



หลักเกณฑ์ทั่วไปในการจัดการข้อมูล

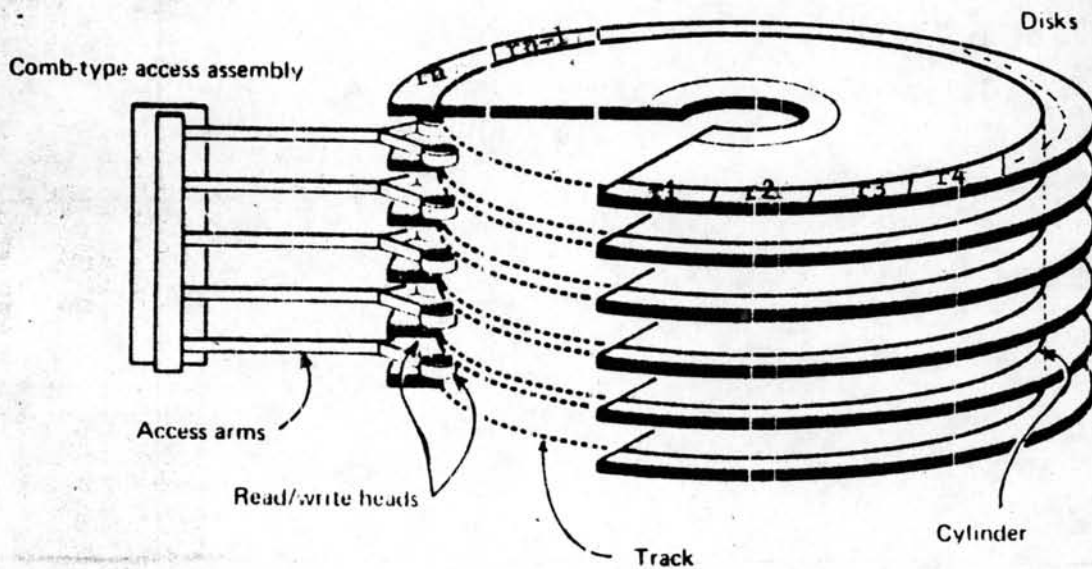
การจัดเก็บข้อมูลในยุคแรกของการนำคอมพิวเตอร์มาใช้ จะเก็บในรูปของบัตรคอมพิวเตอร์ เนื่องจากว่าบัตรคอมพิวเตอร์นี้มีอัตราการถ่ายถอดข้อมูลต่ำ โดยปกติแล้วประมาณ ๑๐๐๐ บัตรต่อนาที ทำให้การทำงานเป็นไปได้ช้า เพราะต้องเสียเวลารอคอยในการอ่านข้อมูลเพื่อที่จะนำไปประมวลผล อีกประการหนึ่งการเก็บข้อมูลในรูปของบัตรคอมพิวเตอร์เป็นแบบเรียงลำดับ (Sequential Organization) จึงไม่สามารถที่จะอ่านข้อมูลที่ต้องการได้โดยตรงทันที ต่อมาได้มีการพัฒนาเทปแม่เหล็กเพื่อใช้เป็นสื่อบันทึกข้อมูล ซึ่งมีอัตราในการถ่ายถอดข้อมูลสูงกว่าบัตรคอมพิวเตอร์มาก โดยปกติประมาณ ๒๐๐,๐๐๐ ตัวอักษรต่อวินาที แต่การจัดเก็บข้อมูลในเทปแม่เหล็กยังคงเป็นแบบเรียงลำดับเหมือนบัตรคอมพิวเตอร์ จึงมีการพัฒนาสื่อบันทึกข้อมูลขึ้นอีกเรื่อย ๆ จนสามารถจัดเก็บข้อมูลในสื่อบันทึกข้อมูลแบบเข้าถึงข้อมูลได้โดยตรง (Direct access storage) ทำให้สามารถนำข้อมูลมาใช้ได้เร็วขึ้น ไม่ต้องเสียเวลาในการอ่านข้อมูลเรียงลำดับกันไป ตัวอย่างสื่อบันทึกข้อมูลประเภทนี้ได้แก่ แผ่นจานแม่เหล็ก (Disk) ทรัม (Drum) และอื่น ๆ การจัดข้อมูลในแผ่นจานแม่เหล็กจะสามารถจัดข้อมูลได้หลายแบบมากขึ้น คือจัดได้ทั้งแบบเรียงลำดับ เช่นเดียวกับบัตรและ เทปแม่เหล็ก หรือจัดแบบเข้าถึงข้อมูลได้โดยตรง สำหรับการวิจัยครั้งนี้จะเน้นเฉพาะการจัดเก็บข้อมูลในลักษณะต่าง ๆ ของจานแม่เหล็ก (รายละเอียดของแผ่นจานแม่เหล็ก ดูในภาคผนวก ก) ซึ่งสามารถแยกเป็น ๒ ประเภท^(๑๔) คือ

- การจัดข้อมูลแบบเรียงลำดับ (Sequential Organization)
- การจัดข้อมูลแบบเข้าถึงข้อมูลได้โดยตรง (Direct Organization)



๒.๑ การจัดข้อมูลแบบเรียงลำดับ

๒.๑.๑ ลักษณะ การจัดข้อมูลชนิดนี้เหมือนกับการจัดข้อมูลในบัตรคอมพิวเตอร์และเทปแม่เหล็ก คือระเบียบข้อมูล (record) จะถูกบันทึก เรียงตามลำดับก่อนหลังของระเบียบข้อมูลที่เข้ามา โดยระเบียบข้อมูลอาจถูกจัดเรียงลำดับ (Sort) หรือไม่เรียงลำดับ (Unsort) ก็ได้ ในแฟ้มข้อมูลแบบนี้ระเบียบข้อมูลอาจมีลักษณะความยาวคงที่ (fix length) หรือมีความยาวไม่คงที่ (Variable length) ก็ได้ และมีลักษณะการจัดข้อมูลทางกายภาพ (Physical record) เป็นแบบบล็อก (block) หรือไม่เป็นบล็อก (Unblock) ก็ได้ (๑๒)



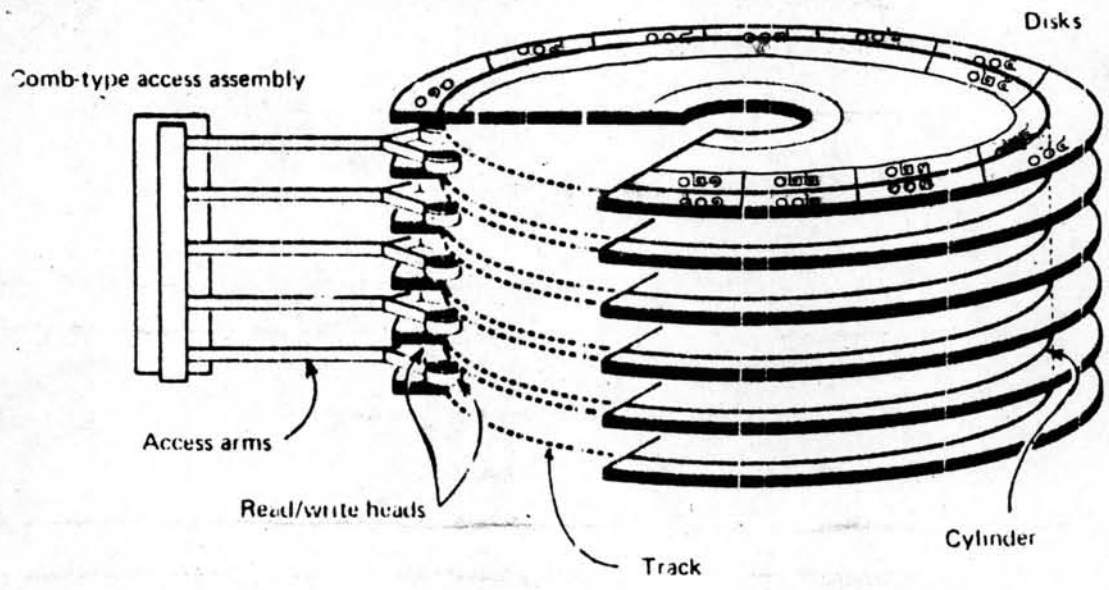
รูปที่ ๒.๑ แสดงลักษณะการจัดข้อมูลแบบ เรียงลำดับ

๒.๑.๒ การสร้าง (Create)

ถ้ามีระเบียบข้อมูล อยู่จำนวนหนึ่งและต้องการสร้างแฟ้มข้อมูล แบบเรียงลำดับ จะมีขั้นตอนการสร้างดังนี้

ขั้นแรกผู้ใช้จะต้องประมาณเนื้อที่ที่จะใช้ในแผ่นจานแม่เหล็กก่อน ซึ่งพิจารณาได้จากความยาวของระเบียบข้อมูล จำนวนระเบียบข้อมูลทั้งหมด และลักษณะการจัดระเบียบข้อมูลว่าเป็นแบบบล็อกหรือไม่ เป็นบล็อก

ขั้นที่สอง บันทึกระเบียบข้อมูลลงในเนื้อที่ดังกล่าว โดยบันทึกระเบียบข้อมูลแรกลงในแทรค (Track) ใด แแทรคหนึ่ง ซึ่งกำหนดไว้ให้เป็นจุดเริ่มต้น และบันทึกระเบียบข้อมูลลำดับต่อมาเรียงต่อกันไปเรื่อย ๆ จนเต็มแทรคแล้ว จึงเลื่อนไปบันทึกระเบียบข้อมูลลงในแทรคต่อไปของไซลินเดอร์ (Cylinder) เดียวกัน เมื่อไซลินเดอร์หนึ่ง ๆ เต็มแล้วจึงเลื่อนไปบันทึกลงใน ไซลินเดอร์ถัดไปเรื่อย ๆ โดยเริ่มต้นตั้งแต่แทรคแรกของไซลินเดอร์นั้น ๆ การบันทึกระเบียบข้อมูลจะทำได้ในลักษณะเช่นนี้ไปจนกว่าจะหมดระเบียบข้อมูลที่ต้องการบันทึก



รูปที่ ๒.๒ แสดงการจัดข้อมูลแบบเรียงลำดับ ในแผ่นจานแม่เหล็ก

จากตัวอย่างในรูปที่ ๒.๒ ถ้าต้องการบันทึกทะเบียนข้อมูล ๒๕ รายการ ลงในแผ่นจานแม่เหล็ก โดยสมมติให้แต่ละแทรคสามารถบันทึกได้ ๑๐ ระเบียนข้อมูล พื้นผิวด้านบนของแผ่นแรก ซึ่งเป็นแทรคแรก ของไฮลินเดอร์ จะบันทึกทะเบียนข้อมูลหมายเลข ๐๐๑-๐๑๐ เมื่อแทรคแรกเต็มก็จะบันทึกลงในแทรค ที่สอง ซึ่งอยู่บนพื้นผิวด้านล่างจนเต็ม แล้วจึงค่อยขยับทิกในแทรคที่สาม บนพื้นผิวด้านบนของแผ่นที่สอง ซึ่งอยู่ใน ไฮลินเดอร์เดียวกันจนเต็มอีก วนเวียนเช่นนี้ไปเรื่อย ๆ จนกระทั่งหมดทะเบียนข้อมูลที่ต้องการ บันทึก

๒.๑.๓ การค้นหาข้อมูล (Search)

ในการค้นหาข้อมูลที่ต้องการจะต้อง เริ่มต้นแฟ้มข้อมูลทุกครั้ง ไม่ว่าจะทะเบียนข้อมูล ที่ต้องการจะอยู่ที่ใด ๆ ของแฟ้มข้อมูลก็ตาม โดยทะเบียนข้อมูลจะถูกอ่าน เรียงตามลำดับจนกระทั่ง พบทะเบียนข้อมูลที่ต้องการ การค้นหาทะเบียนข้อมูลที่ต้องการของการจัดแฟ้มข้อมูลแบบนี้จะทำได้ เร็ว หรือช้าขึ้นอยู่กับลักษณะของการจัดลำดับของการบันทึกทะเบียนข้อมูลในแฟ้มข้อมูลว่า เป็นแบบ เรียงลำดับ หรือไม่เรียงลำดับ ซึ่งจะได้อธิบายถึงการค้นหาทะเบียนข้อมูลที่ต้องการของแต่ละลักษณะการจัดแฟ้ม ข้อมูล ดังนี้

๒.๑.๓.๑ ข้อมูลไม่เรียงลำดับ (Unsorted record file) ในการค้นหา

ข้อมูลที่ต้องการในแฟ้มข้อมูลแบบไม่เรียงลำดับนี้ จะต้องเริ่มอ่านทะเบียนข้อมูลเพื่อค้นหาตั้งแต่ทะเบียน ข้อมูลแรกของแฟ้มข้อมูลไปเรื่อย ๆ จนกว่าจะพบข้อมูลที่ต้องการ ซึ่งจำนวนครั้งในการค้นหาขึ้นอยู่กับ ตำแหน่งของทะเบียนข้อมูลที่ต้องการว่าอยู่ในลำดับที่เท่าใดในแฟ้มข้อมูล ถ้าหากว่าข้อมูลที่ต้องการ ค้นหาไม่อยู่ในแฟ้มข้อมูลนี้ การค้นหาต้องทำไปจนหมดแฟ้มข้อมูล จึงจะสามารถแน่ใจได้ว่าทะเบียน ข้อมูลที่ต้องการนั้นไม่มี

๒.๑.๓.๒ ข้อมูลเรียงลำดับ (Sorted record file) ซึ่งแบ่งได้ ๒ กรณีคือ

- เรียงลำดับจากค่าน้อยไปหาค่ามาก (Ascending)
- เรียงลำดับจากค่ามากไปหาค่าน้อย (Descending)

๒.๑.๓.๒.๑ เรียงลำดับจากค่าน้อยไปหาค่ามาก ในกรณีนี้ จะต้องเริ่ม

อ่านทะเบียนข้อมูลเพื่อค้นหาตั้งแต่ทะเบียนข้อมูลแรกเช่นเดียวกัน อ่านไปเรื่อย ๆ จนกระทั่งพบทะเบียน ข้อมูลที่ต้องการ ในกรณีที่ทะเบียนข้อมูลที่ต้องการไม่อยู่ในแฟ้มข้อมูล ก็สามารถทราบได้โดยไม่จำเป็นต้อง

อ่านระเบียบข้อมูลจนกระทั่งหมดแฟ้มข้อมูลเมื่ออ่านแล้วเปรียบเทียบปรากฏว่า ระเบียบข้อมูลในแฟ้มข้อมูลมีค่ามากกว่า ระเบียบข้อมูลที่ต้องการค้นหา ก็สามารถบอกได้เลยว่า ระเบียบข้อมูลนั้นไม่อยู่ในแฟ้มข้อมูลที่ทำการค้นหา ดังนั้นโดยวิธีนี้จะทำให้ประหยัดเวลาในการค้นหาว่า การจัดข้อมูลแบบไม่เรียงลำดับ

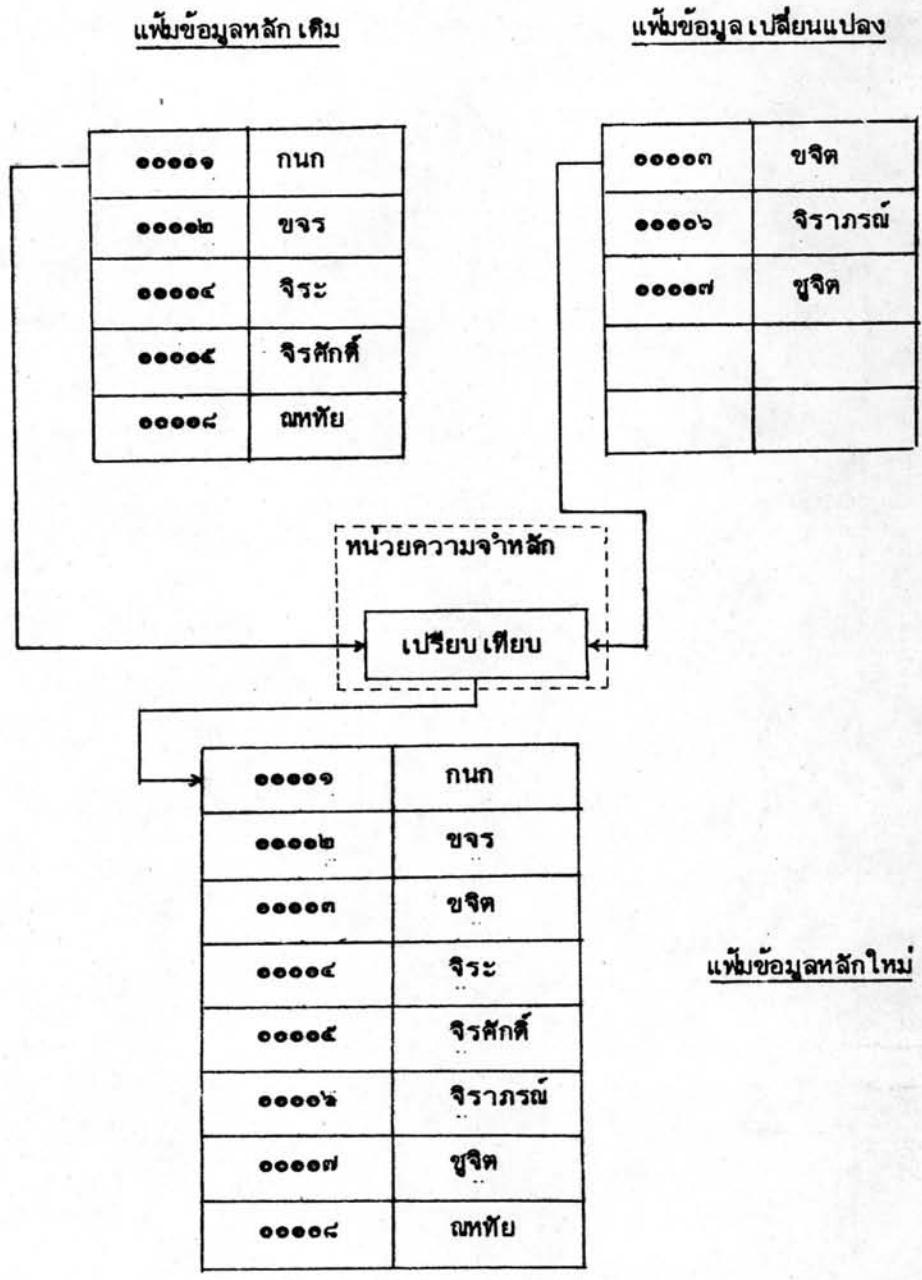
๒.๑.๓.๒.๒ เรียงลำดับจากค่ามากไปหาค่าน้อย มีหลักการในการค้นหา เช่นเดียวกับการเรียงลำดับจากค่าน้อยไปหาค่ามาก ต่างกันตรงที่ว่า กรณีระเบียบข้อมูลที่ต้องการค้นหา ไม่อยู่ในแฟ้มข้อมูล จะสามารถทราบได้เมื่อระเบียบข้อมูลในแฟ้มข้อมูล มีค่าน้อยกว่าระเบียบข้อมูลที่ต้องการค้นหา

๒.๑.๔ การปรับปรุง

ในการปรับปรุงแฟ้มข้อมูลชนิดนี้ จะต้องนำแฟ้มข้อมูลเปลี่ยนแปลง (Transaction file) ที่เรียงลำดับข้อมูลย่อย (field) เช่นเดียวกับแฟ้มข้อมูลหลักเดิม (Old master file) มาเปรียบเทียบกับระเบียบข้อมูลในแฟ้มข้อมูลหลักเดิมเพื่อทำการเพิ่ม ลบ หรือเปลี่ยนแปลงระเบียบข้อมูลในแฟ้มข้อมูลหลัก โดยการบันทึกระเบียบข้อมูลลงในแฟ้มข้อมูลใหม่อีกแฟ้มหนึ่ง ซึ่งเรียกว่าแฟ้มข้อมูลหลักใหม่ (New master file) ซึ่งวิธีการ (Algorithm) ของการเพิ่มเติม ลบ และเปลี่ยนแปลงระเบียบข้อมูลในแฟ้มข้อมูลหลักมีดังนี้

๒.๑.๔.๑ การเพิ่มเติม มีขั้นตอนในการทำ คืออ่านระเบียบข้อมูลจากแฟ้มข้อมูลเปลี่ยนแปลงและอ่านระเบียบข้อมูลจากแฟ้มข้อมูลหลักเดิม แล้วเปรียบเทียบข้อมูลย่อยของระเบียบข้อมูลในแต่ละแฟ้มข้อมูล ถ้าข้อมูลย่อยของระเบียบข้อมูลในแฟ้มข้อมูลหลักเดิม มีค่าน้อยกว่าก็จะบันทึกระเบียบข้อมูลในแฟ้มข้อมูลหลักเดิมลงในแฟ้มข้อมูลหลักใหม่แล้วอ่านระเบียบข้อมูลจากแฟ้มข้อมูลหลักเดิมใหม่อีกเพื่อนำมาเปรียบเทียบค่าต่อไป ถ้าข้อมูลย่อยในแฟ้มข้อมูลหลักเดิมมีค่าน้อยกว่าอีก ก็จะบันทึกระเบียบข้อมูลในแฟ้มข้อมูลหลักเดิมลงในแฟ้มข้อมูลหลักใหม่ ทำเช่นนี้ไปเรื่อย ๆ จนกระทั่งข้อมูลย่อยของแฟ้มข้อมูลหลักเดิมมีค่ามากกว่าข้อมูลย่อยในแฟ้มข้อมูลเปลี่ยนแปลง จึงบันทึกระเบียบข้อมูลในแฟ้มข้อมูลเปลี่ยนแปลงลงในแฟ้มข้อมูลหลักใหม่ แล้วจึงไปอ่านระเบียบข้อมูลจากแฟ้มข้อมูลเปลี่ยนแปลงเพื่อนำไปเปรียบเทียบอีก ทำเช่นนี้ไปเรื่อย ๆ จนกระทั่งหมดระเบียบข้อมูลจากแฟ้มข้อมูลหลักเดิม และแฟ้มข้อมูลเปลี่ยนแปลง

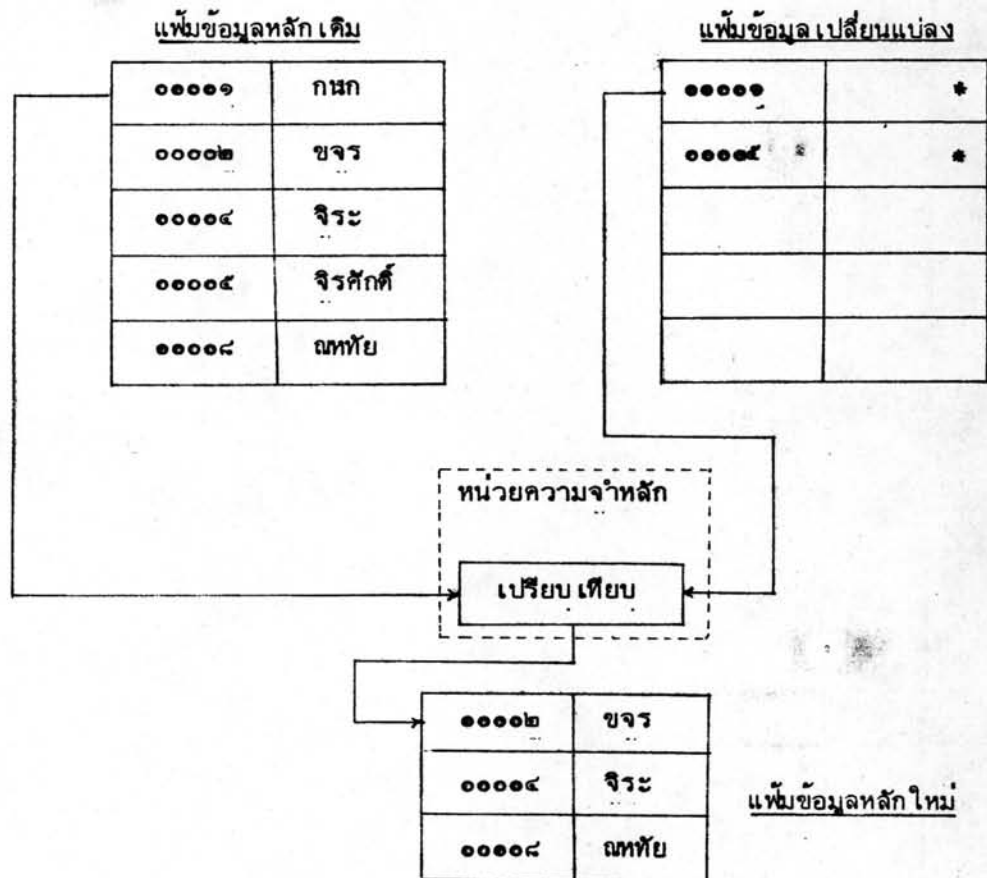
สมมติว่าต้องการเพิ่มระเบียนข้อมูล ในแฟ้มข้อมูลหลักเดิม ซึ่งมีข้อมูลย่อยเป็น เลขประจำตัว และชื่อ ดังแสดงในรูปที่ ๒.๓ ระเบียนข้อมูลที่ต้องการเพิ่มมีเลขประจำตัว ๐๐๐๐๓ ๐๐๐๐๖ และ ๐๐๐๐๗ ตามลำดับ ก็สามารถทำได้โดยอาศัยหลักเกณฑ์ข้างต้น



รูป ๒.๓ แสดงการเพิ่มระเบียนข้อมูลของแฟ้มข้อมูลแบบเรียงลำดับ

๒.๑.๔.๒ การลบทิ้ง อาศัยหลักในการทำ เช่นเดียวกับการเพิ่มเติม ต่างกันที่ว่าจะต้องอ่านระเบียบข้อมูลในแฟ้มข้อมูลหลัก เดิมจนกระทั่งข้อมูลย่อยในแฟ้มข้อมูลหลัก เดิม เท่ากับข้อมูลย่อยในแฟ้มข้อมูล เปลี่ยนแปลง และตรวจสอบดูว่าระเบียบข้อมูลในแฟ้มข้อมูล เปลี่ยนแปลงมีข้อมูลย่อยหนึ่งที่กำหนดไว้มีสัญลักษณ์โคลงสัญลักษณ์หนึ่งที่ระบุว่าเป็นการลบทิ้งหรือไม่ ถ้ามีระเบียบข้อมูลในแฟ้มข้อมูลหลัก เดิม ที่มีข้อมูลย่อย เท่ากันนี้จะไม่ถูกบันทึกลงในแฟ้มข้อมูลหลัก ใหม่

สมมติว่า ต้องการลบระเบียบข้อมูลที่มีเลขประจำตัว ๑๑๑๑๑, ๑๑๑๑๔ ออกจากแฟ้มข้อมูลหลัก เดิม ที่มีเลขประจำตัว ดังแสดงในรูป ๒.๔ ที่สามารถทำได้โดยอาศัยหลักเกณฑ์ข้างต้น

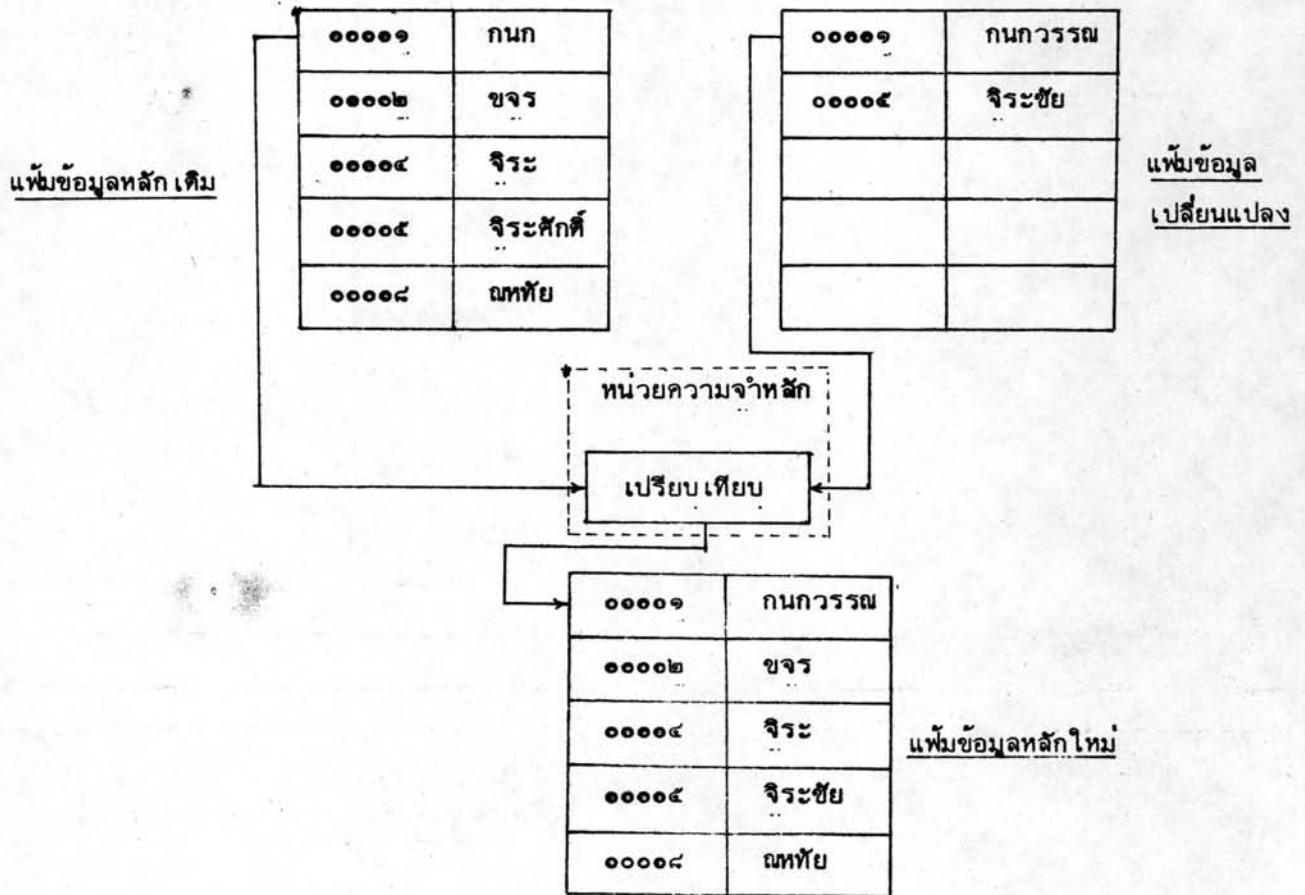


รูป ๒.๔ แสดงการลบระเบียบข้อมูลของแฟ้มข้อมูลแบบเรียงลำดับ

๒.๑.๔.๓ การเปลี่ยนแปลง

อาศัยหลักการเช่นเดียวกับการลบทิ้ง ต่างกันที่ว่า เมื่อพบระเบียบข้อมูลที่มีข้อมูลย่อยตรงกันและตรวจสอบดูว่าระเบียบข้อมูลในแฟ้มข้อมูลเปลี่ยนแปลงไม่มีข้อมูลย่อยที่กำหนดไว้เป็นสัญลักษณ์ที่ระบุว่าเป็นการลบทิ้ง ก็จะมีบันทึกระเบียบข้อมูลของแฟ้มข้อมูลเปลี่ยนแปลงลงในแฟ้มข้อมูลหลักใหม่

ตัวอย่างเช่น เมื่อต้องการเปลี่ยนแปลงระเบียบข้อมูลที่มีเลขประจำตัว ๐๐๐๐๑ จากชื่อ กนกไปเป็น กนกวรรณ และเลขประจำตัว ๐๐๐๐๕ จากชื่อ จิระศักดิ์ ไปเป็น จิระชัย ก็สามารถทำได้โดยอาศัยหลักเกณฑ์ดังที่กล่าวข้างต้น



รูปที่ ๒.๕ แสดงการเปลี่ยนแปลงระเบียบข้อมูลของแฟ้มข้อมูลหลักแบบ เรียงลำดับ

๒.๒ การจัดข้อมูลแบบเข้าถึงข้อมูลได้โดยตรง

การจัดข้อมูลแบบนี้แบ่งได้เป็น ๒ ประเภท^(๑๒) คือ

- การจัดข้อมูลแบบโคเร็ค (Direct Organization)
- การจัดข้อมูลแบบดัชนี (Index Organization)

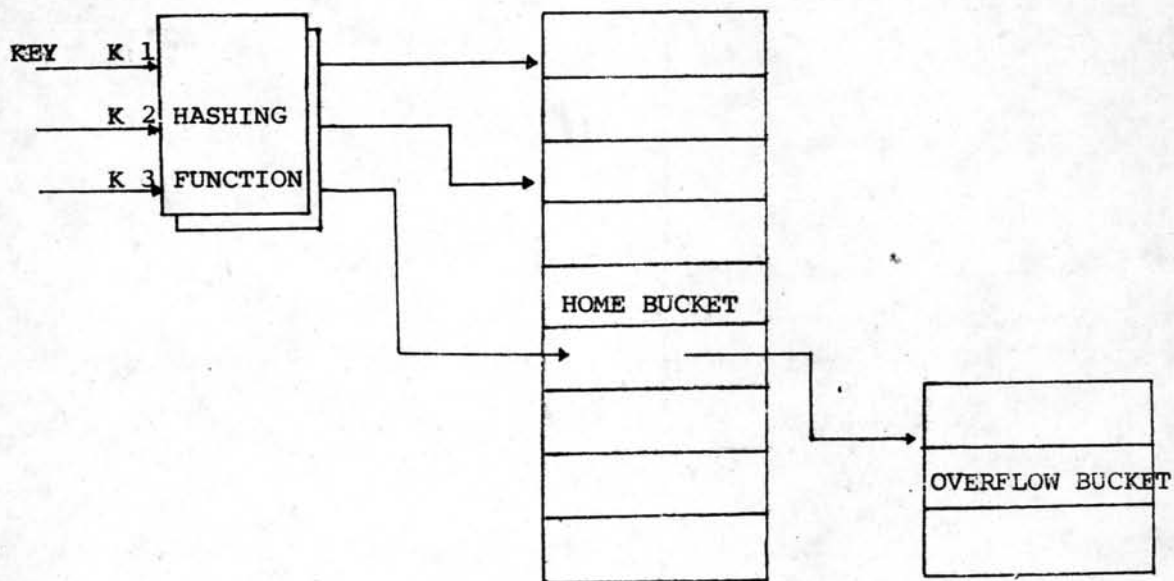
๒.๒.๑ การจัดข้อมูลแบบโคเร็ค

๒.๒.๑.๑ ลักษณะ

การจัดข้อมูลชนิดนี้อาศัยความสัมพันธ์ระหว่าง คีย์ (key) กับตำแหน่งที่เก็บ (Address) ระเบียบข้อมูลบนแผ่นจานแม่เหล็ก การหาคำแหน่งที่เก็บระเบียบข้อมูลได้จากการนำคีย์ไปผ่านการคำนวณโดยใช้ฟังก์ชัน ที่เรียกว่า แฮชฟังก์ชัน (Hashing function) ปกติแล้วลักษณะการเก็บข้อมูลนี้จะแบ่งพื้นที่เก็บในแฟ้มข้อมูลออกเป็น ๒ ส่วน^(๑๓) (ดังรูป ๒.๖)

ก) โฮมบัคเกต (Home bucket) ใช้เก็บข้อมูล

ข) บัคเกตส่วนเกิน (Overflow bucket) ใช้เก็บข้อมูลที่คำนวณได้ตำแหน่งที่เก็บซ้ำกันในโฮมบัคเกตและโฮมบัคเกตนั้น ๆ เต็มแล้ว

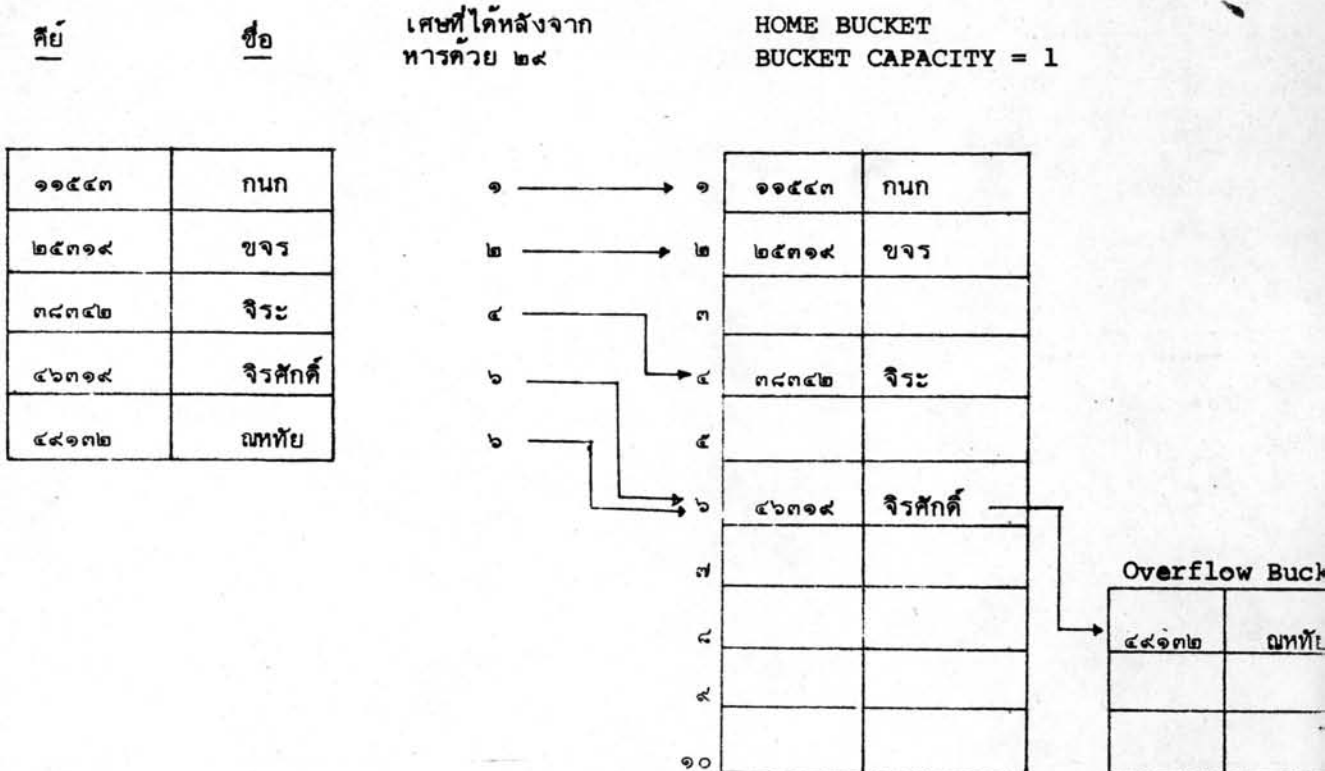


รูปที่ ๒.๖ แสดงการจัดข้อมูลแบบโคเร็ค

๒.๒.๑.๒ การสร้าง

สมมติว่ามีระเบียบข้อมูลอยู่จำนวนหนึ่ง และต้องการสร้างแฟ้มข้อมูลโดยใช้วิธีไคเร็ค ในขั้นแรกผู้ใช้จะต้องประมาณเนื้อที่ที่จะใช้ในแผ่นจานแม่เหล็ก โดยพิจารณาจากจำนวนช่องระเบียบข้อมูล ความยาวของระเบียบข้อมูล รวมทั้งเผื่อที่ไว้สำหรับบักเกตส่วนเกินด้วย จากนั้นจึงเลือกแอสซิงฟังก์ชันที่เหมาะสม (ภาคผนวก ข) เพื่อใช้เปลี่ยนค่าคีย์ของระเบียบข้อมูลให้เป็นตำแหน่งที่จะเก็บข้อมูลในโฮมบักเกต แล้วจึงบันทึกระเบียบข้อมูลลงในโฮมบักเกต (ซึ่งจะบันทึกระเบียบข้อมูลได้เท่าใดขึ้นอยู่กับผู้ใช้จะกำหนด) ตามตำแหน่งที่คำนวณได้ ถ้าตำแหน่งที่เก็บนั้น ๆ มีระเบียบข้อมูลอยู่แล้ว ก็จะเลื่อนไปบันทึกลงในบักเกตส่วนเกินและการบันทึกในบักเกตส่วนเกินนี้ก็จะมีการต่าง ๆ ตามความเหมาะสม และอาศัยหลักเกณฑ์นี้บันทึกระเบียบข้อมูลอื่น ๆ ต่อไปเรื่อย ๆ จนกระทั่งหมดระเบียบข้อมูลที่ต้องการบันทึก

ตัวอย่าง ถ้ามีระเบียบข้อมูลอยู่ ๔ ระเบียบข้อมูล และแต่ละระเบียบข้อมูลจะมีข้อมูลย่อยเป็นเลขประจำตัวและชื่อ โดยอาศัยหลักเกณฑ์ดังกล่าวข้างต้นจะสร้างแฟ้มข้อมูลได้ ดังแสดงในรูปที่ ๒.๗



รูป ๒.๗ การสร้างแฟ้มข้อมูลแบบไคเร็ค

๒.๒.๑.๓ การค้นหาข้อมูล

ในการค้นหาข้อมูล จากแฟ้มข้อมูลที่จะเปรียบเทียบข้อมูลแบบโคเร็ค สามารถทำได้ดังนี้

ขั้นแรกอ่านระเบียนข้อมูลจากแฟ้มข้อมูลเปลี่ยนแปลง แล้วเปลี่ยนค่าคีย์ของระเบียนข้อมูลให้เป็นตำแหน่งที่เก็บโดยใช้แอสซิงฟังก์ชันเดียวกันกับที่ใช้คอนสตรัง จากนั้นจึงค้นหาระเบียนข้อมูลจากตำแหน่งที่เก็บที่คำนวณได้นั้น ถ้าไม่พบระเบียนข้อมูลที่ต้องการ ก็จะต้องค้นหาใน bucket ส่วนเกินต่อไป ถ้าไม่พบระเบียนข้อมูลอีก แสดงว่าระเบียนข้อมูลนั้นไม่ได้อยู่ในแฟ้มข้อมูล

ตัวอย่างเช่น ต้องการค้นหาระเบียนข้อมูลที่มีเลขประจำตัว ๒๕๓๑๔, ๔๔๑๓๒ ก็สามารถทำได้ ดังแสดงในรูปที่ ๒.๘

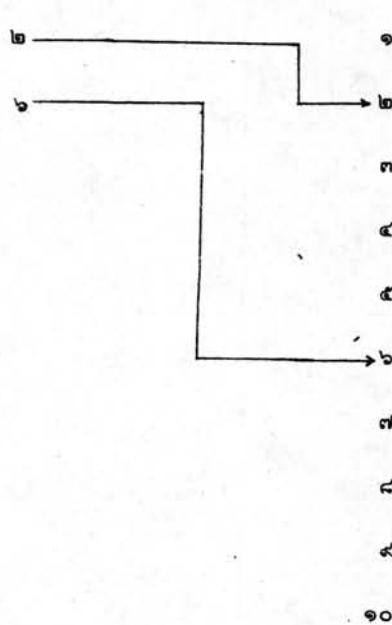
คีย์

เศษที่ได้หลังจาก
หารด้วย ๒๔

HOME BUCKET
BUCKET CAPACITY = 1

๒๕๓๑๔

๔๔๑๓๒



๑๑๕๔๓	กนก	OVERFLOW BUCKET
๒๕๓๑๔	ขจร	
๓๘๓๔๒	จิระ	
๔๖๓๑๔	จิรศักดิ์	

๔๔๑๓๒	ณทัทัย

รูป ๒.๘ แสดงการค้นหาข้อมูลในแฟ้มข้อมูลแบบโคเร็ค



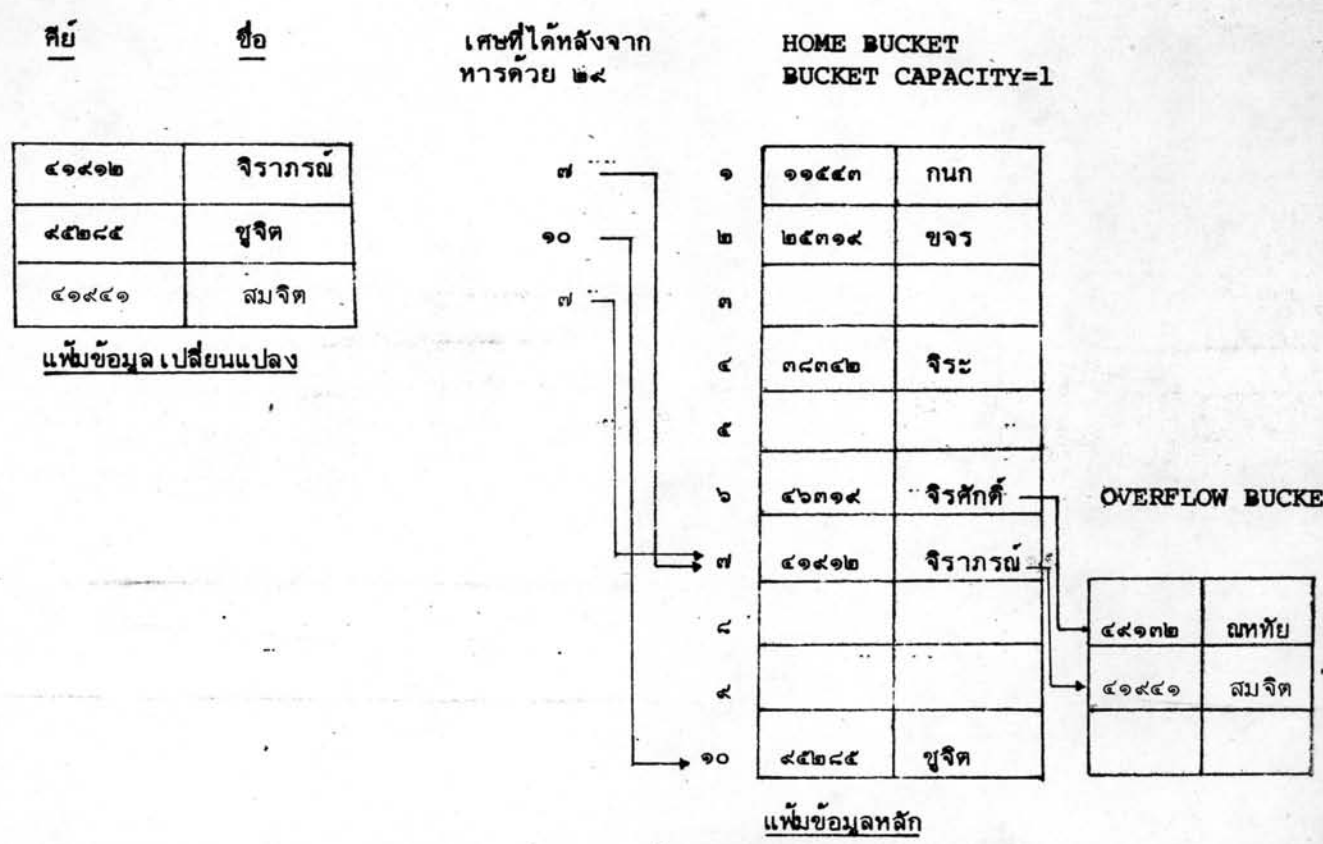
๒.๒.๑.๔ การปรับปรุง

ในการปรับปรุงแท้มข้อมูลที่จัดแบบโคเร็คนี้ ผู้ใช้สามารถทำการเพิ่มเติม ลบทิ้ง หรือ เปลี่ยนแปลงระเบียบข้อมูลได้ โดยปรับปรุงลงในแท้มข้อมูลหลักเดิม โดยไม่จำเป็นต้องมีแท้มข้อมูลหลักใหม่เหมือนกับการปรับปรุงแท้มข้อมูลที่จัดแบบเรียงลำดับ ซึ่งวิธีการของการเพิ่มเติม ลบทิ้ง และการเปลี่ยนแปลงแท้มข้อมูลหลักมีดังนี้

๒.๒.๑.๔.๑ การเพิ่มเติม

ในการเพิ่มเติมระเบียบข้อมูลนั้น ชั้นแรกจะต้องอ่านระเบียบข้อมูลจากแท้มข้อมูลเปลี่ยนแปลง แล้วเปลี่ยนค่าศีย์ของระเบียบข้อมูลให้เป็นตำแหน่งที่เก็บในโฮมบัคเกตโดยใช้แอสซิงฟังก์ชัน เช่นเดียวกับที่ใช้ตอนสร้าง เมื่อได้ตำแหน่งที่เก็บในโฮมบัคเกตแล้วจึงบันทึกระเบียบข้อมูลลงในตำแหน่งนั้น ๆ ซึ่งจะต้องตรวจสอบก่อนว่าที่ตำแหน่งนั้น ๆ มีระเบียบข้อมูลอยู่แล้วหรือไม่ ถ้ามีก็จะบันทึกระเบียบข้อมูลลงในบัคเกตส่วนเกิน

สมมติว่าต้องการเพิ่มระเบียบข้อมูลที่มีค่าศีย์เป็น ๔๑๔๑๒, ๔๕๒๔๔ ลงในแท้มข้อมูลที่สร้างไว้ในรูปที่ ๒.๗ ก็สามารถทำได้โดยอาศัยหลักการเช่นเดียวกันนี้ ดังแสดงในรูปที่ ๒.๘

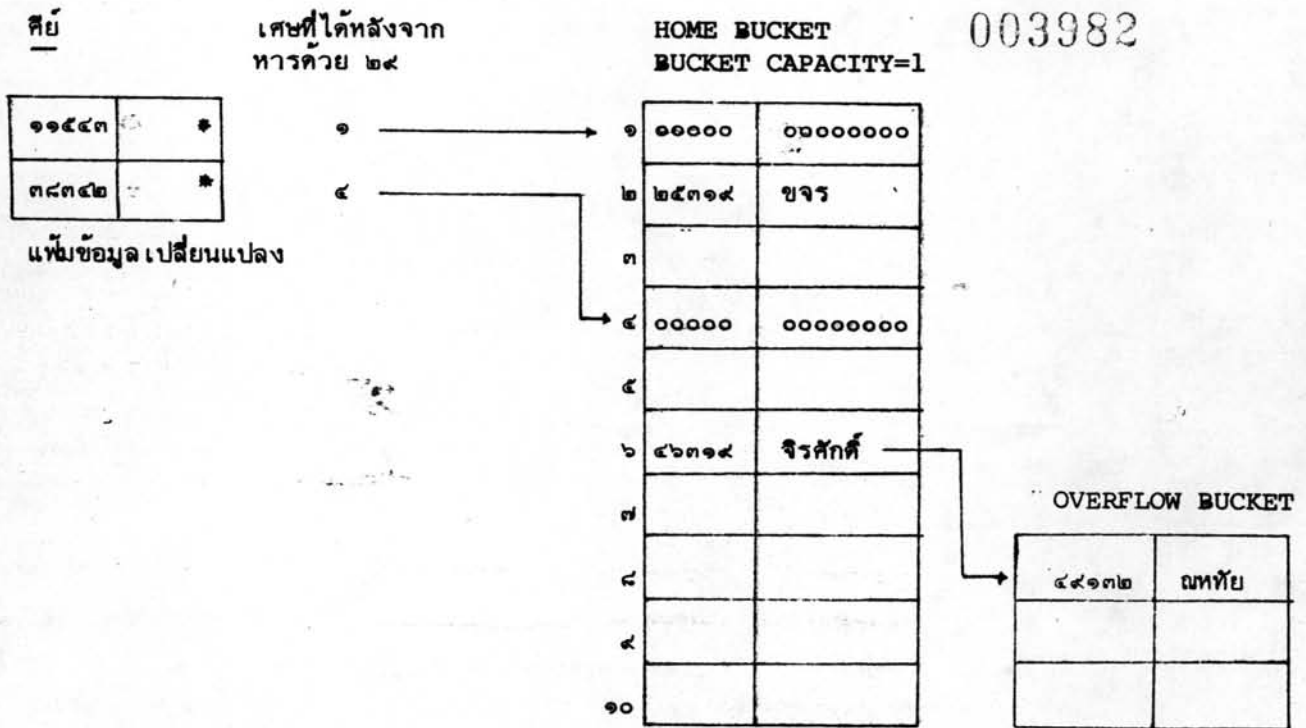


รูป ๒.๘ แสดงการเพิ่มระเบียบข้อมูลลงในแท้มข้อมูลที่จัดแบบโคเร็ค

๒.๒.๑.๔.๒ การลบทิ้ง

ผู้ใช้สามารถทำได้โดยอาศัยหลักเกณฑ์ดังนี้คือ ชั้นแรกจะต้องอ่านระเบียบข้อมูลจากแฟ้มข้อมูลเปลี่ยนแปลง และตรวจสอบดูว่าระเบียบข้อมูลในแฟ้มข้อมูลเปลี่ยนแปลงมีข้อมูลย่อยหนึ่ง กำหนดสัญลักษณ์ใดสัญลักษณ์หนึ่ง ที่ระบุว่าเป็นการลบทิ้งหรือไม่ ถ้ามีก็จะค้นหาระเบียนข้อมูลที่ต้องการในแฟ้มข้อมูลหลักจนพบ เมื่อพบแล้วจะบันทึก เครื่องหมาย (Set flag) ลงในข้อมูลย่อยที่กำหนดหรือจะทำพื้นที่ภายในระเบียบข้อมูลให้เป็นที่ว่างก็ได้ ถ้าค้นหาระเบียนข้อมูลนั้นไม่พบก็แสดงว่าในแฟ้มข้อมูลนั้น ไม่มีระเบียบข้อมูลที่จะให้ลบทิ้ง

ตัวอย่าง ถ้าต้องการลบระเบียบข้อมูลในแฟ้มข้อมูลที่สร้างไว้ในรูปที่ ๒.๗ คือต้องการลบระเบียบข้อมูลที่มีเลขประจำตัว ๑๑๕๔๓, ๓๘๓๔๒ ก็สามารถทำได้โดยอาศัยหลักเกณฑ์ดังกล่าว ดังแสดงในรูปที่ ๒.๑๐

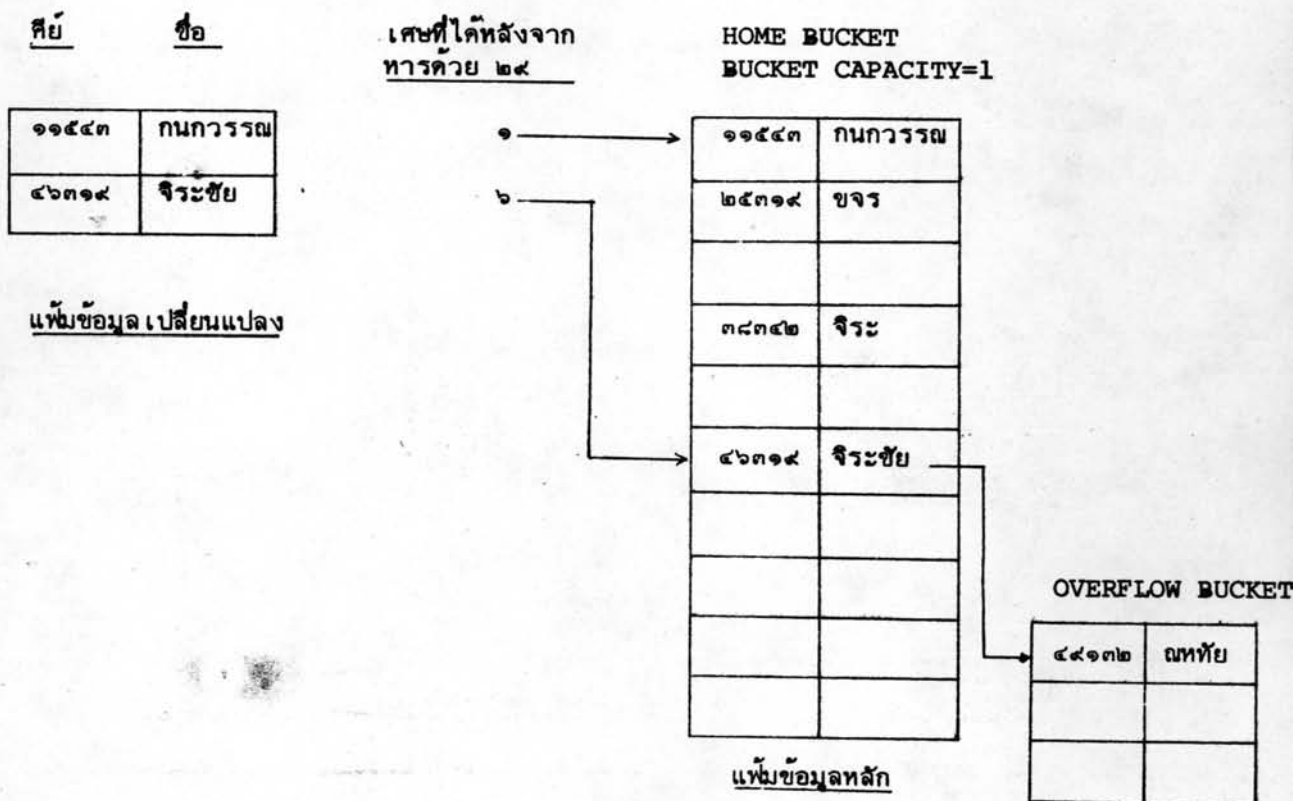


รูป ๒.๑๐ แสดงการลบระเบียบข้อมูลจากแฟ้มข้อมูลที่จัดแบบโคเร็ค

๒.๒.๑.๔.๓ การเปลี่ยนแปลง

อาศัยหลักเกณฑ์เช่นเดียวกับการลบทิ้ง ต่างกันที่ว่า
 ระเบียบข้อมูลในแฟ้มข้อมูลเปลี่ยนแปลงนั้น จะต้องไม่มีข้อมูลย่อยใดข้อมูลย่อยหนึ่ง ที่มีสัญลักษณ์
 ซึ่งระบุไว้ว่าเป็นการลบทิ้ง เมื่อทราบตำแหน่งที่เก็บของระเบียบข้อมูลที่จะเปลี่ยนแปลงแล้ว ก็จ
 ับนทึกระเบียบข้อมูลจากแฟ้มข้อมูลเปลี่ยนแปลงแทนระเบียบข้อมูลเดิมลงไปตำแหน่งนั้น ๆ

ตัวอย่าง ถ้าต้องการเปลี่ยนแปลงระเบียบข้อมูลที่มี
 เลขประจำตัว ๑๑๔๔๓ จากชื่อ กนก ไปเป็น กนกวรรณ และเลขประจำตัว ๔๖๓๑๔ จากชื่อ
 จิระศักดิ์ไปเป็น จิระชัย ก็สามารถทำได้โดยอาศัยหลักการดังกล่าว ดังแสดงในรูปที่ ๒.๑๑



รูป ๒.๑๑ แสดงการเปลี่ยนแปลงระเบียบข้อมูลในแฟ้มข้อมูลแบบโคเร็ค

ลักษณะของการจัดข้อมูลแบบโคเร็คดังที่ได้อธิบายข้างต้นแล้ว จะเห็นได้ว่าข้อมูลที่จัดแบบโคเร็คนี้จะทำงานอย่างมีประสิทธิภาพมากที่สุด ก็ต่อเมื่อมีระเบียนข้อมูลในบ๊อคเกทส่วนเกินน้อยหรือไม่มีเลย เพราะจะทำให้สามารถค้นหาระเบียนข้อมูลได้รวดเร็วโดยไม่ต้องเสียเวลาค้นหาในบ๊อคเกทส่วนเกินอีกทีหนึ่ง อย่างไรก็ตามแฟ้มข้อมูลที่จัดแบบโคเร็คนี้จะทำงานได้มีประสิทธิภาพย่อมขึ้นอยู่กับองค์ประกอบต่าง ๆ ดังนี้

- ๑) ขนาดของบ๊อคเกท (Bucket)
- ๒) แฟ็คกิงเดนซิตี (Packing Density)
- ๓) แอสซิงฟังก์ชันที่ใช้ (Hashing key-to-address transform)
- ๔) วิธีการที่จะจัดการกับระเบียนข้อมูลในพื้นที่ส่วนเกิน (Method of handling overflow)

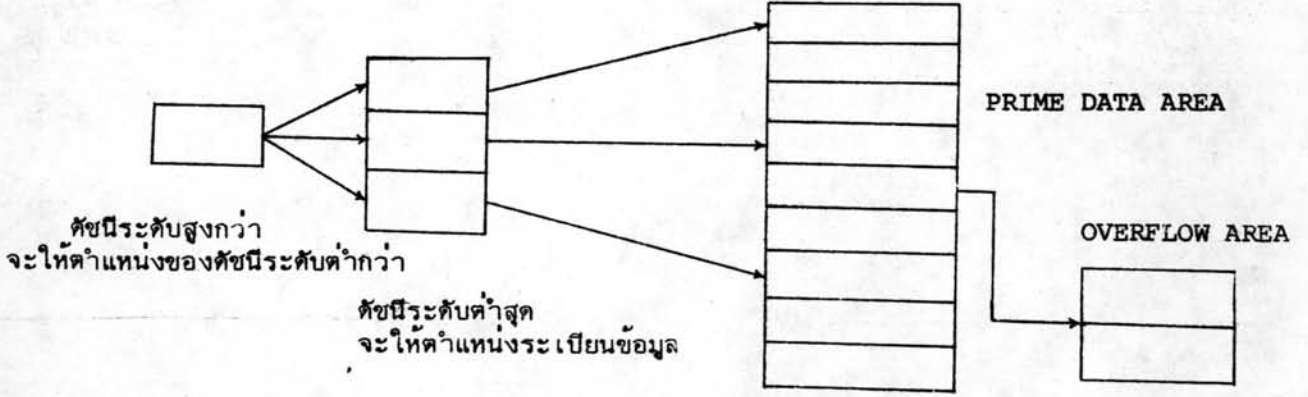
(ดูรายละเอียดในภาคผนวก ข)

๒.๒.๒ การจัดข้อมูลแบบดัชนี

การจัดข้อมูลชนิดนี้ จะแบ่งพื้นที่ในแฟ้มข้อมูลออกเป็น ๓ ส่วน^(๑๒) (ดังแสดงในรูป

๒.๑๒) คือ

- ดัชนี (Index) เป็นส่วนที่ใช้เก็บดัชนีเพื่อใช้ในการค้นหาข้อมูล ซึ่งจะแบ่งออกเป็นกี่ระดับ (level) ก็ได้ ขึ้นอยู่กับขนาดของแฟ้มข้อมูล
- พื้นที่หลัก (Prime data area) เป็นส่วนที่ใช้เก็บระเบียนข้อมูลหลัก ซึ่งแบ่งออกเป็นบ๊อคเกท แต่ละบ๊อคเกทจะบันทึกระเบียนข้อมูลที่ระเบียนข้อมูลก็ได้แล้วแต่ผู้ใช้
- พื้นที่ส่วนเกิน (Overflow area) เป็นส่วนที่ใช้เก็บระเบียนข้อมูลเมื่อมีการเพิ่มระเบียนข้อมูลและบ๊อคเกทในพื้นที่หลักมีระเบียนข้อมูลเต็มแล้ว



รูปที่ ๒.๑๒ แสดงการจัดแฟ้มข้อมูลแบบดัชนี

การจัดแฟ้มข้อมูลแบบ ดัชนีนี้แบ่งได้เป็น ๒ ชนิดคือ

- การจัดข้อมูลแบบดัชนีโดยเรียงลำดับ (Index sequential organization)
- การจัดข้อมูลแบบดัชนีโดยไม่เรียงลำดับ (Index non-sequential organization)

๒.๒.๒.๑ การจัดข้อมูลแบบดัชนีโดยเรียงตามลำดับ

๒.๒.๒.๑.๑ ลักษณะ การจัดการข้อมูลชนิดนี้แบ่งพื้นที่ออกเป็น ส่วน ๆ ตามข้อ ๒.๒.๒ แต่ระเบียบข้อมูลที่จัดจะต้องได้รับการเรียงลำดับก่อน จากค่าน้อยไปหาค่ามากและ ศัพท์จะมีค่าซ้ำกันไม่ได้ ระเบียบข้อมูลนี้จะต้องมีความยาวคงที่ โดยอาจถูกจัดเป็นบล็อกหรือไม่เป็น บล็อกก็ได้แล้วแต่ความเหมาะสมของงาน แต่จะต้องมีศัพท์กำกับอยู่เสมอ

๒.๒.๒.๑.๒ การสร้าง

ระเบียบข้อมูลที่ผู้ใช้จะต้องมีศัพท์เรียงตามลำดับจากค่าน้อย ไปหาค่ามาก และมีขั้นตอนในการทำดังนี้

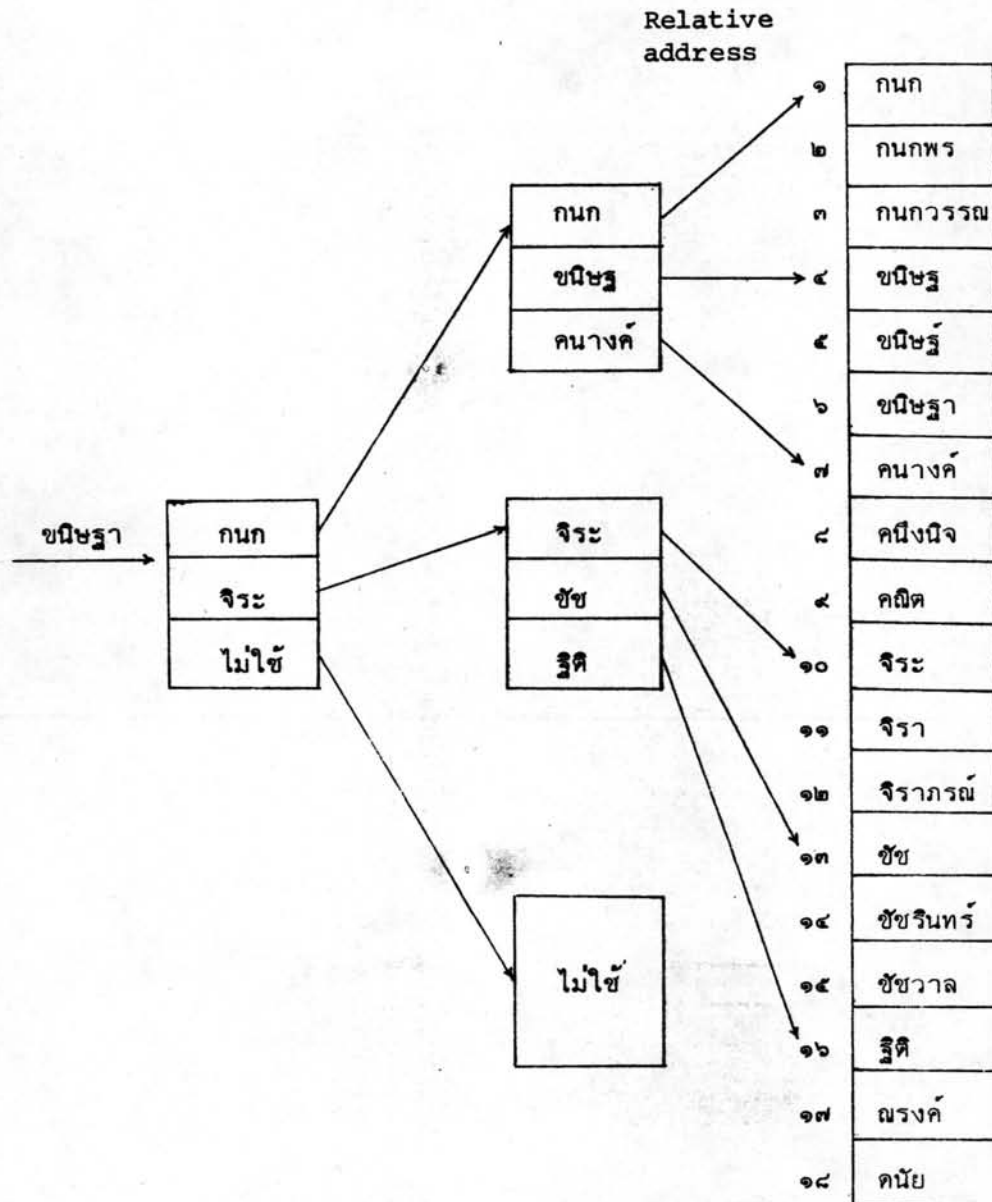
ขั้นแรกผู้ใช้จะต้องคำนวณหาจำนวนเนื้อที่จะใช้ในแผ่นจาน แม่เหล็กก่อน จำนวนเนื้อที่ใช้ดังกล่าวนั้นจะต้องพิจารณาทั้ง ๓ ส่วน คือ ส่วนที่ใช้เป็นพื้นที่ดัชนี พื้นที่หลัก และพื้นที่ส่วนเกิน สำหรับจำนวนเนื้อที่ของพื้นที่หลักที่จะใช้นั้น คำนวณได้จากจำนวนทั้งหมดของระเบียบ ข้อมูลความยาวศัพท์ของระเบียบข้อมูล ความยาวของระเบียบข้อมูล และลักษณะการจัดระเบียบข้อมูลว่าเป็น แบบบล็อกหรือไม่เป็นบล็อก ส่วนพื้นที่ดัชนีนั้นผู้ใช้จะต้องเผื่อเนื้อที่สำหรับสร้างดัชนีให้เพียงพอ ซึ่งพิจารณา ได้จากจำนวนระเบียบข้อมูลทั้งหมด และท้ายสุดสำหรับพื้นที่ส่วนเกินนั้นก็ประมาณเนื้อที่เผื่อไว้ สำหรับกรณี การแก้ไขปรับปรุงแฟ้มข้อมูล

ขั้นต่อมา ทำการบันทึกระเบียบข้อมูลลงในเนื้อที่ดังกล่าว บันทึกไปเรื่อย ๆ จนกระทั่งหมดระเบียบข้อมูลที่ต้องการบันทึก

๒.๒.๒.๑.๓ การค้นหา

เมื่อต้องการค้นหาระเบียบข้อมูลใด ก็สามารถกระทำโดยใช้ ศัพท์ที่ต้องการหาเป็นอินพุท (Input) ทำการเปรียบเทียบกับค่าเริ่มต้นของช่วงต่าง ๆ ในระดับดัชนี (Index level) ถ้าศัพท์มีค่าน้อยกว่าแสดงว่าไม่มีระเบียบข้อมูลนั้น ถ้าค่ามากกว่าก็ต้องทำการเปรียบเทียบต่อไปอีก เพื่อดูว่าควรอยู่ในช่วงนี้หรือช่วงถัดไป โดยเปรียบเทียบกับค่าเริ่มต้นของช่วงถัดไปอีก

ถ้าค่าที่นำมา เปรียบเทียบน้อยกว่าแสดงว่าอยู่ในช่วงที่ผ่านมา แต่ถ้าค่ายังคงมากกว่า ก็ทำการเปรียบเทียบเหมือนเดิมอีก เมื่อทราบว่าอยู่ในช่วงใดแล้ว ก็จะไปทำการค้นหาเช่นเดียวกันนี้อีกในระดับถัดไป ทำดังนี้ต่อไปเรื่อย ๆ แล้วแต่ว่าดัชนีจะมีกี่ระดับ เมื่อทราบแน่นอนแล้วว่าระบุข้อมูลนั้น ๆ อยู่ช่วงใดแล้ว จึงค่อยทำการค้นหาในช่วงนั้นที่ระบุข้อมูลจนกระทั่งพบระบุข้อมูลที่ต้องการ



จากรูป ๒.๑๓ สมมติว่า ต้องการค้นหาคนที่ชื่อ ฆนิษฐา ชั้นแรกจะไปดูที่ดัชนีระดับแรก ก่อน โดยใช้ฆนิษฐาเป็นคีย์นำไปเปรียบเทียบกับค่าเริ่มต้นซึ่งคือกนก จะเห็นได้ว่าฆนิษฐาต้องอยู่ถัดจากกนก จึงทำการเปรียบเทียบต่อไป ซึ่งคือ จิระ แต่ฆนิษฐาต้องอยู่ก่อนจิระ จึงพอจะทราบได้ว่าฆนิษฐาอยู่ในช่วงเดียวกับกนก ก็จะไปหาในดัชนีระดับถัดไปอีก ซึ่งขั้นต้นกนก ก็จะทำให้การเปรียบเทียบเหมือนเดิม จะพบว่าอยู่ในช่วงฆนิษฐา ก็จะไปเริ่มทำการค้นหาโดยเริ่มที่ตำแหน่งที่ ๔ หาไปเรื่อย ๆ จะพบว่าฆนิษฐาอยู่ตำแหน่งที่ ๖ ก็ทำการตั้งระเบียบข้อมูลนี้ออกมา

๒.๒.๒.๑.๔ การปรับปรุง

-การเพิ่มเติม

ในการเพิ่มเติมระเบียบข้อมูลนั้น เพื่อให้ระเบียบข้อมูลทั้งหมดยังคงเรียงตามลำดับอยู่ จะต้องเพิ่มระเบียบข้อมูลลงในตำแหน่งที่เรียงลำดับ เมื่อเพิ่มระเบียบข้อมูลใดระเบียบข้อมูลหนึ่ง ระเบียบข้อมูลอื่นก็ต้องเลื่อน (Shift) ไปหมด ในกรณีที่พื้นที่หลักมีระเบียบข้อมูลเต็มแล้ว ระเบียบข้อมูลใหม่ก็จะถูกบันทึกในพื้นที่ส่วนเกิน

-การลบทิ้ง

หาตำแหน่งของระเบียบข้อมูลที่ต้องการลบทิ้งแล้วบันทึกเครื่องหมาย (Set flag) ลงไปที่ตำแหน่งนั้น เพื่อแสดงการลบทิ้ง ทำเช่นนี้ไปเรื่อย ๆ จนกระทั่งหมดระเบียบข้อมูลที่ต้องการลบทิ้ง

-การเปลี่ยนแปลง

หาตำแหน่งของระเบียบข้อมูลที่ต้องการเปลี่ยนแปลง แล้วบันทึกข้อมูลใหม่ลงไปทับตำแหน่งเดิม

๒.๒.๒.๒ การจัดข้อมูลแบบดัชนีโดยไม่เรียงลำดับ

๒.๒.๒.๒.๑ ลักษณะ

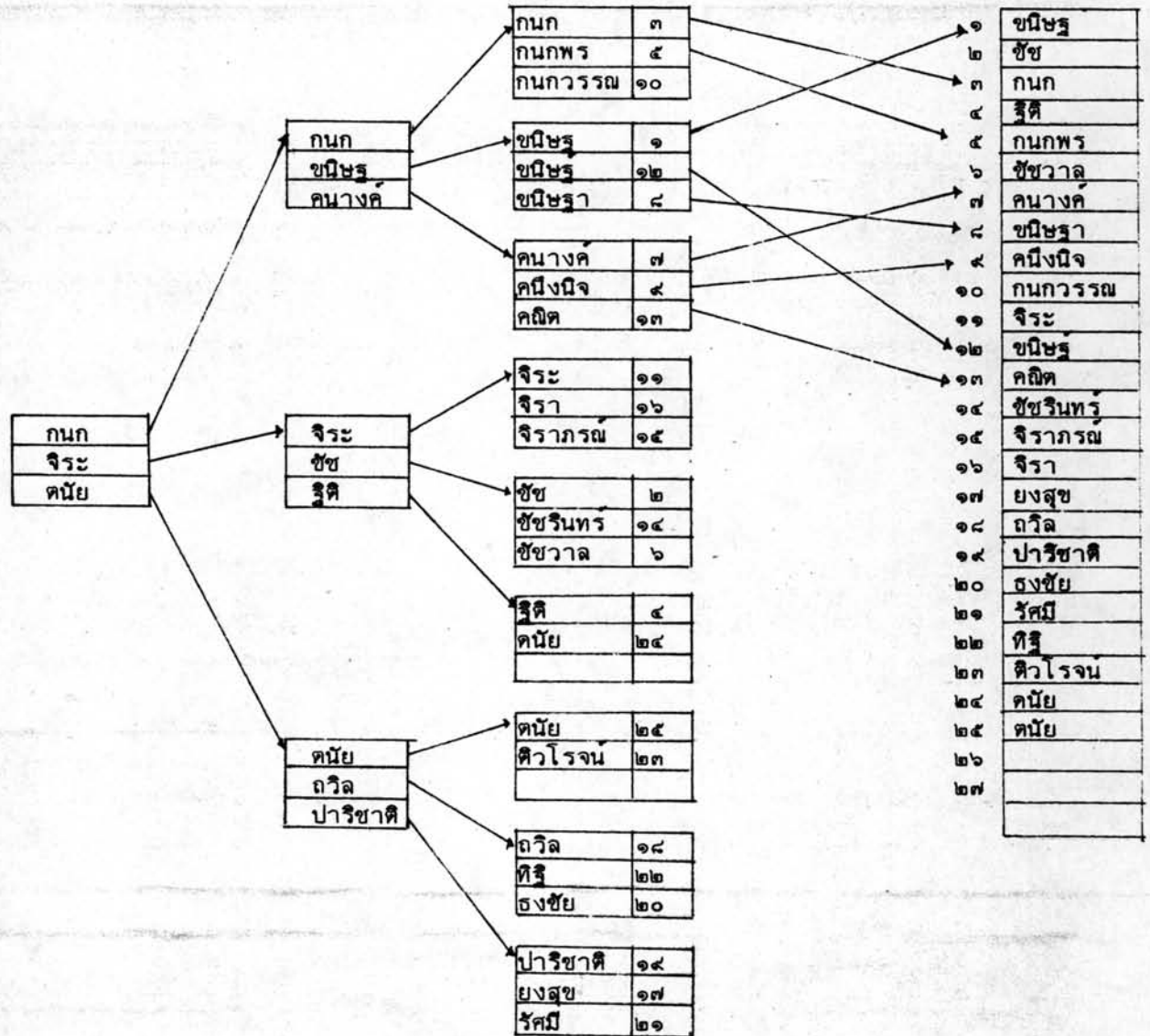
การจัดระเบียบข้อมูลชนิดนี้ก็เหมือนกับการจัดข้อมูลแบบดัชนี โดยเรียงลำดับต่างกันตรงที่ว่าระเบียบข้อมูลไม่จำเป็นต้องเรียงลำดับ

๒.๒.๒.๒.๒ การสร้าง

ประมาณเนื้อที่ที่จะใช้ในแผ่นจานแม่เหล็ก โดยพิจารณาจากจำนวนระเบียบข้อมูลความยาวของคีย์ และระเบียบข้อมูล รวมทั้งเผื่อที่ไว้สำหรับดัชนีด้วย แล้วทำการบันทึกระเบียบข้อมูลลงในเนื้อที่ที่เตรียมไว้ บันทึกต่อไปเรื่อย ๆ จนกระทั่งหมดระเบียบข้อมูลที่ต้องการบันทึก

๒.๒.๒.๒.๓ การค้นหา

ในการค้นหาจะเป็นข้อมูลที่ต้องการ สามารถทำได้โดย ค้นหาที่ดัชนีระดับสูงสุดก่อน แล้วจึงไปยังดัชนีระดับถัดไป เมื่อถึงดัชนีระดับต่ำสุดจะมีตัวชี้ (Pointer) ชี้ไปยังตำแหน่งที่เก็บของระเบียนข้อมูลที่ต้องการ



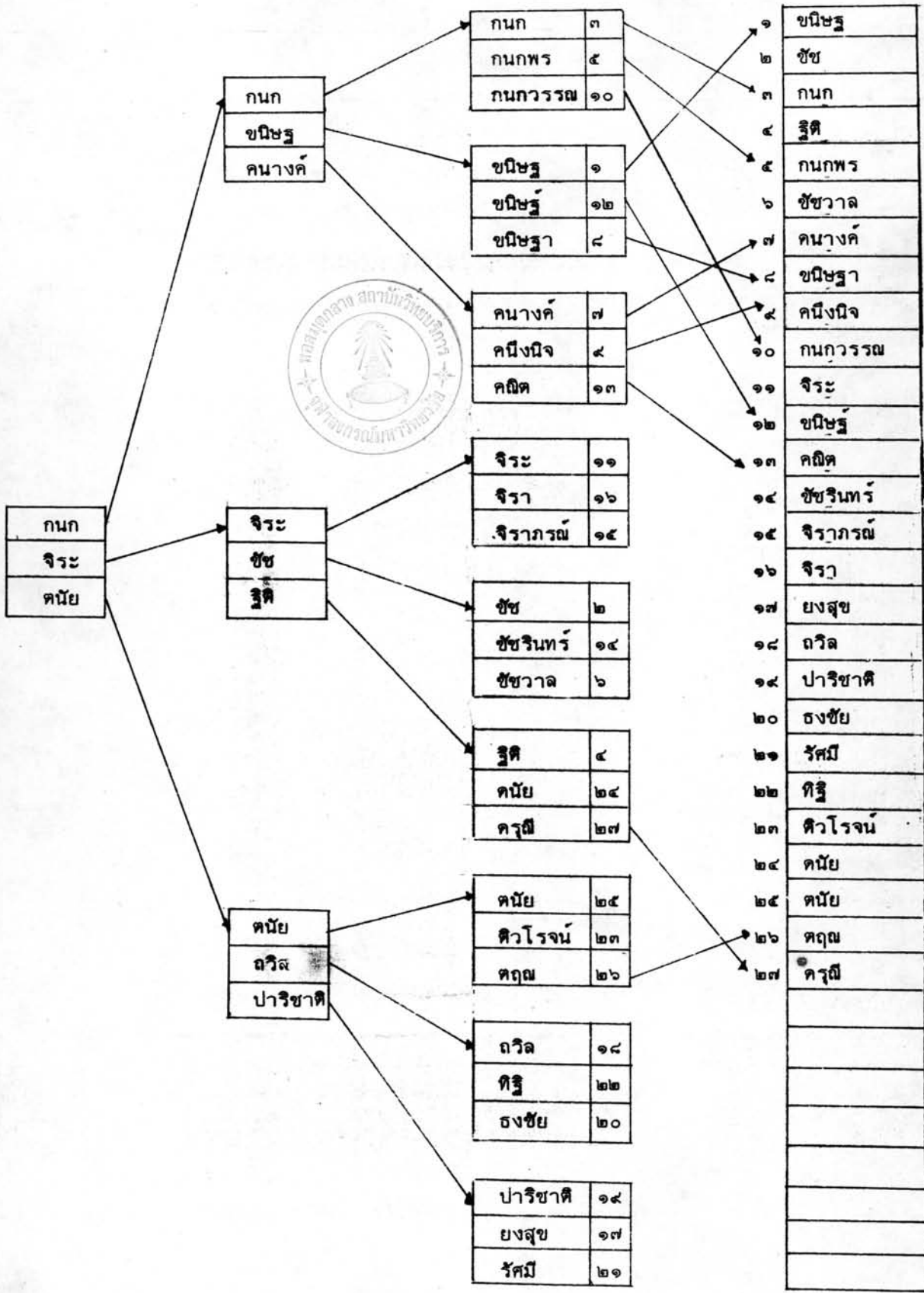
รูป ๒.๑๔ การจัดข้อมูลแบบดัชนีโดยไม่เรียงลำดับ

จากรูป ๒.๑๔ ถ้าต้องการหาคนที่ชื่อ ชนิษฐ์ เริ่มแรกจะดูที่ดัชนีระดับสูงสุดก่อน ระเบียบข้อมูลแรกที่อยู่ คือ กนก จาก กนกจะมีตัวชี้ไปยังระดับถัดไป ทำการค้นหาระเบียนข้อมูลในช่วงของกนกก่อน โดยค้นหาทีละระเบียบข้อมูลพบว่าไม่มีระเบียบข้อมูลที่ต้องการ จึงกลับมาหาในช่วงถัดไปคือในช่วงของชนิษฐ์ ทำการค้นหาอีกจะพบว่า ชนิษฐ์อยู่ที่ตำแหน่งที่เก็บหมายเลข ๑๒ ก็ไปยังตำแหน่งที่เก็บนั้น ถึงระเบียบข้อมูลนี้ออกมา

๒.๒.๒.๒.๔ การปรับปรุง

-การเพิ่มเติม

เมื่อต้องการเพิ่มเติมระเบียบข้อมูล สามารถทำได้โดยเพิ่มลงที่ส่วนท้ายของแฟ้มข้อมูล เช่นต้องการเพิ่มระเบียบข้อมูลของคนที่มีชื่อ ครูณี ตฤณ ลงในแฟ้มข้อมูล ดังแสดงในรูป ๒.๑๔ ก็สามารถทำได้ดัง แสดงในรูปที่ ๒.๑๕



รูปที่ ๒.๑๕ การเพิ่มระเบียบข้อมูลลงในแฟ้มข้อมูลแบบไม่เรียงลำดับ

- การลบทิ้ง

ในการลบทิ้ง เมื่อได้ตำแหน่งของระเบียนข้อมูลที่ต้องการลบทิ้งแล้วใส่เครื่องหมายบันทึกลงไป

- การเปลี่ยนแปลง

หาตำแหน่งของระเบียนข้อมูลที่ต้องการเปลี่ยนแปลง เมื่อได้ตำแหน่งที่ต้องการแล้ว ก็บันทึกข้อมูล

ใหม่ทับลงไปเดิม