บทที่ 1

บทนำ

ความเป็นมาของปัญหา

ในการออกแบบและพัฒนาระบบเริ่มต้นจากผู้ใช้ (user) เกิดปัญหาในระบบปัจจุบันนักวิเคราะห์ระบบ (Systems Analyst) รวบรวมปัญหาทั้งหมดที่เกิดขึ้นกับผู้ใช้เพื่อนำมากำหนดวัตถุประสงค์ของงาน ต่อจากนั้นก็จะทำการศึกษาความเป็นไปได้(Feasibility Studies) แล้ว นักวิเคราะห์ระบบก็จะทำการวิเคราะห์ระบบ (Systems Analysis) เพื่อหาแนวทางการแก้ปัญหา โดยนักวิเคราะห์จะทำงานร่วมกับผู้ใช้งานเพื่อพัฒนาแบบจำลองตรรกะ (Logical Model) ของระบบซึ่งใช้เครื่องมือช่วย เช่น แผนภาพกระแสข้อมูล (Data Flow Diagram) พจนานุกรมข้อมูล (Data Dictionary) ตารางตัดสินใจ (Decision Table) หรือ การอธิบายขั้นตอนการคำนวณ เป็นต้น ตารางตัดสินใจเป็นเครื่องมือที่สำคัญที่จะใช้ช่วยอธิบายขั้นตอนวิธี (Algorithm) ได้ครอบคลุมทั้งหมดซึ่งขั้นตอนการวิเคราะห์ระบบนี้นักวิเคราะห์ระบบจะต้องทราบว่าระบบที่พัฒนาขึ้นนี้จะต้องทำอะไรบ้าง ซึ่งเป็นขั้นตอนที่สำคัญจะต้องมีการตรวจสอบให้ถูกต้องและเข้าใจตรงกันระหว่างนักวิเคราะห์ระบบ ผู้ใช้และผู้บริหาร การใช้ตารางตัดสินใจช่วยอธิบายในส่วนของงานที่มีเงื่อนไขสลับซับซ้อนจะทำให้เข้าใจงานนั้น ๆ ได้ถูกต้องตรงกันหมด ทั้งยังสามารถนำมาใช้ตรวจสอบการทำงานให้ถูกต้องซ้ำได้อีก เพราะลักษณะของตารางตัดสินใจจะรวบรวมข้อมูลของเงื่อนไขทั้งหมดที่จำเป็นต่อการตัดสินใจ เงื่อนไขเชิงผสมที่เป็นไปได้ทั้งหมด (All Possible Combination of Condition) รวมทั้งการกระทำสืบเนื่อง (Action) เมื่อมีการตัดสินใจ ลักษณะของตารางตัดสินใจสามารถนำมาใช้เป็นเอกสารประกอบการวิเคราะห์ระบบ นำมาใช้ในการตรวจสอบและอ้างอิงภายหลังได้ ทั้งยังสะดวกต่อการบำรุงรักษา (Maintenance) นอกจากจะใช้กับงานที่มีเงื่อนไขสลับซับซ้อนแล้วยังเหมาะกับงานที่มีการเปลี่ยนแปลงแก้ไขบ่อย ๆ ด้วย เมื่อวิเคราะห์ระบบเสร็จสิ้นแล้วตารางจะแสดงให้เห็นแนวทางในการออกแบบปฏิกิริยาต่อกัน (Interaction) ระหว่างเงื่อนไขและการกระทำในระบบ ใช้เป็นเอกสารประกอบการออกแบบโดยละเอียด (Detail Design)

ทั้งยังใช้เป็นแนวทางในการตรวจสอบความไม่สมบูรณ์ (Incompleteness) ความซ้ำซ้อน (Redundance) และ ความขัดแย้งกัน (Inconsistency) ในการออกแบบด้วย เมื่อขั้นตอนการออกแบบเสร็จสิ้นก็ถึงขั้นตอนของการทำให้เกิดผล (Implementation) ผู้วางระบบสามารถใช้ตารางตัดสินใจแทนข้อกำหนดคุณลักษณะของโปรแกรม (Program Specification) โดยใช้ข้อความบรรยายซึ่งจะครอบคลุมได้ทุกเงื่อนไข และไม่เกิดการขัดแย้งทางตรรกะ (Logic) ทั้งยังทำให้นักเขียนชุดคำสั่งประยุกต์ (Application Programmer) เข้าใจงานได้ง่ายและถูกต้องตรงกัน ทำให้เขียนโปรแกรมได้มีประสิทธิภาพ โปรแกรมที่ได้นี้จะมีลักษณะสอดคล้องกับเทคนิคการเขียนชุดคำสั่งแบบโครงสร้าง (Structure Programming) ซึ่งจะมีผลในอนาคตต่อการบำรุงรักษาโปรแกรมและเมื่อมีการเพิ่มเติมหรือแก้ไขเงื่อนไขต่าง ๆ ก็จะทราบได้ทันทีว่ากระทบกับส่วนใดของโปรแกรม รวมทั้งยังใช้เป็นเอกสารประกอบการอธิบายโปรแกรมแทนผังงาน (Flowchart) ได้ด้วย

นอกจากประโยชน์ที่กล่าวมาแล้ว ในการวิเคราะห์ระบบสารสนเทศ (Information System) การวิเคราะห์การตัดสินใจ (Decision Analysis) เป็นส่วนที่สำคัญมาก นักวิเคราะห์ใช้วิธีการของตารางตัดสินใจในการรวบรวมสารสนเทศเพิ่มเติมเพื่อเสริมรายละเอียดของระบบความต้องการ (Reguirement) ที่ได้มาจากการวิเคราะห์แผนภาพกระแสข้อมูล (Data Flow Analysis) การวิเคราะห์การตัดสินใจจากการกำหนดเงื่อนไข และการกระทำสืบเนื่องที่เกี่ยวข้องกับการตัดสินใจ แนวความคิด (Concept) ที่สำคัญคือ เงื่อนไขซึ่งหมายถึงสถานการณ์ที่เป็นไปได้ของเหตุการณ์ ซึ่งจะนำไปถึงการเลือกการกระทำ การกระทำที่ว่านี้หมายถึงขั้นตอน การกระทำ หรือกระบวนการ (Procedure) ที่สามารถปฏิบัติเมื่อมีการตัดสินใจ การรวบรวมเงื่อนไขและการกระทำมี 3 วิธีคือ ต้นไม้ตัดสินใจ (Decision Tree) ตารางตัดสินใจ โครงสร้างภาษาอังกฤษ (Structure English) ต้นไม้ตัดสินใจเป็นแผนภาพ (Diagram) ที่แสดงทั้งเงื่อนไขและการกระทำจะเรียงตามลำดับ โดยจะแสดงความสัมพันธ์ระหว่างเงื่อนไขและการกระทำ แต่มีข้อบกพร่องในกรณีที่ระบบมีความสลับซับซ้อน มีขั้นตอนและเงื่อนไขจำนวนมากจะทำให้ ต้นไม้ตัดสินใจมีขนาดใหญ่และเทอะทะ เนื่องจากจำนวนสาขาและทางเดินมากมายทำให้นักวิเคราะห์ระบบวิเคราะห์ได้ยากและไม่ สามารถใช้เป็นแนวทางในการตัดสินใจ ซึ่งปัญหานี้จะไม่เกิดขึ้นถ้าใช้ตารางตัดสินใจ เพราะจะมีการควบคุมขนาดของตารางด้วย วิธีการ 2 วิธี วิธีแรกคือ การรวบรวมเงื่อนไขที่เกิดขึ้นหลาย ๆ เงื่อนไขที่ทำให้เกิดการกระทำอย่างเดียวกันมาทำเป็นรูป ELSE Form ซึ่งรายละเอียดจะอธิบายในบทถัดไป และอีกวิธีการคือ

สร้างตารางหลาย ๆ ตาราง (Multiple Table) ที่ครอบคลุมงานทั้งหมด แล้วจึงนำมาเชื่อมต่อกันทั้งหมด สิ่งที่กล่าวมานี้ทำให้ตารางตัดสินใจใช้ประโยชน์ได้กว้างกว่าวิธีการอื่น

จะเห็นได้ว่าตารางตัดสินใจมีประโยชน์มากในการวิเคราะห์และออกแบบระบบงานทั่วไปและระบบสารสนเทศ ทั้งยังมีประโยชน์อีกหลายด้านเมื่อได้ตารางตัดสินใจที่สมบูรณ์ การนำมาแปลงเป็นโปรแกรมคอมพิวเตอร์นั้นมีหลายวิธี ซึ่งแต่ละวิธีมีขั้นตอนที่ค่อนข้างยุ่งยากและใช้เวลามากพอสมควร นับตั้งแต่ศึกษาภาษาคอมพิวเตอร์ที่เหมาะสมกับงานที่จะใช้การศึกษาเทคนิคในการแปลงตารางให้อยู่ในรูปที่ใช้กับคอมพิวเตอร์ได้ การเขียนคำสั่ง (Coding) การแปลชุดคำสั่งเป็นภาษาเครื่องที่สามารถกระทำการได้ รวมทั้งการทดสอบ (Testing) ซึ่งในแต่ละขั้นตอนมีโอกาสเกิดข้อผิดพลาดได้มากและการแปลงตารางตัดสินใจที่มีเงื่อนไขสลับซับซ้อนด้วยมืออาจผิดพลาดได้ง่าย ดังนั้นถ้าสามารถพัฒนาเครื่องมือช่วย ในการแปลงตารางตัดสินใจเป็นรูปส่วนของโปรแกรมคอมพิวเตอร์ เมื่อนำส่วนที่สร้างขึ้นนี้ ไปเพิ่มเติมส่วนอื่น ๆ นอกเหนือจากส่วนที่เกี่ยวข้องกับเงื่อนไขและการกระทำ ทั้งหมดก็จะได้โปรแกรมที่สมบูรณ์ตามต้องการจะเป็นการช่วยลดปัญหาต่าง ๆ ดังกล่าวข้างต้นได้

ด้วยสาเหตุที่กล่าวมาทั้งหมดนี้ วิทยานิพนธ์นี้จึงมุ่งที่จะศึกษาตารางตัดสินใจตั้งแต่วิธีการสร้าง การทำให้ตารางสมบูรณ์แบบ การประยุกต์ใช้กับงานด้านต่าง ๆ รวมทั้งการศึกษา และพัฒนาเครื่องมือช่วยในการแปลงตารางตัดสินใจไปเป็นส่วนของโปรแกรมคอมพิวเตอร์ซึ่งจะทำให้ผู้ใช้สะดวกที่จะนำไปใช้ เป็นการช่วยประหยัดเวลา การบำรุงรักษาทำได้โดยง่าย ทั้งยังสามารถนำไปดัดแปลงใช้กับภาษาอื่นได้โดยง่ายเพราะมีรูปแบบที่ค่อนข้างแน่นอน

วัตถุประสงค์และขอบเขตของการวิจัย

1. วัตถุประสงค์ของการวิจัย

1.1 เพื่อแสดงการประยุกต์ใช้ตารางตัดสินใจในการวิเคราะห์ปัญหาและวิธีการนำไปใช้ในการช่วยพัฒนาโปรแกรม

1.2 เพื่อออกแบบและพัฒนาโปรแกรมช่วยในการแปลงตารางตัดสินใจให้เป็นส่วนของโปรแกรมคอมพิวเตอร์โดยการเก็บรายละเอียดของเงื่อนไข และการกระทำที่เกี่ยวข้องกับเงื่อนไขทั้งหมดนำมาสร้างเป็นส่วนของโปรแกรม ทำให้สะดวก รวดเร็ว มีความถูกต้องมากขึ้น รวมทั้งการบำรุงรักษาทำได้โดยง่าย

2. ขอบเขตของการวิจัย

2.1 แสดงวิธีการวิเคราะห์ปัญหาโดยใช้ตารางตัดสินใจวิธีการสร้างและการใช้ตารางตัดสินใจประกอบการเขียนโปรแกรมโดยจะแสดงตารางตัดสินใจ 3 ชนิด คือ

2.1.1 Limited-entry Decision Table (LEDT)

2.1.2 Extended-entry Decision Table (EEDT)

2.1.3 Mixed-entry Decision Table (MEDT)

2.2 แสดงวิธีการและเทคนิคต่าง ๆ ที่ใช้ในการแปลงตารางตัดสินใจให้เป็นโปรแกรมคอมพิวเตอร์ รวมทั้งแสดงตัวอย่างประกอบ วิธีการที่แสดงในวิทยานิพนธ์มีดังนี้

2.2.1 วิธีการแปลงเป็นแผนภาพต้นไม้ (Decision Tree Method)

2.2.2 วิธีรูมาสค์ (Rule Mask Method)

2.3 พัฒนาโปรแกรมบนเครื่องไมโครคอมพิวเตอร์ไอบีเอ็มพีซี (IBM/PC หรือ IBM/PC Compatible) ขนาด 16 บิต โปรแกรมที่พัฒนาขึ้นนี้มีความสามารถที่จะแปลงตารางตัดสินใจแบบการรับเข้าจำกัด (LEDT) เป็นส่วนของโปรแกรม NESTED-IF-THEN-ELSE โดยใช้รูปแบบของคำสั่งในภาษาโคบอล (COBOL) ซึ่งในส่วนนี้จะจัดการกับเงื่อนไขที่สลับซับซ้อน

2.3.1 ลักษณะของตารางตัดสินใจที่พัฒนาจะต้องไม่มีลักษณะกำกวม (Non-ambiguous Table) ไม่มีกฎที่ซ้ำซ้อน (Redundancy) หรือขัดแย้งกัน (Contradiction) ลำดับของเงื่อนไขต่าง ๆ ในตารางและลำดับของกฎจะจัดเรียงเช่นไรก็ได้

2.3.2 สิ่งนำเข้าสำหรับโปรแกรมช่วยแปลงตารางตัดสินใจเป็นส่วนของโปรแกรมคอมพิวเตอร์ประกอบด้วย

2.3.2.1 Condition Stub จะอยู่ในรูปของ COBOL statement

2.3.2.2 Action Stub จะอยู่ในรูปของ COBOL Statement

2.3.2.3 Condition Entry จะต้องเป็น Y, N หรือ - เท่านั้น

2.3.2.4 Action Entry จะต้องเป็น X หรือช่องว่าง

ขั้นตอนและวิธีดำเนินการวิจัย

1. ศึกษาการสร้าง การวิเคราะห์ และการใช้ตารางตัดสินใจต่าง ๆ รวมทั้งเทคนิคต่าง ๆ ในการแปลงตารางตัดสินใจให้เป็นโปรแกรมคอมพิวเตอร์ เพื่อเป็นแนวทางในการพัฒนาโปรแกรมช่วยแปลงตารางตัดสินใจเป็นส่วนของโปรแกรมคอมพิวเตอร์

2. ศึกษาภาษาคอมพิวเตอร์ส่วนที่ถูกสร้างขึ้นเพื่อให้ได้โปรแกรมที่มีลักษณะเป็นโครงสร้างและมีความหมายมากที่สุด

3. วิเคราะห์และออกแบบตารางตัดสินใจ

4. พัฒนาโปรแกรมช่วยแปลงตารางตัดสินใจ

5. ทดสอบโปรแกรมที่สร้างขึ้น

6. สรุปผลการทำวิจัยและจัดทำคู่มือการใช้

7. จัดทำวิทยานิพนธ์

ประโยชน์ที่คาดว่าจะได้รับ

1. โปรแกรมที่ได้สามารถนำไปใช้เป็นเครื่องมือช่วยในการพัฒนาโปรแกรมในส่วนที่มีเงื่อนไขสลับซับซ้อน หรือโปรแกรมที่มีการเปลี่ยนแปลงแก้ไขบ่อยครั้ง ส่วนของโปรแกรมจะมีลักษณะเป็นโปรแกรมแบบโครงสร้าง ทำให้สะดวกต่อการบำรุงรักษา

2. ส่วนของโปรแกรมที่ได้สามารถใช้เป็นเอกสารประกอบโปรแกรม

3. สามารถนำไปดัดแปลงใช้กับภาษาคอมพิวเตอร์อื่น ๆ ได้ในภายหลัง

4. เป็นแนวทางในการพัฒนาเครื่องมือช่วยในการเขียนโปรแกรมอื่น ๆ ต่อไป