

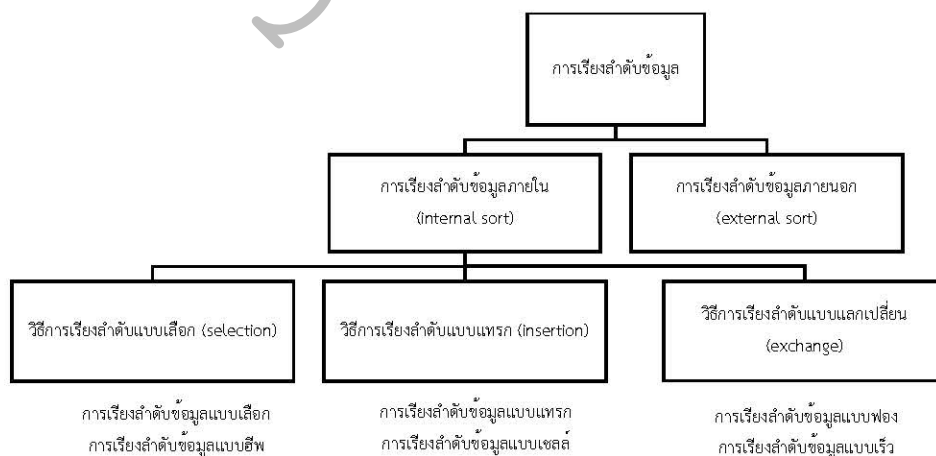
บทที่ 9

sorting

การเรียงลำดับเป็นสิ่งจำเป็นในการค้นหาข้อมูลที่รวดเร็วและมีประสิทธิภาพซึ่งเกี่ยวข้องกับการจัดเรียงข้อมูลเพื่อสร้างลำดับทำให้ง่ายต่อการค้นหา ข้อมูลสามารถเรียงลำดับจากน้อยไปหามาก (ascending order) หรือมากไปหาน้อย (descending order) ตามแอตทริบิวต์ต่าง ๆ เช่น ชื่อ อายุ ขนาด หมายเลขซีเรียล ลำดับความสำคัญ ค่า หรือประเภท การเรียงลำดับเป็นขั้นตอนเบื้องต้นในการค้นหาโดยการค้นหาแบบไบนารีจะเร็วขึ้นอย่างมากในข้อมูลที่เรียงลำดับเมื่อเทียบกับการค้นหาเชิงเส้น มีอัลกอริทึมการเรียงลำดับหลายแบบ บทนี้กล่าวถึงอัลกอริทึมการเรียงลำดับ อาทิ การเรียงแบบเลือก การเรียงแบบฟอง การเรียงแบบแทรก การเรียงแบบผสาน และการเรียงแบบเร็ว

การเรียงลำดับมีอยู่หลายวิธีแต่ละวิธีมีข้อดีข้อเสียแตกต่างกัน การเลือกการเรียงลำดับวิธีใดขึ้นอยู่กับลักษณะของข้อมูลและจำนวนของข้อมูลที่ต้องการเรียงลำดับ การเรียงลำดับแบ่งเป็น 2 ประเภท คือ

1. การเรียงลำดับภายใน (internal sort) ในขณะที่ทำการเรียงลำดับข้อมูลทั้งหมด ต้องอยู่ในหน่วยความจำหลัก
2. การเรียงลำดับภายนอก (external sort) ในขณะที่ทำการเรียงลำดับข้อมูลบางส่วนเท่านั้นที่อยู่ใน หน่วยความจำหลักข้อมูลส่วนที่เหลืออยู่ในความจำสำรอง เช่น จานแม่เหล็ก (magnetic disk) เทปแม่เหล็ก (magnetic tape) เหตุที่ข้อมูล ที่ต้องการเรียงลำดับอยู่ในหน่วยความจำสำรอง เนื่องจากข้อมูลมีมากจนเกินไม่สามารถเก็บในหน่วยความจำหลักได้ทั้งหมด (นิสาชล โตอดิเทพย์, 2541, หน้า 183) ในบทนี้จะกล่าวถึงการเรียงลำดับภายในเท่านั้น



การจัดเรียงแบบฟอง (bubble sort)

การจัดเรียงแบบฟอง (bubble sort) เป็นการเปรียบเทียบข้อมูลที่อยู่ในตำแหน่งที่ใกล้เคียงกัน และให้สลับตำแหน่งกัน ถ้าทั้งสองค่าไม่ตรงตามข้อกำหนดที่กำหนดไว้ เช่น กำหนดให้จัดเรียงข้อมูลจากน้อยไปมาก เป็นต้น และในการจัดเรียงข้อมูลต้องมีข้อมูลในการจัดเรียงมากกว่า 2 ข้อมูล (วิชณู ช่างเนียม, 2556, หน้า 226)

ฟังก์ชันการจัดเรียงฟองโดยการเปรียบเทียบและสลับค่าที่อยู่ติดกันอย่างเป็นระบบจนกว่าจะมีการประมวลผลรายการทั้งหมดเพื่อให้แน่ใจว่าวัตถุที่ใหญ่ที่สุดถูกย้ายไปที่ด้านล่าง กระบวนการเริ่มต้นด้วยการเปรียบเทียบรายการ $n-1$ และ n จากนั้นดำเนินการตามลำดับเป็น $n-2$ และ $n-1$ ต่อเนื่องจนกว่าจะถึง $[1]$ และ $[0]$ วิธีนี้ย้ายองค์ประกอบที่เล็กที่สุดไปยังด้านบนสุดของรายการได้อย่างมีประสิทธิภาพ อัลกอริธึมจะทำซ้ำ $n - 1$ ครั้งผ่านลำดับขององค์ประกอบ n เปรียบเทียบ และสลับองค์ประกอบที่อยู่ติดกันเพื่อค่อยๆ เลื่อนองค์ประกอบขนาดใหญ่ลง การผ่านแต่ละครั้งเกี่ยวข้องกับเปรียบเทียบค่าสองค่าสลับหากจำเป็นและก้าวไปสู่คู่ถัดไป ตัวเลขที่ให้มาแสดงให้เห็นถึงการผ่านเพียงครั้งเดียวโดยใช้การจัดเรียงฟอง (Drozdek, 2013, p. 497 & Sherrod, 2007, p. 92)

การจัดเรียงแบบฟอง (bubble sort) เป็นการเปรียบเทียบข้อมูลที่อยู่ติดกันภายในลิสต์ และสลับตำแหน่งตามเงื่อนไขที่กำหนดไว้ เช่น กำหนดให้จัดเรียงข้อมูลจากน้อยไปมาก หากข้อมูลด้านล่าง (หรือด้านซ้าย) มีค่าน้อยกว่าด้านบน (หรือด้านขวา) จะสลับตำแหน่งกัน เมื่อลิสต์มี n element กระบวนการจัดเรียงแบบฟอง เริ่มจากรอบแรกทำการเปรียบเทียบรายการ n และ $n-1$ และทำการสลับค่าตามเงื่อนไข และทำการเปรียบเทียบข้อมูลคู่ถัดไป $n-1$ และ $n-2$ ต่อไปเรื่อยๆ จนถึงข้อมูลตัวที่สองและตัวแรก (คู่แรก) ในรอบที่สองเริ่มจาก การเปรียบเทียบและสลับค่าที่อยู่ติดกันไปจนครบ $n-1$ รอบ (อุหมาด หมดอาด้า, 2564, หน้า 266-267 และ วิชณู ช่างเนียม, 2556, หน้า 226)

ตัวอย่างที่ 9.1 อัลกอริทึมสำหรับการเรียงลำดับแบบฟอง กำหนดให้ A เป็นลิสต์ มี 6 element $\{23, 78, 45, 8, 56, 32\}$ มีขั้นตอนการเรียงลำดับจากน้อยไปมากดังนี้

ลิสต์	23 78 45 8 56 32
32 < 56 สลับ	23 78 45 8 56 32
32 > 8 ไม่สลับ	23 78 45 8 32 56
8 < 45 สลับ	23 78 8 45 32 56
8 < 78 สลับ	23 8 78 45 32 56
8 < 23 สลับ	8 23 78 45 32 56
	8 23 78 45 32 56

ภาพประกอบที่ 9.1 การเรียงแบบฟองรอบแรก (ก)

ที่มา : ดัดแปลงจาก Bae, (2019, pp. 129-130) & Gilberg & Forouzan (2005, p. 559)

รอบที่หนึ่ง

ลิสต์	23	23	23	23	23	23	8
	78	78	78	78	78	78	32
	45	45	45	45	45	45	78
	8	8	8	8	8	45	45
	56	56	32	32	32	32	56
	32	32	56	56	56	56	32

ขั้นที่ 1 เปรียบเทียบข้อมูลตำแหน่งสุดท้าย (32) และตำแหน่งที่ห้า (56) พบว่าตำแหน่งสุดท้ายมีค่าน้อยกว่าตำแหน่งที่ห้า ต้องสลับตำแหน่ง

ขั้นที่ 2 เปรียบเทียบข้อมูลตำแหน่งที่ห้า (32) และตำแหน่งที่สี่ (8) พบว่าตำแหน่งที่ห้ามีค่ามากกว่าตำแหน่งที่สี่ ไม่ต้องสลับตำแหน่ง

ขั้นที่ 3 เปรียบเทียบข้อมูลตำแหน่งที่สี่ (8) และตำแหน่งที่สาม (45) พบว่าตำแหน่งที่สี่มีค่าน้อยกว่าตำแหน่งที่สาม ต้องสลับตำแหน่ง

ขั้นที่ 4 เปรียบเทียบข้อมูลตำแหน่งที่สาม (8) และตำแหน่งที่สอง (78) พบว่าตำแหน่งที่สามมีค่าน้อยกว่าตำแหน่งที่สอง ต้องสลับตำแหน่ง

ขั้นที่ 5 เปรียบเทียบข้อมูลตำแหน่งที่สอง (8) และตำแหน่งที่หนึ่ง (23) พบว่าตำแหน่งที่สองมีค่าน้อยกว่าตำแหน่งที่หนึ่ง ต้องสลับตำแหน่ง

ขั้นที่ 1 เปรียบเทียบข้อมูลตำแหน่งสุดท้าย (56) และตำแหน่งที่ห้า (32) พบว่าตำแหน่งสุดท้ายมีค่ามากกว่าตำแหน่งที่ห้า ไม่ต้องสลับตำแหน่ง

รอบสอง

8	8	8	8	8
23	23	23	23	23
78	78	32	32	32
45	32	78	78	78
32	45	45	45	45
56	56	56	56	56

ชั้นที่ 2 เปรียบเทียบข้อมูลตำแหน่งที่ห้า (32) และตำแหน่งที่สี่ (45)

พบว่าตำแหน่งที่ห้ามีค่าน้อยกว่าตำแหน่งที่สี่ ต้องสลับตำแหน่ง

ชั้นที่ 3 เปรียบเทียบข้อมูลตำแหน่งที่สี่ (32) และตำแหน่งที่สาม (78)

พบว่าตำแหน่งที่สี่มีค่ามากกว่าตำแหน่งที่สาม ไม่ต้องสลับตำแหน่ง

ชั้นที่ 4 เปรียบเทียบข้อมูลตำแหน่งที่สาม (32) และตำแหน่งที่สอง (23)

พบว่าตำแหน่งที่สามมีค่ามากกว่าตำแหน่งที่สอง ไม่ต้องสลับตำแหน่ง

รอบสาม

8	8	8	8
23	23	23	23
32	32	32	32
78	78	45	45
45	45	78	78
56	56	56	56

ชั้นที่ 1 เปรียบเทียบข้อมูลตำแหน่งสุดท้าย (56) และตำแหน่งที่ห้า (45)

พบว่าตำแหน่งสุดท้ายมีค่ามากกว่าตำแหน่งที่ห้า ไม่ต้องสลับตำแหน่ง

ชั้นที่ 2 เปรียบเทียบข้อมูลตำแหน่งที่ห้า (45) และตำแหน่งที่สี่ (78)

พบว่าตำแหน่งที่ห้ามีค่าน้อยกว่าตำแหน่งที่สี่ ต้องสลับตำแหน่ง

ชั้นที่ 3 เปรียบเทียบข้อมูลตำแหน่งที่สี่ (45) และตำแหน่งที่สาม (32)

พบว่าตำแหน่งที่สี่มีค่ามากกว่าตำแหน่งที่สาม ไม่ต้องสลับตำแหน่ง

รอบสี่

8	8	8
23	23	23
32	32	32
45	45	45
78	56	56
56	78	78

ชั้นที่ 1 เปรียบเทียบข้อมูลตำแหน่งสุดท้าย (56) และตำแหน่งที่ห้า (78)

พบว่าตำแหน่งสุดท้ายมีค่าน้อยกว่าตำแหน่งที่ห้า ต้องสลับตำแหน่ง

ชั้นที่ 2 เปรียบเทียบข้อมูลตำแหน่งที่ห้า (56) และตำแหน่งที่สี่ (45)

พบว่าตำแหน่งที่ห้ามีค่ามากกว่าตำแหน่งที่สี่ ไม่ต้องสลับตำแหน่ง