

**สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง**

**ระบบบริการหาเส้นทางกรุงเทพฯ**

**BANGKOK ROUTES SERVICE SYSTEM**



เลขหมู่.....  
เลขทะเบียน.....**72724**  
วัน,เดือน,ปี.....**22 ส.ย. 2550**

b. 41221738  
i. ....

**ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต**

**ภาควิชาวิศวกรรมสารสนเทศ**

**คณะวิศวกรรมศาสตร์**

**สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง**

**ปีการศึกษา 2549**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**BANGKOK ROUTES SERVICE SYSTEM**

**BY**

**APINAN WUTTAWANIT 46010921**

**ARCHARIYA SIVANART 46010953**

**EKACHAI VIRIYAMAT 46010986**

**A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF  
THE REQUIREMENT FOR THE DEGREE OF  
BACHELOR IN DEPARTMENT OF INFORMATION ENGINEERING  
FACULTY OF ENGINEERING  
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

**2006**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

|                    |                               |                       |
|--------------------|-------------------------------|-----------------------|
| หัวข้อปริญญานิพนธ์ | ระบบบริการหาเส้นทางกรุงเทพฯ   |                       |
| นักศึกษา           | นายอภิรักษ์ วุฑฒะวณิช         | รหัสนักศึกษา 46010921 |
|                    | นายอัจฉริยะ สิวินารณ          | รหัสนักศึกษา 46010953 |
|                    | นายเอกชัย วิริยะมัติ          | รหัสนักศึกษา 46010986 |
| อาจารย์ที่ปรึกษา   | ผศ. มยุรี เลิศเวชกุล          |                       |
| ระดับการศึกษา      | ปริญญาตรี วิศวกรรมศาสตรบัณฑิต |                       |
| ภาควิชา            | วิศวกรรมสารสนเทศ              |                       |
| ปีการศึกษา         | 2549                          |                       |

ปริญญานิพนธ์นี้ได้รับความเห็นชอบจากอาจารย์ที่ปรึกษาเป็นที่เรียบร้อยแล้ว



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

|                    |                               |            |                       |
|--------------------|-------------------------------|------------|-----------------------|
| หัวข้อปริญญานิพนธ์ | ระบบบริการหาเส้นทางกรุงเทพฯ   |            |                       |
| ชื่อนักศึกษา       | นายอภิรักษ์                   | วุฒชะวนิช  | รหัสประจำตัว 46010921 |
|                    | นายอัจฉริยะ                   | ศิวนารถ    | รหัสประจำตัว 46010953 |
|                    | นายเอกชัย                     | วิริยะมัติ | รหัสประจำตัว 46010986 |
| อาจารย์ที่ปรึกษา   | ศศ.มยุรี เลิศเวชกุล           |            |                       |
| ระดับการศึกษา      | ปริญญาตรี วิศวกรรมศาสตรบัณฑิต |            |                       |
|                    | สาขาวิศวกรรมสารสนเทศ          |            |                       |
| ภาควิชา            | วิศวกรรมสารสนเทศ              |            |                       |
| ปีการศึกษา         | 2549                          |            |                       |

### บทคัดย่อ

โครงการนี้เป็นระบบบริการหาเส้นทางการเดินทางภายในกรุงเทพมหานคร โดยอ้างอิงจากสถานที่สำคัญเป็นต้นทางและปลายทาง และใช้อัลกอริทึมในการหาเส้นทางที่คำนึงถึงตัวแปรระยะทางและสภาพการจราจร ณ เวลานั้น เพื่อประมวลผลหาเส้นทางที่เหมาะสมที่สุดสามเส้นทางแรก และนำเส้นทางที่ได้มาแสดงผลโดยใช้แผนที่พร้อมทั้งวาดเส้นแสดงเส้นทางอย่างชัดเจน เพื่อให้ผู้ใช้สามารถเลือกเส้นทางที่จะเดินทางได้อย่างรวดเร็ว การใช้บริการนี้สามารถเข้าถึงผ่านทางเว็บแอปพลิเคชัน (Web Application) ได้

|                       |  |            |          |
|-----------------------|--|------------|----------|
| <b>Thesis Title</b>   | Bangkok Routes Service System              |            |          |
| <b>Student</b>        | Apinan                                     | Wuttawanit | 46010921 |
|                       | Archariya                                  | Sivanart   | 46010953 |
|                       | Ekachai                                    | Viriyamat  | 46010986 |
| <b>Advisor</b>        | Asst. Prof. Mayuree Lertwatechakul         |            |          |
| <b>Graduate Level</b> | Bachelor Degree of Information Engineering |            |          |
| <b>Department</b>     | Information Engineering                    |            |          |
| <b>Academic Year</b>  | 2006                                       |            |          |

### ABSTRACT

This project proposes Bangkok Routes Service System that calculates 3 best routes for 2 endpoints within Bangkok area, by applying SPF algorithm with route metric such as distance and traffic condition.

Bangkok Routes Service System is developed as Web-based application. The result is shown as high-light line onto the Bangkok map that can be easy to understand by user.

# สารบัญ

| เรื่อง  | หน้า |
|---|------|
| บทคัดย่อภาษาไทย                                 | ก    |
| บทคัดย่อภาษาอังกฤษ                              | ข    |
| กิตติกรรมประกาศ                                 | ค    |
| บทที่ 1 บทนำ                                    |      |
| 1.1 บทนำ  | 1    |
| 1.2 วัตถุประสงค์                                | 1    |
| 1.3 ขอบเขตของโครงการ                            | 1    |
| 1.4 ระบบงานโดยรวม                               | 2    |
| 1.5 ขั้นตอนการดำเนินโครงการ                     | 2    |
| บทที่ 2 ทฤษฎีและหลักการ                         |      |
| 2.1 อัลกอริทึมเอสพีเอฟ                          | 3    |
| 2.1.1 ตัวอย่างการทำงานของอัลกอริทึมเอสพีเอฟ     | 3    |
| 2.2 การออกแบบฐานข้อมูลโดยใช้แบบจำลองโนแอม       | 8    |
| 2.2.1 ขั้นตอนในการออกแบบโนแอม                   | 9    |
| 2.2.2 ส่วนประกอบพื้นฐานของโนแอม                 | 9    |
| 2.2.3 สัญลักษณ์ของส่วนประกอบพื้นฐานของโนแอม     | 10   |
| บทที่ 3 หลักการออกแบบและการดำเนินการ            |      |
| 3.1 การออกแบบอัลกอริทึมที่ใช้ในโครงการ          | 12   |
| 3.1.1 การนำอัลกอริทึมเอสพีเอฟมาประยุกต์ใช้      | 12   |
| 3.1.2 ตัวอย่างการทำงานของอัลกอริทึม             | 12   |
| 3.2 การออกแบบฐานข้อมูลโดยใช้แบบจำลองโนแอม       | 19   |
| 3.2.1 แนวทางในการออกแบบฐานข้อมูล                | 19   |
| 3.2.2 การออกแบบฐานข้อมูล                        | 19   |
| 3.2.3 การแปลงแบบจำลองโนแอมเป็นตารางความสัมพันธ์ | 20   |
| 3.2.4 พจนานุกรมข้อมูล                           | 21   |
| 3.3 การออกแบบผังงานของโปรแกรม                   | 22   |
| 3.3.1 แนวทางในการออกแบบโปรแกรม                  | 22   |
| 3.3.2 ผังงานทั้งหมดของโปรแกรม                   | 22   |

## สารบัญ(ต่อ)

| เรื่อง                                   | หน้า |
|--|------|
| 3.3.2.1 ผังงานของโปรแกรมหลัก             | 23   |
| 3.3.2.2 ผังงานของฟังก์ชันย่อย            | 25   |
| <b>บทที่ 4 ผลการทดลอง</b>                |      |
| 4.1 ส่วนติดต่อกับผู้ใช้ (User Interface) | 33   |
| 4.2 ขั้นตอนการใช้งานโปรแกรม              | 34   |
| <b>บทที่ 5 สรุปผลการทดลอง</b>            |      |
| 5.1 สรุปการพัฒนาโครงงาน                  | 42   |
| 5.2 ปัญหาที่พบในการทำโครงงาน             | 42   |
| 5.3 ข้อจำกัดของโครงงาน                   | 42   |
| 5.4 แนวทางในการพัฒนาต่อ                  | 43   |
| บรรณานุกรม                               | 44   |
| ภาคผนวก ก                                |      |
| ภาคผนวก ข                                |      |

## สารบัญรูปภาพ

| รูป  | หน้า |
|--|------|
| รูปที่ 1.1 การทำงานโดยรวมของโปรแกรม                            | 2    |
| รูปที่ 2.1 ตัวอย่างเส้นทางที่มีการระบุค่าระยะทาง               | 3    |
| รูปที่ 2.2 การเก็บเส้นทางเข้าไปในคิวในรอบที่ 1                 | 4    |
| รูปที่ 2.3 การเลือกเส้นทางในรอบที่ 2                           | 4    |
| รูปที่ 2.4 การเก็บเส้นทางเข้าไปในคิวในรอบที่ 2                 | 4    |
| รูปที่ 2.5 การเลือกเส้นทางในรอบที่ 3                           | 5    |
| รูปที่ 2.6 การเก็บเส้นทางเข้าไปในคิวในรอบที่ 3                 | 5    |
| รูปที่ 2.7 การเลือกเส้นทางในรอบที่ 4                           | 5    |
| รูปที่ 2.8 การเก็บเส้นทางเข้าไปในคิวในรอบที่ 4                 | 6    |
| รูปที่ 2.9 การเลือกเส้นทางในรอบที่ 5                           | 6    |
| รูปที่ 2.10 การเก็บเส้นทางเข้าไปในคิวในรอบที่ 5                | 6    |
| รูปที่ 2.11 การเลือกเส้นทางในรอบที่ 6                          | 7    |
| รูปที่ 2.12 การเก็บเส้นทางเข้าไปในคิวในรอบที่ 6                | 7    |
| รูปที่ 2.13 การเลือกเส้นทางในรอบที่ 7                          | 8    |
| รูปที่ 2.14 เส้นทางทั้งหมดที่โหนด A เป็นโหนดเริ่มต้น           | 8    |
| รูปที่ 2.15 สัญลักษณ์ของเอนทิตี                                | 10   |
| รูปที่ 2.16 สัญลักษณ์เลขของรหัสแผนก                            | 10   |
| รูปที่ 2.17 สัญลักษณ์ความสัมพันธ์แบบหนึ่งต่อหนึ่ง (one to one) | 10   |
| รูปที่ 2.18 ความสัมพันธ์แบบหนึ่งต่อหนึ่งแบบเจาะจง              | 10   |
| รูปที่ 2.19 ความสัมพันธ์แบบหนึ่งสิ่งต่อหลายสิ่ง (one to many)  | 11   |
| รูปที่ 2.20 ความสัมพันธ์แบบหลายสิ่งต่อหลายสิ่ง (many to many)  | 11   |
| รูปที่ 3.1 ตัวอย่างเส้นทางและสถานที่                           | 13   |
| รูปที่ 3.2 การนำเส้นทางเข้าสู่คิว                              | 13   |
| รูปที่ 3.3 เส้นทางที่ถูกเลือกในรอบที่ 2                        | 13   |
| รูปที่ 3.4 การนำเส้นทางเข้าสู่คิวในรอบที่ 2                    | 14   |
| รูปที่ 3.5 เส้นทางที่ถูกเลือกในรอบที่ 3                        | 14   |
| รูปที่ 3.6 การนำเส้นทางเข้าสู่คิวในรอบที่ 3                    | 14   |

ณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูปภาพ(ต่อ)

| รูป  | หน้า |
|--|------|
| รูปที่ 3.7 เส้นทางที่ถูกเลือกในรอบที่ 4              | 15   |
| รูปที่ 3.8 การนำเส้นทางเข้าสู่คิวในรอบที่ 4          | 15   |
| รูปที่ 3.9 เส้นทางที่ถูกเลือกในรอบที่ 5              | 15   |
| รูปที่ 3.10 การนำเส้นทางเข้าสู่คิวในรอบที่ 5         | 16   |
| รูปที่ 3.11 เส้นทางที่ถูกเลือกในรอบที่ 6             | 16   |
| รูปที่ 3.12 การนำเส้นทางเข้าสู่คิวในรอบที่ 6         | 16   |
| รูปที่ 3.13 เส้นทางที่ถูกเลือกในรอบที่ 7             | 17   |
| รูปที่ 3.14 การนำเส้นทางเข้าสู่คิวในรอบที่ 7         | 17   |
| รูปที่ 3.15 เส้นทางที่ถูกเลือกในรอบที่ 8             | 18   |
| รูปที่ 3.16 การนำเส้นทางเข้าสู่คิวในรอบที่ 8         | 18   |
| รูปที่ 3.17 เส้นทางที่ 1 DGHI(8)                     | 18   |
| รูปที่ 3.18 เส้นทางที่ 2 ABCF(11)                    | 18   |
| รูปที่ 3.19 เส้นทางที่ 3 DEF(11)                     | 19   |
| รูปที่ 3.20 แบบจำลองในแอม                            | 20   |
| รูปที่ 3.21 ผังงานของโปรแกรมหลักส่วนที่ 1            | 23   |
| รูปที่ 3.22 ผังงานของโปรแกรมหลักส่วนที่ 2            | 24   |
| รูปที่ 3.23 ผังงานของฟังก์ชัน checkplace             | 25   |
| รูปที่ 3.24 ผังงานของฟังก์ชัน checkroad              | 26   |
| รูปที่ 3.25 ผังงานของฟังก์ชัน findjunction ส่วนที่ 1 | 27   |
| รูปที่ 3.26 ผังงานของฟังก์ชัน findjunction ส่วนที่ 2 | 28   |
| รูปที่ 3.27 ผังงานของฟังก์ชัน findmarkednode         | 29   |
| รูปที่ 3.28 ผังงานของฟังก์ชัน cutcost                | 29   |
| รูปที่ 3.29 ผังงานของฟังก์ชัน findnextnode           | 30   |
| รูปที่ 3.30 ผังงานของฟังก์ชัน findcost               | 31   |
| รูปที่ 3.31 ผังงานของฟังก์ชัน draw                   | 32   |

## สารบัญรูปภาพ(ต่อ)

| รูป   | หน้า |
|---|------|
| รูปที่ 4.1 หน้าแรกของส่วนติดต่อกับผู้ใช้                                  | 33   |
| รูปที่ 4.2 ขั้นตอนการเลือกประเภทของสถานที่ต้นทาง                          | 34   |
| รูปที่ 4.3 ขั้นตอนการเลือกสถานที่ต้นทาง                                   | 34   |
| รูปที่ 4.4 ขั้นตอนการเลือกประเภทของสถานที่ปลายทาง                         | 35   |
| รูปที่ 4.5 ขั้นตอนการเลือกสถานที่ปลายทาง                                  | 35   |
| รูปที่ 4.6 ขั้นตอนการเลือกตัวแปร ในการคำนวณ                               | 36   |
| รูปที่ 4.7 ขั้นตอนการแสดงตัวเลือกเส้นทาง                                  | 36   |
| รูปที่ 4.8 ขั้นตอนการแจ้งเมื่อผู้ใช้เลือกสถานที่เดียวกัน                  | 37   |
| รูปที่ 4.9 ขั้นตอนการบอกระยะทางและแสดงแผนที่เมื่อสถานที่อยู่บนถนนเดียวกัน | 37   |
| รูปที่ 4.10 ขั้นตอนการแจ้งเมื่อผู้ใช้ไม่ได้เลือกตัวแปรที่ใช้คำนวณ         | 38   |
| รูปที่ 4.11 ขั้นตอนเมื่อผู้ใช้เลือกจุดแผนที่ของเส้นทางที่ 1               | 38   |
| รูปที่ 4.12 ขั้นตอนเมื่อผู้ใช้เลือกจุดแผนที่ของเส้นทางที่ 2               | 39   |
| รูปที่ 4.13 ขั้นตอนเมื่อผู้ใช้เลือกจุดแผนที่ของเส้นทางที่ 3               | 39   |
| รูปที่ 4.14 ขั้นตอนเมื่อผู้ใช้เลือกจุดแผนที่ขนาดจริงของเส้นทางที่ 1       | 40   |
| รูปที่ 4.15 ขั้นตอนเมื่อผู้ใช้เลือกจุดแผนที่ขนาดจริงของเส้นทางที่ 2       | 40   |
| รูปที่ 4.16 ขั้นตอนเมื่อผู้ใช้เลือกจุดแผนที่ขนาดจริงของเส้นทางที่ 3       | 41   |

๗

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญตาราง

| ตาราง   | หน้า |
|---|------|
| ตารางที่ 3.1 การแปลงแบบจำลองในแอมเป็นตารางสถานที่ | 20   |
| ตารางที่ 3.2 การแปลงแบบจำลองในแอมเป็นตารางถนน     | 20   |
| ตารางที่ 3.3 การแปลงแบบจำลองในแอมเป็นตารางแยก     | 20   |
| ตารางที่ 3.4 พจนานุกรมข้อมูลของตารางสถานที่       | 21   |
| ตารางที่ 3.5 พจนานุกรมข้อมูลของตารางถนน           | 21   |
| ตารางที่ 3.6 พจนานุกรมข้อมูลของตารางแยก           | 21   |



# บทที่ 1

## บทนำ

### 1.1 บทนำ

เนื่องจากปัจจุบันราคาน้ำมันมีราคาสูง การวางแผนการเดินทางจึงเป็นสิ่งสำคัญ ซึ่งช่วยลดค่าใช้จ่ายในส่วนนี้ลงไปได้อย่างมาก อีกทั้งยังประหยัดเวลาในการเดินทาง ดังนั้นโครงงานระบบบริการหาเส้นทางกรุงเทพฯ จึงเกิดขึ้นเพื่อให้บริการในการหาเส้นทางที่เหมาะสมที่สุดสามเส้นทางเพื่อเป็นตัวเลือกให้ผู้ใช้เลือกใช้เส้นทางในการเดินทางในแต่ละวัน โดยอ้างอิงจากสถานที่สำคัญเพื่อเป็นจุดค้นทางและปลายทางที่ต้องการจะไป และใช้อัลกอริทึม (Algorithm) ในการหาเส้นทางมาประยุกต์ใช้งานภายในโปรแกรม ซึ่งผู้ใช้สามารถใช้งานได้ผ่านทางอินเทอร์เน็ต (Internet) ในลักษณะที่เป็นเว็บแอปพลิเคชัน (Web Application) จึงทำให้สามารถใช้งานได้สะดวกและรวดเร็ว

### 1.2 วัตถุประสงค์

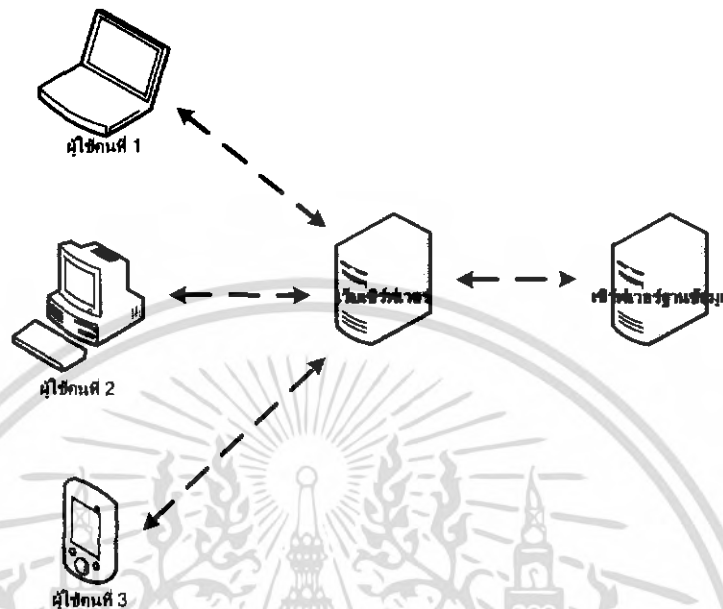
- เพื่อค้นหาเส้นทางที่เหมาะสมในการเดินทางในกรุงเทพมหานคร โดยพิจารณาการเดินทางระหว่างสถานที่สำคัญอ้างอิง 2 จุดเป็นจุดเริ่มต้นและจุดปลายทาง
- เพื่อนำอัลกอริทึมการหาเส้นทางสั้นที่สุดเอสพีเอฟ (Shortest Path First) มาประยุกต์ใช้กับโครงข่ายของเส้นทางการเดินทางจริงในกรุงเทพมหานคร
- เพื่อให้ผู้ใช้สามารถใช้บริการได้อย่างสะดวกและทั่วถึงผ่านทางอินเทอร์เน็ต โดยใช้โปรแกรมเว็บเบราว์เซอร์ซึ่งใช้งานง่ายไม่ซับซ้อน

### 1.3 ขอบเขตของโครงงาน

- ระบบนี้เป็นระบบที่ใช้หาเส้นทางที่เหมาะสมที่สุดระหว่างจุดสองจุดจำนวนสามเส้นทาง
- เป็นระบบที่เหมาะสมสำหรับผู้ใช้ที่ใช้งานบนส่วนบุคคล
- สถานที่อ้างอิงที่อยู่ในซอยจะพิจารณาว่าอยู่บนถนนใหญ่
- ระบบใช้ระบบปฏิบัติการวินโดวส์เซิร์ฟเวอร์ (Window Server)
- แสดงผลโดยใช้แผนที่จริง โดยใช้การวาดเส้นบนแผนที่เพื่อระบุเส้นทาง
- ฐานข้อมูลครอบคลุมพื้นที่กรุงเทพฯ ประมาณ 180 ตารางกิโลเมตร

## 1.4 ระบบการทำงานโดยรวม

- หลักการทำงานของโปรแกรมมีการรับข้อมูลจากผู้ใช้เพื่อประมวลผลที่เซิร์ฟเวอร์ (Server) ก่อนที่จะส่งผลลัพธ์ที่ได้มาแสดงผลที่หน้าเว็บซึ่งแสดงในรูปแบบที่ 1



รูปที่ 1.1 การทำงาน โดยรวมของโปรแกรม

- การประมวลผลข้อมูลนั้นเซิร์ฟเวอร์มีการติดต่อกับฐานข้อมูลเพื่อร้องขอข้อมูลต่าง ๆ ที่จำเป็นต้องใช้ในการประมวลผลหาเส้นทางโดยใช้อัลกอริทึมเอสพีเอฟ
- แสดงผลเส้นทาง 3 เส้นทางเพื่อเป็นตัวเลือกให้กับผู้ใช้บริการ

## 1.5 ขั้นตอนการดำเนินโครงการ

- ศึกษาวิธีการเขียนโปรแกรมโดยใช้เอเอสพีคอตเน็ต (ASP.NET)
- ศึกษาขั้นตอนการทำงานของอัลกอริทึมเอสพีเอฟ
- ออกแบบฐานข้อมูลเส้นทาง ถนน และแยกต่าง ๆ ในกรุงเทพฯ
- ออกแบบโปรแกรมโดยเริ่มจากการเขียนผังงาน (Flow Chart)
- ออกแบบส่วนติดต่อกับผู้ใช้ (User Interface)
- พัฒนาโปรแกรมโดยใช้เอเอสพีคอตเน็ตและเชื่อมต่อกับฐานข้อมูลไมโครซอฟท์แอคเซส (Microsoft Access)
- ทดสอบโปรแกรมเพื่อหาข้อผิดพลาดและทำการแก้ไข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

### ทฤษฎีและหลักการ

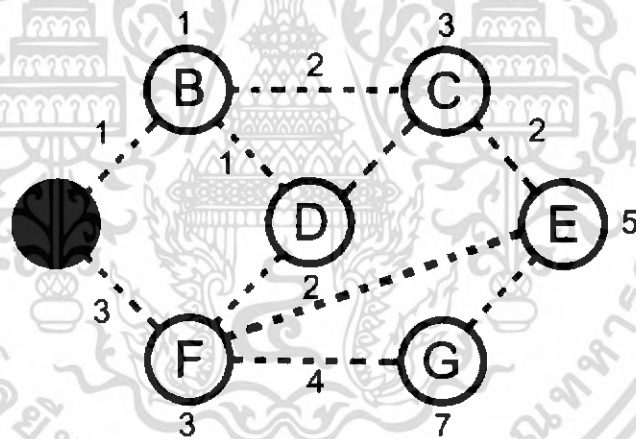
#### 2.1 อัลกอริทึมเอสพีเอฟ

เป็นอัลกอริทึมที่ใช้ในการหาเส้นทางในการส่งข้อมูลจากโหนด (Node) หนึ่ง ไปยังโหนด โหนดหนึ่งภายในเครือข่ายคอมพิวเตอร์ โดยแต่ละโหนดจะเชื่อมต่อกันผ่านทางลิงค์ (Link) ซึ่งแต่ละลิงค์มีค่าน้ำหนัก (Weight) เป็นค่าที่ใช้ตัดสินใจเลือกเส้นทาง โดยค่าน้ำหนักนี้จะหมายถึงค่า ระยะทาง หรือค่าความคับคั่งของข้อมูลก็ได้

ในการหาเส้นทางของอัลกอริทึมเอสพีเอฟนั้นจะใช้การพิจารณาค่าน้ำหนักของแต่ละลิงค์ และเลือกขยายเส้นทางไปในทางที่มีค่าน้ำหนักน้อยที่สุดแล้วทำการขยายเส้นทางย่อยไปเรื่อยๆ จนกว่าจะพบสถานที่ปลายทาง

##### 2.1.1 ตัวอย่างการทำงานของอัลกอริทึมเอสพีเอฟ

จากรูปที่ 2.1 ถ้าให้โหนดต้นทางเป็นโหนด A แล้วทำการหาเส้นทางที่สั้นที่สุดจากโหนด A ไปยังโหนดอื่นในเครือข่ายจะมีวิธีการดังนี้



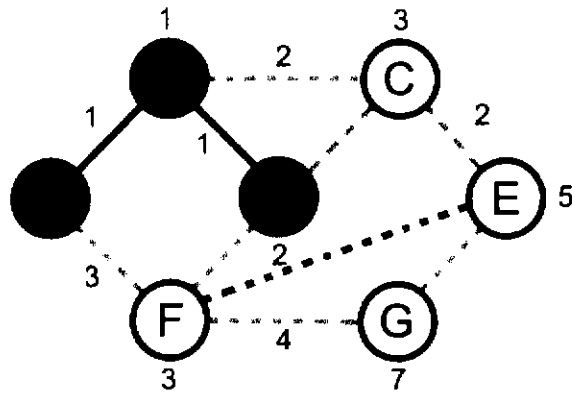
รูปที่ 2.1 ตัวอย่างเส้นทางที่มีการระบุค่าระยะทาง

#### รอบที่ 1

- เลือกโหนดเริ่มต้นคือโหนด A มาเป็นโหนดที่พิจารณาหาเส้นทาง โดยจะต้องไม่มีการวนเส้นทางกลับมาที่โหนดที่พิจารณาอีกเป็นครั้งที่สอง เรียกว่าการมาร์ค (Mark)
- เมื่อพิจารณาจากโหนด A หาโหนดที่มีลิงค์เชื่อมต่อโดยตรงกับโหนด A จะได้แก่ B และ F ที่มีระยะทาง 1 และ 3 ตามลำดับ
- เก็บเส้นทางที่ได้พร้อมทั้งค่าระยะทางรวมของเส้นทางลงในทิวจะ ได้ดังรูปที่ 2.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





รูปที่ 2.5 การเลือกเส้นทางในรอบที่ 3

- เมื่อพิจารณาจากโหนด D หาโหนดที่มีลิงค์เชื่อมต่อกับโดยตรงกับโหนด D จะได้แก่ C และ F ที่มีระยะทาง 3 และ 5 ตามลำดับ

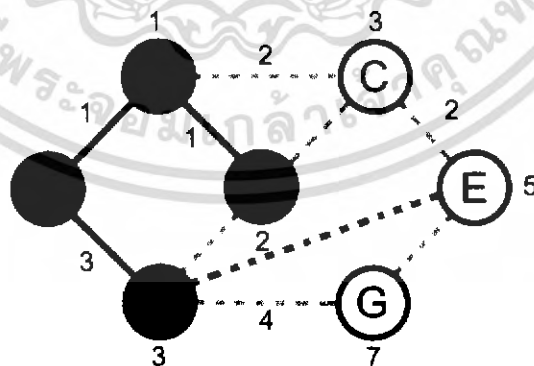
- ลบเส้นทางที่ถูกเลือกคือ ABD(2) ออกจากคิว
- เก็บเส้นทางที่ได้พร้อมทั้งค่าระยะทางรวมของเส้นทางลงในคิว
- และต้องทำการลบเส้นทางที่มีปลายทางเดียวกันแต่มีระยะทางมากกว่าออกจากคิว ดังนั้นจะได้ดังรูปที่ 2.6

|       |        |
|-------|--------|
| AF(3) | ABC(3) |
|-------|--------|

รูปที่ 2.6 การเก็บเส้นทางเข้าไปในคิวในรอบที่ 3

รอบที่ 4

- ทำการเลือกโหนดที่จะทำการมาร์คจากในคิวโดยดูจากค่าระยะทางของเส้นทางที่สั้นที่สุดจะได้เส้นทาง AF ดังรูปที่ 2.7 และจะได้โหนดที่จะมาร์คคือโหนด F



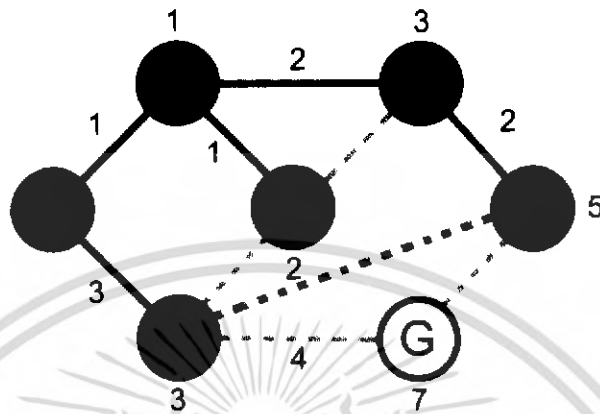
รูปที่ 2.7 การเลือกเส้นทางในรอบที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



### รอบที่ 6

- ทำการเลือกโหนดที่จะทำการมาร์คจากในคิว โดยดูจากค่าระยะทางของเส้นทางที่สั้นที่สุด จะได้เส้นทาง ABCE ดังรูปที่ 2.11 และจะได้โหนดที่จะมาร์คคือ โหนด E



รูปที่ 2.11 การเลือกเส้นทางในรอบที่ 6

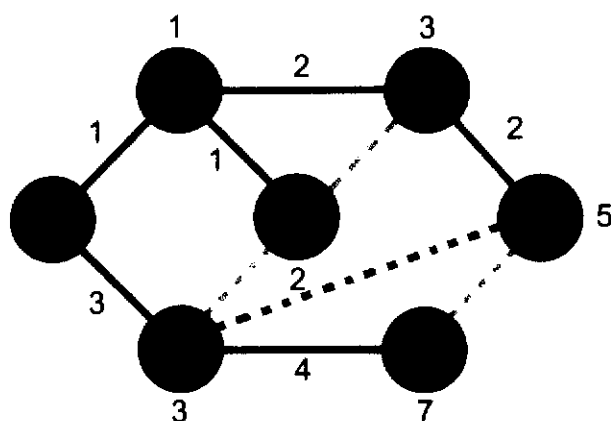
- เมื่อพิจารณาจากโหนด E หาโหนดที่มีลิงค์เชื่อมต่อโดยตรงกับโหนด E จะได้แค่ G ที่มีระยะทาง 5
- ลบเส้นทางที่ถูกเลือกคือ ABCE(5) ออกจากคิว
- เก็บเส้นทางที่ได้พร้อมทั้งค่าระยะทางรวมของเส้นทางลงในคิว
- และต้องทำการลบเส้นทางที่มีปลายทางเดียวกันแต่มีระยะทางมากกว่าออกจากคิว ดังนั้นจะได้ดังรูปที่ 2.12

AFG(7)

รูปที่ 2.12 การเก็บเส้นทางเข้าไปในคิวในรอบที่ 6

### รอบที่ 7

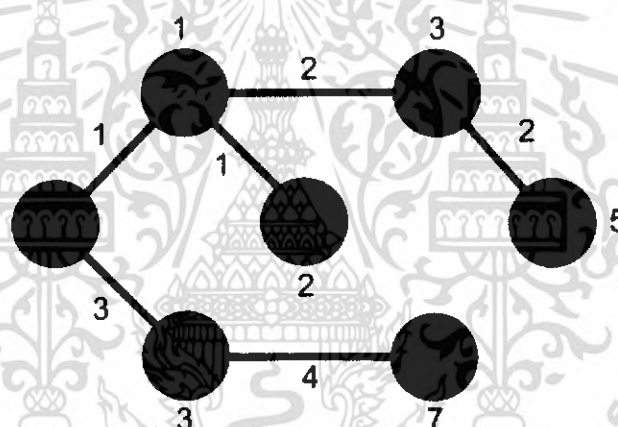
- ทำการเลือกโหนดที่จะทำการมาร์คจากในคิว โดยดูจากค่าระยะทางของเส้นทางที่สั้นที่สุด จะได้เส้นทาง AFG ดังรูปที่ 2.13 และจะได้โหนดที่จะมาร์คคือ โหนด G



รูปที่ 2.13 การเลือกเส้นทางในรอบที่ 7

- ลบเส้นทางที่ถูกเลือกคือ ABCE(5) ออกจากคิว
- พบว่ามีเส้นทางจากโหนด A ถึงทุกโหนดแล้วดังนั้นจึงจบการทำงานของอัลกอริทึมจะได้

ผังรูปที่ 2.14



รูปที่ 2.14 เส้นทางทั้งหมดที่โหนด A เป็น โหนดเริ่มต้น

## 2.2 การออกแบบฐานข้อมูลโดยใช้แบบจำลองไนแอม (NIAM)

NIAM (Nijssen's Information Analysis Methodology) เป็นวิธีการในการออกแบบฐานข้อมูลโดยการแสดงความสัมพันธ์ต่าง ๆ ของข้อมูล มีพื้นฐานมาจากภาษาธรรมชาติ คือมีรูปแบบประโยคเป็นแบบประธาน กริยา กรรม นอกจากนั้นยังสามารถแปลงโครงสร้างฐานข้อมูลเชิงสัมพันธ์ ดังนั้นจึงสะดวกในการออกแบบฐานข้อมูลของระบบงานที่มีขนาดใหญ่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.2.1 ขั้นตอนในการออกแบบในแอม

- กำหนดขอบเขตของงานและสิ่งที่เกิดขึ้นจริงภายในขอบเขตงาน
- วาดไดอะแกรมเป็นโครงสร้างทางแนวคิดโดยคร่าว ๆ จากสิ่งที่เกิดขึ้นจริงจากขอบเขตของงาน
- เติมสัญลักษณ์แสดงยูนิค คอนสเตรน (Unique Constrain)
- ตรวจสอบความถูกต้องของชนิดความจริง
- เติมสัญลักษณ์แสดงเลขคี่คอล (Lexical) แมนคาทอรีโรลและซับไทป์คอนสเตรน (Mandatory Role and Subtype Constrain)
- ตรวจสอบตัวบ่งชี้เฉพาะ (Unique identifier) ของแต่ละชนิดเอนทิตี (Entity)
- เติมสัญลักษณ์แสดงอิวอลิตี้คอนสเตรน (Equality Constrain) เอ็กชวลูชันคอนสเตรน (Exclusion Constrain) และซับเซตคอนสเตรน (Subset Constrain)
- ตรวจสอบความสมบูรณ์ของโครงสร้างทางแนวคิดที่ได้ออกแบบว่าสอดคล้องกับตัวอย่างข้อมูลและไม่มีความซ้ำซ้อนของข้อมูล

### 2.2.2 ส่วนประกอบพื้นฐานของในแอม

- ชนิดเอนทิตี (Entity type) หมายถึง เซตของสิ่งเรามาสนใจ ทั้งที่อยู่ในรูปของสิ่งที่จับต้องได้ และจับต้องไม่ได้ เช่นคน บริษัท ฯลฯ
- ชนิดลาเบล (Label type) หมายถึง เซตของสิ่งที่ใช้บ่งบอกความแตกต่าง หรือชื่อของแต่ละเอนทิตีที่กำหนด เช่น ชื่อ นามสกุล
- โรล (Role) หมายถึง ความสัมพันธ์ที่เกี่ยวข้องกับเอนทิตีที่เชื่อมกันอยู่เช่น พนักงานชื่อนี้ทำงานอยู่ที่บริษัทนี้
- เอลเมนต์แฟคไทป์ (Element fact type) หมายถึง เซตความสัมพันธ์ระหว่างสมาชิกของชนิดเอนทิตีตั้งแต่ 2 ชนิดขึ้นไป โดยที่ชนิดความจริงที่มี 2 ความสัมพันธ์จะเรียกว่า ไบนารีแฟคไทป์ (Binary fact type) ส่วนชนิดความจริงที่มี 3 ความสัมพันธ์จะเรียกว่า เทอนารีแฟคไทป์ (Ternary fact type) สำหรับชนิดความจริงที่มีมากกว่า 3 สัมพันธ์ขึ้นไปจะเรียกว่า เอน-นารีแฟคไทป์ (N-ary fact type)
- เรฟเฟอเรนซ์ ไทป์ (Reference type) หมายถึง เซตของความสัมพันธ์ระหว่างสมาชิกของชนิดเอนทิตีกับสมาชิกของชนิดลาเบลที่มีอยู่
- เนสต์แฟคไทป์ (Nested fact type) หมายถึง ชนิดเอนทิตีที่แสดงความสัมพันธ์ในการกำหนดกลุ่มของชนิดความจริงที่มีตั้งแต่ 2 ความสัมพันธ์ขึ้นไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- อินทิกริตีคอนสเตรน (Integrity constraints) หมายถึงสิ่งที่ใช้แสดงกฎที่ใช้ในการบังคับควบคุมความถูกต้องของข้อมูล

### 2.2.3 สัญลักษณ์ของส่วนประกอบพื้นฐานของแบบจำลองในแอม



รูปที่ 2.15 สัญลักษณ์ของเอนทิตี

อธิบายรูปที่ 2.15 หมายถึง แผนกเป็นเอนทิตีหนึ่ง ๆ ซึ่งอาจจะมีส่วนประกอบย่อย ๆ อีก



รูปที่ 2.16 สัญลักษณ์ตัวเลขของรหัสแผนก

อธิบายรูปที่ 2.16 หมายถึง รหัสแผนกเป็นส่วนประกอบย่อยของแผนก



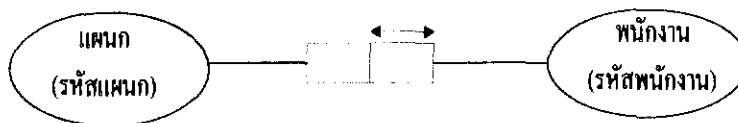
รูปที่ 2.17 สัญลักษณ์ความสัมพันธ์แบบหนึ่งต่อหนึ่ง (one to one)

อธิบายรูปที่ 2.17 หมายถึงรหัสพนักงานหนึ่งรหัสจะเป็นพนักงานได้หนึ่งคน และพนักงานหนึ่งคนจะมีรหัสพนักงานได้เพียงหนึ่งรหัส



รูปที่ 2.18 ความสัมพันธ์แบบหนึ่งต่อหนึ่งแบบเจาะจง

อธิบายรูปที่ 2.18 หมายถึง เป็นความสัมพันธ์เหมือนรูปที่ 2.4



รูปที่ 2.19 ความสัมพันธ์แบบหนึ่งถึงต่อหลายสิ่ง (one to many)

อธิบายรูปที่ 2.19 หมายถึง แผนกหนึ่งแผนกจะมีพนักงานทำงานได้หลายคน แต่พนักงานหนึ่งคนสามารถทำงานได้เพียงหนึ่งแผนก



รูปที่ 2.20 ความสัมพันธ์แบบหลายสิ่งต่อหลายสิ่ง (many to many)

อธิบายรูปที่ 2.20 หมายถึง ที่อยู่หนึ่งที่อยู่สามารถมีพนักงานอยู่ได้หลายคน และพนักงานหนึ่งคนสามารถมีที่อยู่ได้หลายที่อยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

### หลักการออกแบบและการดำเนินการ

#### 3.1 การออกแบบอัลกอริทึมที่ใช้ในโครงการ

##### 3.1.1 การนำอัลกอริทึมเอสพีเอฟมาประยุกต์ใช้

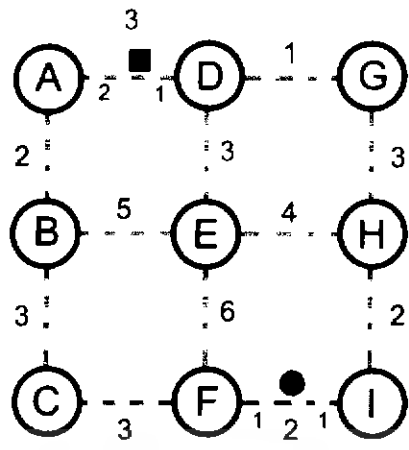
อัลกอริทึมเอสพีเอฟ เป็นอัลกอริทึมในการหาเส้นทางที่สั้นที่สุดในการส่งข้อมูลภายในเครือข่ายคอมพิวเตอร์ (Network Routing Algorithm) วิธีหนึ่ง โดยในการหาเส้นทางที่สั้นที่สุดจากโหนดต้นทางไปยังโหนดปลายทางแบบหนึ่งต่อหนึ่ง (One - to - One) นั้นจะเลือกใช้การเลือกลิงค์ (Link) ย่อยที่มีค่าน้ำหนัก (Weight) ที่น้อยที่สุด แล้วทำการแตกลิงค์ย่อยไปเรื่อย ๆ จนกว่าจะไปถึงโหนดปลายทาง

โครงการนี้ได้มีการดัดแปลงอัลกอริทึมเอสพีเอฟ ให้สามารถทำงานได้ตามความต้องการของโครงการ เนื่องจากโครงการนี้ใช้สถานที่สำคัญต่าง ๆ เป็นจุดต้นทางและจุดปลายทาง แยก (Junction) ไม่ว่าจะเป็นสามแยกหรือสี่แยกก็ตามจะเป็นโหนด และส่วนของถนน (Road Section) จะหมายถึงลิงค์ ทำให้สถานที่ต้นทางและปลายทางจะมีแยกที่นำมาคำนวณอย่างน้อย 2 แยก ดังนั้นโปรแกรมจึงต้องสามารถหาเส้นทางที่เหมาะสมที่สุดจากโหนดเริ่มต้นหลายโหนด (ตั้งแต่ 2 โหนดขึ้นไป) ไปยังโหนดปลายทางหลายโหนด ในอัลกอริทึมเอสพีเอฟ ดังนั้นโหนดต้นทางและปลายทางก็คือแยกแต่ละแยกของถนนที่สถานที่ต้นทางและปลายทางนั้นตั้งอยู่ ดังนั้นจะทำให้โหนดต้นทางและปลายทางมีอย่างน้อย 2 โหนดในกรณีที่สถานที่นั้นติดกับถนนเพียงถนนเดียว แต่ถ้าสถานที่นั้นติดกับถนนมากกว่า 1 ถนน คือกรณีที่ตั้งอยู่ที่มุมถนน เป็นต้น ก็จะทำให้ได้โหนดมากกว่า 2 โหนด ดังนั้นในการหาเส้นทางที่สั้นที่สุดจึงต้องคำนึงถึงระยะทางจากสถานที่ถึงแยกแต่ละแยกทั้งต้นทางและปลายทางด้วย

นอกจากการเพิ่มเติมอัลกอริทึมให้สามารถรองรับการหาเส้นทางจากต้นทางหลายโหนดไปยังปลายทางหลายโหนดแล้ว ยังต้องทำให้สามารถหาเส้นทางที่เหมาะสมที่สุดได้หลายเส้นทางอีกด้วย เนื่องจากในโครงการนี้ต้องการที่จะให้มีทางเลือกให้กับผู้ใช้ได้เลือกหลายเส้นทาง ดังนั้นจะสรุปได้ว่าอัลกอริทึมจะทำการหาเส้นทางโดยการแตกเส้นทางย่อยที่มีค่าน้ำหนักน้อยที่สุดไปเรื่อย ๆ จนกว่าจะเจอแยกปลายทางแยกใดแยกหนึ่งและได้เส้นทางครบจำนวนตามที่ต้องการนั่นเอง

##### 3.1.2 ตัวอย่างการทำงานของอัลกอริทึม

จากรูปที่ 3.1 จะสมมติว่าสถานที่ต้นทางอยู่บนถนนที่มีแยกสองด้านคือ A กับ D และสถานที่ปลายทางอยู่บนถนนที่มีแยกทั้งสองด้านคือ F และ I ดังนั้นสามารถหาเส้นทางที่สั้นที่สุด 3 เส้นทางได้ตามขั้นตอนดังนี้

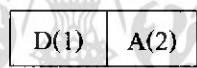


รูปที่ 3.1 ตัวอย่างเส้นทางและสถานที่

■ หมายถึง สถานที่ต้นทาง และ ● หมายถึง สถานที่ปลายทาง

รอบที่ 1

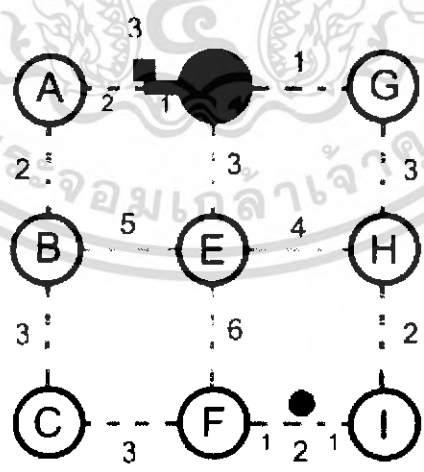
- เก็บแยกต้นทางทั้งสองแยกพร้อมกับระบุระยะทางจากสถานที่ต้นทางกับแยกไว้ในวงเล็บเข้าไปในคิว (queue) โดยเรียงลำดับระยะทางจากน้อยไปมาก จะได้ผลดังรูปที่ 3.2



รูปที่ 3.2 การนำเส้นทางเข้าสู่คิว

รอบที่ 2

- เลือกเส้นทางในคิวที่มีค่าระยะทางน้อยที่สุดมาพิจารณา คือ D(1)



รูปที่ 3.3 เส้นทางที่ถูกเลือกในรอบที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

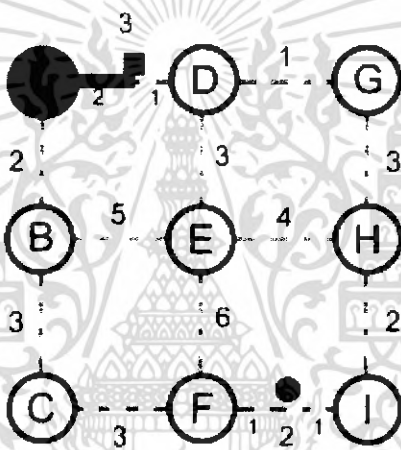
- จาก D(1) ให้แยกสุดท้ายที่อยู่ในเส้นทาง ซึ่งในกรณีนี้คือ D มาหาแยกถัดไป จะได้ E, G โดย A ถือเป็นแยกคั่นทาง จะไม่นำมาพิจารณา
- เพิ่มแยกถัดไปเข้าไปในเส้นทางที่เลือกและรวมค่าระยะทางเดิมกับความยาวของถนนจากแยกที่พิจารณา (D) ถึงแยกข้างเคียงจะได้ระยะทางของเส้นทางใหม่ แล้วนำเส้นทางใหม่เก็บในคิว
- ลบเส้นทางที่เลือก คือ D(1) ออกจากคิว จะได้ผลดังรูปที่ 3.4

|      |       |       |
|------|-------|-------|
| A(2) | DG(2) | DE(4) |
|------|-------|-------|

รูปที่ 3.4 การนำเส้นทางเข้าสู่คิวในรอบที่ 2

รอบที่ 3

- เลือกเส้นทางในคิวที่มีค่าระยะทางน้อยที่สุดมาพิจารณา คือ A(2)



รูปที่ 3.5 เส้นทางที่ถูกเลือกในรอบที่ 3

- จาก A(2) ให้แยกสุดท้ายที่อยู่ในเส้นทาง ซึ่งในกรณีนี้คือ A มาหาแยกถัดไป จะได้ B โดย D ถือเป็นแยกคั่นทาง จะไม่นำมาพิจารณา
- เพิ่มแยกถัดไปเข้าไปในเส้นทางที่เลือกและรวมค่าระยะทางเดิมกับความยาวของถนนจากแยกที่พิจารณา (A) ถึงแยกข้างเคียงจะได้ระยะทางของเส้นทางใหม่ แล้วนำเส้นทางใหม่เก็บในคิว
- ลบเส้นทางที่เลือก คือ A(2) ออกจากคิว จะได้ผลดังรูปที่ 3.6

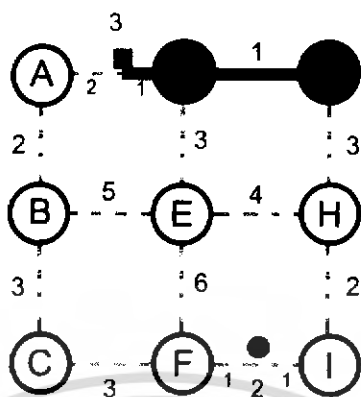
|       |       |       |
|-------|-------|-------|
| DG(2) | DE(4) | AB(4) |
|-------|-------|-------|

รูปที่ 3.6 การนำเส้นทางเข้าสู่คิวในรอบที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## รอบที่ 4

- เลือกเส้นทางในคิวที่มีค่าระยะทางน้อยที่สุดมาพิจารณา คือ DG(2)



รูปที่ 3.7 เส้นทางที่ถูกเลือกในรอบที่ 4

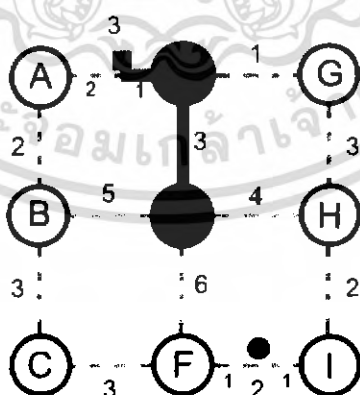
- จาก DG(2) ให้แยกสุดท้ายที่อยู่ในเส้นทาง ซึ่งในกรณีนี้คือ G มาหาแยกถัดไป จะได้ H
- เพิ่มแยกถัดไปเข้าไปในเส้นทางที่เลือกและรวมค่าระยะทางเดิมกับความยาวของถนนจากแยกที่พิจารณา (G) ถึงแยกข้างเคียงจะได้ระยะทางของเส้นทางใหม่ แล้วนำเส้นทางใหม่เก็บในคิว
- ลบเส้นทางที่เลือก คือ DG(2) ออกจากคิว จะได้ผลดังรูปที่ 3.8

|       |       |        |
|-------|-------|--------|
| DE(4) | AB(4) | DGH(5) |
|-------|-------|--------|

รูปที่ 3.8 การนำเส้นทางเข้าสู่คิวในรอบที่ 4

## รอบที่ 5

- เลือกเส้นทางในคิวที่มีค่าระยะทางน้อยที่สุดมาพิจารณา คือ DE(4)



รูปที่ 3.9 เส้นทางที่ถูกเลือกในรอบที่ 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

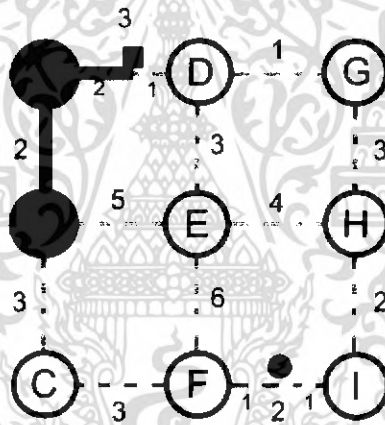
- จาก DE(4) ให้แยกสุดท้ายที่อยู่ในเส้นทาง ซึ่งในกรณีนี้คือ E มาหาแยกถัดไป จะได้ B, F และ H
- เพิ่มแยกถัดไปเข้าไปในเส้นทางที่เลือกและรวมค่าระยะทางเดิมกับความยาวของถนนจากแยกที่พิจารณา (E) ถึงแยกข้างเคียงจะได้ระยะทางของเส้นทางใหม่ แล้วนำเส้นทางใหม่เก็บในคิว
- จะพบว่า F คือแยกปลายทาง ดังนั้น เส้นทาง DEF(10) เป็นเส้นทางที่ต้องการเส้นทางที่ 1 ไม่ต้องเก็บในคิว
- ลบเส้นทางที่เลือก คือ DE(4) ออกจากคิว จะได้ผลดังรูปที่ 3.10

|       |        |        |        |
|-------|--------|--------|--------|
| AB(4) | DGH(5) | DEH(8) | DEB(9) |
|-------|--------|--------|--------|

รูปที่ 3.10 การนำเส้นทางเข้าสู่คิวในรอบที่ 5

รอบที่ 6

- เลือกเส้นทางในคิวที่มีค่าระยะทางน้อยที่สุดมาพิจารณา คือ AB(4)



รูปที่ 3.11 เส้นทางที่ถูกเลือกในรอบที่ 6

- จาก AB(4) ให้แยกสุดท้ายที่อยู่ในเส้นทาง ซึ่งในกรณีนี้คือ B มาหาแยกถัดไป จะได้ C, E
- เพิ่มแยกถัดไปเข้าไปในเส้นทางที่เลือกและรวมค่าระยะทางเดิมกับความยาวของถนนจากแยกที่พิจารณา (B) ถึงแยกข้างเคียงจะได้ระยะทางของเส้นทางใหม่ แล้วนำเส้นทางใหม่เก็บในคิว
- ลบเส้นทางที่เลือก คือ AB(4) ออกจากคิว จะได้ผลดังรูปที่ 3.12

|        |        |        |        |        |
|--------|--------|--------|--------|--------|
| DGH(5) | ABC(7) | DEH(8) | DEB(9) | ABE(9) |
|--------|--------|--------|--------|--------|

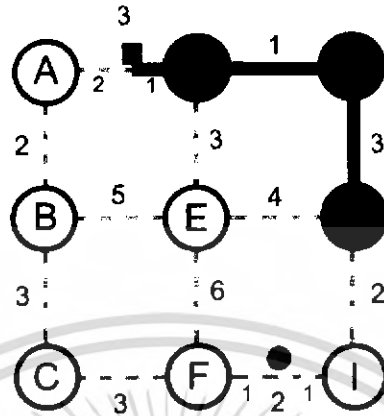
รูปที่ 3.12 การนำเส้นทางเข้าสู่คิวในรอบที่ 6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

### รอบที่ 7

- เลือกเส้นทางในคircuit ที่มีค่าระยะทางน้อยที่สุดมาพิจารณา คือ DGH(5)



รูปที่ 3.13 เส้นทางที่ถูกเลือกในรอบที่ 7

- จาก DGH(5) ให้แยกสุดท้ายที่อยู่ในเส้นทาง ซึ่งในกรณีนี้คือ H มาหาแยกถัดไป จะได้ E และ I
- เพิ่มแยกถัดไปเข้าไปในเส้นทางที่เลือกและรวมค่าระยะทางเดิมกับความยาวของถนนจากแยกที่พิจารณา (E) ถึงแยกข้างเคียงจะได้ระยะทางของเส้นทางใหม่ แล้วนำเส้นทางใหม่เก็บในคircuit
- จะพบว่า I คือแยกปลายทาง ดังนั้น เส้นทาง DGHI(7) เป็นเส้นทางที่ต้องการเส้นทางที่ 2 ไม่ต้องเก็บในคircuit
- ลบเส้นทางที่เลือก คือ DGH(5) ออกจากคircuit จะได้ผลดังรูปที่ 3.14

|        |        |        |        |         |
|--------|--------|--------|--------|---------|
| ABC(7) | DEH(8) | DEB(9) | ABE(9) | DGHE(9) |
|--------|--------|--------|--------|---------|

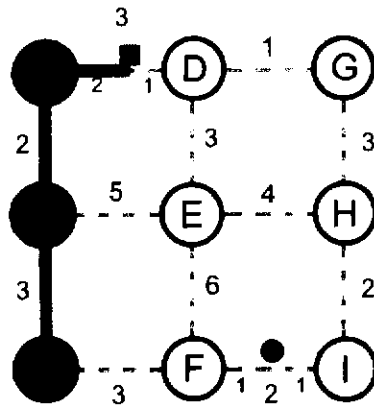
รูปที่ 3.14 การนำเส้นทางเข้าสู่คircuit ในรอบที่ 7

### รอบที่ 8

- เลือกเส้นทางในคircuit ที่มีค่าระยะทางน้อยที่สุดมาพิจารณา คือ ABC(7)

72724

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.15 เส้นทางที่ถูกเลือกในรอบที่ 8

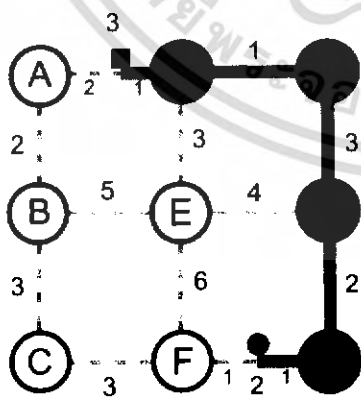
- จาก ABC(7) ให้แยกสุดท้ายที่อยู่ในเส้นทาง ซึ่งในกรณีนี้คือ C มาหาแยกถัดไป จะได้ F
- เพิ่มแยกถัดไปเข้าไปในเส้นทางที่เลือกและรวมค่าระยะทางเดิมกับความยาวของถนนจากแยกที่พิจารณา (E) ถึงแยกข้างเคียงจะได้ระยะทางของเส้นทางใหม่
- จะพบว่า F คือแยกปลายทาง ดังนั้น เส้นทาง ABCF(10) เป็นเส้นทางที่ต้องการเส้นทางที่ 3 ไม่ต้องเก็บในคิว
- ลบเส้นทางที่เลือก คือ ABC(7) ออกจากคิว จะได้ผลดังรูปที่ 3.16

|        |        |        |         |
|--------|--------|--------|---------|
| DEH(8) | DEB(9) | ABE(9) | DGHE(9) |
|--------|--------|--------|---------|

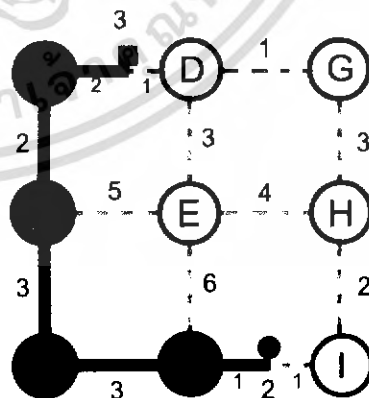
รูปที่ 3.16 การนำเส้นทางเข้าสู่คิวในรอบที่ 8

รอบที่ 9

- เมื่อได้เส้นทางครบ 3 เส้นทาง ให้นำระยะทางจากแยกปลายทางถึงสถานที่ปลายทางรวมกับระยะทางของเส้นทางแต่ละเส้นทางด้วย เพราะฉะนั้นจะได้เส้นทางทั้ง 3 เส้นทางดังนี้

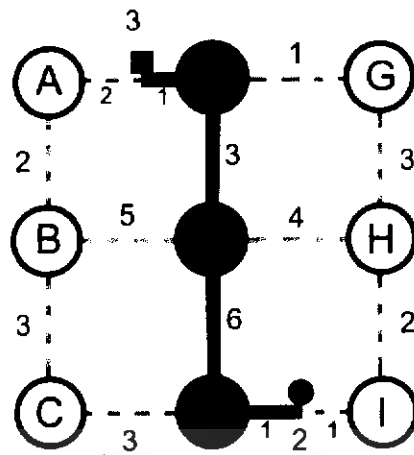


รูปที่ 3.17 เส้นทางที่ 1 DGHI(8)



รูปที่ 3.18 เส้นทางที่ 2 ABCF(11)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.19 เส้นทางที่ 3 DEF(11)

### 3.2 การออกแบบฐานข้อมูลโดยใช้แบบจำลองในแอม

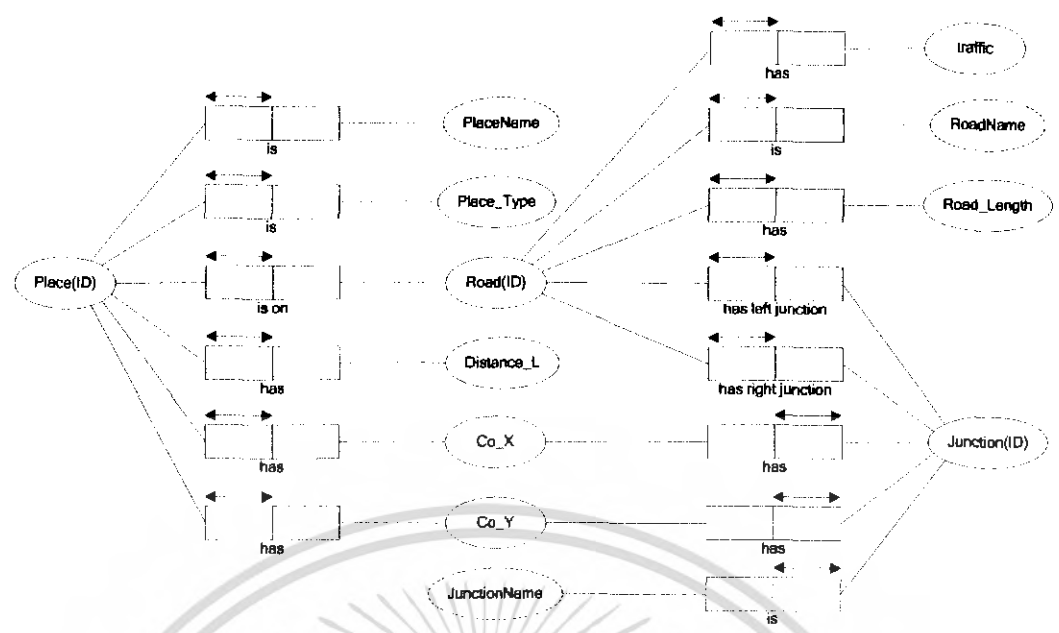
#### 3.2.1 แนวทางในการออกแบบฐานข้อมูล

การออกแบบฐานข้อมูล จะต้องคำนึงถึงข้อมูลที่ต้องนำไปใช้ในอัลกอริทึมเป็นสำคัญ และจากอัลกอริทึมที่ได้ออกแบบไปแล้ว ทำให้ทราบว่าในฐานข้อมูลจะต้องมีข้อมูลดังต่อไปนี้

- แยก ใช้เป็น โหนดในการหาเส้นทาง
  - ความยาวของถนน ใช้เป็นค่าน้ำหนักในการตัดสินใจเลือกเส้นทาง
  - ระยะทางจากสถานที่ถึงแยกทางซ้ายมือของถนน ดูจากทิศในแผนที่ โดยแยกที่อยู่ทางทิศเหนือและทิศตะวันตกของถนนจะถือว่าเป็นแยกทางซ้าย
  - ระยะทางจากสถานที่ถึงแยกทางขวามือของถนน สามารถหาได้จากการนำเอาความยาวถนนลบด้วยระยะทางจากสถานที่ถึงแยกทางซ้ายมือ
  - สภาพจราจร ได้จากการสมมติค่าเป็นตัวเลขตั้งแต่ 1 ถึง 9
  - พิกัดในแกนตั้งและแกนนอน วัดมาจากแผนที่จริงในหน่วยของพิกเซล (Pixel)
- ข้อมูลข้างต้นจะถูกนำไปประกอบกับข้อมูลอื่น เพื่อสร้างเป็นฐานข้อมูลต่อไป

#### 3.2.2 ออกแบบฐานข้อมูล

จากการวิเคราะห์ข้อมูลที่เป็นในฐานข้อมูล ทำให้สามารถออกแบบฐานข้อมูลโดยใช้แบบจำลองในแอมได้ดังนี้



รูปที่ 3.20 แบบจำลองในแอม

3.2.3 การแปลงแบบจำลองในแอม เป็นตารางความสัมพันธ์ (Relational Table)

จากแบบจำลองในแอมที่ได้ สามารถแปลงเป็นตารางความสัมพันธ์ได้ดังนี้

ตารางที่ 3.1 การแปลงแบบจำลองในแอมเป็นตารางสถานที่

| Place_ID | Place_Name | Road_ID | Distance_L | Place_Type | Co_X | Co_Y |
|----------|------------|---------|------------|------------|------|------|
|----------|------------|---------|------------|------------|------|------|

ตารางที่ 3.2 การแปลงแบบจำลองในแอมเป็นตารางถนน

| Road_ID | Left_Junc | Right_Junc | Road_Length | Traffic |
|---------|-----------|------------|-------------|---------|
|---------|-----------|------------|-------------|---------|

ตารางที่ 3.3 การแปลงแบบจำลองในแอมเป็นตารางแยก

| Junc_ID | Junc_Name | Co_X | Co_Y |
|---------|-----------|------|------|
|---------|-----------|------|------|

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.4 พจนานุกรมข้อมูล (Data Dictionary)

ตารางที่ 3.4 พจนานุกรมข้อมูลของตารางสถานที่

| ชื่อ       | ชนิด    | ขนาด | หมายเหตุ                  |
|------------|---------|------|---------------------------|
| Place_ID   | INT     | 5    | รหัสสถานที่               |
| Place_Name | VARCHAR | 30   | ชื่อสถานที่               |
| Road_ID    | VARCHAR | 5    | รหัสถนน                   |
| Distance_L | INT     | 10   | ระยะทางถึงแยกทางซ้ายและบน |
| Place_Type | VARCHAR | 30   | ชนิดสถานที่               |
| Co_X       | INT     | 4    | ตำแหน่งบนแผนที่ในแนวแกน x |
| Co_Y       | INT     | 4    | ตำแหน่งบนแผนที่ในแนวแกน y |

ตารางที่ 3.5 พจนานุกรมข้อมูลของตารางถนน

| ชื่อ        | ชนิด    | ขนาด | หมายเหตุ           |
|-------------|---------|------|--------------------|
| Road_ID     | VARCHAR | 5    | รหัสถนน            |
| Left_Junc   | VARCHAR | 5    | แยกทางซ้าย         |
| Right_Junc  | VARCHAR | 5    | แยกทางขวา          |
| Road_length | INT     | 10   | ความยาวของถนน      |
| Traffic     | INT     | 1    | สภาพการจราจรของถนน |

ตารางที่ 3.6 พจนานุกรมข้อมูลของตารางแยก

| ชื่อ      | ชนิด    | ขนาด | หมายเหตุ                  |
|-----------|---------|------|---------------------------|
| Junc_ID   | VARCHAR | 5    | รหัสแยก                   |
| Junc_Name | VARCHAR | 30   | ชื่อแยก                   |
| Co_X      | INT     | 4    | ตำแหน่งบนแผนที่ในแนวแกน x |
| Co_Y      | INT     | 4    | ตำแหน่งบนแผนที่ในแนวแกน y |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3 การออกแบบผังงานของโปรแกรม

#### 3.3.1 แนวทางในการออกแบบโปรแกรม

ในการออกแบบโปรแกรมให้ใช้อัลกอริทึมเอสพีเอฟ ที่ได้กล่าวไว้ในหัวข้อ 3.1 จะต้องมีการสร้างตัวแปรที่เก็บข้อมูลของเส้นทาง นั่นคือกิวนั่นเอง โดยภาษาเอสพีเอฟมีชนิดตัวแปรแบบอาร์เรย์ลิส (Arraylist) ซึ่งมีการใช้งานที่ง่ายโดยสามารถใส่ข้อมูลเข้าไปได้เรื่อย ๆ ทำให้อาร์เรย์ลิสมีความยาวหรือจำนวนข้อมูลไม่แน่นอน และสามารถใช้กับชนิดข้อมูลได้ทุกรูปแบบ จึงเป็นเหตุผลให้อาร์เรย์ลิสเหมาะสมที่จะนำมาออกแบบเป็นกิวในการเก็บข้อมูลเส้นทางและระยะทาง

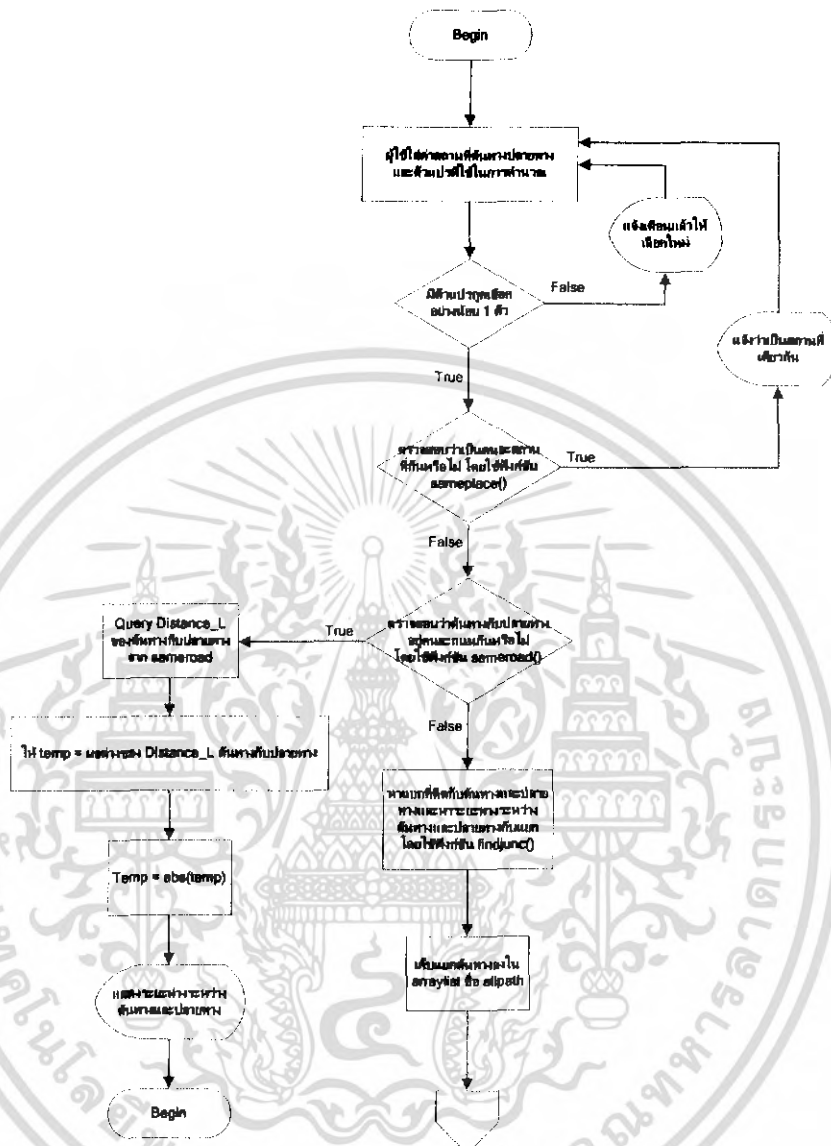
เส้นทางและระยะทางที่กล่าวข้างต้นนั้นจะใช้ตัวแปรชนิดข้อความ (String) มาเป็นตัวแปรที่เก็บข้อมูล โดยจะให้ระยะทางอยู่ข้างหน้าสุดและตามด้วยแยกต่าง ๆ ที่อยู่ในเส้นทางนั้น ๆ เช่น 1234JxxxxJyyyy เป็นต้น เนื่องจากสตริงนั้นมีฟังก์ชันในการจัดการกับข้อมูลที่หลากหลายทำให้สามารถแยกเส้นทางดังกล่าวออกเป็นระยะทางกับตัวเส้นทางได้ง่าย และสามารถหาแยกที่นำมาพิจารณาหาแยกถัดไป คือแยกสุดท้ายของเส้นทางได้ง่ายอีกด้วย

ในการเขียนโปรแกรมของระบบหาเส้นทางนี้จะใช้การแบ่งการทำงานในแต่ละงานที่มีการทำซ้ำ ๆ ออกเป็นฟังก์ชันให้มีการเรียกใช้ในส่วนหลักของโปรแกรม (Main)

#### 3.3.2 ผังงานทั้งหมดของโปรแกรม

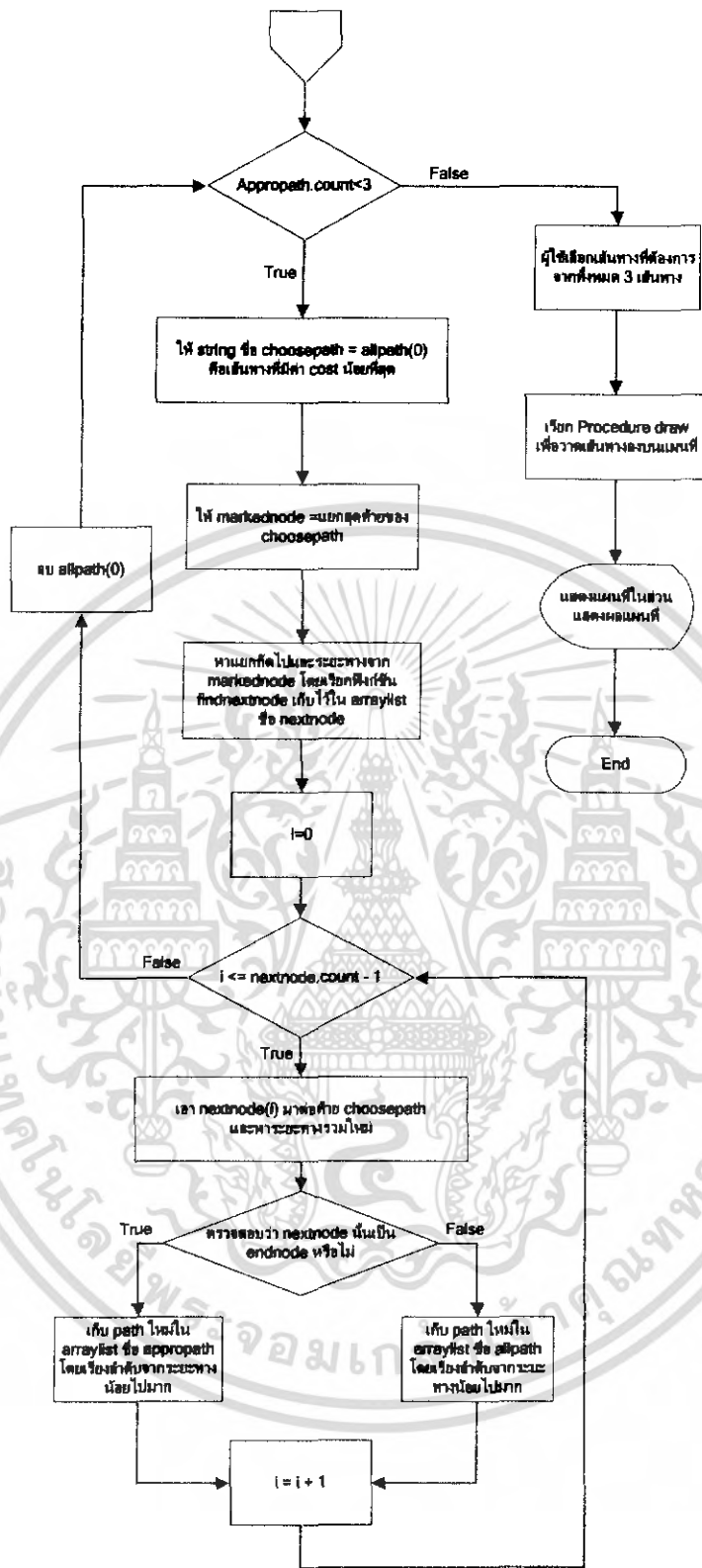
ในการออกแบบโปรแกรมนั้นในขั้นตอนแรกต้องมีการออกแบบผังงาน (Flow Chart) ขึ้นมาก่อน โดยมีผังงานที่ได้ออกแบบมาดังนี้

## 3.3.2.1 ฟังก์ชันของโปรแกรมหลัก



รูปที่ 3.21 ฟังก์ชันของโปรแกรมหลักส่วนที่ 1

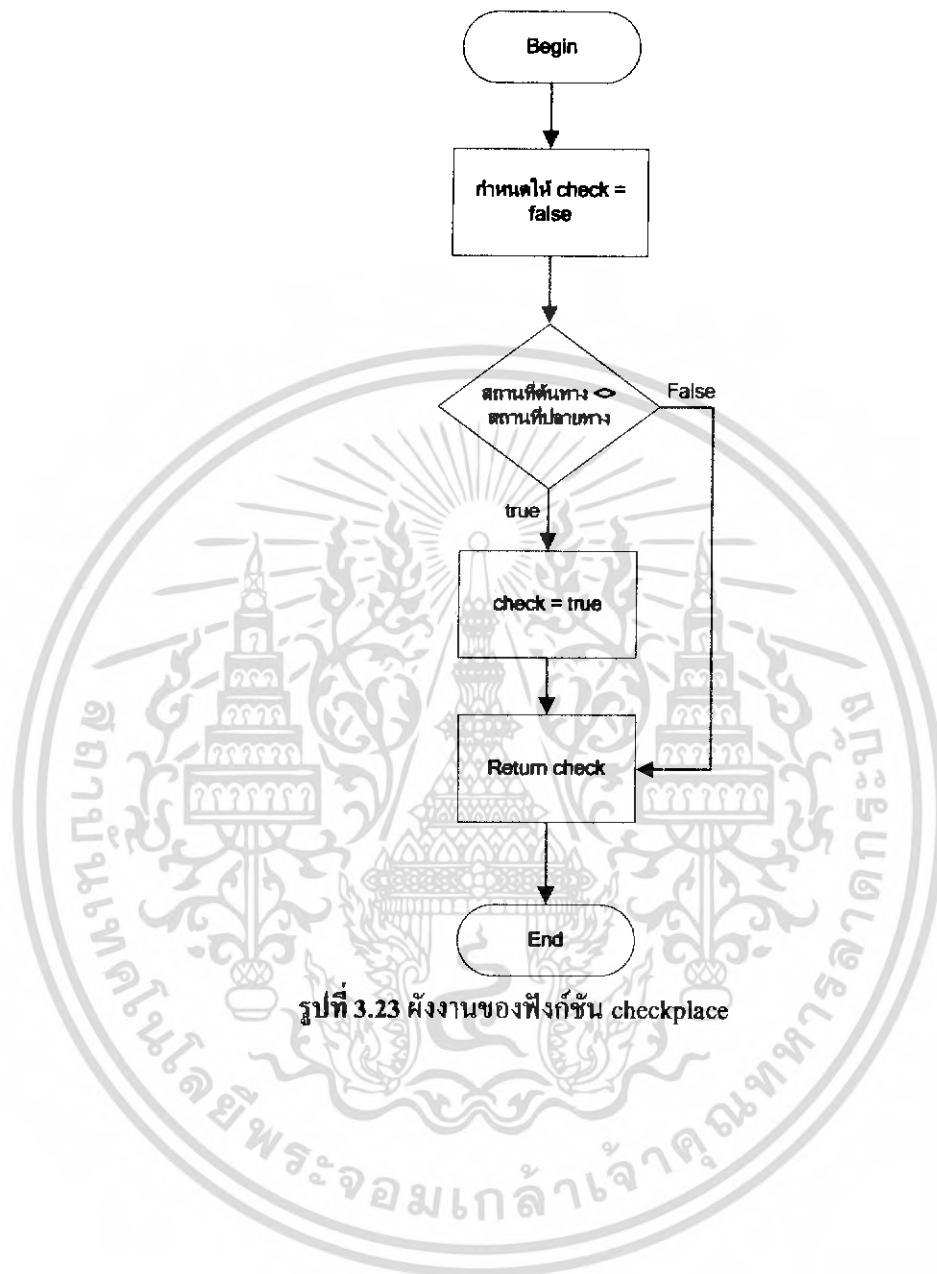
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.22 ฟังก์ชันของโปรแกรมหลักส่วนที่ 2

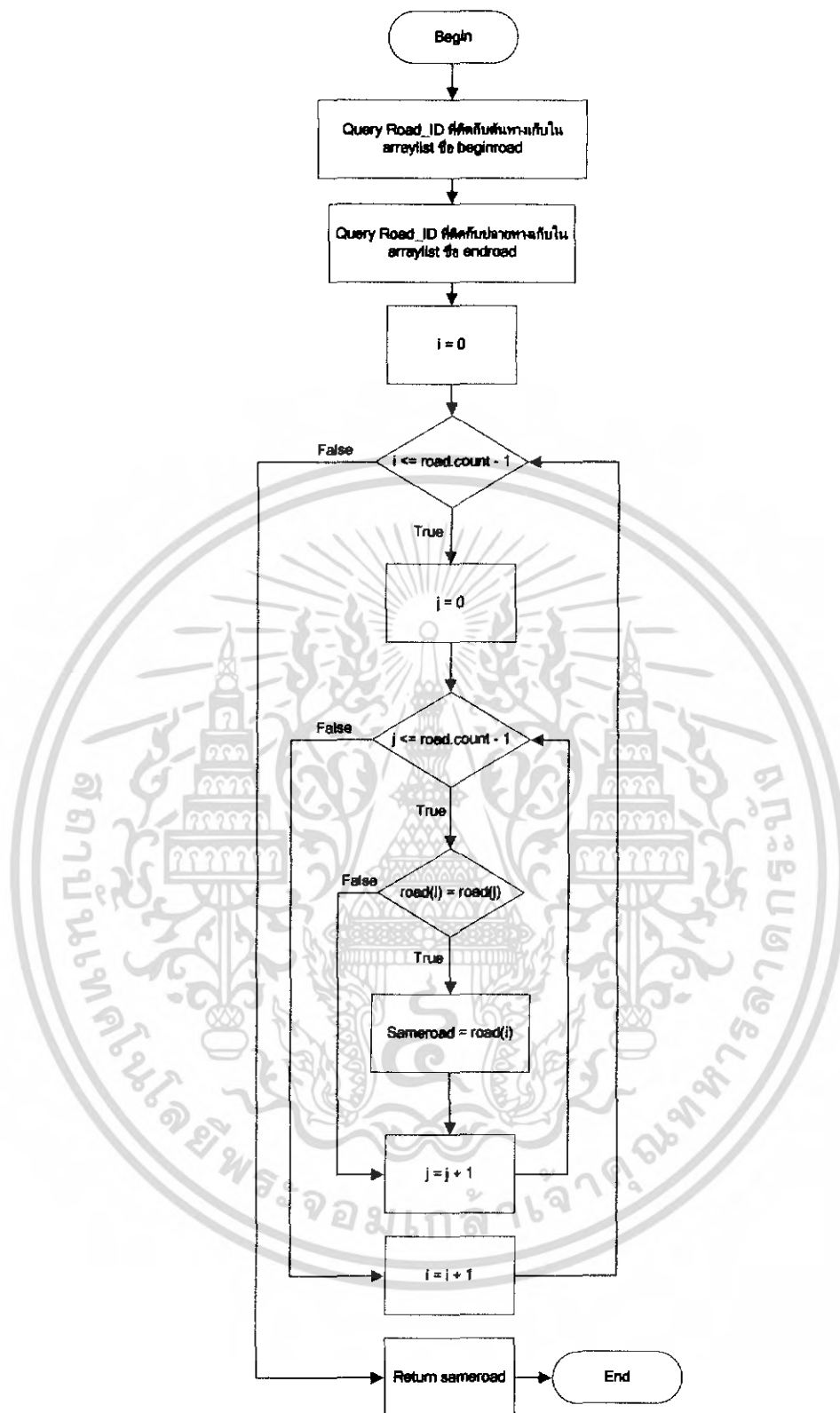
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3.2.2 ฟังก์ชันของฟังก์ชันย่อย



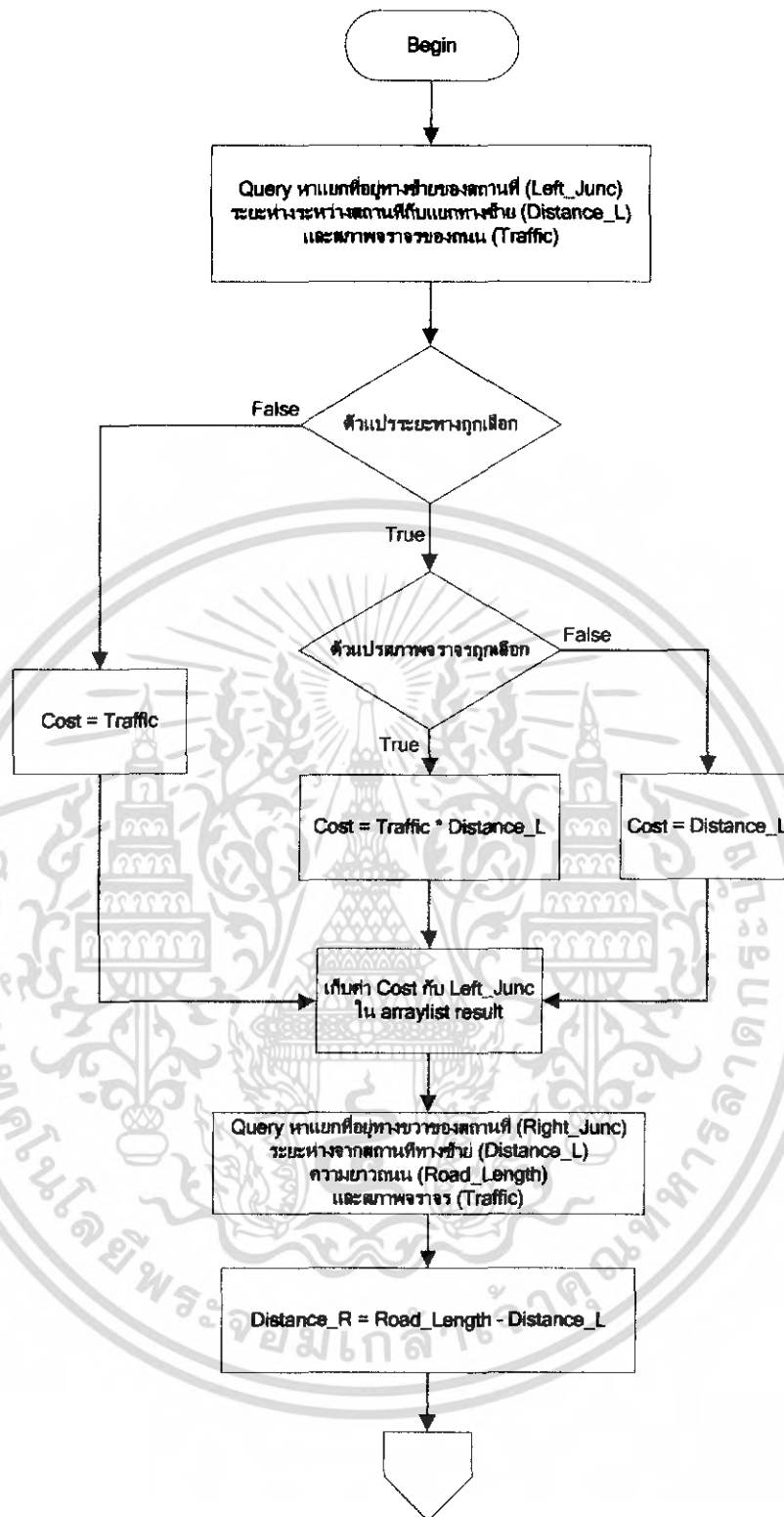
รูปที่ 3.23 ฟังก์ชันของฟังก์ชัน checkplace

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



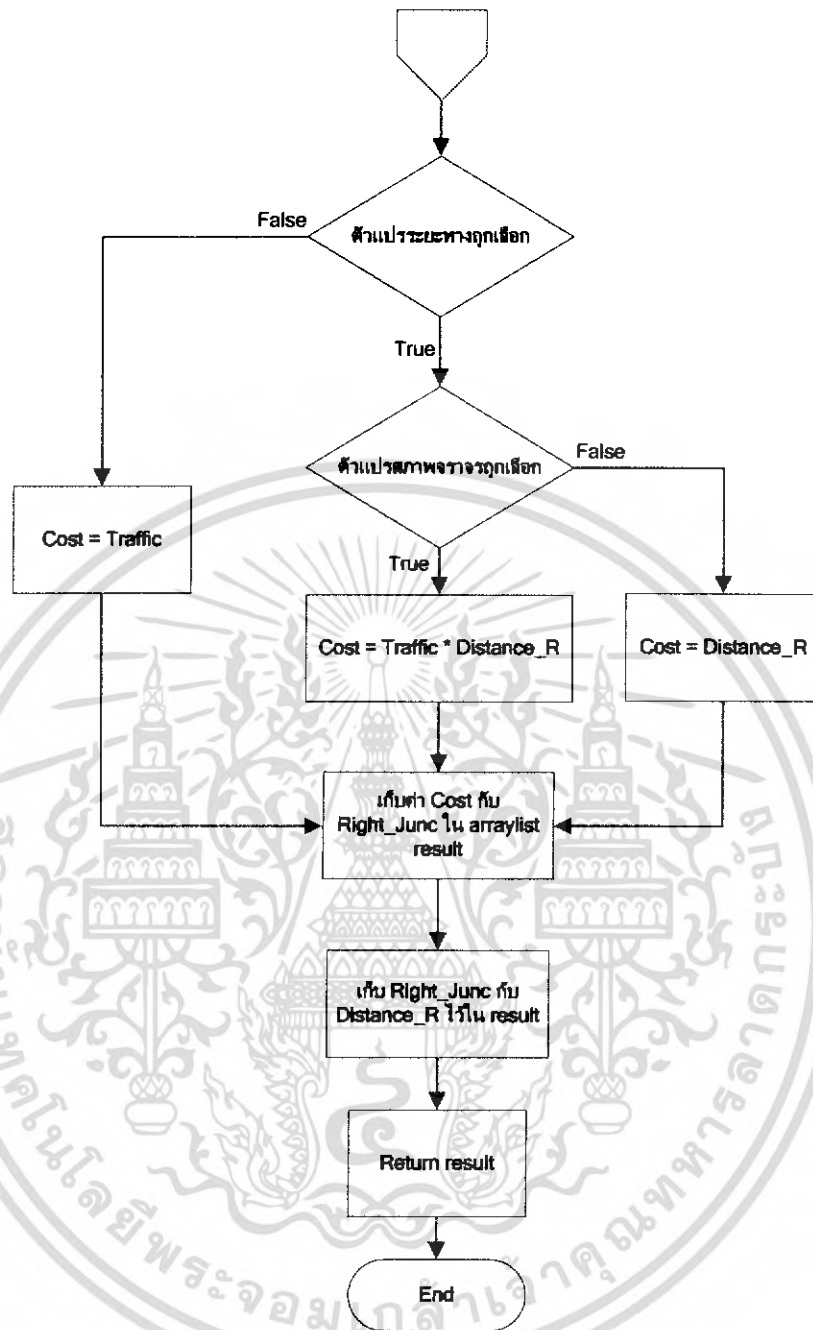
รูปที่ 3.24 ผังงานของฟังก์ชัน checkroad

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



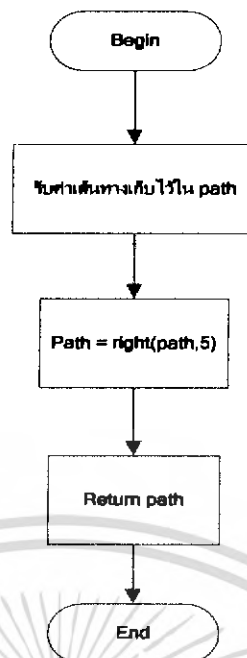
รูปที่ 3.25 ผังงานของฟังก์ชัน findjunction ส่วนที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

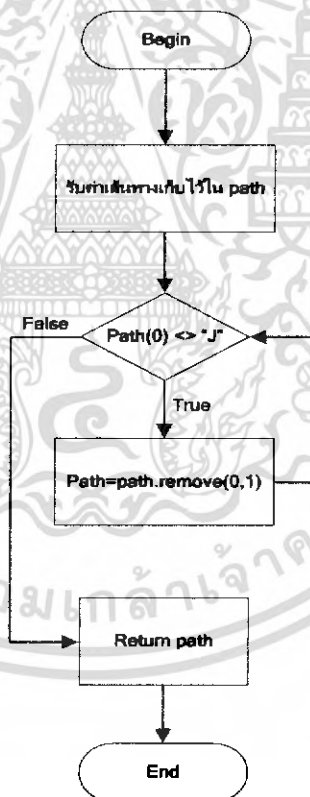


รูปที่ 3.26 ผังงานของฟังก์ชัน findjunction ส่วนที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

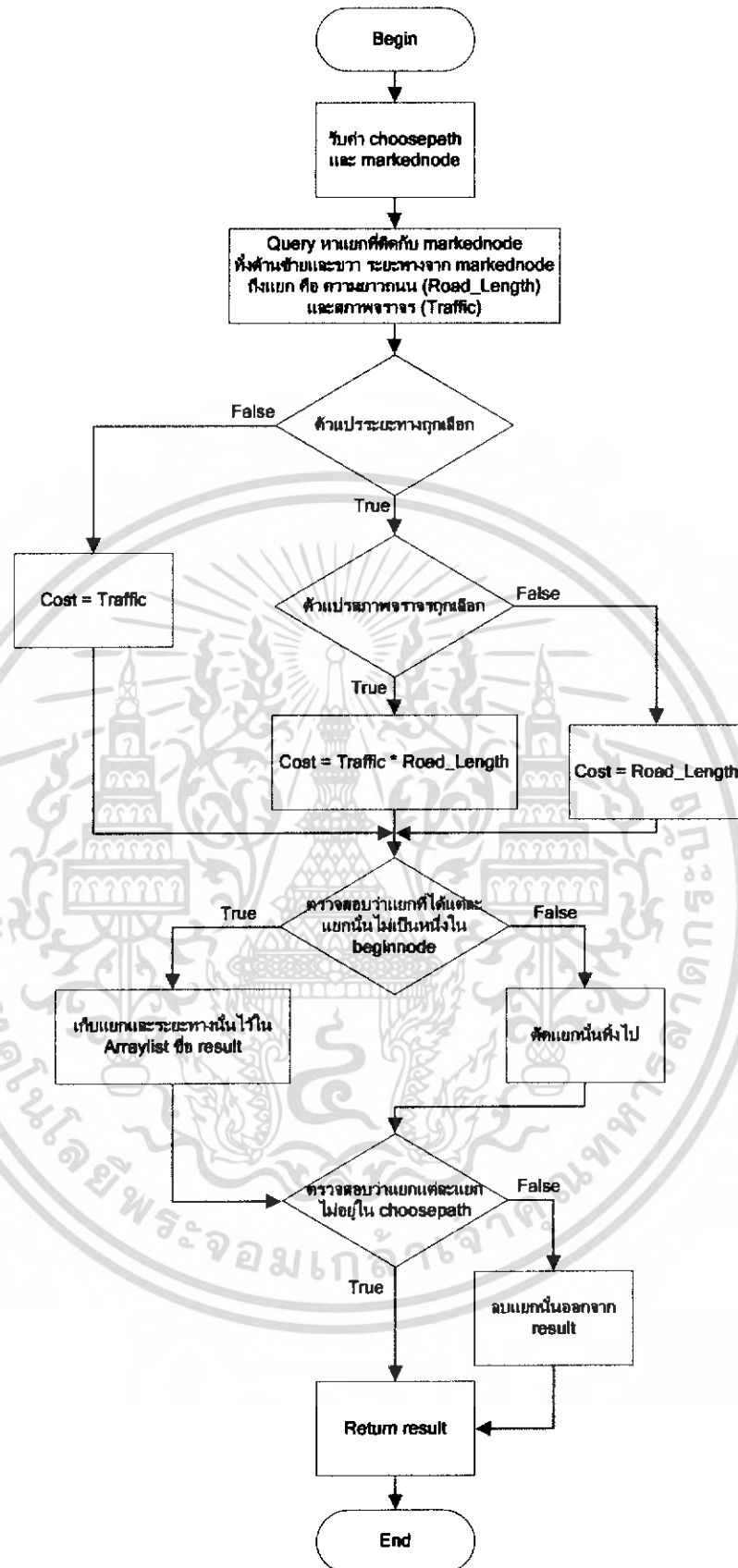


รูปที่ 3.27 ผังงานของฟังก์ชัน findmarkednode



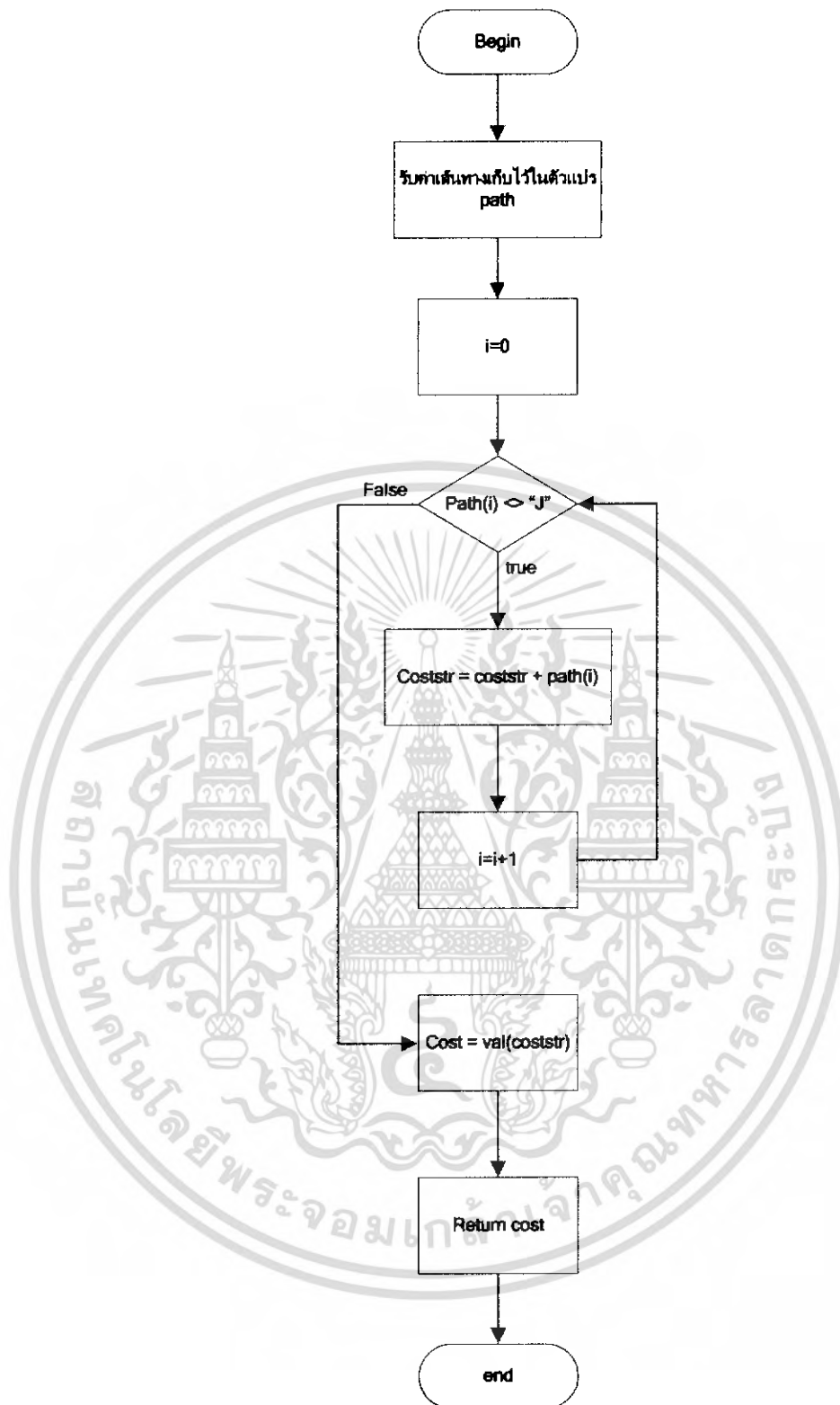
รูปที่ 3.28 ผังงานของฟังก์ชัน cutcost

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



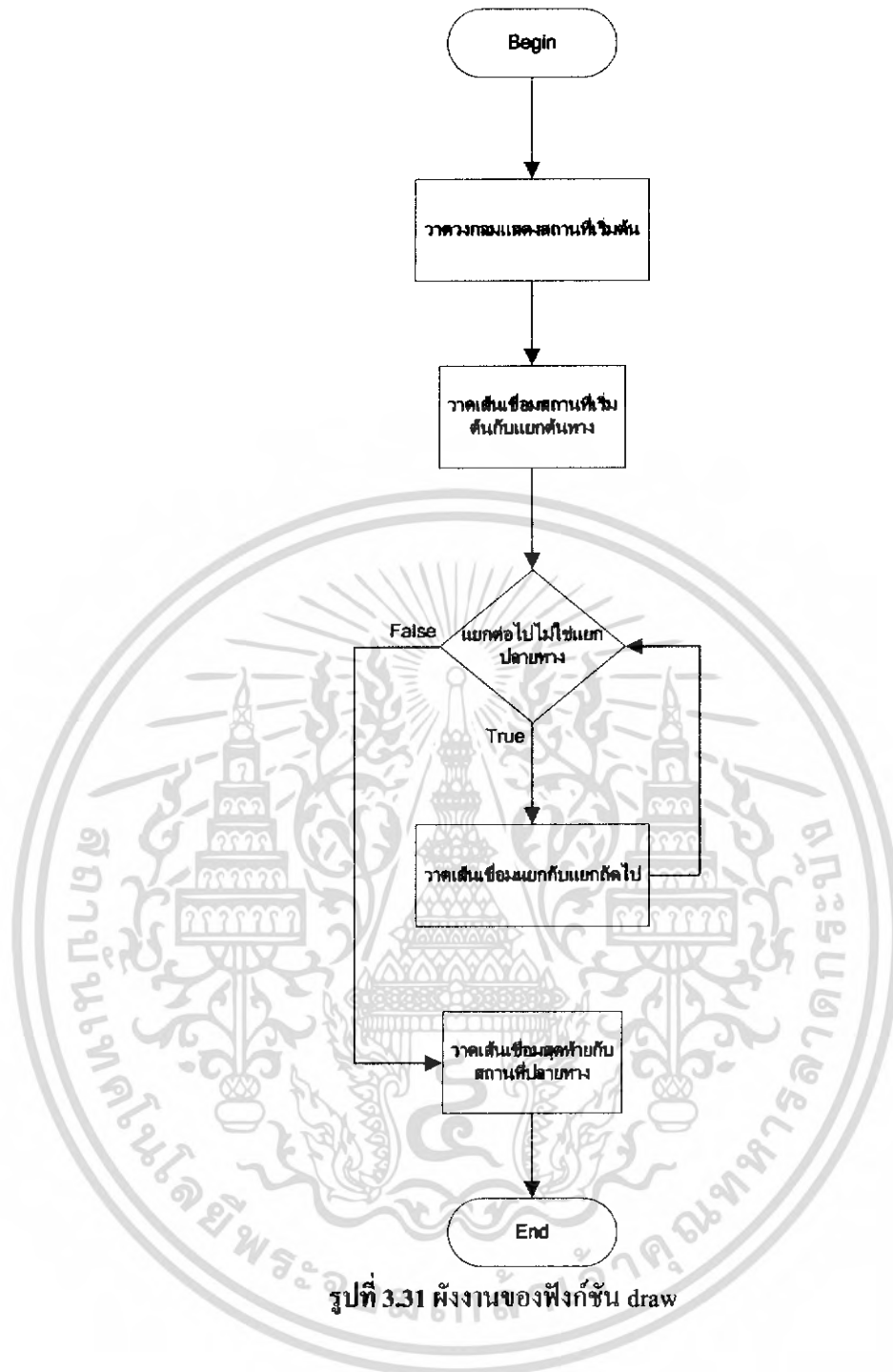
รูปที่ 3.29 ผังงานของฟังก์ชัน findnextnode

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.30 ผังงานของฟังก์ชัน findcost

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

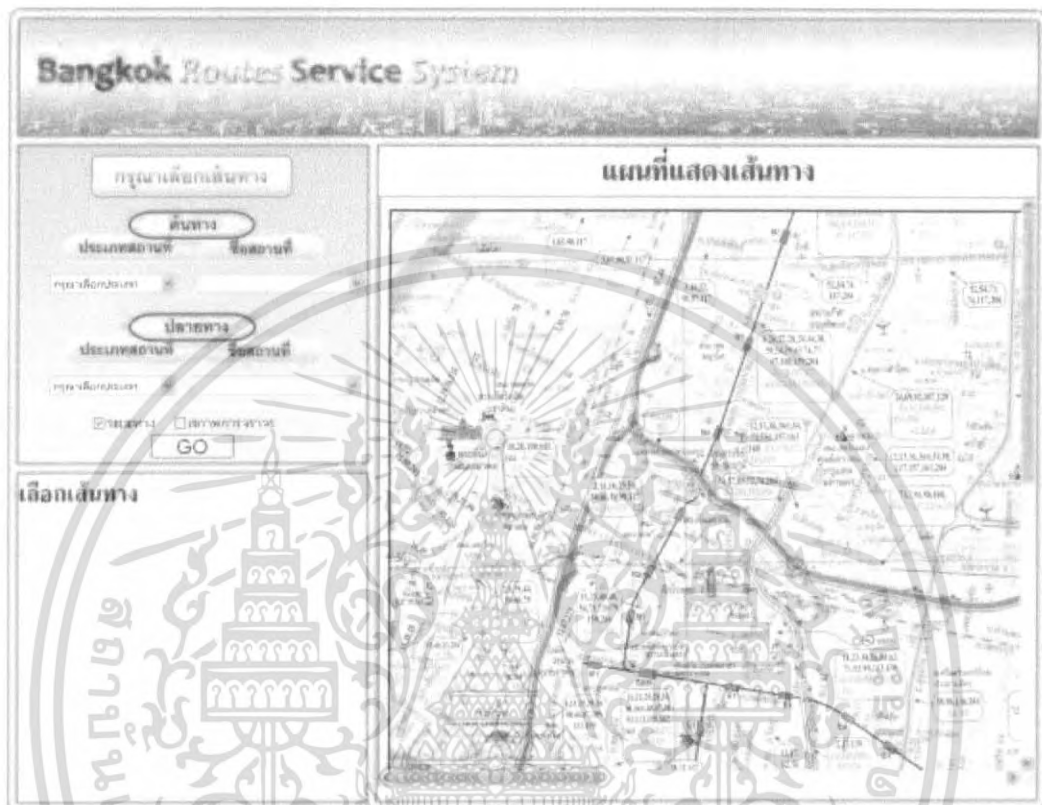


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

### ผลการทดลอง

#### 4.1 ส่วนติดต่อกับผู้ใช้ (User Interface)

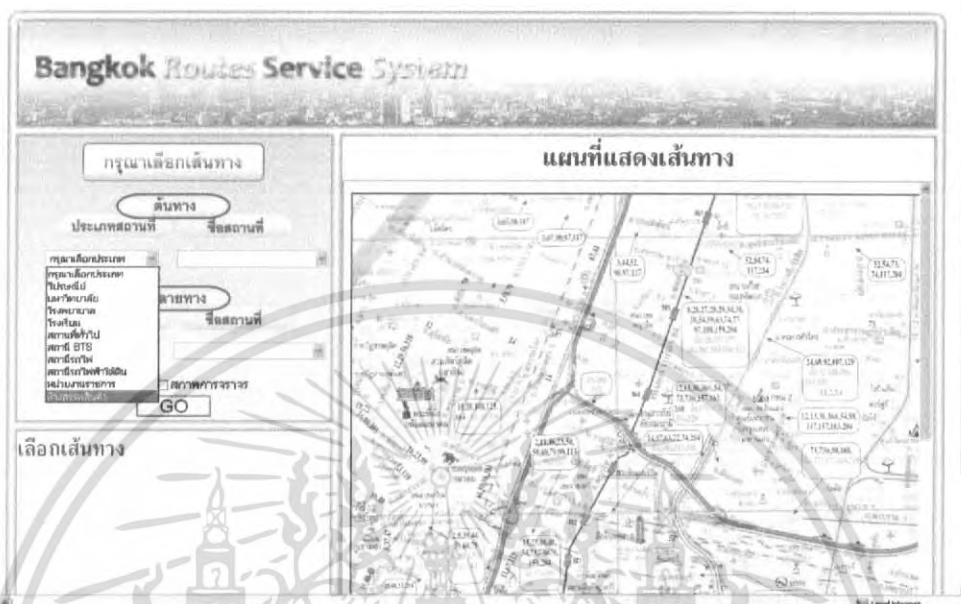


รูปที่ 4.1 แสดงหน้าแรกของส่วนติดต่อกับผู้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4.2 ขั้นตอนการใช้งานโปรแกรม

1. เลือกประเภทของสถานที่ต้นทาง โปรแกรมจะแสดงสถานที่ต้นทางทั้งหมดที่อยู่ในหมวดหมู่ที่เลือก ดังรูปที่ 4.2



รูปที่ 4.2 ขั้นตอนการเลือกประเภทของสถานที่ต้นทาง

2. เลือกสถานที่ต้นทาง ดังรูปที่ 4.3



รูปที่ 4.3 ขั้นตอนการเลือกสถานที่ต้นทาง

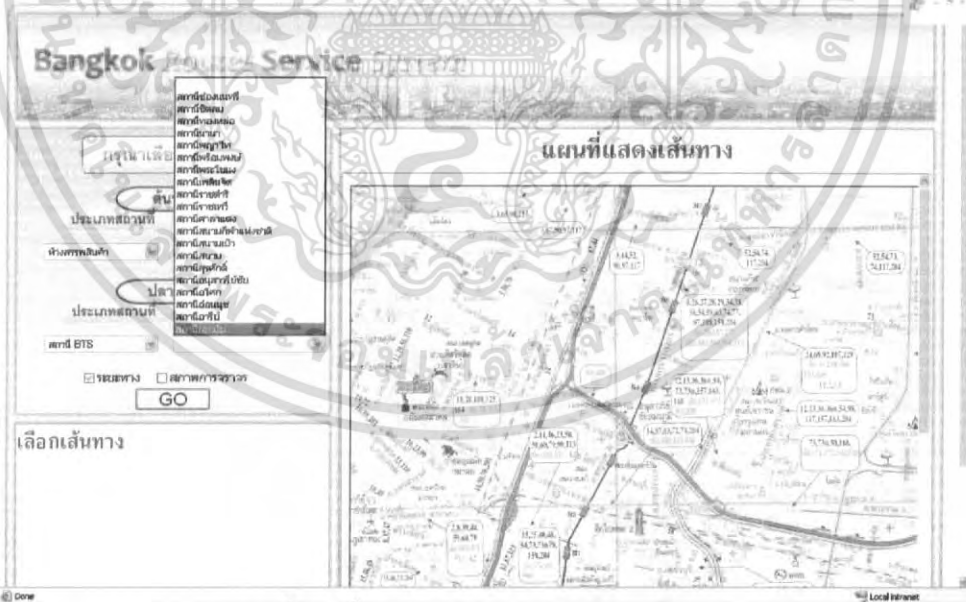
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เลือกประเภทของสถานที่ปลายทาง โปรแกรมจะแสดงสถานที่ปลายทางทั้งหมดที่อยู่ในหมวดหมู่ที่เลือก ดังรูปที่ 4.4



รูปที่ 4.4 ขั้นตอนการเลือกประเภทของสถานที่ปลายทาง

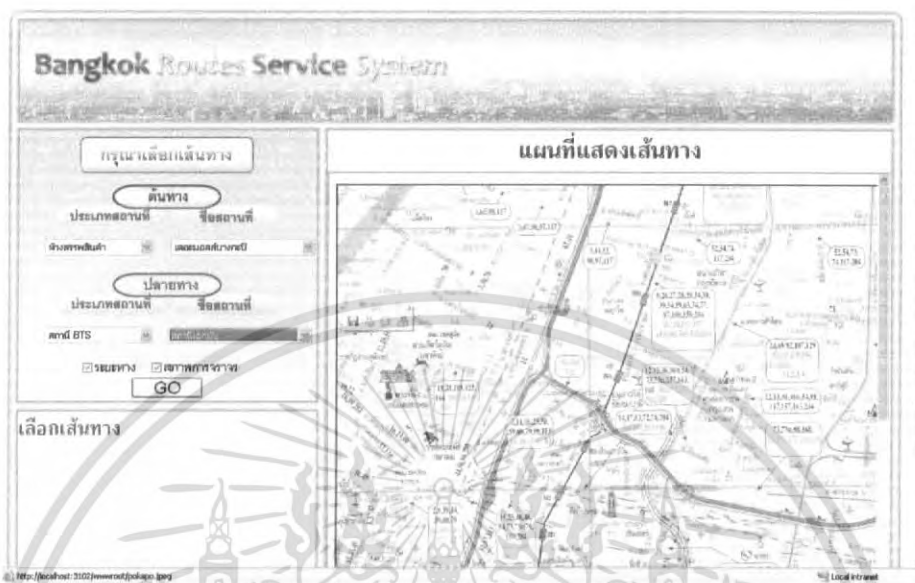
4. เลือกสถานที่ปลายทาง ดังรูปที่ 4.5



รูปที่ 4.5 ขั้นตอนการเลือกสถานที่ปลายทาง

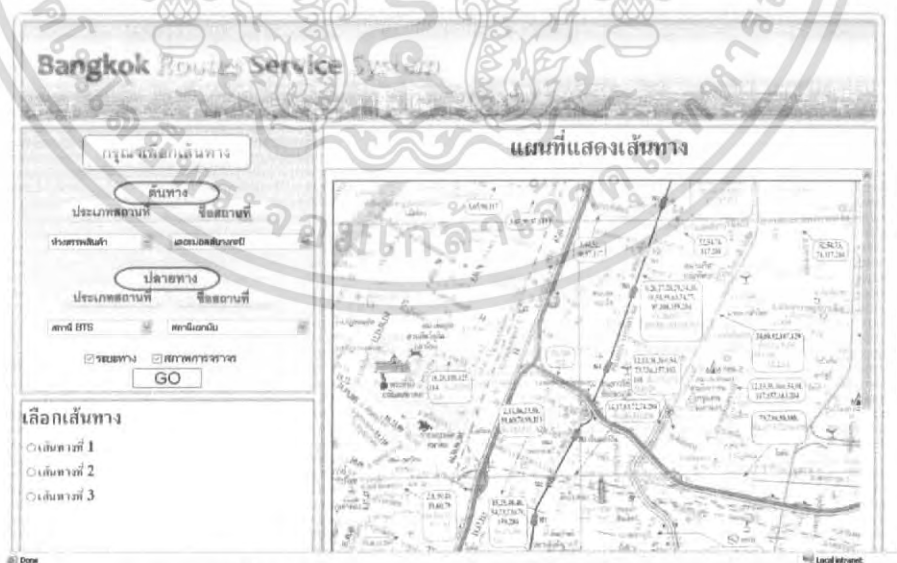
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. เลือกตัวแปรที่ใช้ในการคำนวณซึ่งได้แก่ ระยะทาง และ สภาพการจราจร โดยผู้ใช้สามารถเลือกอย่างใดอย่างหนึ่งหรือทั้งสองอย่างก็ได้ ดังรูปที่ 4.6



รูปที่ 4.6 ขั้นตอนการเลือกตัวแปรในการคำนวณ

6. กดปุ่ม GO เพื่อเริ่มการทำงานของโปรแกรม  
7. เมื่อโปรแกรมทำการคำนวณเส้นทางเสร็จสิ้นจะแสดงตัวเลือกให้ผู้เลือกเส้นทางทั้งหมด 3 เส้นทาง โดยเส้นทางที่ 1 จะเป็นเส้นทางที่ดีที่สุด ดังรูปที่ 4.7



รูปที่ 4.7 ขั้นตอนการแสดงผลตัวเลือกเส้นทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. ในกรณีที่สถานที่ต้นทางและปลายทางที่เลือกเป็นสถานที่เดียวกันจะมีการแจ้งเตือนผู้ใช้ให้เลือกสถานที่ใหม่ ดังรูปที่ 4.8



รูปที่ 4.8 ขั้นตอนการแจ้งเตือนเมื่อผู้ใช้เลือกสถานที่เดียวกัน

9. ในกรณีที่สถานที่ต้นทางและปลายทางที่เลือกตั้งอยู่บนถนนเดียวกันจะทำการคำนวณระยะห่างและแจ้งกับผู้ใช้พร้อมแสดงแผนที่ประกอบ ดังรูปที่ 4.9



รูปที่ 4.9 ขั้นตอนการบอกระยะห่างและแสดงแผนที่เมื่อสถานที่อยู่บนถนนเดียวกัน

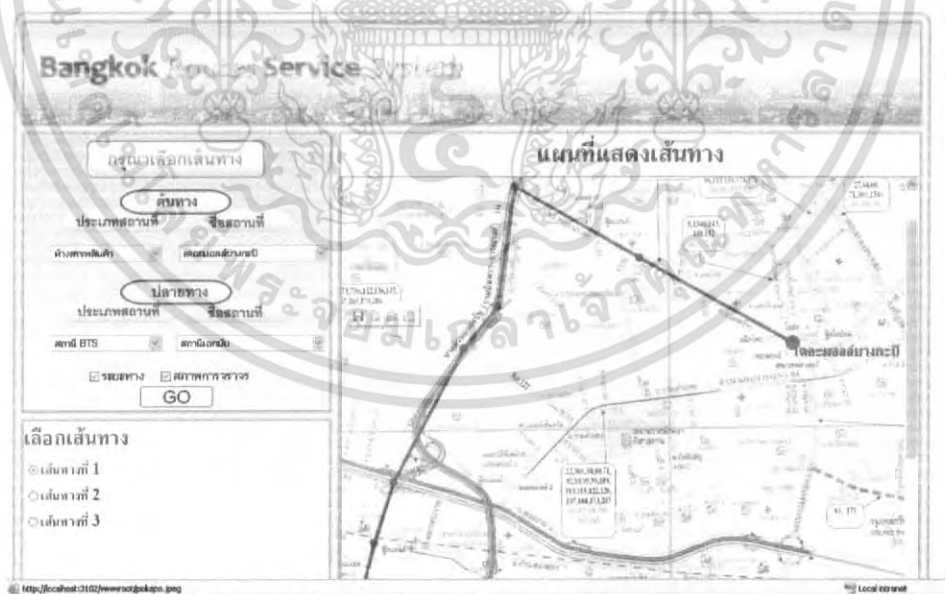
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10. ในกรณีที่ผู้ใช้ไม่ได้เลือกตัวแปรในการคำนวณเส้นทางจะมีการแจ้งเตือนให้ผู้ใช้ทราบ ดังรูปที่ 4.10



รูปที่ 4.10 ขั้นตอนการแจ้งเมื่อผู้ใช้ไม่ได้เลือกตัวแปรที่ใช้คำนวณ

11. เมื่อผู้ใช้เลือกเส้นทาง จะมีแผนที่ที่แสดงเส้นทางนั้นๆ เป็นสีแดงอยู่บนแผนที่ แสดงที่ บริเวณแผนที่แสดงเส้นทาง ดังรูปที่ 4.11 – 1.13

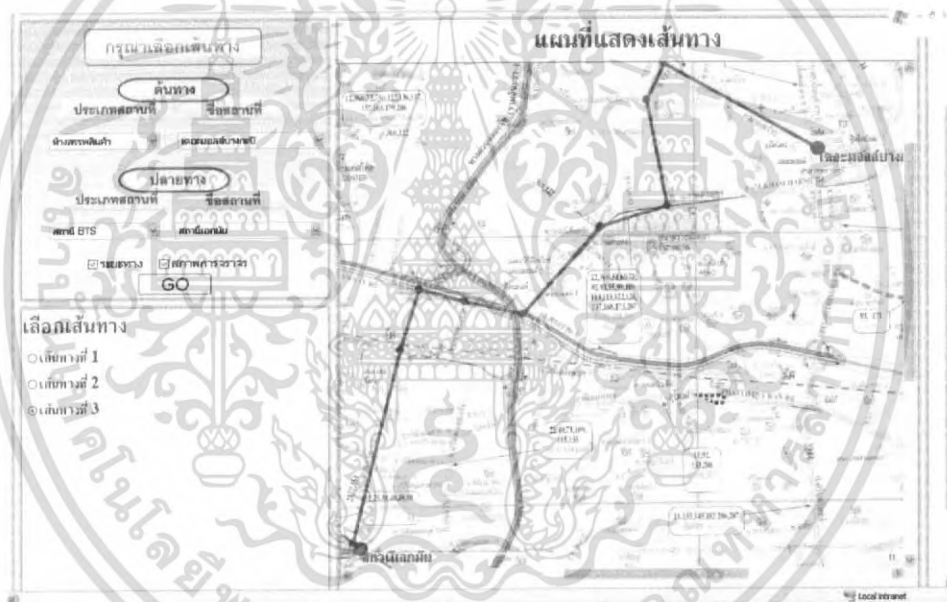


รูปที่ 4.11 ขั้นตอนเมื่อผู้ใช้เลือกดูแผนที่ของเส้นทางที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



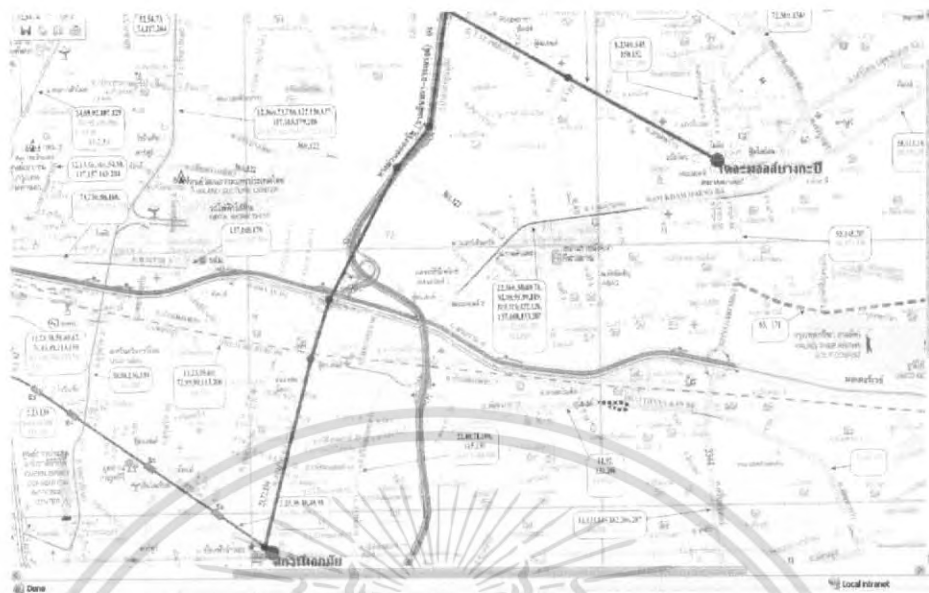
รูปที่ 4.12 ขั้นตอนเมื่อผู้ใช้เลือกดูแผนที่ของเส้นทางที่ 2



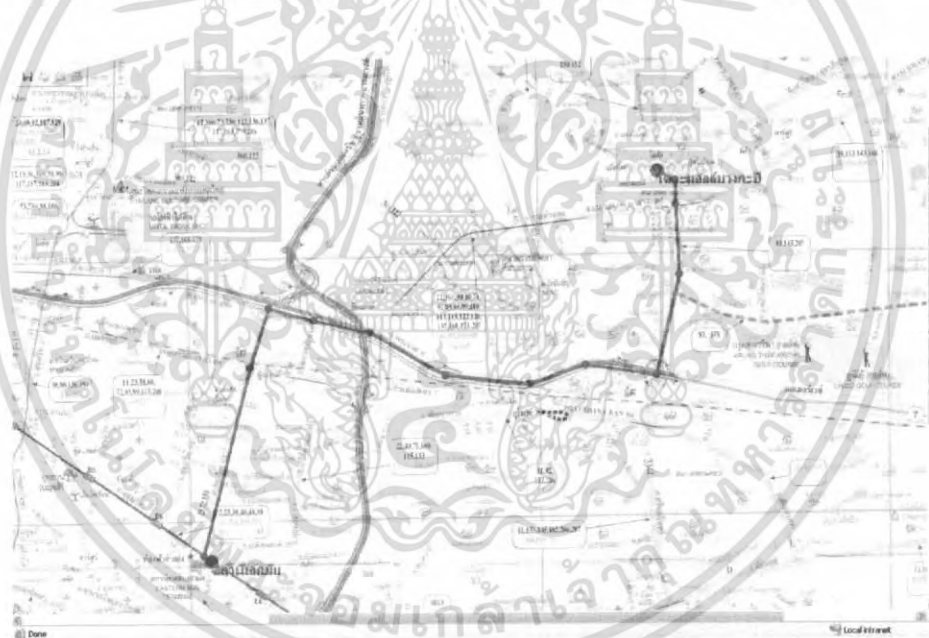
รูปที่ 4.13 ขั้นตอนเมื่อผู้ใช้เลือกดูแผนที่ของเส้นทางที่ 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 12. ผู้ใช้สามารถกดที่รูปเพื่อดูภาพขนาดจริงได้ ดังรูปที่ 4.14-4.16

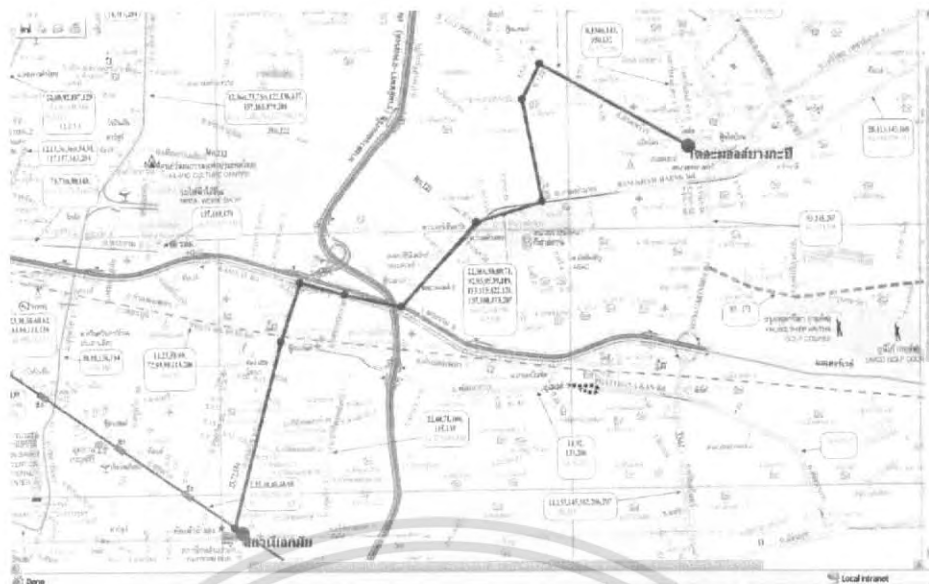


รูปที่ 4.14 ขั้นตอนเมื่อผู้ใช้เลือกดูแผนที่ขนาดจริงของเส้นทางที่ 1



รูปที่ 4.15 ขั้นตอนเมื่อผู้ใช้เลือกดูแผนที่ขนาดจริงของเส้นทางที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.16 ขั้นตอนเมื่อผู้ใช้เลือกแผนที่ขนาดจริงของเส้นทางที่ 3



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### สรุปผลการทดลอง

#### 5.1 สรุปการพัฒนาโครงการ

ระบบสามารถคำนวณหาเส้นทางที่เหมาะสมระหว่างสถานที่ 2 สถานที่ ซึ่งอยู่ในพื้นที่กรุงเทพฯ ประมาณ 180 ตารางกิโลเมตรครอบคลุมตั้งแต่โรงพยาบาลวิชัยยุทธ ถึงสวนหลวง ร.9 ได้อย่างถูกต้อง โดยมีตัวแปรที่นำมาคำนวณคือ ระยะทางซึ่งวัดจากแผนที่จริงและสภาพการจราจรที่ได้มาจากการสุ่มค่าลงในฐานข้อมูล และใช้แผนที่จริงในการแสดงผลได้อย่างชัดเจน โดยระบบนี้ระบบเป็นเว็บแอปพลิเคชัน (Web Application) ที่ใช้ระบบปฏิบัติการวินโดวส์เซิร์ฟเวอร์

#### 5.2 ปัญหาที่พบในการทำโครงการ

- เส้นทางในกรุงเทพฯ มีจำนวนมากและซับซ้อนทำให้ไม่สามารถรวบรวมมาได้ครบถ้วน และต้องใช้เวลาอย่างมากในการเพิ่มข้อมูลลงในฐานข้อมูล ซึ่งทำให้เกิดความผิดพลาดบ่อยครั้ง
- ในการคำนวณหาเส้นทางมีเงื่อนไขที่ซับซ้อนและหลากหลาย เช่น การที่สถานที่บางสถานที่ตั้งอยู่ติดถนนหลายถนนทำให้มีแยกที่ติดกับสถานที่จำนวนมาก หรือการต้องออกแบบโปรแกรมให้ป้องกันการเกิดลูป (loop) และการวนกลับของเส้นทาง ทำให้ยากต่อการพัฒนาให้ครอบคลุม

#### 5.3 ข้อจำกัดของโครงการ

- ไม่มีข้อมูลของเส้นทางในซอยทางลัด และทางด่วน
- ข้อมูลเส้นทางไม่ครอบคลุมทั่วกรุงเทพฯ
- เนื่องจากมีการคำนวณข้อมูลปริมาณมากทำให้ต้องใช้เวลาในการประมวลผล
- ในการแสดงผลเส้นทางโดยการวาดรูปนั้น ไม่สามารถวาดเส้นถนนที่เป็นเส้นโค้งได้ แต่ได้มีการแก้ไขโดยแบ่งถนนเป็นส่วนย่อยแล้ววาดเส้นตรงจุดต่อจุดแทน
- ระบบไม่มีการคำนึงถึงทิศทางการเดินทางจริง คือ ถนนที่มีการเดินทางเดียว การห้ามเลี้ยว การห้ามกลับรถ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 5.4 แนวทางในการพัฒนาต่อ

- เพิ่มฐานข้อมูลให้สมบูรณ์ครอบคลุมทุกพื้นที่ รวมไปถึงเส้นทางที่เป็นซอยทางลัด และทางคั่น
- พัฒนาการแสดงผลให้ชัดเจนและสวยงามยิ่งขึ้น โดยอาจจะใช้การประมวลผลภาพ (Image Processing) เข้ามาช่วยในการพัฒนา
- พัฒนาโปรแกรมให้มีการคำนวณที่น้อยลงเพื่อลดเวลาในการประมวลผล
- พัฒนาให้ข้อมูลสภาพจราจรของถนนต่างๆเปลี่ยนแปลงแบบทันที (Real time)
- พัฒนาให้ระบบคำนึงถึงทิศทางการเดินทางจริง คือ ถนนที่มีการเดินทางเดียว การห้ามเลี้ยว การห้ามกลับรถ เป็นต้น



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บรรณานุกรม

ทวีชัย หงส์สุมาลย์ และสงวนชัย สุวรรณชีวะศิริ. 2546. อินเทอร์เน็ต ASP.NET ฉบับสมบูรณ์.  
กรุงเทพฯ : โปรวิชั่น  
ศูนย์แผนที่บางกอกไคด์. 2548. แผนที่กรุงเทพ-เดินทาง บัสไกด์. กรุงเทพฯ : หจก.  
บางกอกไคด์



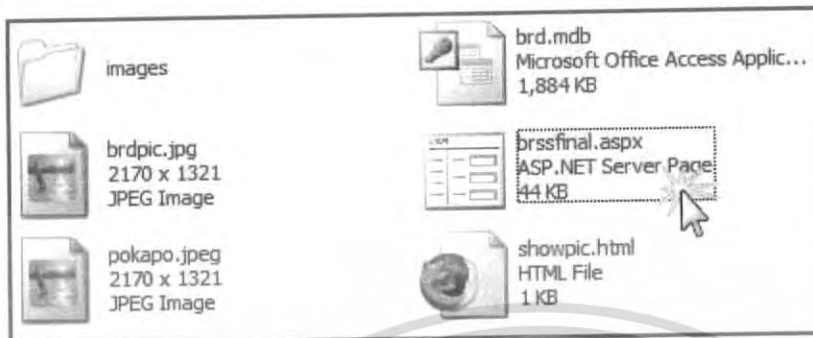
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



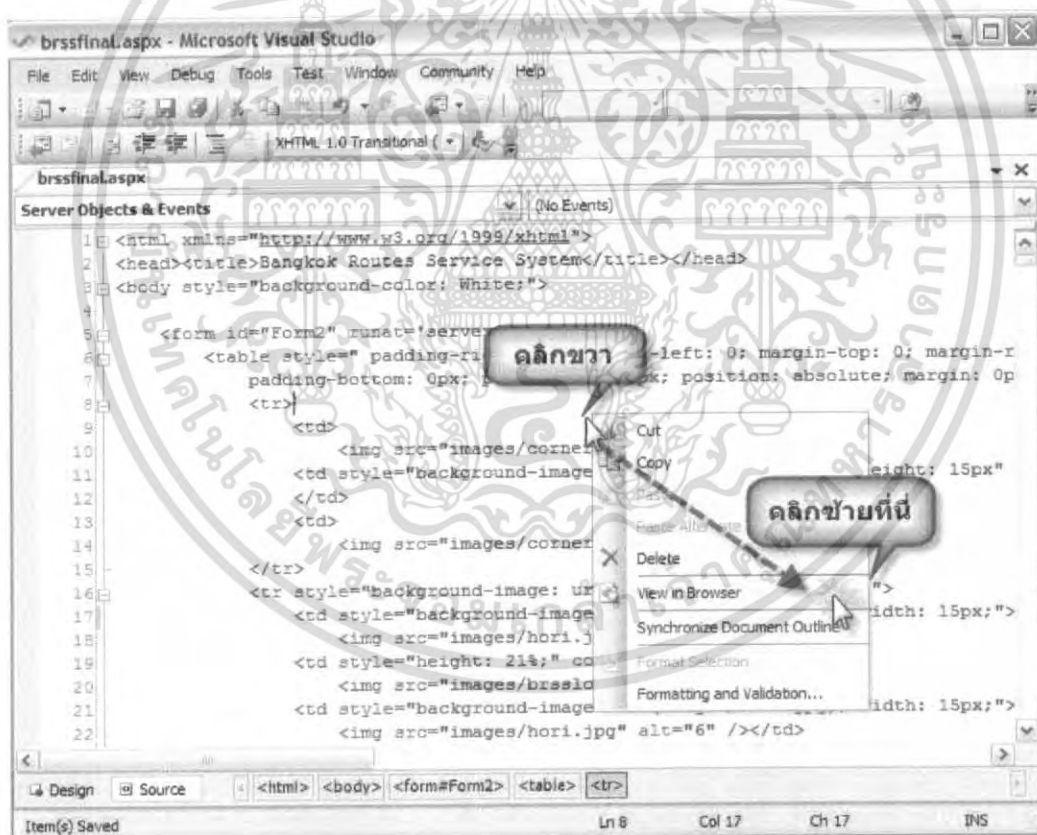
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากติดตั้งโปรแกรมวิชวลสตูดิโอ 2005 (Visual Studio 2005) เรียบร้อยแล้วให้ทำตามขั้นตอนดังนี้

1. คัดลอกแฟ้ม “BRSS” ลงไปไว้บนเครื่องของท่าน โดยในแฟ้มประกอบไปด้วยไฟล์ดังภาพ



2. ดับเบิ้ลคลิกที่ไฟล์ “brssfinal.aspx” จะเป็นการเรียกใช้งาน โปรแกรม จากนั้นคลิกขวาวบริเวณพื้นที่ทำงาน แล้วเลือก “View in Browser” ดังรูป



3. โปรแกรมจะเปิดเว็บเบราว์เซอร์ (Web Browser) เพื่อเริ่มการทำงานโดยอัตโนมัติ ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### Bangkok Routes Service System

กรุณาเลือกเส้นทาง

**ต้นทาง**

ประเภทสถานที่  ชื่อสถานที่

ถนนเลือกประเภท

---

**ปลายทาง**

ประเภทสถานที่  ชื่อสถานที่

ถนนเลือกประเภท

ระยะทาง  สภาพการจราจร

**GO**

**แผนที่แสดงเส้นทาง**

**เลือกเส้นทาง**



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<html xmlns="http://www.w3.org/1999/xhtml">
<head><title>Bangkok Routes Service System</title></head>
<body style="background-color: White;">

    <form id="Form2" runat='server'>
        <table style="padding-right: 0px;margin-left: 0; margin-top: 0;
margin-right: 0; width: 830px; height: 630px; padding-left: 0px;
padding-bottom: 0px; padding-top: 0px; position: absolute;
margin: 0px;"cellpadding="0" cellspacing="0">
            <tr>
                <td>
                    </td>
                    <td style="background-image: url(images/vert.jpg);
height: 15px" colspan="3">
                </td>
                <td>
                    </td>
            </tr>
            <tr style="background-image: url(images/brsslogo2s.jpg);">
                <td style="background-image: url(images/hori.jpg);
width: 15px;">
                    </td>
                    <td style="height: 21%; " colspan="3 ">
                        </td>
                    <td style="background-image: url(images/hori.jpg);
width: 15px;">
                        </td>
            </tr>
            <tr>
                <td>
                    </td>
                    <td style="background-image: url(images/vert.jpg);
height: 15px;">
                </td>
                <td style="width: 16px">
                    </td>
                    <td style="background-image: url(images/vert.jpg);
height: 15px;">
                </td>
                <td>
                    </td>
            </tr>
            <tr>
                <td style="background-image: url(images/hori.jpg);
width: 15px; height: 333px;">
                    </td>
                    <td style="height: 333px; text-align: center;"
valign="top">
                <!------->
                <table id="Table_01" style="height: 309px; width:
390px; background-image: url(images/gg.jpg);" border="0" cellpadding="0"
cellspacing="0">
                    <tr>
                        <td colspan="2">
                            </td>
                        <!--1-->
                    </tr>
                </tr>
            </tr>
        </table>
    </form>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```

        </tr>
        <tr>
            <td colspan="2" style="height: 9px">
                </td>
            <!--9-->
        </tr>
    </table>
    <!------->
    <table style=" width: 412px;background-image:
url(images/gg.jpg);" >
        <tr><td align="center">
            <asp:checkboxlist
RepeatDirection="Horizontal" OnSelectedIndexChanged="main"
ID="parameter" Visible="true" AutoPostBack="false" runat="server"
Width="248px">
                <asp:listitem Value="distance"
Text="ระยะทาง" Selected="True" />
                <asp:listitem Value="traffic"
Text="สภาพการจราจร" Selected="false" />
            </asp:checkboxlist>
            <asp:button text= "GO" id="go"
onclick="main" runat="server" style="color: black" Width="100"
Height="30" Font-Size= "Large" /></td></tr>
        </table>
    </td>
    <td style="width: 16px; background-image:
url(images/hori.jpg); height: 333px;">
        </td>
        <td style="width: 57%; font-size: 12mm; color: blue;
font-family: EurosiaUPC;" rowspan="4"
align="center" valign="top">
            <b>แผนที่แสดงเส้นทาง</b><br />
            <iframe src="showpic.html" style="width:780;
height:700" ></iframe>
        </td>
        <td style="width: 15px; background-image:
url(images/hori.jpg)" rowspan="4">
            </td>
        </tr>
        <tr>
            <td style="height: 15px" >
                </td>
            <td style="height: 15px ; background-
image:url(images/vert.jpg);"></td>
            <td style="width: 16px">
                </td>
        </tr>
        <tr>
            <td style="width: 15px; background-image:
url(images/hori.jpg)">
                </td>
            <td valign="top" style="font-size: 10mm; color: blue;
font-family: EurosiaUPC; ">
                <b>เลือกเส้นทาง</b>
                <br />

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        <asp:ListBox ID="resultlist" AutoPostBack="false"
Visible="false" runat="server" />
        <asp:RadioButtonList runat="server" ID="pathchoose"
OnSelectedIndexChanged="draw" AutoPostBack="true" Visible="false" Font-
Names="EucrosiaUPC" Font-Size="Large" Font-Bold="true" ForeColor="Blue">
            <asp:ListItem value="เส้นทางที่ 1" Selected="False"/>
            <asp:ListItem value="เส้นทางที่ 2" Selected="False"/>
            <asp:ListItem value="เส้นทางที่ 3" Selected="False"/>
        </asp:RadioButtonList>
        <br /> <br /> <br /><br /> <br /> <br />
    </td>

    <td style="width: 16px; background-image:
url(images/hori.jpg)">
        </td>
    </tr>
    <!--
----->
    <tr>
        <td style="width: 15px; background-image:
url(images/hori.jpg)">
            </td>
            <td valign="top" style="font-size: 6mm; color: blue;
font-family: EucrosiaUPC;">
                <br/><br/>
            </td>
            <td style="width: 16px; background-image:
url(images/hori.jpg)">
                </td>
            </tr>
            <!--
----->
    <tr>
        <td style="height: 15px">
            </td>
            <td style="height: 15px; background-image:
url(images/vert.jpg)">
                </td>
            <td style="height: 15px; width: 16px;">
                </td>
            <td style="height: 15px; background-image:
url(images/vert.jpg)">
                </td>
            <td style="height: 15px">
                </td>
        </tr>
    </table>

</form>
</body>
</html>
<%@import namespace="System.Data" %>
<%@import namespace="System.Data.OleDb" %>
<%@import Namespace="system.drawing" %>
<%@import Namespace="system.drawing.imaging" %>
<%@import Namespace="System.threading" %>>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<script language="vb" runat="server">
    Sub page_init(ByVal sender As Object, ByVal e As EventArgs)
        startdraw()
        Dim myconn As New
OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0; Data Source=" +
Server.MapPath("brd.mdb"))
        myconn.Open()
        Dim myda As New OleDbDataAdapter("select Distinct place_type
from Place order by place_type", myconn)
        Dim ds As New DataSet()
        myda.Fill(ds, "path")
        beginplacetype.DataSource = ds.Tables("path")
        beginplacetype.DataBind()
        endplacetype.DataSource = ds.Tables("path")
        endplacetype.DataBind()
        myconn.Close()
        pathchoose.Items(0).Selected = False
        pathchoose.Items(1).Selected = False
        pathchoose.Items(2).Selected = False
    End Sub

    Sub choosebeginplace(ByVal sender As Object, ByVal e As EventArgs)
        Dim name As String
        name = "Select distinct Place_Name from Place where Place_Type =
'" + beginplacetype.SelectedItem.Text + "' order by Place_Name"
        Dim myconn As New
OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0; Data Source=" +
Server.MapPath("brd.mdb"))
        myconn.Open()
        Dim myda As New OleDbDataAdapter(name, myconn)
        Dim ds As New DataSet()
        myda.Fill(ds, "path")
        beginplace.DataSource = ds.Tables("path")
        beginplace.DataBind()
        myconn.Close()
    End Sub

    Sub chooseendplace(ByVal sender As Object, ByVal e As EventArgs)
        Dim name As String
        name = "Select distinct Place_Name from Place where Place_Type =
'" + endplacetype.SelectedItem.Text + "' order by Place_Name"
        Dim myconn As New
OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0; Data Source=" +
Server.MapPath("brd.mdb"))
        myconn.Open()
        Dim myda As New OleDbDataAdapter(name, myconn)
        Dim ds As New DataSet()
        myda.Fill(ds, "path")
        endplace.DataSource = ds.Tables("path")
        endplace.DataBind()
        myconn.Close()
    End Sub

    '-----Main Program-----

    Sub main(ByVal sender As Object, ByVal e As EventArgs)
        'ตรวจสอบว่ามีกรเลือกสถานที่ครบถ้วนหรือไม่
        If beginplacetype.SelectedItem.Text = "กรุณาเลือกประเภท" Or
endplacetype.SelectedItem.Text = "กรุณาเลือกประเภท" Then
            resultlist.DataSource = ""
        End If
    End Sub

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        resultlist.DataBind()
        resultlist.Items.Add("กรุณาเลือกเลือกสถานที่ตั้งเส้นทางและปลายทางให้ครบถ้วน")
        resultlist.Visible = True
        pathchoose.Visible = False
        Exit Sub
    End If
    If beginplace.SelectedItem.Text = "" Or
endplace.SelectedItem.Text = "" Then
        resultlist.DataSource = ""
        resultlist.DataBind()
        resultlist.Items.Add("กรุณาเลือกเลือกสถานที่ตั้งเส้นทางและปลายทางให้ครบถ้วน")
        resultlist.Visible = True
        pathchoose.Visible = False
        Exit Sub
    End If

    If (Not parameter.Items(0).Selected And Not
parameter.Items(1).Selected) Then      'ตรวจสอบว่ามีทางเลือกตัวแปรในการคำนวณหรือไม่
        resultlist.DataSource = ""
        resultlist.DataBind()
        resultlist.Items.Add("กรุณาเลือกตัวแปรที่ใช้คำนวณเส้นทางอย่างน้อยหนึ่งชนิด")
        resultlist.Visible = True
        pathchoose.Visible = False
        Exit Sub
    End If
    pathchoose.Items(0).Selected = False
    pathchoose.Items(1).Selected = False
    pathchoose.Items(2).Selected = False
    'ตรวจสอบว่าสถานที่ตั้งเส้นทางและปลายทางที่เลือกเป็นสถานที่เดียวกันหรือไม่
    If checkplace() = True Then
        resultlist.DataSource = ""
        resultlist.DataBind()
        resultlist.Items.Add("สถานที่ที่ท่านเลือกเป็นสถานที่เดียวกัน กรุณาเลือกใหม่")
        resultlist.Visible = True
        pathchoose.Visible = False
        Exit Sub
    End If
    Dim x As Integer = 0
    Dim y As Integer = 0
    'ตรวจสอบว่าสถานที่ตั้งเส้นทางและปลายทางตั้งอยู่บนถนนเดียวกันหรือไม่
    Dim sameroad As String = checkroad()
    Dim roadq As String = ""
    If sameroad <> "" Then
        roadq = "Select Distance_L From (Select
Place_Name,Distance_L,Road_ID From Place Where Place_Name = '" +
beginplace.SelectedItem.Text + "' or Place_Name = '" +
endplace.SelectedItem.Text + "') as A Where Road_ID = '" + sameroad +
""

        Dim myconn As New
OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0; Data Source=" +
Server.MapPath("brd.mdb"))
        myconn.Open()
        Dim mycommand As New OleDbCommand(roadq, myconn)
        Dim mydatareader As OleDbDataReader =
mycommand.ExecuteReader()
        Dim temp As Integer = 0
        While mydatareader.Read()
            temp = mydatareader.Item("Distance_L") - temp
        End While
    End If

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    If temp < 0 Then
        temp = temp * -1
    End If
    resultlist.DataSource = ""
    resultlist.DataBind()
    resultlist.Items.Add("ทั้งสองที่อยู่บนถนนเดียวกันและห่างกัน"+Str(temp) + " เมตร")
    resultlist.Visible = True
    pathchoose.Visible = False
    myconn.Close()
    sameroaddraw()
    Exit Sub
End If

Dim endnode As New ArrayList
Dim markednode As String = ""
Dim allpath As New ArrayList

'นำโหนดต้นทางเก็บใน allpath
allpath = findjunc(beginplace.SelectedItem.Text)

'นำโหนดปลายทางเก็บใน endnode
endnode = findjunc(endplace.SelectedItem.Text)
Dim tempallpath As String = ""
Dim appropath As New ArrayList

'ต้องมีการใส่เส้นทางสมมติไว้ใน appropath เพื่อป้องกันการผิดพลาดในการวนรูป
appropath.Add("0JXXXX")
Dim i As Integer = 0
Dim j As Integer = 0

'เก็บเส้นทางที่ผ่านแยกเพียงแยกเดียว (ถ้ามี) ไว้เป็นเส้นทางแรก
For i = 0 To allpath.Count - 1
    For j = 0 To endnode.Count - 1
        If cutcost(allpath(i)) = cutcost(endnode(j)) Then
            appropath.Add(Str(findcost(endnode(j)) +
findcost(allpath(i))) + cutcost(allpath(i)))
        End If
    Next
Next

Dim cost As Integer = 0
Dim tempchoosepath As String = ""
Dim temptotalcost As String = ""
Dim choosepath As String = ""
Dim totalcost As Integer
Dim nextnode As New ArrayList
Dim k As Integer = 0
Dim a As Integer = 0
Dim b As Integer = 0

'เริ่มการทำงานวนรูปหาเส้นทางจนกว่าจะครบ 4 เส้นทาง (รวมเส้นทางสมมติด้วย)
While appropath.Count < 4

    choosepath = allpath(0)
    totalcost = findcost(choosepath)

'นำโหนดสุดท้ายของ choosepath มาเป็น markednode
markednode = findmarkednode(choosepath)

'หาโหนดที่อยู่ถัดจาก markednode
nextnode = findnextnode(markednode, choosepath)

'หาค่า cost ของเส้นทางใหม่แล้วใส่เข้าไปใน allpath หรือ appropath แบบเรียงจากน้อยไปมาก
For i = 0 To nextnode.Count - 1

    cost = findcost(nextnode(i))
    temptotalcost = Str(totalcost + cost)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

tempchoosepath = cutcost(choosepath)
For k = 0 To endnode.Count - 1
    If cutcost(nextnode(i)) <> cutcost(endnode(k)) Then
        a = a + 1
    Else
        temptotalcost = Str(Val(temptotalcost) +
findcost(endnode(k)))
    End If
Next
If a <> endnode.Count Then
    'ถ้าถึงแยกปลายทางให้เก็บใน appropath
    For j = 0 To appropath.Count - 1
        If Val(temptotalcost) <= findcost(appropath(j))
Then
            appropath.Insert(j, temptotalcost +
tempchoosepath + cutcost(nextnode(i)))
            Exit For
        End If
        If j = appropath.Count - 1 Then
            appropath.Add(temptotalcost + tempchoosepath
+ cutcost(nextnode(i)))
        End If
    Next
Else
    'ถ้ายังไม่ใช่แยกปลายทางให้เก็บใน allpath เพื่อคำนวณต่อ
    For j = 0 To allpath.Count - 1
        If Val(temptotalcost) <= findcost(allpath(j))
Then
            allpath.Insert(j, temptotalcost +
tempchoosepath + cutcost(nextnode(i)))
            Exit For
        End If
        If j = allpath.Count - 1 Then
            allpath.Add(temptotalcost + tempchoosepath +
cutcost(nextnode(i)))
        End If
    Next
End If
a = 0
Next
allpath.Remove(choosepath) 'ลบเส้นทางที่คิดไปแล้วออก
'ตรวจสอบว่ายังมีเส้นทางที่ผ่านเพียงแยกเดียวอยู่ใน appropath ให้ลบเส้นทางที่ผ่านสองแยกออก เพื่อป้องกันการ
ผิดพลาดในการหาเส้นทาง
Dim check As Boolean = False
For x = 0 To appropath.Count - 1
    If cutcost(appropath(x)).Length = 5 And
cutcost(appropath(x)) <> "JXXXX" Then
        check = True
    End If
Next
If check = True Then
    For x = 0 To appropath.Count - 1
        If cutcost(appropath(x)).Length = 10 Then
            appropath.RemoveAt(x)
            Exit For
        End If
    Next
For x = 0 To appropath.Count - 1
    If cutcost(appropath(x)).Length = 10 Then

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        appropath.RemoveAt(x)
        Exit For
    End If
Next
End If
End While
appropath.RemoveAt(0) 'ลบเส้นทางสมมติออก
resultlist.DataSource = appropath
resultlist.DataBind()
resultlist.Visible = False 'สามารถให้เป็น true เพื่อตรวจสอบผลลัพธ์ได้
pathchoose.Visible = True
End Sub

'-----checkplace-----ใช้ตรวจสอบว่าสถานที่ที่เลือกเป็นสถานที่เดียวกันหรือไม่-----

Function checkplace() As Boolean
    Dim check As Boolean = False
    If beginplace.SelectedItem.Text = endplace.SelectedItem.Text
Then
        check = True
    End If
    Return check
End Function

'-----check road-----ใช้ตรวจสอบว่าสถานที่ที่เลือกอยู่บนถนนเดียวกันหรือไม่-----

Function checkroad() As String
    Dim roadq As String = "Select Road_ID From Place Where
Place_Name =" + beginplace.SelectedItem.Text + "'"
    Dim myconn As New
OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0; Data Source=" +
Server.MapPath("brd.mdb"))
    myconn.Open()
    Dim mycommand As New OleDbCommand(roadq, myconn)
    Dim mydatareader As OleDbDataReader = mycommand.ExecuteReader()
    Dim beginroad As New ArrayList
    Dim sameroad As String = ""
    While mydatareader.Read()
        beginroad.Add(mydatareader.Item("Road_ID"))
    End While
    myconn.Close()
    roadq = "Select Road_ID From Place Where Place_Name =" +
endplace.SelectedItem.Text + "'"
    Dim myconn2 As New
OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0; Data Source=" +
Server.MapPath("brd.mdb"))
    myconn2.Open()
    Dim mycommand2 As New OleDbCommand(roadq, myconn2)
    Dim mydatareader2 As OleDbDataReader =
mycommand2.ExecuteReader()
    Dim endroad As New ArrayList
    While mydatareader2.Read()
        endroad.Add(mydatareader2.Item("Road_ID"))
    End While
    myconn2.Close()
    Dim i As Integer = 0
    Dim j As Integer = 0
    For i = 0 To beginroad.Count - 1
        For j = 0 To endroad.Count - 1

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        If beginroad(i) = endroad(j) Then
            sameroad = beginroad(i)
        End If
    Next
Next
Return sameroad
End Function

```

'-----findjunction-----ใช้หาโหนดที่ติดกับสถานที่-----'

```
Function findjunc(ByVal nodename As String) As ArrayList
```

'หาแยกที่อยู่ด้านซ้ายของสถานที่'

```

    Dim nodeq As String = "SELECT DISTINCT Left_Junc,
Distance_L,Traffic,Place.Road_ID FROM Place, Road WHERE
Place.Road_ID=Road.Road_ID and Place_Name='" + nodename + "'ORDER BY
Distance_L"

```

```

    Dim myconn As New
OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0; Data Source=" +
Server.MapPath("brd.mdb"))
myconn.Open()
Dim mycommand As New OleDbCommand(nodeq, myconn)
Dim mydatareader As OleDbDataReader = mycommand.ExecuteReader()
Dim result As New ArrayList
Dim path As String = ""
Dim k As Integer = 0
Dim i As Integer = 0
Dim checkjunc As String = ""

```

'ทำงานตามตัวแปรที่เลือก โดย item(0) คือ ระยะทาง และ item(1) คือ สภาพจราจร'

```

Then
    If (parameter.Items(0).Selected And parameter.Items(1).Selected)
        While mydatareader.Read()
            If mydatareader.Item("Left_Junc") <> checkjunc Then
                path = mydatareader.Item("Distance_L") *
mydatareader.Item("Traffic")
                path = path + mydatareader.Item("Left_Junc")
                result.Add(path)
                checkjunc = mydatareader.Item("Left_Junc")
            End If
        End While
    ElseIf (parameter.Items(0).Selected And (Not
parameter.Items(1).Selected)) Then
        While mydatareader.Read()
            If mydatareader.Item("Left_Junc") <> checkjunc Then
                path = mydatareader.Item("Distance_L")
                path = path + mydatareader.Item("Left_Junc")
                result.Add(path)
                checkjunc = mydatareader.Item("Left_Junc")
            End If
        End While
    ElseIf ((Not parameter.Items(0).Selected) And
(parameter.Items(1).Selected)) Then
        While mydatareader.Read()
            If mydatareader.Item("Left_Junc") <> checkjunc Then
                path = mydatareader.Item("Traffic")
                path = path + mydatareader.Item("Left_Junc")
                result.Add(path)
            End If
        End While
    End If

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        checkjunc = mydatareader.Item("Left_Junc")
    End If
End While
End If
myconn.Close()

```

*'หาแยกถัดไปที่อยู่ด้านขวาของสถานที่*

```

    Dim nodeq2 As String = "SELECT DISTINCT Right_Junc, Distance_L,
Road_Length,Traffic,Place.Road_ID FROM Place, Road WHERE
Place.Road_ID=Road.Road_ID and Place_Name='" + nodename + "'"
    Dim myconn2 As New
OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0; Data Source=" +
Server.MapPath("brd.mdb"))
    myconn2.Open()
    Dim mycommand2 As New OleDbCommand(nodeq2, myconn2)
    Dim mydatareader2 As OleDbDataReader =
mycommand2.ExecuteReader()
    Dim cost As Integer = 0

```

*'เก็บไหนดโดยเรียงลำดับตามค่า COST จากน้อยไปมาก*

```

    If (parameter.Items(0).Selected And parameter.Items(1).Selected)
Then
    While mydatareader2.Read()
        path = Str((mydatareader2.Item("Road_Length") -
mydatareader2.Item("Distance_L")) * mydatareader2.Item("Traffic")) +
mydatareader2.Item("Right_Junc")
        For i = 0 To result.Count - 1
            If cutcost(path) <> cutcost(result(i)) Then
                k = k + 1
            End If
        Next
        If k = result.Count Then
            cost = findcost(path)
            For i = 0 To result.Count - 1
                If cost <= findcost(result(i)) Then
                    result.Insert(i, path)
                    Exit For
                End If
                If i = result.Count - 1 Then
                    result.Add(path)
                End If
            Next
        End If
        k = 0
    End While
    ElseIf (parameter.Items(0).Selected And (Not
parameter.Items(1).Selected)) Then
        While mydatareader2.Read()
            path = Str((mydatareader2.Item("Road_Length") -
mydatareader2.Item("Distance_L"))) + mydatareader2.Item("Right_Junc")
            For i = 0 To result.Count - 1
                If cutcost(path) <> cutcost(result(i)) Then
                    k = k + 1
                End If
            Next
            If k = result.Count Then
                cost = findcost(path)
                For i = 0 To result.Count - 1

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        If cost <= findcost(result(i)) Then
            result.Insert(i, path)
            Exit For
        End If
        If i = result.Count - 1 Then
            result.Add(path)
        End If
    Next
End If
k = 0
End While
ElseIf ((Not parameter.Items(0).Selected) And
(parameter.Items(1).Selected)) Then
    While mydatareader2.Read()
        path = Str(mydatareader2.Item("Traffic")) +
mydatareader2.Item("Right_Junc")
        For i = 0 To result.Count - 1
            If cutcost(path) <> cutcost(result(i)) Then
                k = k + 1
            End If
        Next
        If k = result.Count Then
            cost = findcost(path)
            For i = 0 To result.Count - 1
                If cost <= findcost(result(i)) Then
                    result.Insert(i, path)
                    Exit For
                End If
                If i = result.Count - 1 Then
                    result.Add(path)
                End If
            Next
        End If
        k = 0
    End While
End If
myconn2.Close()
Dim temp As String = ""
Return result
End Function

'-----findnextnode from marked node--ใช้หาโหนดที่ติดกับ markednode-----

Function findnextnode(ByVal mnode As String, ByVal choosepath As
String) As ArrayList
    Dim c As Integer = 0
    Dim b As Integer = 0
    Dim beginnode As ArrayList =
findjunc(beginplace.SelectedItem.Text)

'หาโหนดที่อยู่ทางขวาของ markednode

    Dim nextqr As String = "select Right_Junc,Road_length,Traffic
from road where Left_Junc =" + mnode + ""
    Dim myconn As New
OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0; Data Source=" +
Server.MapPath("brd.mdb"))
    myconn.Open()
    Dim mycommand As New OleDbCommand(nextqr, myconn)
    Dim mydatareader As OleDbDataReader = mycommand.ExecuteReader()

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Dim result As New ArrayList
Dim costpath As String = ""
If (parameter.Items(0).Selected And parameter.Items(1).Selected)
Then
    While mydatareader.Read()
        'ตรวจสอบว่าไม่มีกรวนกลับไปหาโหนดเริ่มต้นอีก เพื่อป้องกันการวนลูของงเส้นทาง
        For c = 0 To beginnode.Count - 1
            If cutcost(beginnode(c)) <>
mydatareader.Item("Right_Junc") Then
                b = b + 1
            End If
        Next
        If b = beginnode.Count Then
            costpath = mydatareader.Item("Road_Length") *
mydatareader.Item("Traffic")
            costpath = costpath +
mydatareader.Item("Right_Junc")
            result.Add(costpath)
        End If
        b = 0
    End While
    ElseIf (parameter.Items(0).Selected And (Not
parameter.Items(1).Selected)) Then
        While mydatareader.Read()
            For c = 0 To beginnode.Count - 1
                If cutcost(beginnode(c)) <>
mydatareader.Item("Right_Junc") Then
                    b = b + 1
                End If
            Next
            If b = beginnode.Count Then
                costpath = mydatareader.Item("Road_Length")
                costpath = costpath +
mydatareader.Item("Right_Junc")
                result.Add(costpath)
            End If
            b = 0
        End While
    ElseIf ((Not parameter.Items(0).Selected) And
(parameter.Items(1).Selected)) Then
        While mydatareader.Read()
            For c = 0 To beginnode.Count - 1
                If cutcost(beginnode(c)) <>
mydatareader.Item("Right_Junc") Then
                    b = b + 1
                End If
            Next
            If b = beginnode.Count Then
                costpath = mydatareader.Item("Traffic")
                costpath = costpath +
mydatareader.Item("Right_Junc")
                result.Add(costpath)
            End If
            b = 0
        End While
    End If
myconn.Close()

```

'หาโหนดที่อยู่ทางขวาของ markednode

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Dim myconn2 As New
OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0; Data Source=" +
Server.MapPath("brd.mdb"))
myconn2.Open()
Dim nextql As String = "select Left_Junc,Road_length,Traffic
from road where Right_Junc =' " + mnode + "'
Dim mycommand2 As New OleDbCommand(nextql, myconn2)
Dim mydatareader2 As OleDbDataReader =
mycommand2.ExecuteReader()
Dim costpath2 As String = ""
If (parameter.Items(0).Selected And parameter.Items(1).Selected)
Then
While mydatareader2.Read()
For c = 0 To beginnode.Count - 1
If cutcost(beginnode(c)) <>
mydatareader2.Item("Left_Junc") Then
b = b + 1
End If
Next
If b = beginnode.Count Then
costpath2 = mydatareader2.Item("Road_Length") *
mydatareader2.Item("Traffic")
costpath2 = costpath2 +
mydatareader2.Item("Left_Junc")
result.Add(costpath2)
End If
b = 0
End While
ElseIf (parameter.Items(0).Selected And (Not
parameter.Items(1).Selected)) Then
While mydatareader2.Read()
For c = 0 To beginnode.Count - 1
If cutcost(beginnode(c)) <>
mydatareader2.Item("Left_Junc") Then
b = b + 1
End If
Next
If b = beginnode.Count Then
costpath2 = mydatareader2.Item("Road_Length")
costpath2 = costpath2 +
mydatareader2.Item("Left_Junc")
result.Add(costpath2)
End If
b = 0
End While
ElseIf ((Not parameter.Items(0).Selected) And
(parameter.Items(1).Selected)) Then
While mydatareader2.Read()
For c = 0 To beginnode.Count - 1
If cutcost(beginnode(c)) <>
mydatareader2.Item("Left_Junc") Then
b = b + 1
End If
Next
If b = beginnode.Count Then
costpath2 = mydatareader2.Item("Traffic")
costpath2 = costpath2 +
mydatareader2.Item("Left_Junc")
result.Add(costpath2)
End If
b = 0

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        End While
    End If
    myconn2.Close()
    choosepath = cutcost(choosepath)
    Dim k As Integer = 0
    Dim temp2 As String = ""
    Dim i As Integer = 0
    Dim j As Integer = 0
    Dim lenchoosepath As Integer = choosepath.Length / 5
    Dim templist As New ArrayList
    Dim a As Integer = 0
    Dim tempchoosepath As String = ""
    Dim cutchoosepath As String = cutcost(choosepath)
    Dim temp As String = ""
    k = cutchoosepath.Length
    i = result.Count - 1
    tempchoosepath = cutchoosepath
    'ตรวจสอบว่าโหนดที่ได้ไม่เคยอยู่ใน choosepath เพื่อป้องกันการวนลูปของเส้นทาง
    While i >= 0
        For j = 1 To k / 5
            a = tempchoosepath.Length
            temp = findmarkednode(tempchoosepath)
            temp2 = findmarkednode(result(i))
            If temp = temp2 Then
                templist.Add(i)
            End If
            tempchoosepath = tempchoosepath.Remove((a - 5), 5)
        Next
        tempchoosepath = cutchoosepath
        i = i - 1
    End While
    For j = 0 To templist.Count - 1
        result.RemoveAt(templist(j))
    Next
    Return result
End Function

'----- find cost -----ใช้ตัด string แยกค่า cost ออกจากเส้นทาง-----
Function findcost(ByVal path As String) As Integer
    Dim coststr As String = ""
    Dim i As Integer = 0
    While path(i) <> "J"
        coststr = coststr + path(i)
        i = i + 1
    End While
    Dim cost As Integer = Val(coststr)
    Return cost
End Function

'----find marked node-----ใช้ตัด string แยกโหนดสุดท้ายออกจากเส้นทาง-----
Function findmarkednode(ByVal path As String) As String
    path = Right(path, 5)
    Return path
End Function

'-----Cut cost-----ใช้ตัด string เพื่อจะได้เฉพาะเส้นทาง-----

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Function cutcost(ByVal path As String) As String
    While path(0) <> "J"
        path = path.Remove(0, 1)
    End While
    Return path
End Function

'-----startdraw-----ใช้ reset รูปภาพเมื่อเริ่มโปรแกรมใหม่-----

Sub startdraw()
    Dim bm As New Bitmap("C:\Inetpub\wwwroot\brdpic.jpg")
    Response.ContentType = "image/jpeg"
    bm.Save("pokapo.Jpeg", ImageFormat.Jpeg)
    bm.Dispose()
End Sub

'-----sameroaddraw-----ใช้วาดเส้นทางในกรณีที่มีสถานที่อยู่บนถนนเดียวกัน-----

Sub sameroaddraw()
    Dim bm As New Bitmap("C:\Inetpub\wwwroot\brdpic.jpg")
    Dim coq As String = "Select Co_X,Co_Y From Place Where
Place_Name='" + beginplace.SelectedItem.Text + "'"
    Dim myconn As New
OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0; Data Source=" +
Server.MapPath("brd.mdb"))
    myconn.Open()
    Dim mycommand As New OleDbCommand(coq, myconn)
    Dim mydatareader As OleDbDataReader = mycommand.ExecuteReader()
    Dim begincox As String = ""
    Dim begincoy As String = ""
    While mydatareader.Read()
        begincox = mydatareader.Item("Co_X")
        begincoy = mydatareader.Item("Co_Y")
    End While
    myconn.Close()
    Dim coq2 As String = "Select Co_X,Co_Y From Place Where
Place_Name='" + endplace.SelectedItem.Text + "'"
    Dim myconn2 As New
OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0; Data Source=" +
Server.MapPath("brd.mdb"))
    myconn2.Open()
    Dim mycommand3 As New OleDbCommand(coq2, myconn2)
    Dim mydatareader2 As OleDbDataReader =
mycommand3.ExecuteReader()
    Dim endcox As String = ""
    Dim endcoy As String = ""
    While mydatareader2.Read()
        endcox = mydatareader2.Item("Co_X")
        endcoy = mydatareader2.Item("Co_Y")
    End While
    myconn2.Close()
    Dim g As Graphics = Graphics.FromImage(bm)
    Dim objpen As New Pen(Color.Red, 4)
    Dim objfont As New Font("arial", 15, FontStyle.Bold)
    g.DrawString(beginplace.SelectedItem.Text, objfont, Brushes.Red,
begincox, begincoy)
    g.DrawEllipse(New Pen(Color.Red, 5), New Rectangle(begincox - 3,
begincoy - 3, 6, 6))
    g.DrawLine(objpen, New Point(begincox, begincoy), New
Point(endcox, endcoy))

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        g.DrawEllipse(New Pen(Color.Red, 5), New Rectangle(endcox - 3,
endcoy - 3, 6, 6))
        g.DrawString(endplace.SelectedItem.Text, objfont, Brushes.Red,
endcox, endcoy)
        Response.ContentType = "image/jpeg"
        bm.Save("pokapo.jpeg", ImageFormat.Jpeg)
        bm.Dispose()
    End Sub

'-----Draw---ใช้วาดเส้นทางที่เป็นผลลัพธ์-----

Sub draw(ByVal sender As Object, ByVal e As EventArgs)
    resultlist.SelectedIndex = pathchoose.SelectedIndex
    Dim path As String = cutcost(resultlist.SelectedItem.Text)

    Dim coq As String = "Select Co_X,Co_Y From Place Where
Place_Name='" + beginplace.SelectedItem.Text + "'"
    Dim myconn As New
OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0; Data Source=" +
Server.MapPath("brd.mdb"))
    myconn.Open()
    Dim mycommand As New OleDbCommand(coq, myconn)
    Dim mydatareader As OleDbDataReader = mycommand.ExecuteReader()
    Dim begincox As String = ""
    Dim begincoy As String = ""
    While mydatareader.Read()
        begincox = mydatareader.Item("Co_X")
        begincoy = mydatareader.Item("Co_Y")
    End While
    myconn.Close()

    Dim beginjunc As String = Left(path, 5)
    coq = "Select Co_X,Co_Y From Junction Where Junc_ID='" +
beginjunc + "'"
    Dim myconn2 As New
OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0; Data Source=" +
Server.MapPath("brd.mdb"))
    myconn2.Open()
    Dim mycommand2 As New OleDbCommand(coq, myconn2)
    Dim mydatareader2 As OleDbDataReader =
mycommand2.ExecuteReader()
    Dim junc1cox As String = ""
    Dim junc1coy As String = ""
    While mydatareader2.Read()
        junc1cox = mydatareader2.Item("Co_X")
        junc1coy = mydatareader2.Item("Co_Y")
    End While
    myconn2.Close()
    Dim bm As New Bitmap("C:\Inetpub\wwwroot\brdpic.jpg")
    Dim g As Graphics = Graphics.FromImage(bm)
    Dim objpen As New Pen(Color.Red, 4)
    Dim objfont As New Font("arial", 15, FontStyle.Bold)
    g.DrawString(beginplace.SelectedItem.Text, objfont, Brushes.Red,
begincox, begincoy)
'วาดเส้นเชื่อมระหว่างสถานที่ต้นทางกับโหนดต้นทาง
    g.DrawEllipse(New Pen(Color.Red, 9), New Rectangle(begincox - 5,
begincoy - 5, 10, 10))
    g.DrawLine(objpen, New Point(begincox, begincoy), New
Point(junc1cox, junc1coy))
    Dim temppath As String = path
    Dim junc2cox As String = ""

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Dim junc2coy As String = ""
Dim i As Integer = 0
Dim a As Integer = path.Length / 5

Dim junc1 As String = ""
Dim junc2 As String = ""
If a <> 1 Then
    For i = 0 To a - 2      'วาดรูปวาดเส้นเชื่อม โหนดกับ โหนด ไปเรื่อยๆจนถึง โหนดสุดท้าย
        junc1 = Left(tempPath, 5)
        tempPath = tempPath.Remove(0, 5)
        junc2 = Left(tempPath, 5)
        junc1cox = queryco(junc1, "Co_X")
        junc1coy = queryco(junc1, "Co_Y")
        junc2cox = queryco(junc2, "Co_X")
        junc2coy = queryco(junc2, "Co_Y")
        g.DrawLine(objpen, New Point(junc1cox, junc1coy), New
Point(junc2cox, junc2coy))
        g.DrawEllipse(New Pen(Color.Blue, 5), New
Rectangle(junc1cox - 3, junc1coy - 3, 5, 5))
    Next
    Else
        junc2cox = junc1cox
        junc2coy = junc1coy
    End If

    Dim coq3 As String = "Select Co_X,Co_Y From Place Where
Place_Name='" + endplace.SelectedItem.Text + "'"
    Dim myconn3 As New
OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0; Data Source=" +
Server.MapPath("brd.mdb"))
    myconn3.Open()
    Dim mycommand3 As New OleDbCommand(coq3, myconn3)
    Dim mydatareader3 As OleDbDataReader =
mycommand3.ExecuteReader()
    Dim endcox As String = ""
    Dim endcoy As String = ""
    While mydatareader3.Read()
        endcox = mydatareader3.Item("Co_X")
        endcoy = mydatareader3.Item("Co_Y")
    End While
    myconn3.Close()

    g.DrawString(endplace.SelectedItem.Text, objfont, Brushes.Red,
endcox, endcoy)

    g.DrawLine(objpen, New Point(junc2cox, junc2coy), New
Point(endcox, endcoy))      'วาดเส้นเชื่อมระหว่าง โหนดสุดท้ายกับสถานที่ปลายทาง
    g.DrawEllipse(New Pen(Color.Blue, 5), New Rectangle(junc2cox -
3, junc2coy - 3, 5, 5))
    g.DrawEllipse(New Pen(Color.Red, 9), New Rectangle(endcox - 5,
endcoy - 5, 10, 10))
    Response.ContentType = "image/jpeg"
    bm.Save("pokapo.jpeg", ImageFormat.Jpeg)
    bm.Dispose()
End Sub

'-----queryco-----ใช้หาที่คำนวณ x,y เพื่อเป็นจุดในการวาดเส้น-----

Function queryco(ByVal junc As String, ByVal co As String) As String
    Dim ans As String = ""

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Dim coq As String = "Select " + co + " From Junction Where
Junc_ID =' " + junc + "' "
Dim myconn As New
OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0; Data Source=" +
Server.MapPath("brd.mdb"))
myconn.Open()
Dim mycommand As New OleDbCommand(coq, myconn)
Dim mydatareader As OleDbDataReader = mycommand.ExecuteReader()
While mydatareader.Read()
If co = "Co_X" Then
ans = mydatareader.Item("Co_X")
End If
If co = "Co_Y" Then
ans = mydatareader.Item("Co_Y")
End If
End While
myconn.Close()
Return ans
End Function
</script>

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้