



ระบบสืบค้นข้อความ

TEXT RETRIEVAL SYSTEM



วัน เดือน ปี..... 19 ๗.๓ ๒๐๑๙.....
เลขทะเบียน..... ๐๖๔๙๖๖.....
เลขเรียกหนังสือ..... T๖๗๒๖๖ ๔๖.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาคตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต
สาขาวิศวกรรมคอมพิวเตอร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ปีการศึกษา 25๖7
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบสืบค้นข้อความ
TEXT RETRIEVAL SYSTEM



วิทยานิพนธ์สำหรับปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้า วิทยาเขตเจ้าคุณทหาร ลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
ปีการศึกษา 2537

ปริญญานิพนธ์ปีการศึกษา 2537

ภาควิชา วิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบสืบค้นข้อความ

ผู้จัดทำ

1. นายพีรพล โชติธรรมรัตน์

2. นายวีระชัย บำรุงศรี

(ดร. วรวิทย์ สิมโสภา)

อาจารย์ที่ปรึกษา

ระบบสืบค้นข้อความ

ทิรพล โชติธรรมรัตน์

วิระชัย ปารุงศรี

ดร.วรวุฒน์ ถิ่นโกคา อาจารย์ที่ปรึกษา

ปีการศึกษา 2537

บทคัดย่อ

ระบบการสืบค้นข้อความ เป็นระบบที่สามารถจัดการและบริหารข้อมูลประเภทข้อความ บทความ เอกสาร หนังสือ และสิ่งพิมพ์ต่างๆตามที่ใช้ต้องการ ระบบนี้จะจัดข้อมูลดังกล่าวในรูปแบบ โครงสร้างที่จะช่วยให้ผู้ใช้สามารถค้นหาข้อความต่างๆ ได้สะดวกรวดเร็ว และมีประสิทธิภาพ ระบบนี้ยังได้ถูกพัฒนาในรูปแบบมาตรฐานที่สามารถจะนำไปใช้งานบนระบบคอมพิวเตอร์ต่างๆได้อย่างสะดวก บนวินโดว และระบบนี้ได้พัฒนาบนโครงสร้างที่ไม่ซับซ้อน แต่มีประสิทธิภาพสูง ในปัจจุบันระบบนี้กำลังเป็นที่สนใจอย่างกว้างขวางในงานด้านต่างๆ โดยเฉพาะด้านงาน เอกสารต่างๆทั่วโลก ดังนั้นการพัฒนาระบบนี้ขึ้นมาจึงได้มีการทำให้มีความเหมาะสมในหลายๆ ด้าน โดยระบบนี้ได้ถูกออกแบบอย่างไม่ซับซ้อนโดยที่ผู้สนใจสามารถศึกษาได้โดยง่าย และการใช้งานก็ไม่ซับซ้อนนัก และการพัฒนาระบบนี้เป็นเรื่องที่น่าสนใจอย่างยิ่ง โดยเฉพาะการนำมาใช้ใน ระบบภาษาไทยของเราเอง

TEXT RETRIEVAL SYSTEM

Peerapon Chothammarat

Weerachai Bumrungrsi

Worawat Limpoka Advisor

1994

Abstract

Text Retrieval System is a text data management system. It can be used with any kind of text-type documents such as books , documents , print materials , newspaper etc. This system manage all text in a convenience structure to help the user to search any text easily and quickly. This system was developed in a standard form which can work on other systems easily with a few changes even on Windows. It was also designed on a transparent structures with high integrity and quality. Today, many text retrieval system are used in many places around the world. So , we designed it clearly to help anyone who interested in this system can study it easily and we also made it to work with Thai language.

สารบัญ

บทที่ 1 บทนำ	1
บทที่ 2 การวิเคราะห์ข้อความสืบค้น	2
2.1 การวิเคราะห์ข้อความสืบค้น	2
2.2 รูปแบบไวยากรณ์แบบคัสเนอร์	2
2.3 รูปแบบไวยากรณ์ของข้อความสืบค้น	3
2.4 ลักษณะของข้อความสืบค้น	4
2.5 ขั้นตอนการวิเคราะห์ข้อความสืบค้น	8
บทที่ 3 ทฤษฎีของวิชาวลเบสิกบนระบบปฏิบัติการวินโดวส์	11
3.1 โครงงาน	11
3.2 พร็อพเพอร์ตี้	13
3.3 ขุดบ็อก	13
3.4 ตัวโปรแกรม	13
3.5 การออกแบบเมนูของระบบ	15
3.6 ไอคอนบ็อก	15
3.7 มัลติเทิลค็อกคิวเมนต์อินเตอร์เฟซ	18
บทที่ 4 การค้นหาคำในฐานข้อมูล	20
4.1 การค้นหาในข้อความสืบค้นปกติ	20
4.2 การค้นหาในข้อความสืบค้นที่มีตัวปฏิบัติการปกติ	20
4.3 การค้นหาในข้อความสืบค้นที่มีตัวปฏิบัติการพิเศษ	22

บทที่ 5 การทดลองและการแสดงผล	25
5.1 การแสดงผลข้อมูลในเอกสารในการสืบค้นปกติและตัวปฏิบัติการปกติ	25
5.2 การแสดงผลของข้อความสืบค้นที่มีตัวปฏิบัติการพิเศษ	26
5.3 การแสดงผลของรูปแบบของระบบ	29
บทที่ 6 บทวิจารณ์และสรุปผล	31
หนังสืออ้างอิง	32



บทที่ 1

บทนำ

ระบบสืบค้นข้อความ (Text Retrieval System) จัดเป็นระบบจัดฐานข้อมูลแบบหนึ่งซึ่งลักษณะของข้อมูลที่จัดเก็บจะเป็นข้อความ เช่นสารานุกรม ข่าวจากหนังสือพิมพ์ หนังสือ ฯลฯ

ลักษณะการจัดเก็บแบบนี้จะแตกต่างจากระบบจัดการฐานข้อมูลแบบอื่นๆ คือ ในแบบรีเลชันแนล (Relational) แบบไฮเออราซิคอล (Hiaerarchical) และแบบเน็ตเวิร์ค (Network) ซึ่งลักษณะของข้อมูลที่จัดเก็บในรูปแบบเหล่านี้จะมีโครงสร้างเป็นระเบียบแบบแผนที่แน่นอน ซึ่งไม่เหมาะกับการจัดเก็บข้อมูลประเภทข้อความ

ในระบบสืบค้นข้อความ ข้อมูลส่วนใหญ่ที่มักนำมาจัดเก็บจะเป็นลักษณะของการเก็บไว้เป็นการถาวร และ การใช้งานส่วนใหญ่จะมีวัตถุประสงค์หลักเพื่อการค้นหาข้อความ ให้มีความสะดวกรวดเร็ว และมีรูปแบบการค้นหาที่หลากหลาย

ดังนั้นการพัฒนาแบบนี้ขึ้นมาจะเน้นตามวัตถุประสงค์ที่ได้กล่าวมาแล้วข้างต้น โดยในระบบที่พัฒนานี้ จะมีความสามารถในการค้นหาในลักษณะต่างๆ ได้แก่ การค้นหาคำโดด การค้นหาคำโดยใช้ตัวปฏิบัติการ ซึ่งจะทำให้ขอบเขตของการค้นหากว้างขึ้นหรือแคบลง หรือการค้นหาโดยที่ผู้ใช้งานเพียงส่วนหน้าหรือส่วนหลังของคำก็ได้

ระบบสืบค้นข้อความนี้พัฒนาในรูปแบบมาตรฐานนี้สามารถจะนำไปใช้บนระบบปฏิบัติการวินโดวส์ (Windows) ดังนั้นการพัฒนาแบบนี้ได้มีการทำให้มีความเหมาะสมในหลายๆด้าน โดยระบบถูกออกแบบอย่างไม่ซับซ้อน โดยที่ผู้สนใจจะสามารถศึกษาและใช้งานได้โดยง่าย

บทที่ 2

การวิเคราะห์ข้อความสืบค้น

2.1 การวิเคราะห์ข้อความสืบค้น (Query Analysis)

ส่วนสำคัญในการออกแบบและพัฒนาระบบสืบค้นข้อความ (Text retrieval system) คือ ส่วนของการวิเคราะห์ข้อความสืบค้น (Query) ซึ่งจะเป็นส่วนที่แสดงให้เห็นถึงความสามารถของระบบนั้นๆ โดยโครงสร้างของการวิเคราะห์

เนื่องจากข้อความสืบค้นมีด้วยกัน 3 ชนิด ซึ่งจะกล่าวถึงในรูปแบบไวยากรณ์ของข้อความสืบค้น ซึ่งแต่ละประเภทจะต้องมีข้อกำหนดของแต่ละรูปแบบ ทำให้เราต้องกำหนดโครงสร้างของข้อความสืบค้น โดยอาศัย ทฤษฎีต่างๆดังนี้

นิยาม เมต้าแลงกเวจ (Metalinguage) คือ ระบบหรือภาษาชนิดหนึ่ง ซึ่งสามารถใช้เขียนอธิบายให้สามารถเข้าใจภาษาอื่นๆได้

ในการวิเคราะห์ความหมายข้อความสืบค้นเราจำเป็นต้องมีการกำหนดรูปแบบของ เมต้าแลงกเวจ เพื่อที่จะสามารถเข้าใจความหมายของข้อความสืบค้นได้ง่ายขึ้น เพราะรูปแบบของข้อความสืบค้นที่ผู้ใช้ทำการสอบถามกับระบบนั้นจะง่ายต่อการเข้าใจของผู้ใช้ แต่มีความซับซ้อนสำหรับคอมพิวเตอร์

2.2 รูปแบบไวยากรณ์แบคคัสเนอร์ (Backus Naur Form) (BNF)

แบคคัสเนอร์ เป็นรูปแบบเมต้าแลงกเวจ ที่เราพิจารณาเลือกไว้ใช้ในการกำหนดโครงสร้างไวยากรณ์ของข้อความสืบค้นของระบบสืบค้นข้อความ โดยเลือกใช้ในรูปแบที่เพิ่มเติมแบคคัสเนอร์ (Extended BNF Notation) ซึ่งรูปแบบของแบคคัสเนอร์นั้นอธิบายไว้ดังนี้

สัญลักษณ์ที่ใช้	ความหมาย
::=	ถูกกำหนดให้เป็น (is defined as)
< >	สัญลักษณ์อนเทอร์มินอล (Nonterminal Symbol)
{ }	แสดงจำนวนตั้งแต่ 0 ขึ้นไป
[]	แสดงจำนวน 0 หรือ 1
	หรือ (or)
()	ให้เลือกตัวใดตัวหนึ่ง
X	สัญลักษณ์เทอร์มินอล (Terminal Symbol)

ตารางที่ 2.1 ตารางสัญลักษณ์ที่ใช้ในรูปแบบแบคคัสเนอร์

2.3 รูปแบบไวยากรณ์ของข้อความสืบค้น

ในการออกแบบไวยากรณ์สำหรับข้อความสืบค้นนั้นเราได้ทำการออกแบบไว้ซึ่งสามารถอธิบายด้วยรูปแบบไวยากรณ์ของแบคคัสเนอร์ได้ดังรูป

< Query >	::= Word < Operation_Query > < Wildcard_Query >
< Operation_Query >	::= '(' < Operation_Query > < Operator > < Operation_Query > ')' Word
< Operator >	::= ^ /
< Wildcard_Query >	::= '*' < Wild1 > '?' < Wild2 > < Wild2 > '?' < Wild1 > '?'
< Wild1 >	::= Word '*' < Wild1 > '?' < Wild2 >
< Wild2 >	::= Word '?' < Wild2 >

รูปที่ 2.1 รูปแบบไวยากรณ์ของข้อความสืบค้น

ในที่นี้เครื่องหมาย ^ , / , * , ? และ Word เป็นสัญลักษณ์เทอร์มินอล ซึ่งก็เป็นคำและเครื่องหมายที่จะใช้ในการสืบค้นนั่นเอง

2.4 ลักษณะของข้อความสืบค้น

จากรูปแบบไวยากรณ์ที่อธิบายในรูปที่ 2.1 นั้นจะสามารถแบ่งลักษณะข้อความสืบค้นได้ดังต่อไปนี้

2.4.1 ข้อความสืบค้นปกติ (Normal Query)

ข้อความประเภทนี้จะมีลักษณะคำเดี่ยว (Single Word) สำหรับการค้นหา ซึ่งจะต้องใช้ไวยากรณ์

$\langle \text{Query} \rangle ::= \text{Word}$

เช่น computer , engineering , apple , the , game และ คำอื่นๆ

2.4.2 ข้อความสืบค้นที่มีตัวปฏิบัติการปกติ (Normal Operation Query)

ข้อความประเภทนี้จะมีความสามารถในการค้นหากลุ่มคำหรือข้อความที่ผู้ใช้ต้องการในรูปแบบต่างๆกัน ดังนี้

2.4.2.1 การสืบค้นด้วยตัวปฏิบัติการ “และ” (AND Operator)

จากที่ได้กำหนดรูปแบบของโครงสร้างของฐานข้อมูลสำหรับการสืบค้นแล้วว่าในแต่ละฐานข้อมูล จะถูกแบ่งเป็นเอกสาร (Document) ซึ่งเปรียบได้กับหนังสือหนึ่งเล่ม และแต่ละเอกสารจะแบ่งออกเป็นหน้า (Page) ซึ่งแต่ละหน้าในฐานข้อมูลจะมีขนาดเท่ากับจำนวนบรรทัดของแต่ละระบบแสดงผล

การสืบค้นโดยตัวปฏิบัติการ “และ” ก็คือการค้นหาคำหรือกลุ่มคำที่อยู่ในเอกสารชุดเดียวกัน โดยเมื่อค้นพบคำหรือกลุ่มคำที่อยู่ในเอกสารเดียวกันแล้วจึงจะแสดงผลให้ทราบ

เช่น เอกสารหนึ่งเป็นหนังสือเรื่อง “Compiler Writing” ผู้ใช้ต้องการสืบค้นหาเรื่องเกี่ยวกับไวยากรณ์ “Grammar” กับการวิเคราะห์ “Analysis” ผู้ใช้ก็สามารถป้อนข้อความสืบค้นดังนี้

เช่น $\text{Grammar} \wedge \text{Analysis}$

เราจะกำหนดให้ใช้เครื่องหมาย \wedge แทนคำว่า “และ” และคำ ระหว่างที่ต้องการสืบค้นหา จะต้องไม่มีการเว้นช่องไปด้วย

จากข้อความสืบค้นดังกล่าว ผลลัพธ์ที่ได้จะเป็นการแสดงข้อความคำว่า Grammar กับคำว่า Analysis ที่อยู่ในเอกสารเดียวกัน โดยแสดงเอกสารที่สองคำนั้นปรากฏอยู่ เอกสารอื่นที่ไม่ปรากฏคำทั้งสองอยู่ด้วยกันจะไม่แสดงผลออกมา หรือการที่ต้องการสืบค้นโดยใช้ตัวปฏิบัติการ “และ” หลายตัวในการสืบค้นคำหลายๆคำ

เช่น $Grammar^Analysis^Computer^PC$

2.4.2.2 การสืบค้นด้วยตัวปฏิบัติการ “หรือ” (OR Operation)

บางครั้งเราอาจต้องการค้นหาหัวข้อซึ่งอาจจะมีมากกว่าเรื่องเดียว เราจะกำหนดให้การค้นหาด้วยตัวปฏิบัติการ “หรือ” เรากำหนด ให้ใช้เครื่องหมาย / แทนคำว่า “หรือ”

เช่น $Computer/PC$

เอกสารใดที่ปรากฏคำใดคำหนึ่งหรือกลุ่มคำใดกลุ่มคำหนึ่ง หรือ ทั้งสองคำหรือทั้งสองกลุ่มคำก็จะถูกแสดงผล หรือการที่ต้องการสืบค้นโดยใช้ตัวปฏิบัติการ “หรือ” หลายตัวในการสืบค้นคำใดคำหนึ่งในคำหลายๆคำ

เช่น $Grammar/Analysis/Computer/PC$

2.4.2.3 การสืบค้นโดยตัวปฏิบัติการหลายประเภท

นอกจากการใช้ตัวปฏิบัติการดังกล่าวไว้แล้วเรายังสามารถนำเอาตัวปฏิบัติการเหล่านั้นมาใช้รวมกันได้อีกด้วย

เช่น $(communication /computer)^network$
 $(connection^pc)^(network/computer)$

เครื่องหมาย	ความหมาย	การใช้งาน
^	และ(AND)	ค้นหาคำหรือกลุ่มคำซึ่งต้องปรากฏในเอกสารเดียวกัน ทั้งสองคำหรือทั้งสองกลุ่มคำเท่านั้น
/	หรือ(OR)	ค้นหาคำหรือกลุ่มคำซึ่งจะปรากฏในเอกสารคำใดคำหนึ่งหรือกลุ่มคำใดกลุ่มคำหนึ่ง

ตารางที่ 2.2 ตารางสัญลักษณ์ ความหมาย และการใช้งานของตัวปฏิบัติการ

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานในท้องถิ่น มิใช่เผยแพร่ให้บุคคลอื่นโดยไม่ได้รับอนุญาต การนำเอกสารนี้ไปใช้โดยไม่ได้รับอนุญาตถือว่าผิดกฎหมาย ผู้ที่ฝ่าฝืนจะถูกลงโทษตามกฎหมาย

2.4.3 ข้อความสืบค้นที่มีตัวปฏิบัติการพิเศษ (Wildcard Operation Query)

ในที่นี้เราได้ใช้ สัญลักษณ์พิเศษ 2 ชนิด ได้แก่

- '*' แทน จำนวนตัวอักษรที่ละไว้ตั้งแต่ 0 ตัวขึ้นไป
- '?' แทน จำนวนตัวอักษร 1 ตัว

2.4.3.1 ตัวปฏิบัติการ *

ใช้ค้นหาได้ 2 กรณี

2.4.3.1.1 กรณีตัวปฏิบัติการ * อยู่ด้านหลัง

วิธีนี้มีลักษณะเช่น com* ผลลัพธ์ที่ออกมานั้นจะเป็นการ
แสดงคำที่ขึ้นต้นด้วยคำว่า com เช่น ได้คำว่า communication , computer
และ compiler เป็นต้น

2.4.3.1.2 กรณีตัวปฏิบัติการ * อยู่ด้านหน้า

โดยวิธีนี้เราสามารถที่จะค้นหาคำที่เราไม่ทราบได้ว่านำหน้าด้วย
ตัวอักษรใด หรือ เราอาจจะต้องรู้ว่าลงท้ายด้วยคำแบบนี้จะมีคำว่าอะไร
อยู่บ้าง เช่น *tion ผลลัพธ์ที่ออกมานั้นจะเป็นการแสดงคำที่ลงท้ายด้วยคำ
ว่า tion เช่น ได้คำว่า computation , section , animation , selection และ
communication เป็นต้น

2.4.3.2 ตัวปฏิบัติการ ?

การใช้ ? ในการค้นหาคำนั้น จำนวน ? จะหมายถึงจำนวนตัวอักษรที่ไม่
แน่ใจว่าจะเป็นตัวอะไร จำนวน 1 ตัว ใช้ค้นหาได้ 2 กรณี เช่นกัน

2.4.3.2.1 กรณีตัวปฏิบัติการ ? ด้านหลัง

เช่น จะสืบค้นคำว่า com???? อาจจะได้ผล เช่น computer ,
commuter เป็นต้น

2.4.3.2.2 กรณีตัวปฏิบัติการ ? ด้านหน้า

เช่นจะสืบค้นคำว่า ?????er อาจจะได้ผล เช่น computer ,
commuter เป็นต้น

จะเห็นได้ว่าทั้งแบบ * และ ? จะไม่อนุญาตให้มีการใช้เครื่องหมาย
ระหว่างกลางของคำ เช่น c*ter com???er เนื่องจากมีความซับซ้อนสูง

2.4.3.3 การใช้ตัวปฏิบัติการพิเศษร่วมกัน

กรณีนี้เราจะแบ่งได้คล้ายกับการแบ่งใน 2.4.3.1 และ 2.4.3.2 คือ

2.4.3.3.1 กรณีที่มี * นำหน้า

กรณีนี้ เช่น *??ter คือต้องการคำที่มีความยาวตั้งแต่ 5 อักษรขึ้นไป
ไปที่ตามท้ายด้วย ter

2.4.3.3.2 กรณีที่มี * ตามหลัง

กรณีนี้ เช่น com???* คือต้องการคำที่ขึ้นต้นด้วย com ที่มีความยาวตั้งแต่ 6 อักษรขึ้นไป

2.4.3.3.3 กรณีที่มี ? นำหน้า

กรณีนี้ เช่น ??*ter คือต้องการคำที่มีความยาวตั้งแต่ 5 อักษรขึ้นไป
ไปที่ตามท้ายด้วย ter มีความหมายเช่นเดียวกับ 2.4.3.3.1

2.4.3.3.4 กรณีที่มี ? ตามหลัง

กรณีนี้ เช่น com???* คือต้องการคำที่ขึ้นต้นด้วย com ที่มีความยาวตั้งแต่ 6 อักษรขึ้นไป มีความหมายเช่นเดียวกับ 2.4.3.3.2

เครื่องหมาย	ความหมาย
*	อักขระตั้งแต่ 0 ตัวขึ้นไป
?	อักขระใดๆ หนึ่งตัว

ตารางที่ 2.3 ตารางสัญลักษณ์ และ ความหมายของตัวปฏิบัติการพิเศษ

2.4.4 ข้อความสืบค้นที่มีลักษณะใกล้เคียง

ข้อความสืบค้นประเภทนี้จะสามารถค้นหาคำที่ใกล้เคียงกับข้อความสืบค้นที่ต้องการค้นหา เราจะคิดเป็น เปอร์เซ็นต์ (Percent) ของพยางค์ที่ผิดในที่นี้ใช้ 20 เปอร์เซ็นต์ เช่น ในข้อความสืบค้น ที่มี 10 พยางค์ ก็สามารถหาคำที่ใกล้เคียงที่มีพยางค์ที่แตกต่างได้ไม่เกิน 2 พยางค์

ดังตัวอย่างเช่น ต้องการสืบค้นข้อความคำว่า “computer”

ข้อความที่ค้นหาพบเช่นคำว่า commuter , computtr , copputer เป็นต้น

2.5 ขั้นตอนการวิเคราะห์ข้อความสืบค้น

2.5.1 การรับข้อความสืบค้นเข้ามา

เราจะทำการแปลงข้อความสืบค้นโดยอาศัยทฤษฎีคิงที่กล่าวมาแล้วในเรื่อง การวิเคราะห์ข้อความสืบค้น

2.5.2 ทำการวิเคราะห์เพื่อแยกประเภทข้อความสืบค้น

ประเภทของข้อความสืบค้น มี 3 ประเภทดังนี้

- **ประเภทข้อความสืบค้นปกติ** ข้อความประเภทนี้จะมีลักษณะเป็นคำเดี่ยว ไม่ว่าจะ เป็น คำ ตัวเลข อื่นๆ

- **ประเภทข้อความสืบค้นที่มีตัวปฏิบัติการ** ในข้อความแบบนี้เราจะทำการค้นหา คำว่ามีตัวปฏิบัติการปรากฏอยู่หรือไม่ ถ้ามีก็จะทำการแปลงข้อความสืบค้นที่มีให้อยู่ในรูป แถวลำดับ (array) ของ จำนวนเต็มแบบสั้น (short int) โดยใช้ค่าระหว่าง 95 - 254 ซึ่ง เมื่อเทียบใน ตารางรหัส แอสกีแล้วจะไม่ปนกับส่วนที่เป็นตัวอักษรและตัวอักษรบาง ส่วนที่จะถูกสงวนไว้ใช้เป็นสัญลักษณ์พิเศษในการสืบค้น สามารถบรรจุได้มากถึง 160 คำตามทฤษฎี แถวลำดับแต่ละตัวจะแทนด้วยเลขจำนวนคี่ที่กล่าวไว้ โดยที่จะมีแถวลำดับ อีกชุดหนึ่งซึ่งจะทำการเก็บคำสืบค้นไว้ตามลำดับ 0 - 159 ซึ่งจะทำให้สามารถทราบได้ว่า ในแถวลำดับของสัญลักษณ์ค่าจำนวนที่แทนค่านั้นแทนคำใด

สิ่งที่สำคัญที่สุดคือการแปลงรูปแบบสัญลักษณ์ในแถวลำดับให้อยู่ในรูปของโพสฟิกต์ (Postfix or Reverse Polish notation or Suffix) ซึ่งจะกล่าวในเรื่องของการแปลง ข้อความสืบค้นให้อยู่ในรูปของโพสฟิกต์ซึ่งรูปแบบโพสฟิกต์จะทำให้การตีความ หมายข้อความสืบค้นที่มีความยาวและมีความซับซ้อนสูง

2.5.3 การแปลงข้อความสืบค้นไปสู่รูปแบบโพสฟิกต์

การแปลงนี้มีความจำเป็นอย่างยิ่งสำหรับข้อความแบบที่มีตัวปฏิบัติการปกติ เมื่อ เราได้แถวลำดับสัญลักษณ์แล้วเราจะทำการแปลงเป็นรูปแบบโพสฟิกต์เพื่อง่ายต่อการ แปรความหมายโดยอาศัยวิธีการแปลงอินฟิกต์ไปสู่โพสฟิกต์ หรือ รีเวอร์สโพลิสโน เตชั่น (Reverse Polish Notation)

โพลิสโนเตชั่น (Polish Notation) อาจรู้จักกันในชื่อของ ลูคาเซียวิกเซียน โน เตชั่น (Lukasiewiczian Notation) เพื่อให้เกียรติแก่นักตรรกวิทยาชาวโปแลนด์ ชื่อ



Jan Lukasiwicz วิธีก็คือการเปลี่ยนรูปแบบในการคำนวณเช่น บวก ลบ ให้อยู่ในรูปแบบที่ ง่ายขึ้นโดยมีข้อกำหนดดังนี้

1. ทุกอักขระและจำนวนเต็มบวกเป็นสิ่งที่ใช้ได้ถูกต้องในข้อความอินพิกซ์
2. ถ้า A และ B ถูกต้องแล้ว $(A+B), (A-B), (A*B), (A/B)$ ก็สามารถใช้ได้ถูกต้อง
3. รูปแบบอินพิกซ์ที่ถูกต้องคือในข้อ 1 และ ข้อ 2 เท่านั้น

จะเห็นได้ว่าวงเล็บถูกนำมาใช้ด้วยเช่นกัน แต่จะต้องใส่อย่างถูกต้องคือครบคู่ และการตีความในกรณีที่ไม่มีวงเล็บเป็นแบบซ้ายขวา โดยที่ * และ / มีลำดับ (Precedence) ที่สูงกว่า + และ -

ระดับวงเล็บ (Parenthical level) ของตัวปฏิบัติการใดๆจะมีค่าเท่ากับจำนวนวงเล็บที่ล้อมตัวปฏิบัติการนั้นๆไว้

การตรวจสอบอินพิกซ์ที่มีวงเล็บอยู่เป็นเรื่องที่ไม่ง่ายนัก ดังนั้นการแปลงให้อยู่ในรูปโพสท์ฟิสิกส์หรือพรีฟิสิกส์ จะตัดวงเล็บออก ทำให้ง่ายขึ้น

ในที่นี้เราจะกล่าวถึงการแปลงอินพิกซ์เป็นโพสท์ฟิสิกส์ และ ในตารางเป็นตัวอย่างการแปลงตามหลักการข้อกำหนดทั้ง 3 ข้อที่กล่าวไว้

อินพิกซ์	โพสท์ฟิสิกส์
X-Y	XY-
X-Y+Z	XY-Z+
$(W+X)*(Y+Z)$	$WX+YZ+*$

ตารางที่ 2.4

ขั้นตอนการแปลงเป็นโพสท์ฟิสิกส์

จะต้องมีการกำหนดแสดก โดยที่ตำแหน่งบนสุดมีสัญลักษณ์พิเศษ เช่น # แสดกถึงตำแหน่งแรกของแสดกไว้ ขั้นตอนที่เหลือจะเกี่ยวกับการดูค่าพรีซิเดนส์ (Precedence value) ของสัญลักษณ์ ในโพสท์ฟิสิกส์ โดยถ้าค่าพรีซิเดนส์ของสัญลักษณ์ที่อยู่บนสุดของแสดก สัญลักษณ์ที่อยู่บนสุดของแสดก สัญลักษณ์ใหม่ก็จะถูกใส่ลงแสดก (Push) แต่ถ้าน้อยกว่า หรือ เท่ากันก็จะทำการเอาออกจากแสดก (Pop) และนำไปใส่ในข้อความที่เป็นรูปโพสท์ฟิสิกส์

เมื่อได้แถวลำดับของโพสท์ฟิสิกส์แล้วจะทำการแปลงไปสู่แถวลำดับขั้นสุดท้าย

เพื่อที่จะค้นหาจริง

2.5.4 การตีความหมายรูปแบบโพสท์ฟิกซ์

มีขั้นตอนก่อนการในการตีความหมายของโพสท์ฟิกซ์ดังนี้

ขั้นตอนการตีความและลดรูปโพสท์ฟิกซ์

1. หาตัวปฏิบัติการที่อยู่ซ้ายมือสุด
2. เลือกเอาโอเปอเรนด์ (Operand) ที่นำหน้าตัวปฏิบัติการจากข้อ 1 มาสองตัว
3. นำโอเปอเรนด์มาทำตามตัวปฏิบัติการ
4. แทนที่ตัวปฏิบัติการและโอเปอเรนด์ทั้งสองด้วยผลลัพธ์ลงในโพสท์ฟิกซ์

เมื่อเราสามารถแปลงให้เป็นรูปโพสท์ฟิกซ์ได้แล้ว เราก็จะนำไปผ่านกระบวนการตีความหมายรูปแบบโพสท์ฟิกซ์ให้เป็นคำสั่งการทำงานในคอมพิวเตอร์ เพื่อทำการค้นหาแล้วก็จะแยกประเภทการค้นหา โดยเราจะสามารถแบ่งรูปแบบการค้นหา ตามคำสั่งของตัวปฏิบัติการ

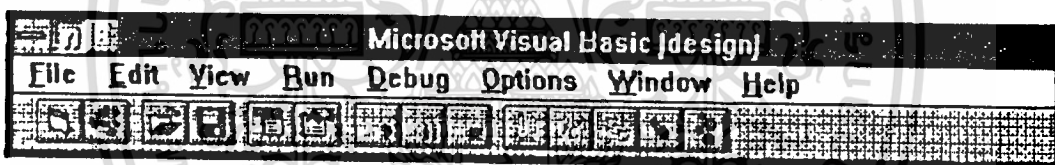


บทที่ 8

ทฤษฎีของวิช่วลเบสิกบนระบบปฏิบัติการวินโดวส์

ระบบสืบค้นข้อความนี้จะทำงานบนระบบปฏิบัติการไมโครซอฟต์วินโดวส์ ซึ่งความหมายของไมโครซอฟต์วินโดวส์เป็นซอฟต์แวร์ประเภทเอนไวรอนเมนต์ (environment) ที่ทำงานในโหมดกราฟิก หรือ กราฟิกเอนไวรอนเมนต์ (graphical environment) คำว่าเอนไวรอนเมนต์นี้หมายถึงลักษณะที่มาครอบตัวโอเปอเรติงซิสเต็ม (operating system) หรือ คอส (DOS) อีกทีหนึ่ง ซึ่งจะทำให้ผู้ใช้สามารถใช้เครื่องคอมพิวเตอร์ที่มีอยู่ได้ง่ายและมีประสิทธิภาพมากขึ้นกว่าการทำงานบนเท็กโหมดของไอเอสเอ็ม คือ คอส ประสิทธิภาพที่เห็นได้ชัดเจนของเอนไวรอนเมนต์ตัวนี้ก็เช่น การจัดการหน่วยความจำที่ขอดีเยี่ยมซึ่งสามารถใช้งานส่วนที่เกิน 640 กิโลไบต์ การทำงานแบบมัลติทาสกิง (multitasking) การสื่อสารกับผู้ใช้ด้วยภาพ และ อื่นๆอีกมากมาย

ส่วนแบบหลักของไมโครซอฟต์วิช่วลเบสิกจะประกอบด้วยเมนูบาร์ (Menu bar) และ ทูลบาร์ (Tool bar) ซึ่งจะมีลักษณะดังรูปที่ 3.1

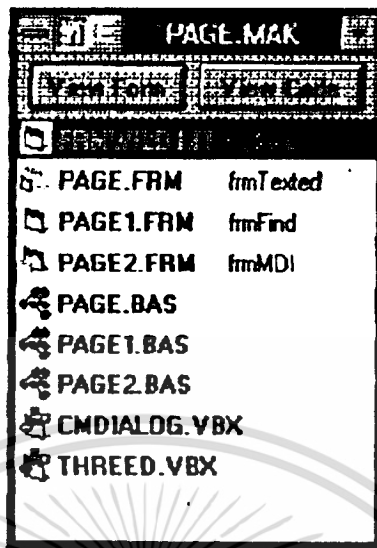


รูปที่ 3.1 รูปแบบหลักของไมโครซอฟต์ไมโครซอฟต์วิช่วลเบสิก

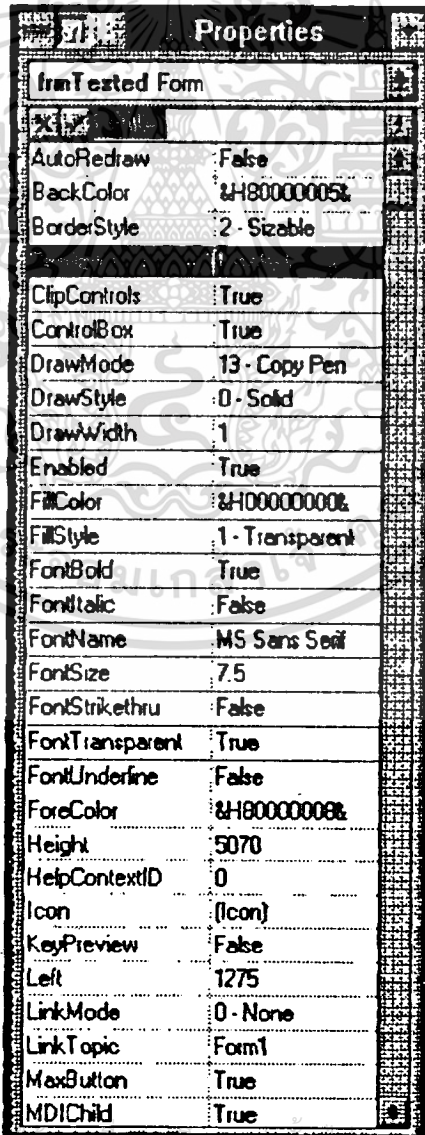
ระบบสืบค้นข้อความจะใช้โปรแกรมภาษาไมโครซอฟต์วิช่วลเบสิกเพราะเป็นโปรแกรมที่ใช้สำหรับบนระบบปฏิบัติการวินโดวส์โดยเฉพาะ ส่วนของรูปแบบไวยากรณ์ของภาษาวิช่วลเบสิกจะมีลักษณะคล้ายควิกเบสิก (quick basic) มาก เราจะกล่าวถึงแต่ทฤษฎีของวิช่วลเบสิกที่แตกต่างจากใช้ภาษาอื่นเขียน โดยเฉพาะส่วนที่ระบบสืบค้นข้อความใช้ ระบบสืบค้นข้อความจะใช้ส่วนประกอบหลักของไมโครซอฟต์วิช่วลเบสิกที่สำคัญๆดังนี้

8.1 โครงการงาน (Project)

โครงการงานในที่นี้ก็จะรวมเพิ่มข้อมูลต่างๆของโปรแกรมที่ทำการเขียนขึ้น โดยจะประกอบด้วย เพิ่มรูปแบบ (Form file) เพิ่มโมดูล (Module file) และ ควบคุมคอนโทรล (Custom Control) โครงการงานจะเป็นไปตามรูปที่ 3.2 ซึ่งจะเป็นตัวอย่างของ ระบบสืบค้นข้อความ



รูปที่ 3.2 รูปของโครงการระบบสืบค้นข้อความ



รูปที่ 3.3 รูปตัวอย่างของทรีอ็อปเทร็คของแอปเจ็ท

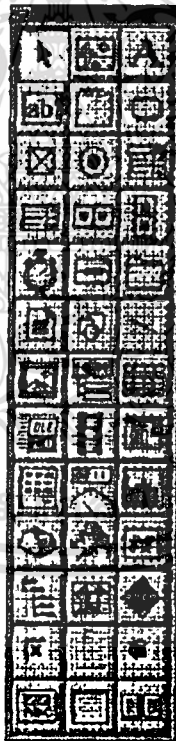
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับดูและใช้เท่านั้น ไม่ให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 พร็อพเพอร์ตี้ (Property)

เป็นส่วนประกอบต่างๆของออบเจ็ก (Object) ซึ่งแต่ละออบเจ็กก็จะมีพร็อพเพอร์ตี้เป็นของตัวเองไม่ว่าจะเป็น สี , ขนาด , ตำแหน่ง , รูป และอื่นๆอีกมาก แล้วแต่ตัวออบเจ็กนั้นๆ ลักษณะของ พร็อพเพอร์ตี้จะเป็นไปตามรูปที่ 3.3 ซึ่งจะเป็นตัวอย่าง พร็อพเพอร์ตี้ของ รูปแบบ(From)

3.3 ทูลบ็อก (Tools Box)

จะเป็นสิ่งที่บรรจุ ออบเจ็กควบคุม (Control object) ออบเจ็กควบคุมนี้จะหมายถึงตัวควบคุมการทำงานของโปรแกรมที่จะทำการเขียนขึ้น หรือในที่นี้ก็คือตัวควบคุมต่างๆในระบบสืบค้นข้อความนั่นเอง ลักษณะของทูลบ็อกจะเป็นไปดังรูปที่ 3.4



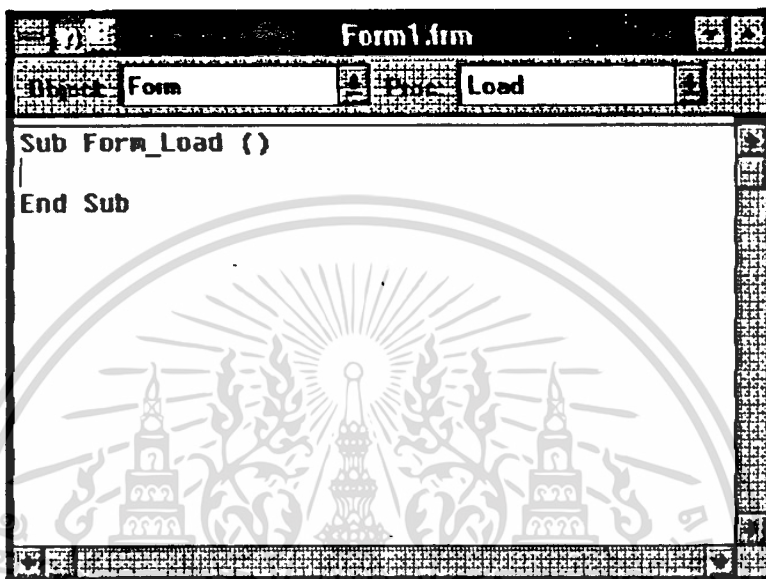
รูปที่ 3.4 รูปออบเจ็กต่างๆในทูลบ็อก

3.4 ตัวโปรแกรม (Attaching code)

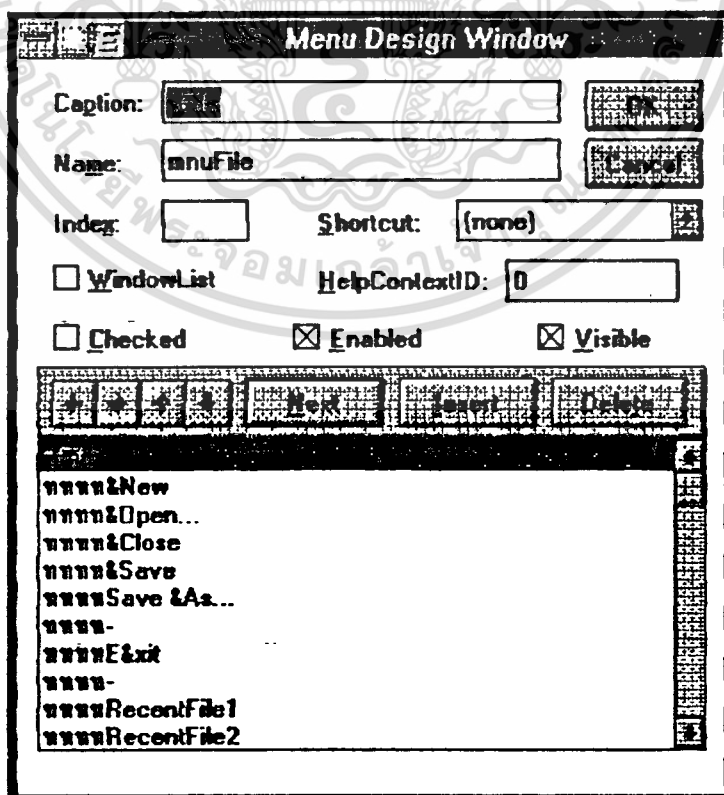
ส่วนของโปรแกรมนี้จะมีลักษณะเป็นแบบหนึ่งออบเจ็กก็จะมีลักษณะเป็น โปรแกรมย่อยของตัวเอง ส่วนของโปรแกรมจะมีลักษณะดังรูปที่ 3.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยโปรแกรมวิชวลเบสิกจะวางไว้ให้สองบรรทัดคือ เริ่มต้นของโปรแกรมย่อยในเหตุการณ์(The Even) ต่างๆของออปเจก ซึ่งจุดของเหตุการณ์นี้เองเป็นจุดเด่นของไมโครซอฟต์ วิชวลเบสิกที่เรียกได้ว่า วิชวลเบสิกเป็น อีเวนต์ ไคร์ฟเวนต์ โปรแกรมมิ่ง แดงแกวช (event driven programming language)



รูปที่ 3.5 รูปตัวอย่างรูปแบบส่วนของโปรแกรมของ Form1.frm



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้รูปที่ 3.6 รูปการออกแบบเมนูอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 การออกแบบเมนูของระบบ (Menu Design Windows)

การออกแบบระบบนั้นจะทำได้ในขณะที่อยู่ในรูปแบบเท่านั้น จะอยู่ในตัวโปรแกรมหรือที่อื่นไม่ได้การออกแบบนั้นจะต้องไปเรียกที่เมนูส่วนหลักของไมโครซอฟต์วิซวลเบสิก ชื่อเมนู ออกแบบเมนู อยู่ใน เมนูวินโดว์ การออกแบบนั้นจะมีส่วนประกอบดังรูป 3.6

ในรูปที่ 3.6 จะเห็นว่ามีส่วนประกอบหลักๆ 2 ส่วนด้วยกันคือ เมนูคอนโทรลหรือฟอรัตี้ (Menu control properties) และ เมนูคอนโทรลลิสต์บ็อก (Menu control list box)

3.5.1 เมนูคอนโทรลหรือฟอรัตี้

จะมีหน้าที่หลายอย่างที่ระบบใช้อยู่ก็จะมีกำหนดชื่อที่ผู้ใช้เห็น , กำหนดชื่อที่โปรแกรมเรียก และ สถานะต่างๆของเมนูที่เรากำหนดขึ้น

3.5.2 เมนูคอนโทรลลิสต์บ็อก

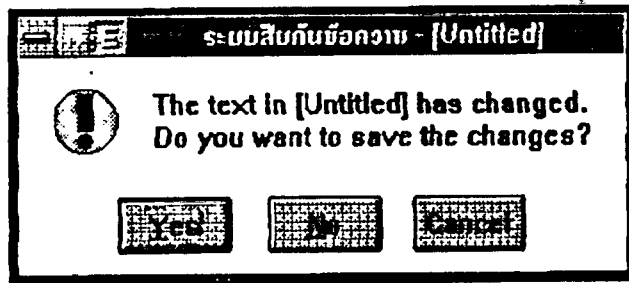
จะเป็นการควบคุมตำแหน่งของเมนูที่เรากำหนดให้ปรากฏแก่สายตาผู้ใช้ โดยจะมีคำสั่งหลักๆคือ ใส่เมนูต่อไป(Next) , แทรกเมนู(Insert) , ลบเมนู(Delete)

3.6 ไคอะล็อกบ็อก (Dialog box)

จะเป็นส่วนที่สำคัญที่สุดส่วนหนึ่งในระบบสืบค้นข้อความ และ ระบบอื่นๆมาก โดยไคอะล็อกบ็อกในไมโครซอฟต์วิซวลเบสิก จะแบ่งได้เป็น 3 ประเภทดังนี้

3.6.1 พรดิไฟเค็ดไคอะล็อกบ็อก (Predefined dialog boxes) จะมีตัวแปรพิเศษต่างๆ ที่จะปรากฏในไคอะล็อกบ็อก ไคอะล็อกประเภทนี้จะมีใช้ด้วยกัน 2 ฟังก์ชัน (Function) คือ

- ฟังก์ชัน เมสเสจบ็อก (MsgBox()) จะเป็นการแสดงรูปแบบประเภทข้อความที่จะแสดงให้ผู้ใช้ได้เห็น โดยมีลักษณะดังรูปที่ 3.7



รูปที่ 3.7 รูปตัวอย่างของเมตเตงมือ

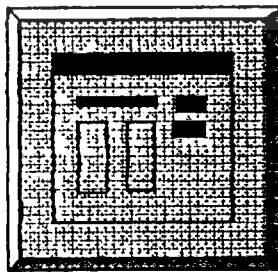
- ฟังก์ชัน อินพุทบ็อก (InputBox()) จะมีลักษณะเช่นเดียวกับฟังก์ชันเมตเตงบ็อกแต่จะมีการให้ผู้ใช้ป้อนข้อมูลลงในรูปแบบของแต่ในระบบสืบค้นข้อความจะไม่ใช้ฟังก์ชันของไอคอลล็อกนี้จะไม่กล่าวต่อไป

3.6.2 คาสต์อิมไอคอลล็อกบ็อก (Custom dialog boxes)

จะเป็นลักษณะของการ ใช้ไอคอลล็อกในรูปแบบที่ต่างไปจากข้อ 3.6.1 แต่ก็มีความหมายเหมือนกันซึ่งในระบบสืบค้นข้อความ จะไม่มีการใช้ไอคอลล็อก แบบนี้ก็จะไม่กล่าวถึง

3.6.3 คอมมอนไอคอลล็อกบ็อก (Common dialog boxes)

ในไอคอลล็อกบ็อกนี้ จะใช้มากในระบบสืบค้นข้อความเพราะคอมมอนไอคอลล็อกบ็อกนี้จะเป็นการดึงไอคอลล็อก หลักๆของบนระบบปฏิบัติการวินโดวส์ เพื่อให้ใช้งานได้สะดวกขึ้น โดยการใช้อคอมมอน ไอคอลล็อกจะต้องมีการเลือก เอาออปเจ็กของตัวมันในทูลบ็อกซึ่งจะแสดงในรูปที่ 3.8 มาไว้ในรูปแบบของระบบ



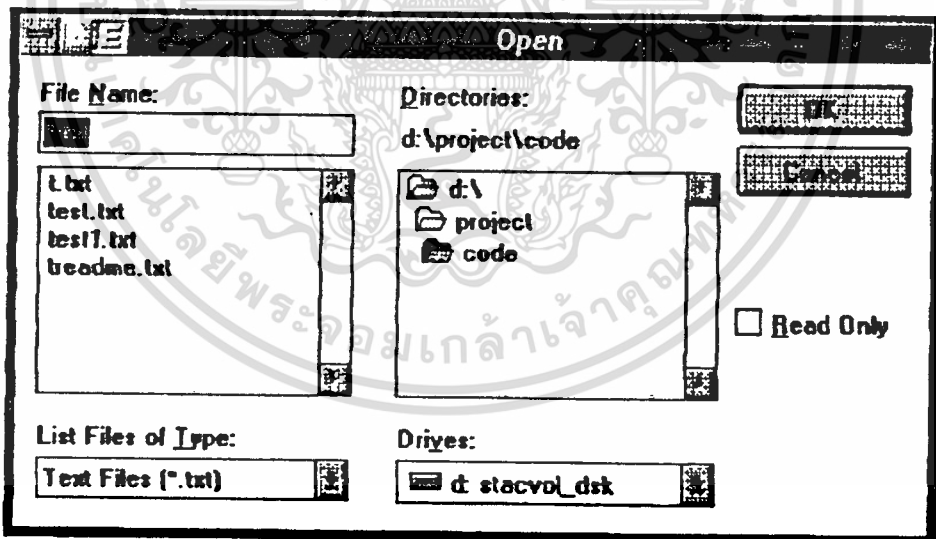
รูปที่ 3.8 รูปออปเจ็กของตัวคอมมอนไอคอลล็อก

จากนั้นก็กำหนดค่าให้ตัวไอคอลล็อกนั้นๆ ในโปรแกรมตามความหมายที่เรา
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ต้องการให้เป็น โดยความหมาย และ ค่าต่างๆนั้นจะกำหนดคั้งตารางที่ 3.1
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามเผยแพร่ลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

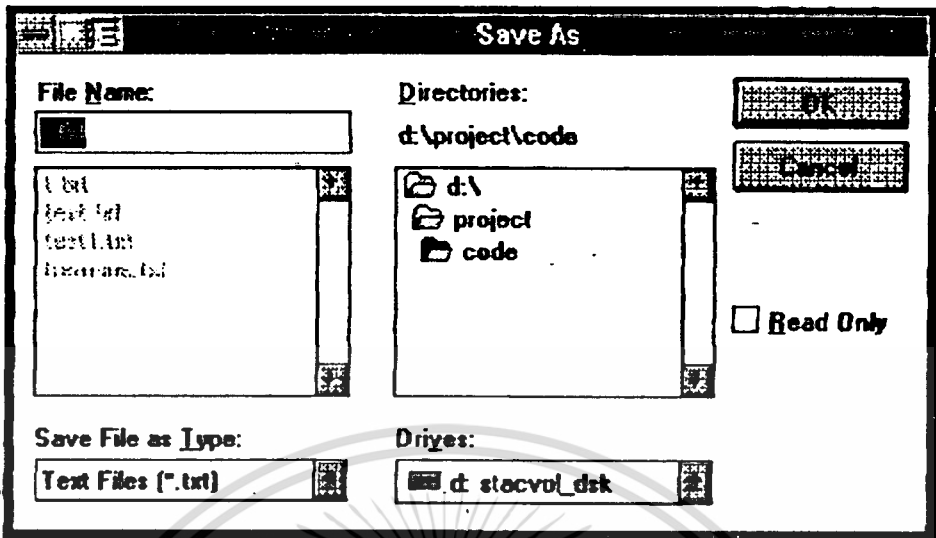
Displayed Common Dialog Box	The Value of the Active Property
Open	1
Save As	2
Color	3
Font	4
Print	5

ตารางที่ 3.1 ตารางกำหนดค่าของคอมมอนไดอะล็อก

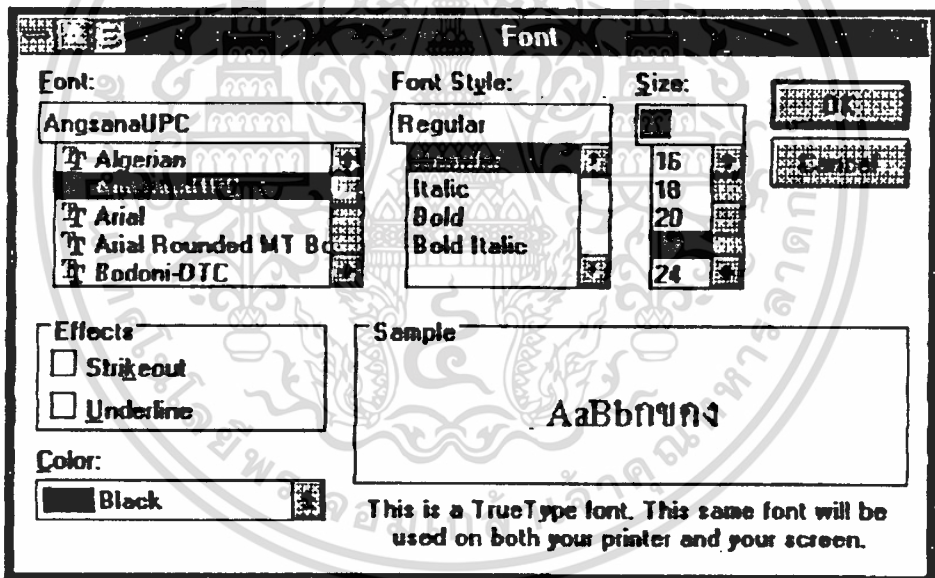
ในระบบสืบค้นข้อความที่ใช้อยู่ 3 กรณีก็คือ เปิดเพิ่มข้อมูล (Open) , บันทึกเป็น (Save As) และ รูปแบบตัวอักษร (Font) จะมีลักษณะไดอะล็อกบ็อกของการเปิดเพิ่มข้อมูลดังรูปที่ 3.9 ลักษณะไดอะล็อกบ็อกการบันทึกเป็นดังรูปที่ 3.10 และลักษณะไดอะล็อกบ็อกของรูปแบบตัวอักษร ดังรูปที่ 3.11 ตามลำดับ



รูปที่ 3.9 รูปไดอะล็อกบ็อกของการเปิดเพิ่มข้อมูล



รูปที่ 3.10 รูปไดอะล็อกบ็อกของการบันทึกเป็น

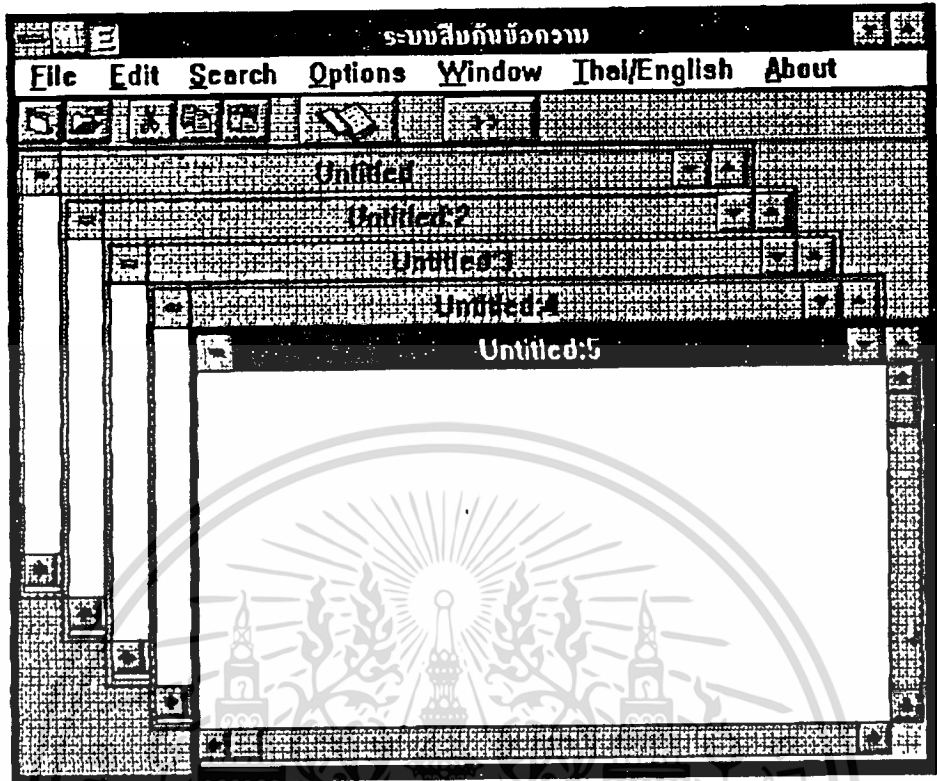


รูปที่ 3.11 รูปไดอะล็อกบ็อกของรูปแบบตัวอักษร

3.7 มัลติเพิลด็อกคิวเมนต์อินเทอร์เฟซ (Multiple Document Interface or MDI)

จะเป็นเรื่องสุดท้ายที่เราจะกล่าวถึงในการใช้ช่วงเบสิกในการสร้างระบบ มัลติเพิลด็อกคิวเมนต์อินเทอร์เฟซ นี้จะมีลักษณะเป็นรูปแบบที่มีการบรรจุรูปแบบหลายๆรูปแบบ โดยรูปแบบของตัวพ่อ (Parent form) จะเรียกว่า รูปแบบเอ็มดีไอ (MDI form) โดยเราจะต้องกำหนดว่ารูปแบบใดเป็นรูปแบบลูก (MDI child) ของตัวรูปแบบพ่อ การกำหนดนี้จะต้องทำการกำหนดใน พร็อพเพอร์ตี้ของตัวลูก ด้วยหลักการของมัลติเพิลด็อกคิวเมนต์อินเทอร์เฟซ จึงทำให้ระบบสืบค้น

เอกสารข้อความมีการเปิดเพิ่มข้อมูลได้หลายเพิ่มข้อมูล ดังรูปที่ 3.12 มอนูญาตให้เข้าไปใช้ประโยชน์ด้านการค้าไม่ว่าการณ์ใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.12 รูปแสดงการเปิดแฟ้มข้อมูล โดยใช้MDIของระบบสืบค้นข้อความ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การค้นหาคำในฐานข้อมูล

เมื่อเราสามารถวิเคราะห์รูปแบบของข้อความสืบค้นได้แล้ว เราจะทำการค้นหาโดยจะแยกตามประเภทข้อความสืบค้น จะมีการทำการค้นหาดังนี้

4.1 การค้นหาในข้อความสืบค้นปกติ

ในกรณีการค้นหาในข้อความสืบค้นปกตินี้ จะทำการตรวจสอบข้อความที่ต้องการสืบค้นว่ามีตัวปฏิบัติการหรือตัวปฏิบัติการพิเศษหรือไม่ ถ้าไม่มีตัวปฏิบัติการหรือตัวปฏิบัติการพิเศษจึงทำการค้นหาในข้อความสืบค้นปกติ

4.1.1 ทำการกำหนดให้ข้อความที่ต้องการสืบค้นเป็นข้อความต้นทาง

4.1.2 ตรวจสอบว่าต้องการทำการค้นหาแบบตรงตัวหรือไม่ ถ้าไม่ต้องการค้นหาแบบตรงตัว จะทำการแปลงข้อความที่ต้องการค้นหาให้เป็นตัวอักษรใหญ่หมด

4.1.3 เก็บตำแหน่ง Cursor ตรวจสอบว่าต้องการค้นขึ้นหรือค้นลงจากตำแหน่ง Cursor ถ้าต้องการค้นขึ้นจะทำการค้นขึ้นไปยังต้นเอกสาร ถ้าต้องการค้นลง จะค้นจะค้นจากตำแหน่ง Cursor ไปยังท้ายเอกสาร

4.1.4 ถ้าพบข้อความที่ต้องการค้นหาในเอกสาร จะกำหนดให้ตำแหน่ง Cursor ไปอยู่ที่คำที่ค้นพบ ถ้าไม่พบคำที่ต้องการค้นหา จะแสดงข้อความขึ้นมาว่าไม่พบคำที่ต้องการค้นหา

4.2 การค้นหาในข้อความสืบค้นที่มีตัวปฏิบัติการปกติ

ในการค้นหาข้อความสืบค้นที่มีตัวปฏิบัติการนี้ จะทำการสืบค้นแบบนี้เมื่อมีการตรวจสอบในคำที่ต้องการค้นหาแล้วพบตัวปฏิบัติการ “และ” หรือ “หรือ” หรือพบตัวปฏิบัติการทั้งสองคำในคำที่ต้องการหาในข้อความสืบค้น การค้นหานี้จะแบ่งได้เป็น 2 กรณี ดังนี้

4.2.1 การค้นหาในข้อความสืบค้นที่มีตัวปฏิบัติการ “และ”

ในการค้นหาในข้อความสืบค้นที่มีตัวปฏิบัติการ “และ” จะเป็นการสืบค้นแล้วได้พบตัวปฏิบัติการ “และ” ในข้อความสืบค้นดังนั้นเราจะมีขั้นตอนการค้นหาดังนี้

4.2.1.1 ในการสืบค้นที่มีตัวปฏิบัติการ “และ” จะทำการสืบค้นคำในเอกสารโดย เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเมื่อการศึกษาวงศ์วานนี้ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า กำหนดให้ ข้อความในเอกสารเป็นข้อความที่จะทำการสืบค้น ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1.2 นำคำแรกของคำในข้อความสืบค้นที่ต้องการค้นหาหาในเอกสาร เมื่อพบคำแรกในเอกสาร จะทำการคำนวณเพื่อหาว่าคำที่เจอพบอยู่ในหน้าไหนของเอกสาร ถ้าไม่พบคำแรกในเอกสารจะออกจากการค้นหา และ จะทำการแจ้งให้ผู้ใช้ทราบว่าไม่พบข้อความสืบค้นที่ผู้ใช้ต้องการค้นหา

4.2.1.3 ในกรณีที่ทำข้อ 4.2.1.2 แล้วคำแรกที่ทำการค้นหา ค้นหาพบ ก็จะนำคำที่สองมาทำการค้นหาในเอกสารในหน้าเดียวที่พบคำคำแรก จะทำการค้นหาในหน้านั้นเพียงหน้าเดียวเพราะ นิยามของตัวปฏิบัติการ “และ” คือค้นหาทุกคำที่ปรากฏในเอกสาร

4.2.1.4 นำคำต่อไปมาค้นหาในเอกสารหน้าเดียวกันกับคำแรกและคำที่สองจนกว่าจะหมดคำที่ต้องการค้นหา

4.2.1.5 ในกรณีที่การค้นหาคำใดคำหนึ่งแล้วไม่พบคำๆนั้น ในเอกสารหน้าเดียวกันกับคำแรกๆ ก็จะออกจากการค้นหาแล้วจะทำการแจ้งให้ผู้ใช้ทราบว่าไม่พบข้อความสืบค้นที่ผู้ใช้ต้องการค้นหาในเอกสารหน้าเดียวกัน

4.2.1.6 ถ้าค้นพบคำทุกคำในเอกสาร จะแสดงเอกสารหน้านั้นขึ้นมา

4.2.2 การค้นหาในข้อความสืบค้นที่มีตัวปฏิบัติการ “หรือ”

ในการค้นหาในข้อความสืบค้นที่มีตัวปฏิบัติการ “หรือ” จะเป็นการสืบค้นแล้วได้พบตัวปฏิบัติการ “หรือ” ในข้อความสืบค้นดังนั้นเราจะมีขั้นตอนการค้นหาดังนี้

4.2.2.1 ในการสืบค้นที่มีตัวปฏิบัติการ “หรือ” จะทำการสืบค้นคำในเอกสารโดยกำหนดให้ ข้อความในเอกสารเป็นข้อความที่จะทำการสืบค้น

4.2.2.2 นำคำแรกของคำในข้อความสืบค้นที่ต้องการค้นหาหาในเอกสาร เมื่อพบคำแรกในเอกสาร จะทำการแสดงเอกสารในหน้าที่ค้นพบคำๆนั้นและมีแถบสีแสดงคำที่ค้นพบ และ สิ้นสุดการค้นหา

4.2.2.3 ในกรณีที่ทำข้อ 4.2.2.2 แล้วคำแรกที่ทำการค้นหา ค้นหาไม่พบ ก็จะนำคำที่สองมาทำการค้นหาในเอกสาร ถ้าพบคำที่สองในเอกสาร จะทำการแสดงเอกสารในหน้าที่ค้นพบคำๆนั้นและมีแถบสีแสดงคำที่ค้นพบ และ สิ้นสุดการค้นหา

4.2.2.4 ทำการค้นหาข้อ 4.2.2.3 ไปเรื่อยๆจนถึงคำที่ต้องการค้นในข้อความสืบค้นตัวสุดท้าย แล้วถ้าไม่พบคำในข้อความสืบค้นเลยสักคำเดียว จะมีการแจ้งให้ผู้ใช้ทราบว่าไม่พบคำใดๆในข้อความสืบค้น

4.8 การค้นหาในข้อความสืบค้นที่มีตัวปฏิบัติการพิเศษ

ในการค้นหาโดยใช้บางส่วนของคำเพื่อค้นหาทั้งคำมี 2 แบบคือ ส่วนหน้าของคำ หรือ ส่วนหลังของคำ จะเป็นการค้นหาว่าในฐานข้อมูลมีคำใดที่มีส่วนหน้าหรือส่วนหลังของคำตามที่ ต้องการ ดังจะมีการค้นหาดังนี้

4.8.1 การค้นหาในข้อความที่มีตัวปฏิบัติการพิเศษ *

การค้นหาในข้อความที่มีตัวปฏิบัติการพิเศษ * สามารถใช้ค้นหาคำได้ 2 กรณี

4.3.1.1 กรณีตัวปฏิบัติการ * อยู่ด้านหลัง

เพื่อช่วยในการอธิบายขั้นตอนการสืบค้นจะยกตัวอย่างเช่น คำว่า com* จะทำการสืบค้นว่ามีเอกสารว่ามีคำว่า com ในเอกสาร หรือไม่ และอยู่ที่ใดบ้าง เมื่อพบคำว่า com จะทำการตรวจสอบตัวอักษรที่อยู่ต่อจากคำว่า com ทีละตัว จนกว่าจะพบเครื่องหมายที่แสดงว่าหมดคำ เครื่องหมายที่ว่าคือ เครื่องหมายเว้นวรรค หรือ เครื่องหมายจบบรรทัด จะทำการนำเอาคำนั้นมาเก็บไว้ใน แอร์เรย์ (array) และหาคำต่อไปในลักษณะเดียวกันเมื่อพบแล้วจะนำมาเปรียบเทียบกับคำใน แอร์เรย์ ถ้าเป็นคำๆเดียวกันก็จะไม่นำคำนั้นมาเก็บในแอร์เรย์ เมื่อค้นหาจนครบแล้วจึงนำคำต่างๆที่เก็บใน แอร์เรย์ มาแสดงให้ผู้ใช้ได้เห็น ถ้าผู้ใช้ต้องการจะค้นหาคำๆไหน ก็นำ เมาส์คลิกที่คำๆนั้นเมื่อผู้ใช้คลิกที่คำๆไหน จะนำคำๆนั้นไปทำการค้นหาในเอกสาร แล้วแสดงตำแหน่งของคำๆนั้นให้ผู้ใช้ทราบ

4.3.1.2 กรณีตัวปฏิบัติการ * อยู่ด้านหน้า

เพื่อช่วยในการอธิบายขั้นตอนการสืบค้นจะยกตัวอย่างเช่น คำว่า *tion จะทำการสืบค้นว่ามีเอกสารว่ามีคำว่า tion ในเอกสาร หรือไม่ และอยู่ที่ใดบ้าง เมื่อพบคำว่า tion จะทำการตรวจสอบตัวอักษรที่อยู่ต่อจากคำว่า tion ว่าพบเครื่องหมายที่แสดงว่าหมดคำ เครื่องหมายที่ว่าคือ เครื่องหมายเว้นวรรค หรือ เครื่องหมายจบบรรทัด หรือไม่ ถ้าไม่ใช่จะค้นหาคำต่อไปถ้าใช่ ก็จะทำการตรวจสอบตัวอักษรที่อยู่ข้างหน้าคำว่า tion ไปทีละตัวอักษร จนกว่าจะพบเครื่องหมายที่แสดงว่าหมดคำ เครื่องหมายที่ว่าคือ เครื่องหมายเว้นวรรค หรือ เครื่องหมายจบบรรทัด จะทำการนำเอาคำนั้นมาเก็บไว้ใน แอร์เรย์ ละหาคำต่อไปในลักษณะเดียวกันเมื่อพบแล้วจะนำมาเปรียบเทียบกับคำในแอร์เรย์ ถ้าเป็นคำๆเดียวกันก็จะไม่นำคำ

นั้นมาเก็บในแอร์เรย์ เมื่อค้นหาจนครบแล้วจึงนำคำต่างๆที่เก็บใน แอร์เรย์ มาแสดงให้ผู้ใช้ได้เห็น ถ้าผู้ใช้ต้องการจะค้นหาคำๆไหน ก็นำ เม้าส์คลิกที่คำๆนั้น เมื่อผู้ใช้คลิกที่คำๆไหน จะนำคำๆนั้นไป ทำการค้นหาในเอกสาร์ แล้วแสดงตำแหน่งของคำๆนั้นให้ผู้ใช้ทราบ

4.3.2 การค้นหาในข้อความที่มีตัวปฏิบัติการพิเศษ ?

การค้นหาในข้อความที่มีตัวปฏิบัติการพิเศษ ? สามารถใช้ค้นหาได้ 2 กรณี

4.3.2.1 กรณีตัวปฏิบัติการ ? อยู่ด้านหลัง

เพื่อช่วยในการอธิบายขั้นตอนการสืบค้นจะยกตัวอย่างเช่นคำว่า com???? จะทำการสืบค้นว่ามีเอกสารว่ามีคำว่า com ในเอกสาร หรือไม่ และอยู่ที่ใดบ้าง เมื่อพบคำว่า com จะทำการตรวจสอบตัวอักษรที่อยู่ข้างหน้าคำว่า com ว่าเป็นเครื่องหมายแสดงว่าหมดคำหรือไม่ ถ้าไม่เป็นจะค้นหาต่อไปถ้าเป็นจะตรวจสอบตัวอักษรที่ตำแหน่งเท่ากับตำแหน่งเริ่มต้นของคำบวกกับจำนวนอักษรที่มีใน คำที่ต้องการค้นหาว่ามีเครื่องหมายแสดงว่าหมดคำหรือไม่ ถ้าไม่เป็นก็จะไปหาคำที่ตำแหน่งอื่นต่อไป ถ้าเป็นเครื่องหมายแสดงว่าหมดคำ หรือไม่ถ้าไม่เป็นจะไปหาคำที่ตำแหน่งอื่นต่อไป ถ้าเป็นเครื่องหมายแสดงว่าหมดคำ จะทำการนำคำนั้น มาเก็บไว้ใน แอร์เรย์ และหาคำต่อไปในลักษณะเดียวกันเมื่อพบแล้วจะนำมา เปรียบเทียบกับคำใน แอร์เรย์ ถ้าเป็นคำๆเดียวกันก็จะไม่นำคำนั้นมาเก็บใน แอร์เรย์ เมื่อค้นหาจนครบแล้วจึงนำคำต่างๆที่เก็บใน แอร์เรย์ มาแสดงให้ผู้ใช้ได้ เห็น ถ้าผู้ใช้ต้องการจะ ค้นหาคำๆไหน ก็นำ เม้าส์คลิกที่คำๆนั้นเมื่อผู้ใช้คลิกที่ คำๆไหน จะนำคำๆนั้นไปทำการค้นหาในเอกสาร์ แล้วแสดงตำแหน่งของคำๆนั้น ให้ผู้ใช้ทราบ

4.3.2.2 กรณีตัวปฏิบัติการ ? อยู่ด้านหน้า

เพื่อช่วยในการอธิบายขั้นตอนการสืบค้นจะยกตัวอย่างเช่นคำว่า ???ter จะทำการสืบค้นว่ามีเอกสารว่ามีคำว่า ter ในเอกสาร หรือไม่ และอยู่ที่ใดบ้าง เมื่อพบคำว่า ter จะทำการตรวจสอบตัวอักษรที่อยู่ต่อจากคำว่า ter ว่าเป็นเครื่องหมายที่แสดงว่าหมดคำ เครื่องหมายที่ว่าเป็น เครื่องหมายเว้นวรรค หรือ เครื่องหมายจบบรรทัด หรือไม่ ถ้าไม่ใช่จะค้นหาต่อไปถ้าใช่ ก็ทำการตรวจสอบที่ตำแหน่งของคำที่ค้นพบลบด้วย จำนวนอักษร ของตัวปฏิบัติการ ? ว่ามีกี่ตัว ถ้า

ที่ตำแหน่งนั้นไม่เป็นเครื่องหมาย แสดงว่าหมคค่า จะทำการค้นหาคำว่า ter ต่อ
 ไปด้วยเป็นเครื่องหมายแสดงว่าหมคค่า จะทำการนำเอาค่านั้นมาเก็บไว้ใน แอร์เรย์
 ละหาค่าต่อไปในลักษณะเดียว เมื่อพบแล้วจะนำมาเปรียบเทียบกับค่า ในแอร์เรย์
 ถ้าเป็นค่าๆเดียวกันก็จะไม่นำค่า นั้นมาเก็บในแอร์เรย์ เมื่อค้นหาจนครบแล้วจึงนำ
 ค่าต่างๆที่เก็บใน แอร์เรย์ มาแสดงให้ผู้ใช้ได้เห็น ถ้าผู้ใช้ต้องการจะค้นหาค่าๆ
 ไหน ก็มา เมาส์คลิกที่ค่าๆนั้นเมื่อผู้ใช้คลิกที่ค่าๆไหน จะนำค่าๆนั้นไป ทำการค้น
 หาในเอกสาร แล้วแสดง ตำแหน่งของค่าๆนั้นให้ผู้ใช้ทราบ

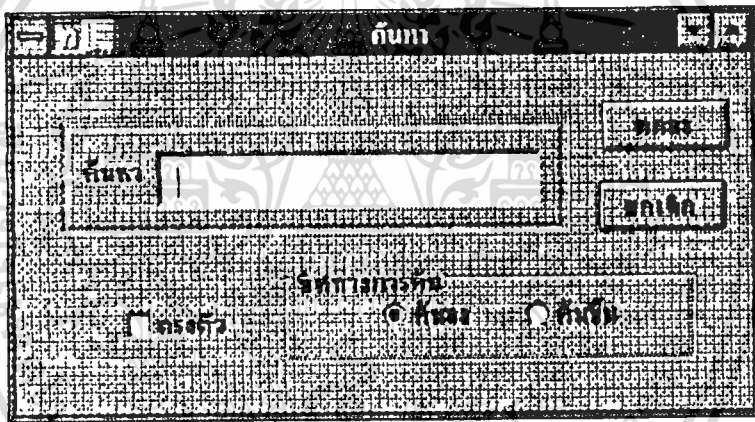


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

การทดลองและการแสดงผล

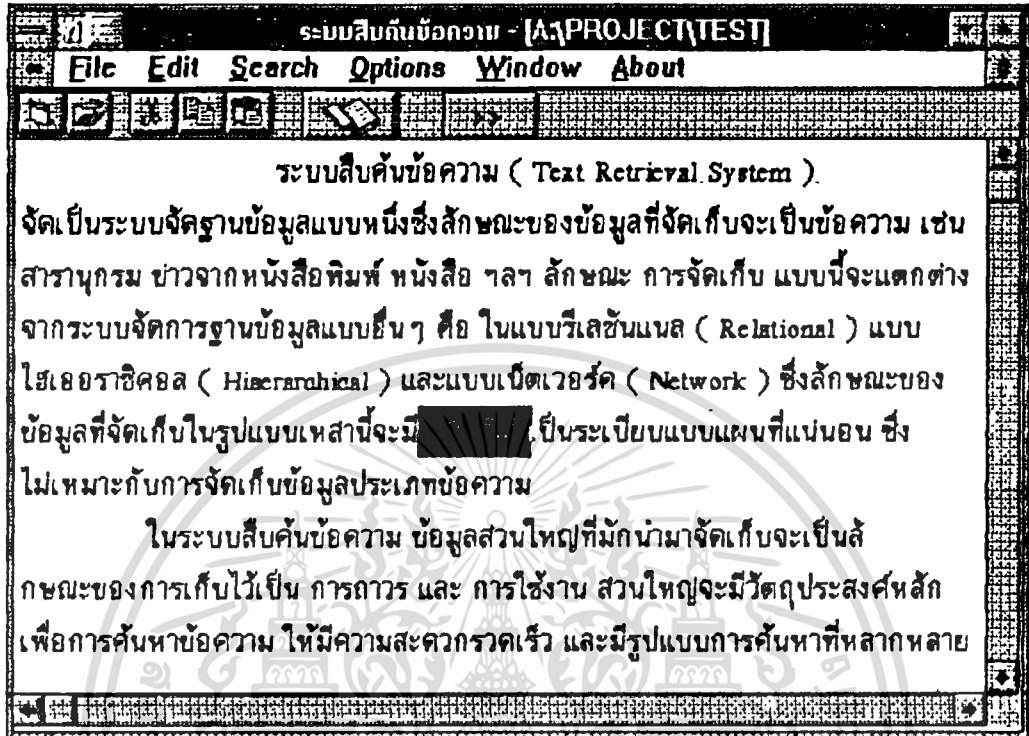
การทดลองสืบค้นข้อความจะแบ่งตามการแสดงผลจะมีอยู่ด้วยกันสองแบบ คือการแสดงผลแบบแสดงผลจากเอกสารจริง ซึ่งจะเกิดจากข้อความสืบค้นที่มีตัวปฏิบัติการปกติ และ ข้อความสืบค้นที่มีตัวปฏิบัติการปกติ ส่วนการแสดงผลอีกแบบคือการแสดงผลแบบเป็นรายการของคำที่พบซึ่งจะพบในการแสดงผลของข้อความสืบค้นแบบที่มีตัวปฏิบัติการพิเศษ การรับข้อความสืบค้นนั้นจะรับข้อมูลใน เท็กซ์บ็อกซ์(Text Box)ในฟอร์มการค้นหาดังรูปที่ 5.1



รูปที่ 5.1 รูปแบบการรับข้อความสืบค้น

5.1 การแสดงผลข้อมูลในเอกสารในการสืบค้นแบบปกติ และ แบบมีตัวปฏิบัติการปกติ

ในการแสดงผลข้อมูลหลังจากข้อความสืบค้นไม่ว่าในแบบปกติ หรือ แบบมีตัวปฏิบัติการปกติ ได้ทำสำเร็จเอกสารได้ทำสำเร็จ จะเป็นการแสดงผลของเอกสารที่มีคำปรากฏในข้อความสืบค้นปรากฏในเอกสารนั้นๆ ตัวอย่างเช่น เราต้องการสืบค้นคำว่า “โครงสร้าง” ก็เหมือนข้อความดังกล่าวถ้าค้นพบข้อมูลก็จะแสดงผล โดยจะมีลักษณะดังรูปที่ 5.2

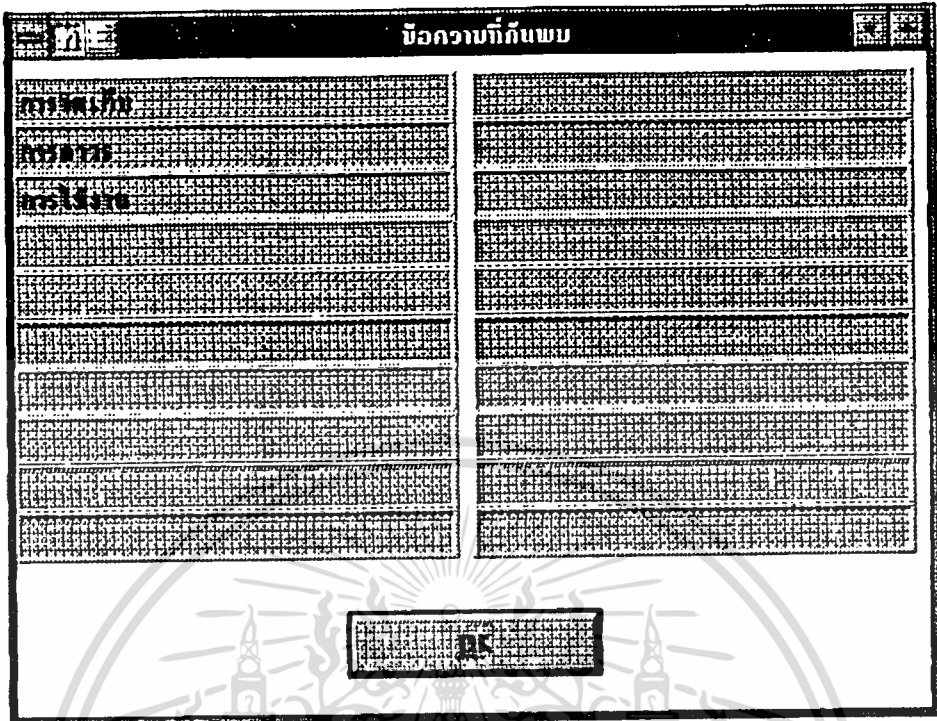


รูปที่ 5.2 รูปการแสดงผลในการสืบค้นปกติหรือมีตัวปฏิบัติการปกติ

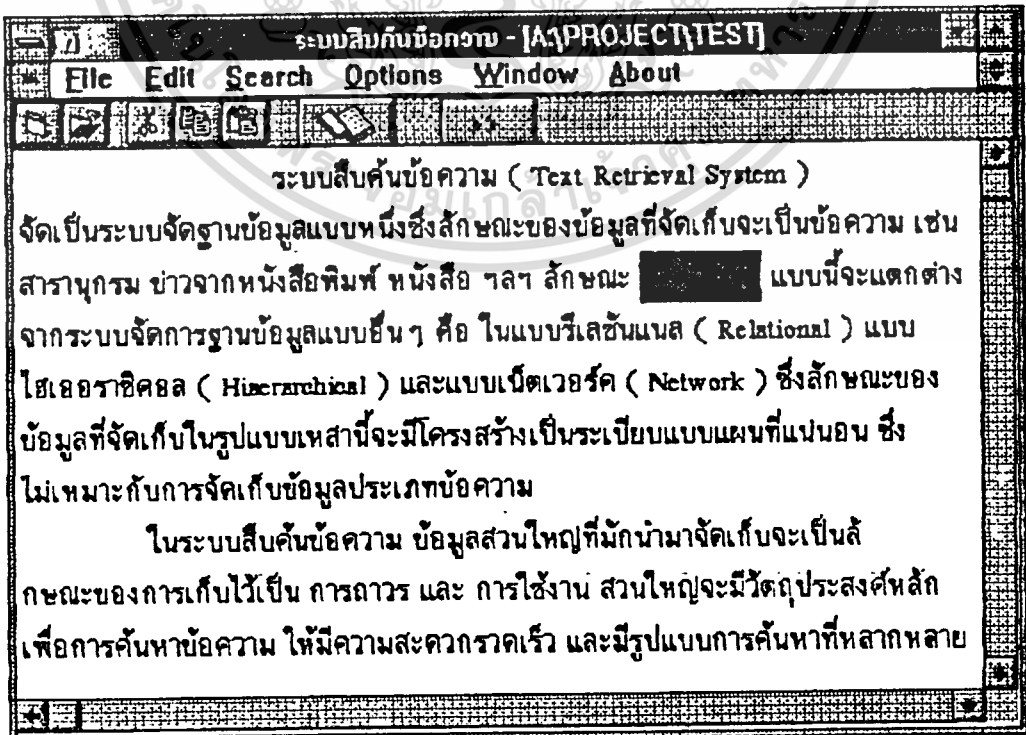
5.2 การแสดงผลของข้อความสืบค้นที่มีตัวปฏิบัติการพิเศษ

การแสดงผลวิธีนี้จะนำเอาค่าที่ตรงกับความต้องการของผู้ใช้ ที่ป้อนผ่านข้อความสืบค้น โดยจะแสดงเรียงตามตำแหน่งของข้อมูลที่ต้องการค้นหาโดยจะแสดงผลครั้งละ 20 ตัว ถ้ามีข้อความค้นหาเกิน 20 ตัวเราก็สามารถดูตัวต่อไปได้มีตัวอย่างอยู่ในคู่มือการใช้งาน ดังตัวอย่างเช่น ถ้าเราต้องการสืบค้นข้อความโดยใช้ตัวปฏิบัติการพิเศษอย่างคำว่า "การ*" โดยการแสดงผลจะมีลักษณะดังรูปที่ 5.3

แล้วถ้าเราใช้เมาส์ (mouse) คลิก (click) ในป้าย (Label) ที่แสดงข้อความที่ตรงกับข้อความสืบค้นที่เราป้อนไป เช่นคลิกคำว่า การจัดการ ดังในรูป 5.3 เคอร์เซอร์ (cursor) ก็จะวิ่งไปยังข้อความนั้นๆในฐานข้อมูลดังรูป 5.4



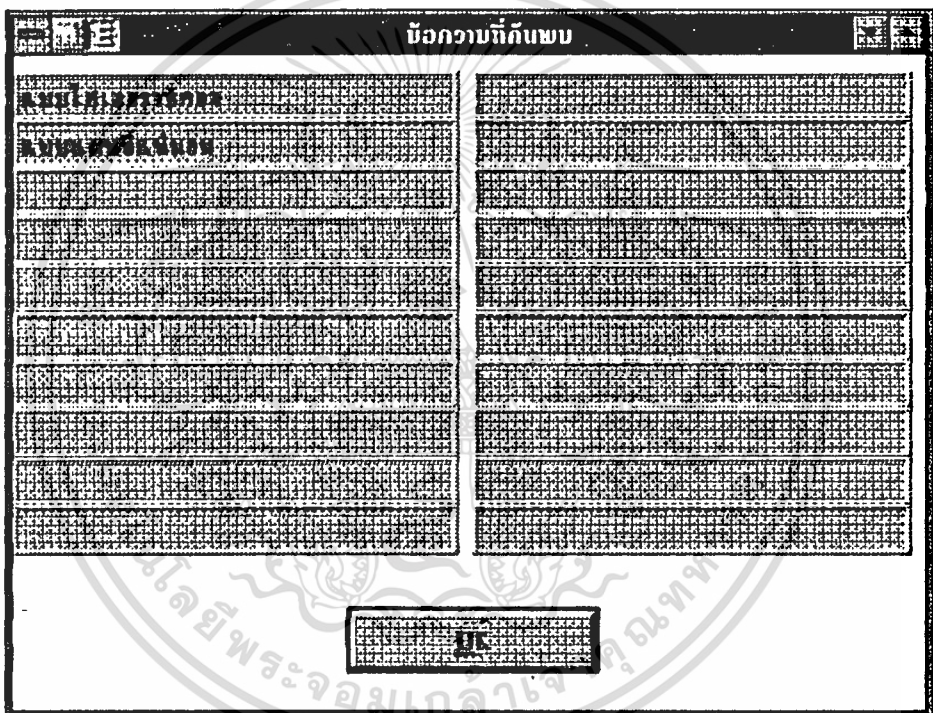
รูปที่ 5.3 รูปการแสดงผลการสืบค้นข้อมูลประเภทใช้ตัวปฏิบัติการพิเศษ



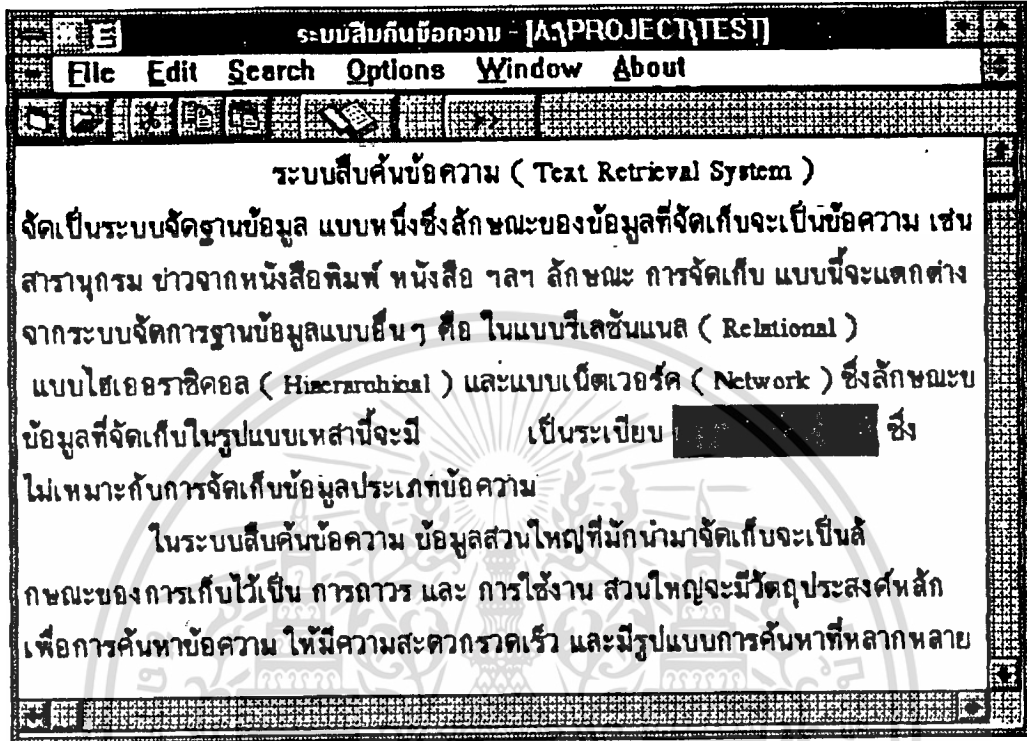
เอกสารนี้เป็นเอกสารที่รูปที่ 5.4 รูปการแสดงผลที่ต่อจากการเลือกข้อความสืบค้นต่อจากรูป 5.3 โยชนด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือตัวอย่างเช่น ถ้าเราต้องการสืบค้นข้อความโดยใช้ตัว ปฏิบัติการพิเศษอย่างคำว่า
 “แบบ?????????” โดยการแสดงผลจะมี ลักษณะดังรูปที่ 5.5

แล้วถ้าเราใช้เมาส์คลิกในป้ายที่แสดงข้อความที่ตรงกับข้อความสืบค้นที่เราป้อนไป เช่น
 คลิกคำว่า การจัดการ ดังในรูป 5.5 เคอร์เซอร์ (cursor) ก็ะวิ่ง ไปยังข้อความนั้นๆ ในฐานข้อมูลดัง
 รูป 5.6



รูปที่ 5.5 รูปการแสดงผลการสืบค้นข้อมูลประเภทใช้ตัวปฏิบัติการพิเศษ ?



รูปที่ 5.6 รูปการแสดงผลที่ต่อจากการเลือกข้อความสืบค้นต่อจากรูป 5.5

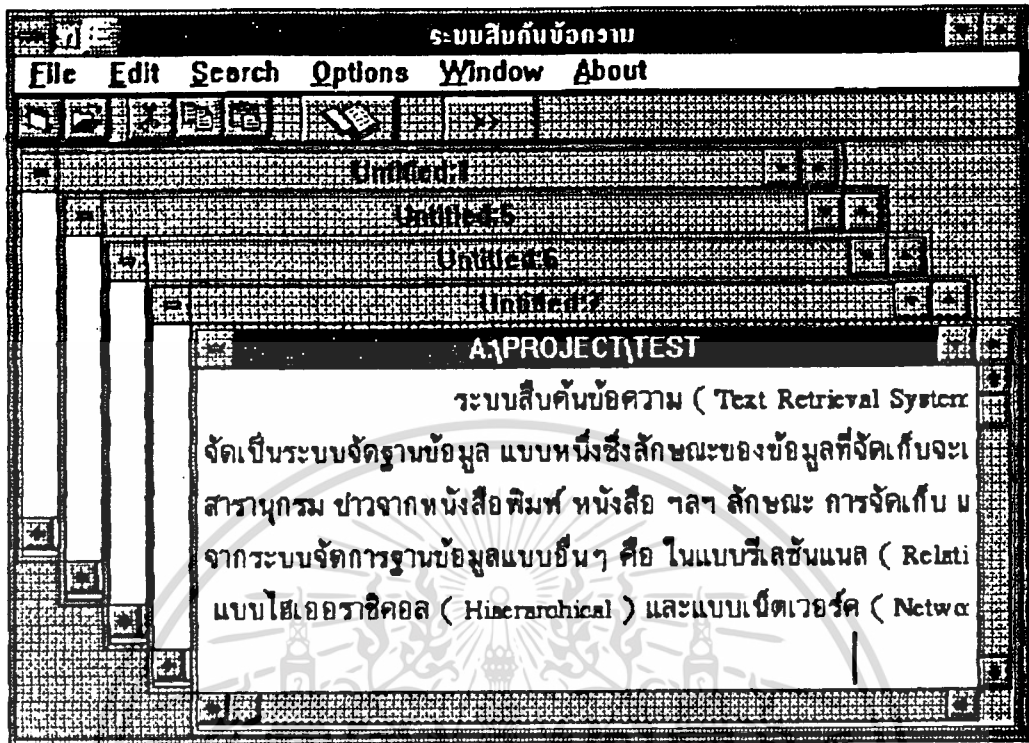
5.8 การแสดงผลของรูปแบบของระบบ

ระบบสืบค้นข้อความนี้จำเป็นจะมีลักษณะเป็นเท็กซ์อีดิเตอร์ (Text editor) เพื่อเป็นตัวจัดเก็บฐานข้อมูลก็ถือเป็นตัวเปิดแก้ไขข้อมูลประเภทตัวอักษร (Text File) เพื่อใช้ในการสืบค้นข้อความ โดยระบบสืบค้นข้อความนี้จะทำงานบนระบบปฏิบัติการวินโดวส์จึงเลือกโปรแกรมภาษาที่ใช้ในการเขียนระบบนี้คือ วิววลเบสิก 3.0 (Visual Basic for Window 3.0) เนื่องจากโปรแกรมภาษานี้จะช่วยประหยัดเวลา ในจัดการเกี่ยวกับหน้าจอ , แก้ไขข้อมูล , รูปแบบของตัวอักษร เป็นอย่างดีโดย

เท็กซ์อีดิเตอร์ที่สร้างขึ้นจะมีลักษณะเช่นเดียวกันกับอีดิเตอร์ทั่วไป คือจะมีความสามารถหลักๆดังนี้

- มีการสร้าง , เปิด , ปิด , แก้ไข และจัดเก็บแก้ไขข้อมูลได้
- มีการจัดการเกี่ยวกับ รูปแบบตัวอักษรได้
- มีการจัดเรียงหน้ากระดาษของแก้ไขข้อมูลต่างๆได้

ลักษณะโดยรวมของระบบสืบค้นข้อความนี้จะแสดงในรูปที่ 5.7



รูปที่ 5.7 รูประบบสืบค้นข้อความ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 8

บทวิจารณ์และสรุป

สำหรับระบบสืบค้นข้อความที่ได้ทดลองพัฒนา และ สร้างขึ้นมาใหม่ จะมีความสามารถต่างๆดังต่อไปนี้ ในด้านการค้นหาข้อมูล เราสามารถที่จะแบ่งประเภทการค้นหาได้เป็น 3 ประเภทคือ

- ประเภทแรก จะเป็นการค้นหาโดยใช้คำเพียงคำเดียว
- ประเภทที่ 2 จะเป็นการค้นหาโดยใช้หลายๆคำพร้อมกันโดยใช้ตัวปฏิบัติการต่างๆ ในการเชื่อมระหว่างคำแต่ละคำ ตัวปฏิบัติการเหล่านี้ได้แก่ ตัวปฏิบัติการ “หรือ” ซึ่งจะทำให้ขอบเขตของการค้นหากว้างขึ้น ตัวปฏิบัติการ “และ” ซึ่งจะทำให้ขอบเขตการค้นหาแคบลง
- ประเภทที่ 3 จะเป็นการค้นหาโดยใช้เพียงส่วนหน้าหรือส่วนหลังของคำ ซึ่งจะมีประโยชน์ในกรณีที่ใช้ไม่รู้ว่คำที่ต้องการที่แท้จริงเป็นคำอะไร
- ประเภทที่ 4 จะเป็นการค้นหาโดยตรวจสอบคำใกล้เคียงกับคำที่ต้องการหา

หนังสืออ้างอิง

1. Tremblay, J.P., and P.G. Sorenren, "The Theory and Practice of Compiler Writing", McGraw-Hill, 1989.
2. D.F. Scott, "Extending Visual Basic for Windows", Sam-Publishing, 1992
3. D.F. Scott, "Visual Basic for Windows Developer's Guide", Sam-Publishing, 1993
4. Nathan Gurewich and Ori Gurewich, "Teach Yourself Visual Basic in 21 Days", Sam-Publishing, 1993

