

ระบบจัดการการให้บริการเกี่ยวกับบริษัทขายเครื่องปรับอากาศ  
ของภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

A program for management of air condition services

นายณัฐกร จองไพจิตรสกุล รหัส 46380015  
นายณัฐพล ชุ่ยคำ รหัส 46380016  
นายณัฐวุฒิ ศรีสุวรรณ รหัส 46380019

ห้องสมุดคณะวิศวกรรมศาสตร์  
วันที่รับ..... 25 / พ.ศ. 2553 / .....

เลขทะเบียน..... ๒๐๑๖๗๒๑ .....

เลขเรียกหนังสือ..... ๗๕ .....

มหาวิทยาลัยนเรศวร ๘๖ 8225

๘๖๗๙

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2549



## ใบรับรองโครงการวิจัย

หัวข้อโครงการ	ระบบจัดการการให้บริการเกี่ยวกับบริษัทขายเครื่องปรับอากาศ		
ผู้ดำเนินโครงการ	นายณัฐกร	จงไพจิตรสกุล	รหัส 46380015
	นายณัฐพล	ขุ่ยคำ	รหัส 46380016
	นายณัฐวุฒิ	ศรีสุวรรณ	รหัส 46380019
อาจารย์ที่ปรึกษา	อาจารย์แสงชัย มังกรทอง		
สาขาวิชา	วิศวกรรมคอมพิวเตอร์		
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์		
ปีการศึกษา	2549		

.....  
 คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเกษตรศาสตร์ อนุมัติให้โครงการฉบับนี้เป็นส่วนหนึ่งของ  
 การศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์  
 คณะกรรมการสอบโครงการวิจัย

.....  
 .....ประธานกรรมการ  
 (อาจารย์แสงชัย มังกรทอง)

.....  
 .....กรรมการ  
 (ดร.พนมขวัญ ธิยะมงคล)

.....  
 .....กรรมการ  
 (อาจารย์ภาณุพงศ์ สอนคม)

หัวข้อโครงการ	ระบบจัดการการให้บริการเกี่ยวกับบริษัทขายเครื่องปรับอากาศ	
ผู้ดำเนินโครงการ	นายณัฐกร จงไพจิตรสกุล	รหัส 46380015
	นายณัฐพล ชุ่ยคำ	รหัส 46380016
	นายณัฐวุฒิ ศรีสุวรรณ	รหัส 46380019
อาจารย์ที่ปรึกษา	อาจารย์แสงชัย มังกรทอง	
สาขาวิชา	วิศวกรรมคอมพิวเตอร์	
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์	
ปีการศึกษา	2549	

**บทคัดย่อ**

โครงการนี้เป็นการออกแบบระบบจัดการการให้บริการเกี่ยวกับบริษัทขายเครื่องปรับอากาศ โดยจะเป็นระบบที่ใช้คำนวณหาระยะทางในการไปให้บริการทุกประเภทของบริษัทขายเครื่องปรับอากาศเช่น การล้าง,ทำความสะอาดเครื่องปรับอากาศ การติดตั้งเครื่องปรับอากาศ การซ่อมเครื่องปรับอากาศ เป็นต้น โดยเราจะคำนวณว่าเราจะไปให้บริการแก่ลูกค้าที่ใดบ้าง และจะต้องไปในเส้นทางใดกลับเส้นทางใดจึงจะทำให้เราไปในระยะทางที่สั้นที่สุดและประหยัดค่าใช้จ่ายในการเดินทางไปให้บริการในแต่ละครั้ง เพื่อให้ได้ผลตามที่ต้องการในการออกไปให้บริการในแต่ละครั้ง

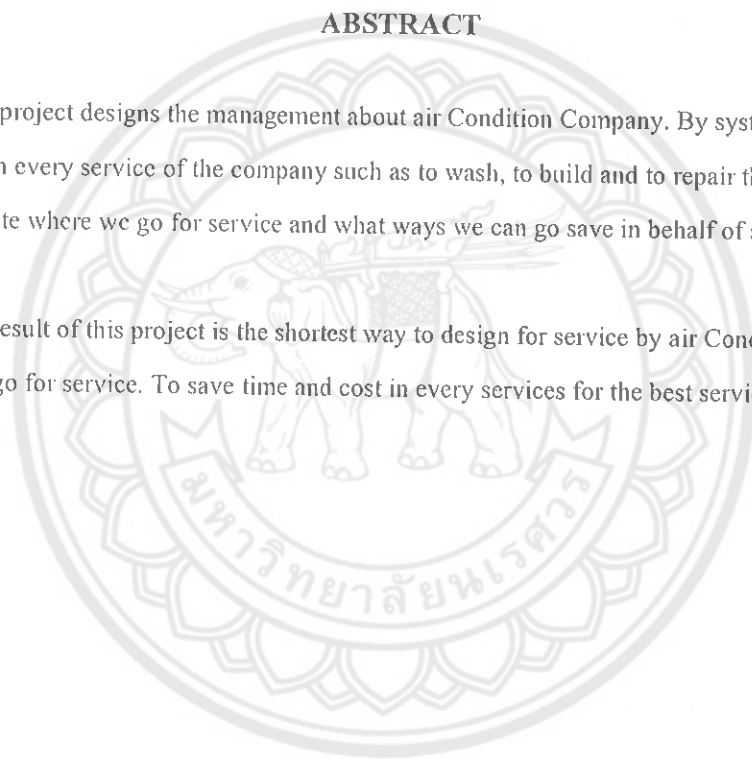
ผลที่ได้จากการทำโครงการนี้คือ ได้ระยะทางที่สั้นที่สุดในการออกไปให้บริการของบริษัทขายเครื่องปรับอากาศในแต่ละครั้งที่ออกไปให้บริการ และจะทำให้ประหยัดค่าใช้จ่ายและเวลาในการออกไปให้บริการในแต่ละครั้งเพื่อให้ได้ผลที่ดีที่สุดในการออกไปให้บริการในแต่ละครั้งของบริษัทขายเครื่องปรับอากาศ

<b>Project Title</b>	A program for management of air condition services		
<b>Name</b>	Mr.Nattakorn	Jongpajitsakuln	ID.46380015
	Mr.Nutthapon	Khuikam	ID.46380016
	Mr.Nutthawut	Srisuwan	ID.46380019
<b>Project Advisor</b>	Mr.Sangchai	Mungkomtong	
<b>Major</b>	Computer Engineering		
<b>Department</b>	Electrical and Computer Engineering		
<b>Academic year</b>	2006		

### ABSTRACT

This project designs the management about air Condition Company. By system to compute the distance in every service of the company such as to wash, to build and to repair the air condition. So we calculate where we go for service and what ways we can go save in behalf of service by our company.

The result of this project is the shortest way to design for service by air Condition Company all time they go for service. To save time and cost in every services for the best service of the company



## กิตติกรรมประกาศ

ในการทำปริญญานิพนธ์ครั้งนี้ได้สำเร็จลุล่วงเป็นไปได้ด้วยดีโดยได้รับคำแนะนำและคำปรึกษาตลอดจนความช่วยเหลือทางด้านความรู้ข้อเสนอนะในการแก้ไขปัญหาต่างๆจากอาจารย์แสงชัย มังกรทอง รวมทั้งเพื่อนๆทุกคน และแขวงการทางจังหวัดพิษณุโลกที่ให้ความสะดวกในด้านของการให้แผนที่เส้นทางในตัวจังหวัดพิษณุโลกและแผนที่ต่าง ๆ มาให้ศึกษา และภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ที่ให้ความสะดวกในด้านอุปกรณ์ต่างๆจนจัดทำปริญญานิพนธ์สำเร็จ ซึ่งผู้จัดทำขอขอบคุณไว้ ณ ที่นี้ด้วย

นายณัฐกร จองไพจิตรสกุล  
นายณัฐพล ชุ่มคำ  
นายณัฐวุฒิ ศรีสุวรรณ



# สารบัญ

	หน้า
ใบรับรองโครงการวิจัย.....	ก
บทคัดย่อภาษาไทย.....	ข
บทคัดย่ออังกฤษ.....	ค
กิตติกรรมประกาศ.....	ง
สารบัญ.....	จ
สารบัญตาราง.....	ช
สารบัญรูป.....	ฉ
<b>บทที่ 1 บทนำ</b>	
1.1 เป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์.....	1
1.3 ขอบเขตของโครงการ.....	1
1.4 กิจกรรมการดำเนินโครงการ.....	2
1.5 ผลที่คาดว่าจะได้รับ.....	3
1.6 งบประมาณ.....	3
<b>บทที่ 2 หลักการและทฤษฎีที่เกี่ยวข้อง</b>	
2.1 ทฤษฎีกราฟ.....	4
2.1.1 พื้นฐานทั่วไปของกราฟ.....	4
2.1.2 กราฟที่ระบุทิศทาง.....	4
2.1.3 THE TRAVELING SALEMAN PROBLEM.....	5
2.1.4 การแสดงกราฟด้วยเมทริกซ์.....	6
2.2 ทฤษฎีของ Dijkstra's Algorithm.....	8
2.3 ทฤษฎีของการเรียงสับเปลี่ยน-Permutation.....	15
2.3.1 จำนวนวิธีของการเรียงสับเปลี่ยนสิ่งของครวละสิ่งจาก สิ่งของnสิ่งที่แตกต่างกัน.....	15



## สารบัญ (ต่อ)

	หน้า
3.3 แนวการคิดของโปรแกรม.....	36
3.4 ขั้นตอนการทำงานของโปรแกรม.....	40
3.5 อธิบายลำดับขั้นตอนการทำงานของโปรแกรม.....	41
3.5.1 เข้าไปแก้ไขข้อมูล.....	41
3.5.2 การวางแผนการเดินทางการ.....	41
3.5.3 ออกจากโปรแกรม.....	41
บทที่ 4 ขั้นตอนการดำเนินงาน	
4. วิธีการใช้งานโปรแกรม.....	42
4.1 เลือกแก้ไขข้อมูล.....	43
4.1.1 กรณีเลือกเส้นทาง.....	43
4.1.2 กรณีเลือกงาน.....	49
4.2 เลือกวางแผนการเดินทาง.....	53
4.3 Action GUI.....	55
บทที่ 5 สรุปผล	
5.1 สรุปผลโครงการ.....	59
5.2 ปัญหาและอุปสรรค.....	59
5.3 แนวทางแก้ไขปัญหา.....	59
5.4 ข้อเสนอแนะ.....	60
เอกสารอ้างอิง.....	61
ภาคผนวก.....	62
ประวัติผู้เขียนโครงการ.....	74



## สารบัญตาราง

ตารางที่	หน้า
2.1 ตารางที่ 2.1 THE TRAVELING SALEMAN PROBLEM	5
2.2 ตารางที่ 2.2 Dijkstra's Algorithm	9
2.3 ตารางที่ 2.3 Dijkstra's Algorithm	10
2.4 ตารางที่ 2.4 Dijkstra's Algorithm	11
2.5 ตารางที่ 2.5 Dijkstra's Algorithm	12
2.6 ตารางที่ 2.6 Dijkstra's Algorithm	13
2.7 ตารางที่ 2.7 Dijkstra's Algorithm	14
3.1 ตารางที่ 3.1 ตารางแสดง Pixel	31
3.2 ตารางที่ 3.2 ตารางแสดง scale	32

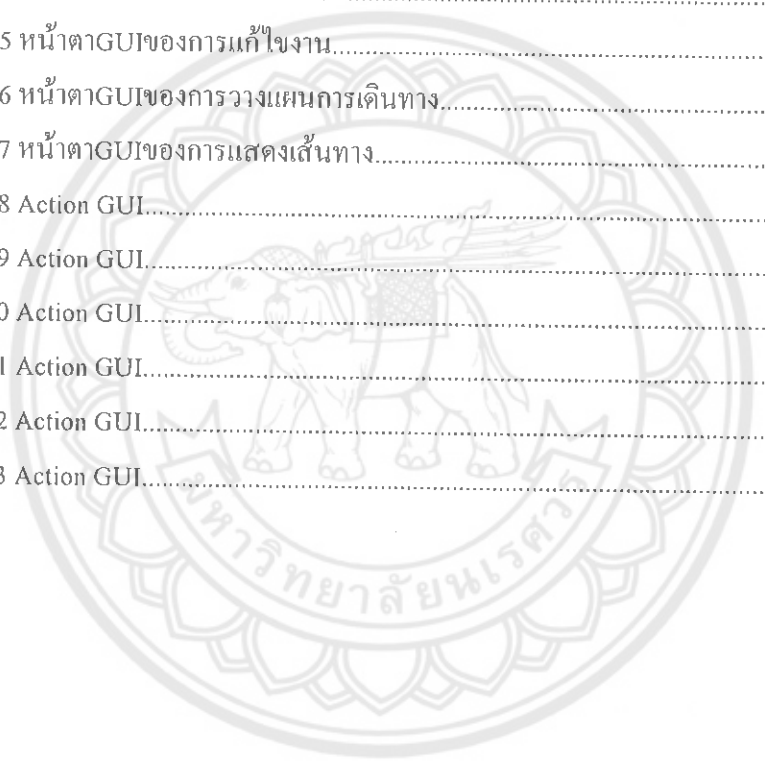


## สารบัญรูป

รูปที่	หน้า
2.1 รูปที่ 2.1 THE TRAVELING SALEMAN PROBLEM.....	5
2.2 รูปที่ 2.2 การแสดงกราฟด้วยเมทริกซ์.....	6
2.3 รูปที่ 2.3 การแสดงกราฟด้วยเมทริกซ์.....	7
2.4 รูปที่ 2.4 Dijkstra's Algorithm.....	9
2.5 รูปที่ 2.5 Dijkstra's Algorithm.....	10
2.6 รูปที่ 2.6 Dijkstra's Algorithm.....	11
2.7 รูปที่ 2.7 Dijkstra's Algorithm .....	12
2.8 รูปที่ 2.8 Dijkstra's Algorithm.....	13
2.9 รูปที่ 2.9 Dijkstra's Algorithm.....	14
2.10 รูปที่ 2.10 Dijkstra's Algorithm.....	16
3.1 รูปที่ 3.1 แผนที่จังหวัดพิษณุโลกก่อนการกำหนดจุดโหนด.....	25
3.2 รูปที่ 3.2 แผนที่ในตัวเมืองพิษณุโลกก่อนการกำหนดจุดโหนด.....	26
3.3 รูปที่ 3.3 การกำหนดจุดโหนดของจังหวัดพิษณุโลก.....	27
3.4 รูปที่ 3.4 การกำหนดจุดโหนดในตัวเมืองพิษณุโลก.....	28
3.5 รูปที่ 3.5 แผนภาพ ER- Diagram.....	29
3.6 รูปที่ 3.6 แผนภาพ Schema.....	30
3.7 รูปที่ 3.7 Class Diagram .....	33
3.8 รูปที่ 3.8 แผนที่ประกอบในการแสดงการหาเส้นทางบริเวณรอบนอก.....	34
3.9 รูปที่ 3.9 แผนที่ประกอบในการแสดงการหาเส้นทางในตัวเมืองจังหวัดพิษณุโลก.....	35
3.10 รูปที่ 3.10 แนวความคิดของโปรแกรม.....	37
3.11 รูปที่ 3.11 แนวความคิดของโปรแกรม.....	39
3.12 รูปที่ 3.12 ขั้นตอนการทำงานของโปรแกรม.....	40
4.1 รูปที่ 4.1 หน้าตาGUIก่อนเข้าสู่โปรแกรม.....	42
4.2 รูปที่ 4.2 หน้าตาGUIของโปรแกรมการจัดการและวางแผนการเดินทาง.....	42
4.3 รูปที่ 4.3 หน้าตาGUIของการเลือกสิ่งที่ต้องการแก้ไข.....	43
4.4 รูปที่ 4.4 หน้าตาGUIของแก้ไขเส้นทาง.....	44
4.5 รูปที่ 4.5 หน้าตาGUIของการปรับเปลี่ยนโหนดเดิม.....	44
4.6 รูปที่ 4.6 หน้าตาGUIของการเลือกสิ่งที่ต้องการแก้ไข.....	45

## สารบัญรูป ( ต่อ )

รูปที่	หน้า
4.7 รูปที่ 4.7 หน้าตาGUIของการแก้ไขเส้นทาง.....	46
4.8 รูปที่ 4.8 หน้าตาGUIของการเพิ่มโหนดใหม่.....	47
4.9 รูปที่ 4.9 หน้าตาGUIของการแก้ไขข้อมูลเส้นทาง.....	48
4.10 รูปที่ 4.10 หน้าตาGUIของการเลือกสิ่งที่ต้องการแก้ไข.....	48
4.11 รูปที่ 4.11 หน้าตาGUIของการออกและประมวลผล.....	49
4.12 รูปที่ 4.12 หน้าตาGUIของการแก้ไขและการเพิ่มงาน.....	50
4.13 รูปที่ 4.13 หน้าตาGUIของการแก้ไขและลบงาน.....	50
4.14 รูปที่ 4.14 หน้าตาGUIของการเพิ่มงาน.....	51
4.15 รูปที่ 4.15 หน้าตาGUIของการแก้ไขงาน.....	52
4.16 รูปที่ 4.16 หน้าตาGUIของการวางแผนการเดินทาง.....	53
4.17 รูปที่ 4.17 หน้าตาGUIของการแสดงเส้นทาง.....	54
4.18 รูปที่ 4.18 Action GUI.....	55
4.19 รูปที่ 4.19 Action GUI.....	56
4.20 รูปที่ 4.20 Action GUI.....	56
4.21 รูปที่ 4.21 Action GUI.....	57
4.22 รูปที่ 4.22 Action GUI.....	57
4.23 รูปที่ 4.23 Action GUI.....	58



# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญของปัญหา

เนื่องจากในปัจจุบันเครื่องปรับอากาศกลายเป็นสิ่งที่ได้รับความนิยมอย่างมากดังนั้นการบริการหลังการขายรวมถึงการติดตั้งใหม่ให้กับลูกค้าจึงมีอยู่อย่างสม่ำเสมอ โดยเฉพาะในเมืองใหญ่ๆที่มีอาคารบ้านเรือนและบริษัทสำนักงานต่างๆอีกมากมาย แต่การที่จะออกไปให้บริการแต่ละครั้งนั้นสิ่งที่ต้องคำนึงถึงก็คือค่าใช้จ่ายในการออกไปแต่ละครั้งว่าคุ้มค่าหรือไม่ประกอบกับในปัจจุบันที่ราคาน้ำมันปรับตัวสูงขึ้น เพราะฉะนั้นการที่จะออกไปให้บริการก็ต้องพิจารณาให้มากขึ้นตามไปด้วย

ดังนั้นผู้จัดทำโครงการจึงได้มีแนวคิดที่จะสร้างโปรแกรมที่จะช่วยในการตัดสินใจสำหรับผู้ประกอบการที่จะกำหนดการไปให้บริการในแต่ละครั้งเพื่อความคุ้มค่ากับค่าใช้จ่ายและเป็นการประหยัดน้ำมันให้มากที่สุด

### 1.2 วัตถุประสงค์

- 1.2.1 ศึกษาทฤษฎีกราฟ ในการหาเส้นทางที่สั้นที่สุด
- 1.2.2 ศึกษาภาษาที่เป็นโปรแกรมเชิงวัตถุ
- 1.2.3 ศึกษาเส้นทางภายในจังหวัดพิษณุโลก
- 1.2.4 ศึกษาข้อมูลการให้บริการชนิดต่างๆ เพื่อจะกำหนดเวลาในการให้บริการแต่ละชนิด

### 1.3 ขอบเขตของโครงการ

- 1.3.1 ใช้ภาษาจาวาในการเขียนโปรแกรม
- 1.3.2 โปรแกรมนี้จะแสดงเส้นทางเฉพาะในพื้นที่ จังหวัดพิษณุโลก
- 1.3.3 การแบ่งโซนของลูกค้าจะพิจารณาดังนี้คือ
  - ในกรณีที่อยู่ใน อ.เมืองจะแบ่งตามหมู่บ้าน, ซอย, ชุมชน
  - ในกรณีที่อยู่อำเภอรอบนอก จะแบ่งตามตำบล หรือกรณีที่เป็นตำบลใหญ่ๆก็จะ แบ่งออกเป็นส่วนๆ ตามความเหมาะสม
- 1.3.4 โปรแกรมจะแสดงผลดังนี้
  - แผนที่ที่จะใช้ในการเดินทาง
  - ตำแหน่งที่ผ่าน
  - ที่อยู่ของลูกค้า
  - ระยะทางทั้งหมด
  - ค่าใช้จ่ายในการเดินทาง (ค่าน้ำมัน)



### 1.5 ผลที่คาดว่าจะได้รับ

- 1.5.1 เพิ่มทักษะในการเขียน โปรแกรมมากขึ้น
- 1.5.2 มีความรู้ความเข้าใจ ในการออกแบบฐานข้อมูล
- 1.5.3 ได้ศึกษาเส้นทางภายในจังหวัดพิษณุโลก
- 1.5.4 ช่วยลดความยุ่งยากในการบริหารจัดการในแต่ละครั้งที่ส่วนใหญ่ผู้มักจะทำ
- 1.5.5 ช่วยตัดสินใจในการออกไปให้บริการลูกค้าในแต่ละครั้ง
- 1.5.6 ช่วยในการเก็บข้อมูลของลูกค้า

### 1.6 งบประมาณ

1.6.1 ค่าน้ำมันรถ,ค่ารถโดยสารสาธารณะ	500	บาท
1.6.2 ค่าหนังสือและค่าเอกสารแผนที่ต่างๆ	1500	บาท
1.6.3 ค่าจัดทำรูปเล่มโครงการ	500	บาท
1.6.4 ค่าวัสดุทางคอมพิวเตอร์ และอื่นๆ	500	บาท
รวมทั้งสิ้น	3000	บาท

(สามพันบาทถ้วน)



## บทที่ 2

# หลักการและทฤษฎีที่เกี่ยวข้อง

### 2.1 ทฤษฎีกราฟ

#### 2.1.1 พื้นฐานทั่วไปของกราฟ

ให้  $G(V, E)$  เป็นกราฟ ดังนั้นกราฟจะประกอบด้วย 2 สิ่งคือ

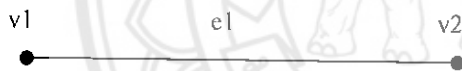
1.เซต  $V$  คือเซตที่มีเอลิเมนต์เป็น โหนด (node) หรือจุดยอด (vertices) หรือจุดเชื่อม (points หรือ nodes)

2.เซต  $E$  คือเซตของเอจ (edges) หรือด้าน หรือเส้นเชื่อมที่เชื่อมระหว่างจุดยอดต่างกันอย่างนิยาม

ถ้ากำหนดให้  $G$  เป็นสัญลักษณ์แทนความหมายของกราฟแล้วองค์ประกอบที่สำคัญของ  $G$  จะประกอบด้วยเซตของ โหนด (node) ซึ่งแทนด้วยสัญลักษณ์  $V$  และเซตของเอจ (edges) แทนด้วยสัญลักษณ์  $E$  และกราฟ  $G$  จะแทนที่ด้วย  $G(V, E)$  ถ้า  $\{v_1, v_2, v_3\}$  เป็นเซตของ โหนดใน  $V$  และ  $\{e_1, e_2, e_3\}$  เป็นเซตของเอจใน  $E$  แล้วเราสามารถเขียนความสัมพันธ์ของ โหนดและเอจได้ดังนี้

$$e_1 = (v_1, v_2)$$

โดยที่  $e_1$  หมายถึงเอจที่เชื่อมต่อระหว่าง โหนด  $v_1, v_2$  และจะเรียก  $v_1, v_2$  ว่าแอดจาเซนต์ โหนด (adjacent node) หรือ เนบอร์ (neighbor) ดังรูป



#### 2.1.2 กราฟที่ระบุทิศทาง

กราฟที่ระบุทิศทาง (Directed graph) เป็นประโยชน์มากในการศึกษาสถานการณ์เคลื่อนไหวนิว เช่น ดิจิตอล คอมพิวเตอร์ หรือระบบการไหลต่างๆ ดังนั้นมาศึกษาถึงคำจำกัดความพื้นฐาน และคุณสมบัติของกราฟระบุทิศทาง ดังนี้

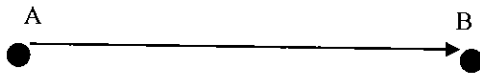
$D$  หรือไดกราฟ (digraph) เป็นกราฟระบุทิศทางประกอบด้วย 2 สิ่ง คือ

(1) เซต  $V$  ที่มีเอลิเมนต์ เรียกว่า จุดยอดหรือจุด หรือ โหนด (Vertices, points or nodes)

(2) เซต  $A$  ของคู่อันดับของจุดยอด ซึ่งเรียกว่าด้าน โค้ง หรืออาร์ค

และสามารถเขียนรูปของไดกราฟด้วยไคอะแกรมบนระนาบได้ โดยเขียนวงกลมเล็กๆ (dot แทนจุดยอด  $v$ ) แต่ละจุดใน  $v$  และเขียนเส้นแทนด้าน โค้งของ  $a = \langle u, v \rangle$  โดยเส้นแทนเส้น โค้งจะมีหัว

ถูกครด้วยเพื่อเป็นการระบุทิศทางของเส้นหรือ โค้งจากจุดเริ่มต้น (initial point)  $u$  ไปยังจุดปลาย  $v$  (terminal point) ของ  $a$



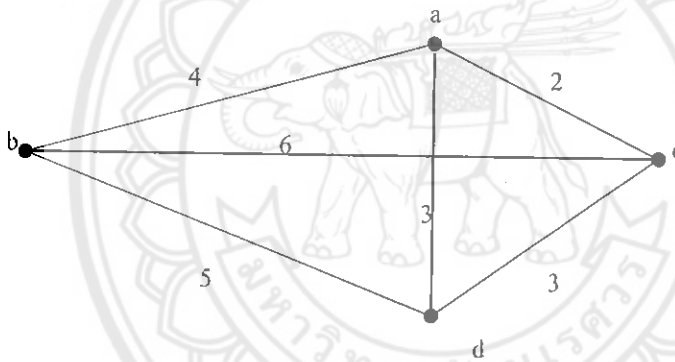
### 2.1.3 THE TRAVELING SALEMAN PROBLEM

เป็นวิธีการที่หาเส้นทางที่สั้นที่สุดในกรณีเดินทาง ตั้งแต่โหนดเริ่มต้นแล้วให้ผ่านโหนดต่างๆ แล้วกลับมาที่โหนดเริ่มต้นอีกครั้ง ตัวอย่างเช่น การเดินทางของเซลล์แมนที่จะเดินทางไปเมืองต่างๆ เพื่อขายของแล้วก็เดินทางกลับมาที่บริษัทในตอนท้าย แต่การเดินทางไปแต่ละที่นั้นยอมไปได้หลายเส้นทาง ดังนั้นเส้นทางที่เป็นไปได้ทั้งหมดที่จะใช้ในการเดินทางนั้นจะหาได้จาก

$(n-1)!/2$  โดยให้  $n$  แทนจำนวนของ เมืองทั้งหมด(โหนด)

และจะพิจารณาเส้นทางที่สั้นที่สุด

ยกตัวอย่างเช่น มีโหนดอยู่ 4 โหนดซึ่งแต่ละโหนดเชื่อมต่อกันดังรูป



รูปที่ 2.1 THE TRAVELING SALEMAN PROBLEM

1. a-b-d-c-a	$4+5+3+2 = 14$
2. a-b-c-d-a	$4+6+3+3 = 16$
3. a-d-b-c-a	$3+5+6+2 = 16$
4. a-c-d-b-a (เป็นเส้นทางที่วนกลับของ 1)	$2+3+5+4 = 14$
5. a-d-c-b-a (เป็นเส้นทางที่วนกลับของ 2)	$3+3+6+4 = 16$
6. a-c-b-d-a (เป็นเส้นทางที่วนกลับของ 3)	$2+6+5+3 = 16$

ตารางที่ 2.1 THE TRAVELING SALEMAN PROBLEM



สังเกตได้ว่าวิธีการเดินทางของ 4,5 และ 6 จะซ้ำกับเส้นทางที่ 1,2 และ 3 ตามลำดับ เพราะฉะนั้นจะเหลือเส้นทางที่เป็นไปได้อยู่ 3 เส้นทางคือ 1,2 และ 3 หรือถ้าไม่เลือก 1,2 และ 3 ก็จะเลือก 4,5 และ 6 แทนได้ และจะเห็นว่าเส้นทางที่ดีที่สุดคือ เส้นทางที่ 1 ที่มีระยะทางเท่ากับ 14 และ Algorithm ที่เหมาะสมที่จะใช้ในการหาค่า Shortest Path โหนด ก็คือ Dijkstra's Algorithm

#### 2.1.4 การแสดงกราฟด้วยเมทริกซ์

ให้  $G$  เป็นกราฟที่มีจุดยอด  $V_1, V_2, \dots, V_m$  และมีด้านเป็น  $E_1, E_2, \dots, E_n$  และเราสามารถจะแทนกราฟในรูปของเมทริกซ์ได้ ซึ่งจะสัมพันธ์กัน

เอลิเมนต์(Element) ในเมทริกซ์  $A$  เป็นดังนี้

ให้  $A = (a_{ij})$  เป็น  $m \times n$  เมทริกซ์ และจะเรียก  $A$  ว่าเป็นแอตจาเซ็นซีเมทริกซ์  $G$

$$\text{ถ้า } a_{ij} = \begin{cases} 1 & \text{ถ้า } \{V_i, V_j\} \text{ เป็นด้านหนึ่งของ } V_i \text{ ที่ประชิดกับ } V_j \\ 0 & \text{หากไม่มีเส้นทางที่ติดกัน} \end{cases}$$

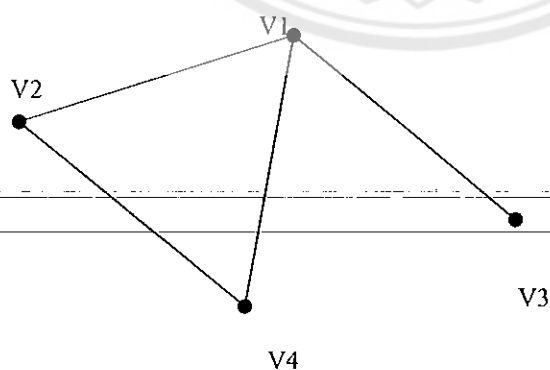
นั่นคือ  $a_{ij}$  จะเท่ากับ 1 ถ้า  $V_i$  เป็นแอตจาเซ็นซีกับ  $V_j$

และ  $a_{ij}$  จะเท่ากับ 0 ถ้า  $V_i$  และ  $V_j$  ไม่เป็นแอตจาเซ็นซี

ถ้า  $a_{ij} = a_{ji}$  แสดงว่า  $A$  เป็นเมทริกซ์สมมาตร

ในกรณีที่ เป็นมัลติกราฟ แอตจาเซ็นซีเมทริกซ์ของมัลติกราฟจะเป็นการใช้  $a_{ij}$  แทนจำนวนด้านของ  $\{V_i, V_j\}$

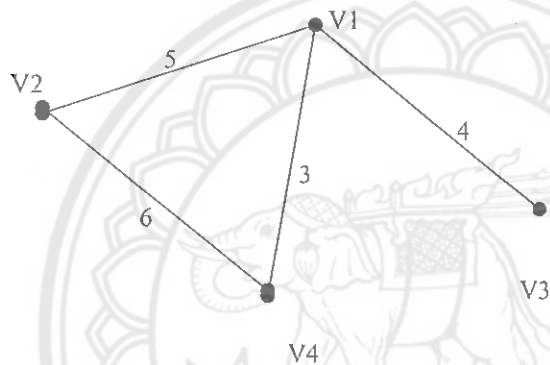
ตัวอย่างดังรูป



รูปที่ 2.2 การแสดงกราฟด้วยเมทริกซ์

$$A = \begin{array}{cccc} & V1 & V2 & V3 & V4 \\ \begin{array}{c} V1 \\ V2 \\ V3 \\ V4 \end{array} & \begin{array}{cccc} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{array} \end{array}$$

เราสามารถได้ระยะทางแทนที่ลงไปได้อดังนี้



รูปที่ 2.3 การแสดงกราฟด้วยเมทริกซ์

จากเมทริกซ์ A จะได้เป็น

$$A = \begin{array}{cccc} & V1 & V2 & V3 & V4 \\ \begin{array}{c} V1 \\ V2 \\ V3 \\ V4 \end{array} & \begin{array}{cccc} 0 & 5 & 4 & 3 \\ 5 & 0 & 0 & 6 \\ 4 & 0 & 0 & 0 \\ 3 & 6 & 0 & 0 \end{array} \end{array}$$

แล้วจะสามารถนำเทคนิคนี้ไปช่วยในการติดต่อกับคอมพิวเตอร์ในการเขียนโปรแกรม

## 2.2 ทฤษฎีของ Dijkstra's Algorithm

Dijkstra's Algorithm เป็น Algorithm ที่ใช้ในการค้นหาเส้นทางที่สั้นที่สุด (Shortest path) ซึ่งแต่ละเส้นทาง (path) จะมีการคิดค่าน้ำหนัก (weight) ในการค้นหาเส้นทางจากโหนดเริ่มต้น ไปยังโหนดปลายทาง โหนดจะมีลักษณะการทำงานดังนี้

1. โดยเราจะเรียกโหนดตามชื่อที่ให้ไป และแต่ละโหนด (v) เราจะอ้างถึงฟังก์ชัน  $d(v)$  และ  $pred(v)$  ในตอนเริ่มแรกทุกโหนดจะมีค่า  $d(v) = \text{Infinity}$  และ  $pred(v) = 0$

2. สำหรับเส้นทางที่เชื่อมระหว่างโหนด i และ j เราจะเรียกว่า  $arc(i,j)$  ส่วนระยะทางระหว่างโหนด i และ j เราจะเรียกว่า  $weight(i,j)$

3. เราจะแบ่งกลุ่มของโหนดเป็น 2 กลุ่ม คือ กลุ่มโหนดเส้นทึบ และ กลุ่มโหนดเส้นปะ ในตอนเริ่มต้นทุกโหนดจะเป็นเส้นปะ

4. เลือกโหนดเริ่มต้นจากกลุ่มเส้นปะ สมมติว่าเป็นโหนด k และ ให้  $d(k) = 0$

5. เลือกโหนด j จากกลุ่มเส้นปะ โดยที่มีค่า  $d(j)$  น้อยที่สุดในกลุ่มเส้นปะ ระบายเส้นทึบใส่โหนด j และ  $arc(j, pred(j))$

6. พิจารณาทุกเส้นทางที่มีปลายหนึ่งติดกับ โหนดที่มีเส้นทึบ และ อีกปลายหนึ่งยังเป็นเส้นปะ สมมติว่าเป็น  $arc(i,j)$  เราจะได้โหนด i เส้นทึบ และ โหนด j เส้นปะ และ เรียกโหนด j เป็นโหนดที่ติดกับโหนด i (Adjacent node) ถ้า  $d(i) + weight(i,j) < d(j)$  เราก็จะเปลี่ยนค่าของ  $d(j)$  โดยให้  $d(j) = d(i) + weight(i,j)$  และ  $pred(j) = i$

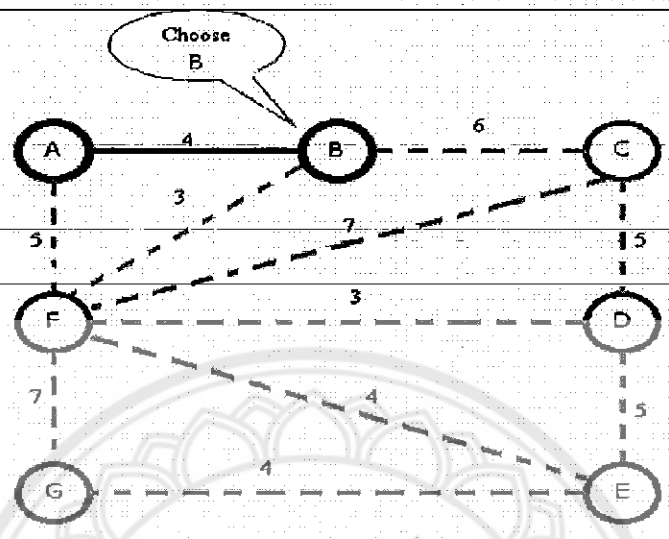
7. ทำแบบขั้นที่ 5, และ 6 จนกระทั่งทุกโหนดเป็นเส้นทึบ

แล้วเราก็จะได้เส้นทางที่สั้นที่สุดจากโหนดที่เริ่มต้นไปยังทุกๆ โหนด

หมายเหตุ  $d(v)$  และ  $pred(v)$  คือ เป็นฟังก์ชันที่จะคอยบอกเราว่า ระยะทางที่สั้นที่สุดจากจุดเริ่มต้นถึงโหนด v ต้องเดินทางเป็นระยะทางอย่างน้อยเท่ากับ  $d(v)$  และ ต้องเดินทางผ่าน โหนด  $pred(v)$

ตัวอย่างของ Dijkstra's Algorithm

รอบที่ 1



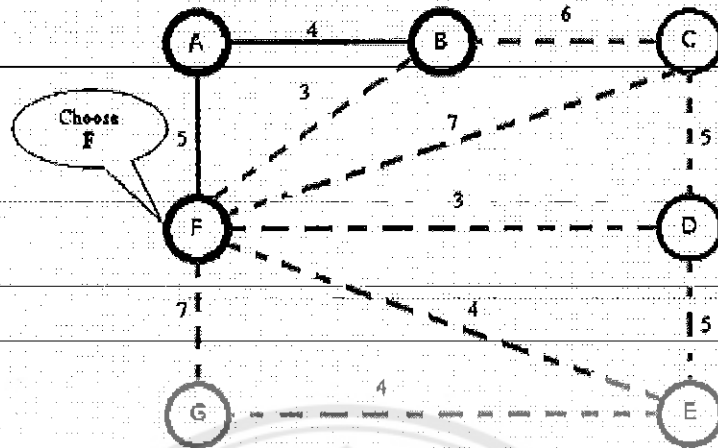
รูปที่ 2.4 Dijkstra's Algorithm

รอบที่ 1	โหนดที่มีสีแสดล้อมรอบ	Adjacent Node	เส้นทางที่พิจารณา	weight(i,j)	weight(i,j) + d(i)	d(v)
1	A	B	(A,B)	4	4	4
		F	(A,F)	5	5	5

โหนด	d(v)	pred(v)
A	0	0
B	4	A
C	infinity	0
D	infinity	0
E	infinity	0
F	infinity	0
G	infinity	0

ตารางที่ 2.2 Dijkstra's Algorithm

รอบที่ 2



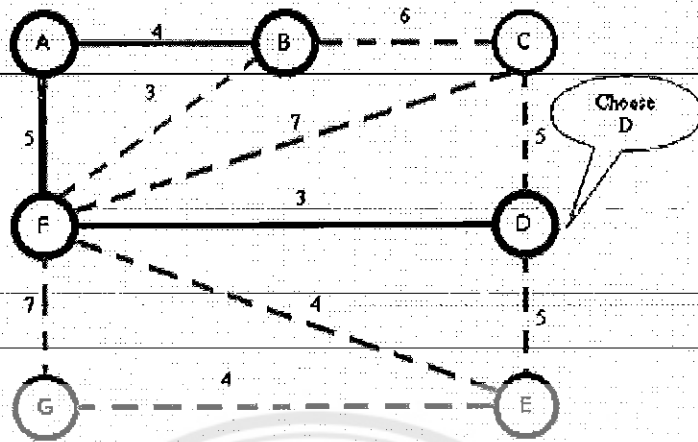
รูปที่ 2.5 Dijkstra's Algorithm

รอบที่	โหนดที่มีสีแดงล้อมรอบ	Adjacent node	เส้นทางที่พิจารณา	weight(i,j)	weight(i,j) + d(i)	ค่าใหม่ ของ d(v)
2	A,B	C	(B,C)	6	10	10
		F	(A,F)	5	5	5
			(B,F)	3	7	5

โหนด	d(v)	pred(v)
A	0	0
B	4	A
C	infinity	0
D	infinity	0
E	infinity	0
F	5	A
G	infinity	0

ตารางที่ 2.3 Dijkstra's Algorithm

รอบที่ 3



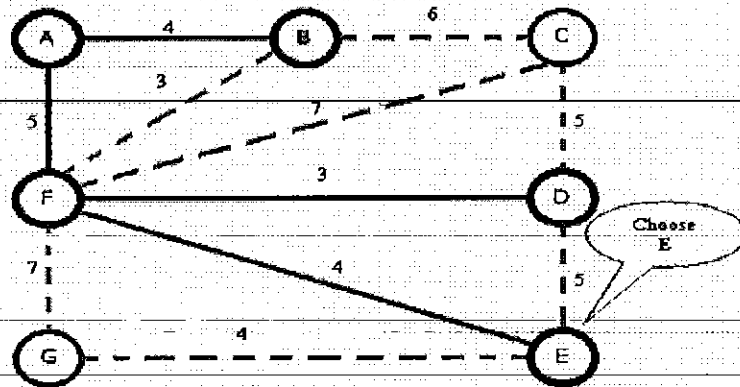
รูปที่ 2.6 Dijkstra's Algorithm

รอบที่	โหนดที่มีสีแดง ล้อมรอบ	Adjacent node	เส้นทางที่ พิจารณา	weight(i,j)	weight(i,j) +d(i)	ค่าใหม่ ของ d(v)
3	A,B,F	C	(B,C)	6	10	10
		D	(D,F)	3	8	8
		E	(E,F)	4	9	9
		G	(F,G)	7	12	12

โหนด	d(v)	pred(v)
A	0	0
B	4	A
C	infinity	0
D	8	F
E	infinity	0
F	5	A
G	infinity	0

ตารางที่ 2.4 Dijkstra's Algorithm

รอบที่ 4



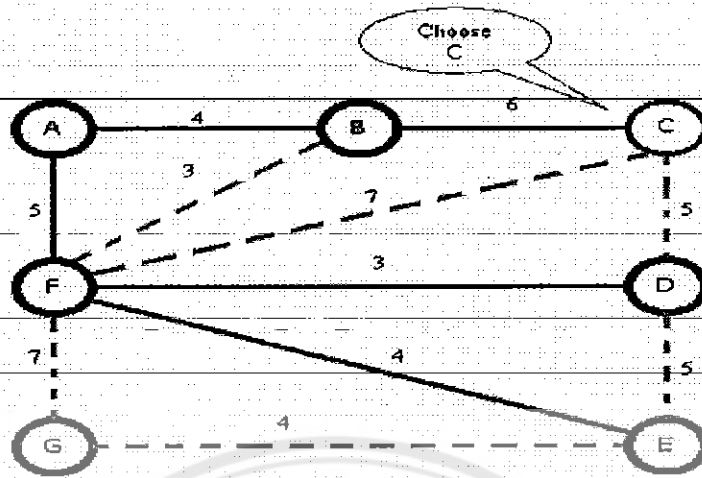
รูปที่ 2.7 Dijkstra's Algorithm

รอบที่	โหนดที่มีสีแสด คือรอบ	Adjacent node	เส้นทางที่ พิจารณา	weight(i,j)	weight(i,j) +d(i)	ค่าใหม่ ของ d(v)
4	A,B,D,F	C	(B,C)	6	10	10
			(D,C)	5	13	10
		E	(D,E)	5	13	9
			(F,E)	4	9	9
		G	(F,G)	7	12	12

โหนด	d(v)	pred(v)
A	0	0
B	4	A
C	infinity	0
D	8	F
E	9	F
F	5	A
G	infinity	0

ตารางที่ 2.5 Dijkstra's Algorithm

รอบที่ 5



รูปที่ 2.8 Dijkstra's Algorithm

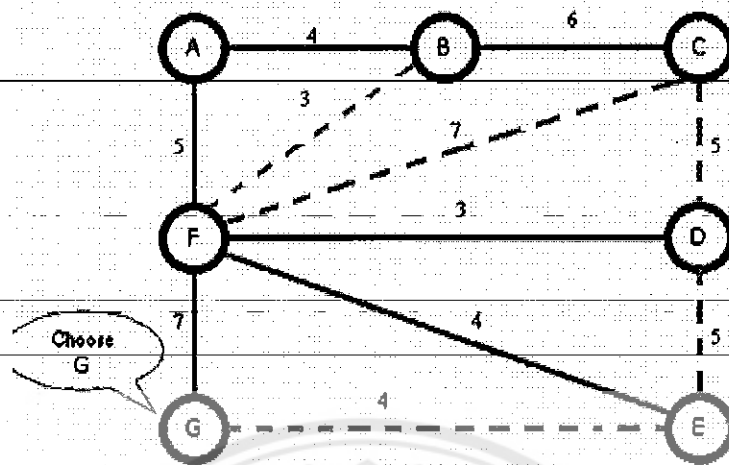
รอบที่	โหนดที่มีสีแดง	Adjacent node	เส้นทางที่พิจารณา	weight(i,j)	weight(i,j) + d(i)	ค่าใหม่ของ d(v)
5	A,B,D,E,F	C	(B,C)	6	10	10
			(D,C)	5	13	10
		G	(E,G)	4	13	12
			(F,G)	7	12	12

โหนด	d(v)	pred(v)
A	0	0
B	4	A
C	10	B
D	8	F
E	9	F
F	5	A
G	infinity	0

ตารางที่ 2.6 Dijkstra's Algorithm



รอบที่ 6



รูปที่ 2.9 Dijkstra's Algorithm

รอบที่	โหนดที่มีสีแดง	Adjacent node	เส้นทางที่พิจารณา	weight(i,j)	weight(i,j) + d(i)	ค่าใหม่ของ d(v)
6	A,B,D,E,F,G	G	(E,G)	4	13	12
			(F,G)	7	12	12

โหนด	d(v)	pred(v)
A	0	0
B	4	A
C	10	B
D	8	F
E	9	F
F	5	A
G	12	F

ตารางที่ 2.7 Dijkstra's Algorithm

เราก็จะได้ว่าถ้าเราต้องการเดินทางจาก A ไปยัง E เราต้องเดินทางผ่านเมือง F แล้วมีระยะทางในการเดินทาง เท่ากับ 9

## 2.3 ทฤษฎีของการเรียงสับเปลี่ยน Permutation

เป็นการจัดเรียงสิ่งของโดยมีลำดับหรือมีขั้นตอน เช่น การเรียงอักษรครวละ 3 ตัว จากอักษร 4 ตัว ABCD ในกรณีนี้เมื่อทดลองเรียงจริงจะได้ทั้งหมด 24 วิธีดังนี้

ABC ABD ADB ACB ACD ADC

BAC BAD BCA BCD BDA BDC

CAB CAD CBA CBD CDA CDB

DAB DAC DBA DBC DCA DCB

### 2.3.1 จำนวนวิธีของการเรียงสับเปลี่ยนสิ่งของครวละ r สิ่ง จากสิ่งของ n สิ่งที่แตกต่างกัน

จะมีวิธีโดยอาศัยแนวความคิดของหลักการนับ

ตำแหน่งแรกของการเรียงเราเลือกสิ่งของอันใดอันหนึ่ง จาก n สิ่งได้ n วิธี

และตำแหน่งที่สองของการเรียงเราเลือกสิ่งของอันใดอันหนึ่ง จากสิ่งที่เหลือได้  $n-1$  วิธี

และตำแหน่งที่ r ของการเรียงเราเลือกสิ่งของอันใดอันหนึ่ง ได้  $n-r+1$  วิธี

ดังนั้นจำนวนวิธีทั้งหมดของการเรียงสับเปลี่ยนเท่ากับ  $(n)(n-1)(n-2) \dots (n-r+1)$

เขียนแทนด้วย  $P(n,r) = (n)(n-1)(n-2) \dots (n-r+1)$

โดยใช้สัญลักษณ์ของแฟคตอเรียลจะได้ว่า  $P(n,r) = n!/(n-r)!$

เราสรุปเป็นกฎสำหรับจำนวนวิธีของการเรียงสับเปลี่ยน ได้ดังนี้จำนวนวิธีของการเรียงสับเปลี่ยนสิ่งของครวละ r สิ่ง จากสิ่งของ n สิ่งที่แตกต่างกัน เขียนแทนด้วย  $P(n,r)$  หรือ  $nPr$  จะได้ว่า

$$P(n,r) = n!/(n-r)!$$

### 2.3.2 จำนวนวิธีของการเรียงสับเปลี่ยนสิ่งของ n สิ่งที่แตกต่างกัน ครวละ n สิ่ง

$$P(n,n) = n!$$

$$\text{จาก } P(n,r) = n!/(n-r)!$$

$$\text{จะได้ว่า } P(n,n) = n!/(n-n)!$$

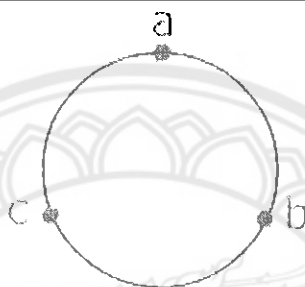
$$= n!/0!$$

$$= n!$$

ตัวอย่าง จงหาจำนวนวิธีในการเรียงสับเปลี่ยน ตัวอักษร 3 ตัว

$$\begin{aligned} \text{วิธีทำ จาก } P(n,n) &= n! \\ \text{จะได้ว่า } P(3,3) &= 3! \\ &= (3)(2)(1) = 6 \text{ วิธี} \end{aligned}$$

2.3.3 จำนวนวิธีของการเรียงสับเปลี่ยนสิ่งของ  $n$  สิ่งที่แตกต่างกัน คราวละ  $n$  สิ่งเป็นวงกลม การจัดเรียงสับเปลี่ยนสิ่งของเป็นวงกลมมีความแตกต่างจากการเรียงลำดับเป็นแนวเส้นตรง เพราะว่าวงกลมไม่มีจุดหัวท้ายที่แน่นอนเหมือนเส้นตรง



รูปที่ 2.10 Dijkstra's Algorithm

จะสังเกตเห็นว่าแต่ละวิธีของการเรียงแบบวงกลม เช่น abc เมื่อเลื่อนลำดับจุดเริ่มต้นเป็นจุดต่อไปก็จะได้เป็น bca และ cab ตามลำดับ ซึ่งจะนับได้เป็นสามวิธีของการเรียงสับเปลี่ยนแบบเส้นตรง ทั้งๆที่สามวิธีดังกล่าวถือเป็นวิธีเดียวกันในการเรียงแบบวงกลม

ดังนั้นในการเรียงสับเปลี่ยนแบบวงกลมจะต้องตรึงจุดหนึ่งไว้เป็นจุดคงที่ ส่วนที่เหลืออีก  $n-1$  จุด นำไปจัดเรียงแบบเส้นตรงได้  $(n-1)!$  วิธี

จำนวนวิธีของการเรียงสิ่งของ  $n$  สิ่งที่แตกต่างกันเป็นวงกลม

จะเรียงได้  $(n-1)!$  วิธี

2.3.4 จำนวนวิธีของการเรียงสับเปลี่ยนสิ่งของ  $n$  สิ่งที่มีสิ่งของเหมือนกัน  $k$  สิ่ง

ในการเรียงสับเปลี่ยนสิ่งของ  $n$  สิ่งที่แตกต่างกัน คราวละ  $n$  สิ่ง ได้จำนวนวิธีเป็น  $n!$  วิธี แต่ถ้าในจำนวนสิ่งของ  $n$  สิ่ง มีของที่เหมือนกัน  $k$  สิ่ง เราจะเห็นได้ว่าของที่เหมือนกันเมื่อสลับตำแหน่งกันก็จะได้วิธีเรียงสับเปลี่ยนที่ไม่แตกต่างกัน

และในสิ่งของ  $k$  จะสลับตำแหน่งกันได้  $k!$  วิธีซึ่งจะถือเป็นจำนวนวิธีเพียง 1 วิธีของการเรียงสิ่งของทั้งหมด

ดังนั้นจากจำนวนวิธีเดิม  $n!$  วิธีและมีสิ่งของเหมือนกัน  $k$  สิ่งจะนับจำนวนวิธีเรียงสับเปลี่ยนที่แตกต่างกัน  $n!/k!$  วิธี

จำนวนวิธีของการเรียงสับเปลี่ยนสิ่งของ  $n$  สิ่งที่มีสิ่งของเหมือนกัน  $k$  สิ่ง

จะเรียงได้  $n!/k!$  วิธี

## 2.4 ภาษา JAVA

จาวาเป็นภาษาสำหรับการเขียนโปรแกรมที่ถูกสร้างขึ้นมาเพื่อรองรับการออกแบบซอฟต์แวร์ที่มีการเชื่อมโยงอินเทอร์เน็ต โดยภาษาจาวานั้นสนับสนุนแนวความคิดของ โปรแกรมเชิงวัตถุ หรือที่รู้จักกันดีที่เรียกว่า OOP (Object - Oriented Programming) โดยมีความสามารถเฉพาะตัวต่างจากโปรแกรมภาษาชั้นสูงอื่นๆ เช่น C, C++ ในเรื่องของการทำงานข้ามระบบปฏิบัติการ หรือ Platform ได้โดยไม่ต้องมีคอมไพร์ใหม่

จาวาเป็นโปรแกรมที่เกิดขึ้นมาไม่นานแต่มีโครงสร้างของภาษาและความสามารถที่น่าใช้งานอยู่หลายประการ จาวาเป็นภาษาโปรแกรมที่สามารถนำไปสร้างสรรค์งานได้ในแทบทุกรูปแบบที่ภาษาโปรแกรมชั้นดีทำได้ แต่เนื่องจากที่เป็นภาษาเขียนโปรแกรมแบบ OOP ทำให้ไปสร้างความหวาดหวั่นใจในความยุ่งยากให้กับโปรแกรมมือใหม่บ้าง ขณะที่นักเขียนโปรแกรมแบบดั้งเดิมที่ใช้ภาษา ซี , ปาสคาล แม้จะสนใจอยากจะได้เรียนรู้ภาษาจาวา แต่ก็อาจมีอุปสรรคเกี่ยวกับความไม่เข้าใจ หรือสับสนในโครงสร้างการทำงานของโปรแกรม

นอกจากเนื้อหาการเขียนโปรแกรมพื้นฐานที่ควรรู้แล้ว จาวายังมีการพัฒนาแพ็คเกจต่างๆ ที่สามารถนำไปใช้งานในด้านเว็บเพจอีกมากมาย เช่น Applet, Swing, JSP, JavaBeans รวมทั้งการเชื่อมต่อฐานข้อมูลด้วย JDBC

### 2.4.1 รูปแบบของการเขียนโปรแกรมในภาษาจาวา

โปรแกรมที่ถูกพัฒนาด้วยภาษาจาวา ถูกแบ่งออกเป็นสองรูปแบบหลักๆคือ

1. จาวา Application เป็น โปรแกรมจาวาทั่วไปที่ทำงานได้ด้วยตัวของมันเอง (Stand Alone Application-เหมือนกับไฟล์ .EXE ใน Windows) ซึ่งสามารถใช้งานได้ทั้งบน Windows ลินุกซ์ และยูนิกซ์ ตระกูลต่างๆ

2. จาวา Applet เป็นโปรแกรมจาวาขนาดเล็กซึ่งรันด้วยตัวเองไม่ได้ ต้องถูกโปรแกรมเรียกไปใช้งานและจะถูกนำไปใช้บนอินเทอร์เน็ตเท่านั้น

### 2.4.2 การใช้ Java ติดต่อกับฐานข้อมูล

Java สามารถเขียนโปรแกรมติดต่อกับฐานข้อมูล เพื่อดำเนินการอย่างใดอย่างหนึ่งได้ เช่น การเพิ่ม การลบ การปรับปรุง การเรียกดู เป็นต้น การเขียนโปรแกรมโดยใช้ภาษา Java ติดต่อกับฐานข้อมูล จะต้องอาศัยตัวกลางในการดำเนินการติดต่อระหว่างภาษา Java กับฐานข้อมูลนั้นๆซึ่งเรียกว่า“Database Driver” ในที่นี้จะใช้ Driver ที่ชื่อว่า “JDBC”

### 2.4.3 การเขียนโปรแกรมติดต่อกับฐานข้อมูลโดยใช้ JDBC

มีขั้นตอนการทำงานดังนี้

#### 1. สร้างการติดต่อกับฐานข้อมูล มี2ขั้นตอน คือ

1.1 การโหลด Driverขึ้นมาทำงาน สามารถทำได้โดยเรียกใช้ static Method“forName ()” ซึ่งมีรูปแบบดังนี้

```
Class.forName (“DriverName”)
```

DriverName คือ ชื่อdriver ที่ต้องการ โหลดขึ้นมาเพื่อใช้ติดต่อกับฐานข้อมูล เช่น

```
Class.forName (“com.mysql.jdbc.Driver”)
```

1.2 การเชื่อมต่อกับฐานข้อมูล จะต้องใช้ข้อมูล3อย่างคือ URL, User Login และ Password URL ข้อมูลที่ระบุเส้นทางไปยังฐานข้อมูลที่ต้องการติดต่อที่อาจเก็บไว้ในserver หรือภายในเครื่อง client ยกตัวอย่างเช่น

User Login ชื่อของผู้มีสิทธิเข้าไปใช้งานฐานข้อมูล แต่ถ้าหากไม่กำหนดUser Login ไว้ก็สามารถละเว้นได้

Password รหัสผ่านของผู้มีสิทธิเข้าใช้งานซึ่งจะใช้คู่กับ User Login เสมอ หากละเว้น User Login ก็ต้องละเว้น Password ด้วยเช่นกัน

หลังจากได้ข้อมูลที่ต้องการแล้วให้นำข้อมูลทั้งหมดมาใช้เป็น Argumentให้กับ

Method“getConection ()” ดังนี้

```
Connection OjectName = DriverManager.getConnection (url, userName, password);
```

(กรณีทีละเว้น User Login ก็ให้เหลือเฉพาะurl)

2. การดำเนินการกับฐานข้อมูล หลังจากที่เชื่อมต่อกับฐานข้อมูลแล้วการที่จะดำเนินการกับฐานข้อมูลด้วยคำสั่ง SQL จะต้องใช้ Method ของ Class Statement และ Class Prepared Statement

Statement ใช้งานกับประโยค SQL เพียงครั้งเดียว

PreparedStatement เพิ่มความสามารถในการสร้างประโยค SQL ที่ถูกคอมไพล์ไว้แล้วนำมาเรียกใช้ได้อีก ทำงานได้เร็วกว่าการใช้ Statement

ตัวอย่างโปรแกรมที่ใช้ Statement

#### 2.4.4 การดาวน์โหลด Database Driver และการติดตั้ง

Database Driver สำหรับติดต่อกับ MySQL สามารถดาวน์โหลดได้จาก

[www.mysql.com/products/connector/](http://www.mysql.com/products/connector/) จากนั้นเลือกเวอร์ชันที่ออกใหม่ที่สุดในที่นี้ใช้

เวอร์ชัน 5.0.5 ที่เป็นไฟล์ .zip จากนั้นทำการแตกไฟล์แล้วนำไฟล์ .jar ไปเก็บไว้ใน

C:\Program Files\Java\jdk1.5.0\_09\lib

จากนั้นทำการ set environment ดังนี้

CLASSPATH = .;%p%;%CLASSPATH%

p = C:\jdk1.5.2\_13\lib\mysql-connector-java-5.0.5-bin.jar

JAVA\_HOME = C:\Program Files\Java\jdk1.5.0\_09

#### 2.5 ความรู้ทั่วไปเกี่ยวกับระบบฐานข้อมูล

ฐานข้อมูลเป็นการจัดเก็บข้อมูลอย่างเป็นระบบ ทำให้ผู้ใช้สามารถใช้ข้อมูลที่เกี่ยวข้องในระบบงานต่างๆ ร่วมกันได้ โดยที่จะไม่เกิดความซ้ำซ้อนของข้อมูล และยังสามารถหลีกเลี่ยงความขัดแย้งของข้อมูลด้วย อีกทั้งข้อมูลในระบบก็จะต้องเชื่อถือได้ และเป็นมาตรฐานเดียวกัน โดยจะมีการกำหนดระบบความปลอดภัยของข้อมูลขึ้น

นับได้ว่าปัจจุบันเป็นยุคของสารสนเทศ เป็นที่ยอมรับกันว่า สารสนเทศเป็นข้อมูลที่ผ่านการกลั่นกรองอย่างเหมาะสม สามารถนำมาใช้ประโยชน์อย่างมากมาย ไม่ว่าจะเป็นการนำมาใช้งานด้านธุรกิจ การ

บริหาร และกิจการอื่นๆ องค์กรที่มีข้อมูลปริมาณมากๆ จะพบความยุ่งยากลำบากในการจัดเก็บข้อมูล ตลอดจนการนำข้อมูลที่ต้องการออกมาใช้ให้ทันต่อเหตุการณ์ ดังนั้นคอมพิวเตอร์จึงถูกนำมาใช้เป็นเครื่องมือช่วยในการจัดเก็บข้อมูล การประมวลผลข้อมูล ซึ่งทำให้ระบบการจัดเก็บข้อมูลเป็นไปได้อย่างสะดวก ทั้งนี้ โปรแกรมแต่ละโปรแกรมจะต้องสร้างวิธีควบคุมและจัดการกับข้อมูลขึ้นเอง ฐานข้อมูลจึง

เข้ามามีบทบาทสำคัญอย่างมาก โดยเฉพาะระบบงานต่างๆ ที่ใช้คอมพิวเตอร์ การออกแบบและพัฒนา  
ระบบฐานข้อมูล จึงต้องคำนึงถึงการควบคุมและการจัดการความถูกต้อง ตลอดจนประสิทธิภาพในการ  
เรียกใช้ข้อมูลด้วย

### 2.5.1 ความสำคัญของการประมวลผลแบบระบบฐานข้อมูล

จากการจัดเก็บข้อมูลรวมเป็นฐานข้อมูลจะก่อให้เกิดประโยชน์ดังนี้

1) สามารถลดความซ้ำซ้อนของข้อมูลได้ การเก็บข้อมูลชนิดเดียวกันไว้หลายๆ ที่ ทำให้  
เกิดความซ้ำซ้อน (Redundancy) ดังนั้นการนำข้อมูลมารวมเก็บไว้ในฐานข้อมูล จะช่วยลดปัญหาการเกิด  
ความซ้ำซ้อนของข้อมูลได้ โดยระบบจัดการฐานข้อมูล (Database Management System: DBMS) จะ  
ช่วยควบคุมความซ้ำซ้อนได้ เนื่องจากระบบจัดการฐานข้อมูลจะทราบได้ตลอดเวลาว่ามีข้อมูลซ้ำซ้อน  
กันอยู่ที่ใดบ้าง

2) หลีกเลี่ยงความขัดแย้งของข้อมูลได้ หากมีการเก็บข้อมูลชนิดเดียวกันไว้หลาย ๆ ที่และ  
มีการปรับปรุงข้อมูลเดียวกันนี้ แต่ปรับปรุงไม่ครบทุกที่ที่มีข้อมูลเก็บอยู่ก็จะทำให้เกิดปัญหาข้อมูลชนิด  
เดียวกัน อาจมีค่าไม่เหมือนกันในแต่ละที่ที่เก็บข้อมูลอยู่ จึงก่อให้เกิดความขัดแย้งของข้อมูลขึ้น  
(Inconsistency)

3) สามารถใช้ข้อมูลร่วมกันได้ ฐานข้อมูลจะเป็นการจัดเก็บข้อมูลรวมไว้ด้วยกัน ดังนั้น  
หากผู้ใช้ต้องการใช้ข้อมูลในฐานข้อมูลที่มาจากแฟ้มข้อมูลต่างๆ ก็จะสามารถทำได้โดยง่าย

4) สามารถรักษาความถูกต้องเชื่อถือได้ ของข้อมูลการจัดเก็บข้อมูลในฐานข้อมูล อาจมี  
ข้อผิดพลาดที่เกิดขึ้น เช่น จากการที่ผู้ป้อนข้อมูลป้อนข้อมูลผิดพลาด คือป้อนจากตัวเลขหนึ่งไปเป็นอีก  
ตัวเลขหนึ่ง โดยเฉพาะกรณีมีผู้ใช้หลายคนต้องใช้ข้อมูลจากฐานข้อมูลร่วมกัน หากผู้ใช้คนใดคนหนึ่ง  
แก้ไขข้อมูลผิดพลาดก็ทำให้ผู้อื่นได้รับผลกระทบตามไปด้วย ในระบบจัดการฐานข้อมูล (DBMS) จะ  
สามารถใส่กฎเกณฑ์เพื่อควบคุมความผิดพลาดที่เกิดขึ้น

5) สามารถกำหนดความเป็นมาตรฐานเดียวกันของข้อมูลได้ การเก็บข้อมูลร่วมกันไว้ใน  
ฐานข้อมูลจะทำให้สามารถกำหนดมาตรฐานของข้อมูลได้รวมทั้งมาตรฐานต่าง ๆ ในการจัดเก็บข้อมูล  
ให้เป็นไปในลักษณะเดียวกันได้ เช่นการกำหนดรูปแบบการเขียนวันที่ ในลักษณะ วัน/เดือน/ปี หรือ ปี/  
เดือน/วัน ทั้งนี้จะมีผู้ที่คอยบริหารฐานข้อมูลที่เรียกว่า ผู้บริหารฐานข้อมูล (Database Administrator:  
DBA) เป็นผู้กำหนดมาตรฐานต่างๆ

6) สามารถกำหนดระบบความปลอดภัยของข้อมูลได้ ระบบความปลอดภัยในที่นี้เป็นการ  
ป้องกันไม่ให้ผู้ใช้ที่ไม่มีสิทธิมาใช้ หรือมาเห็นข้อมูลบางอย่างในระบบ ผู้บริหารฐานข้อมูลจะสามารถ  
กำหนดระดับการเรียกใช้ข้อมูลของผู้ใช้แต่ละคนได้ตามความเหมาะสม

7) เกิดความเป็นอิสระของข้อมูล ในระบบฐานข้อมูลจะมีตัวจัดการฐานข้อมูลที่ทำหน้าที่  
เป็นตัวเชื่อมโยงกับฐานข้อมูล โปรแกรมต่าง ๆ อาจไม่จำเป็นต้องมีโครงสร้างข้อมูลทุกครั้ง ดังนั้นการ

แก้ไขข้อมูลบางครั้ง จึงอาจกระทำเฉพาะกับโปรแกรมที่เรียกใช้ข้อมูลที่เปลี่ยนแปลงเท่านั้น ส่วนโปรแกรมที่ไม่ได้เรียกใช้ข้อมูลดังกล่าว ก็จะเป็นอิสระจากการเปลี่ยนแปลง

### 2.5.2 โปรแกรมฐานข้อมูลที่นิยมใช้

1) โปรแกรมฐานข้อมูล เป็นโปรแกรมหรือซอฟต์แวร์ที่ช่วยจัดการข้อมูลหรือรายการต่าง ๆ ที่อยู่ในฐานข้อมูล ไม่ว่าจะเป็นการจัดเก็บ การเรียกใช้ การปรับปรุงข้อมูล โปรแกรมฐานข้อมูล จะช่วยให้ผู้ใช้สามารถค้นหาข้อมูลได้อย่างรวดเร็ว ซึ่งโปรแกรมฐานข้อมูลที่นิยมใช้มีอยู่ด้วยกันหลายตัว เช่น Access, FoxPro, Clipper, dBase, FoxBASE, Oracle, SQL เป็นต้น โดยแต่ละโปรแกรมจะมีความสามารถต่างกัน บางโปรแกรมใช้ง่ายแต่จะจำกัดขอบเขตการใช้งาน บางโปรแกรมใช้งานยากกว่า แต่จะมีความสามารถในการทำงานมากกว่า

2) โปรแกรม Access นับเป็นโปรแกรมที่นิยมใช้กันมากในขณะนี้ โดยเฉพาะในระบบฐานข้อมูลขนาดใหญ่ สามารถสร้างแบบฟอร์มที่ต้องการจะเรียกดูข้อมูลในฐานข้อมูล หลังจากนั้นก็กรอกข้อมูลในฐานข้อมูลเรียบร้อยแล้ว จะสามารถค้นหาหรือเรียกดูข้อมูลจากเขตข้อมูลใดก็ได้ นอกจากนี้ Access ยังมีระบบรักษาความปลอดภัยของข้อมูล โดยการกำหนดรหัสผ่านเพื่อป้องกันความปลอดภัยของข้อมูลในระบบได้ด้วย

3) โปรแกรม FoxPro เป็นโปรแกรมฐานข้อมูลที่มีผู้ใช้งานมากที่สุด เนื่องจากใช้ง่ายทั้งวิธีการเรียกจากเมนูของ FoxPro และประยุกต์โปรแกรมขึ้นใช้งาน โปรแกรมที่เขียนด้วย FoxPro จะสามารถใช้กับ dBase คำสั่งและฟังก์ชันต่าง ๆ ใน dBase จะสามารถใช้งานบน FoxPro ได้ นอกจากนี้ใน FoxPro ยังมีเครื่องมือช่วยในการเขียนโปรแกรม เช่น การสร้างรายงาน

4) โปรแกรม dBase เป็นโปรแกรมฐานข้อมูลชนิดหนึ่ง การใช้งานจะคล้ายกับโปรแกรม FoxPro ข้อมูลรายงานที่อยู่ในไฟล์บน dBase จะสามารถส่งไปประมวลผลในโปรแกรม Word Processor ได้ และแม้แต่ Excel ก็สามารอ่านไฟล์ .DBF ที่สร้างขึ้นโดยโปรแกรม dBase ได้ด้วย

5) โปรแกรม SQL เป็นโปรแกรมฐานข้อมูลที่มีโครงสร้างของภาษาที่เข้าใจง่าย ไม่ซับซ้อน มีประสิทธิภาพการทำงานสูง สามารถทำงานที่ซับซ้อนได้โดยใช้คำสั่งเพียงไม่กี่คำสั่ง โปรแกรม SQL จึงเหมาะที่จะใช้กับระบบฐานข้อมูลเชิงสัมพันธ์ และเป็นภาษาหนึ่งที่มีผู้นิยมใช้กันมาก โดยทั่วไปโปรแกรมฐานข้อมูลของบริษัทต่าง ๆ ที่มีใช้อยู่ในปัจจุบัน เช่น Oracle, DB2 ก็มักจะมีคำสั่ง SQL ที่ต่างจากมาตรฐานไปบ้างเพื่อให้เป็นจุดเด่นของแต่ละโปรแกรมไป



## 2.6 การเขียน MySQL

SQL ย่อมาจาก Structured Query Language เป็นภาษาที่ใช้ในการจัดการข้อมูลของฐานข้อมูลเชิงสัมพันธ์ MySQL เป็นระบบจัดการฐานข้อมูลที่นิยมใช้กันอย่างแพร่หลาย ไม่ว่าจะเป็นผู้ใช้ทั่วไปหรือหน่วยงานองค์กรหลายแห่ง ต่างก็นำ MySQL ไปประยุกต์ใช้ในการพัฒนาระบบสารสนเทศด้วยกันทั้งสิ้น ด้วยเหตุที่สามารถใช้งานได้ฟรีในแพลตฟอร์ม โดยไม่เสียค่าใช้จ่าย ทั้งยังมีประสิทธิภาพและความเร็วในการทำงานไม่แพ้ระบบการจัดการฐานข้อมูลอื่นๆ

### 2.6.1 ประโยชน์ของ SQL

SQL เป็นภาษาเกี่ยวกับการจัดการฐานข้อมูล ไม่ว่าจะใช้ในในเรื่องของการนิยามข้อมูล การเรียกใช้ หรือการควบคุม การใช้คำสั่งเหล่านี้ที่มีในระบบจัดการฐานข้อมูล (Database Management System: DBMS) เช่น ACCESS dBase ORACLE, DB2 ฯลฯ จะช่วยประหยัดเวลาในการพัฒนาระบบงาน หรือนำไปใช้ในส่วนของการสร้างฟอร์ม (FORM) การทำรายงาน (REPORT) ของระบบงานต่างๆ ได้รวดเร็วยิ่งขึ้น ประโยชน์ของภาษา SQL อีกประการก็คือ โปรแกรมระบบฐานข้อมูลส่วนใหญ่สนับสนุนภาษา SQL แทบทั้งสิ้น

### 2.6.2 คำสั่งในภาษา SQL

#### 1) คำสั่งสร้างฐานข้อมูล

Syntax

```
CREATE DATABASE ชื่อฐานข้อมูล
```

เช่น CREATE DATABASE person

#### 2) คำสั่งสร้างลบข้อมูล

Syntax

```
DROP DATABASE ชื่อฐานข้อมูล
```

เช่น DROP DATABASE person

#### 3) คำสั่งสร้างตารางข้อมูล

Syntax

```
CREATE TABLE table_name (
```

```
{ column_name data_type [ DEFAULT default_expr ] [ column_constraint [ ... ] ] |
```

```
table_constraint } [, ... ]
```

)

### 2.6.3 การกำหนดข้อบังคับ (Constraint)

Constraint มีไว้เพื่อป้องกันไม่ให้คีย์ข้อมูลที่ไม่ถูกต้องลงในฐานข้อมูล สามารถกำหนดในขณะสร้าง Table (คำสั่ง create table) หรือ เพิ่มหรือเปลี่ยนแปลง (คำสั่ง alter table) มี 2 แบบ

1. Column Constraint คือ กำหนด Constraint เฉพาะ 1 Column มี Data Type, Not Null, Unique, Check, Primary Key, References

2. Table Constraint คือ กำหนด Constraint มากกว่า 1 Column มี Unique, Check, Primary Key, References

Constraint มีทั้งหมด 6 ชนิด

1. Data Type เช่น Column ที่เป็น Numeric จะไม่สามารถคีย์ข้อมูลที่เป็น String, Date,

Boolean

2. Not Null กำหนดให้ Column นั้น ต้องมีค่า เป็น Null ไม่ได้

3. Unique กำหนดให้ Column นั้น มีค่าที่ไม่ใช่ Null ไม่ซ้ำกัน สามารถมี Null ได้

มากกว่า 1 Row

4. Check กำหนดให้ Column นั้น มีค่าที่เฉพาะค่าที่กำหนดไว้เท่านั้น

5. Primary Key กำหนดให้ Column นั้น Not Null and Unique ใน 1 Table มี Primary Key ได้ 1 Primary Key เท่านั้น

6. References (Foreign Key) กำหนดให้ Column นั้น มีค่า เป็น Null หรือ Not Null เมื่อเป็น Not Null แล้วค่าที่เก็บใน Column ต้องเป็นค่าที่อยู่ใน Column ที่เป็น Primary Key ของ Table ที่อ้างอิง

### บทที่ 3

## การออกแบบโปรแกรม

#### 3.1. การออกแบบฐานข้อมูล

เมื่อได้ทำการศึกษาหลักการและทฤษฎีที่เกี่ยวข้องกับระบบจัดการการให้บริการเกี่ยวกับบริษัทขายเครื่องปรับอากาศเรียบร้อยแล้ว ขั้นตอนต่อไปก็จะเป็นการบริหารงานโดยวางแผนวิธีการดำเนินงานไว้เป็นขั้นตอน ซึ่งจะทำให้การทำงานมีระบบระเบียบมากขึ้น การดำเนินงานต่างๆ สามารถทำเป็นขั้นตอนได้ดังนี้

1. ศึกษาการทำงานของระบบจัดการการให้บริการเกี่ยวกับบริษัทขายเครื่องปรับอากาศ
2. ออกแบบฐานข้อมูลของระบบจัดการการให้บริการเกี่ยวกับบริษัทขายเครื่องปรับอากาศ
3. ออกแบบโปรแกรมสำหรับผู้ใช้งาน
4. เขียนโปรแกรมจัดการระบบจัดการการให้บริการเกี่ยวกับบริษัทขายเครื่องปรับอากาศ

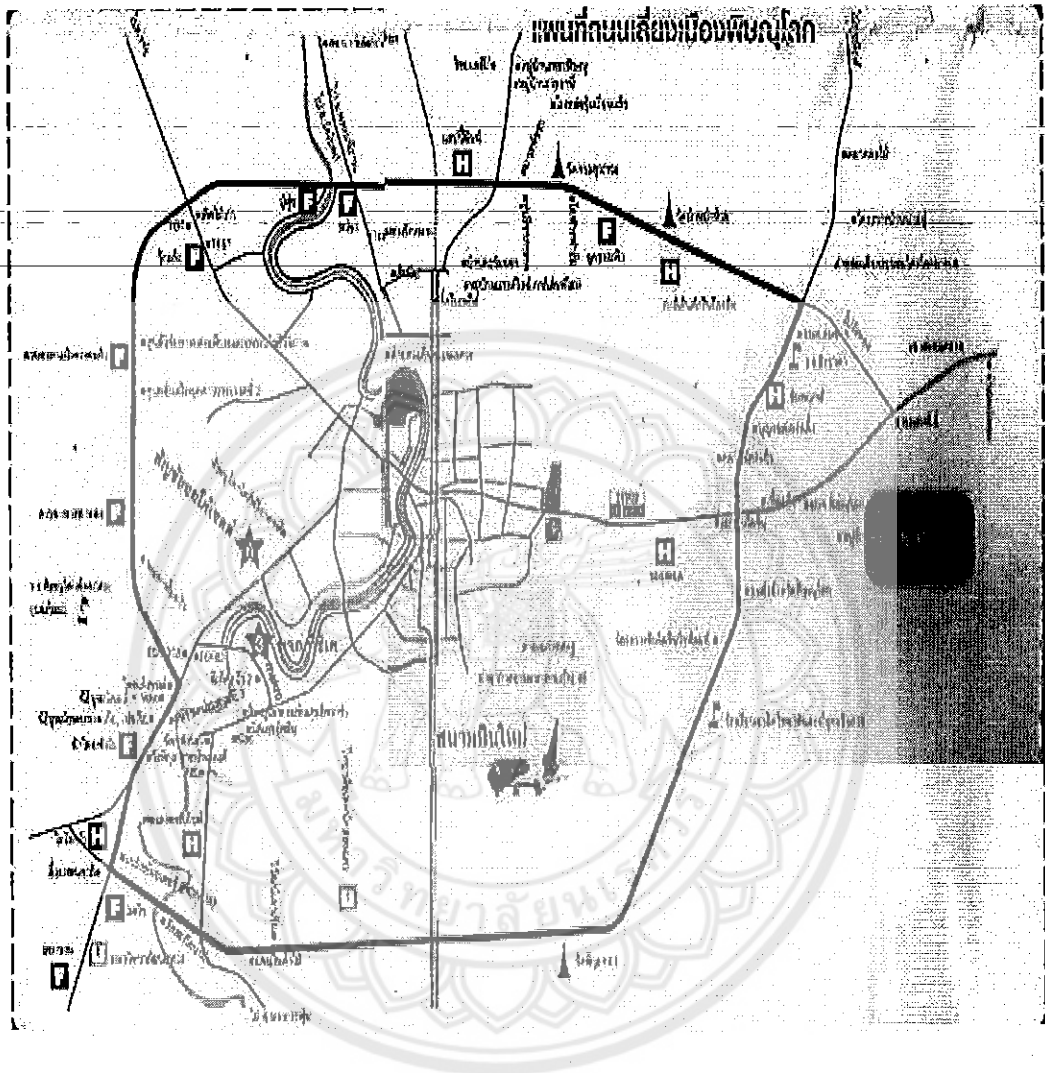
##### 3.1.1 ศึกษาการทำงานของระบบจัดการการให้บริการเกี่ยวกับบริษัทขายเครื่องปรับอากาศ

เริ่มด้วยการศึกษาการทำงานของระบบจัดการการให้บริการเกี่ยวกับบริษัทขายเครื่องปรับอากาศเดิมของบริษัท เพื่อทำการเก็บข้อมูล ทำให้ทราบถึงปัญหาที่เกิดขึ้นของบริษัทกรณีตัวอย่าง เนื่องจากการให้บริการเกี่ยวกับการให้บริการเครื่องปรับอากาศนั้น มีลูกค้าเป็นจำนวนมากที่มาใช้บริการ โดยจะแบ่งการให้บริการออกเป็นสองส่วนใหญ่ๆคือ การให้บริการบริเวณในตัวเมืองพิษณุโลก และรอบๆนอกตัวเมืองพิษณุโลก ซึ่งจะต้องมีกำหนดระยะเวลาในการให้บริการแก่ลูกค้าที่แน่นอนในแต่ละครั้งที่จะไปให้บริการ อีกทั้งยังต้องมีการตรวจสอบว่า ลูกค้าที่เราให้บริการนั้นยังอยู่ในระยะเวลาประกันหรือไม่ โดยเจ้าหน้าที่หรือพนักงานเจ้าของร้านจะเป็นผู้ควบคุมการและจัดสรรงานให้แก่ลูกจ้างนายช่างไปทำการให้บริการแก่ลูกค้าที่ร้องขอมา ซึ่งทำให้เกิดความล่าช้าในการให้บริการแก่ลูกค้า และนอกจากนี้เจ้าหน้าที่หรือพนักงานเจ้าของร้านยังได้มีการแบ่งการให้บริการแก่ลูกค้าออกเป็น 2 ประเภท คือ การให้บริการบริเวณในตัวเมืองพิษณุโลก และบริเวณรอบๆนอกตัวเมืองพิษณุโลก

จากนั้นได้ทำการศึกษากระบวนการจัดการการให้บริการเกี่ยวกับบริษัทขายเครื่องปรับอากาศ เพื่อใช้เป็นข้อมูลในการออกแบบโปรแกรม ซึ่งเปรียบเสมือนการให้บริการแก่ลูกค้าที่มาร้องขอการให้บริการ ให้มีความรวดเร็วขึ้นและมีลำดับคิวการให้บริการที่แน่นอนด้วย และยังทำให้บริษัทมีการจัดเก็บข้อมูลลูกค้าที่เคยมาใช้บริการอีกด้วยเพื่อเป็นการเก็บข้อมูลผู้ที่เคยใช้บริการของบริษัท ครั้งต่อไปบริษัทก็สามารถมีข้อมูลของลูกค้าคนนั้นๆเพื่อช่วยต่อการตัดสินใจในการให้บริการในครั้งต่อไป



แผนที่ในตัวเมืองพิษณุโลกก่อนการกำหนดจุดโหนด



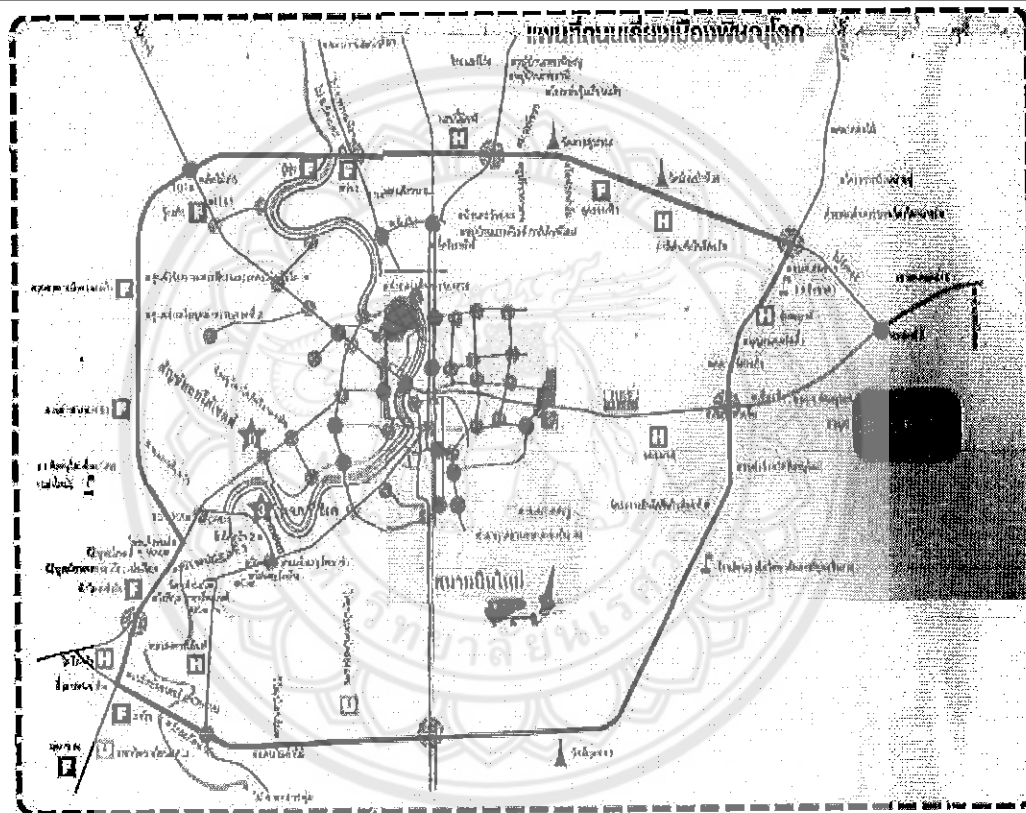
รูปที่ 3.2 แผนที่ในตัวเมืองพิษณุโลกก่อนการกำหนดจุดโหนด



**หมายเหตุ** จุดสีแดงในแผนที่จะแสดงถึงแต่ละโหนดที่จะใช้ในการหาแต่ละเส้นทางที่จะไปถึงแต่ละตำแหน่งนั้นๆ

#### การกำหนดจุดโหนดในตัวเมืองพิษณุโลก

ส่วนการแบ่งโซนภายในจังหวัดพิษณุโลกเราจะแบ่งโดยอ้างอิงจากแต่ละอำเภอเป็นหนึ่งโซน แต่ภายในส่วนของอำเภอเมืองเราจำเป็นต้องแบ่งให้ละเอียดมากขึ้น เพราะในพื้นที่ตัวเมืองมีผู้ใช้บริการมากกว่าพื้นที่อื่นๆ เราจะแบ่ง โหนดในพื้นที่ในตัวเมืองได้ดังนี้



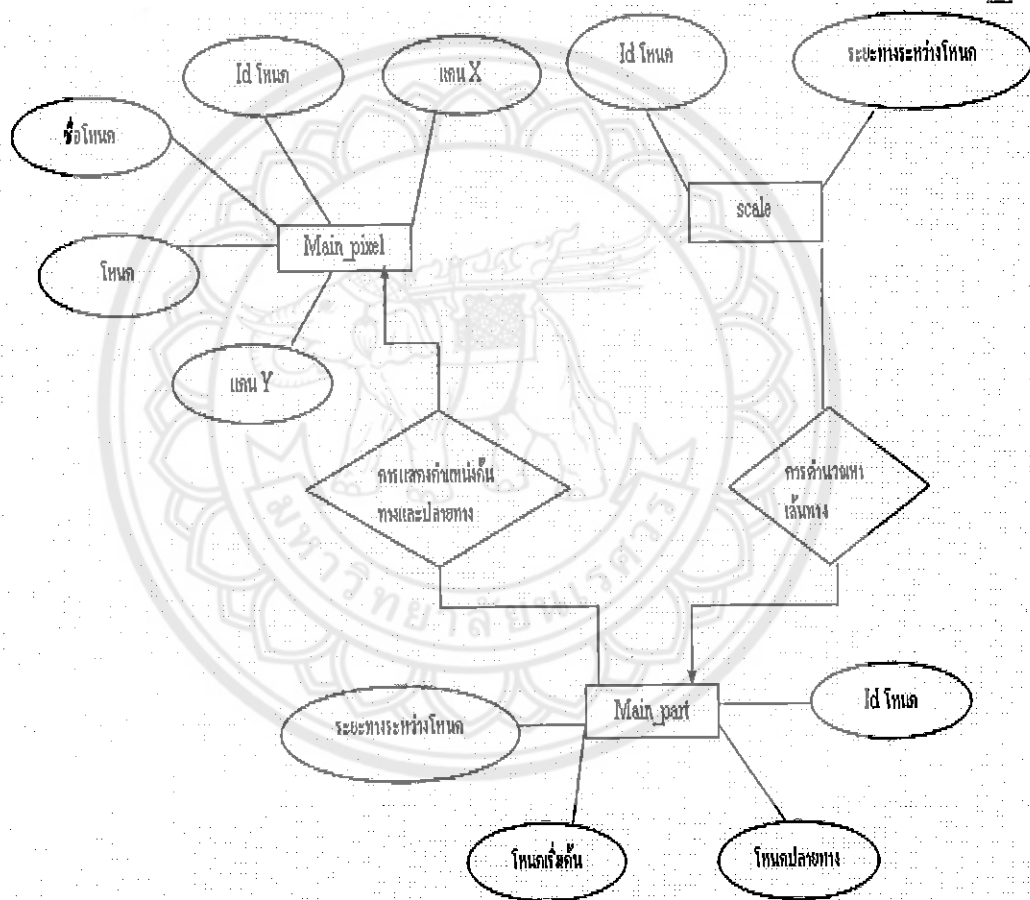
รูปที่ 3.4 การกำหนดจุดโหนดในตัวเมืองพิษณุโลก

และเราก็กำหนดแต่ละโหนดโดยจะใช้ชื่อของตำบลที่มีโหนดนั้นอยู่ ถ้าเป็นในเมืองก็จะใช้สถานที่ที่อยู่ใกล้เคียงเพื่อจะกำหนดเป็นชื่อของโหนดนั้นๆ โดยเราจะได้โหนดที่ต้องมาประมาณ 117 โหนด

และเราก็นำข้อมูลของโหนดต่างๆ มาเป็นฐานข้อมูล เราจะเขียนโดยใช้ My SQL ในการเขียนฐานข้อมูลของโปรแกรมนี้

3.1.2 ออกแบบฐานข้อมูลของระบบจัดการการให้บริการเกี่ยวกับบริษัทขายเครื่องปรับอากาศ MySQL เป็นฐานข้อมูลเชิงสัมพันธ์ (Relational Database Management System) RDBMS ก็ สามารถทำงานกับตารางข้อมูลหลายตารางพร้อมๆ กัน โดยสามารถแสดงความสัมพันธ์ของตาราง เหล่านั้นด้วย field ที่ใช้ร่วมกัน

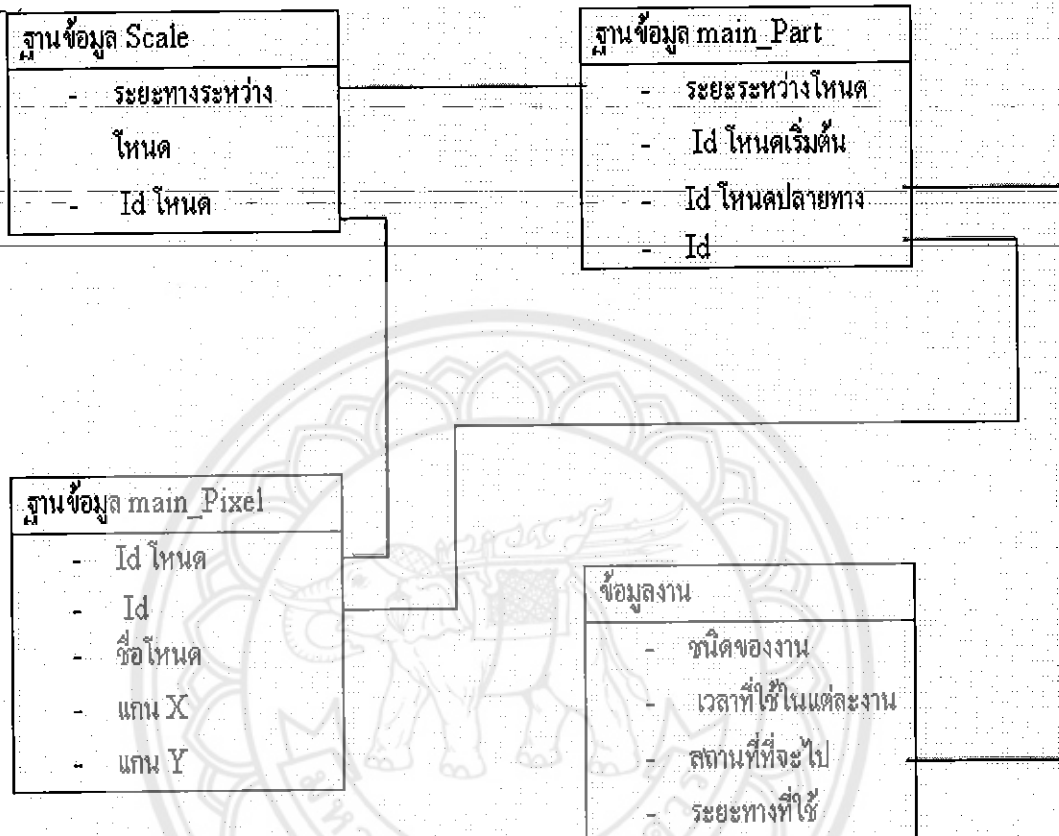
### E/R-Diagram



รูปที่ 3.5 แผนภาพ ER- Diagram



## Schema



รูปที่ 3.6 แผนภาพ Schema

โดยเราจะสามารถทำตารางโดยใช้โปรแกรม My SQL จะได้ตารางของฐานข้อมูลดังนี้  
ตัวอย่างตารางแสดงตำแหน่งของแต่ละโหนดในฐานข้อมูล

id	NodeName	Post_X	Post_Y	Node
1	สามแยกป่อภาค	7.1810	.6530	A01
2	นลภาค	6.2490	2.1600	A02
3	แยกท่าสะแก	5.2360	3.2670	A03
4	ป่าแดง	4.7430	3.3870	A04
5	แยกป่าแดง	4.6100	3.1470	A05
6	บ้านคอง	4.6230	2.9600	A06
7	คันช้าง	3.4910	3.5870	A07
8	วัดโบสถ์	3.4640	5.1600	A08
9	ทับยายเชียง	3.3910	4.4130	A09
10	ตงประดำ	3.0380	3.8270	A10
11	มะตอง	2.4380	4.4800	B11
12	พรหมพิราม	2.6250	4.9200	B12
13	ท่าช้าง	2.7310	5.3870	B13
14	จอมทอง	2.7450	5.8800	B14
15	ไผ่ชวลอน	2.5710	5.7470	B15
16	วังม่วง	2.1320	4.3780	B16
17	วังวน	1.7450	4.7870	B17
18	บ้านพร้าว	6.3420	4.2400	B18
19	นาบัว	6.9550	3.8270	B19
20	ปอโพธิ์	7.6480	3.9330	B20
21	นครไทย	6.7280	4.4000	C21
22	ทหนองกระพิว	6.6220	4.8520	C22
23	บ้านแยง	6.5020	5.8270	C23
24	แก่งไสภา	5.4760	5.8000	C24
25	บ้านกลาง	5.1830	5.3200	C25

ตารางที่ 3.1 ตารางแสดง Pixel

ตัวอย่างการแสดงระยะทางของแต่ละโหนดที่สามารถเชื่อมต่อกัน

node	A01	A02	A03	A04	A05	A06	A07	A08	A09	A10	B11	B12	B13	B14	B15
A01	0.0000	31.3500	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
A02	31.3500	0.0000	26.6800	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
A03	0.0000	26.6800	0.0000	6.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
A04	0.0000	0.0000	6.0000	0.0000	6.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
A05	0.0000	0.0000	0.0000	6.0000	0.0000	2.6600	26.6800	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
A06	0.0000	0.0000	0.0000	0.0000	2.6600	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
A07	0.0000	0.0000	0.0000	0.0000	26.6800	0.0000	0.0000	30.6800	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
A08	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	30.6800	0.0000	14.6700	0.0000	0.0000	0.0000	14.6700	0.0000	0.0000
A09	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	14.6700	0.0000	10.0000	0.0000	14.0000	0.0000	0.0000	0.0000
A10	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	10.0000	0.0000	15.3400	0.0000	0.0000	0.0000	0.0000
B11	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	15.3400	0.0000	9.3300	0.0000	0.0000	0.0000
B12	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	14.0000	0.0000	9.3300	0.0000	9.3300	0.0000	0.0000
B13	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	14.6700	0.0000	0.0000	0.0000	9.3300	0.0000	0.0000	0.0000
B14	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	7.3300
B15	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	7.3300	0.0000

ตารางที่ 3.2 ตารางแสดง scale

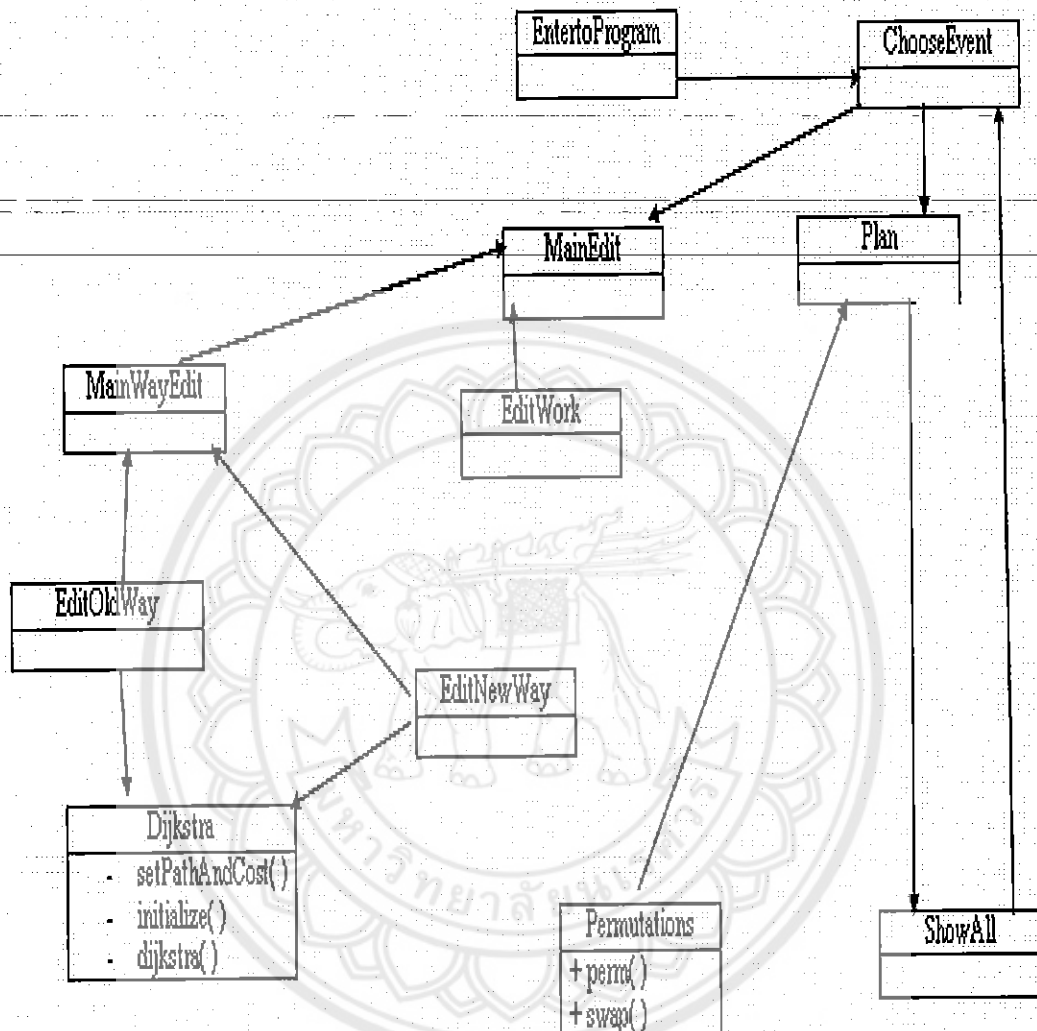
โดยเราจะนำฐานข้อมูลเหล่านี้ไปใช้กับการคำนวณในโปรแกรมที่ต้องการหาระยะทางในการไปแต่ละที่ที่ต้องการไป

เมื่อเราได้ฐานข้อมูลที่จะนำไปเชื่อมต่อกับโปรแกรมของเราแล้วเราก็จะนำแผนที่ที่แบ่งโหนดแล้วมาทำการแบ่งพื้นที่โดยจะแบ่งพื้นที่เป็นแต่ละตำบลเพื่อจะได้เห็นแต่ละตำบลในจังหวัดพิษณุโลกได้เด่นชัดขึ้น

### 3.1.3 ออกแบบโปรแกรมสำหรับผู้ใช้งาน

ในส่วนของการออกแบบโปรแกรมสำหรับผู้ใช้งานนั้น เราจะออกแบบโปรแกรมโดยแนวความคิดของโปรแกรม จากทฤษฎีในบทที่ 2 สิ่งที่จะต้องใช้ในการสร้างโปรแกรมคือ อัลกอริทึมในการหา Short-test Path โดยเลือกใช้ Dijkstra's algorithm และ วิธีการเรียงสับเปลี่ยน (Permutation) หลักการและขั้นตอนการคิดเพื่อให้ได้ผลลัพธ์ ออกมาแสดง ซึ่งจะพูดถึงรายละเอียดในหัวข้อ 3. แนวความคิดของโปรแกรมด้านล่างต่อไป

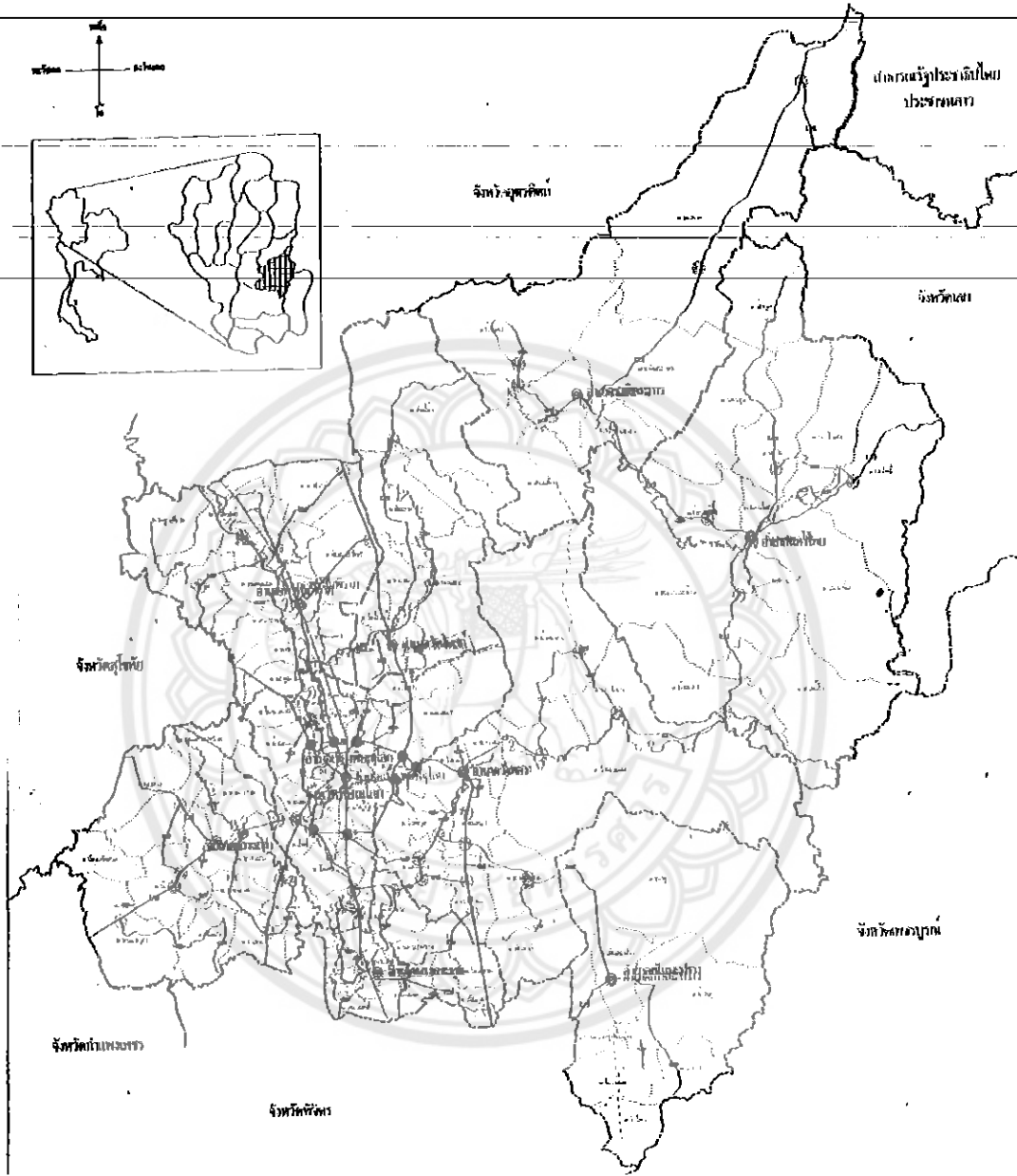
### Class Diagram



รูปที่ 3-7 Class Diagram

ส่วนหน้าตาของแผนที่ ที่จะแสดงบนตัวโปรแกรมเพื่อประกอบการหาเส้นทางการเดินทางที่สั้นที่สุด โดยจะได้เป็นแผนที่ที่ทำเรียบร้อยแล้วเป็นดังนี้

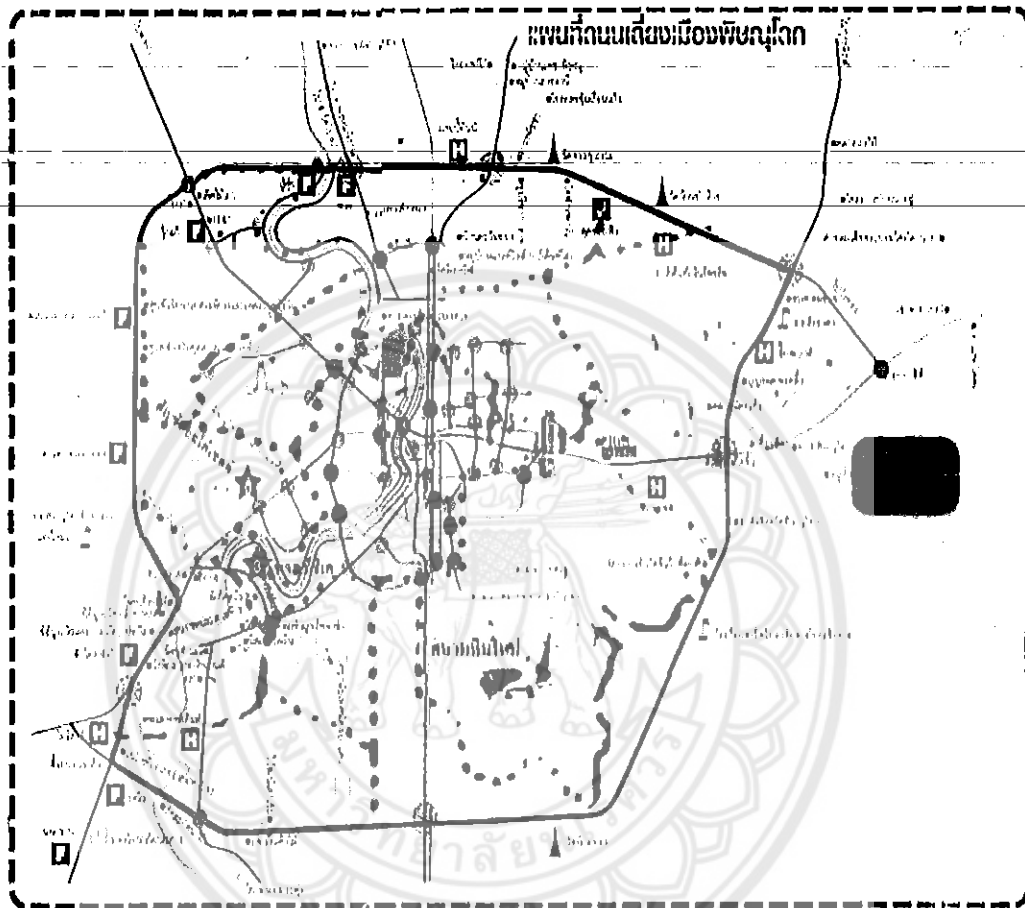
### แผนที่ประกอบในการแสดงการหาเส้นทางบริเวณรอบนอกของโปรแกรมนี้



รูปที่ 3.8 แผนที่ประกอบในการแสดงการหาเส้นทางบริเวณรอบนอก

ส่วนของในตัวเมืองก็จะได้แผนที่เป็นดังนี้

แผนที่ประกอบในการแสดงการหาเส้นทางในตัวเมืองจังหวัดพิษณุโลกของโปรแกรมนี้



รูปที่ 3.9 แผนที่ประกอบในการแสดงการหาเส้นทางในตัวเมืองจังหวัดพิษณุโลก

และเราจะนำแผนที่ที่ทำการแบ่งสี่แยกเป็นตำบลนี้มาแสดงในส่วนของ โปรแกรมด้วย

3.1.4 เขียนโปรแกรมจัดการระบบจัดการการให้บริการเกี่ยวกับบริษัทขายเครื่องปรับอากาศ  
การเขียนโปรแกรมนี้นั้นจะเขียนด้วยภาษาจาวา และจะพุดถึงแนวคิดรายละเอียดในหัวข้อ

ถัดไป

### 3.2 ทำไมต้องเลือกใช้ภาษาจาวาในการเขียนโปรแกรมนี้

เราจะใช้ภาษา JAVA ในการเขียนโปรแกรม เพราะภาษา JAVA นั้นมีลักษณะเด่นอยู่หลายข้อที่เป็นเหตุผลสนับสนุนให้โปรแกรมเมอร์และนักพัฒนาโปรแกรมทั่วโลกนิยมเลือกนำมาสร้างเป็นผลงานของตน ซึ่งได้แก่

#### 1.ทำงานอิสระ (Platform Independent)

การความโหดโปรแกรมมาใช้งานบนเครื่องของผู้ใช้จะสามารถลดเวลาในการโหลดโปรแกรมมาจากเซิร์ฟเวอร์ทุกครั้งที่มีการสั่งการ หรือ Interactive ระหว่างผู้ใช้กับ โปรแกรมจาวา จึงเพิ่มความยืดหยุ่นให้กับโปรแกรมจาวาบนอินเทอร์เน็ตเป็นอย่างมาก

#### 2.ความง่ายของตัวภาษา

หลักไวยากรณ์ของจาวามีความคล้ายคลึงกับภาษา C และ C++ เป็นอย่างมาก ทำให้ผู้ไม่เคยเรียนเขียนโปรแกรมมาก่อนสามารถเรียนรู้ได้อย่างรวดเร็ว ส่วนผู้ที่เคยเขียนโปรแกรมก็ใช้เวลาศึกษาเพิ่มเติมน้อยมาก

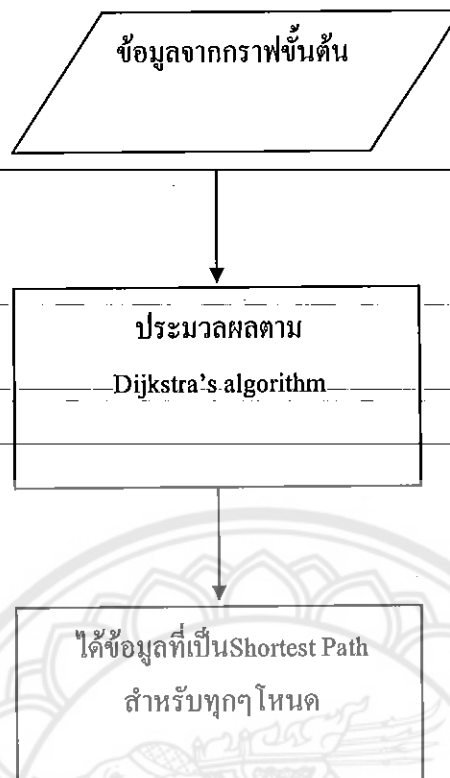
#### 3.ความปลอดภัย (Security)

การป้องกันเมื่ออยู่หลายระดับตั้งแต่ตัวภาษาจาวาเอง ที่ไม่อนุญาตให้ใช้งานพอยเตอร์ หรือการเข้าถึงฮาร์ดแวร์โดยตรง รวมไปถึงการป้องกันไม่ให้มีการดัดแปลง Java Byte Code ที่สร้างขึ้นมาจากขนาดเดียวกันก็ฝัง Security Manager Object เข้าไปในโปรแกรมที่เขียนด้วยจาวา ไปคอยควบคุมการทำงานของโปรแกรมไม่ให้ก่อความเสียหายให้กับระบบที่รันอีกด้วย

### 3.3. แนวความคิดของโปรแกรม

จากทฤษฎีในบทที่2 สิ่งที่จะต้องใช้ในการสร้างโปรแกรมคือ อัลกอริทึมในการหา Short test Path โดยเลือกใช้ Dijkstra's algorithm และ วิธีการเรียงสับเปลี่ยน (Permutation) หลักการและขั้นตอนการคิดเพื่อให้ได้ผลลัพธ์ ออกมาแสดงได้ดังนี้

3.3.1. ใช้ Dijkstra's algorithm ประมวลผลจากกราฟเพื่อให้ได้ข้อมูลที่เป็น Shortest Path สำหรับทุกๆ โหนดบนแผนที่



รูปที่ 3.10 แนวการคิดของโปรแกรม

ในขั้นตอนการประมวลผลตาม Dijkstra's algorithm ใน รูปที่ 3.10 แนวการคิดของโปรแกรมนั้น จะมีการแสดง code ประกอบเพื่อแสดงแนวคิดการทำงานขั้นตอนการประมวลผลตาม Dijkstra's algorithm ตามรูปด้านบน

หมายเหตุประมวลผลด้วยDijkstra

```

public void dijkstra(char x){
    initialize(s.size()); //initialize weight array
    weight[x-'A'] = 0;    //initial weight at vertex x
    pq.add(new Edge(x-'A', 0)); //start vertex

    //as long as there still vertices to process
  
```



```

while(!pq.isEmpty()) {
    Edge edge = (Edge)pq.first();
    int u = edge.getSource();
    //checking neighbors of u
    //update weights if necessary
    for(int v = 0; v < s.size(); v++) {
        Edge e = getEdge(u, v);
        if(e != null) {
            if(weight[v] > weight[u] + e.getCost()) {
                weight[v] = weight[u] + e.getCost();
                path[v]=path[u]+u+v;

                pq.add(new Edge(v, weight[v]));
            }
        }
    }
    pq.remove(edge); //already processed
}

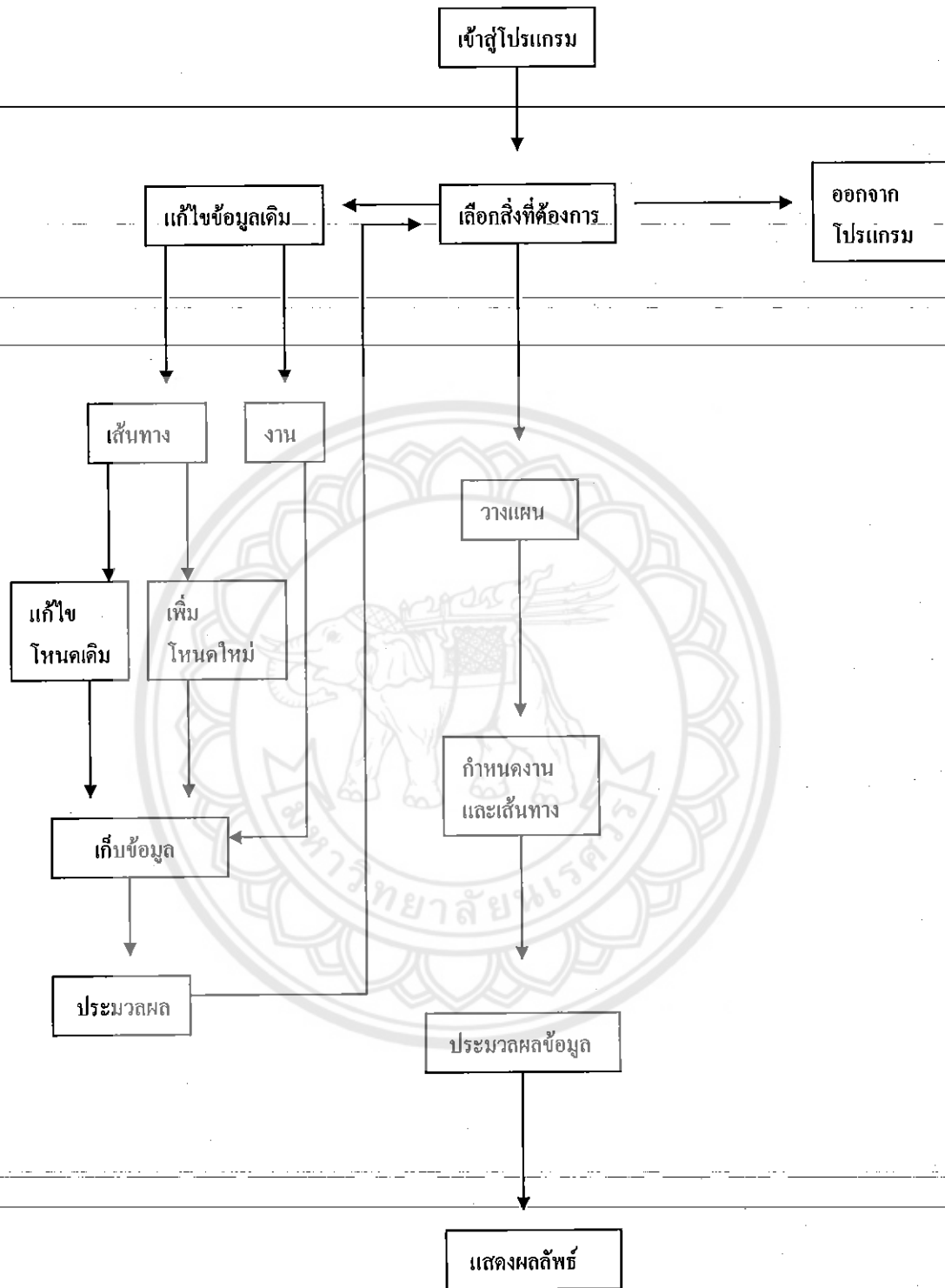
```

3.3.2. ใส่โหนดที่ต้องการเดินทางลงไปจากนั้นใช้หลักการของการเรียงสับเปลี่ยน (Permutation) สำหรับการสับเปลี่ยนของเหตุการณ์ที่เป็นไปได้ทั้งหมดจากนั้นเปรียบเทียบ โหนดที่ติดกันจากข้อมูล Shortest Path ของโหนดถูกเลือกแล้วนำระยะทางที่ได้จากข้อมูล Shortest Path มารวมกันแล้ว เปรียบเทียบกับทุกเหตุการณ์ว่า เหตุการณ์ใดมีระยะทางน้อยที่สุดก็เลือกเหตุการณ์นั้นและนำไปค้นหา Pixel ของแต่ละโหนดที่ถูกเก็บไว้ในฐานข้อมูลแล้วแสดงเส้นทางออกมาทางจอภาพ สามารถแสดงได้ ดังนี้

แนวความคิดของโปรแกรม

รูปที่ 3.11 แนวความคิดของโปรแกรม

### 3.4 ขั้นตอนการทำงานของโปรแกรม



รูปที่ 3.12 ขั้นตอนการทำงานของโปรแกรม

### 3.5 อธิบายลำดับขั้นตอนการทำงานของโปรแกรม

เข้าสู่โปรแกรมจากนั้น จะมีให้เลือกว่าต้องการทำอะไร โดยมีให้เลือก3 อย่างคือต้อง

#### 3.5.1. เข้าไปแก้ไขข้อมูล

เลือกการแก้ไขข้อมูล ก็จะเป็นการเข้าไปแก้ไขข้อมูลส่วนที่เกี่ยวข้องกับการวางแผนการเดินทางทั้งหมดซึ่งได้แก่ เส้นทางและงาน

##### 3.5.1.1 เส้นทาง แบ่งออกเป็นอีก2 แบบคือ แบบเดิม และแบบใหม่

- *แบบเดิม* จะเป็นการแก้ไขในกรณีที่ เส้นทางเดิมได้ถูกแก้ไขโดยธรรมชาติ เช่นมีการตัดถนนเส้นใหม่ หรือมีการวัดค่าระยะทางใหม่ที่มีค่าถูกต้องกว่าค่าตั้งต้น
- *แบบใหม่* จะเป็นการเพิ่มจุดใหม่พร้อมกับรายละเอียดเข้าไปโดยจุดใหม่นี้อาจจะได้มาจากเพื่อการพัฒนาโปรแกรมให้มีประสิทธิภาพมากขึ้น

3.5.1.2 งาน เป็นการแก้ไขรายละเอียดของงาน และเพิ่มชนิดของงานเข้าไปกรณีที่เปิดให้บริการงานชนิดใหม่ขึ้นมา

เมื่อทำการเปลี่ยนแปลงแก้ไขข้อมูลแล้ว โปรแกรมก็จะทำการเก็บข้อมูลไว้ แล้วจะทำการประมวลผลสำหรับเก็บค่า Short Test Path ไว้สำหรับใช้ในการวางแผนการเดินทางต่อไป

#### 3.5.2. การวางแผนการเดินทางการ

เมื่อเลือกวางแผนการเดินทาง ก็จะเป็นการเข้าไปวางแผนการเดินทางว่าต้องการไปที่จุดไหนบ้าง พร้อมกับระบุงานสำหรับแต่ละจุดที่จะต้องไป เมื่อวางแผนเรียบร้อยแล้ว โปรแกรมก็จะทำการประมวลผลข้อมูลที่ใส่เข้าไปโดยพิจารณาหลักของ Short Test Path กับ Traveling salesman problem จากนั้นก็จะแสดงรูปภาพของแผนที่รวมทั้งเส้นทาง และรายละเอียดของงานออกมา

#### 3.5.3. ออกจากโปรแกรม

เป็นการออกจากโปรแกรมเมื่อไม่ต้องการใช้งาน

## บทที่ 4

### ขั้นตอนการดำเนินงาน

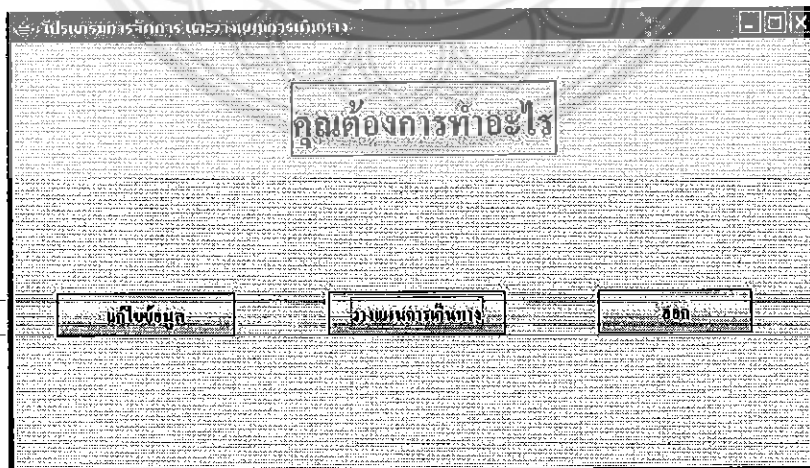
#### 4. วิธีการใช้งานโปรแกรม

เมื่อเปิดโปรแกรมแล้วจะพบหน้าต่างดังรูป จากนั้นให้คลิกปุ่มเข้าสู่โปรแกรมเพื่อเริ่มใช้งานโปรแกรม



รูปที่ 4.1 หน้าตาGUIก่อนเข้าสู่โปรแกรม

จากนั้นจะพบหน้าต่างดังรูปข้างล่างนี้เพื่อสำหรับเลือกการใช้งานซึ่งได้แก่ แก้ไขข้อมูล, วางแผนการเดินทางและออกจากโปรแกรม



รูปที่ 4.2 หน้าตาGUIของโปรแกรมการจัดการและวางแผนการเดินทาง

ในตอนต่อไปจะขออธิบายการเลือกใช้งานตามกรณีดังนี้

#### 4.1 เลือกแก้ไขข้อมูล

การแก้ไขข้อมูลจะเป็นการเปลี่ยนแปลงค่าของสิ่งต่างๆภายในฐานข้อมูล โดยการเปลี่ยนแปลงนั้นก็จะมีผลต่อการประมวลผลด้วย ซึ่งการแก้ไขข้อมูลสามารถแก้ไขข้อมูลได้ดังต่อไปนี้คือ

##### 4.1.1 การแก้ไขเส้นทาง

การแก้ไขเส้นทางนั้นจะเป็นการกำหนดรายละเอียดในส่วนของการจัดการกับโหนดต่างๆที่อยู่ในแผนที่ในเรื่องของการเชื่อมต่อระหว่างโหนด และการกำหนดพิกัด แบ่งการแก้ไขเส้นทางเป็น2แบบดังนี้

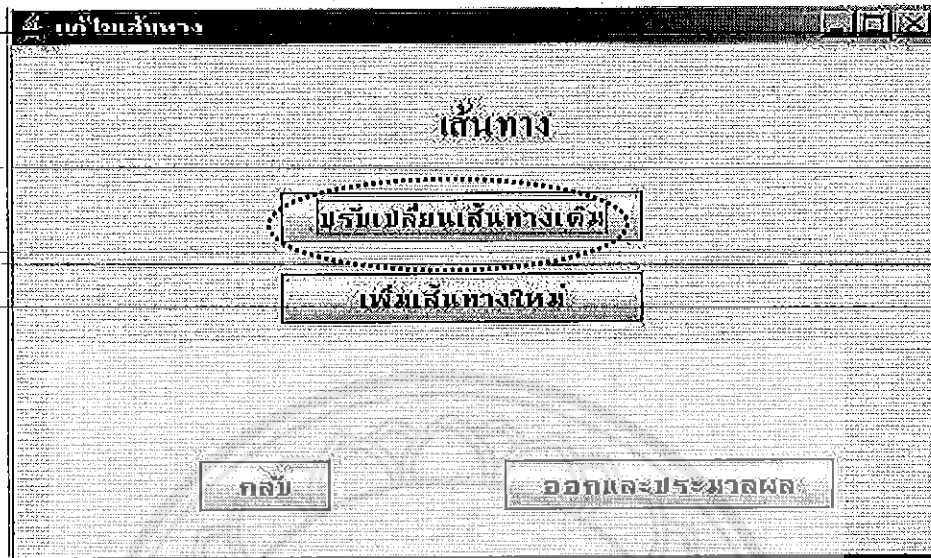
4.1.1.1 การปรับเปลี่ยนเส้นทางเดิม เป็นการแก้ไขในส่วนของรายละเอียดของโหนดที่มีอยู่แล้วให้สามารถลบและเพิ่มการเชื่อมต่อระหว่างโหนด และพิกัดบนแผนที่ที่สามารถทำได้ดังนี้

1.กดปุ่ม"แก้ไขข้อมูล" เมื่อเริ่มเข้าสู่โปรแกรมจากนั้นจะปรากฏหน้าต่าง ให้เลือกแก้ไขข้อมูล จากนั้นให้กดปุ่ม"เส้นทาง"ดังรูป



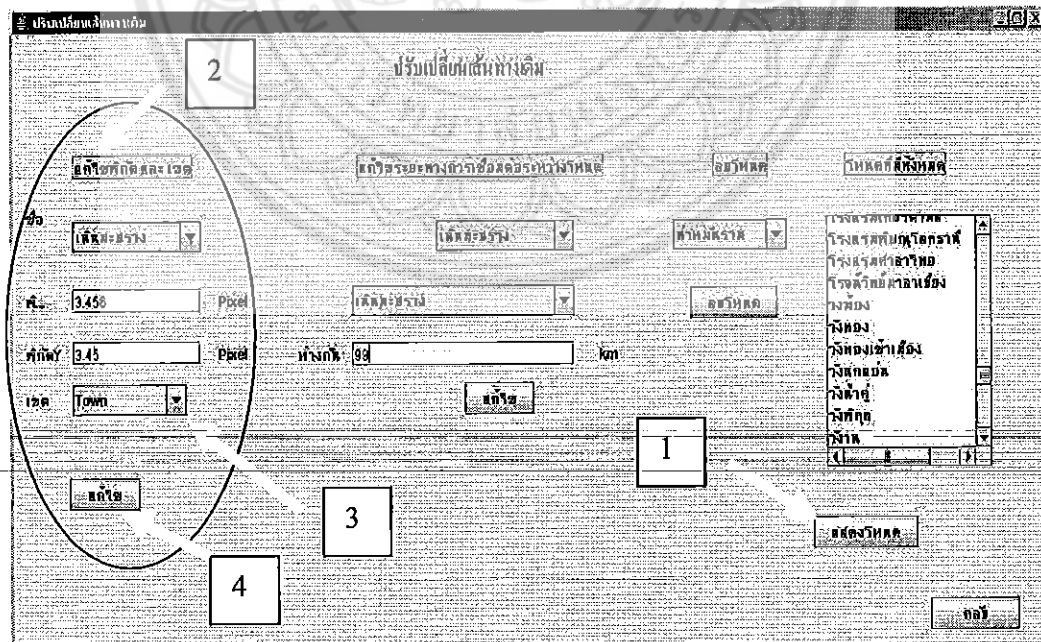
รูปที่ 4.3 หน้าตาGUIของการเลือกสิ่งที่ต้องการแก้ไข

2. จากนั้นจะปรากฏหน้าต่างให้เลือกแก้ไขเส้นทาง ให้กดปุ่ม"ปรับเปลี่ยนเส้นทางเดิม" ดั่งรูป



รูปที่ 4.4 หน้าตาGUIของแก้ไขเส้นทาง

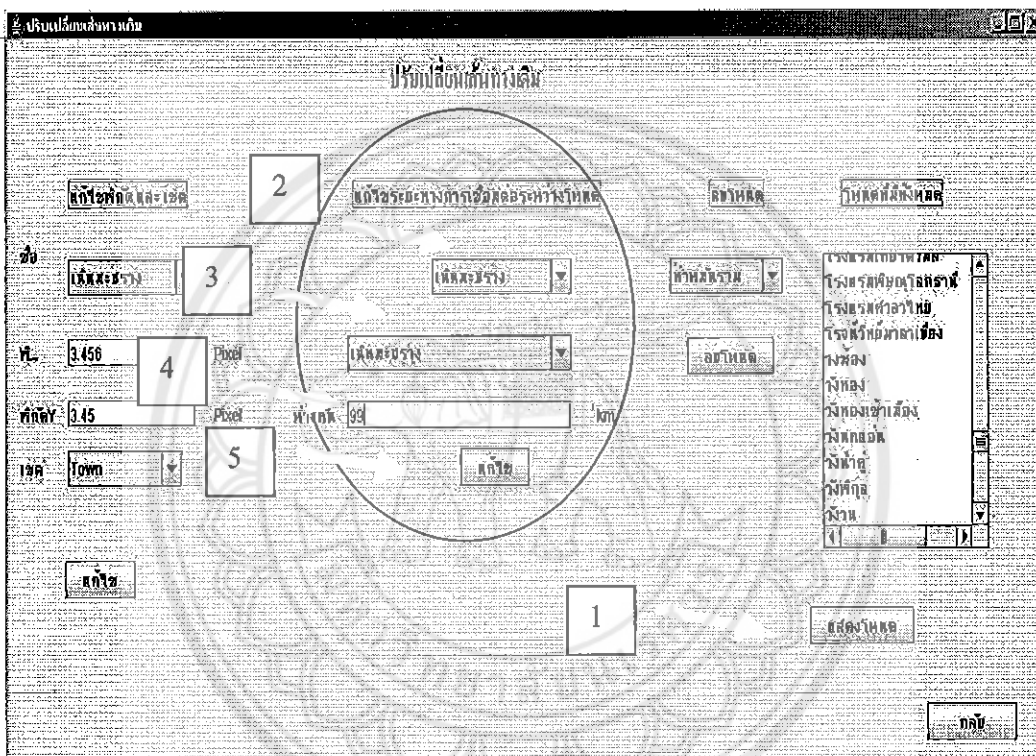
3. จากนั้นจะปรากฏหน้าต่างสำหรับการปรับเปลี่ยนโหนดเดิม ดังรูป



รูปที่ 4.5 หน้าตาGUIของการปรับเปลี่ยนโหนดเดิม

การปรับเปลี่ยนโหนดเดิมแบ่งออกเป็น 3 ส่วนคือ

- การแก้ไขพิกัด และ เขต (สำหรับการใช้วาดรูปลากเส้นทางทำได้ดังนี้
  1. ให้กดปุ่ม"แสดงโหนด"
  2. มาเลือกโหนดที่ต้องการแก้ไขจากCombo box
  3. ใส่พิกัด X และ Y ลงในช่องที่ระบุ และเลือกเขตที่อยู่ของโหนดมี 2 เขตคือ Outlay(นอกเมือง) กับ Town(ในเมือง)
  4. กดปุ่ม"แก้ไข" เพื่อยืนยันการแก้ไข

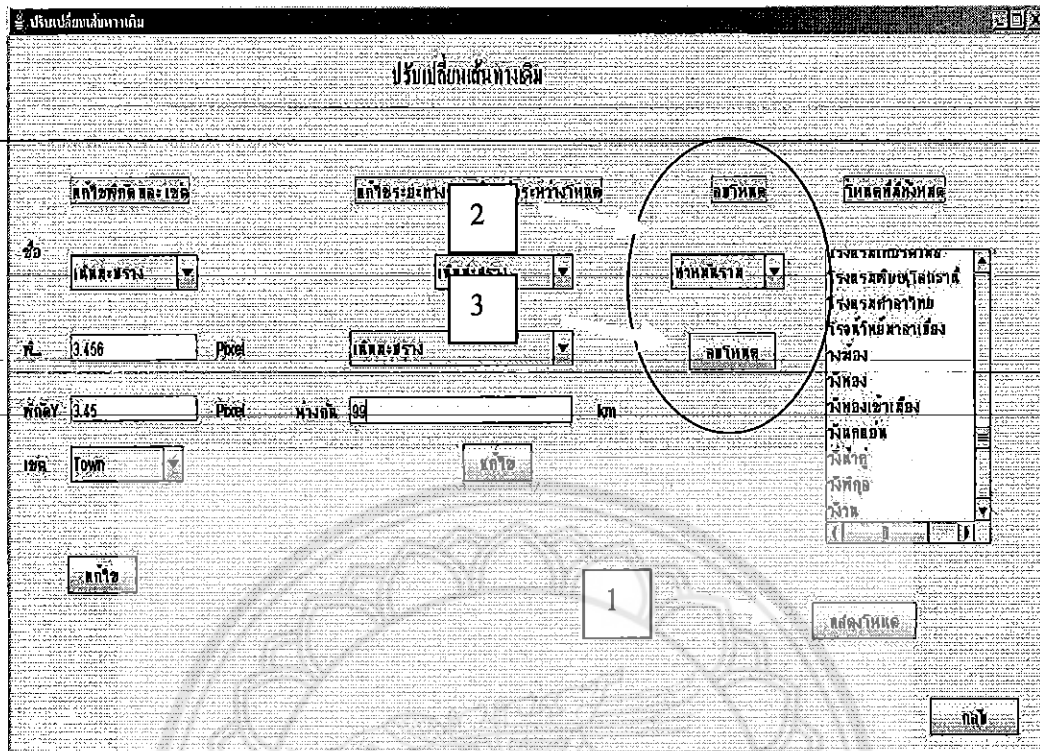


รูปที่ 4.6 หน้าตาGUIของการเลือกสิ่งที่ต้องการแก้ไข

- การแก้ไขระยะทางที่เชื่อมต่อกันระหว่าง 2 โหนด

1. กดปุ่ม"แสดงโหนด"
2. เลือกจุดเริ่มต้น
3. เลือกจุดปลายทาง
4. ใส่ค่าระยะทางใหม่ที่ต้องการเปลี่ยน
5. กดปุ่ม"แก้ไข" เพื่อยืนยันการแก้ไข



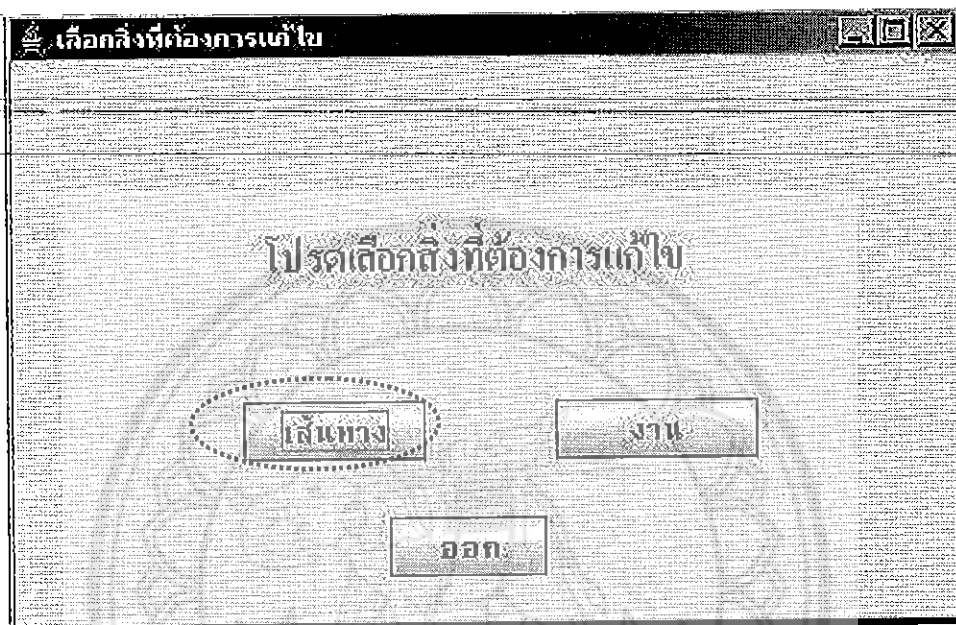


รูปที่ 4.7 หน้าตาGUIของการแก้ไขเส้นทาง

- การลบ โหนดที่มีอยู่แล้ว
  1. กดปุ่ม"แสดง โหนด"
  2. เลือกจุดที่ต้องการลบ
  3. กดปุ่ม"ลบ โหนด" เพื่อยืนยันการลบ
- เมื่อแก้ไขข้อมูลเสร็จแล้วให้กดปุ่ม"กลับ"

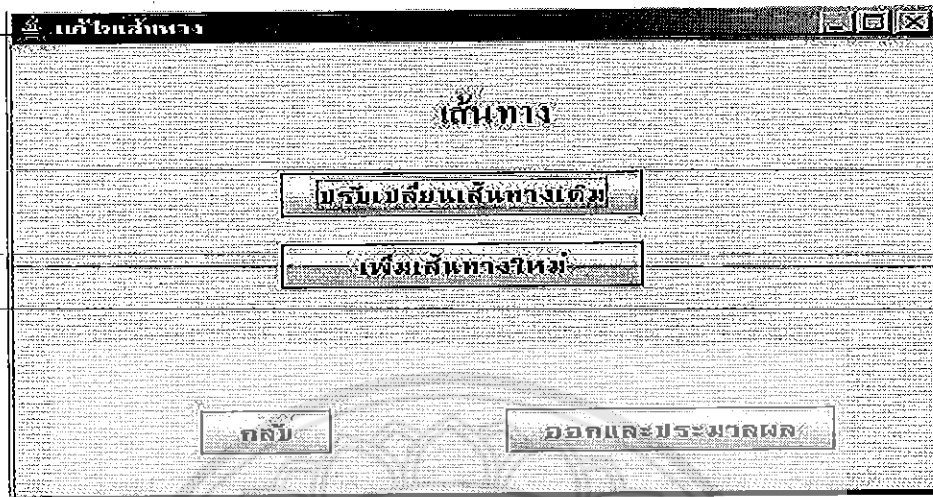
4.1.1.2 การเพิ่มเส้นทางใหม่ เป็นการเพิ่มโหนดใหม่ให้กับฐานข้อมูลและตั้งค่ารายละเอียดของโหนดใหม่ทั้งหมดทั้งในเรื่องของชื่อ โหนด, รหัส, การติดต่อกับโหนดอื่นในแผนที่, พิกัด สามารถทำได้ดังนี้

1. กดปุ่ม"แก้ไขข้อมูล" เมื่อเริ่มเข้าสู่โปรแกรมจากนั้นจะปรากฏหน้าต่าง ให้เลือกแก้ไขข้อมูล จากนั้นให้กดปุ่ม"เส้นทาง"ดังรูป



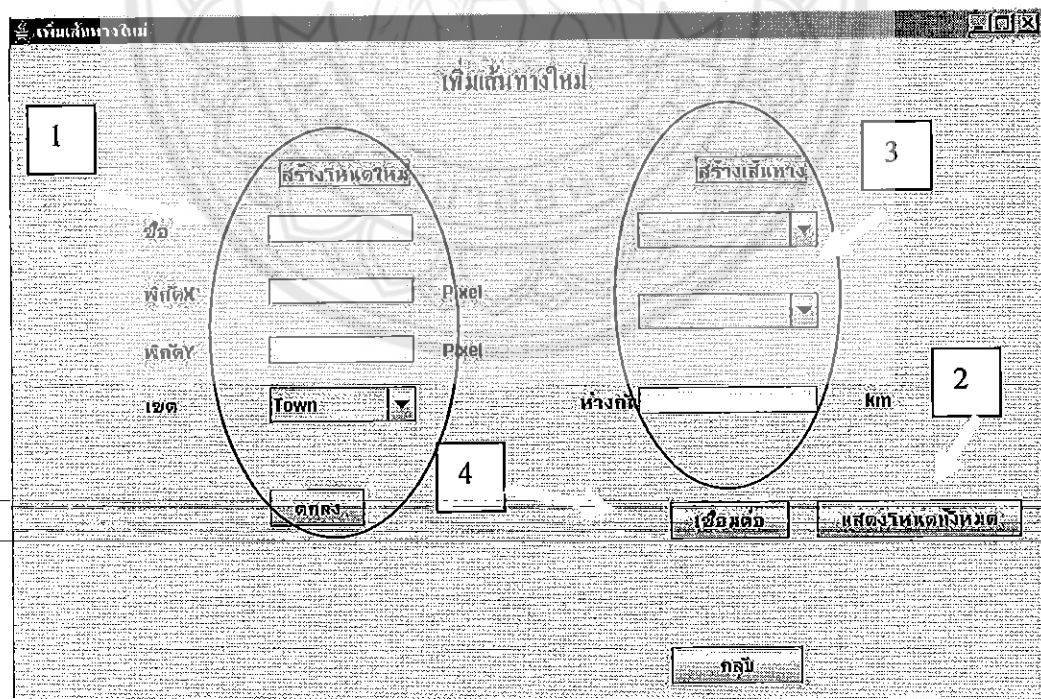
รูปที่ 4.8 หน้าตาGUIของการเพิ่มโหนดใหม่

2. จากนั้นจะปรากฏหน้าต่างให้เลือกแก้ไขเส้นทาง ให้กดปุ่ม  
"เพิ่มเส้นทางใหม่" ดังรูป



รูปที่ 4.9 หน้าตาGUIของการแก้ไขข้อมูลเส้นทาง

3. จากนั้นจะปรากฏหน้าต่างสำหรับเพิ่มโหนดใหม่ดังรูป

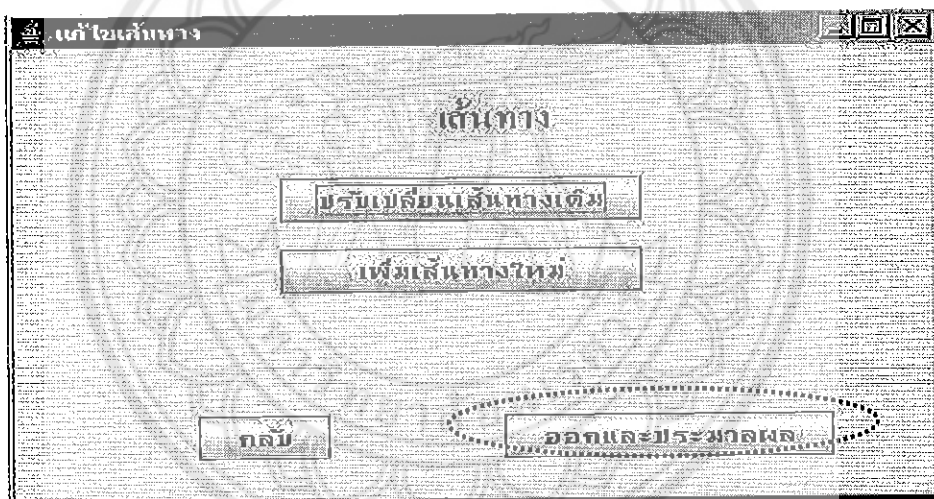


รูปที่ 4.10 หน้าตาGUIของการเลือกสิ่งที่ต้องการแก้ไข

การเพิ่มโหนดใหม่ทำได้ดังนี้

- ทำการตั้งชื่อ โหนดใหม่โดยให้ใส่ชื่อ .พิกัด XและY เขตแล้วกดปุ่ม"ตกลง"เพื่อทำการสร้างลงฐานข้อมูล
- กดปุ่ม"แสดงโหนดทั้งหมด"
- ทำการเชื่อมต่อ โหนดที่สร้างใหม่โดยเลือกจากCombo box ว่าต้องการเชื่อมต่อกับโหนดไหนบ้าง โดยให้เลือกเป็นทั้งต้นทางและปลายทาง(ด้านบนต้นทาง ด้านล่างปลายทาง ) จากนั้นให้ใส่ระยะทางที่ห่างกัน แล้วกดปุ่ม"เชื่อมต่อ"
- เมื่อเพิ่มโหนดจนครบแล้วให้กดปุ่ม"กลับ"

เมื่อแก้ไขข้อมูลเส้นทางเรียบร้อยแล้ว แล้วกดปุ่ม "กลับ" ออกมาไม่ว่าจะแก้ไขเส้นทางแบบใดก็ตามจะออกมาที่หน้าข่าล่างอีกครั้ง จากนั้นให้ทำการกดปุ่มออกและประมวลผลเพื่อทำการUpdate ฐานข้อมูล ในส่วนของShortest path ทุกครั้งเพื่อประสิทธิภาพในการประมวลผลหาระยะทางต่อไป

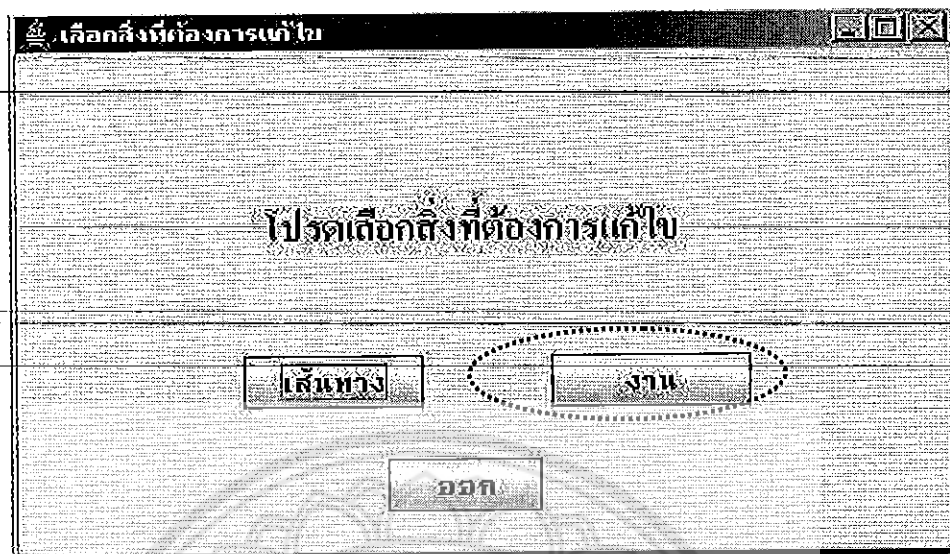


รูปที่ 4.11 หน้าตาGUIของการออกและประมวลผล

#### 4.1.2 แก้ไขงาน

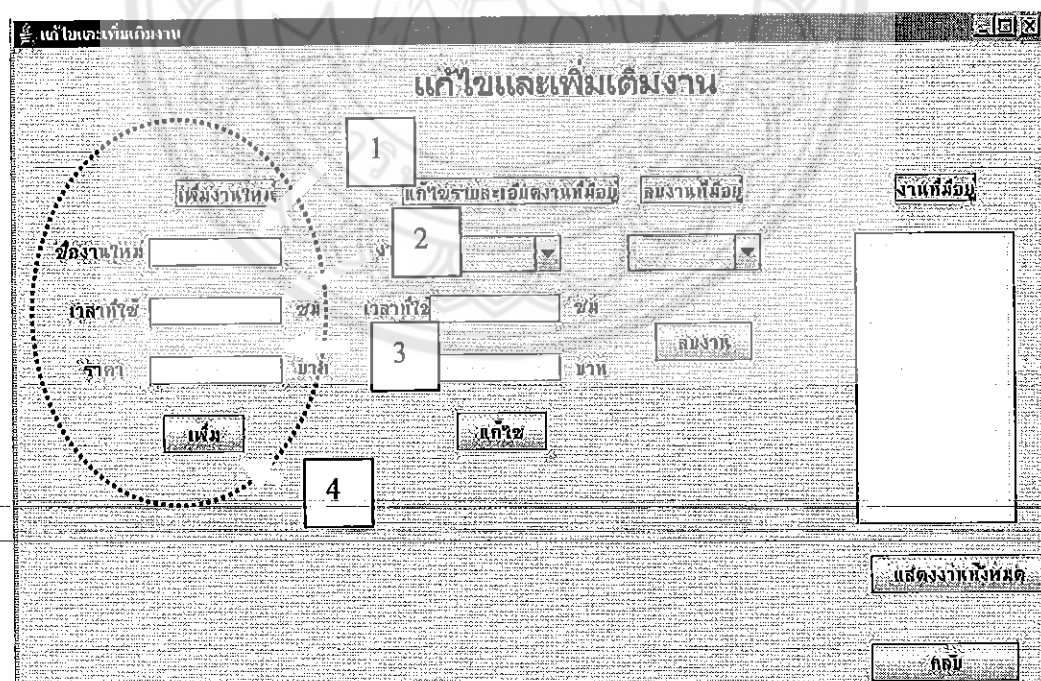
การแก้ไขงานนั้นจะเป็นการเข้าไปแก้ไขในส่วนของการเพิ่ม-ลบ-ชนิดของงรวมถึงการใช้เวลาของงานแต่ละชนิดด้วย การแก้ไขงานสามารถทำได้ดังนี้

จากหน้าตัวเลือกสิ่งที่ต้องการแก้ไข จากนั้นกดปุ่ม"งาน"



รูปที่ 4.12 หน้าตาGUIของการแก้ไขและการเพิ่มงาน

จากนั้นจะปรากฏหน้าต่างให้แก้ไขงานดังรูป



รูปที่ 4.13 หน้าตาGUIของการแก้ไขและเติมงาน

- การเพิ่มงานใหม่ทำได้ดังนี้
1. พิมพ์ชื่องานลงไป
  2. ใส่เวลาของงานที่ใช้ต่อหน่วย(ชม.)
  3. ใส่ราคาในการให้บริการลงไป
  4. กดปุ่ม"เพิ่ม"เพื่อทำการเพิ่มงานลงฐานข้อมูล

รูปที่ 4.14 หน้าตาGUIของการเพิ่มงาน

-การแก้ไขรายละเอียดของงานที่มีอยู่ ทำได้ดังนี้

1. กดปุ่ม"แสดงงานทั้งหมด"
2. เลือกชนิดของงานจากCombo box
3. พิมพ์เวลาที่ต้องการเปลี่ยนลงไป
4. พิมพ์ราคาการให้บริการลงไป
5. กดปุ่ม"แก้ไข"เพื่อทำการแก้ไขงาน

แก้ไขเลขเพิ่มเติมงาน

เพิ่มงานใหม่    แก้ไข    ลบงาน

ชื่อพนักงาน     งาน     เวลาที่ใช้  ชม.    ราคา  บาท

งานที่เลือก

แสดงงานทั้งหมด

ลบ

รูปที่ 4.15 หน้าตาGUIของการแก้ไขงาน

- การลบงานที่มีอยู่ทำได้ดังนี้
  1. กดปุ่ม"แสดง โหนดทั้งหมด"
  2. เลือกงานที่ต้องการลบจากCombo box
  3. กดปุ่ม"ลบ" เพื่อทำการลบงาน
- เมื่อแก้ไขงานเรียบร้อยแล้วให้กดปุ่ม"กลับ"

## 4.2 เลือกวางแผนการเดินทาง

รูปที่ 4.16 หน้าตาGUIของการวางแผนการเดินทาง

เมื่อทำการกดปุ่มวางแผนการเดินทางก็จะปรากฏหน้าต่างดังรูป

1. กดปุ่ม “แสดงข้อมูล”
2. เลือกจุดที่เป็นที่ตั้งของบริษัทจาก Combo box
3. เลือกงานที่จะต้องไปทำจาก Combo box
4. ใส่งานที่จะไปทำของแต่ละจุด
5. ใส่หน่วยของงานที่จะไปให้บริการ
6. เมื่อป้อนจุดที่ต้องการจะไปครบตามที่ต้องการแล้วให้กดปุ่ม “ต่อไป”



โปรแกรมจะทำการประมวลผลแล้วจะปรากฏหน้าต่างดังรูป

แสดงข้อมูลการเดินทางและการให้บริการ

จุดที่ต้องเดินทางผ่าน: ข้อมูลโดยรวม

แม่ทวนคลอง	ระยะทางทั้งหมด: 6.4974	กิโลเมตร
โรงพยาบาลมาเชียง	ค่าเช่ารถในการเดินทาง: 17,542.98	บาท
แม่ทวนหลังจังหวัด	รวม: 5,100.00	บาท
โรงเรียนมาลาเมือง	ต้องใช้เวลานั่งรถ: 5	ชั่วโมง
แม่ทวนคลอง		
ร้านดอกจิมเฮลเลอร์ฟลอริดา		
หมู่บ้านเทศบาล		
วัดพระตะโกลก		
หมู่บ้านเทศบาล		
ร้านดอกจิมเฮลเลอร์ฟลอริดา		

จุดที่ต้องไปทางแม่ทวนหลังจังหวัด: จุดที่ต้องไปทางวัดพระตะโกลก: จุดที่ต้องไปทางสถานรถไฟ

งาน:  งาน:  งาน:

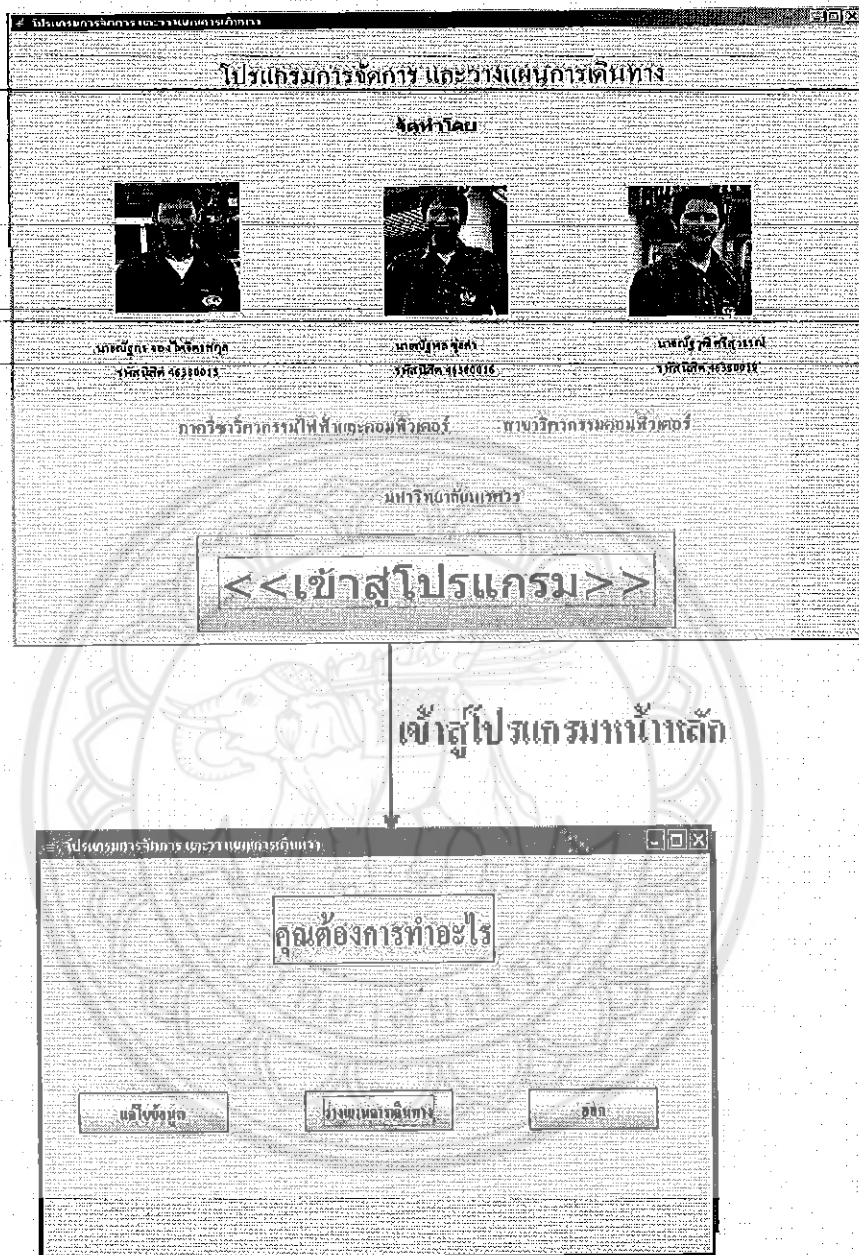
เวลาว่าง:  เวลาใส่:  เวลาใส่:

<<กลับ แสดงข้อมูลจากโปรแกรมประมวลผล ออกจากโปรแกรม

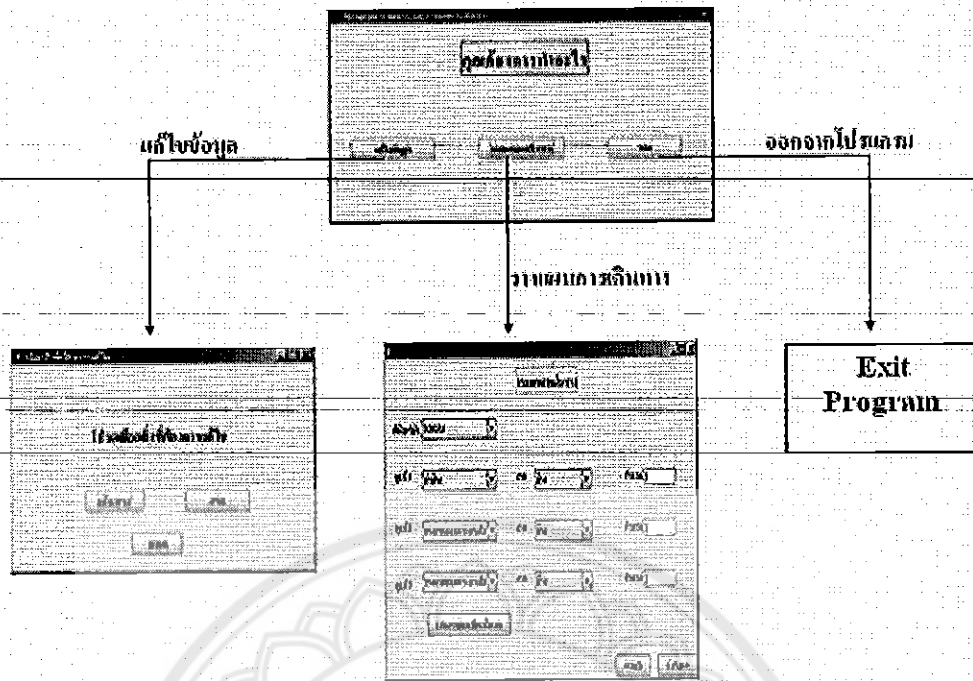
รูปที่ 4.17 หน้าตาGUIของการแสดงเส้นทาง

เมื่อเริ่มต้นจะยังไม่มีข้อมูลในการประมวลผลให้กดปุ่ม”แสดงข้อมูลจากการประมวลผล” จากนั้นข้อมูลจากการประมวลผลก็จะแสดงออกมาดังรูป และถ้าต้องการกลับไปวางแผนใหม่อีกครั้งก็ให้กดปุ่ม”กลับ” เพื่อกลับไปหน้าเดิมอีกครั้ง

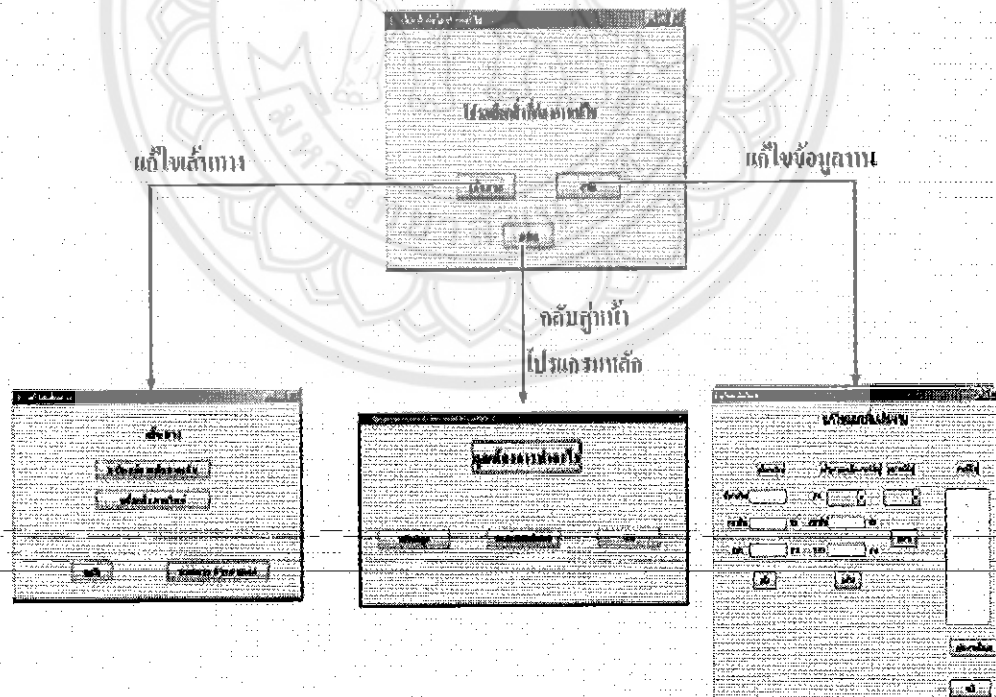
4.3 Action GUI



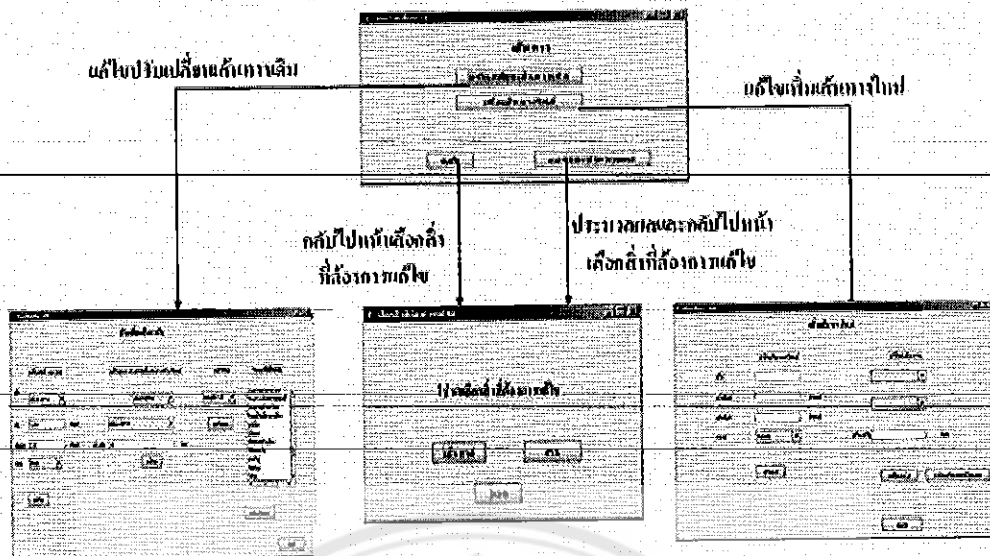
รูปที่ 4.18 Action GUI



รูปที่ 4.19 Action GUI



รูปที่ 4.20 Action GUI



รูปที่ 4.21 Action GUI

This screenshot shows the 'ประมวลผลค่าที่ส่ง' (Process Shipping Values) window. It features a table with columns for 'ค่าที่ส่ง' (Shipping Value), 'ค่าที่ส่งต่อ' (Next Shipping Value), and 'ค่าที่ส่งต่อ' (Next Shipping Value). The table contains three rows of data. Below the table is a 'ประมวลผลค่าที่ส่ง' (Process Shipping Values) button and 'ตกลง' (OK) and 'ยกเลิก' (Cancel) buttons.

ทำการประมวลผล

This screenshot shows the 'ข้อมูลสินค้า' (Product Information) window. It is divided into two panes: 'ข้อมูลสินค้า' (Product Information) on the left and 'ข้อมูลลูกค้า' (Customer Information) on the right. The left pane contains a list of product details, and the right pane contains a form with fields for 'ชื่อลูกค้า' (Customer Name), 'เลขที่บัญชี' (Account Number), and 'สาขา' (Branch). There are 'ตกลง' (OK) and 'ยกเลิก' (Cancel) buttons at the bottom.

รูปที่ 4.22 Action GUI



## บทที่ 5

### สรุปผล

#### 1. สรุปผลโครงการ

ผลที่ได้จากโครงการระบบจัดการการให้บริการเกี่ยวกับบริษัทขายเครื่องปรับอากาศนี้ ทำให้เราช่วยในการตัดสินใจได้ในระดับหนึ่ง แต่ยังไม่ถูกต้องทั้งหมดเนื่องจากข้อจำกัดของด้านรายละเอียดของแผนที่ ที่มี และผู้ที่นำไปใช้ไม่ปฏิบัติตาม อาจจะเพราะเนื่องจากความเคยชินในการใช้เส้นทางเก่า หรือ ชูระส่วนตัว ผลที่ได้จากโปรแกรมนี้จึงเป็นการแนะนำว่าควรปฏิบัติตามเพื่อให้ได้ผลประโยชน์สูงสุด ผู้จัดทำหวังเป็นอย่างยิ่งว่าจะมีการพัฒนา โปรแกรมนี้ให้มีประสิทธิภาพสูงขึ้น การทำงานที่ซับซ้อน ภายได้ข้อจำกัดที่ซับซ้อนมากขึ้น

ผลที่คาดว่าจะได้รับ ผู้จัดทำโครงการจึงได้มีแนวคิดที่จะสร้าง โปรแกรมที่จะช่วยในการตัดสินใจ สำหรับผู้ประกอบการที่จะกำหนดการไปให้บริการในแต่ละครั้งเพื่อความคุ้มค่ากับค่าใช้จ่าย และเป็นการประหยัดน้ำมันให้มากที่สุดและใช้งานได้อย่างมีประสิทธิภาพ

#### 2. ปัญหาและอุปสรรค

ปัญหาที่พบบ่อยในการสร้างระบบจัดการการให้บริการเกี่ยวกับบริษัทขายเครื่องปรับอากาศ คือ ปัญหาการจัดเก็บฐานข้อมูล ความละเอียดของแผนที่ การแสดงรายละเอียดเส้นทางการเดินทางบนภาพแผนที่ และแผนที่ที่ใช้ในการประมวลผลนี้อาจจะไม่ใช่เส้นทางที่สั้นที่สุดเพราะว่าในการเก็บฐานข้อมูล จะมีแค่เส้นทางหลวงที่ปรากฏอยู่ในแผนที่เท่านั้น ซึ่งในความเป็นจริงจะมีเส้นทางและถนนแยกต่างๆ มากมายกว่านี้ เกินกว่าเจ้าหน้าที่( กรมโยธาธิการ , เทศบาล, และ ้องการบริหารส่วนจังหวัด) ที่เกี่ยวข้องจะจัดเก็บได้ทั้งหมด

#### 3. แนวทางแก้ไขปัญหา

จากปัญหาในข้อ 2. เกิดจากข้อจำกัดด้านแผนที่ แม้แต่หน่วยงานรับผิดชอบที่เกี่ยวข้องด้านแผนที่โดยตรงก็ไม่สามารถมีข้อมูลแผนที่เส้นทางถนนครบถ้วนหมด จึงจำเป็นต้องมีการไปหาข้อมูลด้วยตัวเอง เส้นทางทางบางจุดต้องใช้การจัดเก็บข้อมูลเอง แต่จะไม่แสดงในแผนที่ที่จะทำการจัดเก็บในฐานข้อมูลเท่านั้น

#### 4. ข้อเสนอแนะ

จากระบบการจัดการการให้บริการเกี่ยวกับบริษัทขายเครื่องปรับอากาศนั้น ได้มีการพัฒนาโดยใช้ภาษาจาวา และ ฐานข้อมูล MySQL ซึ่งการใช้เครื่องมือพัฒนาเหล่านี้ทำให้ระบบที่ได้ มีความเสถียร สะดวกต่อการปรับปรุงแก้ไข และเข้าใจได้ง่าย สามารถทำให้ผู้อื่นนำไปพัฒนาต่อได้ ซึ่งอาจพัฒนาให้เป็นระบบที่มีขนาดใหญ่ และมีความซับซ้อนมากยิ่งขึ้นรวมถึงการประมวลผลได้เร็วขึ้นด้วย



## เอกสารอ้างอิง

- [1] วนิกา เหมาะกุล. กลณิตศาสตร์ ดิสกรีต (DISCRETE MATHEMATICS) .กรุงเทพมหานคร:  
บริษัท ซีเอ็ดดูเคชั่น จำกัด.2535
- [2] วีรศักดิ์ ชิงดาว. java programming Volume II.กรุงเทพมหานคร:  
บริษัท ซีเอ็ดดูเคชั่น จำกัด.2547
- [3] Rosen, Discrete Mathematics and Its Applications, McGraw-Hill,-1999—
- [4] “แนะนำทฤษฎีของ Dijkstra’s Algorithm” [Online]. Available:  
[http://vcharkarn.com/magazine/issue4/issue004\\_chaw.php](http://vcharkarn.com/magazine/issue4/issue004_chaw.php)
- [5] “แนะนำทฤษฎีของการเรียงสับเปลี่ยน Permutation” [Online]. Available:  
<http://e-learning.snru.ac.th/els/ris24/permutation.html>
- [6] “เว็บของกรมทางหลวง” [Online]. Available:  
[www.doh.go.th/dohweb/index.html](http://www.doh.go.th/dohweb/index.html)
- [7] Y.Jaruwan “ความรู้ทั่วไปเกี่ยวกับระบบฐานข้อมูล.” [Online]. Available:  
<http://www.chandra.ac.th/office/ict/document/it04page01.html.htm>. 2001
- [8] มณีโชติ สมานไทย. “คู่มือการออกแบบฐานข้อมูลและภาษา SQL ฉบับผู้เริ่มต้น”.  
กรุงเทพมหานคร: โนนสุทธาการพิมพ์. 2546



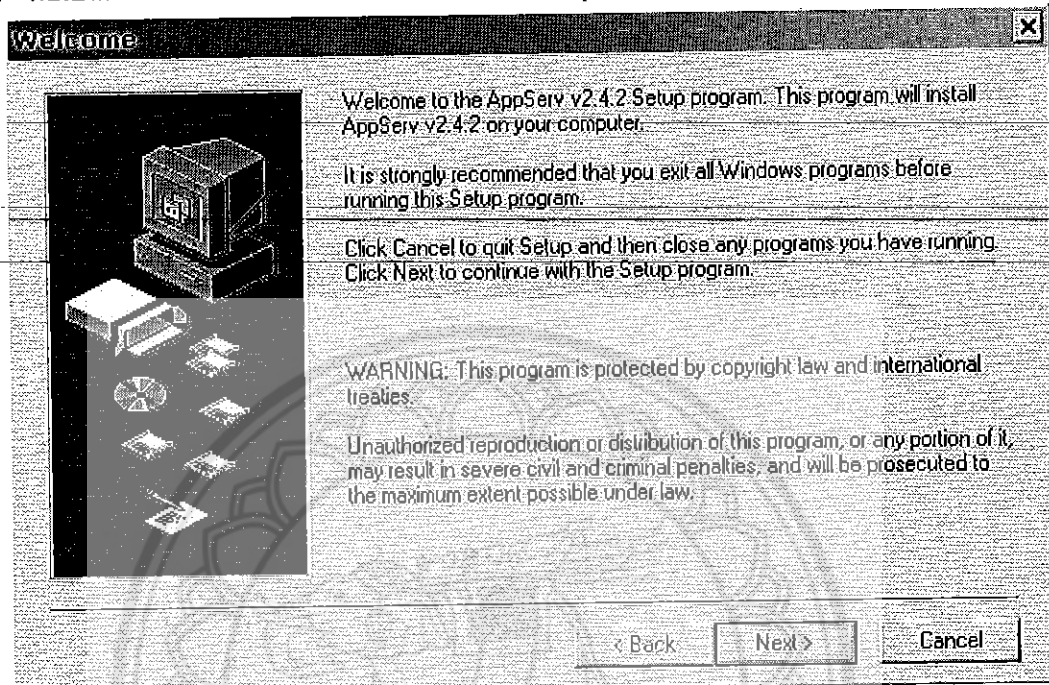




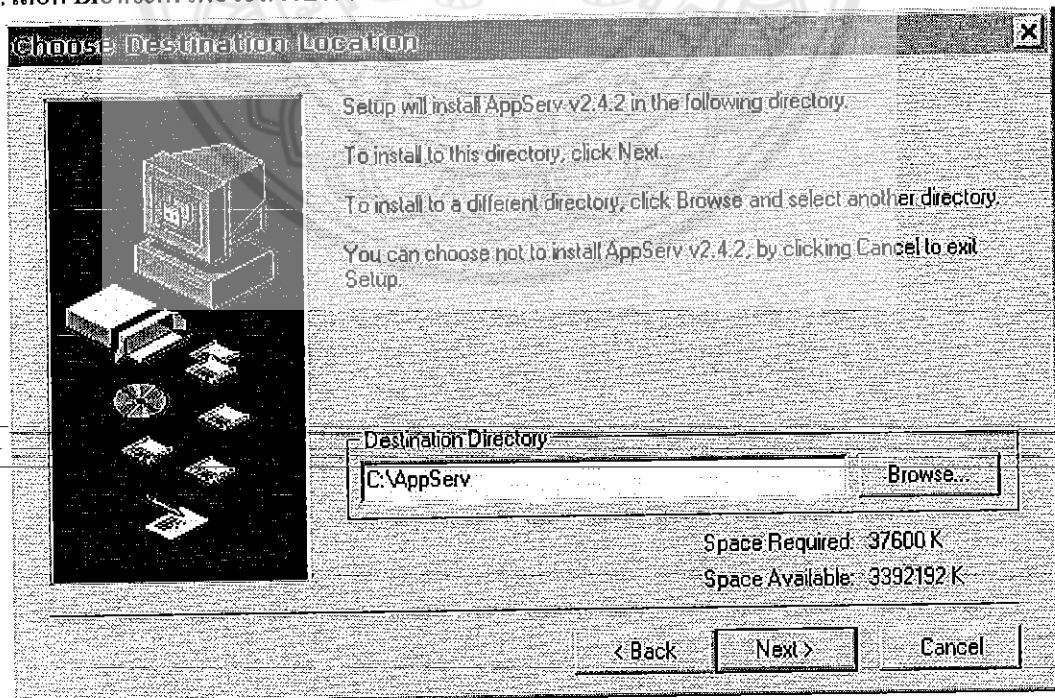
## ภาพตัวอย่างการติดตั้งโปรแกรมสำหรับโปรแกรม Appserve v2.4.2

<http://www.appservnetwork.com/> แหล่งดาวน์โหลด Appserve

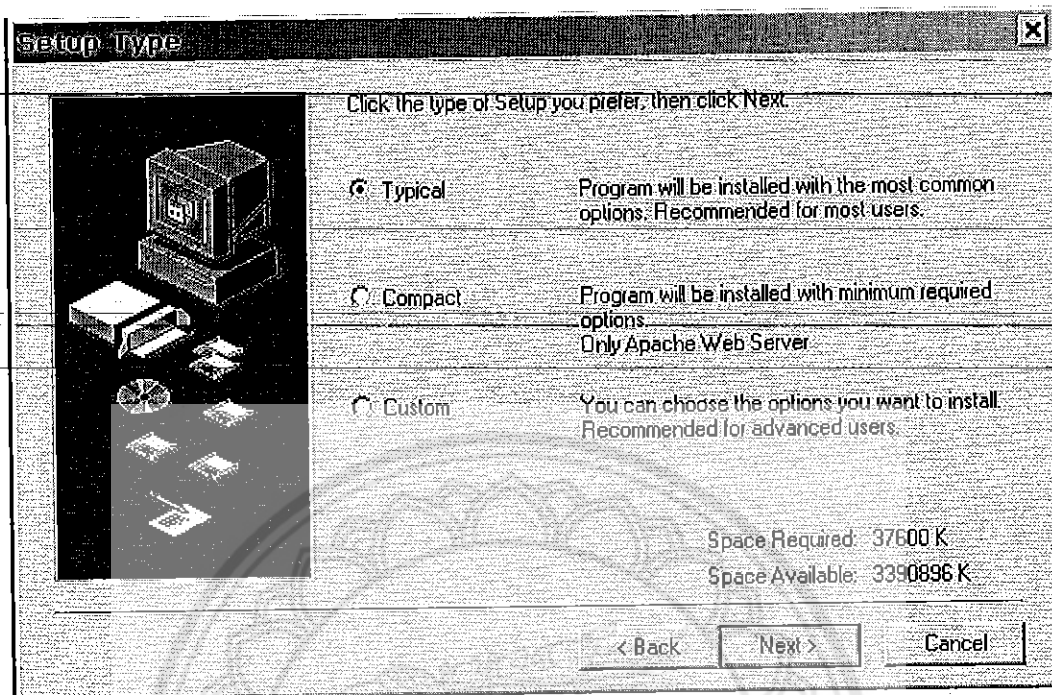
1. ค้างเบ็ดคลิกที่ไฟล์ติดตั้งจะมีหน้าต่างขึ้นมาแล้วคลิกปุ่ม Next



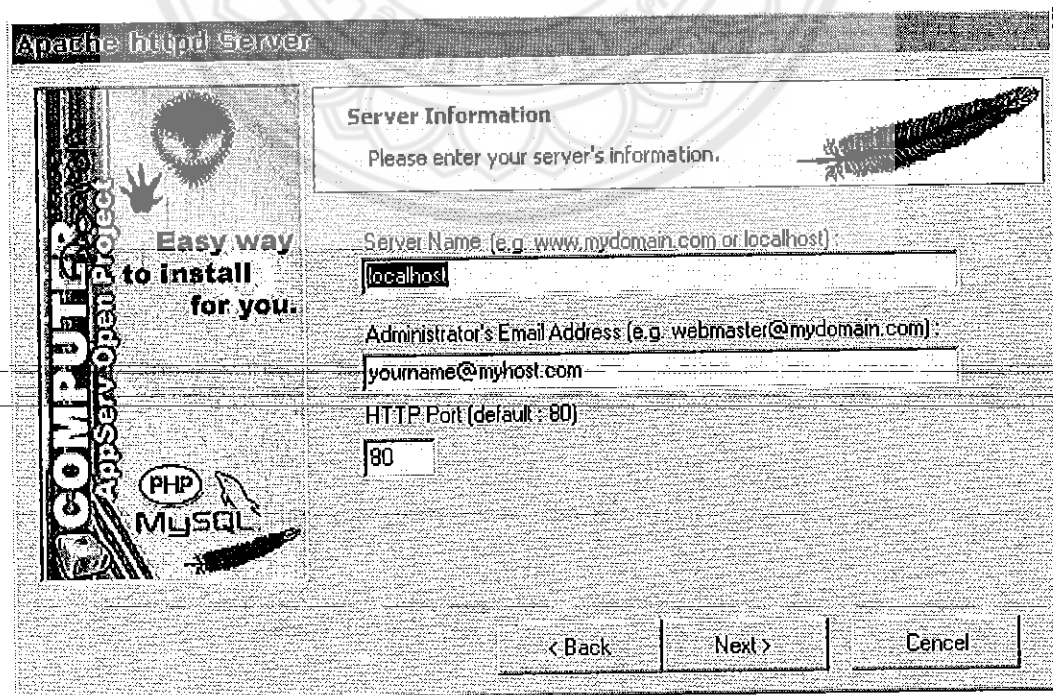
2. เลือก Browse... เพื่อจะลง โปรแกรมในที่ใดแล้วกด Next



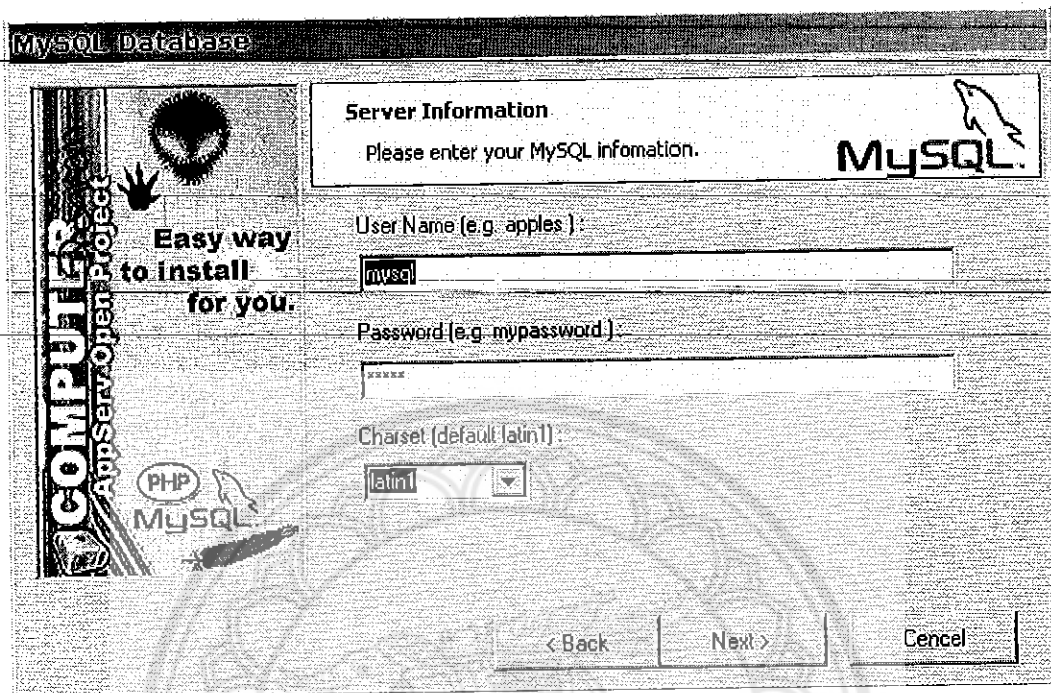
### 3. เลือกที่ Typical แล้วกดปุ่ม Next



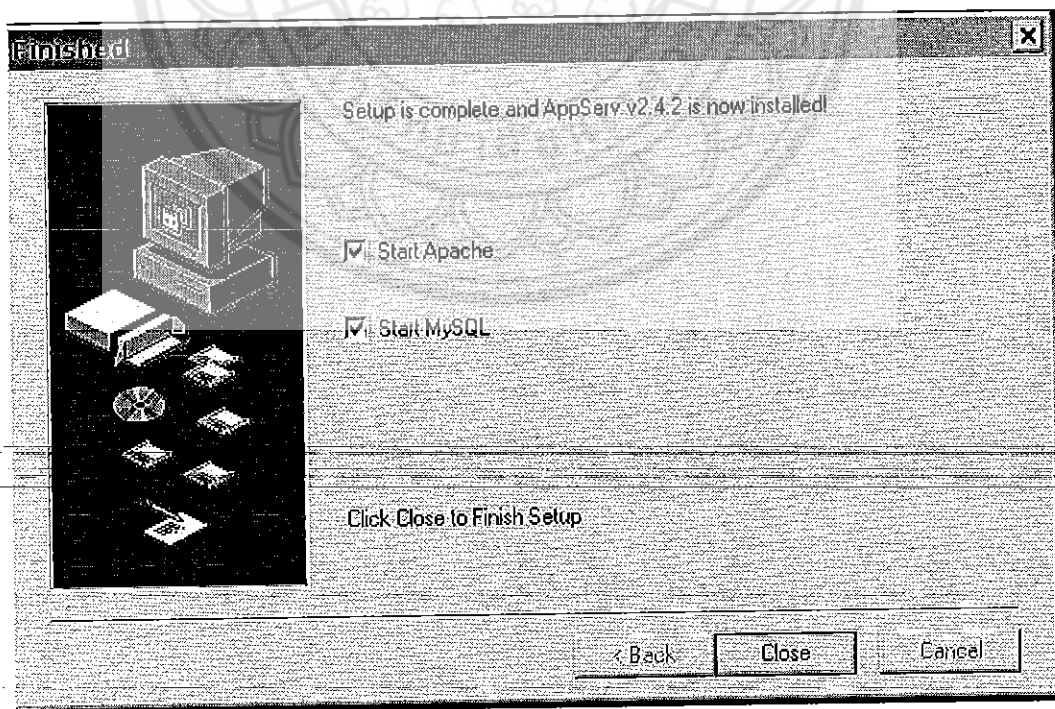
4. เมื่อมีหน้าต่างนี้ขึ้นมาให้ใส่ Server Name , Administrator's Email Address และ HTTP Port เมื่อใส่เรียบร้อยแล้วก็ให้กดปุ่ม Next



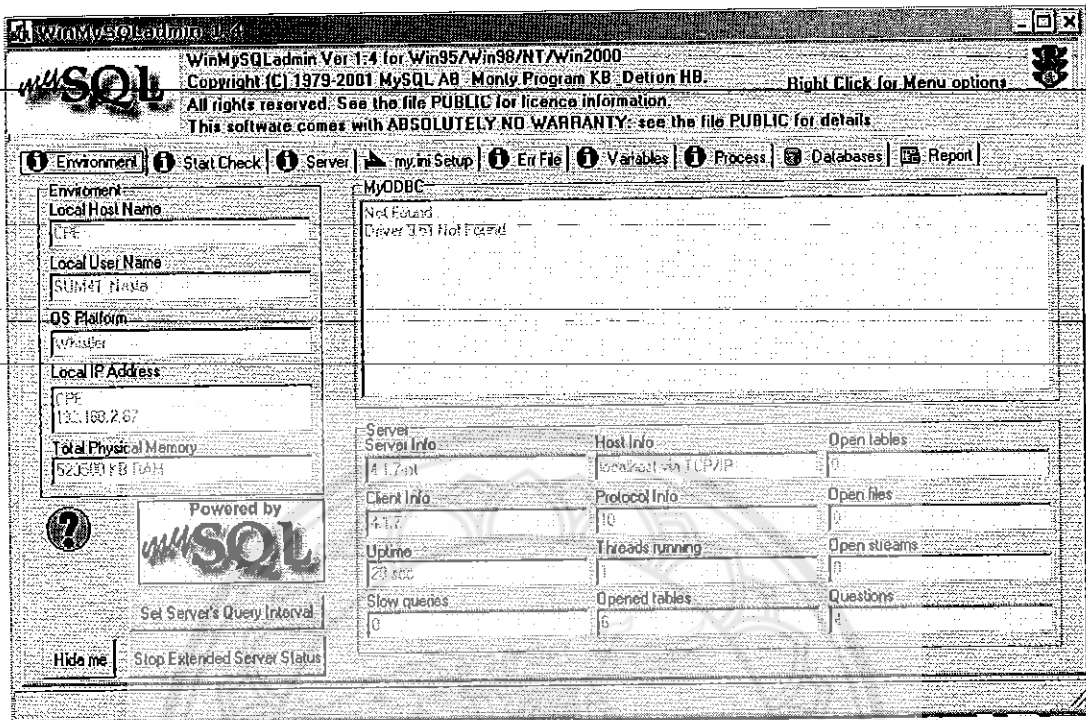
5. จากนั้นก็จะมีหน้าต่างนี้ให้ใส่ที่ User Name , Password และ Charset จากนั้นก็กดปุ่ม Next



6. เมื่อมีหน้าต่างนี้มาก็ทำการกดปุ่ม Close

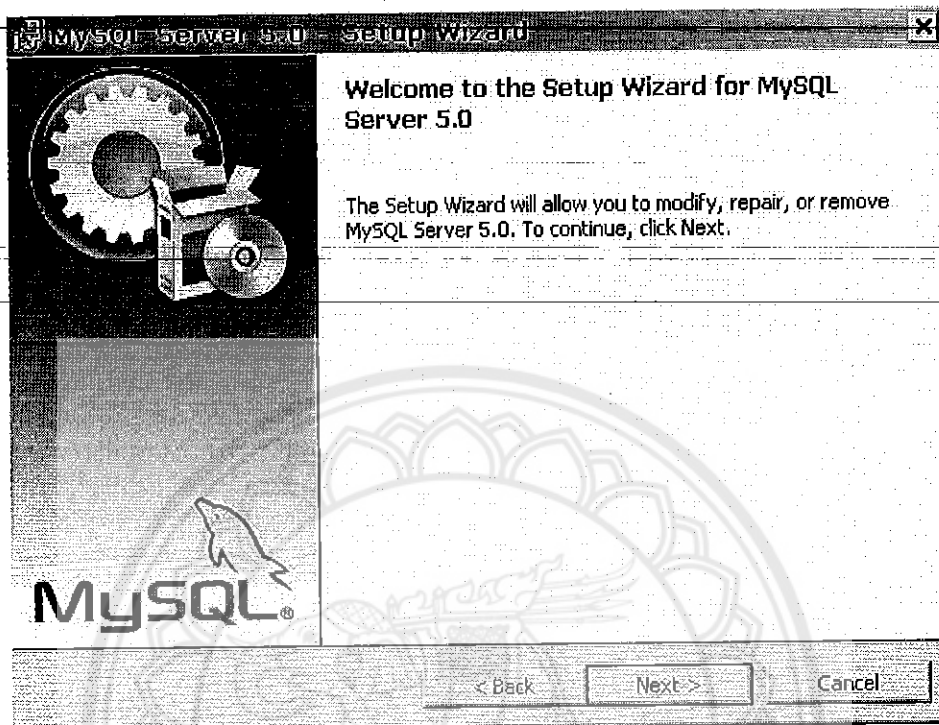


7. เราก็จะได้หน้าต่างนี้ขึ้นมาเป็นอันเสร็จสิ้นของการลงโปรแกรมนี้

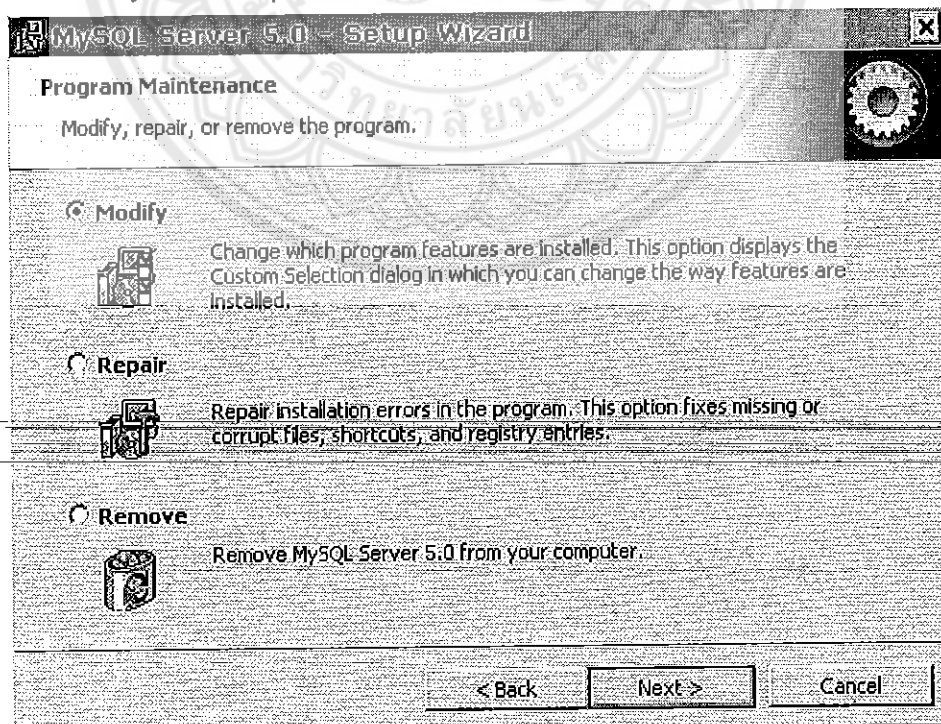


## วิธีการติดตั้ง MySQL Server 5.0

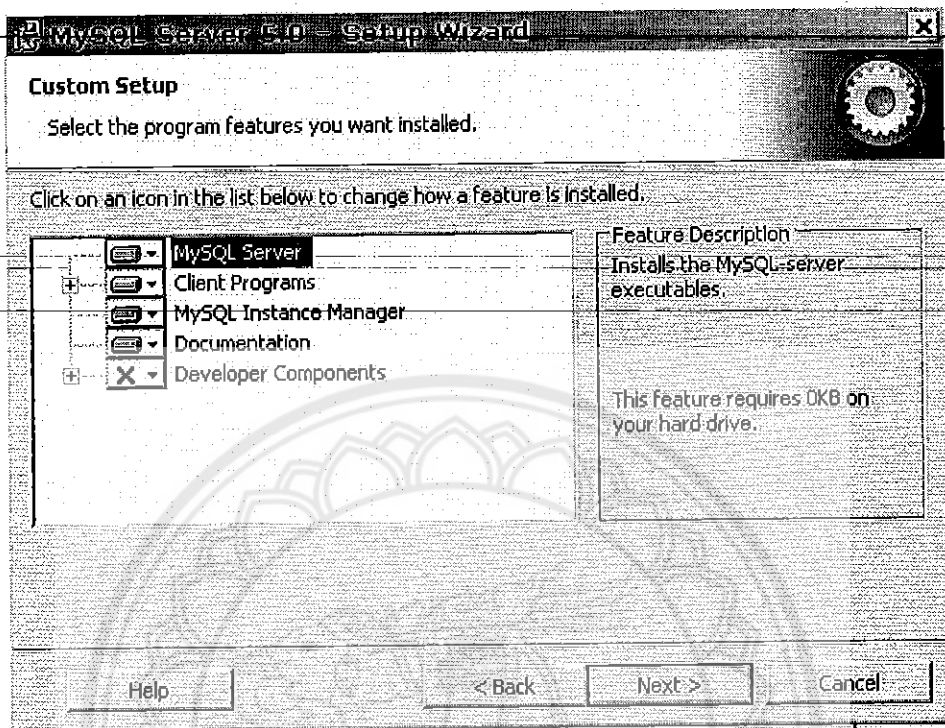
1. ดับเบิลคลิกที่ไฟล์ติดตั้งจะมีหน้าต่างนี้ขึ้นมาแล้วให้กดที่ปุ่ม Next



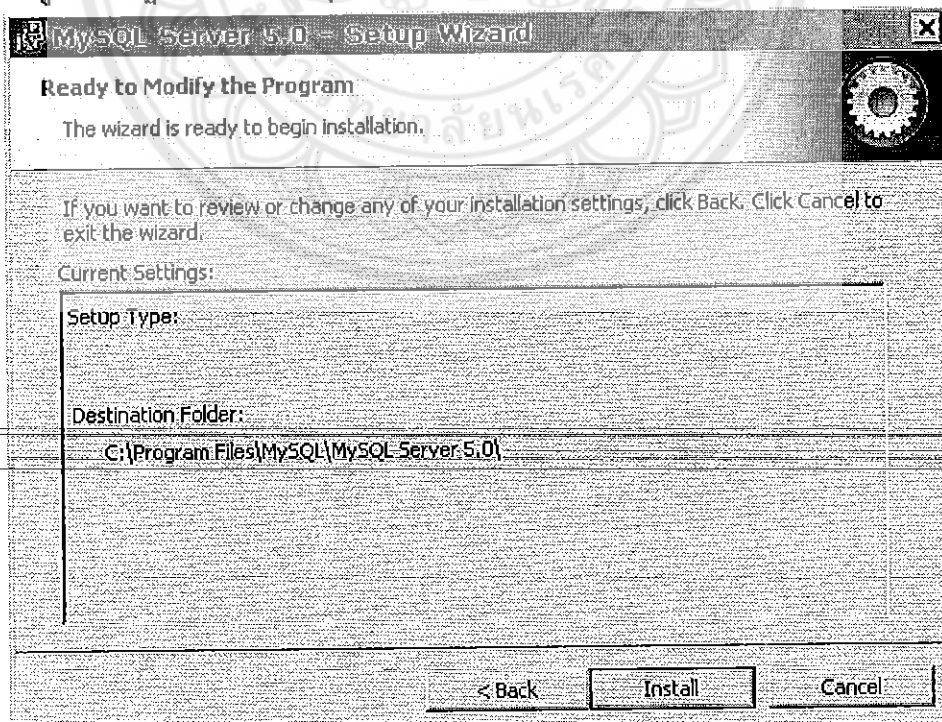
2. เลือกตรง Modify แล้วให้กดปุ่ม Next



3. รอสักครู่จะปรากฏหน้าต่าง ให้ติดตั้งตัวพัฒนาโปรแกรม ให้กดปุ่ม Next

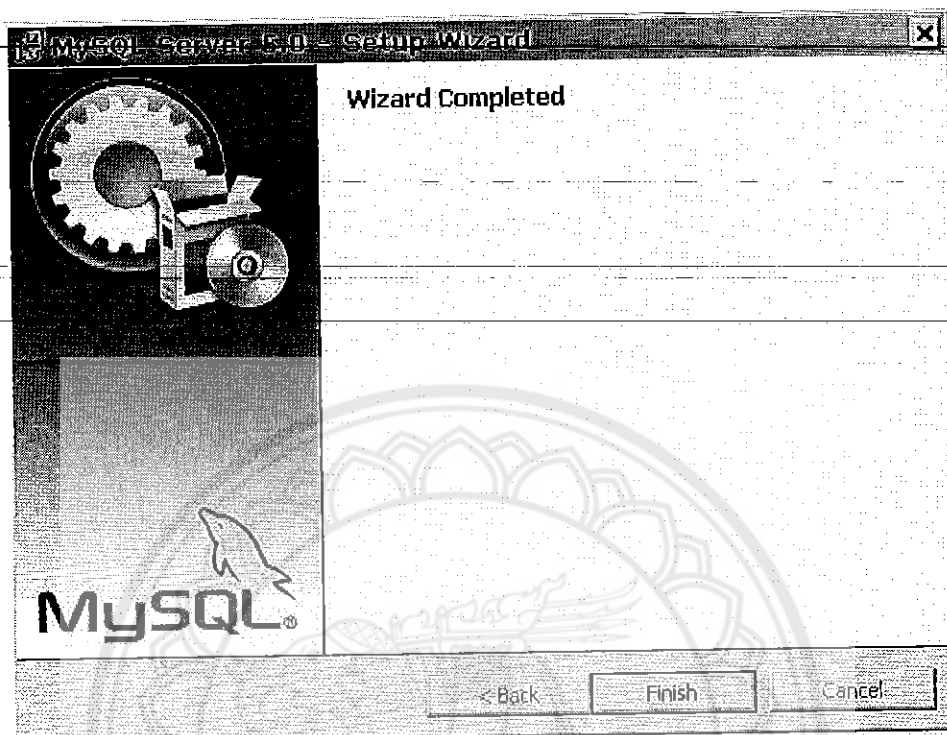


4. รอสักครู่จะปรากฏหน้าต่าง ให้กดปุ่ม Install เพื่อทำการติดตั้ง





5. เมื่อทำการติดตั้งเสร็จเรียบร้อยแล้ว จะปรากฏหน้าต่างดังรูป ให้กดปุ่ม Finish เป็นอันติดตั้งโปรแกรมเสร็จเรียบร้อย

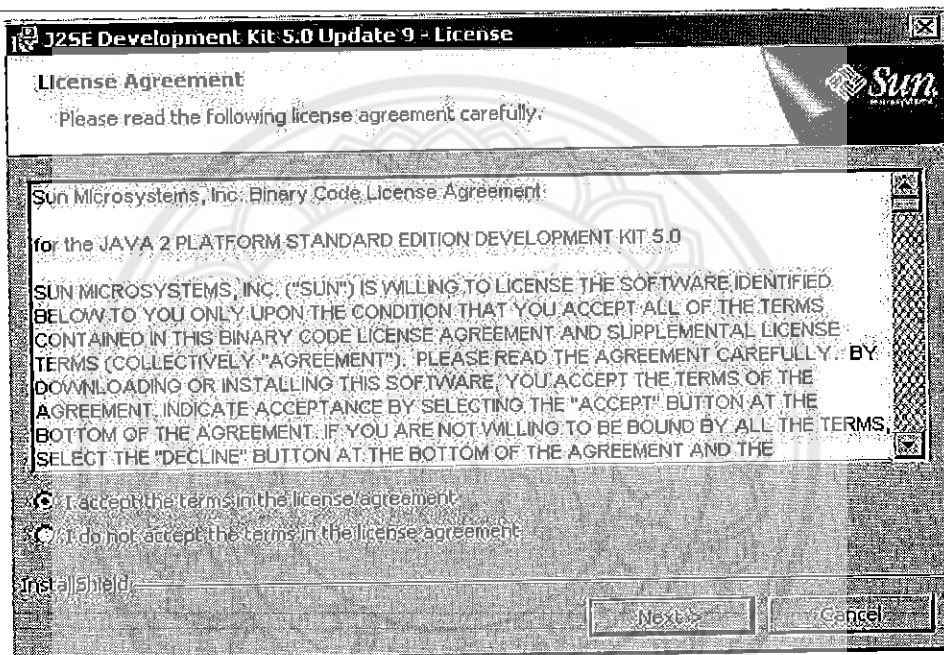


## ขั้นตอนการลงJ2SE 5.0

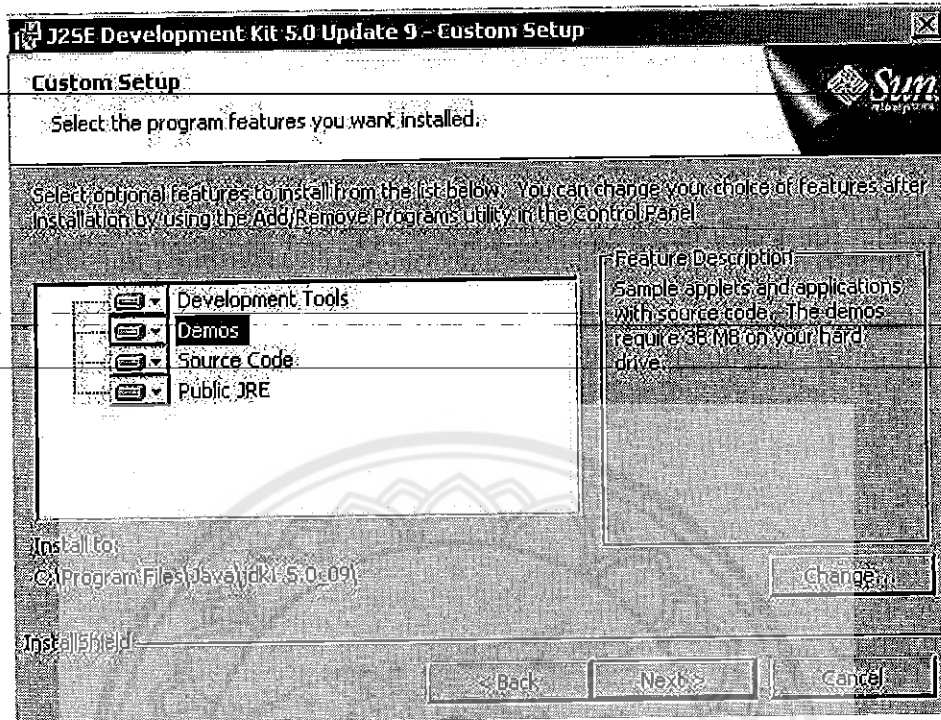
ทำการ Download ตัวติดตั้งจาก <http://www.sun.com> จากนั้นจะได้ตัวติดตั้งที่เป็นไฟล์  
สำหรับติดตั้ง ในการทำโครงการนี้ให้เลือกใช้ Jdk-1\_5\_0\_09-windows-i586-p การติดตั้งทำได้ดังนี้

1.ดับเบิลคลิกที่ไฟล์ติดตั้ง ก็จะมีหน้าต่างสำหรับการยินยอมรับเงื่อนไขของโปรแกรมให้เลือก

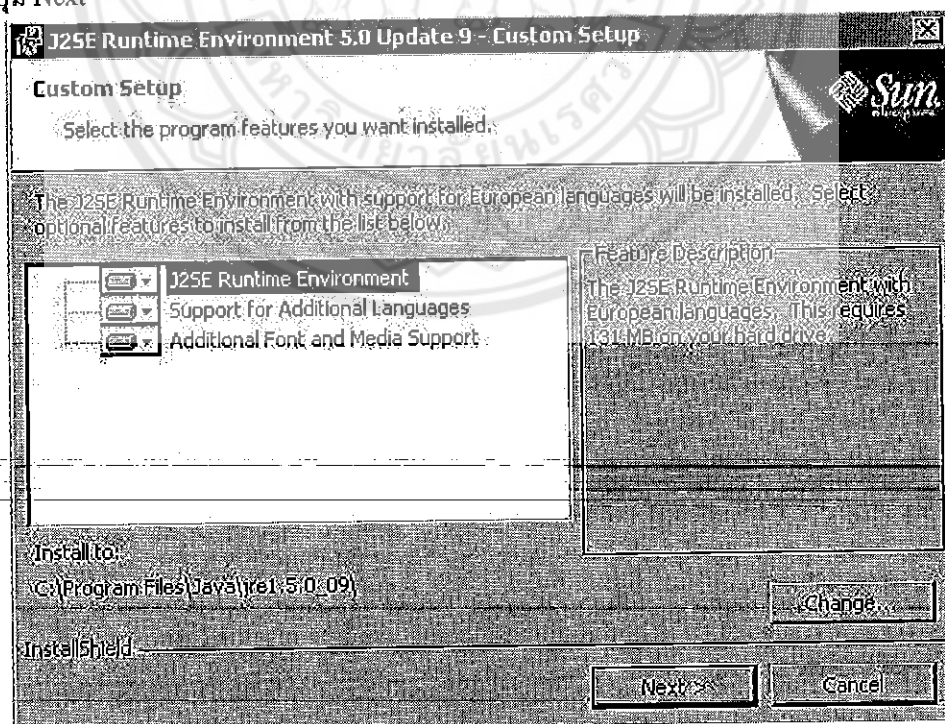
I accept the term in the license agreement แล้วคลิกปุ่ม Next



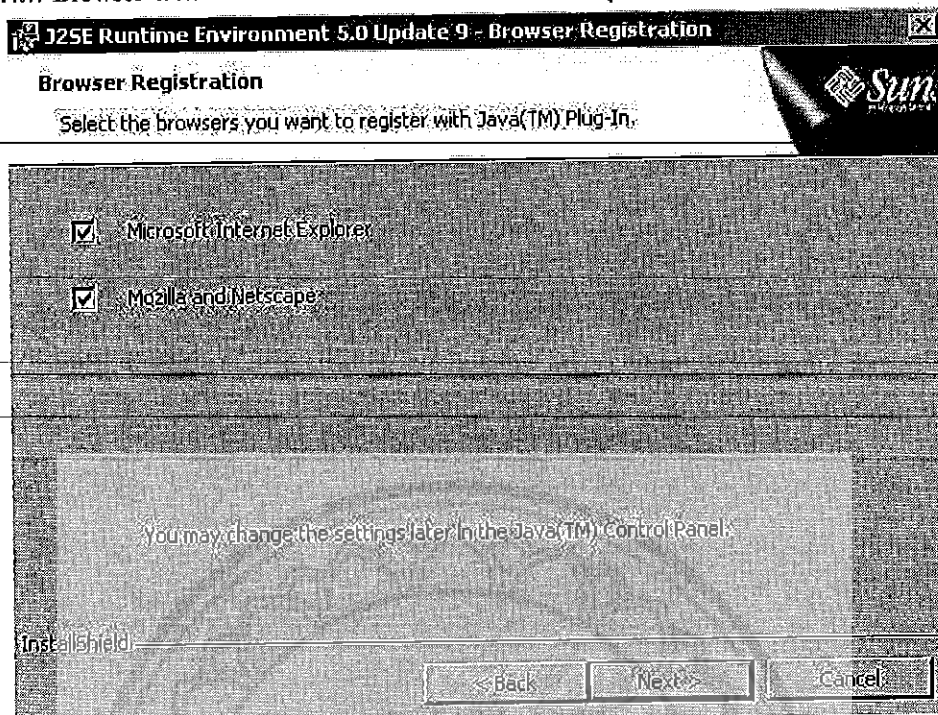
2. รอสักครู่จะปรากฏหน้าต่าง ให้ติดตั้งตัวพัฒนาโปรแกรม ให้คลิกปุ่ม Next



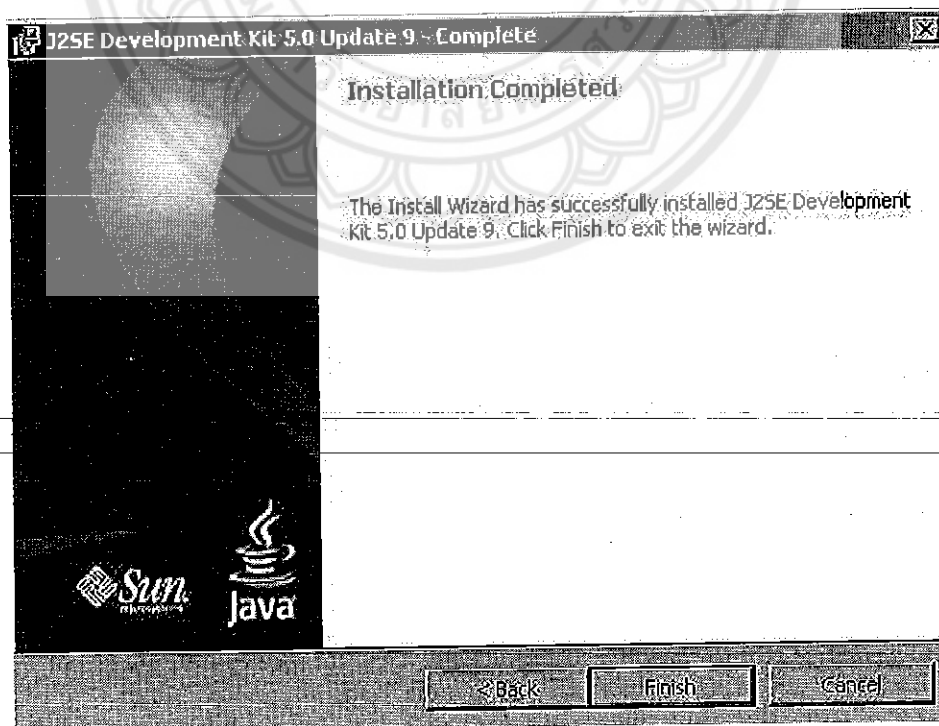
3. เมื่อโปรแกรมติดตั้งตัวพัฒนาโปรแกรมเสร็จ ก็จะมีหน้าต่างสำหรับติดตั้ง java runtime environment ให้คลิกปุ่ม Next



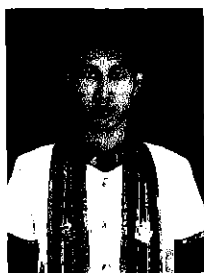
4. กำหนด Browser สำหรับใช้จับJava ให้เลือกทั้งหมด แล้วกดปุ่มNext



5. เมื่อทำการติดตั้งเสร็จเรียบร้อยแล้ว จะปรากฏหน้าต่างดังรูป ให้กดปุ่ม Finish เป็นอันติดตั้งโปรแกรมเสร็จเรียบร้อยแล้ว



## ประวัติผู้เขียนโครงการ



ชื่อ นายจตุกร จองไพจิตรสกุล  
 ภูมิลำเนา 141 หมู่ 6 ต.ศรีษะเกษ อ.นาน้อย จ.น่าน 55150

### ประวัติการศึกษา

- จบมัธยมศึกษาจาก โรงเรียนศรีสวัสดิ์ วิทยาคาร จังหวัดน่าน
- ปัจจุบันกำลังศึกษาอยู่ระดับปริญญาตรีชั้นปีที่ 4 สาขาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail: sum41\_nesta@hotmail.com



ชื่อ นายณัฐพล นุ้ยคำ  
 ภูมิลำเนา 111 หมู่ 2 ต.บ้านเหล่า อ.แม่ใจ จ.พะเยา 56130

### ประวัติการศึกษา

- จบมัธยมศึกษาจาก โรงเรียนพะเยาพิทยาคม
- ปัจจุบันกำลังศึกษาอยู่ระดับปริญญาตรีชั้นปีที่ 4 สาขาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail: nutthapon\_x@yahoo.co.th



ชื่อ นายณัฐวุฒิ ศรีสุวรรณ  
 ภูมิลำเนา 24/1 ถ. พหลโยธิน ต.แม่ต้า อ.เมือง จ.พะเยา 56000

### ประวัติการศึกษา

- จบมัธยมศึกษาจาก โรงเรียนพะเยาพิทยาคม
- ปัจจุบันกำลังศึกษาอยู่ระดับปริญญาตรีชั้นปีที่ 4 สาขาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail: arm\_nutthawut@hotmail.com