



บทที่ 1

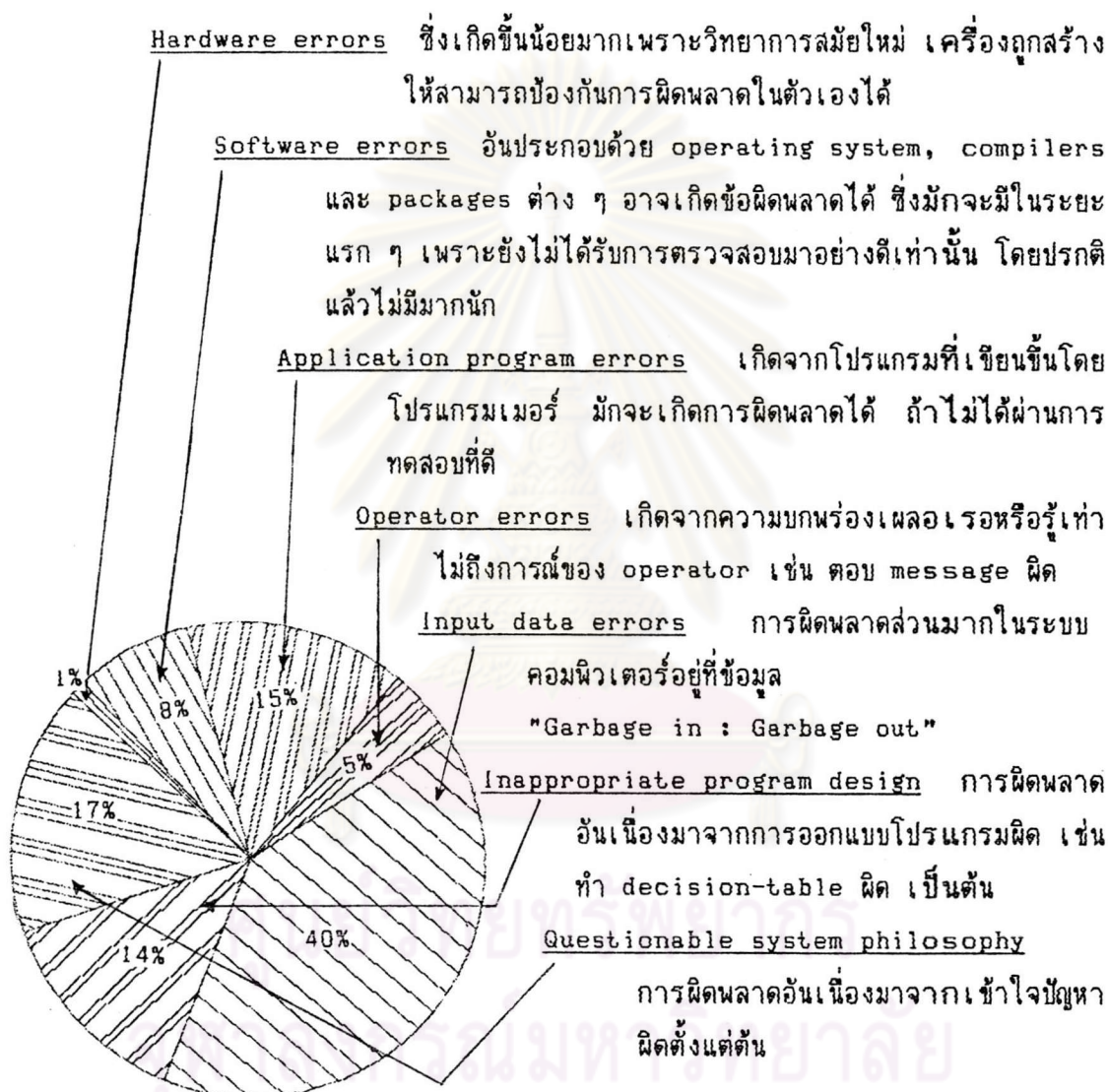
บทนำ

1.1 ความเป็นมาของปัญหา

ในปัจจุบันนี้ได้มีการนำคอมพิวเตอร์มาประยุกต์กับงานด้านต่าง ๆ เพื่ออำนวยความสะดวกสบายเป็นอย่างมาก ทั้งนี้เป็นผลมาจากได้มีการพัฒนาให้เครื่องคอมพิวเตอร์มีความคล่องตัวในการนำไปใช้งานได้มากขึ้น จะเห็นได้จากไมโครคอมพิวเตอร์ที่ใช้กันอย่างแพร่หลายในปัจจุบันนี้ มีความสามารถในการทำงานใกล้เคียงกับคอมพิวเตอร์ขนาดใหญ่แต่ราคาถูกลงกว่ามาก นอกจากนี้ ยังมีโปรแกรมสำเร็จรูปที่ใช้กับงานด้านต่าง ๆ จำหน่ายในราคาค่อนข้างย่อมเยาอีกด้วย

อย่างไรก็ตามแม้ว่ามีการพัฒนาการใช้งานคอมพิวเตอร์ประมวลผลข้อมูลได้อย่างสะดวกและลดความยุ่งยากลงไปมากแล้วก็ตาม แต่ก็พบอยู่เป็นประจำว่าเกิดความผิดพลาดในการประมวลผลข้อมูลด้วยเครื่องคอมพิวเตอร์ ซึ่งสาเหตุส่วนใหญ่มาจากการนำเข้าของข้อมูลที่ผิดพลาด (Input data error) จะเห็นได้จากงานศึกษาวิจัยที่มีลักษณะการประมวลผลตามลำดับขั้นตอนการดำเนินงานที่คล้ายคลึงกัน คือ เก็บรวบรวมข้อมูลดิบลงรหัส บันทึกข้อมูลลงสื่อเก็บข้อมูล ตรวจสอบความถูกต้องของข้อมูลที่บันทึกไว้ แก้ไขข้อมูลในสื่อเก็บข้อมูลเมื่อพบว่ามีข้อมูลผิดพลาด และขั้นสุดท้ายจึงทำการประมวลผลข้อมูล ดังนั้น ถ้าหากข้อมูลนำเข้าบันทึกในสื่อเก็บข้อมูลไม่ถูกต้อง รายงานผลลัพธ์ที่ได้จากการประมวลผลย่อมผิดพลาดไปด้วย (Garbage in garbage out) ยิ่งข้อมูลที่มีปริมาณมาก ๆ ความผิดพลาดที่จะเกิดจากการบันทึกข้อมูลลงในสื่อเก็บข้อมูลย่อมเกิดขึ้นได้มากด้วย ถ้าหากใช้คนตรวจสอบความถูกต้อง จะต้องเสียเวลาและแรงงานเพิ่มมากขึ้น ยิ่งไปกว่านั้นแม้กษายังมีข้อผิดพลาดหลงเหลืออยู่ในข้อมูล ทั้งนี้เพราะความสามารถของมนุษย์มีข้อจำกัดในการตรวจสอบความถูกต้องของข้อมูลจำนวนมาก โดยเฉพาะอย่างยิ่งข้อมูลประเภทตัวเลขล้วน ๆ แล้ว ยังมีข้อผิดพลาดสูงกว่าปกติ

1.1.1 แหล่งต่างๆ ที่ก่อให้เกิดการผิดพลาดในการประมวลผลด้วยคอมพิวเตอร์ [5]
(Sources of errors in data processing system)



รูปที่ 1.1 แสดงแหล่งที่ก่อให้เกิดการผิดพลาดในการประมวลผลด้วยคอมพิวเตอร์

ดังจะเห็นแล้วว่า แหล่งที่ก่อให้เกิดการผิดพลาดในระบบการประมวลผลด้วยคอมพิวเตอร์มีแหล่งใหญ่มาจากตัวข้อมูลนั่นเอง

1.1.2 การควบคุมความถูกต้อง (Accuracy Control)

การควบคุมให้ข้อมูล มีความถูกต้องก่อนที่จะนำไปทำการประมวลผล (Data Processing) ด้วยเครื่องคอมพิวเตอร์ เป็นสิ่งจำเป็นอย่างยิ่งสำหรับกรณีที่มีข้อมูลเป็นจำนวนมาก

การควบคุมส่วนนำข้อมูลเข้า (Input Control)

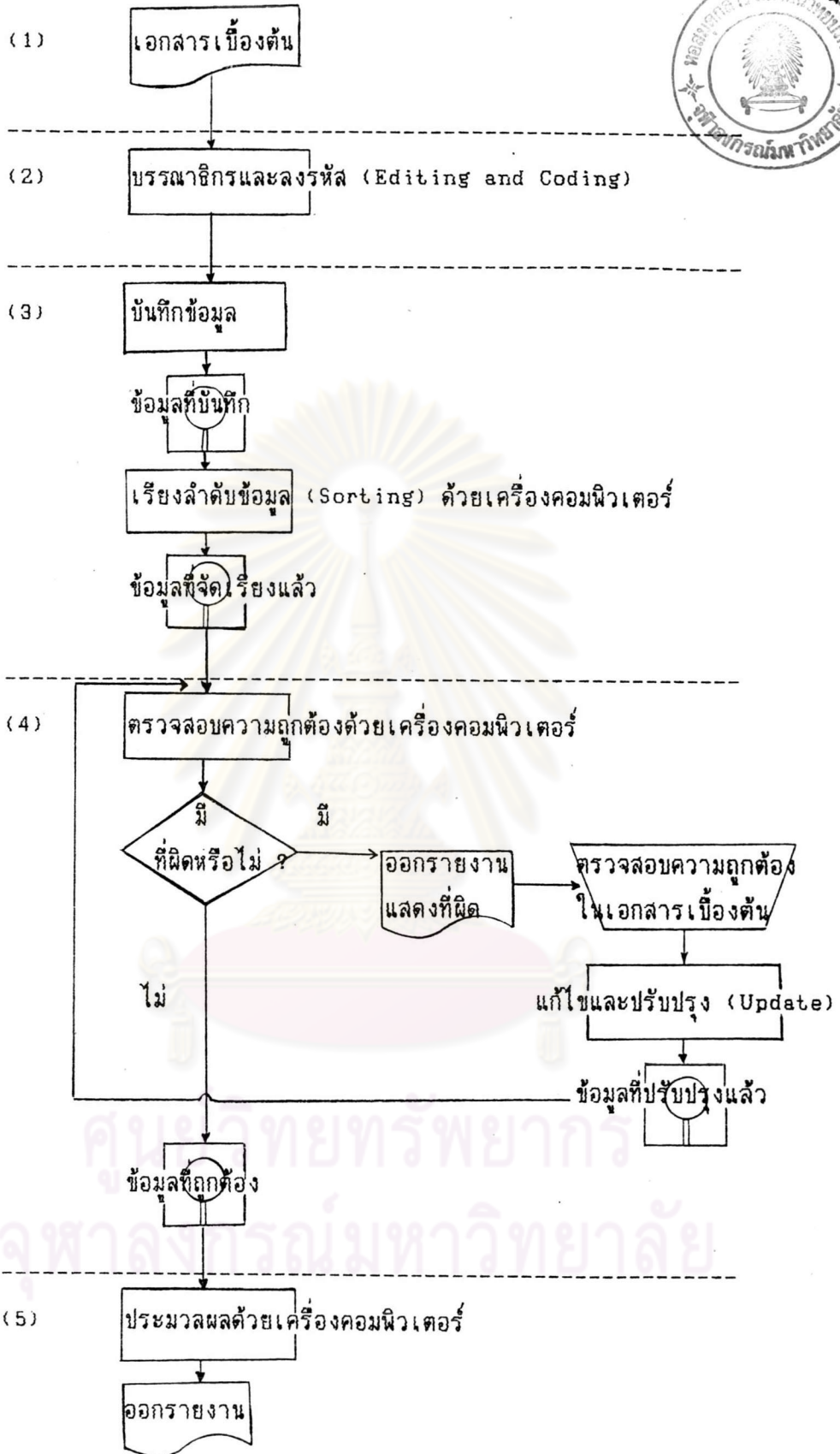
1. ในการประมวลผลที่มีเอกสารเบื้องต้น (Source Documents) นั้น การควบคุมประกอบด้วย การให้หมายเลขกำกับแบบสอบถาม (Pre-numbered) กำหนดรหัสที่จำเป็นบางส่วนไว้ในแบบสอบถาม ควบคุมจำนวนแบบสอบถามที่ส่งออกและได้รับในแต่ละชุด

2. การตรวจสอบรายการและลงรหัส ในแบบสอบถามที่ได้รับจากขั้นที่ 1 ตลอดจนการตรวจสอบจำนวนแบบสอบถาม โดยทั่วไปจะมีการฝึกอบรมเจ้าหน้าที่ที่ทำการตรวจสอบรายการและลงรหัส เพื่อลดความผิดพลาดในการปฏิบัติงาน

3. เมื่อได้ตรวจสอบรายการและลงรหัสแล้ว ขั้นต่อไปจะเป็นการบันทึกรายการข้อมูลลงจานแม่เหล็ก หรือเทปแม่เหล็ก (Key-to-disk or Key-to-tape) ก่อนที่จะนำไปจัดเรียงลำดับ (Sorting) ข้อมูลด้วยเครื่องคอมพิวเตอร์ เพื่อความรวดเร็วและเหมาะสมแก่การปฏิบัติงานในขั้นต่อไป

4. เมื่อบันทึกและจัดเรียงลำดับข้อมูลแล้ว จะต้องนำมาตรวจสอบความถูกต้องด้วยเครื่องคอมพิวเตอร์ (Machine Editing) ก่อนที่จะนำข้อมูลไปประมวลผลต่อไป

การควบคุมส่วนนำข้อมูลเข้าของการประมวลผล สามารถอธิบายเป็นแผนภาพ ดังในรูปที่ 1.2



รูปที่ 1.2 แสดงการควบคุมส่วนนำข้อมูลเข้า

1.1.3 งานวิจัยที่เกี่ยวข้อง

1.1.3.1 ฝ่ายพัฒนาและอบรมคอมพิวเตอร์ สำนักงานสถิติแห่งชาติได้เล็งเห็นความสำคัญ และคุณค่าของการตรวจสอบความถูกต้องของข้อมูล จึงได้มีการเขียนโปรแกรมขึ้น เพื่อใช้ในการบรรณาธิกรข้อมูลด้วยคอมพิวเตอร์ เรียกว่า โปรแกรมสำเร็จรูป VSEDIT [3] ซึ่งโปรแกรมที่เขียนขึ้นต้องใช้กับเครื่องคอมพิวเตอร์ขนาดใหญ่ การทำงาน จะวิ่งภายใต้ระบบควบคุมการปฏิบัติการ (Operating System) ของเครื่องนั้น

โปรแกรมดังกล่าว ประกอบด้วยคำสั่งที่ใช้งานทั้งหมด 6 คำสั่ง ได้แก่ คำสั่ง EDITVS, POS, IF, IFF, CRDNUM และ ENDEDT โดยแต่ละคำสั่งประกอบด้วย 2 ส่วน คือ operation name และ operand ซึ่งมีการใช้งานดังนี้

1) คำสั่ง EDITVS เป็นคำสั่งให้ตรวจสอบหมายเลขประจำระเบียบว่ามีค่าอยู่ในนัลย์ที่กำหนดหรือไม่ มีการเข้าชื้อนหรือขาดหายไปหรือไม่ กรณีที่แต่ละชุดข้อมูลมีมากกว่า 1 ระเบียบ รูปแบบคำสั่งเป็นดังนี้

$$\text{EDITVS } C = (c_1, c_2), L = (l_1, l_2), \text{MIN} = (m_1, m_2), \\ \text{MAX} = (x_1, x_2), \text{INDEV} = y, A = (a, b)$$

Operands ของ operation EDITVS ประกอบด้วย

$C = (c_1, c_2)$ เป็นการบอกตำแหน่งคอลัมน์เริ่มต้นของหมายเลขประจำข้อมูล (identification) ที่จะตรวจสอบแบ่งได้สูงสุด 2 ช่วง โดยเรียง Major และ Minor จากซ้ายไปขวา

$L = (l_1, l_2)$ เป็นการบอกความยาวของหมายเลขประจำข้อมูล ที่ระบุไว้ใน C operand โดยมีจำนวน 1 เท่ากับที่แบ่งใน C operand

$\text{MIN} = (m_1, m_2)$ เป็นการบอกค่าต่ำสุดของแต่ละเขตข้อมูล (field) ของหมายเลขประจำข้อมูลทีระบุไว้ใน C และ L operands โดยมีจำนวนช่วงเท่ากันด้วย ค่า m ต้องเป็นตัวเลขเท่านั้น

- MAX = (x_1, x_2) เป็นการบอกค่าสูงสุดของแต่ละเขตข้อมูล (field) ของหมายเลขประจำข้อมูลที่ระบุไว้ใน C และ L operands โดยมีจำนวนช่วงเท่ากันด้วย ค่า x ต้องเป็นตัวเลขเท่านั้น
- INDEV = y เป็นการบอกว่าสื่อที่ใช้บันทึกข้อมูลเป็นบัตร หรือเทป ปรกติถ้าไม่บอกจะถือว่าเป็นเทป
- A = (a, b) เป็นการบอกความยาวของข้อมูลที่จะตรวจสอบ โดย
 a เป็นค่าความยาวของระเบียน (record)
 b เป็นค่าความยาวของบล็อก (block)

2) คำสั่ง POS เป็นคำสั่งใช้ตรวจสอบค่าที่เป็นไปได้ (Possible codes) ของแต่ละเขตข้อมูลที่กำหนด รูปแบบคำสั่งเป็นดังนี้

$$\text{POS } C = (i, l, s), N = (a_1 - a_2, a_3 - a_4, \dots, a_{n-1} - a_n),$$

$$A = (a_1 - a_2, a_3 - a_4, \dots, a_{n-1} - a_n)$$

3) คำสั่ง IF เป็นคำสั่งใช้ตรวจสอบความสัมพันธ์กันระหว่างเขตข้อมูลหนึ่งกับอีกเขตข้อมูลหนึ่งเท่านั้น

4) คำสั่ง IFF เป็นคำสั่งใช้ตรวจสอบความสัมพันธ์กันระหว่างเขตข้อมูล ต่อเขตข้อมูล ในลักษณะที่หาความสัมพันธ์ของเขตข้อมูล หรือชุดของเขตข้อมูล

5) คำสั่ง CRDNUM เป็นคำสั่งให้นำหน้าคำสั่งประเภท POS, IF หรือ IFF ของแต่ละระเบียน เพื่อบอกว่าคำสั่งต่อ ๆ ไปที่ตามมาจะให้ตรวจสอบที่ระเบียนใด

6) คำสั่ง ENEDT เป็นคำสั่งการสิ้นสุดของโปรแกรม โดยคำสั่งนี้จะ เป็นคำสั่งสุดท้ายของโปรแกรม ต้องมีอยู่ทุกครั้ง

ลักษณะและข้อกำหนดต่าง ๆ ของโปรแกรม VSEDIT มีดังนี้

1. ใช้กับเครื่องคอมพิวเตอร์ขนาดใหญ่เท่านั้น
 2. โปรแกรมนี้เขียนด้วยภาษาแอสเซมบลีย์ โดยใช้แมคโครฟีเจอร์ (Macro Feature) คำสั่งใช้งานจะต้องมีรูปแบบตามที่กำหนดเท่านั้น ผู้ใช้จะต้องใช้เวลาในการศึกษา และเข้าใจตัวแปรค่าต่าง ๆ ในคำสั่งอย่างถูกต้องแม่นยำก่อน จึงจะใช้งานได้
 3. เนื่องจากคำสั่งในโปรแกรมจะต้องใช้รวมอยู่ในส่วนของภาษาควบคุมงาน (Job Control Language เรียก โดยย่อว่า JCL) ผู้ใช้จะต้องทราบว่าโปรแกรมดังกล่าวจะต้องดำเนินการภายใต้ระบบควบคุมการปฏิบัติการชื่ออะไร เพราะระบบควบคุมการปฏิบัติการของเครื่องคอมพิวเตอร์แต่ละระบบ จะมีภาษาที่ใช้ติดต่อกับระบบ แตกต่างกันไป
 4. ข้อมูลที่ส่งให้ตรวจสอบ ในระเบียบแรกโปรแกรมจะถือว่าเป็นข้อมูลที่ต้อง และไม่ตรวจสอบให้
 5. กรณีที่ข้อมูลแต่ละระเบียนยาวเกินกว่า 90 ไบต์แล้ว โปรแกรมไม่สามารถแสดงผลส่วนที่เกินกว่า 90 ไบต์ได้
 6. กรณีที่มีการใช้คำสั่งเป็นจำนวนมาก จะทำให้ขนาดของโปรแกรมใหญ่มากขึ้นด้วย และถ้าขนาดของโปรแกรมใหญ่เกิน 40 กิโลไบต์แล้ว โปรแกรมจะไม่สามารถทำงานได้
 7. การตรวจสอบความสัมพันธ์ของข้อมูล (Relational check) ข้อมูลต้องอยู่ภายในระเบียบเดียวกัน ไม่สามารถตรวจสอบข้อมูลที่อยู่ข้ามระเบียบได้
- สรุป ความสามารถและคุณสมบัติของ โปรแกรม VSEDIT ดังต่อไปนี้

ตารางที่ 1.1 แสดงคุณสมบัติของโปรแกรม VSEDIT

คุณสมบัติของโปรแกรม	VSEDIT
ใช้กับเครื่องคอมพิวเตอร์ประเภท	เมนเฟรม
การตรวจสอบความถูกต้องของการจัดลำดับ	-
การตรวจสอบความครบถ้วนสมบูรณ์ของระเบียบ	
- ตรวจสอบระเบียบที่ขาดหายไป	X
- ตรวจสอบการซ้ำซ้อนของระเบียบ	X
- ตรวจสอบระเบียบเกิน	-

ตารางที่ 1.1 แสดงคุณสมบัติของโปรแกรม VSEDIT (ต่อ)

คุณสมบัติของโปรแกรม	VSEDIT
การตรวจสอบความถูกต้องของข้อมูล	
- ตรวจสอบค่าเป็นไปได้ของข้อมูล ในช่วงที่กำหนด	X
- ตรวจสอบค่าเป็นไปได้ของข้อมูล ที่มีค่า ไม่ต่อเนื่อง	-
- ตรวจสอบความสัมพันธ์กันของข้อมูล ที่อยู่ภายในระเบียบเดียวกัน	X
- ตรวจสอบความสัมพันธ์กันของข้อมูล ที่อยู่ต่างระเบียบกัน	-
- เลือกตรวจสอบข้อมูลเฉพาะบางระเบียบได้	X
- ตรวจสอบข้อมูลที่เป็นภาษาไทยและอังกฤษ	X
ประเภทของจำนวนระเบียบในแต่ละชุดข้อมูล	
- ตรวจสอบข้อมูลประเภทจำนวนระเบียบคงที่	X
- ตรวจสอบข้อมูลประเภทจำนวนระเบียบไม่คงที่	-
คุณสมบัติอื่น ๆ	
- ขนาดของโปรแกรมจะแปรผันตามจำนวนคำสั่ง ที่ใช้ตรวจสอบ ถ้าขนาดของโปรแกรมใหญ่เกิน 40 กิโลไบต์ จะทำงานไม่ได้	X
- แสดงผลการตรวจสอบข้อมูลของระเบียบ ที่มี ความยาวเกิน 90 ไบต์ได้	-

หมายเหตุ x มีคุณสมบัติ
- ไม่มีคุณสมบัติ



1.1.3.2 โปรแกรม DENTRY [4] เป็นโปรแกรมสำหรับเตรียมข้อมูลบนเครื่องไมโครคอมพิวเตอร์ ซึ่งสามารถใช้บันทึก และตรวจสอบความถูกต้องของข้อมูลที่ป้อนเข้ามาได้ ซึ่งแต่ละระเบียบที่ป้อนเข้ามา จะต้องมีความยาวคงที่ (Fixed length record) ตามที่ผู้ใช้กำหนด ในส่วนของการตรวจสอบความถูกต้องของข้อมูลของโปรแกรม DENTRY มี ลักษณะดังนี้

1. การตรวจสอบจำนวนข้อมูล (Checking the transactions)

ก) ตรวจสอบค่าผลรวมของข้อมูล (Batch totals) วิธีตรวจสอบแบบนี้ คือ คำนวณหาค่าผลรวมของเขตข้อมูลบางเขตข้อมูล ที่มีค่าเป็นตัวเลข ในแต่ละระเบียบข้อมูล แล้วนำผลรวมที่ได้ขึ้นมาเปรียบเทียบกับผลรวมที่ได้จากเครื่องเตรียมข้อมูล (ในกรณีที่เครื่องเตรียมข้อมูลนั้นมีคุณสมบัติในการตรวจสอบแบบนี้) ภายหลังการป้อนข้อมูลเรียบร้อยแล้ว ถ้าการเปรียบเทียบได้ผลที่เท่ากันแสดงว่าข้อมูลที่ป้อนมีจำนวนครบถ้วนสมบูรณ์ แต่ถ้าหากได้ผลที่ไม่เท่ากันย่อมหมายความว่ามีความผิดพลาดเกิดขึ้นระหว่างการป้อนข้อมูล ซึ่งอาจเนื่องมาจากการหาผลรวมที่ผิดของผู้ใช้เอง หรือข้อมูลที่ป้อนไม่ครบถ้วนหรือค่าของข้อมูลที่ป้อนผิด แต่ไม่อาจจะระบุได้ว่าผิดที่ระเบียบใด

ข) ตรวจสอบลำดับของข้อมูล (Sequence test) มักใช้กับข้อมูลที่มีเขตข้อมูล ที่มีลักษณะที่เรียงลำดับต่อเนื่องกัน

2. การตรวจสอบเนื้อหาของข้อมูล (Checking the transaction data)

ก) ตรวจสอบข้อมูลที่จำเป็นจะต้องปรากฏ (Existence test) การตรวจสอบวิธีนี้มักใช้กับข้อมูลบางเขตข้อมูลที่จำเป็นจะต้องมีในแต่ละระเบียบข้อมูล

ข) ตรวจสอบช่วงของข้อมูล (Range test) คือการตรวจว่าข้อมูลนั้นอยู่ในช่วงที่กำหนดไว้ หรือไม่ โดยเปรียบเทียบกับค่าสูงสุดและต่ำสุดของข้อมูล

ค) ตรวจสอบประเภทของข้อมูล (Type test) ว่าเป็นตัวอักษรหรือตัวเลข

จากการตรวจสอบความถูกต้องของข้อมูลในแต่ละโปรแกรม ที่มีลักษณะ และข้อกำหนดต่าง ๆ ตามที่กล่าวมาแล้วข้างต้น จะเห็นได้ว่า ข้อมูลยังมีโอกาสผิดพลาด หรือไม่สามารถตรวจสอบความถูกต้องของข้อมูลอีกหลาย ๆ ส่วน ซึ่งเป็นส่วนสำคัญที่จะทำให้โปรแกรมมีประสิทธิภาพในการตรวจสอบมากยิ่งขึ้น เช่น

ในโปรแกรม VSEDIT ไม่สามารถตรวจสอบความสัมพันธ์ของข้อมูลที่อยู่ต่างระบบกัน (Relational check) ไม่สามารถใช้คำสั่งที่ละมาก ๆ สำหรับการตรวจสอบข้อมูลในแต่ละครั้ง เพราะจะทำให้โปรแกรมใหญ่ขึ้น (ไม่เกิน 40 กิโลไบต์) จำกัดความยาวของระบบข้อมูลที่ตรวจสอบ และอื่น ๆ เป็นต้น

ส่วนในโปรแกรม DENTRY เป็นโปรแกรมสำหรับใช้ในการบันทึกข้อมูลซึ่งเหมาะสำหรับใช้บันทึกข้อมูลเป็นหลัก ในด้านการตรวจสอบข้อมูลของโปรแกรมนี้ ยังไม่สามารถตรวจสอบความสัมพันธ์ระหว่างเขตข้อมูล (Relational codes check) ที่อยู่ภายในระบบเดียวกัน หรือต่างระบบกันได้ ไม่สามารถตรวจสอบความเป็นไปได้ของเขตข้อมูลที่มีค่าครอบคลุมตั้งแต่ค่าที่อยู่ในช่วงที่กำหนด และค่าโดด ๆ ที่อยู่นอกเหนือจากช่วงที่กำหนดได้ (Possible codes check) มีการกำหนดความยาวของระบบข้อมูลต้องคงที่ (Fixed length record) ตรวจสอบข้อมูลได้ทีละ 1 ระบบ และอื่น ๆ เป็นต้น

เพื่อที่จะขจัดความผิดพลาดของข้อมูลที่เป็นไปได้ให้เหลือน้อยที่สุด หรือเพื่อให้ได้มาซึ่งข้อมูลที่มีความถูกต้องครบถ้วนที่สุดก่อนที่จะนำไปประมวลผล ดังนั้นงานวิจัยนี้ จึงได้ศึกษาและพัฒนาโปรแกรมบรรณาธิการข้อมูล ที่จะเอื้ออำนวยต่อการตรวจสอบความถูกต้องของข้อมูลโดยเฉพาะที่มีประสิทธิภาพยิ่งขึ้น และประกอบกับเหตุผลที่ว่า ปัจจุบันนี้ไมโครคอมพิวเตอร์ได้มีประสิทธิภาพมาก มีผู้ใช้แพร่หลายมากขึ้นเรื่อย ๆ มีหน่วยความจำหลัก (Main storage) จุได้มากขึ้น มีสื่อบันทึกข้อมูลที่เป็นจานแม่เหล็กทั้งชนิดอ่อนและชนิดแข็งให้เลือกใช้ได้ตามความเหมาะสม อีกทั้งความต้องการของผู้ใช้ ที่ต้องการให้โปรแกรมมีความคล่องตัว และมีขีดความสามารถสูงสุดเท่าที่จะเป็นไปได้ ดังนั้นงานวิจัยนี้ จึงได้พัฒนาโปรแกรมบรรณาธิการข้อมูลให้ใช้บนเครื่องคอมพิวเตอร์ขนาดเล็ก หรือเครื่องไมโครคอมพิวเตอร์ที่ดำเนินงานภายใต้ระบบปฏิบัติการ MS-DOS ซึ่งใช้กันแพร่หลายในปัจจุบันนี้

1.2 วัตถุประสงค์

1.2.1 เพื่อศึกษาและพัฒนาระบบการบรรณาธิกรข้อมูลด้วยเครื่องคอมพิวเตอร์ก่อนที่จะนำข้อมูลไปประมวลผล

1.2.2 เพื่อพัฒนาโปรแกรมสำเร็จรูปที่สามารถตรวจสอบความถูกต้องของข้อมูลทั้งที่เป็นภาษาไทยและอังกฤษได้

1.3 ขอบเขตของการวิจัย

1.3.1 การวิจัยใช้เครื่องไมโครคอมพิวเตอร์ ภายใต้ระบบปฏิบัติการ MS-DOS มีหน่วยความจำหลักไม่น้อยกว่า 640 กิโลไบต์ พร้อมตู้ขั้วจานแม่เหล็กชนิดอ่อน (floppy disk drive)

1.3.2 พัฒนาโปรแกรม โดยใช้ภาษาคอมพิวเตอร์ระดับสูง (high level language)

1.3.3 โปรแกรมที่พัฒนาขึ้นจะสามารถตรวจสอบความถูกต้องของข้อมูล โดยแบ่งออกเป็น 3 ส่วนได้แก่

- ก) ตรวจสอบความถูกต้องของการจัดลำดับหมายเลขประจำชุดข้อมูล
- ข) ตรวจสอบความสมบูรณ์ของระเบียบ
 - ตรวจสอบระเบียบที่ขาดหายไป
 - ตรวจสอบการซ้ำซ้อนของระเบียบ
 - ตรวจสอบระเบียบเกิน
- ค) ตรวจสอบความถูกต้องของข้อมูล
 - ตรวจสอบค่าเป็นไปได้ของข้อมูล (Possible codes ckeck) ที่มีค่าของข้อมูลอยู่ในช่วงที่กำหนด และรวมทั้งค่าโดด ๆ ที่อยู่นอกเหนือจากช่วงที่กำหนด
 - ตรวจสอบความสัมพันธ์กันของข้อมูล (Relational codes check) ทั้งที่เป็นข้อมูลภายในระเบียบเดียวกัน และต่างระเบียบกัน ในแต่ละหมายเลขประจำตัวอย่างประชากรชุดนั้น ๆ

1.3.4 โปรแกรมที่พัฒนาขึ้นจะสามารถตรวจสอบความถูกต้องของข้อมูล ทั้งประเภทที่เป็นจำนวนระเบียบคงที่ และประเภทจำนวนระเบียบไม่คงที่

1.4 ขั้นตอนการดำเนินการวิจัย

1.4.1 ศึกษากระบวนการตรวจสอบความถูกต้องของข้อมูล โดยใช้คอมพิวเตอร์และเลือกใช้ภาษาคอมพิวเตอร์ (computer language) ที่เหมาะสมกับระบบงาน

1.4.2 ศึกษาความต้องการของผู้ใช้ ในด้านการตรวจสอบความถูกต้องของข้อมูล ก่อนนำไปประมวลผล

1.4.3 ออกแบบระบบบรรณาธิกรข้อมูล (System Design)

1.4.4 ออกแบบ และพัฒนาโปรแกรมที่ช่วยจัดทำระบบบรรณาธิกรข้อมูล (Program Design and Implementation)

1.4.5 ทบทวน ทดสอบ และปรับปรุง ต้นแบบของระบบบรรณาธิกรข้อมูลและโปรแกรมที่ได้พัฒนาขึ้น

1.4.6 สรุปผลการวิจัย

1.5 ประโยชน์ที่จะได้รับจากการวิจัย

1.5.1 ความรู้ และเทคนิคที่ใช้ในการวิจัย สามารถใช้เป็นแนวทางในการพัฒนาโปรแกรมสำเร็จรูป หรือระบบการตรวจสอบงานอื่น ๆ ที่มีลักษณะใกล้เคียงกัน

1.5.2 โปรแกรมสำเร็จรูปที่พัฒนาขึ้น สามารถนำไปใช้ตรวจสอบความถูกต้องของข้อมูล ก่อนที่จะนำไปประมวลผล จะทำให้ได้ผลลัพธ์ที่ถูกต้องสมบูรณ์ยิ่งขึ้น

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย