



## บทที่ 4

### การพัฒนาระบบการแปลงตารางตัดสินใจ เป็นส่วนของโปรแกรมคอมพิวเตอร์

#### โปรแกรมหลัก (Main Program)

เป็นโปรแกรมเริ่มต้นของระบบการแปลงตารางตัดสินใจ

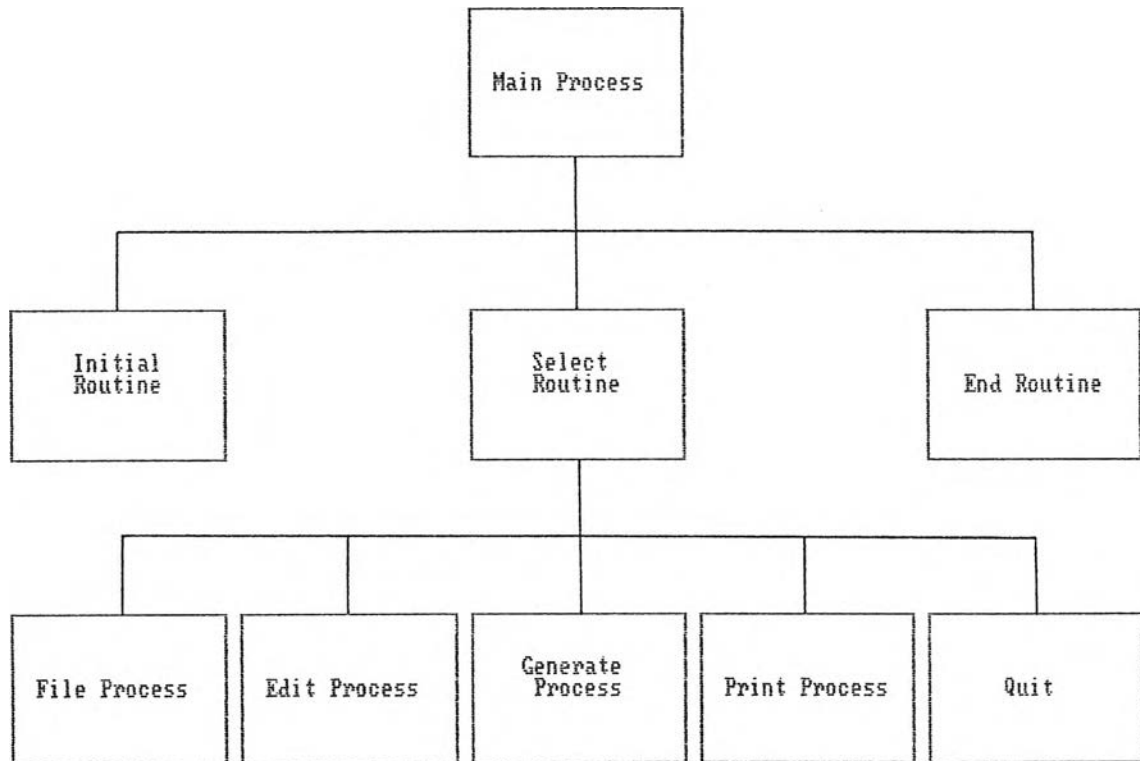
#### 1. หน้าที่ของโปรแกรม

ทำการแสดงรายการหลัก (Main Menu) ควบคุมการทำงานของระบบงานย่อยต่าง ๆ ทำการรับคำสั่งจากผู้ใช้ว่าต้องการเลือกระบบงานย่อยใด ซึ่งระบบงานย่อยดังกล่าวประกอบด้วย

- 1.1 การจัดการเพิ่มข้อมูล (File Process)
- 1.2 การรับและแก้ไขข้อมูล (Edit Process)
- 1.3 การสร้างส่วนของโปรแกรมภาษาโคบอล (Generate Process)
- 1.4 การพิมพ์ส่วนของโปรแกรมและตารางตัดสินใจ (Print Process)

#### 2. ผังโครงสร้างของโปรแกรม

(แสดงในรูปที่ 4.1)



รูปที่ 4.1 แผนผังโครงสร้างของโปรแกรมหลัก

### 3. รายละเอียดของโปรแกรม มีขั้นตอนดังต่อไปนี้

#### 3.1 การเริ่มต้นเตรียมระบบ

กำหนดค่าเริ่มต้นให้กับรายการเลือก กำหนดค่าเริ่มต้นให้กับจอภาพ ทำให้จอภาพว่าง แสดงรายการหลัก (Main Menu) ทางจอภาพ ลบล้างหน่วยความจำ (Clear Memory)

#### 3.2 ส่วนเลือกรายการหลัก

ในส่วนนี้จะทำการรับข้อมูลจากแผงแป้นอักขระ (Keyboard)ว่าจะเลือกรายการที่ปรากฏบนจอภาพว่าจะเลือกการทำงานใด การเลือกรายการ กระทำได้ 2 วิธี คือ โปรแกรมได้จัดการให้รับข้อมูลโดยการเลื่อนแถบสว่างไปยังรายการที่ต้องการแล้วกดแป้นป้อนเข้า (ENTER) หรือทำได้โดยการกดแป้นอักขระตัวแรกของรายการนั้น ๆ ที่ต้องการ รายการหลักที่กล่าวมามีรายการดังต่อไปนี้

3.2.1 File ถ้าเลือกรายการนี้จะทำการจัดการแก้ไขข้อมูล (ซึ่งจะกล่าวในรายละเอียดของโปรแกรมต่อไป) โดยปรากฏรายการย่อยให้เลือกดังต่อไปนี้

3.2.1.1 New เป็นการเริ่มต้นการทำงานกับแก้ไขข้อมูลใหม่

3.2.1.2 Load เป็นการดึงแก้ไขข้อมูลที่มีอยู่แล้วเข้ามาในหน่วยความจำ

3.2.1.3 Save จะนำข้อมูลทั้งหมดในหน่วยความจำลงเก็บในแฟ้มข้อมูลชนิด TBL

3.2.2 Edit ถ้าเลือกรายการนี้จะทำการรับและแก้ไขข้อมูล (ซึ่งจะกล่าวถึงรายละเอียดของโปรแกรมส่วนนี้ต่อไป) โดยปรากฏรายการย่อยให้เลือกดังต่อไปนี้

3.2.2.1 Condition รับและแก้ไขข้อมูลเงื่อนไข

3.2.2.2 Action รับและแก้ไขข้อมูลการกระทำ

3.2.2.3 Rule รับและแก้ไขจำนวนกฎในตาราง

3.2.2.4 Table รับและแก้ไขข้อมูลในตารางตัดสินใจ

3.2.3 Generate ถ้าเลือกรายการนี้จะทำการสร้างส่วนของโปรแกรมภาษาโคบอลจากข้อมูลที่มีอยู่ในหน่วยความจำ (ซึ่งจะกล่าวถึงรายละเอียดของโปรแกรมส่วนนี้ต่อไป)

3.2.4 Print ถ้าเลือกรายการนี้จะทำการเก็บผลลัพธ์ที่ได้ลงพิมพ์ข้อมูลหรือพิมพ์ออกเครื่องพิมพ์ (ซึ่งจะกล่าวถึงรายละเอียดของโปรแกรมส่วนนี้ต่อไป) โดยปรากฏรายการย่อยให้เลือกดังต่อไปนี้

3.2.4.1 Table to Printer พิมพ์ตารางตัดสินใจ  
ออกทางเครื่องพิมพ์

3.2.4.2 Table to File เก็บตารางตัดสินใจลงพิมพ์  
ข้อมูล LST

3.2.4.3 Program to Printer พิมพ์ส่วนของ  
โปรแกรมที่สร้างออกทางเครื่องพิมพ์

3.2.4.4 Program to File เก็บส่วนของโปรแกรมที่  
สร้างลงพิมพ์ข้อมูล COB

3.2.5 Quit ถ้าเลือกรายการนี้จะเป็นการกำหนดค่าของ flag ให้สิ้นสุดการทำงานของส่วนเลือกรายการเป็นการจบโปรแกรม

### 3.3 การจบโปรแกรม

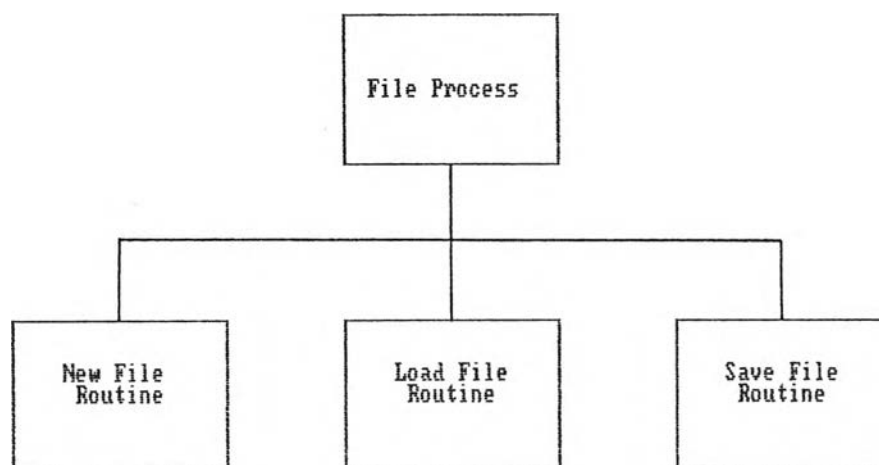
จะจัดการทำให้จอภาพว่าง ลบล้างหน่วยความจำ กำหนดตัวชี้ตำแหน่ง (cursor) อยู่ในตำแหน่ง (0,0) แล้วจะมีข้อความปรากฏทางจอภาพว่าโปรแกรมจบแบบปกติ เป็นการจบการทำงานของโปรแกรม

## โปรแกรมจัดการแฟ้มข้อมูล (File Processing Program)

### 1. หน้าที่ของโปรแกรม

ทำการลบล้างหน่วยความจำ เตรียมที่สำหรับแฟ้มข้อมูลใหม่ ทำการอ่านข้อมูลจากแฟ้มข้อมูลที่ได้สร้างไว้เรียบร้อยแล้วมาไว้ในหน่วยความจำ และทำการเก็บข้อมูลในหน่วยความจำลงแฟ้มข้อมูลแบบ TBL

### 2. ผังโครงสร้างของโปรแกรม



รูปที่ 4.2 ผังแสดงโครงสร้างของโปรแกรมจัดการแฟ้มข้อมูล

### 3. รายละเอียดของโปรแกรม มีขั้นตอนดังต่อไปนี้

ทำการเริ่มต้นโปรแกรม เก็บจอภาพปัจจุบันกำหนด flag ของจอภาพใหม่ ให้เป็น active หลังจากนั้นจะทำการทำจอภาพให้ว่าง สร้างกรอบแสดงรายการ แสดงรายการเลือก (Menu) ย่อสทางจอภาพดังต่อไปนี้

#### 3.1 New

#### 3.2 Load

#### 3.3 Save

จะรับข้อมูลจากแผงแป้นอักขระ ถ้าข้อมูลที่รับจากแผงแป้นอักขระเป็นแป้นหลัก (Escape) จอภาพจะถูกทำให้ว่างแล้วแสดงรายการหลัก ถ้าข้อมูลที่รับเกิดจากการกดแป้นลูกศรซ้ายและขวา จอภาพจะแสดงรายการย่อยของรายการหลักถัดไปตามที่จำนวนครั้งที่กด ถ้าข้อมูลที่รับเกิดจากการเลื่อนแถบสว่างโดยการกดแป้นลูกศรขึ้นหรือลง แล้วกดแป้นป้อนเข้า ซึ่งเป็นกรเลือกรายการในรายการย่อยนั้น จะทำงานตามรายการย่อยที่เลือกนั้น นอกจากนี้การเลือกรายการย่อยยังสามารถใช้แป้นพิเศษโดยการกดแป้นตัวอักษรตัวแรกของรายการย่อยที่ต้องการก็จะทำงานตามรายการที่เลือกเช่นกัน รายละเอียดของแต่ละส่วนของโปรแกรมมีดังต่อไปนี้

#### 3.1 ถ้าเลือก N หรือ New

ทำการลบล้างหน่วยความจำ กำหนดค่า เริ่มต้น ให้กับตัวแปรหน่วยความจำ (Memory variable) ทุกตัวเพื่อเตรียมรับข้อมูลใหม่

#### 3.2 ถ้าเลือก L หรือ Load

จะรับชื่อแฟ้มข้อมูล TBL โดยใส่เฉพาะชื่อไม่ต้องใส่ extension ซึ่งโปรแกรมจะทำการเพิ่ม extension ให้ โปรแกรมจะทำการตรวจสอบว่ามีแฟ้มข้อมูลนี้อยู่หรือไม่ถ้าไม่มีจะทำการแสดง Error Message และให้ผู้ใช้งานใส่ชื่อใหม่ ถ้ามีจะตรวจสอบแฟ้มข้อมูลว่าเป็นแฟ้มข้อมูลที่ต้องการหรือไม่โดยการตรวจสอบชนิดของแฟ้มข้อมูล ถ้าถูกต้องจะนำข้อมูลในแฟ้มข้อมูลมาไว้ในหน่วยความจำ ข้อมูลในแฟ้มข้อมูลได้แก่

- 3.2.1 จำนวนเงื่อนไขทั้งหมด
- 3.2.2 จำนวนการกระทำทั้งหมด
- 3.2.3 จำนวนกฎทั้งหมด
- 3.2.4 รายละเอียดของเงื่อนไขที่อยู่ในรูปอักขระ (String)
- 3.2.5 รายละเอียดของการกระทำอยู่ในรูปสายอักขระ
- 3.2.6 กฎทั้งหมดที่เก็บในตารางตัดสินใจจะอยู่ในรูปที่เรียงต่อกัน โดยเงื่อนไขและการกระทำ 1 ชุดจะขึ้นด้วยช่องว่าง 1 ช่อง

### 3.3 ถ้าเลือก S หรือ Save

จะทำการนำข้อมูลจากหน่วยความจำลงเก็บในแฟ้มข้อมูล TBL ซึ่งถ้าเป็นแฟ้มข้อมูลที่สร้างขึ้นใหม่จะรับชื่อโดยให้ผู้ใช้กำหนด (ชื่อแฟ้มข้อมูลมีความยาวได้สูงสุด 8 ตัวอักษร) แต่ถ้าเป็นแฟ้มข้อมูลเก่าจะใช้ชื่อเดิม การกำหนดชื่อไม่ต้องใส่ extension โปรแกรมจะเพิ่ม extension .TBL ให้การเก็บลงแฟ้มข้อมูลโปรแกรมจะใส่ชนิดของแฟ้มข้อมูลว่า DECISION ให้ที่ตำแหน่งแรกของแฟ้มข้อมูล

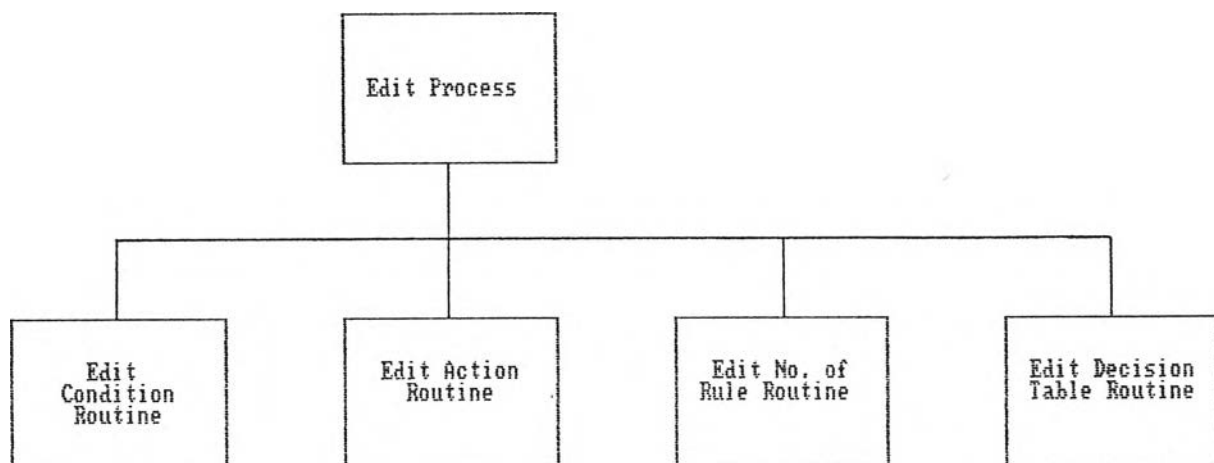


## โปรแกรมบรรณาธิการเพิ่มข้อมูล (Edit Processing Program)

### 1. หน้าที่ของโปรแกรม

จะทำการสร้างหรือแก้ไขปรับปรุงข้อมูลต่าง ๆ ในแฟ้มข้อมูล TBL เพื่อเตรียมข้อมูลไว้สำหรับสร้างส่วนของโปรแกรมภาษาโคบอล

### 2. ผังโครงสร้างของโปรแกรม



รูปที่ 4.3 ผังแสดงโครงสร้างของโปรแกรมบรรณาธิการเพิ่มข้อมูล

### 3. รายละเอียดของโปรแกรม มีขั้นตอนดังต่อไปนี้

แสดงรายการย่อยของรายการหลัก Edit ทางจอภาพ รับข้อมูลจากแผงแป้นอักขระว่าจะเลือกรายการใด การเลือกรายการจากรายการเลือกย่อยทำได้ 2 วิธีคือการเลื่อนแถบสว่างใช้แป้นลูกศรขึ้นหรือลงเลื่อนไปยังรายการที่ต้องการแล้วกดแป้นป้อนเข้า อีกวิธีคือการแป้นตัวอักษรตัวแรกของรายการที่ต้องการ ถ้าต้องการให้จอภาพแสดงรายการหลักจะกดแป้นหลัก เมื่อโปรแกรมรับค่าแป้นหลัก จะทำการทำให้จอภาพว่างแล้วแสดงรายการหลัก หรือมีอีกวิธีโดยการกดแป้นลูกศรซ้ายหรือขวาโปรแกรมจะทำการแสดงรายการย่อยของรายการหลักถัดไปตามทิศทางที่ได้รับค่าลูกศรมา รายละเอียดของส่วนของโปรแกรม แต่ละรายการมีดังต่อไปนี้

#### 3.1 ถ้าเลือก C หรือ Condition

3.1.1 ทำการแสดงผลกรอบรับข้อมูลของเงื่อนไข (Condition Frame) ทั้งหมดบนจอภาพปัจจุบัน จะปรากฏช่องว่างให้ใส่เงื่อนไขโดยมีจำนวนเงื่อนไขสูงสุด 8 เงื่อนไข ซึ่งจะรับข้อมูลได้ 60 ตัวอักษรต่อ 1 เงื่อนไข การที่โปรแกรมสามารถรับเงื่อนไขได้เป็นจำนวนข้อมูลสูงสุด 8 เงื่อนไขเนื่องจากเนื้อหาบางส่วนบนจอภาพในขณะที่รับและแสดงผลของตารางต้องใช้ในการแสดงข้อความบางอย่างและแสดงผลกรอบตารางตัดสินใจ จึงมีเนื้อหาให้ใส่เงื่อนไขได้สูงสุด 8 เงื่อนไข รวมทั้งการกระทำก็สามารถรับได้สูงสุด 8 การกระทำ เนื่องจากสาเหตุเดียวกันนี้ ส่วนขนาดของข้อมูลแต่ละเงื่อนไขสามารถเพิ่มให้ขนาดใหญ่กว่านี้ได้แต่ไม่เกิน 75 ตัวอักษร เนื่องจากจอภาพมี 80 ตัวอักษร เนื้อหาบางส่วนต้องใช้ในการแสดงข้อความและตีกรอบ

3.1.2 แสดงข้อมูลของเงื่อนไขทั้งหมดบนกรอบรับข้อมูลบนจอภาพ (ถ้ามีข้อมูลในหน่วยความจำ) ในกรณีที่เปิดเพิ่มข้อมูลใหม่ตำแหน่งที่รับข้อมูลบนจอภาพจะถูกทำให้ว่างรอรับข้อมูลที่จะใส่ลงในแต่ละช่อง

3.1.3 ทำการรับข้อมูลหรือแก้ไขเปลี่ยนแปลงข้อมูลเงื่อนไขที่แสดงบนจอภาพโดยข้อมูลทั้งหมดที่ได้แก้ไขจะนำลงเก็บในหน่วยความจำบนพื้นที่ที่ได้เตรียมไว้เก็บข้อมูลชั่วคราวเพื่อนำไปใช้ในขั้นตอนของการสร้างส่วนของโปรแกรมภาษาโคบอล การ

เพิ่ม ลด หรือแก้ไขเงื่อนไขผู้ใช้สามารถใช้เป็นต่อไปนี้ได้

ก) Del ใช้สำหรับลบตัวอักษรตรงตำแหน่งที่ตัวชี้ตำแหน่ง  
ข) Backspace ใช้สำหรับลบตำแหน่งที่อยู่ทางซ้ายของ  
ตัวชี้ตำแหน่ง

ค) การป้อนตัวอักษรทับบนตัวอักษรเดิม

3.1.4 เลิกจากการแก้ไขข้อมูลโดยการกดปุ่มหลัก โปรแกรม  
จะนับจำนวนเงื่อนไขเพื่อหาจำนวนเงื่อนไขทั้งหมด

### 3.2 ถ้าเลือก A หรือ Action

3.2.1 ทำการแสดงกรอบรับข้อมูลของการกระทำ (Action  
Frame) ทั้งหมดบนจอภาพปัจจุบัน จะปรากฏช่องว่างให้ใส่การกระทำโดยมีจำนวนการ  
กระทำสูงสุด 8 การกระทำ ซึ่งจะรับข้อมูลได้ 60 ตัวอักษรต่อ 1 การกระทำ ส่วนขนาด  
ของข้อมูลแต่ละการกระทำสามารถเพิ่มให้ขนาดใหญ่กว่านี้ได้แต่ไม่เกิน 75 ตัวอักษร

3.2.2 แสดงข้อมูลของการกระทำทั้งหมดบนกรอบรับข้อมูลบนจอภาพ  
(ถ้ามีข้อมูลในหน่วยความจำ) ในกรณีที่เปิดเพิ่มข้อมูลใหม่ตำแหน่งที่รับข้อมูลบนจอภาพจะถูก  
ทำให้ว่างรอรับข้อมูลที่จะใส่ลงในแต่ละช่อง

3.2.3 ทำการรับข้อมูลหรือแก้ไขเปลี่ยนแปลงข้อมูลการกระทำที่  
แสดงบนจอภาพโดยข้อมูลทั้งหมดที่ได้แก้ไขจะนำลงเก็บในหน่วยความจำบนพื้นที่ที่ได้เตรียมไว้  
เก็บข้อมูลชั่วคราวเพื่อนำไปใช้ในขั้นตอนของการสร้างส่วนของโปรแกรมภาษาโคบอล การ  
เพิ่ม ลด หรือแก้ไขเงื่อนไขผู้ใช้สามารถใช้เป็นต่อไปนี้ได้

ก) Del ใช้สำหรับลบตัวอักษรตรงตำแหน่งตัวชี้ตำแหน่ง  
ข) Backspace ใช้สำหรับลบตำแหน่งที่อยู่ทางซ้ายของ  
ตัวชี้ตำแหน่ง

ค) การป้อนตัวอักษรทับบนตัวอักษรเดิม

3.2.4 เลิกจากการแก้ไขข้อมูลโดยการกดปุ่มหลัก โปรแกรม  
จะนับจำนวนการกระทำเพื่อหาจำนวนการกระทำทั้งหมด

### 3.3 ถ้าเลือก R หรือ Rule

#### 3.3.1 แสดงกรอบรับจำนวนกฎ

3.3.2 รับจำนวนกฎทั้งหมดในตารางตัดสินใจ จำนวนกฎสูงสุด 36 กฎ เนื่องจากการแสดงตารางบนจอภาพเนื้อที่บางส่วนใช้ในการตีกรอบของตาราง แสดงลำดับที่ของเงื่อนไข ลำดับที่ของการกระทำ และลำดับที่ของกฎ รวมทั้งตารางจะแสดง ELSE rule ให้ซึ่งจะใช้เนื้อที่บนจอภาพอีกส่วนหนึ่ง

### 3.4 ถ้าเลือก T หรือ TABLE

#### 3.4.1 สร้างกรอบรับค่าของตารางตัดสินใจ

3.4.2 กำหนดตำแหน่งของกฎต่าง ๆ ที่ปรากฏบนจอภาพ กำหนดตำแหน่งเริ่มต้นของเงื่อนไข กำหนดตำแหน่งเริ่มต้นของการกระทำ

3.4.3 แสดงข้อมูลของกฎในตาราง (ถ้ามี) ในกรณีที่เปิดใหม่ข้อมูลใหม่ช่องต่าง ๆ จะว่างให้ใส่ข้อมูล

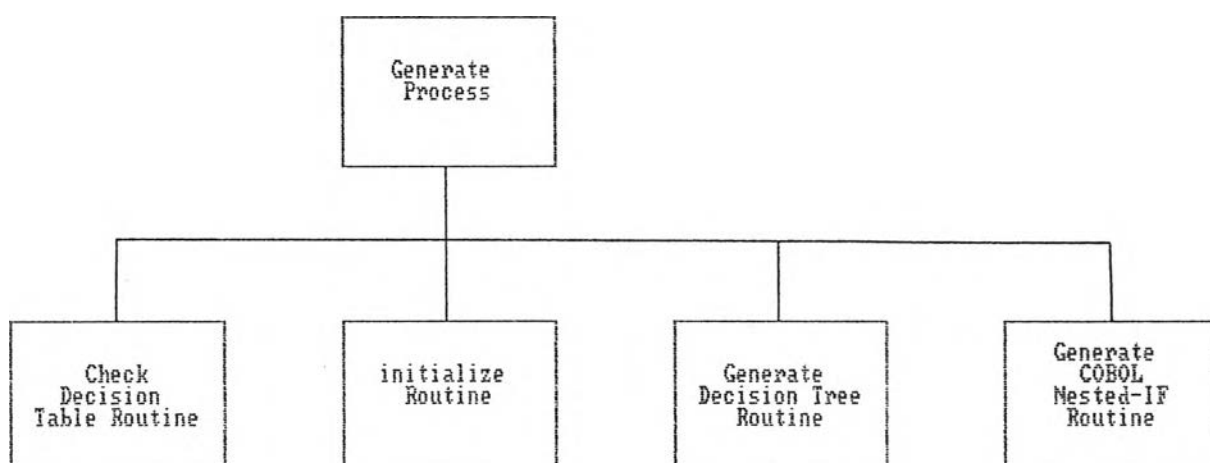
3.4.4 ทำการรับกฎหรือแก้ไขกฎ รายการบางรายการของกฎในส่วนของเงื่อนไขกำหนดให้รับเฉพาะค่า Y, N หรือ - รายการของกฎในส่วนของ การกระทำกำหนดให้รับเฉพาะค่า X หรือ Space เท่านั้น การแก้ไขข้อมูลในตารางทำได้โดยการป้อนตัวอักษรที่กล่าวมาแล้วค่าใดค่าหนึ่งกับบนค่าที่ต้องการแก้ไข เงื่อนไขในช่อง ELSE rule โปรแกรมกำหนดไม่รับข้อมูลแต่ในส่วนการกระทำใส่ข้อมูลได้ โดยข้อมูลทั้งหมดที่ได้แก้ไขจะนำลงเก็บในหน่วยความจำบนพื้นที่ที่ได้เตรียมไว้เก็บข้อมูลชั่วคราวเพื่อนำไปใช้ในขั้นตอนของการสร้างส่วนของโปรแกรมภาษาโคบอล

## โปรแกรมสร้างส่วนของโปรแกรมภาษาโคบอล (Generate Program)

### 1. หน้าที่ของโปรแกรม

ทำการสร้างส่วนของโปรแกรมภาษาโคบอลจากข้อมูลของตารางตัดสินใจที่ได้เก็บไว้ในหน่วยความจำ โดยส่วนของโปรแกรมจะมีรูปเป็น Nested-IF

### 2. ผังโครงสร้างของโปรแกรม



รูปที่ 4.4 ผังแสดงโครงสร้างของโปรแกรมสร้างส่วนของโปรแกรมภาษาโคบอล

### 3. รายละเอียดของโปรแกรม มีขั้นตอนดังต่อไปนี้

#### 3.1 การตรวจสอบตารางตัดสินใจ

ในส่วนนี้จะทำการตรวจสอบข้อมูลของตารางตัดสินใจที่อยู่ในหน่วยความจำหลัก ซึ่งเป็นข้อมูลที่ได้มาจากการจัดการแฟ้มข้อมูลรวมทั้งขั้นตอนการแก้ไขแฟ้มข้อมูล การตรวจสอบจะมี 2 ขั้นตอน คือ

##### 3.1.1 ตรวจสอบความสมบูรณ์ของตารางตัดสินใจ

เริ่มจากการตรวจสอบว่าจำนวนเงื่อนไข จำนวนการกระทำ และจำนวนกฎจะต้องไม่เท่ากับ 0 ถ้ามีค่าใดค่าหนึ่งเป็น 0 จะแสดง Error Message ทางจอภาพ และจบโปรแกรม หลังจากนั้นจะตรวจสอบค่าในตารางในส่วนของเงื่อนไขว่ามีค่าเป็น Y ,N หรือ - จะตรวจสอบทุกรายการถ้ามีรายการหนึ่งรายการใดมีค่าไม่ตรงกับที่กำหนดข้างต้น จะแสดง Error Message และจบโปรแกรม

##### 3.1.2 ตรวจสอบความกำกวม(Ambiguity) ของตารางตัดสินใจ

จะตรวจสอบค่าในตารางตัดสินใจที่ละ 2 กฎโดยจะตรวจสอบค่าของเงื่อนไขทุก ๆ เงื่อนไขว่าตรงกันหมดหรือไม่ จะตรวจสอบเช่นนี้กับกฎทุกคู่ในตารางจนครบ ถ้าพบว่ามีคู่หนึ่งคู่ใดที่มีค่าของเงื่อนไขตรงกันทั้งหมดจะตรวจสอบการกระทำของกฎทั้งคู่ต่อ โดย

ก) ถ้าค่าของการกระทำตรงกันทั้งหมด แสดงว่าเกิดการซ้ำซ้อน (Redundancy) กับกฎทั้งคู่ จะแสดง Error Message บอกถึงกฎทั้งคู่ที่เกิดกรณีซ้ำซ้อน แล้วจบโปรแกรม

ข) ถ้าค่าของการกระทำแตกต่างกันแสดงว่าเกิดการขัดแย้งกัน (Contradiction) กับกฎทั้งคู่ จะแสดง Error Message บอกถึงกฎที่ขัดแย้งกันนั้น แล้วจบโปรแกรม

### 3.2 การเริ่มต้นเตรียมระบบ

3.2.1 จะทำการลบล้างหน่วยความจำในส่วนที่เก็บต้นไม้ตัดสินใจที่สร้างก่อนหน้านี้(ถ้ามีการสร้าง) และส่วนของโปรแกรมที่สร้างไว้ด้วย

3.2.2 จะทำการทำให้จอภาพว่าง แสดงจอภาพเริ่มการสร้างส่วนของโปรแกรม โดยพิมพ์คำว่า 'Generate Screen' ที่บรรทัดแรกของจอภาพ

3.2.3 กำหนดค่าเริ่มต้นทั้งหมดที่ใช้ในการสร้างต้นไม้ตัดสินใจ

3.2.3.1 ทำการ Allocate ที่ให้กับ Head\_node ที่สำหรับเก็บตารางค่าของเงื่อนไขของตารางตัดสินใจ (val\_tbl) รวมทั้งที่เก็บตารางเก็บตำแหน่งของกฎและเงื่อนไข (rule\_arr และ cond\_arr) และทำการกำหนดตัวชี้ (pointer) จาก Head\_node ไปยังตารางเก็บค่าของเงื่อนไข และแถวลำดับ (array) เก็บตำแหน่งของกฎและเงื่อนไข

3.2.3.2 กำหนดค่าเริ่มต้นให้กับ Head\_node

### 3.3 การสร้างต้นไม้ตัดสินใจ

จะวนทำหัวข้อต่อไปนี้จะมีการกำหนดจุดแตกกิ่ง (node) ที่จะสร้างตารางย่อย โดยเริ่มต้นจาก Head\_node คือเริ่มต้นจากรากของต้นไม้ก่อน จะทำไปเรื่อยๆ จนกว่าจะทำให้สาขาทางด้านซ้ายและขวาของจุดแตกกิ่งมีค่าเป็น NULL

3.3.1 คำนวณค่าของ Column Count (CC) ของกฎแต่ละกฎของตารางตัดสินใจที่จุดแตกกิ่งที่อยู่

3.3.2 คำนวณค่าของ Dash Count (DC) ของกฎแต่ละกฎของตารางตัดสินใจที่จุดแตกกิ่งที่อยู่

3.3.3 เลือกเงื่อนไขที่จะใช้ทดสอบเพื่อแยกตารางที่จุดแตกกิ่งที่เป็นตารางย่อยซ้ายขวาโดยจะเลือกจากเงื่อนไขที่มีค่า Dash count น้อยที่สุด ถ้ามีเงื่อนไขที่มีค่าต่ำสุดหลายเงื่อนไขจะเลือกเงื่อนไขลำดับแรก

3.3.4 สร้างตารางย่อยจากตารางที่จุดแตกกิ่งที่เป็นการแตกสาขาของต้นไม้มิสร้างตารางย่อยทางซ้ายและขวา โดยทำตามลำดับดังนี้

3.3.4.1 กำหนดค่าเริ่มต้นคือตำแหน่งของเงื่อนไขที่เลือกไว้ทดสอบ

3.3.4.2 กำหนดค่าของเงื่อนไขที่จะใช้ทดสอบค่าทางด้านซ้าย( $l\_select$ ) ว่าเป็น Y หรือ E และค่าทางด้านขวา( $r\_select$ ) ว่าเป็น N หรือ E ซึ่งได้มาจากการตรวจสอบค่าของเงื่อนไขที่ถูกเลือกว่ามีรายการที่มีค่าของ Y, N หรือ -

3.3.4.3 สร้างจุดแตกกิ่งใหม่ที่ชี้ไปยังตารางย่อยที่สร้างใหม่ เป็นการสร้างสาขาทางด้านซ้ายของตาราง ขั้นตอนนี้จะไม่สร้างตารางย่อยใหม่ ถ้าเกิดกรณีใด กรณีหนึ่งใน 3 กรณีดังนี้

- ถ้าค่าของเงื่อนไขที่ใช้ทดสอบเป็น E (ค่าที่ได้จากข้อ 3.3.4.2)

- ถ้าเงื่อนไขที่เหลืออยู่เพียง 1 เงื่อนไขซึ่งก็คือ เงื่อนไขที่เลือก

- ถ้ามีกฎเหลืออยู่เพียง 1 กฎ และกฎนั้นมีค่าของเงื่อนไขที่ไม่ใช่เงื่อนไขที่เลือกมีค่าเป็นขีด ("-")

การสร้างตารางย่อยทำตามขั้นตอนดังต่อไปนี้

ก) การจัดสรร (Allocate) ที่สำหรับจุดแตกกิ่งใหม่ รวมทั้งที่สำหรับเก็บตารางย่อยใหม่ และตารางที่เก็บตำแหน่งของกฎและเงื่อนไขที่ตรงกับตารางตัดสินใจเริ่มต้น ( $head\_node$  นี้) ซึ่งจะใช้ประโยชน์ของตำแหน่งนี้ในตอนการสร้างส่วนของโปรแกรม

ข) กำหนดค่าให้จุดแตกกิ่งที่สร้างใหม่ ซึ่งจำนวนเงื่อนไขจะลดลง 1 เงื่อนไข จำนวนกฎจะเท่ากับจำนวนของค่าของเงื่อนไขที่เลือกว่ามีค่าตรงกับค่าที่กำหนดไว้ทดสอบก็ค่า เช่นทางด้านซ้ายกำหนดให้ค่าที่ใช้ทดสอบเป็น Y ดังนั้นจะตรวจสอบค่าของเงื่อนไขที่เลือกว่าเป็น Y หรือ - จำนวนที่ค่า ค่าที่ได้จะเป็นค่า ของจำนวนกฎของตารางใหม่เป็นต้น รวมทั้งกำหนดให้ตัวชี้ของจุดแตกกิ่งชี้ไปยังตารางย่อยที่



สร้างใหม่

ค) กำหนดค่าทั้งหมดของตารางย่อย ตารางย่อยจะมีเฉพาะกฎที่มีค่าของเงื่อนไขที่เลือกตรงกับค่าที่กำหนดไว้ ("Y" หรือ "-") โดยจะตัดแถวของเงื่อนไขที่เลือกออกจากตารางย่อย

ง) กำหนดค่าให้  $l\_node$  ซึ่งเป็นตัวชี้ของจุดแตกกิ่งที่สร้างจุดแตกกิ่งใหม่และตารางย่อยใหม่นี้ให้ที่มาจากจุดแตกกิ่งที่สร้างใหม่นี้

3.3.4.4 สร้างจุดแตกกิ่งใหม่ทางด้านขวาที่จะชี้ไปยังตารางย่อยที่สร้างใหม่ ค่าของเงื่อนไขที่ใช้ทดสอบทางด้านขวามีค่าเป็น  $N$  ขึ้นตอนการทำงานจะเหมือนขั้นตอนที่ 3.3.4.3 ทั้งหมด จะแตกต่างกันที่กฎที่จะเลือกใส่ตารางย่อยต้องเป็นกฎที่มีค่าของเงื่อนไขที่เลือกไว้เป็น  $N$  หรือ - เท่านั้น แล้วตัดแถวของเงื่อนไขที่ใช้ทดสอบออกจากตารางดังนั้นเงื่อนไขในตารางใหม่จะมีจำนวนลดลง 1 เงื่อนไข เมื่อสร้างจุดแตกกิ่งและตารางย่อยทางขวาเสร็จเรียบร้อยแล้ว จะกำหนดให้  $r\_node$  ซึ่งเป็นตัวชี้ของจุดแตกกิ่งที่นำมาสร้างจุดแตกกิ่งใหม่นี้ ชี้มาที่จุดแตกกิ่งทางขวาที่สร้างใหม่

ตารางย่อยที่สร้างขึ้นมาใหม่ทั้งซ้ายและขวานี้จะมีจำนวนเงื่อนไขลดลง 1 เงื่อนไขทุกครั้ง ต่อจากนั้นจะทำการตรวจสอบทางด้านซ้ายของตน ไม่ตัดสินใจว่าเป็น NULL ( $l\_node = \text{NULL}$ ) หรือไม่ ถ้าไม่สร้างต้นไม้มัดตัดสินใจต่อไปโดยใช้จุดแตกกิ่งทางด้านซ้ายเป็นการเริ่มต้นการทำงานครั้งต่อไปซึ่งจะเริ่มทำตั้งแต่ขั้นตอนที่ 3.3.1 ซึ่งจะทำการเรียก Routine ของตัวเองเช่นนี้เรื่อยๆ จนกว่าจุดแตกกิ่งทางด้านซ้ายจะเป็น NULL นั่นคือสุดปลายต้นไม้ทางด้านซ้าย แล้วจะทำการตรวจสอบจุดแตกกิ่งทางด้านขวาด้วยวิธีเดียวกันและจะทำขั้นตอนทุกอย่างเหมือนทางด้านซ้ายแตกต่างกันที่ค่าที่ใช้ทดสอบ การทำงานของขั้นตอนที่ 3.3.4 นี้เป็นแบบ Recursive ซึ่งจะได้ต้นไม้มัดตัดสินใจที่สมบูรณ์ที่ได้มาจากการแปลงตารางตัดสินใจที่กำหนด

### 3.4 การสร้างส่วนของโปรแกรมภาษาโคบอล

3.4.1 กำหนดให้สิ่งออก (output) เป็นจอภาพ

3.4.2 กำหนดค่าเริ่มต้นให้กับตัวแปรเก็บจำนวนบรรทัด

### 3.4.3 เริ่มการทำงานสร้าง Nested-IF ดังนี้

#### 3.4.3.1 การสร้างส่วน IF Statement ที่มีรูปแบบต่อไปนี้

แบบต่อไปนี้

#### IF Condition

จะเก็บคำสั่งลงบัฟเฟอร์ที่ละบรรทัดก่อนที่จะนำออกมาแสดงทางจอภาพ การสร้างคำสั่งนี้จะมีการกำหนดคอลัมน์เริ่มต้นให้กับคำสั่งตามข้อกำหนดของภาษา และจะเลื่อนเข้าอีกตามลำดับขั้นของการซ้อนของเงื่อนไข เพื่อให้อ่านส่วนที่สร้างเสร็จแล้วได้ง่าย ใส่รายการเงื่อนไขที่ใช้ทดสอบซึ่งมาจากค่าของจุดแตกกิ่งของต้นไม้ตัดสินใจที่สร้างขึ้น (ขั้นตอนที่ 3.3) เงื่อนไขแรกที่ทดสอบจะมาจาก head\_node

3.4.3.2 ตรวจสอบ l\_node ของจุดแตกกิ่งนั้นว่าเป็น NULL หรือไม่ (เป็นการตรวจสอบทางด้านซ้ายของต้นไม้ตัดสินใจ) ถ้าไม่ทำขั้นตอนที่ 3.4.3 อีก โดยเงื่อนไขที่ใส่จะเป็นเงื่อนไขที่ใช้ทดสอบของจุดแตกกิ่งที่ถูกชี้โดย l\_node จะวนทำเช่นนี้ตามทางเดินของต้นไม้ตัดสินใจ ซึ่งเป็นการสร้าง IF ของเงื่อนไขทางซ้ายของต้นไม้ตัดสินใจซ้อนกันจนสุดปลายต้นไม้ โปรแกรมส่วนนี้ทำงานแบบ Recursive ถ้าตรวจสอบแล้วเป็น NULL จะทำขั้นตอนถัดไป

#### 3.4.3.3 การสร้างการกระทำของเงื่อนไขซึ่งมีรูปแบบ

ดังนี้

```
IF CONDITION1
  IF CONDITION2
    .
    .
    .
  ACTION
```

การกระทำนี้จะ ได้จากการหากฎที่มี การทดสอบเงื่อนไขตรงกับที่ได้ทำการทดสอบมาแล้ว ซึ่งทำได้จากการตรวจสอบจากต้นไม้ตัดสินใจ ตรวจสอบจากจุดแตกกิ่งสุดท้ายว่า l\_select ที่เลือกจะตรงกับค่าเงื่อนไขของกฎใด

เมื่อหากฎได้แล้วจะหาว่ากฎนี้มีการกระทำไบบ้าง นำการกระทำทุกการกระทำมาแสดงออกทางจอภาพ ซึ่งมีการเว้นช่องว่างหน้าการกระทำแบบเดียวกับเงื่อนไข ซึ่งจะเว้นมากน้อยตามขั้นของการซ้อน ถ้าตรวจสอบแล้วพบว่ากฎนั้น ไม่มีการกระทำจะพิมพ์คำว่า NEXT SENTENCE ถ้าเงื่อนไขที่ทดสอบทั้งหมดไม่ตรงกับกฎในตารางแสดงว่าเป็น ELSE rule จะตรวจสอบว่ากฎ ELSE rule มีการกระทำไบบ้างจะเอามาพิมพ์ แต่ถ้าไม่มีการกระทำจะพิมพ์คำว่า NEXT SENTENCE

3.4.3.4 พิมพ์คำว่า ELSE ซึ่งจะย่อหน้าตามขั้นของการซ้อนกันของเงื่อนไข

3.4.3.5 ตรวจสอบ r\_node ของจุดตกที่นั้นว่าเป็น NULL หรือไม่ (เป็นการตรวจสอบทางด้านขวาของต้นไม้ตัดสินใจ) ถ้าไม่จะทำขั้นตอนที่ 3.4.3 อีก โดยเงื่อนไขที่ใส่จะเป็นเงื่อนไขที่ใช้ทดสอบของจุดตกที่นั้นโดย r\_node จะวนทำเช่นนี้ตามทางเดินของต้นไม้ตัดสินใจ ซึ่งเป็นการสร้าง IF ของเงื่อนไขทางขวาของต้นไม้ตัดสินใจซ้อนกันจนสุดปลายต้นไม้ โปรแกรมส่วนนี้ทำงานแบบ Recursive ถ้าตรวจสอบแล้วเป็น NULL จะทำขั้นตอนถัดไป

3.4.3.6 สร้างการกระทำของเงื่อนไข จะตรวจหาจากจุดตกที่สุดท้ายว่า r\_select ที่เลือกจะตรงกับค่าเงื่อนไขของกฎใด เมื่อหากฎได้แล้วจะหาว่ากฎนี้มีการกระทำไบบ้าง นำการกระทำทุกการกระทำ มาแสดงทางจอภาพซึ่งมีการเว้นช่องว่างหน้าการกระทำแบบเดียวกับเงื่อนไข ซึ่งจะเว้นมากน้อยตามขั้นของการซ้อน ถ้าตรวจสอบแล้วพบว่ากฎนั้น ไม่มีการกระทำจะพิมพ์คำว่า NEXT SENTENCE ถ้าเงื่อนไขที่ทดสอบทั้งหมดไม่ตรงกับกฎในตารางแสดงว่าเป็น ELSE rule จะตรวจสอบว่ากฎ ELSE rule มีการกระทำไบบ้างจะเอามาพิมพ์ แต่ถ้าไม่มีการกระทำจะพิมพ์คำว่า NEXT SENTENCE

3.4.3.7 พิมพ์บรรทัดสุดท้ายของ Nested-IF จะตรวจสอบก่อนว่าได้สร้างส่วนของโปรแกรมภาษาโคบอลจนหมดแล้ว จะพิมพ์จุด(.) ให้กับบรรทัดสุดท้าย เป็นการจบการทำงานของขั้นตอนนี้

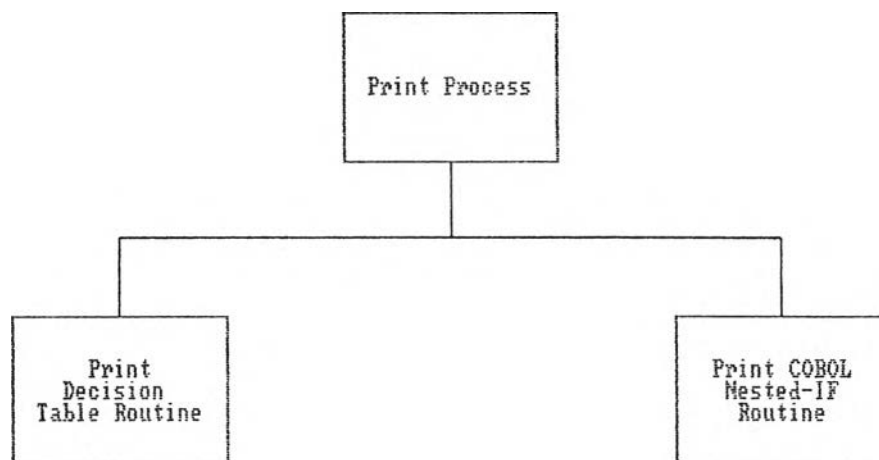
การสร้าง Nested-IF นี้จะเป็นการสร้างจากต้นไม้ตัดสินใจโดยเริ่มจากรากและไปทางด้านซ้าย (YES) จนสุดปลายต้นไม้แล้วจะสร้างทางด้านขวา (NO) จนหมดทั้งสาขาทางด้านขวา เมื่อได้ครบหมดจะใส่จุดให้เป็นการสิ้นสุดคำสั่ง

## โปรแกรมพิมพ์ตารางและส่วนของโปรแกรมภาษาโคบอล (Print Program)

### 1. หน้าที่ของโปรแกรม

พิมพ์ส่วนของโปรแกรมภาษาโคบอลที่สร้างขึ้นและตารางตัดสินใจออกทางเครื่องพิมพ์ หรือเก็บลงแฟ้มข้อมูล

### 2. ผังโครงสร้างของโปรแกรม



รูปที่ 4.5 ผังแสดงโครงสร้างของโปรแกรมพิมพ์ตารางและส่วนของโปรแกรม

### 3. รายละเอียดของโปรแกรม

3.1 แสดงรายการย่อยของรายการเลือกแบบติงลงของรายการหลัก Print และรับข้อมูลจากแผงแป้นอักขระว่าเลือกรายการใด

3.1 ถ้าเลือกการพิมพ์ตารางตัดสินใจโดย

3.1.1 พิมพ์ตารางตัดสินใจออกทางเครื่องพิมพ์

โปรแกรมจะจัดการนำข้อมูลทั้งหมดของตารางตัดสินใจที่เก็บไว้ในหน่วยความจำพิมพ์ออกทางเครื่องพิมพ์ รายละเอียดที่จะพิมพ์มีดังนี้

ก) บรรทัดที่ 1 จะแสดงข้อความว่า

" \*\*\* DECISION TABLE \*\*\* "

ข) จำนวนเงื่อนไข

ค) จำนวนการกระทำ

ง) จำนวนกฎ

จ) รายการของเงื่อนไข

ฉ) รายการของการกระทำ

ช) ตารางตัดสินใจ

3.1.2 เก็บตารางตัดสินใจลงแฟ้มข้อมูล

3.1.2.1 ตรวจสอบชื่อแฟ้มข้อมูลว่ากำหนดไว้หรือยัง

3.1.2.2 รับชื่อแฟ้มข้อมูลในกรณีที่ยังไม่ได้กำหนดไว้ (ชื่อมีความยาวสูงสุด 8 ตัวอักษร) ระบบที่สร้างนี้จะกำหนดให้ชื่อแฟ้มข้อมูลที่เก็บตารางตัดสินใจและส่วนของโปรแกรมนั้นชื่อเดียวกันแต่มี extension ต่างกัน การกำหนดชื่อไม่ต้องใส่ extension โปรแกรมจะเพิ่ม extension .LST ให้กับแฟ้มข้อมูล

3.1.2.3 เก็บข้อมูลทั้งหมดของตารางตัดสินใจในหน่วยความจำลงแฟ้มข้อมูล LST ที่เปิดไว้แล้ว ข้อมูลที่เก็บมีรายละเอียดเหมือนที่เก็บในหัวข้อ

3.1.1

### 3.2 ถ้าเลือกการพิมพ์ส่วนของโปรแกรมภาษาโคบอล โดย

#### 3.2.1 พิมพ์ส่วนของโปรแกรมภาษาโคบอลออกจากเครื่องพิมพ์

จะทำการพิมพ์ส่วนของโปรแกรมภาษาโคบอลในรูปของ Nested-IF ที่สร้างจากข้อมูลตารางตัดสินใจที่กำหนดออกจากเครื่องพิมพ์

#### 3.2.2 เก็บส่วนของโปรแกรมภาษาโคบอลลงแฟ้มข้อมูล

3.2.2.1 ตรวจสอบชื่อแฟ้มข้อมูลว่ากำหนดไว้หรือยัง

3.2.2.2 รับชื่อแฟ้มข้อมูลในกรณีที่ยังไม่ได้กำหนดไว้ (ชื่อ มีความยาวสูงสุด 8 ตัวอักษร) ระบบที่สร้างนี้จะกำหนดให้ชื่อแฟ้มข้อมูลที่เก็บตารางตัดสินใจ และส่วนของโปรแกรมมีชื่อเดียวกันแต่มี extension ต่างกัน การกำหนดชื่อไม่ต้องใส่ extension โปรแกรมจะเพิ่ม extension .COB ให้กับแฟ้มข้อมูล

3.2.2.3 เก็บส่วนของโปรแกรมภาษาโคบอลในรูป Nested -IF ที่สร้างขึ้นซึ่งอยู่ในหน่วยความจำลงแฟ้มข้อมูล COB ที่เปิดแฟ้มไว้แล้ว