

บทที่ 4

การทดสอบระบบ

เมื่อพัฒนาโปรแกรมตามที่ได้ออกแบบเป็นที่เรียบร้อยแล้ว ขั้นตอนต่อไป คือ การทดสอบการประมวลผลของโปรแกรมทั้งในด้านความถูกต้องตามที่ออกแบบ และในด้านประสิทธิภาพในการใช้งาน ขั้นตอนในการทดสอบแบ่งออกเป็น 4 ขั้นตอน คือ สร้างแฟ้มข้อมูลนำเข้าเพื่อทดสอบ ทดสอบการทำงานของโปรแกรมตัวแปล ทดสอบการทำงานของโปรแกรมเครื่องมือแสดงผลข้อมูลไฮเปอร์ดอกคิวเมนต์ และทดสอบการทำงานของโปรแกรม OLE

การประเมินผลเวลาที่ใช้ในการทำงาน เป็นเวลาที่ทำการทดสอบทำบนเครื่องคอมพิวเตอร์แบบเพนเทียม 166 เมกกะเฮิร์ตซ์ หน่วยความจำหลัก 16 เมกกะไบต์

4.1 สร้างแฟ้มข้อมูลนำเข้า

ข้อมูลที่ใช้ในการทดสอบเป็นข้อมูลในรูปแบบ RTF เวอร์ชัน 1.x โดยสร้างจากโปรแกรมสำเร็จรูปไมโครซอฟต์เวิร์ด เวอร์ชัน 7 สำหรับวินโดวส์ 95 แฟ้มข้อมูลมีชื่อว่า infile.rtf กำหนดให้แฟ้มข้อมูลมีเนื้อหาครอบคลุม ดังต่อไปนี้

1. ใช้ตัวอักษรรูปแบบ และสีต่าง ๆ กัน
2. ให้มีรูปภาพบิตแมพ ปรากฏในแฟ้มข้อมูล แบบใช้เชื่อมโยงไปยังแฟ้มข้อมูลบิตแมพ
3. ให้มีแฟ้มข้อมูลเสียงแบบ .wav (ขีดเส้นกลางข้อความ) โดยใช้ข้อความซ่อน ระบุสารบบและชื่อแฟ้มข้อมูลเสียง
4. ให้มีข้อความส่วนที่เป็นการกระโดด (ขีดเส้นใต้คู่) โดยใช้ข้อความซ่อน ระบุชื่อหัวข้อที่จะกระโดดไป
5. ให้มีข้อความส่วนที่เป็นการเรียกให้แสดงหน้าต่างแบบผุดขึ้น (ขีดเส้นใต้เดี่ยว) โดยใช้ข้อความซ่อน ระบุชื่อหัวข้อที่จะแสดงบนหน้าต่าง
6. ให้มีหัวข้อ ชื่อเรื่อง และคำหลัก

แฟ้มข้อมูลนำเข้าที่สร้างขึ้นประกอบด้วย 8 หัวข้อ 6 ชื่อเรื่อง และ 28 คำหลัก โดยคำหลัก "doitung" จะเป็นคำหลักของ 2 หัวข้อ คือ หัวข้อ wat และ หัวข้อ place แต่ละหัวข้อจะประกอบด้วยตัวอักษรที่มีรูปแบบ สี และขนาดที่แตกต่างกันออกไป แฟ้มข้อมูลนำเข้าเพื่อทดสอบระบบแสดงไว้ในภาคผนวก ข โดยจะแสดงทั้งรูปแบบที่ปรากฏบนเอกสารไมโครซอฟต์เวิร์ด และรูปแบบ RTF

หัวข้อที่	ชื่อหัวข้อ	ชื่อเรื่อง	คำหลัก	รายละเอียดอื่น ๆ
1	content	-	<ul style="list-style-type: none"> ● content ● Chiangrai tourism 	<ul style="list-style-type: none"> ● รูปภาพปิดแมพ ● เพิ่มข้อมูลเสียง ● ข้อความที่เรียกให้แสดงหัวข้อ amphur หน้าต่างแบบผุดขึ้น ● ข้อความที่เรียกให้แสดงหัวข้อ subamphur หน้าต่างแบบผุดขึ้น ● ข้อความที่เรียกให้กระโดดไปยังหัวข้อ wat ● ข้อความที่เรียกให้กระโดดไปยังหัวข้อ place ● ข้อความที่เรียกให้กระโดดไปยังหัวข้อ hilltribe ● ข้อความที่เรียกให้กระโดดไปยังหัวข้อ shopping ● ข้อความที่เรียกให้กระโดดไปยังหัวข้อ hotel
2	amphur	amphur in Chiangrai	<ul style="list-style-type: none"> ● amphur 	
3	subamphur	subamphur in Chiangrai	<ul style="list-style-type: none"> ● subamphur 	
4	wat	temple and ancients	<ul style="list-style-type: none"> ● temple ● ancient ● doi tung ● prakeaw ● wat ● prasingh ● museum 	

ตารางที่ 4 -1 สรุปเนื้อหาเพิ่มข้อมูลนำเข้า

หัวข้อที่	ชื่อหัวข้อ	ชื่อเรื่อง	คำหลัก	รายละเอียดอื่น ๆ
5	place	natural tourist place	<ul style="list-style-type: none"> ● natural place ● nature ● place ● kok river ● doi maeslong ● doi tung 	
6	hilltribe	hilltribe village	<ul style="list-style-type: none"> ● hilltribe ● hilltribe village ● e-go ● a-ka ● mong ● muser 	
7	shopping	shopping place	<ul style="list-style-type: none"> ● shopping ● handmade ● handmade center 	
8	hotel	-	<ul style="list-style-type: none"> ● hotel ● resort ● camp 	

ตารางที่ 4-1 สรุปเนื้อหาเพิ่มข้อมูลนำเข้า (ต่อ)

4.2 ทดสอบการทำงานของโปรแกรมตัวแปล

เมื่อทดสอบวิ่งโปรแกรมตัวแปลโดยระบุชื่อเพิ่มข้อมูลนำเข้าเป็น infile.rtf แล้ว จะได้ผลลัพธ์เป็นเพิ่มข้อมูลผลลัพธ์ชื่อ infile.hdv ซึ่งโปรแกรมตัวแปลใช้ชื่อเดิมของเพิ่มข้อมูลนำเข้า แล้วให้นามสกุลใหม่เป็น .hdv เนื้อหาของเพิ่มข้อมูลผลลัพธ์ที่ได้จากการทดสอบจะแสดงไว้ในภาคผนวก ค

การอ้างอิงตำแหน่งในแฟ้มข้อมูลผลลัพธ์ เป็นการอ้างอิงตำแหน่งไบต์แบบออฟเซต คือ เริ่มต้นไบต์แรก เป็นตำแหน่ง 0 (ศูนย์) และนับเรื่อยไปในไบต์ถัด ๆ ไป

แฟ้มข้อมูลผลลัพธ์ที่เป็นผลจากการทดสอบ มีเนื้อหาประกอบด้วย 5 ส่วน อธิบายโดยสังเขป ได้ดังนี้

1. ส่วนหัว

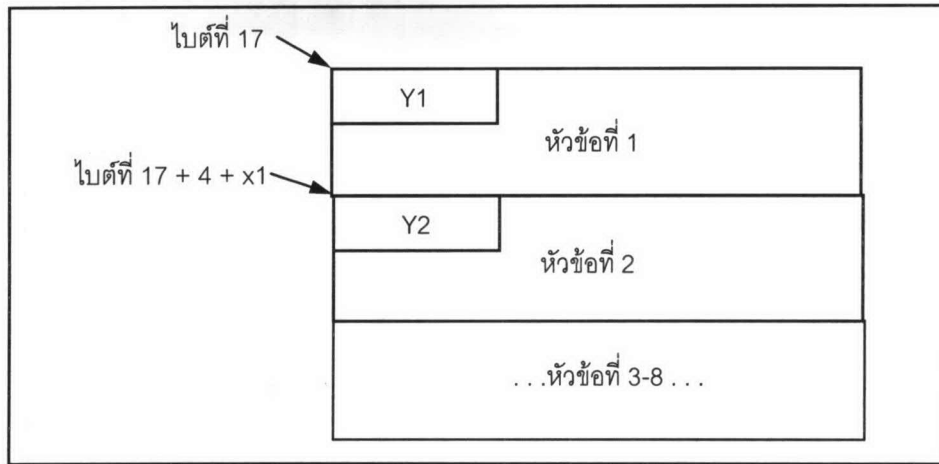
- ไบต์ที่ 1-4 เก็บตัวเลขแบบไบนารี มีค่าเป็น 16 เสมอ หมายถึง ข้อมูลหัวข้อแรกเริ่มเก็บที่ตำแหน่งออฟเซตที่ 16 เสมอ
- ไบต์ที่ 5-8 เก็บตัวเลขแบบไบนารี เก็บค่าที่บอกตำแหน่งออฟเซตของรายละเอียดหัวข้อ ในรูปที่ 4-1 แทนค่านี้ด้วย x1
- ไบต์ที่ 9-12 เก็บตัวเลขแบบไบนารี เก็บค่าที่บอกตำแหน่งออฟเซตของรายละเอียดชื่อเรื่อง ในรูปที่ 4-1 แทนค่านี้ด้วย x2
- ไบต์ที่ 13-16 เก็บตัวเลขแบบไบนารี เก็บค่าที่บอกตำแหน่งออฟเซตของรายละเอียดคำหลัก ในรูปที่ 4-2 แทนค่านี้ด้วย x3

ไบต์ที่	0	4	8	12
	16	X1	X2	X3

รูปที่ 4-1 รูปจำลองเนื้อหาส่วนหัวของแฟ้มข้อมูลผลลัพธ์

2. ส่วนหัวข้อ

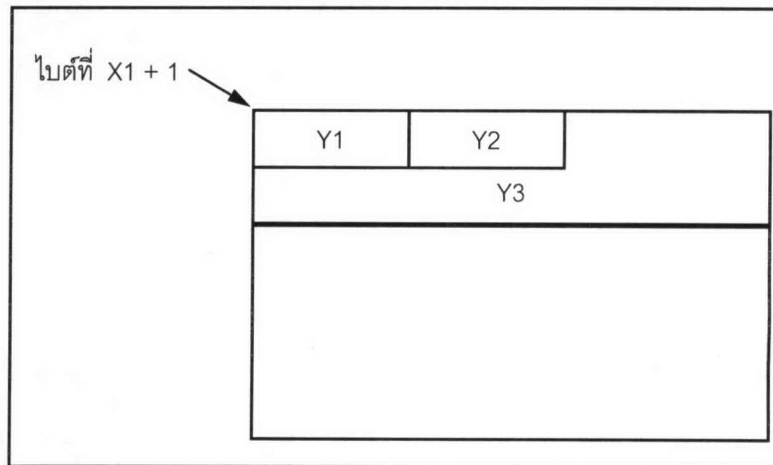
- ไบต์ที่ 17-20 เก็บตัวเลขแบบไบนารี บอกจำนวนความยาวของข้อมูลหัวข้อที่ 1 หรือ content ในรูปที่ 4-2 แทนค่าความยาวของข้อมูลหัวข้อที่ 1 ด้วย Y1
- ไบต์ที่ 21 นับถัดไปจนครบจำนวนความยาวข้อมูล เก็บข้อมูลหัวข้อที่ 1 หรือ content ในรูปที่ 4-2 คือส่วนที่แทนด้วย หัวข้อที่ 1
- ไบต์ถัดไป 4 ไบต์ เก็บความยาวของข้อมูลหัวข้อ และไบต์ถัดไปจำนวนเท่าความยาวข้อมูล เก็บข้อมูลของหัวข้อที่ 2 จนถึง หัวข้อที่ 8 ตามลำดับ



รูปที่ 4-2 รูปจำลองเนื้อหาส่วนหัวข้อของแฟ้มข้อมูลผลลัพธ์

3. ส่วนรายละเอียดหัวข้อ

- ไบต์ถัดไป 4 ไบต์ เก็บตัวเลขแบบไบนารี มีค่าเป็น 8 หมายถึง จำนวนหัวข้อทั้งหมดในระบบ ในรูปที่ 4-3 จำนวนหัวข้อทั้งหมดแทนด้วย Y1 โดยเริ่มต้นที่ไบต์ที่ X1 ซึ่งอ้างไว้ในรูปที่ 4-1
- ไบต์ถัดไป 4 ไบต์ เก็บหมายเลขประจำหัวข้อ 64 ไบต์ถัดไปเก็บชื่อหัวข้อ และ 4 ไบต์ถัดไปเก็บตำแหน่งเริ่มต้นของข้อมูลหัวข้อ ตั้งแต่หัวข้อที่ 1 ถึง 8 ตามลำดับ ตัวอย่างเช่น ในรูปที่ 4-3 หัวข้อที่ 1 หมายเลขประจำหัวข้อ แทนด้วย Y2 และชื่อหัวข้อ แทนด้วย Y3

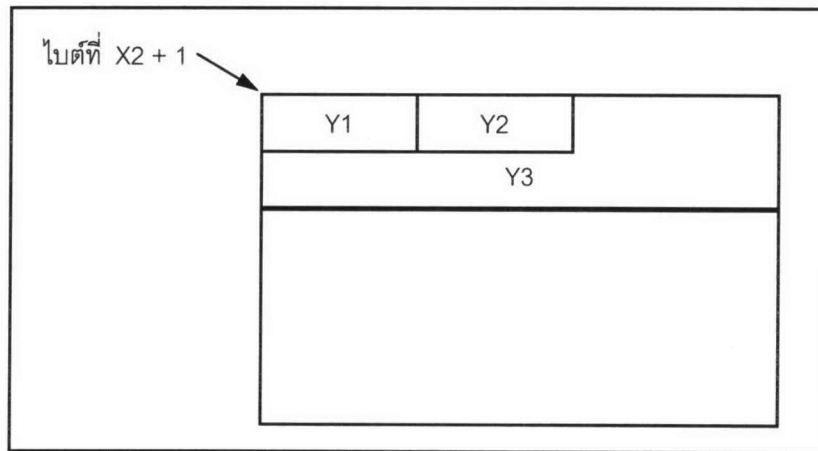


รูปที่ 4-3 รูปจำลองเนื้อหาส่วนรายละเอียดหัวข้อ

4. ส่วนรายละเอียดชื่อเรื่อง

- ไบต์ถัดไป 4 ไบต์ เก็บตัวเลขแบบไบนารี มีค่าเป็น 6 หมายถึง จำนวนชื่อเรื่องทั้งหมดในระบบ ในรูปที่ 4-4 แทนจำนวนชื่อเรื่องทั้งหมดด้วย Y1 เริ่มต้นที่ไบต์ที่ X2 + 1 ซึ่งอ้างไว้ในรูปที่ 4-1

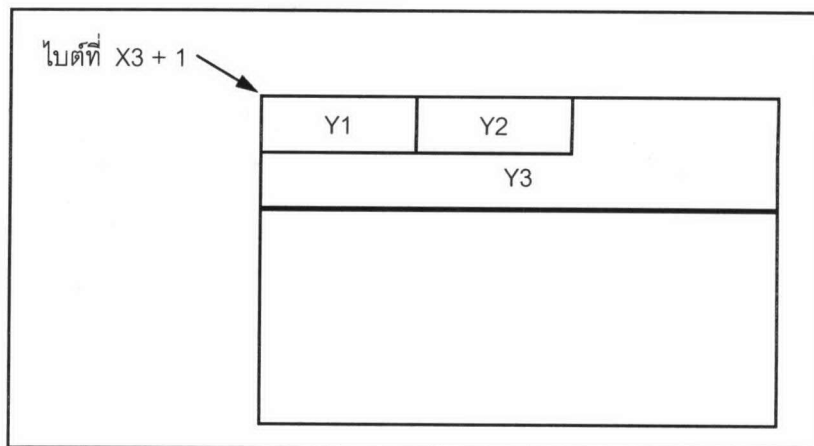
- ไบต์ถัดไป 4 ไบต์ เก็บหมายเลขประจำหัวข้อของชื่อเรื่อง และ 128 ไบต์ถัดไปเก็บชื่อเรื่อง ตั้งแต่ชื่อเรื่องของหัวข้อที่ 2 ถึง 7 ตามลำดับ ตัวอย่างเช่น ในรูปที่ 4-4 หัวข้อที่ 1 หมายเลขประจำหัวข้อ แทนด้วย Y2 และชื่อหัวข้อ แทนด้วย Y3



รูปที่ 4-4 รูปจำลองเนื้อหาส่วนรายละเอียดชื่อเรื่อง

5. ส่วนรายละเอียดคำหลัก

- ไบต์ถัดไป 4 ไบต์ เก็บตัวเลขแบบไบนารี มีค่าเป็น 28 หมายถึงจำนวนคำหลักทั้งหมดในระบบ ในรูปที่ 4-5 แทนจำนวนคำหลักทั้งหมดด้วย Y1 โดยเริ่มต้นที่ไบต์ที่ $X3 + 1$ ซึ่งอ้างไว้ในรูปที่ 4-1
- ไบต์ถัดไป 4 ไบต์ เก็บหมายเลขประจำหัวข้อของคำหลัก และ 128 ไบต์ถัดไปเก็บชื่อคำหลัก ตั้งแต่คำหลักของหัวข้อที่ 1 ถึง 8 ตามลำดับ ตัวอย่างเช่น ในรูปที่ 4-5 หัวข้อที่ 1 หมายเลขประจำหัวข้อ แทนด้วย Y2 และชื่อหัวข้อ แทนด้วย Y3



รูปที่ 4-5 รูปจำลองเนื้อหาส่วนรายละเอียดคำหลัก

จากการทดสอบพบว่า แฟ้มข้อมูลผลลัพธ์มีเนื้อหาตรงตามรูปแบบที่ออกแบบไว้ ซึ่งเป็นรูปแบบที่โปรแกรมแสดงข้อมูลไฮเปอร์ดอกคิวเมนต์จะสามารถอ่านไปประมวลผลได้ ทั้งนี้ ชื่อหัวข้อ ชื่อเรื่อง และคำหลัก

ต้องเป็นภาษาอังกฤษ และแยกแยะตัวพิมพ์เล็กและพิมพ์ใหญ่ เนื่องจาก RTF เก็บภาษาไทยแบบ 7 บิต และโปรแกรมตัวแปลไม่ได้ตรวจสอบเรื่องภาษาของชื่อดังกล่าว

การแปลเพิ่มข้อมูลนำเข้าให้เป็นเพิ่มข้อมูลผลลัพธ์ใช้เวลา 3 วินาทีในการประมวลผล โดยที่เพิ่มข้อมูลนำเข้ามีขนาด 27 กิโลไบต์ และเพิ่มข้อมูลผลลัพธ์มีขนาด 44 กิโลไบต์

4.3 ทดสอบการทำงานของโปรแกรมเครื่องมือแสดงข้อมูลไฮเปอร์ดอกคิวเมนต์

การทดสอบโปรแกรมเครื่องมือแสดงข้อมูลไฮเปอร์ดอกคิวเมนต์ ทำโดยให้โปรแกรมอ่านข้อมูลจากเพิ่มข้อมูลผลลัพธ์ ชื่อ infile.hdv เข้ามาทำการประมวลผลตามหน้าที่ต่าง ๆ ในโปรแกรม โดยที่ต้องมีเพิ่มข้อมูลรูปภาพบิตแมพ และเพิ่มข้อมูลเสียงอยู่ในสารบบเดียวกับที่สร้างในเพิ่มข้อมูลนำเข้าด้วย

ผลการทดสอบแต่ละหน้าที่การทำงาน เป็นดังนี้

4.3.1 การเปิดเพิ่มข้อมูล

เมื่อเลือก ชื่อเพิ่มข้อมูลแล้ว โปรแกรมจะแสดงข้อมูลของหน้าแรก ซึ่งจะเรียกว่าเป็นหน้าสารบัญ ถูกต้องตามที่กำหนดไว้ ใช้เวลา 10 วินาที

4.3.2 การแสดงตัวอักษร และ รูปภาพ

4.3.2.1 ข้อความทั่วไป โปรแกรมสามารถแสดงข้อความตามรูปแบบที่กำหนดในเพิ่มข้อมูลผลลัพธ์ ได้แก่ ฟอนต์ รูปแบบตัวอักษร ขนาด และ สี รวมถึงการเว้นวรรค และ ย่อหน้าใหม่

4.3.2.2 รูปภาพ โปรแกรมสามารถวาดภาพได้ถูกต้องตามจริง

4.3.2.3 ข้อความ สำหรับการกระโดด สามารถแสดงเป็นสีเขียว ชิดเส้นใต้ ได้ถูกต้อง และเมื่อเลื่อนเมาส์ไปบนบริเวณข้อความตัวชี้ของเมาส์จะเปลี่ยนเป็นรูปกดปุ่ม

4.3.2.4 ข้อความสำหรับการแสดงข้อมูลบนหน้าต่างซ้อนแบบผุดขึ้น สามารถแสดงเป็นสีน้ำเงิน ชิดเส้นใต้ ได้ถูกต้อง และเมื่อเลื่อนเมาส์ไปบนบริเวณข้อความตัวชี้ของเมาส์จะเปลี่ยนเป็นรูปกดปุ่ม

4.3.2.5 ข้อความสำหรับประมวลผลเสียง สามารถแสดงเป็นสีเขียว ตัวเอียง ชิดเส้นใต้ ได้ถูกต้อง และเมื่อเลื่อนเมาส์ไปบนบริเวณข้อความตัวชี้ของเมาส์จะเปลี่ยนเป็นรูปกดปุ่ม

จากการทดสอบพบว่า

- โปรแกรมจะไม่ประมวลผลแท็บ (Tab) รวมถึงการจัดวางตำแหน่งข้อความ เช่น กลาง หน้ากระดาษ หรือชิดขวา โดยโปรแกรมจะเริ่มวางข้อความนับจากด้านซ้ายเสมอ

- รูปภาพจะแสดงขีดซ้ายเสมอเช่นกัน และพื้นที่ด้านขวาของรูปภาพจะไม่สามารถแสดงข้อมูลอื่นได้
- โปรแกรมไม่ได้จัดการเรื่องการตัดคำเมื่อต้องขึ้นบรรทัดใหม่
- ข้อความที่ไม่ใช่พื้นที่สำคัญ จะต้อง ไม่ใช่ ขีดเส้นใต้เดี่ยว ขีดเส้นใต้คู่ หรือ ขีดกลางข้อความ เนื่องจากคุณลักษณะดังกล่าวใช้เป็นข้อกำหนดของพื้นที่สำคัญ แต่ถ้าใช้คุณลักษณะดังกล่าวกับข้อความปกติ โปรแกรมจะเห็นเป็นพื้นที่สำคัญ แต่จะไม่มีชื่อหัวข้อที่จะเชื่อมโยงไปหา โปรแกรมจะไปเรียกหัวข้อสุดท้ายของข้อมูลมาแสดงผล ซึ่งเป็นผลลัพธ์ที่ไม่ถูกต้อง

4.3.3 การแสดงข้อมูลบนจอซ้อนแบบผุดขึ้น

โปรแกรมสามารถแสดงข้อความ และรูปภาพ บนหน้าต่างซ้อนแบบผุดขึ้นได้ โดยที่ความกว้างของหน้าต่างจะเท่ากับความกว้างของเอกสาร กล่าวคือ ยังไม่มีการผันแปรตามความยาวของข้อความ ส่วนความยาวจะแปรผันตามความยาวของข้อความ

4.3.4 การกระโดด

เมื่อคลิกเมาส์บนข้อความสำหรับการกระโดด หรือข้อความสีเขียว ขีดเส้นใต้ โปรแกรมจะแสดงข้อมูลของหัวข้อที่เลือก ดังนี้

- 4.3.4.1 คลิกที่ข้อความ “วัดและโบราณสถาน” โปรแกรมกระโดดไปยังหัวข้อ wat ใช้เวลา 10 วินาที
- 4.3.4.2 คลิกที่ข้อความ “แหล่งท่องเที่ยวธรรมชาติ” โปรแกรมกระโดดไปยังหัวข้อ place ใช้เวลา 10 วินาที
- 4.3.4.3 คลิกที่ข้อความ “ศิลปะหัตถกรรม” โปรแกรมกระโดดไปยังหัวข้อ hilltribe ใช้เวลา 7 วินาที
- 4.3.4.4 คลิกที่ข้อความ “สถานที่พัก” โปรแกรมกระโดดไปยังหัวข้อ shopping ใช้เวลา 7 วินาที
- 4.3.4.5 คลิกที่ข้อความ “ร้านอาหาร” โปรแกรมกระโดดไปยังหัวข้อ hotel ใช้เวลา 7 วินาที

4.3.5 การทำงานแฟ้มข้อมูลเสียง

เมื่อคลิกเมาส์บนข้อความที่จะประมวลผลเสียง คือ “เสียงจากเสียงราย” ที่หัวข้อ content โปรแกรมให้เสียงตามที่เก็บไว้ในแฟ้มข้อมูลเสียง “c:\thesis\picNsnd\intro.wav” ซึ่งมีขนาด 53 กิโลไบต์ ใช้เวลาประมวลผล 3 วินาที

จากการทดสอบพบว่า ขณะที่โปรแกรมทำการประมวลผลเสียง จะไม่สามารถทำงานอื่นได้ ต้องรอให้เสียงจบเสียงก่อน จึงทำงานอื่นต่อได้

4.3.6 การค้นหาจากคำหลัก

เมื่อเลือกค้นหา ที่จอภาพจะปรากฏรายชื่อคำหลักครบตามที่กำหนดไว้ใน แฟ้มข้อมูลนำเข้า เมื่อเลือก “doi tung” ซึ่งเป็นคำหลักของ 2 หัวข้อ คือ wat และ place โปรแกรมจะเปิดหน้าต่างใหม่แสดงรายชื่อชื่อเรื่องของทั้ง 2 หัวข้อ ได้ถูกต้อง และเมื่อเลือก 1 ชื่อเรื่อง โปรแกรมสามารถแสดงข้อมูลของชื่อเรื่องที่เลือกได้ถูกต้อง และสำหรับกรณีคำหลักนั้นเป็นของ 1 หัวข้อ ทำการทดสอบโดยเลือก “nature” โปรแกรมสามารถแสดงข้อมูลของหัวข้อของคำหลักที่เลือกได้ถูกต้อง

สำหรับความเร็วในการขึ้นรายชื่อคำหลัก หรือรายชื่อชื่อเรื่อง ในการทดสอบกับข้อมูลชุดนี้ถือได้ว่าไม่มีนัยสำคัญ เนื่องจากมีจำนวนคำหลักเพียง 28 คำ ทั้งนี้ได้มีการทดสอบในเรื่องการเพิ่มชื่อในรายการจำนวน 10,000 รายการ ใช้เวลาประมาณ 3 วินาที

ตามทฤษฎีที่กำหนดในโครงสร้างข้อมูลใช้เนื้อที่ 4 ไบต์ในการเก็บจำนวนหัวข้อทำให้สามารถมีได้ถึง 4,294,967,295 หัวข้อ อย่างไรก็ตามในทางปฏิบัติจะมีได้ไม่ถึงจำนวนดังกล่าว เนื่องจากแฟ้มข้อมูลผลลัพธ์มีขนาดสูงสุดได้ที่ 4,294,967,295 ไบต์เช่นกัน และแต่ละหัวข้อยังต้องใช้เนื้อที่ในการเก็บรายละเอียดการควบคุมของ RTF อีกจำนวนหนึ่งซึ่งมากน้อยตามคุณลักษณะของข้อมูลในหัวข้อ

4.3.7 การเขียนบันทึกข้อความ

การบันทึกข้อความโปรแกรมจะแสดงหน้าต่างสำหรับให้พิมพ์ข้อความให้กับหัวข้อที่แสดงบนหน้าต่างขณะนั้น จากนั้นข้อความจะปรากฏที่ด้านบนของหน้าต่าง โดยไม่มีการจัดรูปแบบ หรือคุณลักษณะใด ๆ ให้กับตัวอักษร โดยเมื่อใดที่มีการแสดงข้อมูลของหัวข้อที่มีการบันทึกข้อความไว้ ข้อความจะปรากฏพร้อมกับข้อมูลของหัวข้อนั้นเสมอ ข้อความที่บันทึกนี้สามารถเรียกขึ้นมาแก้ไข และลบออกได้

จากการทดสอบพบว่า สามารถทำงานได้ตามขอบเขตงานที่กำหนด โดยที่ข้อความจะต้องเป็นบรรทัดเดียว และยาวไม่เกินความกว้างของเอกสาร มิฉะนั้นจะข้อความบันทึกส่วนที่เกินจะไม่แสดงบนหน้าต่าง

4.3.8 การเลือกให้แสดงข้อมูลจากรายชื่อหัวข้อที่เคยเรียกดูแล้ว

จากการทดสอบพบว่าเมื่อมีการกระโดด โปรแกรมจะบันทึกชื่อหัวข้อไว้ในรายชื่อประวัติ (history list) ดังนั้น เมื่อเลือกเมนู history... จะปรากฏรายชื่อหัวข้อให้เลือก เมื่อเลือกหัวข้อแล้ว ที่หน้าต่างจะแสดงข้อความของหัวข้อนั้นให้ทันที ไม่มีนัยสำคัญของเวลาที่ใช้

เมื่อมีการบันทึกหัวข้อในประวัติจนครบ 10 หัวข้อแล้ว เมื่อมีการกระโดดอีก หัวข้อที่ 11 จะถูกเก็บเป็นหัวข้อที่ 10 โดยโปรแกรมจะลบหัวข้อที่ 1 ออก แล้วเลื่อนรายชื่อหัวข้อขึ้น 1 ลำดับ

4.3.9 การเรียกให้แสดงข้อมูลของหัวข้อถัดไปที่เคยเรียกดูแล้ว

จากการทดสอบพบว่า โปรแกรมสามารถเรียกข้อมูลของหัวข้อถัดไปที่เคยเรียกดูแล้ว หรือที่โปรแกรมเคยกระโดดไปแล้ว โดยที่เมื่อเรียกดูจนถึงหัวข้อสุดท้ายที่มีการกระโดด เมื่อเลือกทำงานในข้อนี้ โปรแกรมจะไม่ทำงานใด ๆ

4.3.10 การไปหัวข้อย้อนกลับ

จากการทดสอบพบว่า โปรแกรมสามารถเรียกข้อมูลของหัวข้อก่อนหน้าที่เคยเรียกดูแล้ว หรือที่โปรแกรมเคยกระโดดไปแล้ว โดยที่เมื่อเรียกดูจนถึงหัวข้อแรกที่เคยกระโดดไป แล้วเลือกทำงานในข้อนี้โปรแกรมจะไม่ทำงานใด ๆ

4.4 ทดสอบการทำงานของโปรแกรม OLE

ทำการทดสอบ OLE Server ชื่อ "HyperDocument" ซึ่งมีเม็ทธอด (method) ให้เรียกใช้ 3 เม็ทธอด คือ OpenFile Jump และ CloseFile ด้วยการเขียนโปรแกรม Visual Basic 4.0 โปรเจ็คใหม่ กำหนดให้เมื่อไหลหน้าต่าง ให้โปรแกรมสร้างออบเจ็คของไฮเปอร์ดอกคิวเมนต์ สร้างปุ่มให้เรียกใช้ OpenFile โดยส่งชื่อ "c:\thesis\infile.hdv" ให้ สร้างปุ่มเรียกใช้ Jump โดยส่งชื่อหัวข้อ "wat" ไปให้ และปุ่มเรียกใช้ CloseFile เพื่อปิดหน้าต่างไฮเปอร์ดอกคิวเมนต์

เมื่อวิ่งโปรแกรม หน้าต่างของโปรแกรม จะปรากฏขึ้นพร้อมกับหน้าต่างไฮเปอร์ดอกคิวเมนต์ เมื่อกดปุ่ม OpenFile ที่หน้าต่างไฮเปอร์ดอกคิวเมนต์จะไหลข้อมูลหัวข้อแรกเข้ามา และเมื่อกดปุ่ม Jump ที่หน้าต่างไฮเปอร์ดอกคิวเมนต์จะไหลข้อมูลหัวข้อของ "wat" เข้ามา และเมื่อกดปุ่ม CloseFile หน้าต่างไฮเปอร์ดอกคิวเมนต์จะปิดไป

จะเห็นว่าโปรแกรมเครื่องมือแสดงข้อมูลไฮเปอร์ดอกคิวเมนต์ สามารถทำงานได้ตามมาตรฐาน OLE Server โดยสามารถให้โปรแกรมอื่นเรียกใช้เม็ทธอดที่มีได้