้องการคำนวณจำนวนเงินจากบัตรข้อมูลทั้งหมดแยกออกเป็น 4 กลุ่ม ตามชนิดของบัตรข้อมูลในคอลัมน์ที่ 1 แล้วพิมพ์เอาที่พูดดังนี้

**PROOF TOTALS FOR WEEKLY PAYROLL**

<b>REGULAR</b>	7	<b>\$756.28</b>
<b>OVERTIME</b>	3	<b>\$50.41</b>
<b>BONUS</b>	2	<b>\$25.00</b>
<b>OTHERS</b>	2	<b>\$51.00</b>
<b>GRAND TOTAL =</b>	<b>14</b>	<b>\$1,341.69</b>

```

1 IDENTIFICATION DIVISION.
2 PROGRAM-ID. ASSIGN7.
3 AUTHOR. PRAPUN.
4 *****
5 ENVIRONMENT DIVISION.
6 CONFIGURATION SECTION.
7 SOURCE-COMPUTER. VAX-11.
8 OBJECT-COMPUTER. VAX-11.
9 INPUT-OUTPUT SECTION.
10 FILE-CONTROL.
11 SELECT IN-FILE ASSIGN TO DISK.
12 SELECT OUT-FILE ASSIGN TO DISK.
13 *****
14 DATA DIVISION.
15 FILE SECTION.
16 FD IN-FILE LABEL RECORD IS STANDARD
17 VALUE OF ID IS 'ASSIGN7.DAT'
18 DATA RECORD IS IN-REC.
19 01 IN-REC.
20 05 PEPSI PIC 99.
21 05 PIC X(5) VALUE SPACE.
22 05 COLAR PIC 99.
23 05 PIC X(5) VALUE SPACE.
24 05 FASER PIC 99.
25 05 PIC X(5) VALUE SPACE.
26 05 FANTA PIC 99.
27 FD OUT-FILE LABEL RECORD IS STANDARD
28 VALUE OF ID IS 'ASSIGN7.OUT'
29 DATA RECORD IS OUT-REC.
30 01 OUT-REC.
31 05 FILLER PIC X(80).
32 *****
33 WORKING-STORAGE SECTION.
34 77 END-FILE PIC X VALUE 'N'.
35 77 I PIC 9 VALUE ZERO.
36 77 J PIC 9 VALUE ZERO.
37 01 HEAD-1.
38 PIC X(23) VALUE SPACE.
39 05 PIC X(50) VALUE
40 'NIGHT SPORT SOFTDRINK PRODUCE COMPANY LIMITED'
41 01 HEAD-2.
42 05 PIC X(36) VALUE SPACE.
43 us PIC X(15)
44 VALUE 'WEEKLY REPORT'.
45 01 HEAD-3.
46 05 PIC X(24) VALUE SPACE.
47 05 PIC X(6) VALUE 'PEPSI'.
48 05 PIC X(5) VALUE SPACE.
49 05 PIC X(6) VALUE 'COLA'.
50 05 PIC X(4) VALUE SPACE.
51 05 PIC X(6) VALUE 'FASER'.
52 05 PIC X(4) VALUE SPACE.
53 05 PIC X(6) VALUE 'FANTA'.
54 05 PIC X(8) VALUE SPACE.
55 05 PIC X(9) VALUE 'TOTAL-DAY'.
56 01 LINE-1.
57 05 PIC X(9) VALUE SPACE.

```

ASSIGN7  
SOURCE LISTING

Z-SEP-1987 10  
2-SEP-1987 10

```

58          05          PIC          X(67) VALUE ALL ' '.
60          01 HEAD-OUT.
61          05          PIC          X(9)          VALUE          SPACE.
          05 DAY-OUT          PIC X(10).
62          05          PIC          X(6)          VALUE          SPACE.
63          05          TYPE1          PIC ZZZ.
64          05          PIC          X(5)          VALUE          SPACE.
65          05 TYPE2          PIC ZZZ.
66          05          PIC          X(7) VALUE          SPACE.
68          05 TYPE3          PIC ZZZ.
69          05          TYPE4          PIC          X(7) VALUE          SPACE.
          PIC ZZZ.
70          05          PIC          X(13)          VALUE          SPACE.
71          05 TOTAL-DAY          PIC ZZZZ.
72          0 1 TABLE1.
73          05 DAY-ROY OCCURS 8 TIMES.
          IO TYPE-COL          PIC 9(4) OCCURS 5 TIMES.
7 4
75          0 1 TABLE2.
76          05          PIC          PIC          X(10)          X(10) VALUE 'MONDAY'.
78          05          PIC          X(10) VALUE 'TUESDAY'.
79          05          PIC          X(10) VALUE 'WEDNESDAY'.
80          05          PIC          X(10) VALUE 'THURSDAY'.
81          05          PIC          X(10) VALUE 'FRIDAY'.
82          05          PIC          X(10) VALUE 'SATURDAY'.
          05          PIC          X(10) VALUE 'TOTAL-TYPE'.
II: 01 TABLE3 R E D E F I N E S TABLE2.
85          0 5          DAY-NAME PIC X(10) OCCURS 8 TIMES.
86          PROCEDURE          DIVISION.
87          BEGIN.
88          OPEN INPUT IN-FILE, OUTPUT OUT-FILE.
89          WRITE OUT-REC FROM HEAD-1 AFTER 3.
90          URITE OUT-REC FRO" HEAD-2 AFTER 2.
91          URITE OUT-REC FROM LINE-1 AFTER 2.
92          WRITE OUT-REC FROM HEAD-3.
9: 9          URITE OUT-REC FROM LINE-1.
          MOVE ZERO TO TABLE1.
95          PERFORM READ-FILE VARYING I FROM 1 BY 1 UNTIL I > 7
96          PERFORM CAL-PRINT VARYING I FROM 1 BY 1 UNTIL I > 7
          AFTER          FROM 1 UNTIL , > 4 .
98          PERFORM PRINT-OUT VARYING I FROM 1 BY 1 UNTIL I > 8.
99          PERFORM CLOSE-FILE.
100         STOP RUN.
101         READ-FILE.
102         READ IN-FILE AT END MOVE 'Y' TO END-FILE.
103         MOVE PEPSI TO TYPE-COL (11).
104         MOVE COLAR TO TYPE-COL (12).
105         MOVE FASER TO TYPE-COL (13).
106         MOVE FANTA TO TYPE-COL (14).
107         CAL-PRINT.
108         ADD TYPE-COL (11) TO TYPE-COL (15).
109         ADD TYPE-COL (11) TO TYPE-COL (81).
110         ADD TYPE-COL (11) TO TYPE-COL (85).
111         PRINT-OUT.
112         ROVE DAY-NAME (1) TO DAY-OUT.
113         MOVE TYPE-COL (11) TO TYPE1.
114         MOVE TYPE-COL (12) TO TYPE2.

```

```

115      MOVE TYPE-COL (I3) TO TYPE3.
116      MOVE TYPE-COL (I4) TO TYPE4.
117      MOVE TYPE-COL (I5) TO TOTAL-DAV.
118      IF I = 8 WRITE OUT-REC FROM LINE-1
119              WRITE OUT-REC FROM HEAD-OUT,
120      ELSE "RITE OUT-REC FROM HEAD-OUT.
121      CLOSE-FILE.
122      WRITE OUT-REC FROM LINE-I.
123      CLOSE IN-FILE, OUT-FILE.
    
```

รูปแบบ input

67	39	20	58
50	35	22	62
58	40	LO	60
70	36	24	64
66	32	21	57
70	30	25	63
40	10	15	30

รูปแบบ output

NIGHT SPORT SOFTDRINK PRODUCE COMPANY LIMITED

WEEKLY REPORT

	PEPSI	COLA	FASER	FANTA	TOTAL-DAY
SUNDAY	67	39	20	58	184
MONDAY	50	35	22	62	169
TUESDAY	58	40	20	60	178
WEDNESDAY	70	36	24	64	194
THURSDAY	66	32	21	57	176
FRIDAY	70	30	25	63	188
SATURDAY	40	10	15	30	95
TOTAL-TYPE	421	222	147	394	1184