เปรียบเทียบอัลกอริทึมการใช้รูปแบบ Sort **ต่างๆ**

จากการทดลองการ Sort ทั้งสามแบบผ่านชุดตัวอย่างข้อมูลที่ใช้ทดลอง สรุปได้ว่า

1. **Bubble Sort**

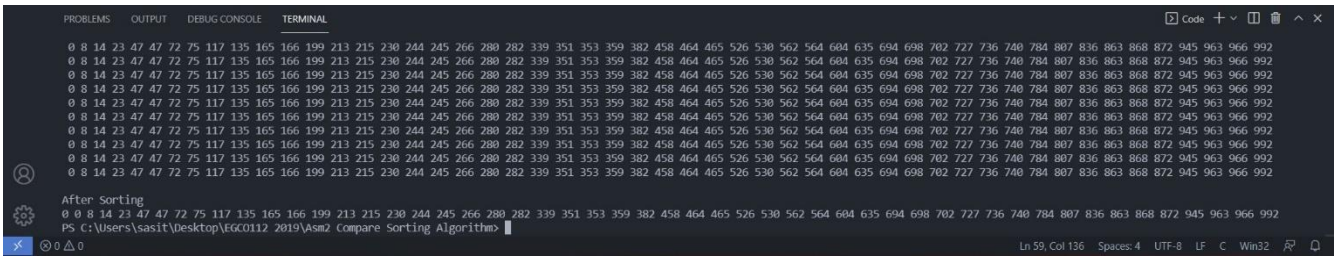
มีหลักการคือแต่ละรอบการทำงานจะมีการเปรียบเทียบข้อมูลเป็นคู่ที่อยู่ติดกัน ซึ่งหากอยู่ผิดตำแหน่งก็จะจัดเรียงใหม่ด้วยการสลับตำแหน่งกัน กล่าวคือ การทำงานจะเริ่มจากข้อมูลตัวสุดท้ายภายในลิสต์และทำการเปรียบเทียบกับค่าถัดไปที่อยู่ข้างหน้าซึ่งอยู่ติดกัน โดยหากค่าตัวเลขที่อยู่ท้ายมีค่าน้อยกว่า (กรณีเรียงตัวเลขจากน้อยไปมาก) ก็จะทำการสลับตำแหน่งกับตัวก่อนหน้า

ข้อดี

- เป็นวิธีเรียงลำดับที่ง่ายที่สุด การจัดเรียงลำดับข้อมูลไม่สลับซับซ้อน
- ถ้าข้อมูลมีจำนวนน้อยหรือผ่านการจัดเรียงลำดับมาก่อนอยู่แล้ว การเรียงลำดับวิธีนี้จะมีความเร็วมาก

ข้อเสีย

- มีการสลับตำแหน่งเกิดขึ้นหลายรอบมาก เนื่องจากต้องเปรียบเทียบทีละคู่แล้ววนกลับมาตรวจสอบหรือเปรียบเทียบอีกรอบ ซึ่งถ้าหากมีชุดตัวเลขที่เยอะจะทำให้จัดเรียงได้ช้าหรือใช้ระยะเวลานาน



2. **Selection Sort**

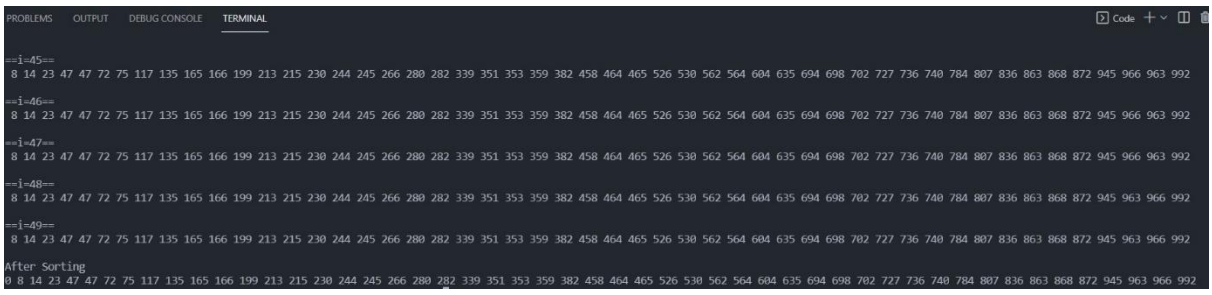
เป็นการจัดเรียงโดยจะค้นหาหรือสแกนค่าตัวเลขที่มีค่าน้อยที่สุดภายในลิสต์ โดยในรอบแรกจะเริ่มต้นสแกนค้นหาตั้งแต่ตัวแรกจนถึงตัวสุดท้าย หลังจากพบค่าที่น้อยที่สุดแล้วก็จะสลับตำแหน่งกัน จากนั้นในรอบถัดไปก็จะขยับตำแหน่งไปยังตัวถัดไปซึ่งก็คือตัวที่สอง แล้วก็สแกนหาค่าตัวเลขที่มีค่าน้อยที่สุดตั้งแต่ตัวที่สองจนถึงตัวสุดท้าย เมื่อพบแล้วก็จะสลับตำแหน่งกันเหมือนเดิม และจะทำแบบนี้จนครบทุกตัวเลข

ข้อดี

- การทำงานเข้าใจง่าย
- ทำงานได้ดีกับข้อมูลที่มีปริมาณน้อย

ข้อเสีย

- แม้ว่า Selection Sort จะเป็นวิธีที่เรียบง่ายตรงไปตรงมาคล้ายกับการเรียงด้วยมือ แต่ก็เป็นวิธีที่มีจำนวนรอบการทำงานมากที่สุด ดังนั้นหากมีข้อมูลในปริมาณมาก การเรียงลำดับข้อมูลด้วยวิธีนี้จึงไม่เหมาะ



3. **Insertion Sort**

ขั้นตอนการจัดเรียงแบบ Insertion Sort นั้น เป็นการนำเทคนิคการจัดเรียงแบบ Selection Sort มาปรับปรุงเพิ่มเติม โดยแทนที่จะอ่านหรือสแกนข้อมูลหรือตัวเลขจนครบทุกตัวเพื่อหาค่าน้อยที่สุด แต่สิ่งที่ Insertion Sort ทำคือจะเปรียบเทียบค่ากับตำแหน่งถัดไปแทน โดยถ้าตำแหน่งถัดไปมีค่าน้อยกว่าก็จะนำมาสลับตำแหน่งกัน

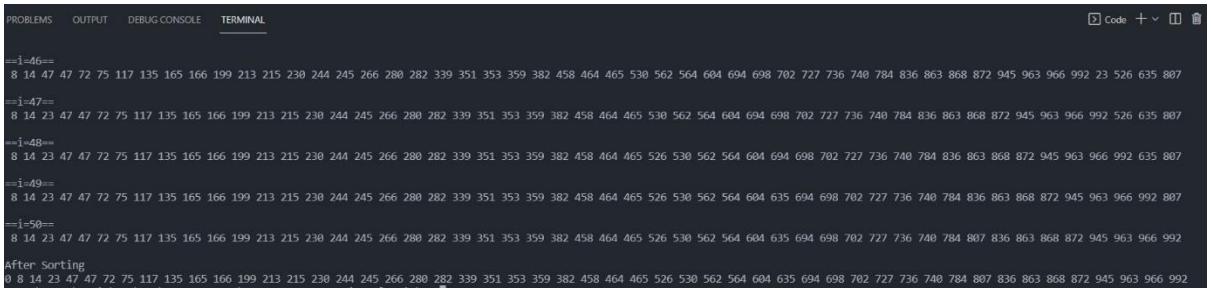
ข้อดี

- ในกรณีที่ข้อมูลนั้นถูกจัดเรียงมาค่อนข้างที่จะสมบูรณ์แล้ว การใช้ Insertion Sort นั้นสามารถทำงานได้อย่างมีประสิทธิภาพ

- ถ้าจะต้องการหาค่าที่น้อยที่สุดในชุดข้อมูล Insertion sort นั้นเป็นตัวเลือกที่ดีหนึ่งตัวเลือก

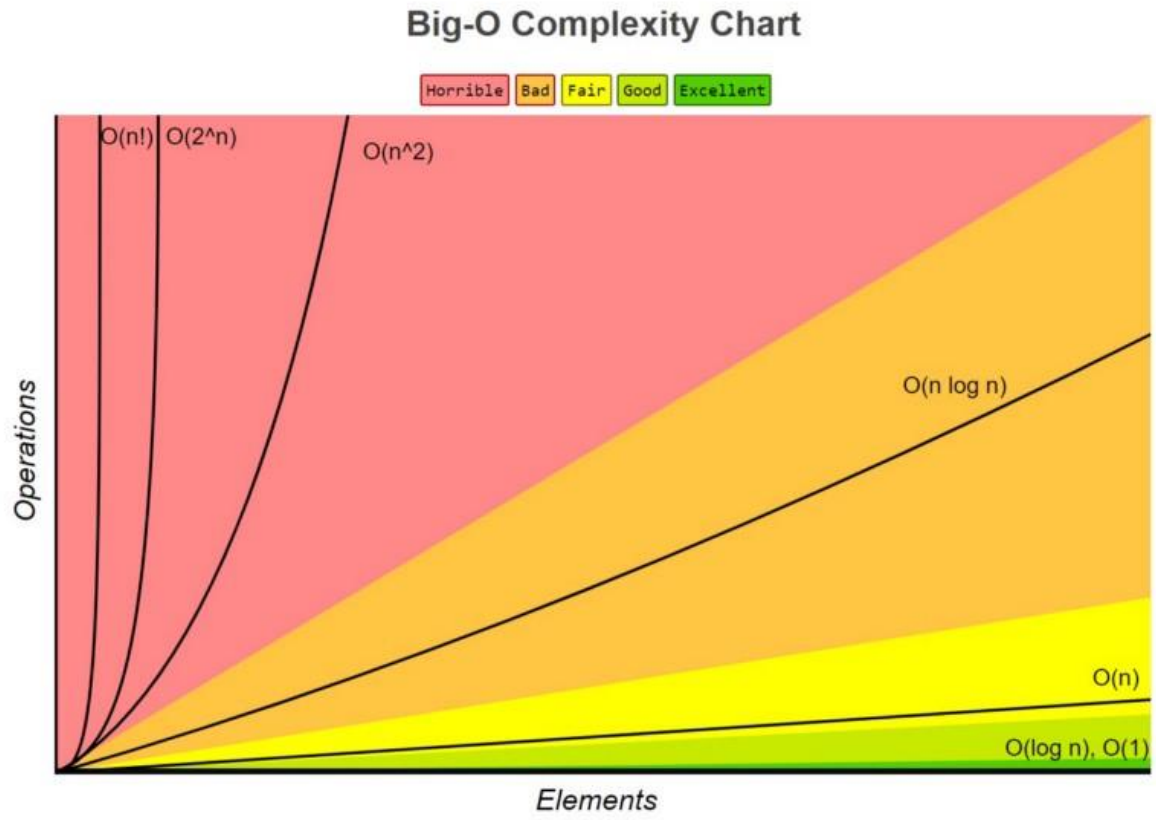
ข้อเสีย

- ไม่มี



จากการศึกษาเพิ่มเติม ทำให้ทราบว่าประสิทธิภาพการจัดเรียงของแต่ละอัลกอริทึมจะแทนด้วยสัญลักษณ์บิ๊กโอ(O) ที่ย่อมาจาก Big-O Notation หรือ Order of Magnitude ซึ่งก็คือฟังก์ชันที่ได้มาจากการประมาณค่าทางคณิตศาสตร์เพื่อพิจารณาอัตราการเติบโตของฟังก์ชัน เช่น ประสิทธิภาพการจัดเรียงที่ได้คือ O(n) ย่อมดีกว่า $O(n^2)$ เป็นต้น ซึ่งตัว n หมายถึงจำนวนข้อมูล หาก n มีปริมาณที่มากจะส่งผลทำให้ฟังก์ชันนั้นทำงานได้อย่างไม่มีประสิทธิภาพ และอาจถูกจัดอยู่ใน Worst Case ซึ่งจากผลการทดลองที่ได้มานั้น ได้มาจากข้อมูลที่ถูกทดสอบโดยที่มีปริมาณที่เยอะและน้อยมากที่สุด โดยในกรณีที่แย่ที่สุด ดีที่สุด และค่าเฉลี่ยของ Sort แต่ละตัวนั้น ถูกจัดแบ่งออกมาได้ดังนี้

Sorting	Best	Average	Worst
Bubble Sort	O(n)	$O(n^2)$	$O(n^2)$
Selection Sort	$O(n^2)$	$O(n^2)$	$O(n^2)$
Insertion Sort	O(n)	$O(n^2)$	$O(n^2)$



ซึ่งกล่าวคือ

1. ในกรณีที่ดีที่สุดของ Bubble Sort คือ O(n) เนื่องจากหากข้อมูลถูกจัดมาแล้ว จะทำให้ช่วยลดระยะเวลาในการทำงานมากขึ้น ส่วนในกรณีแย่สุดคือ $O(n^2)$ เนื่องจากหากข้อมูลยังไม่ถูกจัดเรียง จะทำให้เสียเวลาในการเปรียบเทียบในแต่ละคู่
2. ในกรณีที่ดีที่สุดและแย่ที่สุดของ Selection Sort คือ $O(n^2)$ เนื่องจากไม่ว่าข้อมูลจะถูกจัดมาแล้วหรือยังไม่ถูกจัดมา ยังไงก็ต้องตรวจสอบทุกคู่ก่อนอยู่ดี
3. ในกรณีที่ดีที่สุดของ Insertion Sort คือ O(n) เนื่องจากหากข้อมูลที่ถูกป้อนเข้ามาถูกจัดเรียงอยู่แล้วจะทำให้การทำงานรวดเร็วมากขึ้น ส่วนในกรณีที่แย่ที่สุดคือในกรณีที่ข้อมูลยังไม่ถูกจัดเรียง ซึ่งหากเป็นเช่นนั้นจะทำให้อัลกอริทึมทำงานในลักษณะลูปซ้อนชนิดกำลังสอง ซึ่งทำให้ระยะการทำงานช้าลง

ชุดข้อมูลที่ใช้ทดลอง

1. (694 166 458 863 213 872 604 359 564 698 47 117 465 47 75 339 245 14 992 464 135 945 165 282 280 244 784 382 868 199 72 736 727 353 215 530 836 966 266 8 963 740 230 351 702 562 23 526 635 807)
2. (5 6 9 7 6 5 4 1 0 1 5)

อ้างอิงแหล่งที่มาจากเว็บไซต์และหนังสือ

1. https://visualgo.net/en/sorting
2. https://www.freecodecamp.org/news/all-you-need-to-know-about-big-o-notation-to-crack-your-next-coding-interview-9d575e7eec4
3. โครงสร้างข้อมูล โดย โอภาส เอี่ยมสิริวงศ์