บทที่ 4

การออกแบบโครงสร้างข้อมูลสำหรับฟังก์ชันจัดการแฟ้มข้อมูลดัชนี

โครงสร้างแฟ้มข้อมูล (File Structure)

แฟ้มข้อมูลที่เกี่ยวข้องกับการทำงานของฟังก์ชันจัดการแฟ้มข้อมูล ประกอบด้วย

1. แฟ้มข้อมูลดัชนี

เป็นแฟ้มข้อมูลที่เก็บบันทึกระเบียนข้อมูลที่ส่งมา เนื่องจากฟังก์ชันการเพิ่มเติมข้อมูล หลังจากที่ทำการเพิ่มเติมระเบียนข้อมูลฟังก์ชันจะสร้างดัชนีขึ้นในแฟ้มข้อมูลเดียวกัน ซึ่งกำหนดตามคุณสมบัติของคีย์ในแฟ้มข้อมูลนั้น เพื่อใช้สำหรับการค้นหาและดึงระเบียนข้อมูลออกจากแฟ้ม

คุณสมบัติของระเบียนข้อมูล ประกอบด้วย

1. เนื้อหาระเบียนข้อมูล อาจประกอบด้วยข้อมูลชนิดเดียวกันหรือหลายชนิดรวมกัน เช่น ข้อมูลที่เป็นกลุ่มตัวอักษร หรือเป็นข้อมูลเลขจำนวนเต็ม ซึ่งเนื้อหาของข้อมูลแต่ละชนิดจะต้องไม่มีรหัสควบคุมใด ๆ คั่นอยู่

2. ระเบียนข้อมูลทุกระเบียนมีความยาวคงที่(fix length)

3. ความยาวระเบียนข้อมูลกำหนดได้มากที่สุดเท่ากับ 4090 ไบต์

4. ความยาวระเบียนข้อมูล ซึ่งต้องกำหนดในระหว่างสร้างแฟ้มข้อมูลดัชนีขึ้นกับการกำหนดขนาดแผ่นภายในแฟ้ม และจำนวนชุดคีย์ซึ่งกำหนดให้ภายในชุดคีย์นั้นมีค่าคีย์ที่ซ้ำกันได้

คุณสมบัติของคีย์ ประกอบด้วย

1. เนื้อหาคีย์อาจประกอบด้วยข้อมูลจากตำแหน่งต่าง ๆ ในระเบียนข้อมูลเพียงเซกเมนต์เดียวหรืออาจมากกว่าหนึ่งเซกเมนต์คีย์ขึ้นไป โดยที่เนื้อหาคีย์นั้นอาจเหลื่อม (overlap) กันได้

2. คีย์ทุกชุดที่กำหนดขึ้นต้องมีความยาวคงที่ และความยาวทั้งหมดของคีย์ต้องเท่ากับ ผลรวมของความทุกเซกเมนต์คีย์ที่กำหนดในชุดคีย์นั้น โดยที่ความยาวคีย์ที่สามารถกำหนดได้มากที่สุดเท่ากับ 255 ไบต์

3. ภายในแฟ้มข้อมูลดัชนี กำหนดให้มีเนื้อที่สำหรับบันทึกคุณสมบัติคีย์ได้มากที่สุด 24 เซกเมนต์ จำนวนชุดคีย์จะขึ้นกับจำนวนเซกเมนต์ที่กำหนดให้แต่ละชุดคีย์ กล่าวคือ ถ้าแฟ้มข้อมูลดัชนีที่มีเพียงชุดคีย์เดียว จะสามารถกำหนดเซกเมนต์คีย์ได้มากที่สุด 24 เซกเมนต์ และ ในทางกลับกันถ้าชุดคีย์แต่ละชุดมีเพียงเซกเมนต์คีย์เดียว แฟ้มข้อมูลดัชนีนั้น. จะสามารถกำหนดชุดคีย์ได้มากที่สุด 24 ชุดเช่นกัน นอกจากนี้จำนวนเนื้อที่สำหรับบันทึกคุณสมบัติคีย์ยังขึ้นกับขนาดแผ่นในแฟ้ม คือ ถ้ากำหนดขนาดแผ่นเท่ากับ 512 ไบต์จะกำหนดคุณสมบัติของคีย์ได้มากที่สุดเพียง 12 เซกเมนต์ แต่ถ้ากำหนดขนาดแผ่นมากกว่า 512 ไบต์ขึ้นไป จะสามารถกำหนดคุณสมบัติของคีย์ได้มากถึง 24 เซกเมนต์

4. ประเภทข้อมูลของคีย์ กำหนดได้ 2 ชนิด คือ

4.1 คีย์ที่ประกอบด้วยกลุ่มตัวอักษร

4.2 คีย์ที่เป็นเลขจำนวนเต็มไม่คิดเครื่องหมาย(unsigned integer) ขนาดคีย์ประเภทนี้กำหนดได้เพียง 2 หรือ 4 ไบต์เท่านั้น

5. การกำหนดคุณสมบัติคีย์ เพื่อให้มีการตรวจสอบเมื่อทำการเพิ่มเติมหรือเปลี่ยนแปลง หรือค้นหาข้อมูล โดยระบุค่าแฟลกให้กับคีย์ ประกอบด้วย

5.1 คุณสมบัติของคีย์ที่กำหนดให้มีคีย์ที่ซ้ำกันได้ เมื่อมีการเพิ่มข้อมูลใหม่ในแฟ้ม

5.2 คุณสมบัติของคีย์ที่สามารถเปลี่ยนแปลงค่าได้ เมื่อมีการแก้ไขระเบียนข้อมูลในแฟ้ม

5.3 คุณสมบัติการใช้ชุดการสลับลำดับตัวอักษร(Alternate Sequence Character set) เพื่อใช้เปลี่ยนลำดับของรหัสตัวอักษรภายในคีย์ก่อนทำการเปรียบเทียบค่าคีย์และจัดลำดับให้กับคีย์

5.4 คุณสมบัติที่ระบุว่าคีย์เป็นตัวอักษรไทย เพื่อให้การเปรียบเทียบข้อมูลเป็นไปตามลักษณะตัวอักษรไทย

องค์ประกอบภายในแฟ้มข้อมูลดัชนี ได้ออกแบบให้ใช้บันทึกทั้งระเบียนข้อมูลและโครงสร้างดัชนีของชุดคีย์อยู่ภายในแฟ้มเดียวกัน และเพื่อให้ใช้งานได้กับช่องใส่แผ่น

ของบัฟเฟอร์ที่ออกแบบไว้ ข้อมูลทั้งหมดในแฟ้มข้อมูลดัชนีจึงแบ่งหน่วยการบันทึกข้อมูลออกเป็นส่วน ๆ มีขนาดเท่ากัน เรียกว่า แผ่น สำหรับการส่งผ่านข้อมูลจากจานบันทึกไปเก็บชั่วคราวในบัฟเฟอร์หน่วยความจำ ซึ่งสามารถกำหนดขนาดแผ่นได้ตั้งแต่ 512 - 4096 ไบต์ และต้องเป็นผลคูณของ 512 ไบต์เช่นเดียวกับขนาดช่องใส่แผ่น ประเภทแผ่นที่กำหนดในแฟ้มข้อมูลดัชนี ประกอบด้วย

1. แผ่นระเบียนควบคุม (Header Page หรือ Control Information Page) ประกอบด้วยระเบียนควบคุมที่เกี่ยวกับคุณสมบัติของระเบียนข้อมูลคีย์ต่าง ๆ ที่กำหนดขึ้นในการสร้างแฟ้ม รวมทั้งสารสนเทศที่เกี่ยวข้องกับการเชื่อมโยงแผ่นทุกชนิด ที่มีภายในแฟ้มข้อมูลดัชนีนั้น เนื้อหาภายในแผ่นประกอบด้วยบล๊อกข้อมูลที่สำคัญดังนี้

1.1 บล๊อกสารสนเทศควบคุม ประกอบด้วย

1.1.1 หมายเลขแผ่นระเบียนควบคุม มีค่าเท่ากับ 0 เสมอ

1.1.2 แฟลกที่ระบุสถานภาพแฟ้มว่า กำลังอยู่ในระหว่างการบันทึกข้อมูลจากช่องใส่แผ่นลงจานบันทึกหรือไม่

1.1.3 ลักษณะการเปิดแฟ้มข้อมูลดัชนี รูปแบบการเปิดแฟ้มประกอบด้วย

1.1.3.1 การเปิดแฟ้มข้อมูลดัชนี และสร้างแฟ้มสำรองแผ่นเพื่อใช้ในระหว่างการเพิ่มหรือลบ หรือแก้ไขข้อมูล

1.1.3.2 การเปิดแฟ้มข้อมูลดัชนีเพียงอย่างเดียว

1.1.3.3 การเปิดแฟ้มข้อมูลดัชนีในลักษณะการใช้อ่านเฉพาะระเบียนข้อมูลเพียงอย่างเดียว ในกรณีที่ต้องการนำระเบียนข้อมูลออกจากแฟ้มที่เสียหาย เนื่องจากไม่สามารถเปิดใช้ในลักษณะ 2 รูปแบบแรก

1.1.4 แฟลกที่ระบุความสามารถในการเข้าถึงระเบียนข้อมูลหลังจากเปิดแฟ้มรูปแบบการเข้าถึงระเบียนข้อมูล ประกอบด้วย

1.1.4.1 การค้นหาและอ่านและบันทึกระเบียนข้อมูลได้

1.1.4.2 การค้นหาและอ่านได้เพียงอย่างเดียว โดยการใช้คีย์สำหรับการค้นหา หรือ อ่านระเบียนข้อมูลจากตำแหน่งปัจจุบัน

1.1.4.3 การอ่านระเบียนข้อมูลจากตำแหน่งปัจจุบันในแผ่นข้อมูล ใช้สำหรับการอ่านระเบียนข้อมูลจากแฟ้มที่เสียหาย

1.1.5 หมายเลขที่ใช้ระบุแฟ้ม (File Identical Number)

1.1.6 จำนวนแผ่นทั้งหมดภายในแฟ้ม

1.1.7 จำนวนที่ว่างของแผ่นที่มีในแฟ้ม(Number of Free Page)

1.1.8 หมายเลขแผ่นที่ว่างแผ่นแรกที่จะนำกลับมาใช้ (First Free Page Number)

1.1.9 หมายเลขแผ่นข้อมูลเริ่มต้น (First Data Page Number)

1.1.10 ตำแหน่งสุดท้ายของระเบียนข้อมูล (Last Data Address) ประกอบด้วย หมายเลขแผ่นข้อมูลสุดท้าย (Last Data Page Number) และ ลำดับที่ระเบียนข้อมูลในแผ่น (Last Data Item Number)

1.1.11 หมายเลขแผ่นแผนที่ข้อมูลเริ่มต้น (First Data Map Page Number)

1.1.12 ตำแหน่งสุดท้ายของสมาชิกแผนที่ข้อมูล (Last Data Map address) ประกอบด้วย หมายเลขแผ่นแผนที่ข้อมูลสุดท้าย และ ลำดับที่สมาชิกในแผ่นแผนที่ข้อมูล

1.1.13 ขนาดแผ่นของแฟ้มข้อมูลดัชนี

1.1.14 ความยาวระเบียนข้อมูล

1.1.15 จำนวนระเบียนข้อมูลที่กำหนดขึ้นในแฟ้ม โดยที่ จำนวนระเบียนข้อมูลที่แท้จริง = จำนวนระเบียนที่กำหนดขึ้น - จำนวนที่ว่างของระเบียนข้อมูล

1.1.16 จำนวนที่ว่างของระเบียนข้อมูล (Number of Free Record Slots)

1.1.17 ตำแหน่งแรกของที่ว่างระเบียนข้อมูลที่จะนำกลับมาใช้ (First Free Record slot address) ประกอบด้วย หมายเลขแผ่นข้อมูล และ ลำดับที่ระเบียนข้อมูลในแผ่น

1.1.18 จำนวนระเบียนข้อมูลที่มีได้ในแผ่นข้อมูล (Number of records on Data Page)

1.1.19 จำนวนสมาชิกที่มีได้ในแผ่นแผนที่ข้อมูล (Number of items on Data Map Page)

1.1.20 จำนวนชุดคีย์ในแฟ้ม (Number of Key)

1.1.21 จำนวนชุดคีย์ที่กำหนดให้คีย์มีค่าซ้ำกันได้ (Number of Duplicate key)

1.1.22 ไดเรคทอรี่ปัจจุบัน (Current Directory) ที่เก็บบันทึกแฟ้มข้อมูลดัชนี

1.1.23 หน่วยบันทึกแฟ้มสำรองแผ่นสำหรับแฟ้มข้อมูลดัชนี ประกอบด้วย หมายเลขหน่วยจานบันทึก ได้แก่ 1 แทน A: , 2 แทน B: ตามลำดับ และ ชื่อแฟ้มสำรองแผ่นกับชื่อ extension

1.1.24 ชื่อชุดการสลับลำดับตัวอักษรที่กำหนดภายในแฟ้ม

1.2 บล๊อกสารสนเทศของชุดคีย์ ประกอบด้วยแถวลำดับของระเบียนควบคุมที่เรียงตามลำดับของชุดคีย์ที่กำหนดขึ้น ภายในระเบียนควบคุมนั้นประกอบด้วย

1.2.1 จำนวนดัชนีภายในแผ่นดัชนี (Number of index on Index Page)

1.1.2 หมายเลขแผ่นดัชนีที่เป็นโนดรากของชุดคีย์

1.2.3 ลำดับที่ของชุดคีย์ที่กำหนดให้มีคีย์ที่ซ้ำกันได้ ถ้าชุดคีย์นั้นไม่กำหนดให้คีย์มีค่าซ้ำกัน ลำดับที่จะเท่ากับ 0 เสมอ

1.2.4 ลำดับที่เซกเมนต์คีย์ภายในบล๊อกสารสนเทศของเซกเมนต์คีย์ เพื่อใช้ระบุตำแหน่งเริ่มต้นของเซกเมนต์แรกสำหรับชุดคีย์นั้น

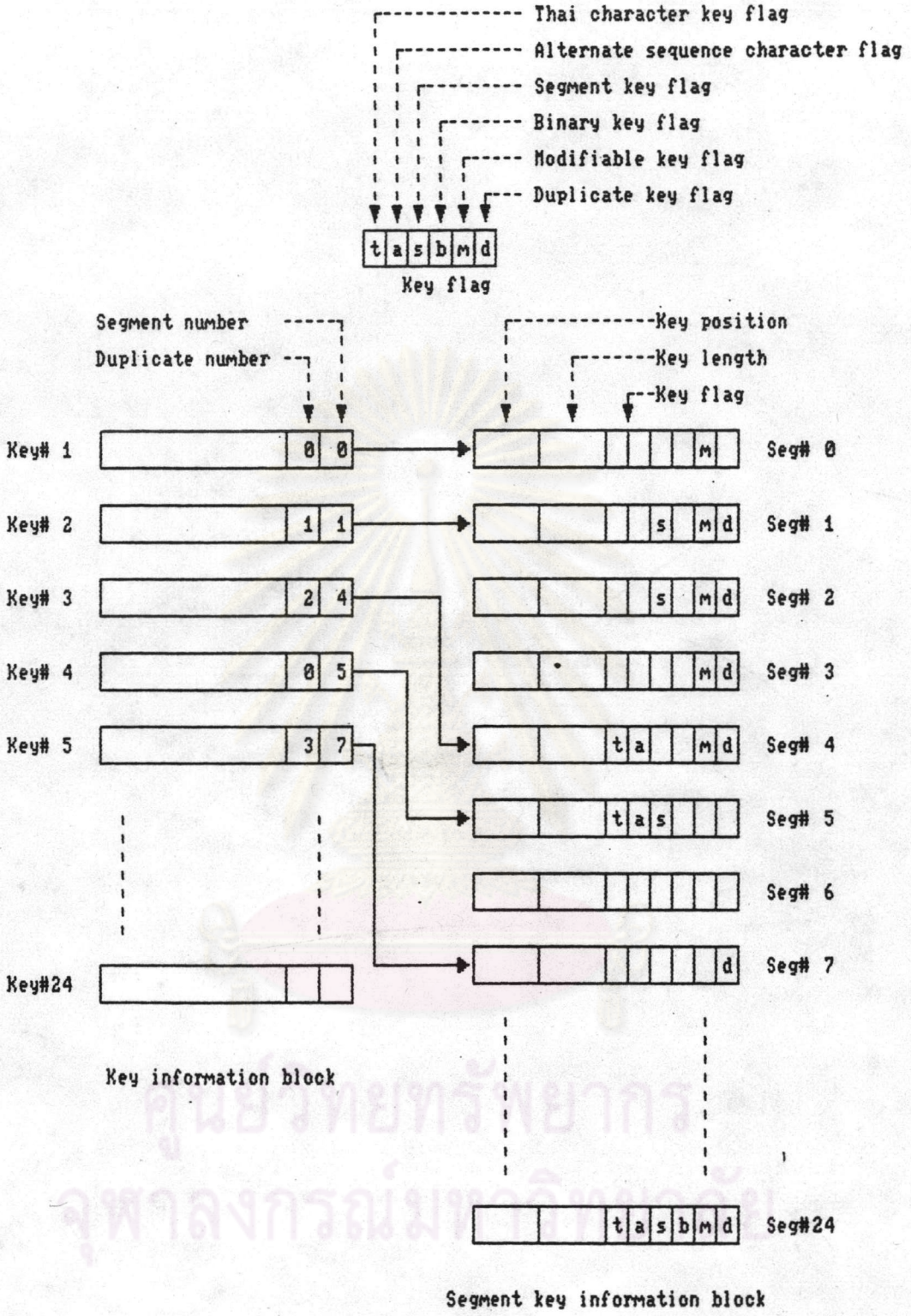
1.2.5 จำนวนดัชนีที่สร้างขึ้นในชุดคีย์

1.3 บล๊อกสารสนเทศสำหรับเซกเมนต์คีย์ ประกอบด้วยแถวลำดับของข้อมูลเซกเมนต์คีย์ที่กำหนดมาจากชุดคีย์ต่าง ๆ เรียงติดต่อกัน ภายในแต่ละเซกเมนต์ประกอบด้วย

1.3.1 ตำแหน่งเริ่มต้นของคีย์ภายในระเบียนข้อมูล

1.3.2 ความยาวของเซกเมนต์

1.3.3 แฟลกที่ระบุ คุณสมบัติของเซกเมนต์คีย์ ประกอบด้วย



ภาพที่ 4.1 ลักษณะโครงสร้างบล็อกสารสนเทศของชุดคีย์และเซกเมนต์คีย์

1.3.3.1 คุณสมบัติที่อนุญาตให้คีย์มีค่าซ้ำกันได้

1.3.3.2 คุณสมบัติที่อนุญาติให้คีย์เปลี่ยนแปลงค่าได้

1.3.3.3 คุณสมบัติที่แสดงว่าคีย์เป็นข้อมูลชนิดเลขจำนวนเต็ม

1.3.3.4 คุณสมบัติที่แสดงว่าคีย์เป็นข้อมูลชนิดกลุ่มตัวอักษรที่มีการจัดเรียงตามหลักของอักษรภาษาไทย

1.3.3.5 คุณสมบัติที่ระบุการใช้ชุดการสลับลำดับตัวอักษรในการเปรียบเทียบค่าคีย์

1.3.3.6 คุณสมบัติที่ระบุว่าเป็นเซกเมนต์คีย์ ซึ่งหมายความว่า ข้อมูลเซกเมนต์คีย์ถัดไปจะเป็นเซกเมนต์ที่อยู่ในชุดคีย์เดียวกับเซกเมนต์นี้

1.4 บล๊อกข้อมูลของชุดการสลับลำดับตัวอักษร จะถูกกำหนดขึ้นต่อเมื่อมีการระบุแฟลกคุณสมบัติการใช้ชุดการสลับลำดับตัวอักษร ให้กับเซกเมนต์คีย์เซกเมนต์ใดเซกเมนต์หนึ่ง และสามารถกำหนดชุดการสลับลำดับได้เพียงชุดเดียวในแฟ้มข้อมูลดัชนีนั้น บล๊อกข้อมูลนี้ จะกำหนดขึ้นในแผ่นระเบียนควบคุมเท่านั้นเมื่อขนาดแผ่นของแฟ้มมีมากกว่า 512 ไบต์ แต่ถ้าแผ่นมีขนาดเท่ากับ 512 ไบต์ บล๊อกข้อมูลนี้จะถูกบรรจุในแผ่นหมายเลขที่ 1 ของแฟ้ม ภายในบล๊อกประกอบด้วยรหัสแอสกีของตัวอักษรที่ถูกจัดลำดับใหม่ ซึ่งมีตั้งแต่รหัสแอสกีที่ 0 ถึงรหัสที่ 255 โดยการบรรจุรหัสแอสกีลำดับที่ใหม่ ลงในตำแหน่งที่ของรหัสเดิมภายในตารางรหัสแอสกี 256 ตำแหน่ง

2. แผ่นดัชนี (Index Page) ใช้บันทึกโนดที่เป็นส่วนหนึ่งของโครงสร้างบี-ทรี สำหรับแต่ละชุดคีย์ ภายในแผ่นมีการเรียงลำดับดัชนีตามค่าของคีย์ ซึ่งประกอบด้วย

2.1 ระเบียนควบคุมสำหรับแผ่นดัชนี ประกอบด้วย

2.1.1 หมายเลขแผ่น

2.1.2 จำนวนดัชนีที่มีอยู่ในแผ่นนั้น

2.1.3 หมายเลขแผ่นดัชนี ซึ่งมีค่าคีย์ภายในแผ่นนั้นน้อยกว่าคีย์ของดัชนีลำดับแรกในแผ่นดัชนีนี้ (Extra index page)

2.2 แถวลำดับของดัชนีภายในแผ่น ซึ่งเนื้อหาภายในดัชนีกำหนดได้เป็น 2 แบบ โดยขึ้นกับ

2.2.1 การกำหนดไม่ให้คีย์มีค่าที่ซ้ำกันได้ ภายในดัชนีจึงประกอบด้วย

2.2.1.1 คีย์หรือเป็นกลุ่มของเซกเมนต์คีย์ที่ไม่มีรหัสควบคุมใด ๆ คั่น

2.2.1.2 หมายเลขแผ่นดัชนีที่ค่าคีย์ภายในแผ่นมากกว่าคีย์ของดัชนีนี้ แต่มีค่าน้อยกว่าคีย์ของดัชนีลำดับถัดไป

2.2.1.3 ตำแหน่งของระเบียนข้อมูล (data record address) สำหรับคีย์ของดัชนีนี้ ประกอบด้วย หมายเลขแผ่นข้อมูล และ ลำดับระเบียนข้อมูลภายในแผ่น

2.2.2 การกำหนดให้คีย์มีค่าซ้ำกันได้ ภายในดัชนีประกอบด้วย

2.2.2.1 คีย์หรือเป็นกลุ่มของเซกเมนต์คีย์ที่ไม่มีรหัสควบคุมใด ๆ คั่น

2.2.2.2 หมายเลขแผ่นดัชนีที่ค่าคีย์ภายในแผ่นมากกว่าคีย์ของดัชนี แต่มีค่าน้อยกว่าคีย์ของดัชนีลำดับถัดไป

2.2.2.3 ตำแหน่งของระเบียนข้อมูลแรก

2.2.2.4 ตำแหน่งของระเบียนข้อมูลสุดท้ายที่มีคีย์ที่ซ้ำกับระเบียนแรก

การคำนวณหาจำนวนดัชนีที่มีได้ในแผ่นดัชนี ได้แบ่งออกเป็น 2 แบบ เช่นเดียวกันคือ

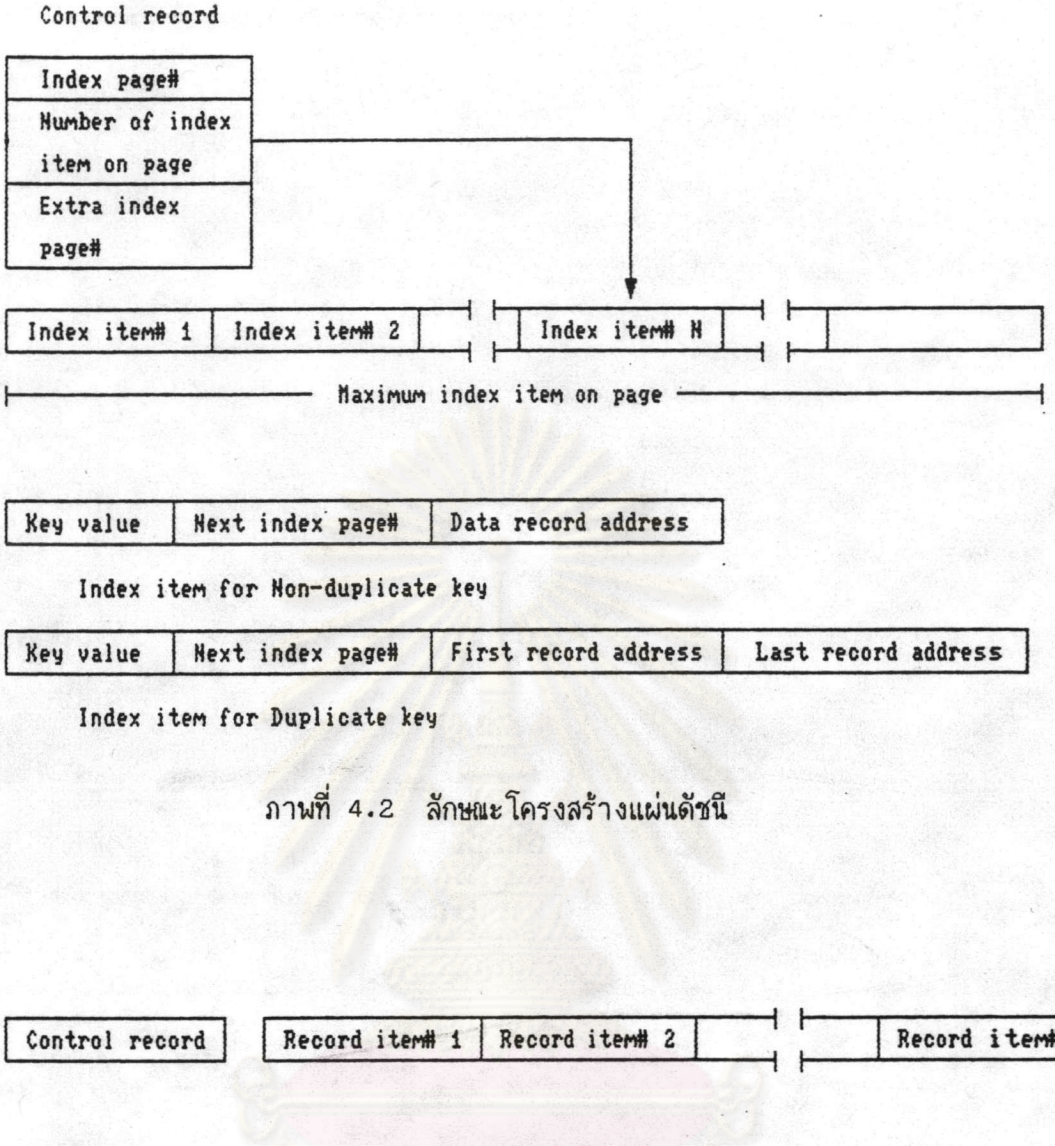
1. ในกรณีกำหนดไม่ให้มีคีย์ที่ซ้ำกันได้

$$\text{จำนวนดัชนีในแผ่น} = \frac{(\text{ขนาดแผ่น} - \text{ขนาดระเบียนควบคุม})}{(\text{ขนาดของดัชนีที่กำหนดไม่ให้มีคีย์ที่ซ้ำกัน})}$$

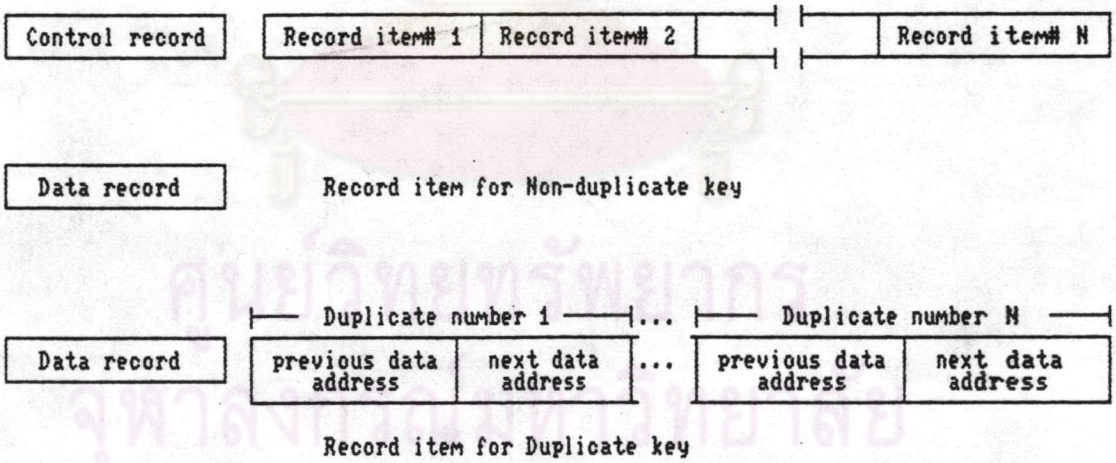
2. ในกรณีที่กำหนดให้มีคีย์ที่ซ้ำกันได้

$$\text{จำนวนดัชนีในแผ่น} = \frac{(\text{ขนาดแผ่น} - \text{ขนาดระเบียนควบคุม})}{(\text{ขนาดของดัชนีที่กำหนดให้มีคีย์ที่ซ้ำกันได้})}$$

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



ภาพที่ 4.2 ลักษณะโครงสร้างแผ่นดัชนี



ภาพที่ 4.3 ลักษณะโครงสร้างแผ่นข้อมูล

3. แผ่นข้อมูล (Data Page) ที่ใช้บันทึกระเบียนข้อมูลโดยเรียงลำดับของระเบียนข้อมูลตามลำดับการเพิ่มเติมระเบียนข้อมูลภายในแฟ้มข้อมูลดัชนี ลำดับจะเป็นไปตามนี้เมื่อไม่มีการลบระเบียนข้อมูลใด ๆ ในแฟ้ม ถ้ามีการลบระเบียนข้อมูล การเพิ่มเติมระเบียนในครั้งต่อไปจะใช้ที่ว่างของระเบียนที่ถูกลบไปสำหรับบันทึกข้อมูล ภายในแผ่นข้อมูลประกอบด้วย

3.1 ระเบียนควบคุมสำหรับแผ่นข้อมูล ประกอบด้วย

3.1.1 หมายเลขแผ่น

3.1.2 หมายเลขแผ่นข้อมูลก่อนหน้า (previous data page)

3.1.3 หมายเลขแผ่นข้อมูลถัดไป (next data page)

3.2 แถวลำดับของระเบียนข้อมูล สำหรับเนื้อหาที่บันทึกในแผ่นข้อมูลของแต่ละระเบียน จะขึ้นกับการกำหนดให้คีย์มีค่าที่ซ้ำกันได้หรือไม่ เช่นเดียวกับแผ่นดัชนี เนื้อหาจึงแบ่งออกเป็น 2 รูปแบบคือ

3.2.1 ในกรณีกำหนดไม่ให้คีย์มีค่าซ้ำกันได้ เนื้อหาของแถวลำดับจะประกอบด้วยระเบียนข้อมูลเพียงอย่างเดียว ซึ่งเรียงติดต่อกันโดยไม่มีรหัสควบคุมใด ๆ คั่นระหว่างระเบียน

3.2.2 ในกรณีกำหนดให้คีย์มีค่าซ้ำกันได้ เนื้อหาในแถวลำดับจะประกอบด้วย ระเบียนข้อมูล และ แถวลำดับของตำแหน่งระเบียนข้อมูลที่ใช้อ้างถึงกันภายในกลุ่มระเบียนข้อมูลที่มีค่าคีย์เท่ากัน จำนวนสมาชิกภายในแถวลำดับนี้ จะเท่ากับจำนวนชุดคีย์ที่กำหนดให้คีย์มีค่าซ้ำกันได้ โดยที่เรียงลำดับตามชุดคีย์ที่กำหนดคุณสมบัตินี้ แต่ละสมาชิกประกอบด้วย ตำแหน่งระเบียนข้อมูลก่อนหน้า และ ตำแหน่งระเบียนข้อมูลถัดไป

การคำนวณจำนวนระเบียนข้อมูลที่มีได้ในแผ่นข้อมูล จึงแบ่งเป็น 2 แบบเช่นเดียวกับแผ่นดัชนีคือ

1. ในกรณีที่กำหนดไม่ให้คีย์มีค่าซ้ำกันได้

$$\text{จำนวนระเบียนข้อมูลภายในแผ่น} = \frac{(\text{ขนาดแผ่น} - \text{ขนาดระเบียนควบคุม})}{\text{ความยาวระเบียนข้อมูล}}$$

I10308684

2. ในกรณีที่กำหนดให้คีย์มีค่าซ้ำกันได้

จำนวนระเบียนข้อมูลในแผ่น =

$$\frac{\text{(ขนาดแผ่น - ขนาดระเบียนควบคุม)}}{\text{ความยาวระเบียนข้อมูล + (ขนาดตำแหน่งระเบียนทั้งสอง * จำนวนชุดคีย์ที่ซ้ำกัน)}}$$

4. แผ่นแผนที่ข้อมูล (Data Map Page) ใช้สำหรับฟังก์ชันการค้นหาและอ่านระเบียนข้อมูล โดยการระบุหมายเลขระเบียน(Record Number) แผ่นแผนที่ข้อมูลจะกำหนดภายในแฟ้มข้อมูลดัชนี ต่อเมื่อมีการระบุการใช้แผ่นชนิดนี้ในระหว่างการสร้างแฟ้มข้อมูลดัชนี ภายในแผ่นแผนที่ข้อมูล ประกอบด้วย

4.1 ระเบียนควบคุมสำหรับแผ่นแผนที่ข้อมูล ประกอบด้วย

4.1.1 หมายเลขแผ่น

4.1.2 หมายเลขแผ่นแผนที่ข้อมูลก่อนหน้า

4.1.3 หมายเลขแผ่นแผนที่ข้อมูลถัดไป

4.2 แถวลำดับที่มีเนื้อหาของแต่ละสมาชิก ประกอบด้วย

4.2.1 หมายเลขแผ่นข้อมูล

4.2.2 จำนวนระเบียนข้อมูลที่มีในแผ่นข้อมูลนั้น

การกำหนดแฟ้มข้อมูลดัชนี ผู้ใช้สามารถกำหนดได้เองตามความต้องการ ทั้งชื่อแฟ้ม และ extension แต่สำหรับ extension ต้องไม่เป็น .BKU เพราะได้กำหนดให้เป็น extension ของแฟ้มสำรองแผ่น

2. <u>แฟ้มชุดการสลับลำดับตัวอักษร</u> (Alternate Character Sequence file)

เป็นแฟ้มข้อมูลที่ประกอบด้วยชุดการสลับตัวอักษร ซึ่งใช้ในกรณีที่ผู้ใช้ต้องการเปลี่ยนแปลงรหัสแอสกีตัวอักษรที่ใช้อยู่เดิมให้เป็นรหัสใหม่ก่อนทำการเปรียบเทียบค่าคีย์ เพื่อทำให้การจัดลำดับของคีย์เป็นไปตามลำดับที่กำหนดในรหัสใหม่ แฟ้มชุดการสลับลำดับตัวอักษรจะถูกนำมาใช้ ต่อเมื่อผู้ใช้ระบุคุณสมบัติของคีย์ใดคีย์หนึ่ง ให้มีการใช้ชุดการสลับลำดับตัวอักษรในระหว่างสร้างแฟ้มข้อมูลดัชนี ข้อมูลที่เป็นรหัสแอสกีซึ่งกำหนดขึ้นใหม่ในแฟ้มชุดการสลับลำดับ จะนำมาบันทึกไว้ในแฟ้มข้อมูลดัชนีเมื่อสร้างแฟ้มนั้นสำเร็จ ผู้ใช้สามารถ

กำหนดรูปแบบการจัดลำดับของคีย์ได้เองตามต้องการ โดยการกำหนดรหัสตัวอักษรขึ้นใหม่ และเก็บบันทึกไว้ในแฟ้มข้อมูลที่มีรูปแบบดังนี้

1. รหัสควบคุมซึ่งใช้ในการตรวจสอบ เพื่อระบุได้ว่าแฟ้มข้อมูลนี้เป็นแฟ้มชุดการสลับลำดับตัวอักษรรหัสควบคุมต้องเป็นเลขจำนวนเต็มขนาด 2 ไบต์ มีค่าเท่ากับ 00ACh โดยเริ่มจากตำแหน่งที่ 0 ของแฟ้ม

2. เนื้อที่สำหรับกำหนดชื่อชุดการสลับลำดับที่สร้างขึ้น มีขนาด 8 ไบต์ เริ่มจากตำแหน่งที่ 2 ของแฟ้ม

3. เนื้อที่ของรหัสแอสกีที่ผู้ใช้กำหนดขึ้น มีขนาด 256 ไบต์ เริ่มจากตำแหน่งที่ 10 ของแฟ้มเป็นต้นไป โดยที่รหัสซึ่งกำหนดขึ้นใหม่ จะต้องบันทึกไว้ในตำแหน่งที่ตรงกับรหัสแอสกีเดิม สำหรับรหัสที่ไม่มีการเปลี่ยนแปลง กำหนดให้บันทึกรหัสเดิมนั้นไว้ตรงตำแหน่งของรหัสนั้นเช่นกัน ตัวอย่างเช่น เมื่อต้องการเปลี่ยนรหัสแอสกีของตัวอักษรภาษาอังกฤษในช่วงของอักษรตัวเล็ก(lower case) มีรหัสแอสกีตั้งแต่ 61h - 7Ah ให้เป็นอักษรตัวใหญ่(upper case) ที่มีรหัสแอสกีตั้งแต่ 41h - 5Ah โดยการเปลี่ยนรหัสแอสกีภายในตำแหน่งที่ 61h จนถึง 7Ah ของเนื้อที่ส่วนนี้ให้มีค่ารหัสเป็น 41h ไปจนถึง 5Ah ตามลำดับ

```
Displacement ---------------- Hex codes ------------------    ASCII value
 0000(0000)  AC 00 55 50 50 45 52 20 20 20 00 01 02 03 04 05  ..UPPER   ......
 0016(0010)  06 07 08 09 0A 0B 0C 0D 0E 0F 10 11 12 13 14 15  ................
 0032(0020)  16 17 18 19 1A 1B 1C 1D 1E 1F 20 21 22 23 24 25  .......... !"#$%
 0048(0030)  26 27 28 29 2A 2B 2C 2D 2E 2F 30 31 32 33 34 35  &'()*+,-./012345
 0064(0040)  36 37 38 39 3A 3B 3C 3D 3E 3F 40 41 42 43 44 45  6789:;<=>?@ABCDE
 0080(0050)  46 47 48 49 4A 4B 4C 4D 4E 4F 50 51 52 53 54 55  FGHIJKLMNOPQRSTU
 0096(0060)  56 57 58 59 5A 5B 5C 5D 5E 5F 60 41 42 43 44 45  VWXYZ[\]^_`ABCDE
 0112(0070)  46 47 48 49 4A 4B 4C 4D 4E 4F 50 51 52 53 54 55  FGHIJKLMNOPQRSTU
 0128(0080)  56 57 58 59 5A 7B 7C 7D 7E 7F 80 81 82 83 84 85  VWXYZ{|}~.......
 0144(0090)  86 87 88 89 8A 8B 8C 8D 8E 8F 90 91 92 93 94 95  ................
 0160(00A0)  96 97 98 99 9A 9B 9C 9D 9E 9F A0 A1 A2 A3 A4 A5  ................
 0176(00B0)  A6 A7 A8 A9 AA AB AC AD AE AF B0 B1 B2 B3 B4 B5  ................
 0192(00C0)  B6 B7 B8 B9 BA BB BC BD BE BF C0 C1 C2 C3 C4 C5  ..............+..
 0208(00D0)  C6 C7 C8 C9 CA CB CC CD CE CF D0 D1 D2 D3 D4 D5  ................
 0224(00E0)  D6 D7 D8 D9 DA DB DC DD DE DF E0 E1 E2 E3 E4 E5  ................
 0240(00F0)  E6 E7 E8 E9 EA EB EC ED EE EF F0 F1 F2 F3 F4 F5  ................
 0256(0100)  F6 F7 F8 F9 FA FB FC FD FE FF 00 00 00 00 00 00  ................
 0272(0110)  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
```

ภาพที่ 4.4 ตัวอย่างแฟ้มชุดสลับลำดับตัวอักษรสำหรับเปลี่ยนตัวอักษรภาษาอังกฤษตัวเล็กให้เป็นตัวอักษรตัวใหญ่

สำหรับการกำหนดชื่อแฟ้มชุดการสลับลำดับตัวอักษร ผู้ใช้สามารถกำหนดได้เองตามต้องการทั้งชื่อแฟ้มและ extension

3. <u>แฟ้มสำรองแผ่น</u> (Page-Backup file)

เป็นแฟ้มข้อมูลที่ถูกสร้างขึ้นหลังจากการเปิดแฟ้มข้อมูลดัชนี เพื่อใช้ป้องกันการสูญหายของข้อมูลที่อาจเกิดขึ้นได้กับแฟ้มข้อมูลดัชนี ภายในแฟ้มประกอบด้วยข้อมูลเดิมของแผ่นต่าง ๆ ในแฟ้มข้อมูลดัชนี ก่อนที่ข้อมูลจะเปลี่ยนไปเมื่อมีการเพิ่มเติมหรือการลบหรือการแก้ไขค่าระเบียนข้อมูล เนื้อหาภายในแฟ้มสำรองแผ่นจะเปลี่ยนแปลงตลอดเวลาและจะบันทึกไว้เฉพาะข้อมูลเดิมของแผ่นที่ต้องเปลี่ยนไป เมื่อมีการเปลี่ยนแปลงข้อมูลภายในแฟ้มข้อมูลดัชนีครั้งหลังสุดเท่านั้น แฟ้มสำรองแผ่นจะนำมาใช้ต่อเมื่อการเปิดแฟ้มข้อมูลดัชนีในครั้งต่อไปถูกตรวจพบว่า ข้อมูลภายในแฟ้มอาจเกิดความเสียหาย เนื่องจากการทำงานของคอมพิวเตอร์หยุดกระทันหัน ได้แก่เกิดไฟฟ้าดับ เป็นต้น ข้อมูลเดิมจากแผ่นต่าง ๆ ในแฟ้มสำรองแผ่นจะถูกอ่านและนำไปบันทึกในตำแหน่งเดิมของแฟ้มข้อมูลดัชนี

การกำหนดชื่อแฟ้มสำรองแผ่น จะกระทำโดยฟังก์ชันเปิดแฟ้มข้อมูลดัชนี ชื่อแฟ้มจะเป็นชื่อเดียวกับแฟ้มข้อมูลดัชนี และมี extension เป็น .BKU

<u>บล๊อกสารสนเทศแฟ้มสำหรับแฟ้มข้อมูลดัชนี</u> (File Information Block)

เป็นบล๊อกข้อมูลที่รวบรวมสารสนเทศ ซึ่งควบคุมการใช้งานและการระบุตำแหน่งแผ่นชนิดต่าง ๆ เพื่อการเข้าถึงข้อมูลภายในแฟ้มข้อมูลดัชนี เนื้อที่ซึ่งใช้เป็นบล๊อกสารสนเทศแฟ้มสำหรับแฟ้มข้อมูลดัชนี จะต้องกำหนดขึ้นภายในเนื้อที่ของโปรแกรมประยุกต์ ดังนั้นบล๊อกข้อมูลนี้จึงเป็นพารามิเตอร์หนึ่งที่ต้องส่งผ่านไปทุกครั้งเพื่อให้ข้อมูลต่าง ๆ ที่เกิดจากการเรียกใช้ฟังก์ชันจัดการแฟ้มข้อมูลดัชนีส่งกลับมาบันทึกภายในบล๊อกข้อมูลนี้ ข้อมูลภายในบล๊อกสารสนเทศจะเริ่มต้นกำหนดค่า เมื่อมีการเปิดใช้แฟ้มข้อมูลดัชนีและค่าต่าง ๆ จะเปลี่ยนแปลงไปตามตำแหน่งอ้างอิงที่ใช้ในการเข้าถึงระเบียนข้อมูลต่าง ๆ ภายในแฟ้ม โดยที่บล๊อกสารสนเทศหนึ่งบล๊อก ใช้แทนการควบคุมการใช้แฟ้มข้อมูลดัชนีเพียงหนึ่งแฟ้มเช่นกัน จำนวนเนื้อที่ของบล๊อกสารสนเทศ จึงควรกำหนดให้เท่ากับจำนวนแฟ้มข้อมูลดัชนีที่ต้องการเปิดใช้พร้อมกัน

ในระหว่างการทำงาน เนื้อหาภายในบล๊อกสารสนเทศแฟ้มจะประกอบด้วยบล๊อกข้อมูล 2 บล๊อก คือ

1. File Control Block หรือ เอฟซีบี (FCB) เป็นบล๊อกข้อมูลที่เอมเอสดอสนำมาใช้ควบคุมการทำงานต่าง ๆ ที่เกี่ยวข้องกับแฟ้มข้อมูล ได้แก่ การเปิด การปิด การลบแฟ้มข้อมูลตลอดจนการอ่านและการบันทึกข้อมูล เนื้อหาภายในเอฟซีบีต้องเป็นไปตามรูปแบบที่กำหนดของเอมเอสดอส ประกอบด้วย

1.1 หมายเลขหน่วยจานบันทึก (Drive Number)

1.2 ชื่อแฟ้ม (File Name)

1.3 ชื่อ Extension

1.4 ตำแหน่งปัจจุบันของบล๊อกข้อมูล (Current Block)

1.5 ขนาดระเบียน (Record Size)

1.6 ขนาดแฟ้ม (File Size)

1.7 วันที่บันทึกข้อมูลครั้งสุดท้าย (Date of last write)

1.8 เวลาที่บันทึกข้อมูลครั้งสุดท้าย (Time of last write)

1.9 เนื้อที่สำรองของเอมเอสดอส

1.10 ตำแหน่งปัจจุบันของระเบียน (Current Record)

1.11 ตำแหน่งสัมพัทธ์ของระเบียน (Relative Record)

2. บล๊อกข้อมูลบอกตำแหน่ง เป็นบล๊อกข้อมูลที่ประกอบด้วยข้อมูลที่อ่านมาจากแผ่นระเบียนควบคุม และข้อมูลที่ระบุตำแหน่งปัจจุบัน ซึ่งมาจากการเข้าถึงระเบียนข้อมูลครั้งล่าสุด ภายในบล๊อกข้อมูลบอกตำแหน่ง ประกอบด้วย

2.1 ข้อมูลจากบล๊อกสารสนเทศควบคุมของแผ่นระเบียนควบคุม

2.2 ข้อมูลจากบล๊อกสารสนเทศของชุดคีย์ เฉพาะระเบียนควบคุมของชุดคีย์ที่ใช้งานในขณะนั้น

2.3 ตำแหน่งระเบียนข้อมูล ประกอบด้วย หมายเลขแผ่นข้อมูล และ ลำดับที่ระเบียนข้อมูลในแผ่น

2.4 ตำแหน่งคีย์ที่ค้นพบครั้งหลังสุด ประกอบด้วย หมายเลขแผ่นดัชนี และ ลำดับที่ดัชนีในแผ่น

2.5 หมายเลขชุดคีย์ที่ใช้ในขณะนั้น

2.6 ตำแหน่งแผนที่ข้อมูลที่ใช้ (สำหรับในกรณีที่ทำการค้นหาและอ่านระเบียบโดยการอ้างถึงหมายเลขระเบียนข้อมูล) ประกอบด้วย หมายเลขแผ่นแผนที่ข้อมูล และลำดับที่สมาชิกในแผ่น

2.7 หมายเลขระเบียนข้อมูลที่อ้างถึง

<u>การนำเนื้อที่ว่างในแฟ้มข้อมูลกลับมาใช้</u>

เนื้อที่ว่างภายในแฟ้มข้อมูล เกิดขึ้นจากการลบระเบียนข้อมูลออกจากแฟ้มข้อมูล ซึ่งทำให้เกิดเนื้อที่ว่างของระเบียนข้อมูล และที่ว่างของแผ่นดัชนีที่เกิดจากคีย์ของระเบียนข้อมูลนั้นต้องถูกลบออกไปจนทำให้ไม่มีสมาชิกเหลืออยู่ในแผ่นดัชนี เมื่อต้องลบระเบียนข้อมูลออกเป็นจำนวนมาก จะทำให้เกิดเนื้อที่ว่างกระจายทั่วไปในแฟ้มข้อมูลดัชนี การนำเนื้อที่ว่างเหล่านั้นกลับมาใช้สำหรับการเพิ่มเติมระเบียนข้อมูลในครั้งต่อไป จึงขึ้นอยู่กับ

1. วิธีการที่ทำให้ทราบว่า มีที่ว่างเกิดขึ้นภายในแฟ้มข้อมูล

2. การออกแบบโครงสร้างข้อมูล ที่สามารถทำให้ที่ว่างที่เกิดขึ้นอยู่ในลักษณะที่สะดวกต่อการนำกลับมาใช้

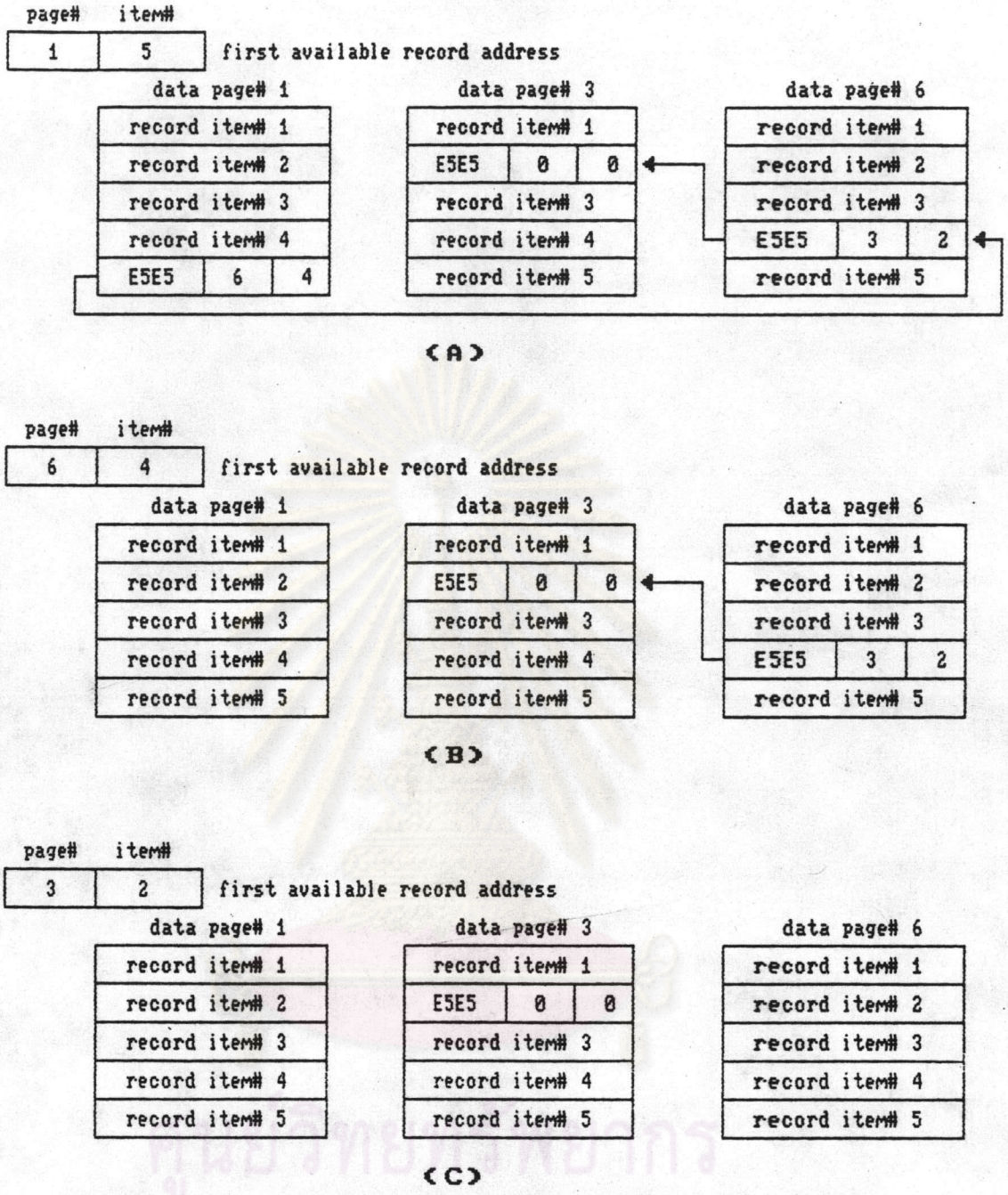
สำหรับแฟ้มข้อมูลดัชนีได้ออกแบบให้สามารถนำที่ว่างกลับมาใช้ได้อีก โดยแบ่งกลุ่มที่ว่างออกเป็น ที่ว่างระเบียนข้อมูล และที่ว่างของแผ่นดัชนี ลักษณะโครงสร้างข้อมูลที่นำมาใช้กับที่ว่างทั้งสองกลุ่มนี้ จะเป็นโครงสร้างข้อมูลลิงค์ลิสต์ทางเดียว(Single Linked List structure) ซึ่งเป็นการเชื่อมต่อที่ว่างหรือสมาชิกต่าง ๆ ภายในลิสต์นั้นโดยการใช้พอยต์เตอร์บอกตำแหน่งสมาชิกลำดับถัดไป เนื้อหาภายในที่ว่างจะประกอบด้วย

1. รหัสควบคุมที่ระบุว่าเป็นที่ว่าง เป็นรหัสของเลขจำนวนเต็มขนาด 2 ไบต์ และมีค่าเท่ากับ E5E5h

2. พอยต์เตอร์ที่บอกตำแหน่งสมาชิกลำดับถัดไป

2.1 สำหรับที่ว่างของแผ่นดัชนี พอยต์เตอร์ที่ใช้จะเป็นหมายเลขแผ่น

2.2 สำหรับที่ว่างของระเบียนข้อมูล พอยต์เตอร์ที่ใช้ จะประกอบด้วยหมายเลขแผ่นข้อมูล และ ลำดับที่ระเบียนข้อมูลที่ถูกลบในแผ่น



ภาพที่ 4.5 ลักษณะการเชื่อมต่อเนื้อที่ว่างในแฟ้มข้อมูลเพื่อนำกลับมาใช้ได้อีก

(A) เริ่มต้นจากเนื้อที่ว่างในแผ่นข้อมูลที่ 1 ลำดับสมาชิกที่ 5

(B) และ (C) การนำเนื้อที่ว่างมาใช้เมื่อมีการเพิ่มระเบียนข้อมูล

การควบคุมการใช้ที่ว่างภายในลิสต์ ได้กำหนดให้มีการทำงานในลักษณะการใช้สแตก(stack) กล่าวคือ เมื่อมีที่ว่างใหม่ซึ่งเกิดจากการลบระเบียนข้อมูล ที่ว่างนั้นจะถูกนำมารวมอยู่ในกลุ่มของที่ว่างโดยการพุช (push) สมาชิกลงส่วนบนสแตก (top of stack) และเมื่อต้องการนำที่ว่างกลับไปใช้ สามารถนำที่ว่างออกจากการเชื่อมต่อกันโดยการป๊อป (pop) สมาชิกออกจากส่วนบนสแตก ซึ่งจะต้องมีเนื้อที่หรือตัวแปรหนึ่งเป็นส่วนที่ใช้ระบุว่ายังมีสมาชิกของที่ว่างอยู่ในสแตกหรือไม่ โดยการบันทึกพอยต์เตอร์ที่บอกตำแหน่งสมาชิกส่วนบนสแตกไว้ตลอดเวลา ที่มีการเปลี่ยนแปลงในสแตก เมื่อใดที่ค่าพอยต์เตอร์ในตัวแปรนั้นเป็นนัลพอยต์เตอร์(Null pointer) นั่นแสดงว่าไม่มีการระบุตำแหน่งของสมาชิกตัวต่อไปในสแตก ซึ่งหมายความว่าไม่มีที่ว่างเหลือสำหรับนำกลับมาใช้ในการเพิ่มเติมระเบียน การบันทึกระเบียนข้อมูลใหม่ในครั้งต่อไป จึงต้องบันทึกในเนื้อที่ที่กำหนดขึ้นใหม่ภายในแฟ้มข้อมูล สำหรับข้อมูลพอยต์เตอร์ที่ใช้ระบุว่า เป็นนัลพอยต์เตอร์ ได้กำหนดให้ค่าพอยต์เตอร์นั้นมีค่าเท่ากับ 0

<u>การเปรียบเทียบข้อมูลภาษาไทย</u>

ในการเรียงลำดับข้อมูลชนิดตัวอักษร จะต้องใช้การเปรียบเทียบค่าข้อมูล เพื่อให้สามารถจัดลำดับข้อมูลได้ตรงตามต้องการ กล่าวคือ ต้องเปรียบเทียบรหัสแอสกีของตัวอักษรทุกตัวภายในข้อมูล สำหรับการเรียงลำดับข้อมูลจากที่มีค่าน้อยไปถึงค่ามากที่สุด ตัวอักษรตัวแรกของข้อมูลถือว่า เป็นตัวอักษรที่ให้น้ำหนักในการจัดเรียงมากที่สุด และลดหลั่นลงมาตามลำดับ ตัวอักษรที่มีค่ารหัสแอสกีน้อยจัดว่ามีน้ำหนักมาก ก็คือทำให้ข้อมูลนั้นถูกจัดอันดับไว้ก่อนข้อมูลอื่น การเปรียบเทียบในลักษณะนี้จะใช้กันโดยทั่วไปกับการเปรียบเทียบข้อมูลตัวอักษรภาษาอังกฤษเท่านั้น

แต่สำหรับข้อมูลตัวอักษรที่เป็นภาษาไทย ซึ่งกำหนดรหัสภาษาไทยมาจากมหาวิทยาลัยเกษตรศาสตร์ หรือ สำนักงานมาตราฐานอุตสาหกรรม จะทำการเปรียบเทียบในลักษณะเดียวกับตัวอักษรภาษาอังกฤษ อาจทำให้ลำดับข้อมูลที่ได้ผิดพลาดไป ทั้งนี้มีสาเหตุมาจาก

1. สระที่นำหน้าพยัญชนะ ได้แก่ เ- แ- โ- ใ- ไ- สระเหล่านี้ได้ถูกกำหนดให้มีรหัสแอสกีที่มากกว่าตัวพยัญชนะ ทำให้คำที่มีสระเหล่านี้นำหน้ามีน้ำหนักลดลงลำดับของคำจึงถูกจัดไว้อันดับท้าย เช่น "ดิเรก" จะมีลำดับหลัง "ดิศักดิ์"

รหัสแอสกี	ตัวอักษร	รหัสแอสกี	ตัวอักษร	รหัสแอสกี	ตัวอักษร	รหัสแอสกี	ตัวอักษร
161	ก	179	ฌ	197	ล	215	◌ั
162	ข	180	ด	198	ฦ	216	◌ั
163	ฃ	181	ต	199	ว	217	เ
164	ค	182	ถ	200	ศ	218	แ
165	ฅ	183	ท	201	ษ	219	โ
166	ฆ	184	ธ	202	ส	220	ใ
167	ง	185	น	203	ห	221	ไ
168	จ	186	บ	204	ฬ	222	◌า
168	ฉ	187	ป	205	อ	223	◌ๆ
170	ช	188	ผ	206	ฮ	224	ฯ
171	ซ	189	ฝ	207	◌ะ	225	◌◌
172	ฌ	190	พ	208	◌◌	226	◌◌
173	ญ	191	ฟ	209	◌า	227	◌◌
174	ฏ	192	ภ	210	◌ำ	228	◌◌
175	ฏ	193	ม	211	◌◌	229	◌◌
176	ฐ	194	ย	212	◌◌	230	◌◌
177	ฑ	195	ร	213	◌◌		
178	ฒ	196	ฤ	214	◌◌		

ตารางที่ 4.1 รหัสที่กำหนดสำหรับการจัดเรียงภาษาไทย

```
Displacement ---------- Hex codes ----------   ASCII value
 0000(0000)  AC 00 4B 55 26 55 50 20 20 20 00 01 02 03 04 05   ..KU&UP  ......
 0016(0010)  06 07 08 09 0A 0B 0C 0D 0E 0F 10 11 12 13 14 15   ................
 0032(0020)  16 17 18 19 1A 1B 1C 1D 1E 1F 20 21 22 23 24 25   .......... !"#$%
 0048(0030)  26 27 28 29 2A 2B 2C 2D 2E 2F 30 31 32 33 34 35   &'()*+,-./012345
 0064(0040)  36 37 38 39 3A 3B 3C 3D 3E 3F 40 41 42 43 44 45   6789:;<=>?@ABCDE
 0080(0050)  46 47 48 49 4A 4B 4C 4D 4E 4F 50 51 52 53 54 55   FGHIJKLMNOPQRSTU
 0096(0060)  56 57 58 59 5A 5B 5C 5D 5E 5F 60 41 42 43 44 45   VWXYZ[\]^_`ABCDE
 0112(0070)  46 47 48 49 4A 4B 4C 4D 4E 4F 50 51 52 53 54 55   FGHIJKLMNOPQRSTU
 0128(0080)  56 57 58 59 5A 7B 7C 7D 7E 7F 80 81 82 83 84 85   VWXYZ{|}~.......
 0144(0090)  86 87 88 89 8A 8B 8C 8D 8E 8F 90 91 92 93 94 95   ................
 0160(00A0)  96 97 98 99 9A 9B 9C 9D 9E 9F A0 A1 A2 A4 A6 A7   ................
 0176(00B0)  A8 A9 AA AB AC AD AE AF B0 B1 B2 B3 B4 B5 B6 B7   ................
 0192(00C0)  B8 B9 BA BB BC BD BE BF C0 C1 C2 C3 C4 C5 C7 C8   ................
 0208(00D0)  C9 CA CB CC CD C5 CF 20 D1 D2 D9 DA DB DC DD DF   ....... ........
 0224(00E0)  E0 D7 D8 D3 D4 D5 D6 D0 E7 E5 E1 E2 E3 E4 E6 20   ...............
 0240(00F0)  FF FF FF FF EA EB EC ED EE EF F0 F1 F2 F3 F4 F5   ................
 0256(0100)  F6 F7 F8 F9 FA FB FC FD FE FF 00 00 00 00 00 00   ................
 0272(0110)  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
```

ภาพที่ 4.6 เนื้อหาภายในแฟ้ม KU&UP.ALT

```
Displacement ---------- Hex codes ----------   ASCII value
 0000(0000)  AC 00 54 49 53 26 55 50 20 20 00 01 02 03 04 05   ..TIS&UP  ......
 0016(0010)  06 07 08 09 0A 0B 0C 0D 0E 0F 10 11 12 13 14 15   ................
 0032(0020)  16 17 18 19 1A 1B 1C 1D 1E 1F 20 21 22 23 24 25   .......... !"#$%
 0048(0030)  26 27 28 29 2A 2B 2C 2D 2E 2F 30 31 32 33 34 35   &'()*+,-./012345
 0064(0040)  36 37 38 39 3A 3B 3C 3D 3E 3F 40 41 42 43 44 45   6789:;<=>?@ABCDE
 0080(0050)  46 47 48 49 4A 4B 4C 4D 4E 4F 50 51 52 53 54 55   FGHIJKLMNOPQRSTU
 0096(0060)  56 57 58 59 5A 5B 5C 5D 5E 5F 60 41 42 43 44 45   VWXYZ[\]^_`ABCDE
 0112(0070)  46 47 48 49 4A 4B 4C 4D 4E 4F 50 51 52 53 54 55   FGHIJKLMNOPQRSTU
 0128(0080)  56 57 58 59 5A 7B 7C 7D 7E 7F 80 81 82 83 84 85   VWXYZ{|}~.......
 0144(0090)  86 87 88 89 8A 8B 8C 8D 8E 8F 90 91 92 93 94 95   ................
 0160(00A0)  96 97 98 99 9A 9B 9C 9D 9E 9F A0 A1 A2 A3 A4 A5   ................
 0176(00B0)  A6 A7 A8 A9 AA AB AC AD AE AF B0 B1 B2 B3 B4 B5   ................
 0192(00C0)  B6 B7 B8 B9 BA BB BC BD BE BF C0 C1 C2 C3 C4 C5   ................
 0208(00D0)  C6 C7 C8 C9 CA CB CC CD CE E0 CF D0 D1 D2 D3 D4   ................
 0224(00E0)  D5 D6 D7 D8 E7 DB DC DD DE 20 D9 DA DB DC DD DE   ......... ......
 0240(00F0)  DF E5 E1 E2 E3 E4 EC E6 EE EF F0 F1 F2 F3 F4 F5   ................
 0256(0100)  F6 F7 F8 F9 FA FB FC FD FE FF 00 00 00 00 00 00   ................
 0272(0110)  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
```

ภาพที่ 4.7 เนื้อหาภายในแฟ้ม TIS&UP.ALT

2. รหัสแอสกีของไม้หันอากาศ (ั) ที่กำหนดจากมหาวิทยาลัยเกษตรศาสตร์ ถูกกำหนดให้มีค่ามากกว่าสระอิ ทำให้เกิดความผิดพลาดกับการเรียงลำดับด้วย

3. วรรณยุกต์ ได้แก่ ่ ้ ๊ ๋ เมื่อแทรกอยู่ในคำใดจะทำให้คำเหล่านั้นมีน้ำหนักลดลงเหมือนกัน เช่น "ชุ่ม" จะมีลำดับหลัง "ชุลีพร"

เพื่อให้การเปรียบเทียบข้อมูลตัวอักษรภาษาไทยทำได้ในลักษณะเดียวกับข้อมูลอักษรภาษาอังกฤษ จึงต้องมีการจัดเรียงน้ำหนักตัวอักษรในคำขึ้นใหม่ก่อนที่จะทำการเปรียบเทียบตัวอักษรแต่ละตัว โดยอาศัยสาเหตุที่เกิดขึ้นมากำหนดวิธีการจัดเรียงน้ำหนักตัวอักษร ดังนี้

1. สำหรับคำที่มีสระนำหน้าพยัญชนะ ให้ทำการสลับที่ระหว่างสระดังกล่าวกับพยัญชนะนั้น เพื่อเป็นการลดน้ำหนักของสระที่นำหน้าพยัญชนะเหล่านั้น

2. สำหรับวรรณยุกต์ให้ทำการสลับที่ระหว่างวรรณยุกต์กับตัวอักษรถัดไปเช่นเดียวกัน เพื่อลดน้ำหนักของวรรณยุกต์นั้น

3. สำหรับไม้หันอากาศ(ั) ในรหัสภาษาไทยเกษตร จะต้องเปลี่ยนรหัสแอสกีที่ใช้ในขณะนั้นชั่วคราวให้มีค่าอยู่ระหว่าง "ะ" และ "า" เพื่อให้มีน้ำหนักที่ถูกต้อง

ในการกำหนดคุณสมบัติให้กับคีย์ เมื่อคีย์ชุดใดถูกกำหนดคุณสมบัติของคีย์เป็นตัวอักษรไทย คีย์ชุดนั้นจะต้องกำหนดคุณสมบัติการใช้ชุดการสลับลำดับตัวอักษรด้วย และชุดการสลับลำดับนั้นจะต้องมีลำดับตำแหน่งของตัวอักษรไทยที่ถูกต้อง ตรงตามน้ำหนักของตัวอักษรที่ใช้ น้ำหนักตัวอักษรไทยที่กำหนดขึ้นเริ่มต้นจากตัวอักษรพยัญชนะที่เรียงลำดับจาก "ก" ถึง "ฮ" และตัวอักษรสระจาก ะ ั า ำ ิ ี ึ ื ุ ู เ- แ- โ- ใ- ไ- และตัวอักษรวรรณยุกต์ ่ ้ ๊ ๋ ในงานวิจัยได้เตรียมชุดการสลับลำดับอักษรเพื่อใช้กับตัวอักษรไทยที่กำหนดรหัสมาจากมหาวิทยาลัยเกษตรศาสตร์ และสำนักงานมาตราฐานอุตสาหกรรม ภายในแฟ้มชุดการสลับลำดับตัวอักษรชื่อ KU&UP.ALT และ TIS&UP.ALT ตามลำดับ

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

<u>การออกแบบบัฟเฟอร์สำหรับช่องใส่แผ่น</u>

บัฟเฟอร์สำหรับช่องใส่แผ่น เป็นบัฟเฟอร์ที่กำหนดขึ้นสำหรับบรรจุข้อมูลจากจานบันทึกไว้ใช้งานให้นานที่สุดเพื่อลดเวลาที่ต้องเสียไปกับการอ่านจากจานบันทึก มีการกำหนดเนื้อที่หน่วยความจำให้กับบัฟเฟอร์ในระหว่างการติดตั้งโปรแกรมให้ประจำในหน่วยความจำ ข้อมูล

ที่อ่านมาจากจานบันทึกกำหนดให้มีขนาดที่เท่ากัน หรือ เป็นแผ่น จำนวนช่องใส่แผ่นจะขึ้นกับการกำหนดขนาดบัฟเฟอร์และขนาดช่องใส่แผ่น ซึ่งทำให้แบ่งเนื้อที่ออกเป็นช่องใส่แผ่นที่มีขนาดเท่า ๆ กัน และไม่สามารถระบุจำนวนช่องใส่แผ่นได้อย่างแน่นอน การเข้าถึงข้อมูลในช่องใส่แผ่นต่าง ๆ กระทำได้ จากการคำนวณระยะห่างของช่องใส่แผ่นนั้น กับตำแหน่งเริ่มต้นของบัฟเฟอร์

```
PagePtr = Pgmem + OfsPgmem
```

โดย PagePtr แทน ตำแหน่งเริ่มต้นของช่องใส่แผ่นที่ต้องการเข้าถึง และ Pgmem แทนตำแหน่งเริ่มต้นของบัฟเฟอร์สำหรับช่องใส่แผ่น ส่วน OfsPgmem เป็นระยะห่างจากตำแหน่งเริ่มต้นของบัฟเฟอร์ถึงช่องใส่แผ่นที่ต้องการ ซึ่งมาจากผลคูณของขนาดช่องใส่แผ่น กับหมายเลขช่องใส่แผ่น (Page frame number) โดยเริ่มต้นจากช่องใส่แผ่นที่ 0 เป็นต้นไป

การทำงานที่เกี่ยวข้องกับช่องใส่แผ่น เมื่อมีความต้องการอ่านข้อมูลจากแฟ้มข้อมูล

1. ทำการค้นหาข้อมูลจากช่องใส่แผ่นทุกช่องภายในบัฟเฟอร์ก่อน โดยค้นหาหมายเลขของแผ่นซึ่งบรรจุข้อมูลนั้น เมื่อพบแผ่นที่ต้องการก็สามารถนำไปใช้งานได้ทันที
2. เมื่อค้นหาแผ่นที่ต้องการไม่พบ จึงจำเป็นต้องอ่านข้อมูลของแผ่นนั้นจากแฟ้มข้อมูล โดยจะต้องเตรียมช่องใส่แผ่นที่ว่างสำหรับบรรจุข้อมูลที่ส่งผ่านมา
3. ถ้าไม่พบแผ่นซึ่งบรรจุข้อมูลที่ต้องการในบัฟเฟอร์ และไม่มีช่องใส่แผ่นที่ว่างสำหรับบรรจุข้อมูล ทำให้เกิดภาวะการขาดแคลนช่องใส่แผ่น จึงต้องอาศัยใช้ช่องใส่แผ่นที่ไม่จำเป็นต้องใช้งานอีกต่อไป หรือ เป็นช่องใส่แผ่นที่ไม่ได้ถูกอ้างถึงเป็นเวลานาน เพื่อบรรจุข้อมูลลงแทนที่ข้อมูลเดิม

ในการค้นหาและควบคุมการใช้ช่องใส่แผ่น ตลอดจนการจัดลำดับช่องใส่แผ่นที่ว่าง ซึ่งต้องถูกแทนที่เมื่อเกิดภาวะขาดแคลนช่องใส่แผ่น ได้ออกแบบลำดับการใช้ช่องใส่แผ่นอยู่ในลักษณะของแถวคอย (Queue) ภายในแถวคอยประกอบด้วยสมาชิกซึ่งมีจำนวนเท่ากับจำนวนช่องใส่แผ่น และมีข้อมูลที่ใช้เข้าถึงช่องใส่แผ่นเหล่านั้นเพื่อเป็นตัวแทนในการจัดลำดับ การแทนที่โดยกำหนดให้เป็นไปตามหลักการแทนที่แบบไม่ได้ใช้นานที่สุด (least recently used) เมื่อมีการเข้าถึงข้อมูลภายในช่องใส่แผ่นใด สมาชิกในแถวคอยของช่องใส่แผ่นนั้นจะถูกจัดให้

อยู่ในลำดับสุดท้ายของแถวคอยเสมอ นั่นคือช่องใส่แผ่นของสมาชิกที่อยู่ในลำดับแรกของแถวคอยจะเป็นช่องใส่แผ่นที่มีโอกาสถูกแทนที่มากที่สุด แต่เนื่องจากสมาชิกในแถวคอยจะต้องถูกเปลี่ยนแปลงลำดับอยู่เสมอเมื่อถูกอ้างถึง เพื่อให้เกิดความสะดวกต่อการย้ายลำดับสมาชิกในแถวคอยการเชื่อมต่อระหว่างสมาชิกต่าง ๆ จึงได้กำหนดให้มีโครงสร้างเป็นแบบลิงค์ลิสต์ 2 ทาง (double linked list structure) เนื้อหาภายในสมาชิกจะประกอบด้วย

1. หมายเลขแฟ้มข้อมูลเพื่อใช้ระบุว่า ข้อมูลในช่องใส่แผ่นเป็นของแฟ้มข้อมูลใด
2. หมายเลขแผ่นภายในแฟ้มข้อมูล
3. Ofspgmem เพื่อใช้ระบุตำแหน่งช่องใส่แผ่น
4. แฟลกที่แสดงว่ามีการแก้ไขข้อมูลภายในช่องใส่แผ่น
5. ตำแหน่งสมาชิกในลำดับก่อนหน้า
6. ลำดับสมาชิกในลำดับถัดไป

เนื้อที่แถวคอย จะเป็นเนื้อที่อีกส่วนหนึ่งซึ่งไม่เกี่ยวข้องกับบัฟเฟอร์สำหรับช่องใส่แผ่น จำนวนสมาชิกในแถวคอย แต่ขึ้นกับจำนวนช่องใส่แผ่นด้วยเช่นกัน จึงทำให้การเข้าถึงสมาชิกต่าง ๆ ทำในลักษณะเดียวกับช่องใส่แผ่นในบัฟเฟอร์ กล่าวคือ ทำการอ้างอิงจากตำแหน่งเริ่มต้นของแถวคอยด้วยระยะห่างจากตำแหน่งของสมาชิกเหล่านั้น

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย