
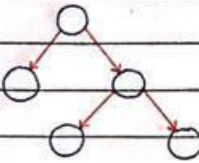
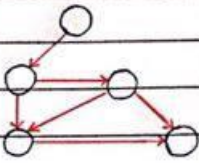


คำถามที่ ๑

- 1) โครงสร้างของแบบแผนกราฟ ทวี และกราฟ ทวิส่วนต่างอย่างไร
- 2) โครงสร้างของแบบแผนกราฟที่มีอยู่ในรูปอย่างไร
- 3) แนวคิดพื้นฐานเกี่ยวกับกราฟสองส่วน
- 4) กราฟแบบมีทิศทาง คือ
- 5) กราฟแบบไม่มีทิศทาง คือ
- 6) กราฟแบบมีทิศทางต่างจากกราฟแบบไม่มีทิศทางอย่างไร
- 7) กราฟที่หาเส้นทางที่สั้นที่สุดของกราฟแบบกราฟที่หาเส้นทางที่สั้นที่สุดของกราฟ
- 8) เส้นทาง (Path) คือ
- 9) วงจรที่สั้นที่สุด คือ
- 10) Connected ส่วนหนึ่งของกราฟ
- 11) การเชื่อมต่อแบบทวิส่วน คือ
- 12) การเชื่อมต่อแบบทวิส่วน คือ
- 13) ไม่มีการเชื่อมต่อ คือ
- 14) ดีกรี (Degree) คือ
- 15) การดำเนินการของกราฟที่มีทิศทาง อย่างไร
- 16) กราฟทวิส่วนที่เล็ก คือ
- 17) กราฟทวิส่วนที่เล็กที่มีทิศทางอย่างไร
- 18) กราฟทวิส่วนที่เล็ก คือ
- 19) กราฟที่มีทิศทาง คือ
- 20) กราฟที่มีทิศทาง คือ
- 21) กราฟที่มีทิศทางที่เล็ก คือ
- 22) กราฟที่มีทิศทางที่เล็ก
- 23) กราฟที่มีทิศทางที่เล็ก
- 24) กราฟที่มีทิศทางที่เล็ก
- 25) โครงสร้างของกราฟที่มีทิศทางในกราฟที่มีทิศทาง
- 26) กราฟที่มีทิศทาง คือ
- 27) กราฟที่มีทิศทางที่มีทิศทางอย่างไร
- 28) ดีกรีที่เล็ก คือ
- 29) ดีกรีที่เล็ก คือ
- 30) Spanning Tree / Minimum Spanning Tree / Shortest Path คือ

คำถามบทที่ 3

1)

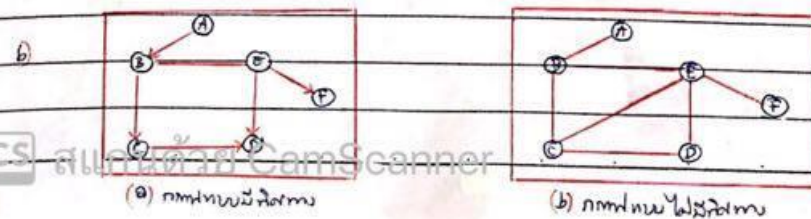
		
(a) Linear List จะมีทั้ง Predecessor และ Successor	(b) Tree จะมีทั้ง Predecessor และ Successor	(c) Graph จะมีทั้ง Predecessor และ Successor

2) โครงสร้างข้อมูลแบบกราฟ (Graph) มีประโยชน์ในหลายๆด้าน เช่น การหาเส้นทางที่สั้นที่สุด การหาความสัมพันธ์ระหว่างสิ่งต่างๆ การจัดการข้อมูลที่มีโครงสร้างที่ซับซ้อน เช่น การออกแบบระบบคอมพิวเตอร์ การจัดการข้อมูลที่มีโครงสร้างที่ซับซ้อน เช่น การออกแบบระบบคอมพิวเตอร์

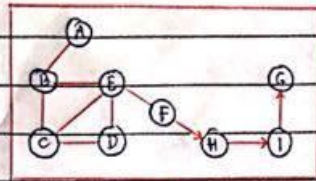
3) แนวคิดพื้นฐานเกี่ยวกับกราฟ (Basic concepts) กราฟ (Graph) คือ เซตของโหนดที่เรียกว่า โหนด (Vertex/Vertices) และเซตของเส้นเชื่อมที่เรียกว่า เส้น (Edges) ซึ่งโหนดและเส้นเชื่อมจะประกอบกันเป็นกราฟ

4) กราฟแบบมีทิศทาง (Directed Graph) กราฟที่มีทิศทางเรียกว่า กราฟ Digraph หรือ กราฟที่มีทิศทาง ซึ่งโหนดและเส้นเชื่อมจะประกอบกันเป็นกราฟที่มีทิศทาง

5) กราฟแบบไม่มีทิศทาง (Undirected Graph) กราฟที่ไม่มีทิศทางเรียกว่า กราฟ Simple Graph หรือ กราฟที่ไม่มีทิศทาง ซึ่งโหนดและเส้นเชื่อมจะประกอบกันเป็นกราฟที่ไม่มีทิศทาง

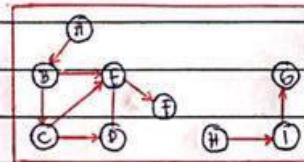


(ข้อ 12)



(b)

13) การพิจารณาความสัมพันธ์ของกราฟที่ไม่มีทิศทางจะเป็นการเชื่อมต่อแบบใดบ้างที่แสดงว่ากราฟนั้นสามารถเชื่อมโยงกันได้ทั้งหมดหรือไม่ (connected/disjoint graph) ซึ่งสามารถพิจารณาได้จากกราฟที่ไม่มีทิศทางที่แสดงในรูป (c)



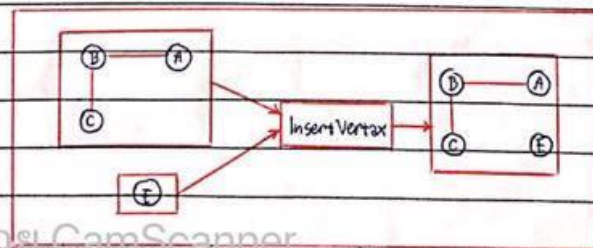
(c)

14) ดีกรี (Degree) ดีกรีของโหนดหรือจุดยอดในกราฟ คือจำนวนของเส้นเชื่อมที่เชื่อมโหนดนั้นกับโหนดอื่นในกราฟ (a) จำนวนดีกรีของโหนด B คือ 3 และดีกรีของโหนด E คือ 4 นอกจากนี้ ดีกรีของโหนด A และโหนด G เป็น 1 (b) ดีกรี (Outdegree) และอินดีกรี (Indegree) ของโหนดในกราฟ คือจำนวนของเส้นเชื่อมที่เข้าและออกจากโหนดนั้น ในกรณีนี้ อินดีกรีของโหนด A คือ 1 และอินดีกรีของโหนด B คือ 2 ส่วนอินดีกรีของโหนด E คือ 3 และอินดีกรีของโหนด G คือ 1

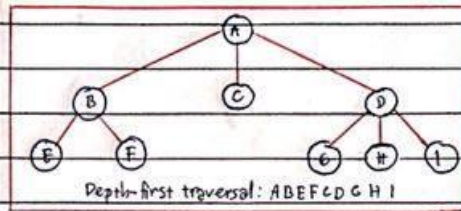
15) การดำเนินการของกราฟ (Graph Operations) เป็นการดำเนินการกับกราฟที่มีลักษณะเป็น 2 มิติ ซึ่งสามารถใช้ดำเนินการกับกราฟได้หลายอย่าง เช่น การเพิ่มโหนด การลบโหนด การเพิ่มเส้นเชื่อม การลบเส้นเชื่อม และการดำเนินการอื่นๆที่เกี่ยวข้องกับกราฟ

16) การเพิ่มโหนด (Insert Vertex) การเพิ่มโหนดคือการเพิ่มโหนดใหม่ลงในกราฟ โดยไม่จำเป็นต้องเพิ่มเส้นเชื่อมใดๆให้กับโหนดนั้น ซึ่งจะทำให้กราฟมีขนาดเพิ่มขึ้น และอาจส่งผลต่อคุณสมบัติของกราฟได้

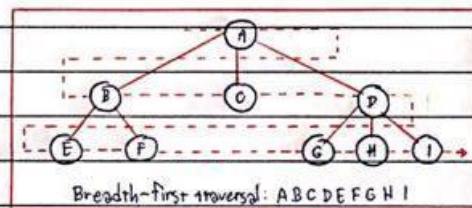
17)



23) การท่องเที่ยวไปในกราฟแบบแนวลึก (Depth-First Traversal) เป็นวิธีท่องเที่ยวไปในกราฟที่วนการประมวลผลทาง ฮอร์เท็กซ์ในแนวตั้งตามทิศทางของฮอร์เท็กซ์นั้นก่อน แล้วจึงค่อยพัวไปพัวฮอร์เท็กซ์ปริมิตที่ต่อเนื่องกันต่อไป



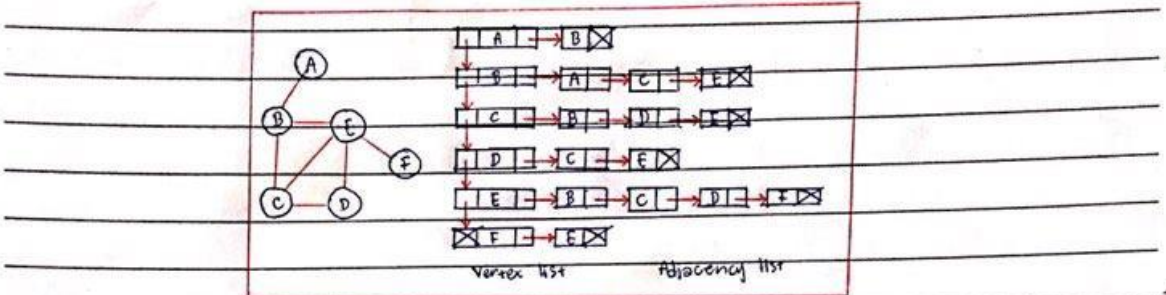
24) การท่องเที่ยวไปในกราฟแบบแนวกว้าง (Breadth-First Traversal) วิธีท่องเที่ยวไปในกราฟแบบแนวกว้างจะวนการประมวลฮอร์เท็กซ์ที่เป็นฮอร์เท็กซ์ปริมิตที่ต่อเนื่องกันก่อนที่จะวนในฮอร์เท็กซ์ถัดไป คือ การพัวตัววิธีนี้ฮอร์เท็กซ์ปริมิตที่ต่อเนื่องกันจะวนการประมวลผลจนกว่าจะฮอร์เท็กซ์ปริมิต



25) โครงสร้างการเก็บข้อมูลในกราฟ (Graph Storage Structures) การแทนที่กราฟ เพื่อใช้ในการจัดเก็บข้อมูลของ 2 กลุ่มตัวแปร กลุ่มแรกคือกราฟแบบแนวลึกหรือแนวกว้าง และกลุ่มที่สองคือกราฟแบบแนวลึกหรือแนวกว้าง ซึ่งโดยทั่วไปแล้วสองกลุ่มนี้จะใช้วิธีการเก็บข้อมูลที่แตกต่างกัน กล่าวคือถ้าเราใช้วิธีแนวลึกหรือแนวกว้างในการประมวลผลกราฟแนวลึกหรือแนวกว้าง จะมีการใช้วิธีเก็บข้อมูลของฮอร์เท็กซ์ที่ต่อเนื่องกัน

26) แมทริกซ์ประชิด (Adjacency Matrix) แมทริกซ์ประชิดจะใช้จากเมอร์ อารีเฟนเป็นต้นมา สำหรับเก็บฮอร์เท็กซ์และใช้แมทริกซ์ อารีเฟนเป็นต้นมา ซึ่งถ้ามีฮอร์เท็กซ์ต่อเนื่องกันจะแสดงด้วยเลข 1 และถ้าไม่มีฮอร์เท็กซ์ต่อเนื่องกันจะแสดงด้วยเลข 0

27) สำหรับข้อจำกัดของกราฟแบบแนวลึกหรือแนวกว้าง จะมีการใช้วิธีเก็บข้อมูลของฮอร์เท็กซ์ที่ต่อเนื่องกัน ซึ่งถ้ามีฮอร์เท็กซ์ต่อเนื่องกันจะแสดงด้วยเลข 1 และถ้าไม่มีฮอร์เท็กซ์ต่อเนื่องกันจะแสดงด้วยเลข 0

[illegible][illegible]

30) Spanning Tree คือ กราฟที่ประกอบด้วยโหนดทั้งหมดในกราฟ โดยที่ Minimum Spanning Tree คือ กราฟที่เชื่อมต่อโหนดทั้งหมดในกราฟด้วยเส้นเชื่อมที่มีน้ำหนักน้อยที่สุด และเป็นกราฟที่ไม่มีวัฏจักร (acyclic) และไม่มีเส้นเชื่อมซ้ำ (simple graph) Shortest Path คือ เป็นกราฟที่หาเส้นทางจากโหนดหนึ่งไปยังโหนดหนึ่งด้วยเส้นทางที่มีน้ำหนักน้อยที่สุด

สรุปทฤษฎีกราฟ

• กราฟ คือ กลุ่มของโหนดที่เรียกว่า **โหนด (Vertex)** ที่เชื่อมโหนดต่างๆ ในกราฟด้วย **เส้นเชื่อม (Edges)** หรือ **อาร์ค (Arcs)**

• กราฟแบบไม่มีทิศทาง หรือ **โหนดกราฟ** จะมีทิศทางที่ชี้ไปรอบทิศทาง

• กราฟแบบไม่มีทิศทาง เส้นที่เชื่อมโหนดต่างๆ เรียกว่า **เส้นเชื่อม** ที่เชื่อมโหนดต่างๆ ไปมา

• **ทิศทาง** คือ เส้นที่ชี้ไปรอบทิศทาง

• **โหนด** คือ โหนดที่เชื่อมโหนดต่างๆ ในกราฟแบบไม่มีทิศทาง

• **โหนดที่ติดกัน (Adjacent Vertex)** หมายถึง โหนดที่เชื่อมโหนดที่ติดกัน

• **เส้นทาง (Path)** คือ เส้นทางที่เชื่อมโหนดต่างๆ ในกราฟแบบไม่มีทิศทาง

• **วัฏจักร (Cycle)** คือ เส้นทางที่เชื่อมโหนดต่างๆ ในกราฟแบบไม่มีทิศทาง

• **ลูป (Loop)** คือ เส้นทางที่เชื่อมโหนดต่างๆ ในกราฟแบบไม่มีทิศทาง

• กราฟที่เป็นลักษณะต่อเนื่อง (**Connected**) หมายถึง โหนดต่างๆ ในกราฟจะเชื่อมโหนดต่างๆ กันโดยไม่มีจุดตัดหรือโหนดที่แยกจากกัน

• **ดีกรี (Degree)** ของโหนด คือ จำนวนของโหนดที่เชื่อมโหนดที่ติดกัน

• **ดีกรีของโหนด** คือ จำนวนของโหนดที่เชื่อมโหนดที่ติดกัน

• **ดีกรีของโหนด** คือ จำนวนของโหนดที่เชื่อมโหนดที่ติดกัน

• **ดีกรีของโหนด** คือ จำนวนของโหนดที่เชื่อมโหนดที่ติดกัน

• **ดีกรีของโหนด** คือ จำนวนของโหนดที่เชื่อมโหนดที่ติดกัน

• **ดีกรีของโหนด** คือ จำนวนของโหนดที่เชื่อมโหนดที่ติดกัน

• **ดีกรีของโหนด** คือ จำนวนของโหนดที่เชื่อมโหนดที่ติดกัน

• **ดีกรีของโหนด** คือ จำนวนของโหนดที่เชื่อมโหนดที่ติดกัน

အိတ်မက်စ် ၅

- ၁) အိတ်မက်စ် မြေပုံဆွဲရာ Sorting ခံ့
- ၂) Sort Concepts ခံ့
- ၃) Internal Sorting ခံ့
- ၄) External Sorting ခံ့
- ၅) Sort Order ခံ့
- ၆) Sort Stability ခံ့
- ၇) Sort Efficiency ခံ့
- ၈) အိတ်မက်စ် မြေပုံဆွဲရာ အိတ်မက်စ် ခံ့
- ၉) Selection Sort ခံ့
- ၁၀) Heap Sort ခံ့
- ၁၁) Insertion Sort ခံ့
- ၁၂) Bubble Sort ခံ့
- ၁၃) Quick Sort ခံ့
- ၁၄) Quick Sort မြေပုံဆွဲရာ အိတ်မက်စ်
- ၁၅) အိတ်မက်စ် မြေပုံဆွဲရာ Quick Sort အိတ်မက်စ်
- ၁၆) အိတ်မက်စ် မြေပုံဆွဲရာ Quick Sort ခံ့
- ၁၇) အိတ်မက်စ် မြေပုံဆွဲရာ Bubble Sort
- ၁၈) အိတ်မက်စ် မြေပုံဆွဲရာ Insertion Sort
- ၁၉) Merge Sort ခံ့
- ၂၀) အိတ်မက်စ် မြေပုံဆွဲရာ အိတ်မက်စ်
- ၂၁) အိတ်မက်စ် မြေပုံဆွဲရာ အိတ်မက်စ် အိတ်မက်စ် အိတ်မက်စ် ခံ့
- ၂၂) Sort Phase အိတ်မက်စ်
- ၂၃) အိတ်မက်စ် မြေပုံဆွဲရာ sortleft နှင့် sortright အိတ်မက်စ်
- ၂၄) အိတ်မက်စ် ၂၃ အိတ်မက်စ် ၂ အိတ်မက်စ်
- ၂၅) Insertion အိတ်မက်စ်
- ၂၆) အိတ်မက်စ် Insertion နှင့် Bubble အိတ်မက်စ်
- ၂၇) အိတ်မက်စ် မြေပုံဆွဲရာ အိတ်မက်စ်
- ၂၈) အိတ်မက်စ် မြေပုံဆွဲရာ selection နှင့် Heap အိတ်မက်စ်
- ၂၉) အိတ်မက်စ် မြေပုံဆွဲရာ အိတ်မက်စ်
- ၃၀) အိတ်မက်စ် မြေပုံဆွဲရာ Merge အိတ်မက်စ်

5

-10

-15

-20-

-25

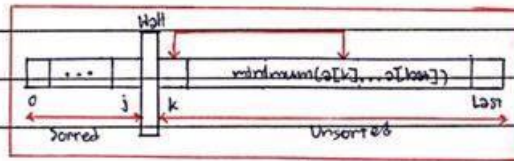
30

7) ประสิทธิภาพของการเรียงลำดับเป็นวิธีที่ดัดแปลงวิธีการเรียงลำดับของอัลกอริทึมที่ง่ายและสามารถดัดแปลงได้
งานวิจัยเกี่ยวกับการปรับปรุงประสิทธิภาพของอัลกอริทึมการเรียงลำดับที่มีประสิทธิภาพโดยทั่วไปนี้
จากขั้นตอนการทำการเรียงลำดับของวิธีการเรียงลำดับแบบใหม่ในขั้นต้น ซึ่งประสิทธิภาพการเรียงลำดับของอัลกอริทึม
จะเหมาะสมกับวิธีการที่ง่าย เช่น ประสิทธิภาพการเรียงลำดับที่ $O(n)$ ต่อมาคือ $O(n^2)$ เป็นต้น

8) วิธีการเรียงลำดับ มีอยู่หลายวิธีที่ต่างกันซึ่งสามารถนำมาใช้เพื่อทำการเรียงลำดับข้อมูล ซึ่งแต่ละวิธีก็จะมี
ลักษณะการทำงานที่ต่างกันออกไปทั้งในเรื่อง ขั้นตอนการทำงาน วิธีการเรียงลำดับของข้อมูล วิธีการทำงาน ซึ่งต่อไปนี้เป็น

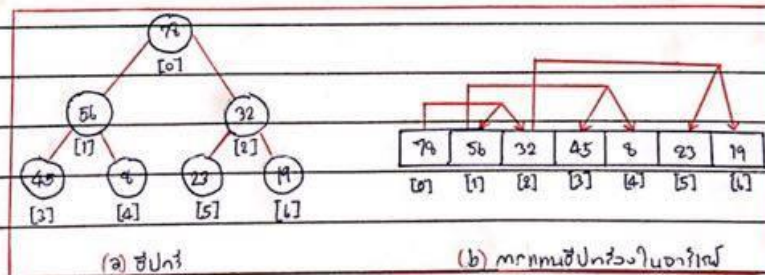
(1) Selection Sort (2) Heap Sort (3) Insertion Sort (4) Bubble Sort (5) Quick Sort (6) Merge Sort

9) Selection Sort เป็นวิธีการเรียงลำดับที่มีขั้นตอนการทำงานที่ง่ายและสามารถนำมาใช้เพื่อทำการเรียงลำดับข้อมูล
วิธีหนึ่งที่มีอยู่ในลักษณะการทำงานที่ง่ายและสามารถนำมาใช้เพื่อทำการเรียงลำดับข้อมูล



วิธีการเรียงลำดับ Selection

10) Heap Sort เป็นวิธีการเรียงลำดับที่มีขั้นตอนการทำงานที่ง่ายและสามารถนำมาใช้เพื่อทำการเรียงลำดับข้อมูล
การที่ Max-Heap หรือมีค่าของค่ารากในต้นไม้ Min-Heap ซึ่งในขั้นต้นจะมีความหมายว่าวิธีการเรียงลำดับ
จากประเภทที่ซับซ้อนขึ้นหรือมีการทำงานที่ง่ายและสามารถนำมาใช้เพื่อทำการเรียงลำดับข้อมูล

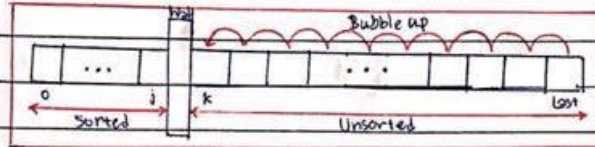


(a) ต้นไม้

(b) การแทนต้นไม้วัดในอาร์เรย์

11) Insertion Sort เป็นวิธีการเรียงลำดับที่มีขั้นตอนการทำงานที่ง่ายและสามารถนำมาใช้เพื่อทำการเรียงลำดับข้อมูล
การที่ Max-Heap หรือมีค่าของค่ารากในต้นไม้ Min-Heap ซึ่งในขั้นต้นจะมีความหมายว่าวิธีการเรียงลำดับ
จากประเภทที่ซับซ้อนขึ้นหรือมีการทำงานที่ง่ายและสามารถนำมาใช้เพื่อทำการเรียงลำดับข้อมูล

12) Bubble Sort หรือวิธีเรียงสับเปลี่ยนฟอง เป็น ขั้นตอนแรกคือ ทดสอบการสลับที่ของสมาชิกที่อยู่ติดกัน และถ้าจำเป็นก็สลับที่กันจนกว่าจะเรียงตัวถูกต้อง



วิธีเรียงสับเปลี่ยนฟอง (Bubble Sort)

5

13) Quick Sort มีวิธีการเหมือน Bubble Sort แต่ Quick Sort ใช้วิธีการสลับที่ของสมาชิกที่อยู่ติดกัน (Exchange) เพื่อเป็นการลดการสลับที่ลงได้ วิธีการของ Quick Sort จะเรียงสับเปลี่ยนฟองที่อยู่ติดกันและสลับที่กันจนกว่าจะเรียงตัวถูกต้อง

10

14) ทดสอบการสับเปลี่ยนของ Quick Sort จะมีการเลือกสมาชิกหนึ่งตัวที่เรียกว่า "Pivot" เพื่อใช้เป็นค่าแบ่งส่วนข้อมูล และวิธีของ Quick Sort จะใช้วิธีการสับเปลี่ยนฟอง (Exchange) เพื่อเป็นการลดการสลับที่ลงได้

15

- 1) ส่วนที่หนึ่ง คือ กลุ่มสมาชิกที่น้อยกว่า Pivot key
- 2) ส่วนที่สอง คือ Pivot key ที่ใช้แบ่งข้อมูล
- 3) ส่วนที่สาม คือ กลุ่มสมาชิกที่มากกว่าหรือเท่ากับ Pivot key

15) วิธีการของ Quick Sort จะใช้วิธีการสับเปลี่ยนฟอง (Exchange) เพื่อเป็นการลดการสลับที่ลงได้ และวิธีการของ Quick Sort จะใช้วิธีการสับเปลี่ยนฟอง (Exchange) เพื่อเป็นการลดการสลับที่ลงได้

20

16) การเรียงสับเปลี่ยนฟอง (Bubble Sort) เป็นวิธีการเรียงสับเปลี่ยนฟอง (Bubble Sort) ที่ใช้วิธีการสับเปลี่ยนฟอง (Exchange) เพื่อเป็นการลดการสลับที่ลงได้

25

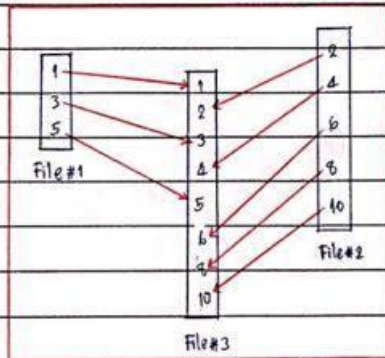
17) วิธีการเรียงสับเปลี่ยนฟอง (Bubble Sort) เป็นวิธีการเรียงสับเปลี่ยนฟอง (Bubble Sort) ที่ใช้วิธีการสับเปลี่ยนฟอง (Exchange) เพื่อเป็นการลดการสลับที่ลงได้

30

การเลือกวิธีเรียงสับเปลี่ยนของ Insertion เป็นวิธีที่เหมาะสมที่สุดสำหรับข้อมูลที่ไม่เป็นระเบียบ
ในขั้นตอนการเลือกวิธีเรียงสับเปลี่ยนจะขึ้นอยู่กับขนาดของข้อมูลที่จะเรียง ถ้าข้อมูลมีขนาดเล็ก
การเลือกวิธีเรียงสับเปลี่ยนที่เหมาะสมที่สุดคือ Insertion Sort ซึ่งมีขั้นตอนการทำงานดังนี้

1) Merge Sort คือวิธีการเรียงสับเปลี่ยนที่แบ่งข้อมูลออกเป็นครึ่งๆ ซ้ำๆ จนเหลือข้อมูลเพียง 1 ชิ้น
การเลือกวิธีเรียงสับเปลี่ยนที่เหมาะสมที่สุดคือ Merge Sort ซึ่งมีขั้นตอนการทำงานดังนี้
1. แบ่งข้อมูลออกเป็นครึ่งๆ ซ้ำๆ จนเหลือข้อมูลเพียง 1 ชิ้น
2. เปรียบเทียบข้อมูลทั้งสองครึ่ง และจัดเรียงข้อมูลใหม่
3. ทำซ้ำขั้นตอนที่ 1 และ 2 จนกว่าข้อมูลทั้งหมดจะถูกจัดเรียงเรียบร้อย
4. นำข้อมูลทั้งหมดมาเรียงกันใหม่
5. ทำซ้ำขั้นตอนที่ 1 และ 2 จนกว่าข้อมูลทั้งหมดจะถูกจัดเรียงเรียบร้อย

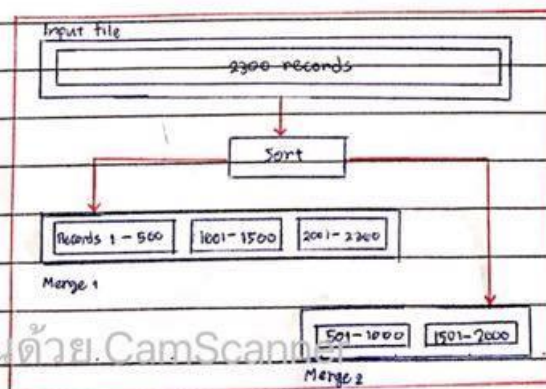
20)



21)

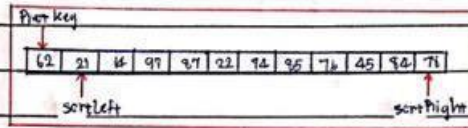
วิธีการเรียง	กรณีดีที่สุด (Best-case)	กรณีเฉลี่ย (Average-case)	กรณีเลวที่สุด (Worst-case)
Selection Sort	$O(n^2)$	$O(n^2)$	$O(n^2)$
Insertion Sort	$O(n)$	$O(n^2)$	$O(n^2)$
Bubble Sort	$O(n)$	$O(n^2)$	$O(n^2)$
Quick Sort	$O(n \log n)$	$O(n \log n)$	$O(n^2)$
Merge Sort	$O(n \log n)$	$O(n \log n)$	$O(n \log n)$

22)



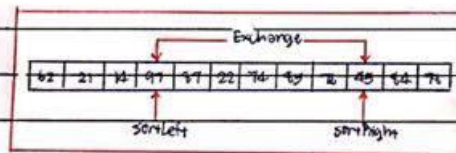
23) การเลือกตัวเลขมา sortLeft และ sortRight

- sortLeft จะเลือกตัวเลขมา sortLeft ที่น้อยกว่า Pivot key
- sortRight จะเลือกตัวเลขมา sortRight ที่มากกว่า Pivot key

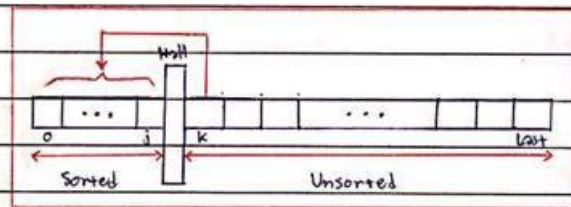


24) การเลือกตัวเลขมา sortLeft และ sortRight

- sortLeft จะเลือกตัวเลขมา sortLeft ที่น้อยกว่า Pivot key
- sortRight จะเลือกตัวเลขมา sortRight ที่มากกว่า Pivot key



25)



การเลือกตัวเลขมา Insertion

26) การเลือกตัวเลขมา sortLeft และ sortRight จะเลือกตัวเลขมา sortLeft ที่น้อยกว่า Pivot key และเลือกตัวเลขมา sortRight ที่มากกว่า Pivot key การเลือกตัวเลขมา sortLeft และ sortRight จะเลือกตัวเลขมา sortLeft ที่น้อยกว่า Pivot key และเลือกตัวเลขมา sortRight ที่มากกว่า Pivot key

27)

365 blue	119 purple	119 purple
212 green	212 green	212 blue
476 white	212 yellow	212 green
212 yellow	212 blue	212 yellow
119 purple	365 blue	365 blue
737 green	443 red	443 red
212 blue	567 yellow	567 yellow
443 red	737 green	737 green
567 yellow	976 white	976 white

(a) การเลือกตัวเลขมา sortLeft (b) การเลือกตัวเลขมา sortRight (c) การเลือกตัวเลขมา sortLeft และ sortRight

CS สแกนด้วย CamScanner

n	Number of loops		
	selection	Sort	Heap Sort
25		625	116
100		10,000	664
500		250,000	4,482
1000		1,000,000	9,945
2000		4,000,000	10,969

29) อัลกอริทึมที่ 9 ประสิทธิภาพค่อนข้างดีในแง่ที่มัน มีประสิทธิภาพสูงและถูกกว่า อัลกอริทึมที่ 8 เช่น
หากเราใช้ อัลกอริทึมที่ 9 ในการเรียงลำดับข้อมูลที่มีขนาดใหญ่มากๆ เราจะได้ผลลัพธ์ที่เร็วกว่า
หากเราใช้ อัลกอริทึมที่ 8 ในการเรียงลำดับข้อมูลที่มีขนาดใหญ่มากๆ เราจะได้ผลลัพธ์ที่ช้ากว่า
อีกอย่างหนึ่งที่เราสามารถสังเกตได้ก็คือ อัลกอริทึมที่ 9 มีความซับซ้อนน้อยกว่า

30) ขั้นตอนการเรียงทวน Merge สำหรับประสิทธิภาพการเรียงทวนของ Merge นั้น เนื่องจากขั้นตอน
การรวมข้อมูลเป็นกลุ่มย่อยๆ แล้วจึงค่อยๆ รวมกลุ่มย่อยเหล่านั้นเข้าด้วยกัน ทำให้การเรียงทวนของข้อมูลนั้น
มีประสิทธิภาพการเรียงทวน Merge จึงมีค่าเท่ากับ $O(n \log n)$

