บทที่ 7
ผลการทดลองใช้วิธีการระบุวัตถุซอฟต์แวร์ที่เป็นไปได้แบบอื่น ๆ

เนื้อหาในบทนี้เกี่ยวข้องกับ การนำเสนอผลการทดลองใช้วิธีการระบุวัตถุซอฟต์แวร์ที่เป็นไปได้แบบอื่นๆ คือ วิธีการระบุวัตถุซอฟต์แวร์ที่ประยุกต์จากวิธีการจัดกลุ่มข้อมูลแบบลำดับชั้น (แบบปกติ) และวิธีการระบุวัตถุซอฟต์แวร์ที่เป็นไปได้ซึ่งประยุกต์วิธีการวิเคราะห์คอนเซ็ปต์ ในโปรแกรมต้นฉบับเชิงโครงสร้างภาษาซีจำนวนสามโปรแกรม โดยได้แบ่งการอธิบายแบ่งออกเป็น 2 หัวข้อ คือ (1) ผลการทดลองใช้วิธีการระบุวัตถุซอฟต์แวร์ที่เป็นไปได้ซึ่งประยุกต์จากวิธีการจัดกลุ่มข้อมูลแบบลำดับชั้น (แบบปกติ) และ (2) ผลการทดลองใช้วิธีการระบุวัตถุซอฟต์แวร์ที่เป็นไปได้ซึ่งประยุกต์ใช้วิธีการวิเคราะห์คอนเซ็ปต์ โดยมีรายละเอียดดังนี้

7.1 ผลการทดลองใช้วิธีการระบุวัตถุซอฟต์แวร์ที่เป็นไปได้ซึ่งประยุกต์จากวิธีการจัดกลุ่มข้อมูลแบบลำดับชั้น (แบบปกติ)

1) ผลการคัดแยกส่วนประกอบเดิม ผลการกำหนดนิยามความสัมพันธ์ ผลการตรวจสอบความสัมพันธ์ และผลการสร้างเมตริกซ์ข้อมูล

เนื่องจากขั้นตอนการทำงานที่ใช้ในการคัดแยกส่วนประกอบเดิม การกำหนดนิยามความสัมพันธ์ การตรวจสอบความสัมพันธ์ และการสร้างเมตริกซ์ข้อมูลของวิธีการนี้ เป็นเช่นเดียวกับขั้นตอนการทำงานของวิธีการในบทที่ 6 ผลที่ได้จากการทำงานจึงเป็นอย่างเดียวกัน กล่าวคือ รายการส่วนประกอบเดิมเหมือนกับตารางที่ 6.1 - 6.3 นิยามความสัมพันธ์เฉพาะเจาะจงเหมือนกับตารางที่ 6.4 – 6.6 ตารางความสัมพันธ์เหมือนกับตารางที่ 6.7 – 6.10 และเมตริกซ์ข้อมูลเหมือนกับตารางที่ 6.11 –6.13

2) ผลการจัดเรียงส่วนประกอบเดิม

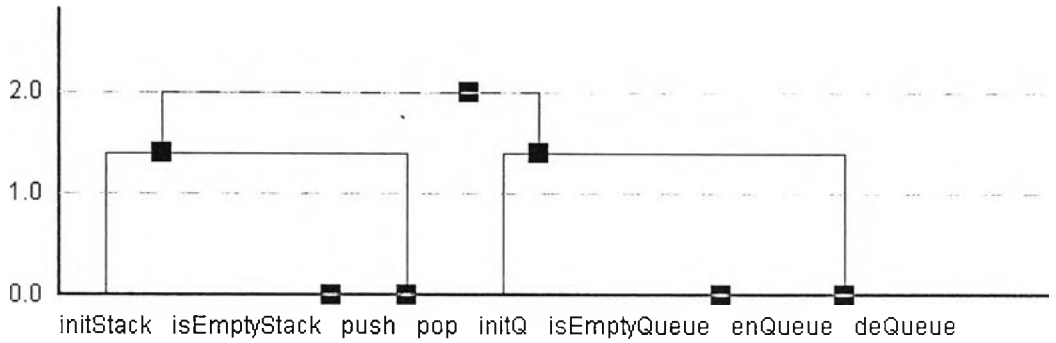
ผลที่ได้จากการทำงานในขั้นตอนนี้คือ เมตริกซ์ความแตกต่างที่ถูกสร้างขึ้นระหว่างการรวมคลัสเตอร์ แต่เนื่องจากเมตริกซ์ดังกล่าวมีเป็นจำนวนมาก จึงได้แยกเมตริกซ์เหล่านั้นออกไปแสดงในภาคผนวก ข

3) ผลการสร้างเดนโดแกรม

ผลที่ได้จากการทำงานในขั้นตอนนี้คือ เดนโดแกรมที่แสดงลำดับการรวมกันของคลัสเตอร์ต่างๆ ซึ่งสามารถแจกแจงตามชนิดของโปรแกรมที่ใช้ในการทดลองได้ดังนี้

a. โปรแกรมแถวลำดับแถวคอยแบบแยกส่วน

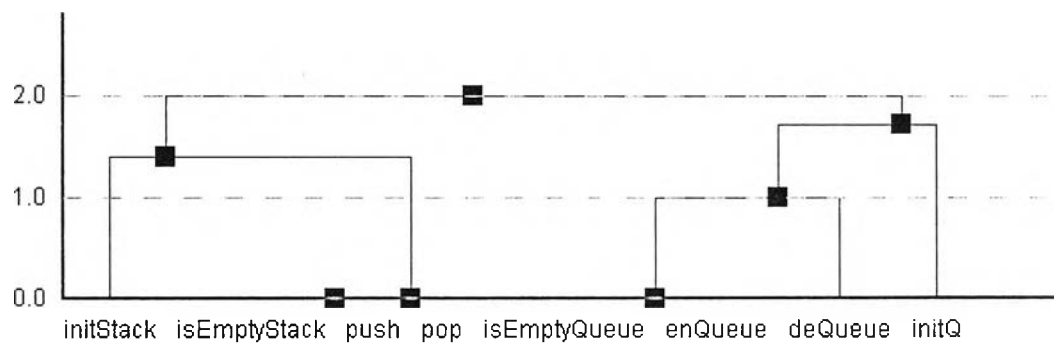
จากเดนโดแกรมในรูปที่ 7.1 จะเห็นได้ว่า วิธีการจัดกลุ่มข้อมูลฯ (แบบปกติ) สามารถจัดเรียงส่วนคำสั่งเป็นคลัสเตอร์ที่มีลักษณะเป็นกลุ่มสมบูรณ์ได้ครบทั้ง 2 คลัสเตอร์ คือ คลัสเตอร์ที่ประกอบด้วย initStack, isEmptyStack, push, pop และคลัสเตอร์ที่ประกอบด้วย initQ isEmptyQueue, enQueue, deQueue



รูปที่ 7.1 เดนโดแกรมของโปรแกรมแถวลำดับแถวคอยแบบแยกส่วน
จากการใช้วิธีการจัดกลุ่มข้อมูลฯ (แบบปกติ)

b. โปรแกรมแถวลำดับแถวคอยแบบผูกติด

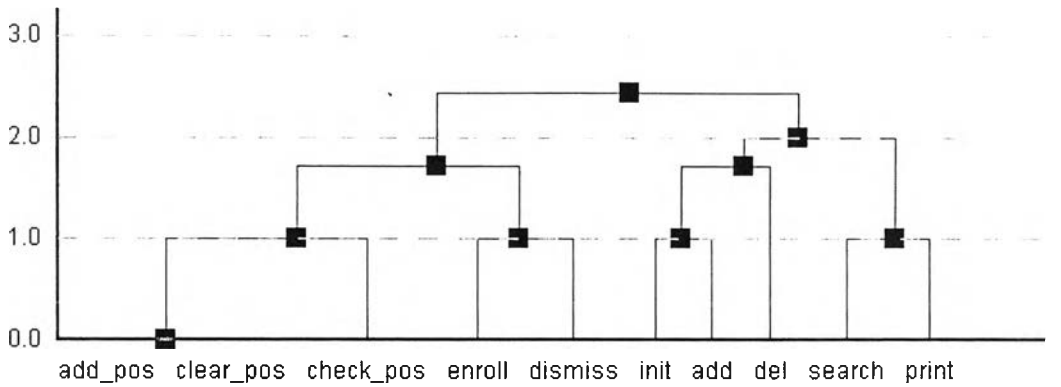
จากเดนโดแกรมในรูปที่ 7.2 จะเห็นได้ว่า วิธีการจัดกลุ่มข้อมูลฯ (แบบปกติ) สามารถจัดเรียงส่วนคำสั่งเป็นคลัสเตอร์ที่มีลักษณะเป็นกลุ่มสมบูรณ์ได้ครบทั้ง 2 คลัสเตอร์ คือ คลัสเตอร์ที่ประกอบด้วย initStack, isEmptyStack, push, pop และคลัสเตอร์ที่ประกอบด้วย initQ isEmptyQueue, enQueue, deQueue



รูปที่ 7.2 เดนโดแกรมของโปรแกรมแถวลำดับแถวคอยแบบผูกติด
จากการใช้วิธีการจัดกลุ่มข้อมูลฯ (แบบปกติ)

c. โปรแกรมแถวลงทะเบียนพนักงาน

จากเดนโดแกรมในรูปที่ 7.3 จะเห็นได้ว่า วิธีการจัดกลุ่มข้อมูลฯ (แบบปกติ) สามารถจัดเรียงส่วนคำสั่งเป็นคลัสเตอร์ที่มีลักษณะเป็นกลุ่มสมบูรณ์ได้ครบทั้ง 2 คลัสเตอร์ คือ คลัสเตอร์ที่ประกอบด้วย add_pos, clear_pos, check_pos, enroll, dismiss และคลัสเตอร์ที่ประกอบด้วย init, add, search, print, del ทั้งนี้มีข้อสังเกตว่า ลำดับการสร้างกลุ่มสมบูรณ์ init, add, search, print, del ในกรณีนี้เหมือนกับลำดับการสร้างกลุ่มสมบูรณ์ของวิธีการจัดกลุ่มข้อมูลฯ ในกรณีเพิ่มค่าน้ำหนัก



รูปที่ 7.3 เดนโดแกรมของโปรแกรมลงทะเบียนพนักงาน
จากการใช้วิธีการจัดกลุ่มข้อมูลฯ (แบบปกติ)

7.2 ผลการทดลองใช้วิธีการระบุวัตถุซอฟต์แวร์ที่เป็นไปได้ซึ่งประยุกต์ใช้วิธีการวิเคราะห์คอนเซ็ปต์

การทดลองใช้วิธีการระบุวัตถุซอฟต์แวร์ที่เป็นไปได้ซึ่งประยุกต์ใช้วิธีการวิเคราะห์คอนเซ็ปต์ กระทำตามขั้นตอนที่ระบุไว้ในงานของ Siff และ Reps [7] ซึ่งได้ผลการทำงานออกมาดังนี้

1) ผลการคัดแยกส่วนประกอบเดิม ผลการกำหนดนิยามความสัมพันธ์ และผลการตรวจสอบความสัมพันธ์

เนื่องจากขั้นตอนการทำงานที่ใช้ในการคัดแยกส่วนประกอบเดิม การกำหนดนิยามความสัมพันธ์ การตรวจสอบความสัมพันธ์ของวิธีการนี้ เป็นเช่นเดียวกับขั้นตอนการทำงานของวิธีการในบทที่ 6 ผลที่ได้รับจากการทำงานจึงเป็นอย่างเดียวกัน กล่าวคือ รายการส่วนประกอบเดิมเหมือนกับตารางที่ 6.1 - 6.3 นิยามความสัมพันธ์เฉพาะเจาะจงเหมือนกับตารางที่ 6.4 – 6.6 และตารางความสัมพันธ์เหมือนกับตารางที่ 6.7 – 6.10

2) ผลการสร้างคอนเซ็ปต์

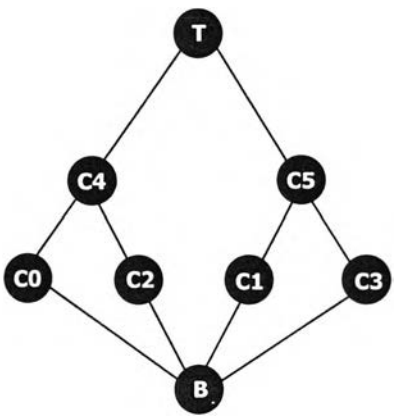
ผลที่ได้จากการทำงานในขั้นตอนนี้คือ ตารางคอนเซ็ปต์และโครงข่ายคอนเซ็ปต์ ซึ่งสามารถแจกแจงตามชนิดของโปรแกรมที่ใช้ในการทดลองได้ดังนี้

a. โปรแกรมแถวลำดับแถวคอยแบบแยกส่วน

จากตารางคอนเซ็ปต์ในตารางที่ 7.1 จะเห็นได้ว่า วิธีการวิเคราะห์คอนเซ็ปต์สามารถจัดเรียงส่วนคำสั่งเป็นคอนเซ็ปต์ที่มีลักษณะเป็นกลุ่มสมบูรณ์ได้ครบทั้ง 2 คอนเซ็ปต์ คือ คอนเซ็ปต์ C4 ({INS, IES, PSH, POP}, {USTK}) และคอนเซ็ปต์ C5 ({INQ, IEQ, ENQ, DEQ}, {UQUE}) ความสัมพันธ์ระหว่างคอนเซ็ปต์ดังกล่าวกับคอนเซ็ปต์อื่นๆ ในโดเมนถูกแสดงไว้ในโครงข่ายคอนเซ็ปต์ ดังรูปที่ 7.4

ตารางที่ 7.1 ตารางคอนเซ็ปต์ของโปรแกรมแถวลำดับแถวคอยแบบแยกส่วน

TOP	(ALL, @)
C5	({INQ, IEQ, ENQ, DEQ}, {UQUE})
C4	({INS, IES, PSH, POP}, {USTK})
C3	({IEQ, ENQ, DEQ}, {AQUE, UQUE})
C2	({IES, PSH, POP}, {ASTK, USTK})
C1	({INQ}, {RQUE, UQUE})
C0	({INS}, {RSTK, USTK})
BOT	(@, ALL)



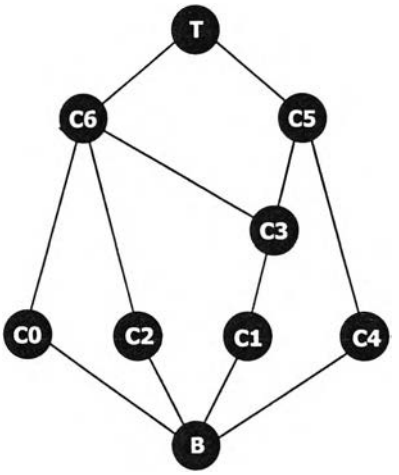
รูปที่ 7.4 โครงข่ายคอนเซ็ปต์ของโปรแกรมแถวลำดับแถวคอยแบบแยกส่วน

b. โปรแกรมแถวลำดับแถวคอยแบบผูกติด

จากตารางคอนเซ็ปต์ในตารางที่ 7.2 จะเห็นได้ว่า วิธีการวิเคราะห์คอนเซ็ปต์สามารถจัดเรียงส่วนคำสั่งเป็นคอนเซ็ปต์ที่มีลักษณะเป็นกลุ่มสมบูรณ์ได้เพียงคอนเซ็ปต์เดียว คือ คอนเซ็ปต์ C5 ({INQ, IEQ, ENQ, DEQ}, {UQUE}) ความสัมพันธ์ระหว่างคอนเซ็ปต์ดังกล่าวกับคอนเซ็ปต์อื่นๆ ในโดเมนถูกแสดงไว้ในโครงข่ายคอนเซ็ปต์ ดังรูปที่ 7.5

ตารางที่ 7.2 ตารางคอนเซ็ปต์ของโปรแกรมแถวลำดับแถวคอยแบบผูกติด

TOP	(ALL, @)
C6	({INS, IES, IEQ, PSH, ENQ, POP}, {USTK})
C5	({INQ, IEQ, ENQ, DEQ}, {UQUE})
C4	({IEQ, ENQ, DEQ}, {AQUE, UQUE})
C3	({IEQ, ENQ}, {AQUE, UQUE, USTK})
C2	({IES, PSH, POP}, {ASTK, USTK})
C1	({INQ}, {RQUE, UQUE})
C0	({INS}, {RSTK, USTK})
BOT	(@, ALL)



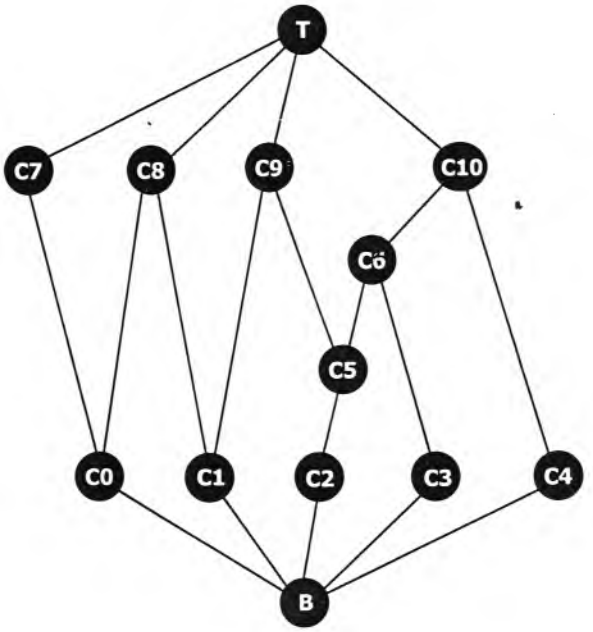
รูปที่ 7.5 โครงข่ายคอนเซ็ปต์ของโปรแกรมแถวลำดับแถวคอยแบบผูกติด

c. โปรแกรมแถวลงทะเบียนพนักงาน

จากตารางคอนเซ็ปต์ในตารางที่ 7.3 จะเห็นได้ว่า วิธีการวิเคราะห์คอนเซ็ปต์สามารถจัดเรียงส่วนคำสั่งเป็นคอนเซ็ปต์ที่มีลักษณะเป็นกลุ่มสมบูรณ์ได้เพียงคอนเซ็ปต์เดียว คือ คอนเซ็ปต์ C10 ({INI, ADD, DEL, SRC, PRN}, {ULIST}) ความสัมพันธ์ระหว่างคอนเซ็ปต์ดังกล่าวกับคอนเซ็ปต์อื่นๆ ในโดเมนถูกแสดงไว้ในโครงข่ายคอนเซ็ปต์ ดังรูปที่ 7.6

ตารางที่ 7.3 คอนเซ็ปต์จากโปรแกรมลงทะเบียนพนักงาน

TOP	(ALL, @)
C10	({INI, ADD, DEL, SRC, PRN}, {ULIST})
C9	({ENR, SRC, PRN}, {UEMP})
C8	({CHK, ENR, DSM}, {UCUR})
C7	({ADP, CLP, CHK}, {UMAX})
C6	({INI, ADD, SRC, PRN}, {ULIST, UHDP})
C5	({SRC, PRN}, {ULIST, UHDP, UEMP})
C4	({DEL}, {ALIST, ULIST})
C3	({ADD}, {ULIST, UHDP, AEMP})
C2	({SRC}, {RLIST, ULIST, UHDP, UEMP})
C1	({ENR}, {UCUR, UEMP})
C0	({CHK,}, {UMAX, UCUR})
BOT	(@, ALL)



รูปที่ 7.6 โครงข่ายคอนเซ็ปต์ของโปรแกรมลงทะเบียนพนักงาน