

บทที่ 9-1 Sorting

บทเรียนย่อย

9.1 Sorting Concept

9.2 Insertion Sort Algorithm

9.3 Insertion Sort Implementation

9.4 Selection Sort Algorithm

9.5 Selection Sort Implementation

9.6 Bubble Sort Algorithm

9.7 Bubble Sort Implementation

วัตถุประสงค์

- นิสิตมีความรู้ และความเข้าใจเกี่ยวกับแนวคิดในการเรียงลำดับข้อมูลรูปแบบต่าง ๆ
- นิสิตสามารถเขียนโปรแกรมเพื่อดำเนินการตามแนวคิดในการเรียงลำดับข้อมูลรูปแบบต่าง ๆ
- นิสิตสามารถนำแนวคิดในการเรียงลำดับข้อมูลมาประยุกต์ใช้งานในการพัฒนาโปรแกรม

บทที่ 9-1 Sorting

บทเรียนย่อย

9.1 Sorting Concept

9.2 Insertion Sort Algorithm

9.3 Insertion Sort Implementation

9.4 Selection Sort Algorithm

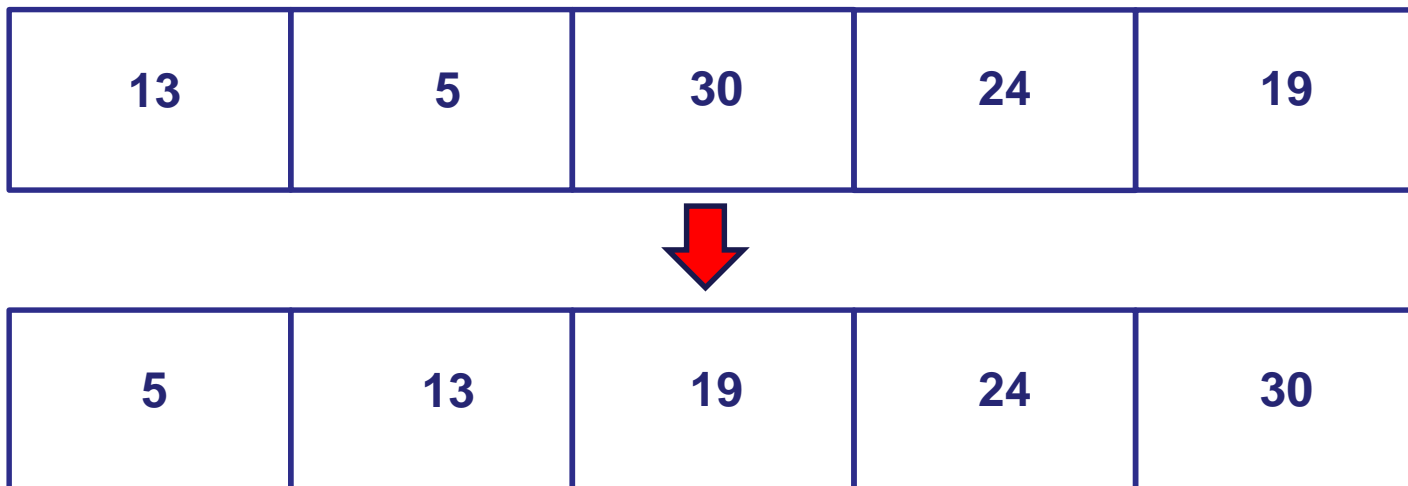
9.5 Selection Sort Implementation

9.6 Bubble Sort Algorithm

9.7 Bubble Sort Implementation

9.1 Sorting Concept

Sorting (การเรียงลำดับข้อมูล) คือ การเรียงลำดับข้อมูลที่จัดเก็บในหน่วยความจำของคอมพิวเตอร์ที่เป็นลักษณะโครงสร้างข้อมูลแบบ Array (อาร์เรย์) และ Pointer (พอยเตอร์) โดยอาจจะเรียงลำดับจากน้อยไปมาก หรือจากมากไปน้อยก็ได้ เช่น การเรียงข้อมูลนักศึกษาจากรหัสนักศึกษาโดยเรียงลำดับจากน้อยไปมาก เป็นต้น



Sort Algorithms (รูปแบบการเรียงลำดับข้อมูล)

รูปแบบการเรียงลำดับข้อมูลที่ได้รับความนิยม มีดังนี้

- Insertion Sort (การเรียงลำดับแบบแทรก)
- Selection Sort (การเรียงลำดับแบบเลือก)
- Bubble Sort (การเรียงลำดับแบบบับเบิล)
- Merge Sort (การเรียงลำดับแบบผสาน)
- Shell Sort (การเรียงลำดับแบบเชลล์)
- Quick Sort (การเรียงลำดับแบบเร็ว)

บทที่ 9-1 Sorting

บทเรียนย่อย

9.1 Sorting Concept

9.2 Insertion Sort Algorithm

9.3 Insertion Sort Implementation

9.4 Selection Sort Algorithm

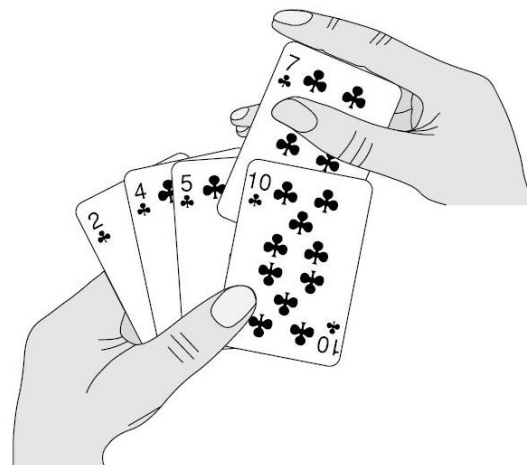
9.5 Selection Sort Implementation

9.6 Bubble Sort Algorithm

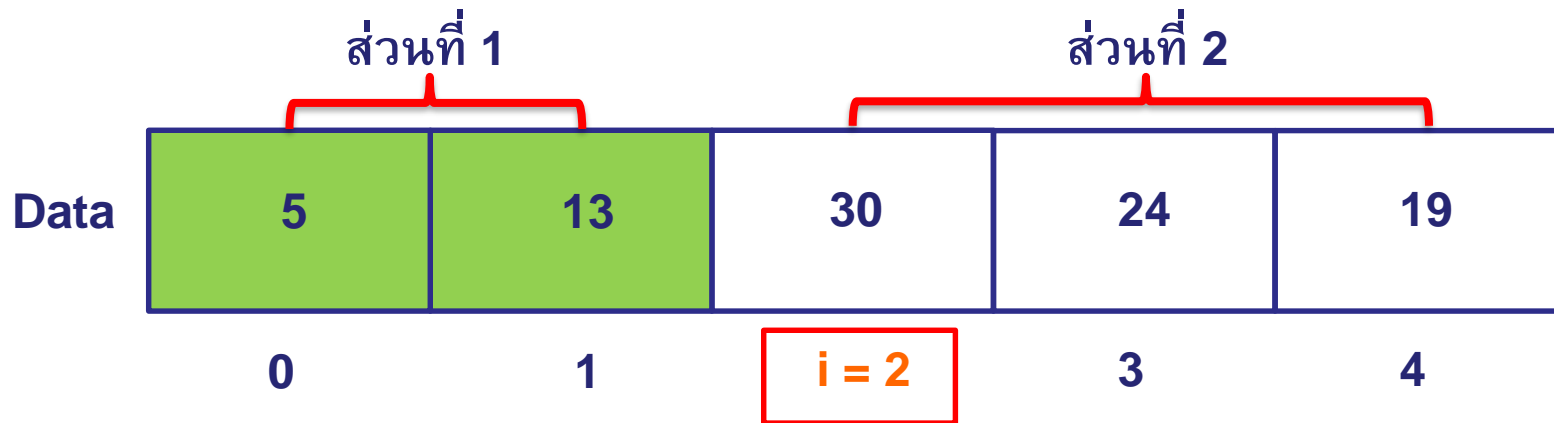
9.7 Bubble Sort Implementation

9.2 Insertion Sort Algorithm

Insertion Sort (การเรียงลำดับแบบแทรก) เป็นการเรียงลำดับแบบง่าย โดยอาศัยเทคนิคมาจากการจัดเรียงไพ่ในมือของผู้เล่น คือ ผู้เล่นจะทำการหยิบไพ่ในกองมาทีละใบ โดยจะเริ่มเปรียบเทียบเมื่อหยิบไพ่ใบที่ 2 ไปจนถึงใบสุดท้าย ถ้าไพ่ใบที่หยิบขึ้นมา น้อยกว่าใบที่อยู่ก่อนหน้าจะนำไปแทรกในตำแหน่งที่ถูกต้อง ซึ่งทำลักษณะนี้ไปเรื่อย ๆ จนไพ่หมดกองจะได้ไพ่ที่เรียงลำดับในมืออย่างถูกต้อง



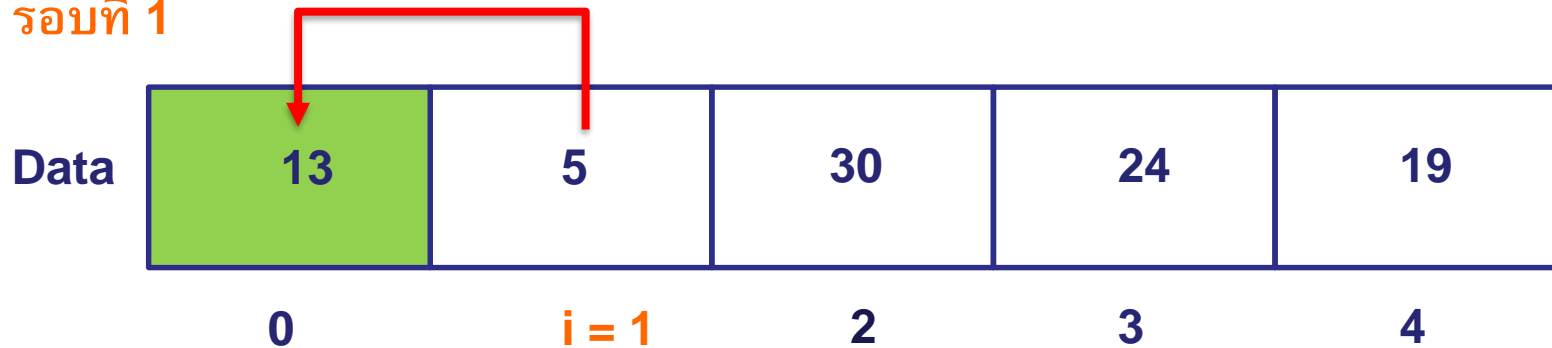
ลักษณะการดำเนินการของ Insertion Sort



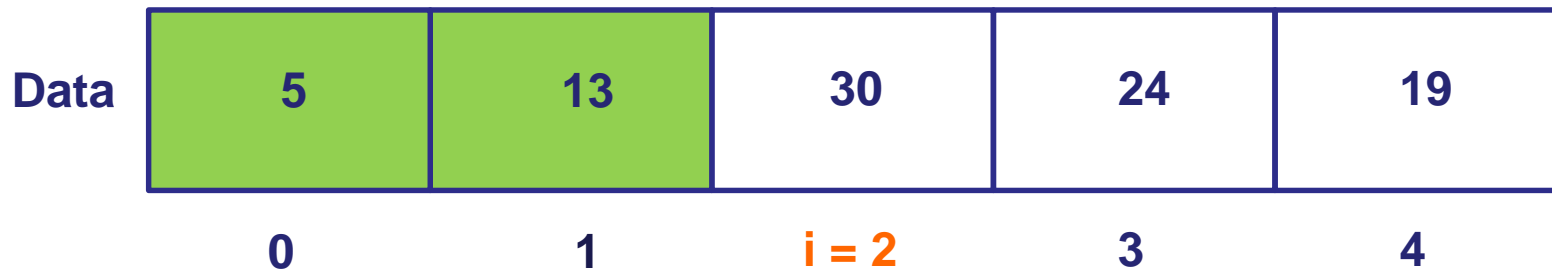
ลักษณะการดำเนินการของ Insertion Sort จะมีการแบ่งส่วนข้อมูลออกเป็น 2 ส่วน โดยส่วนแรก คือ ส่วนที่อยู่ด้านหน้าของตำแหน่งที่ i (ตั้งแต่ข้อมูลตัวแรก ถึง ข้อมูลตัวที่ $i - 1$) ซึ่งเป็นข้อมูลที่ถูกเรียงลำดับแล้ว และอีกส่วนคือข้อมูลตั้งแต่ตำแหน่งที่ i จนถึงข้อมูลตัวสุดท้าย ซึ่งเป็นข้อมูลที่ยังไม่ถูกเรียงลำดับ โดยการจัดเรียงจะนำข้อมูลในตำแหน่งที่ i ไปแทรกในส่วนแรกเพื่อให้ข้อมูลเรียงลำดับอย่างถูกต้อง

ลักษณะการดำเนินการของ Insertion Sort [2]

รอบที่ 1

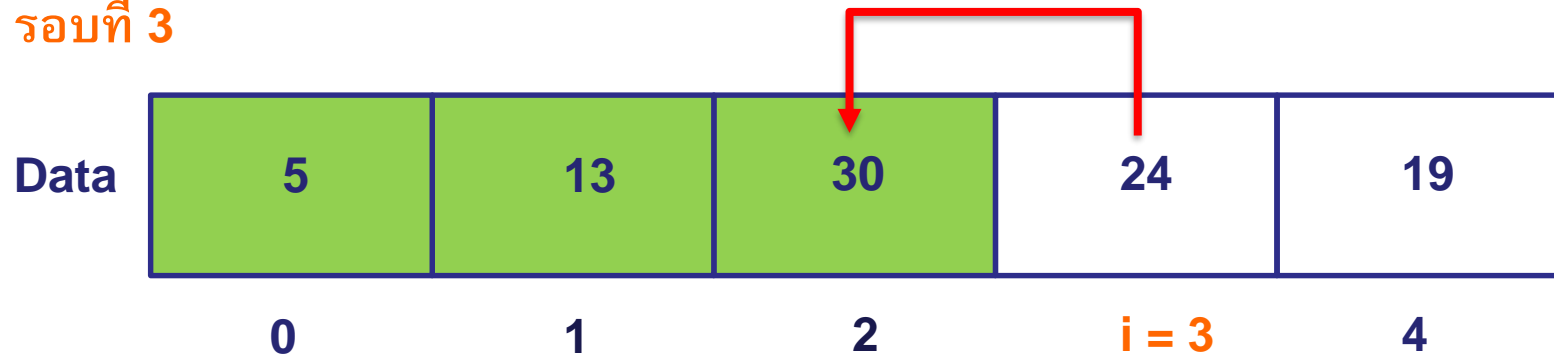


รอบที่ 2

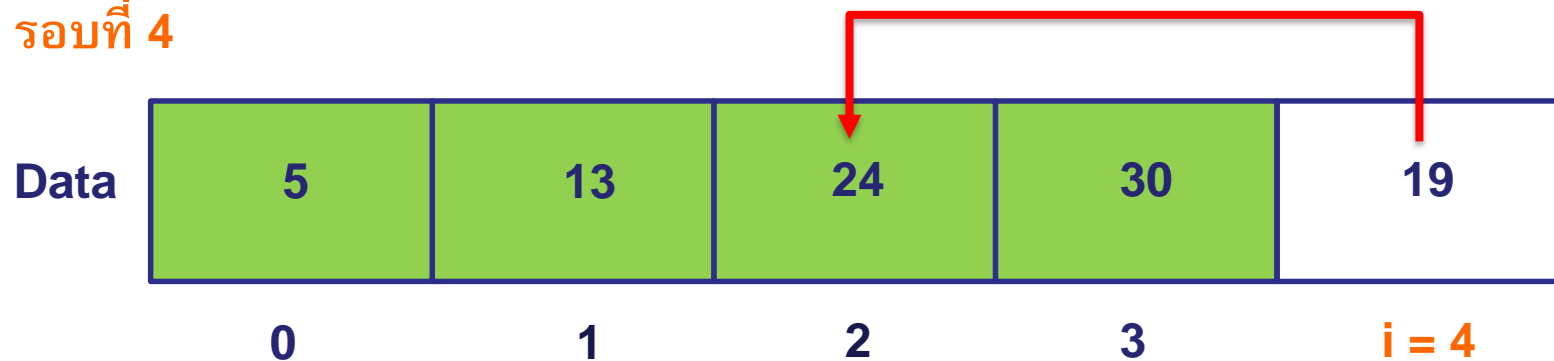


ลักษณะการดำเนินการของ Insertion Sort [3]

รอบที่ 3



รอบที่ 4



ลักษณะการดำเนินการของ Insertion Sort [4]

ผลลัพธ์

Data	5	13	19	24	30
	0	1	2	3	$i = 4$

Time :: Best: $O(n)$ Worst: $O(n^2)$ Average: $O(n^2)$

บทที่ 9-1 Sorting

บทเรียนย่อย

9.1 Sorting Concept

9.2 Insertion Sort Algorithm

9.3 Insertion Sort Implementation

9.4 Selection Sort Algorithm

9.5 Selection Sort Implementation

9.6 Bubble Sort Algorithm

9.7 Bubble Sort Implementation

9.3 Insertion Sort Implementation

Algorithm : Insertion Sort

Pre : data[] (int) , size (int)

Post : ข้อมูลถูกจัดเรียงลำดับจากน้อยไปมาก

1. เริ่มต้น

2. ประกาศตัวแปรชื่อ temp , i และ j เป็นเลขจำนวนเต็ม

3. ทำซ้ำ ตั้งแต่ $i = 1$ จนมีค่าเท่ากับ $size - 1$

- 3.1 กำหนดให้ temp มีค่าเท่ากับ data ตำแหน่งที่ i

- 3.2 กำหนดให้ j มีค่าเท่ากับ $i - 1$

.....

9.3 Insertion Sort Implementation [2]

.....

3.3 ทำซ้ำจนกว่า j มีค่าน้อยกว่า 0 หรือ data ตำแหน่งที่ j จะมีค่าไม่มากกว่า temp

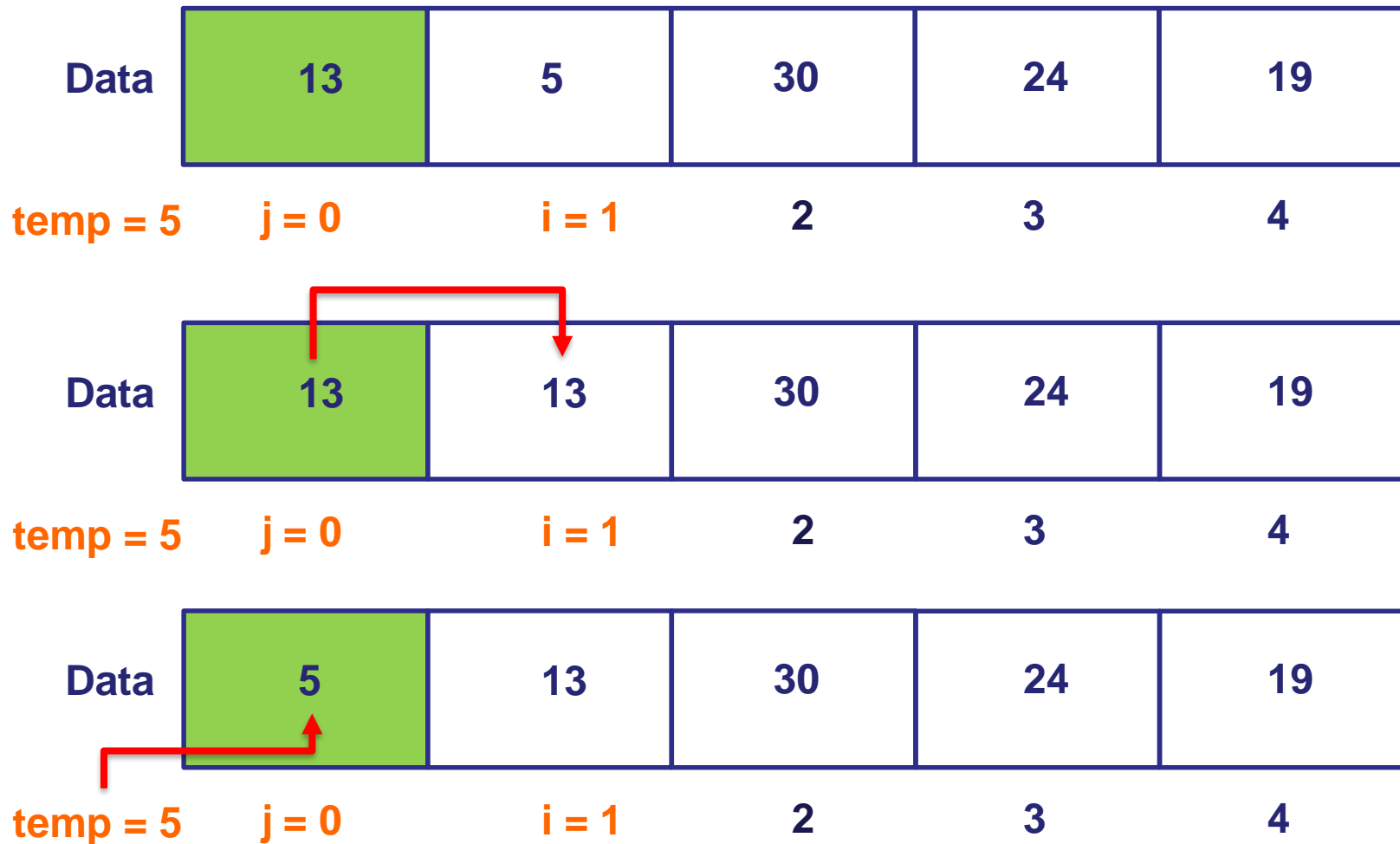
3.3.1 กำหนดให้ data ตำแหน่งที่ $j + 1$ มีค่าเท่ากับ data ตำแหน่งที่ j

3.3.2 กำหนดให้ j มีค่าเท่ากับ $j - 1$

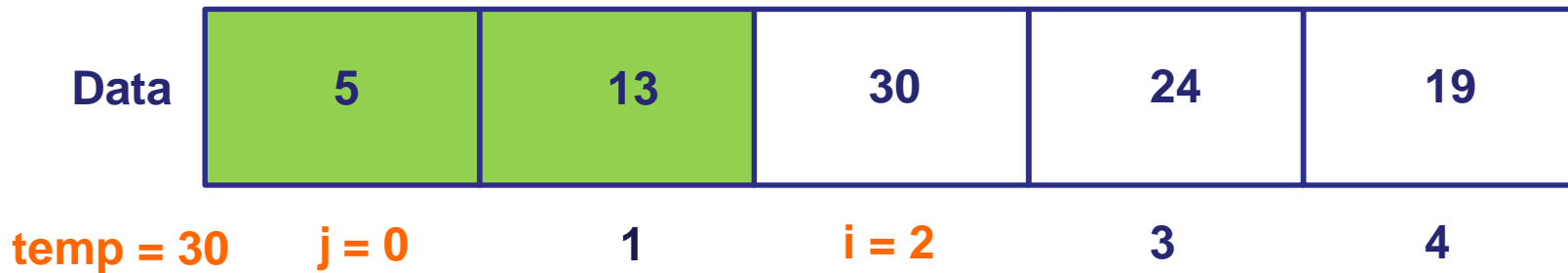
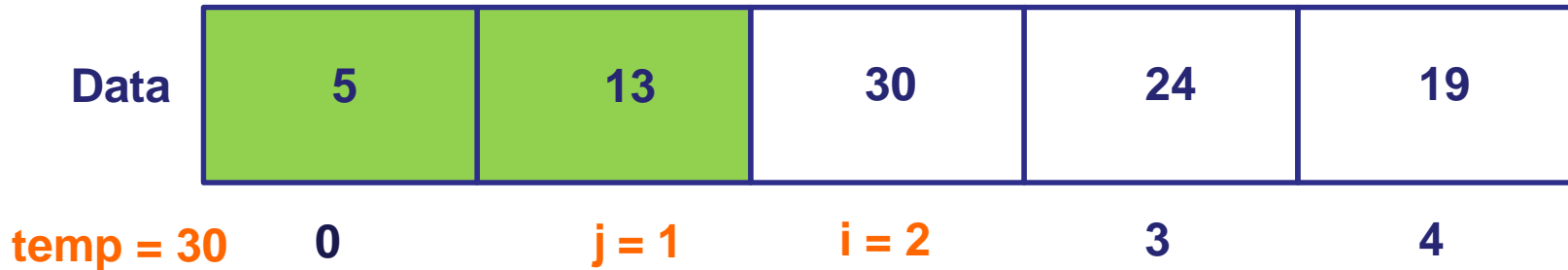
3.4 กำหนดให้ data ตำแหน่งที่ $j + 1$ มีค่าเท่ากับ temp

4. จบการทำงาน

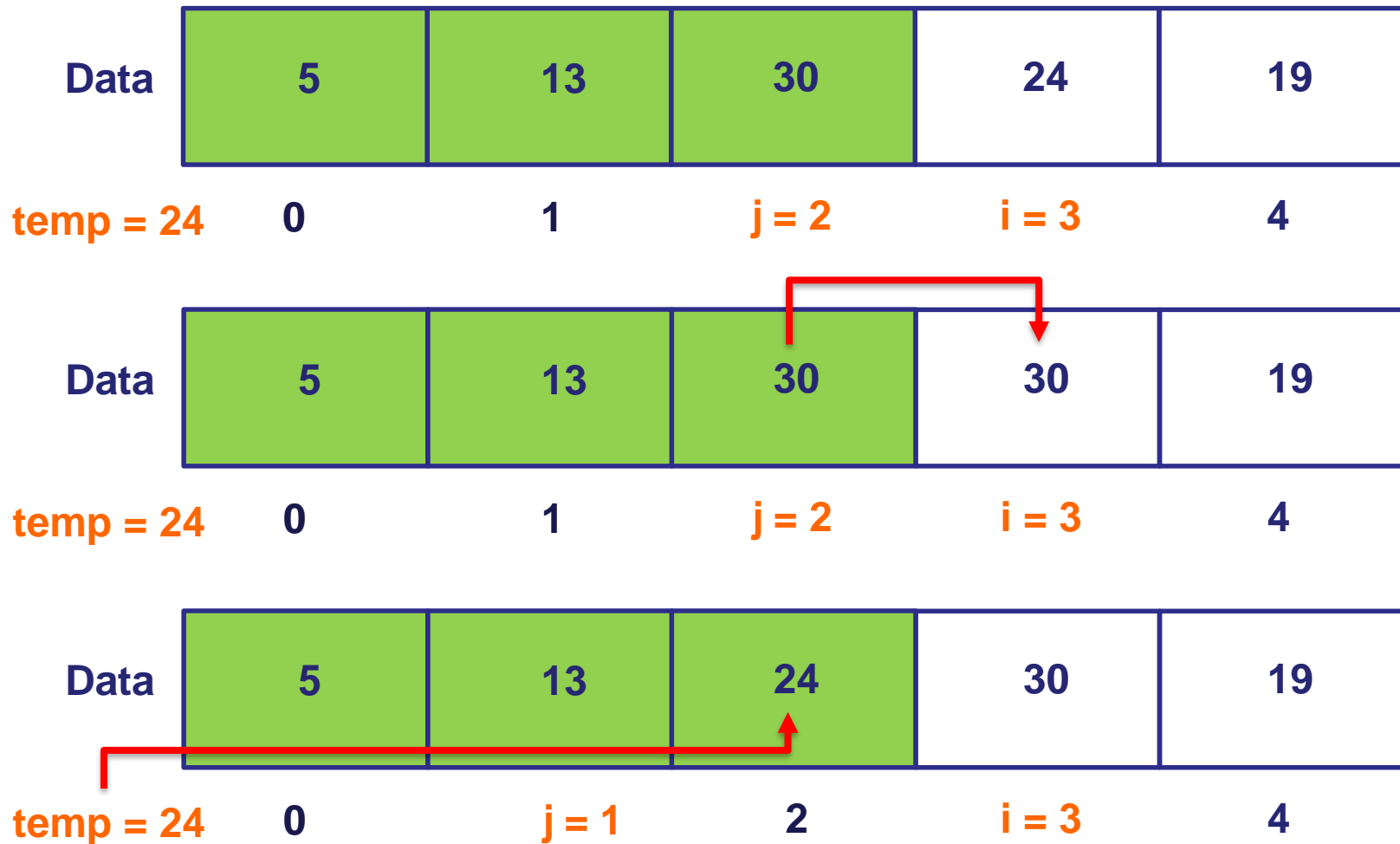
การทำงานของ Insertion Sort รอบที่ i มีค่าเท่ากับ 1



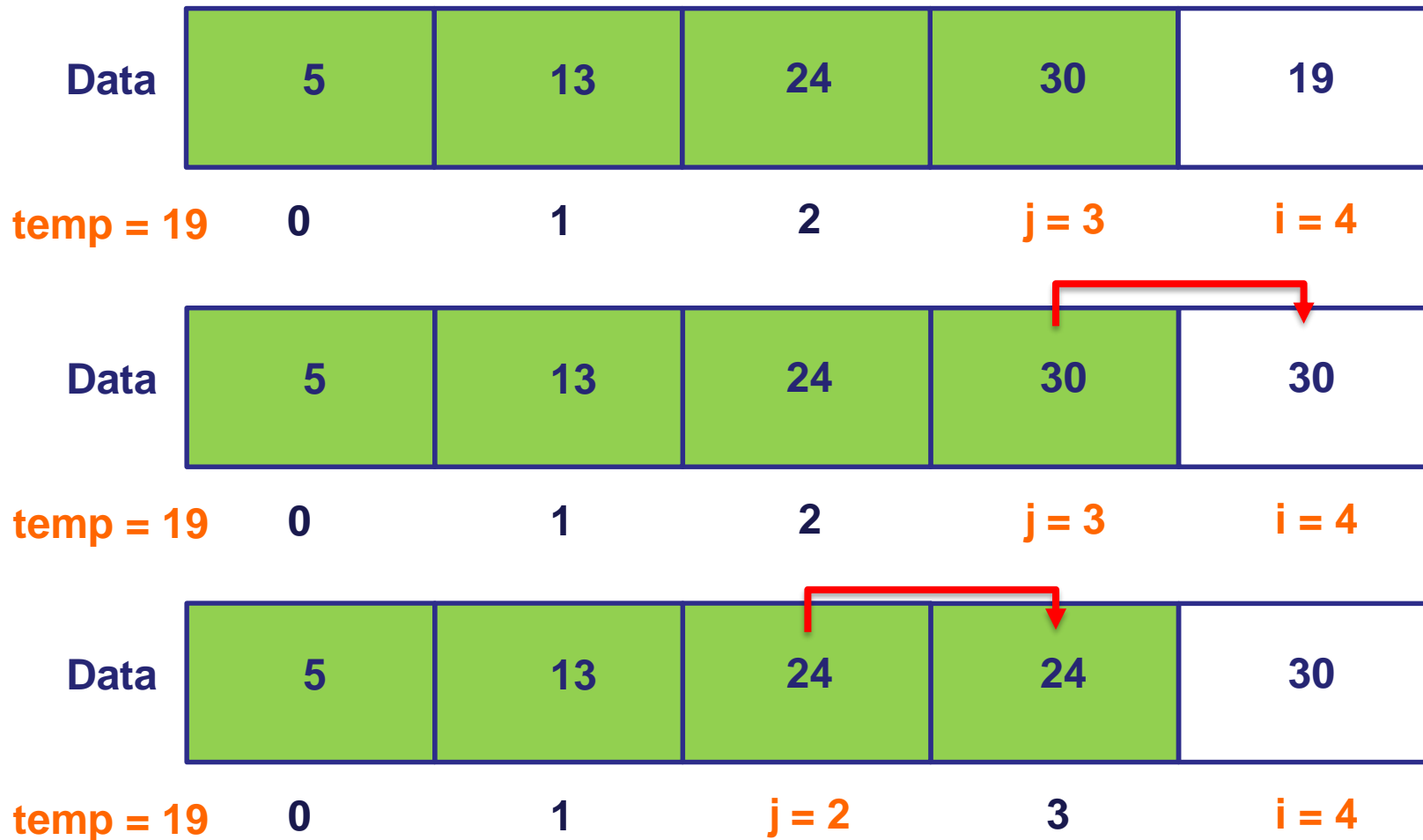
การทำงานของ Insertion Sort รอบที่ i มีค่าเท่ากับ 2



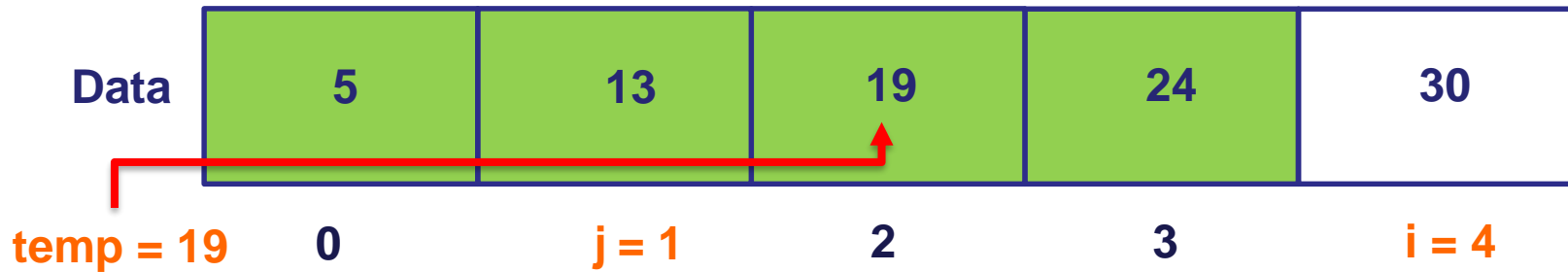
การทำงานของ Insertion Sort รอบที่ i มีค่าเท่ากับ 3



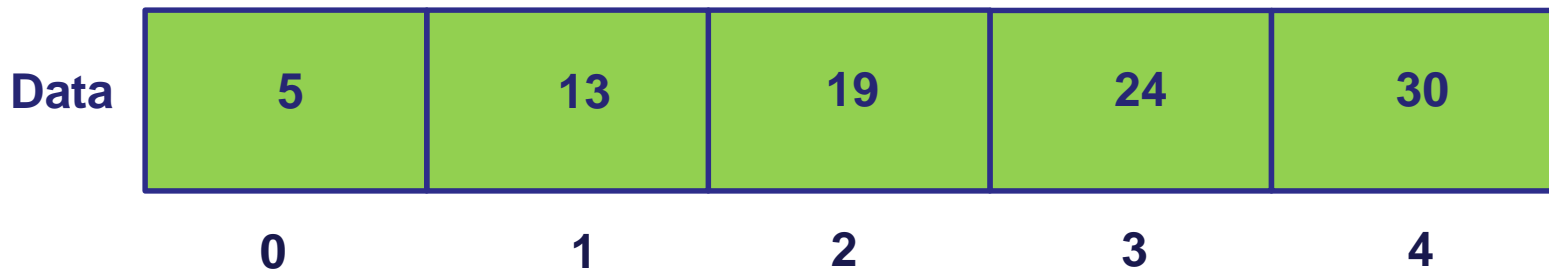
การทำงานของ Insertion Sort รอบที่ i มีค่าเท่ากับ 4



การทำงานของ Insertion Sort รอบที่ i มีค่าเท่ากับ 4 [2]



*** ผลลัพธ์ ***



บทที่ 9-1 Sorting

บทเรียนย่อย

9.1 Sorting Concept

9.2 Insertion Sort Algorithm

9.3 Insertion Sort Implementation

9.4 Selection Sort Algorithm

9.5 Selection Sort Implementation

9.6 Bubble Sort Algorithm

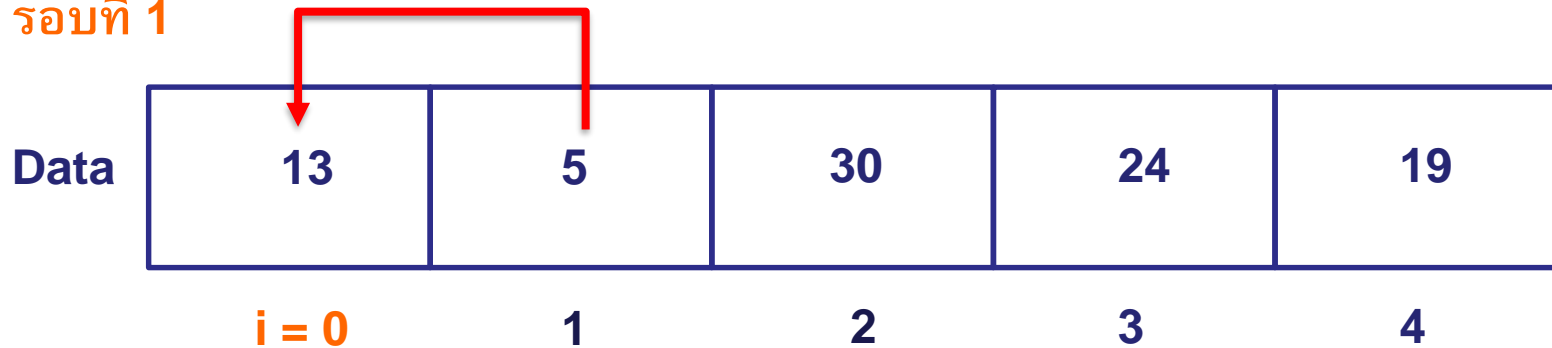
9.7 Bubble Sort Implementation

9.4 Selection Sort Algorithm

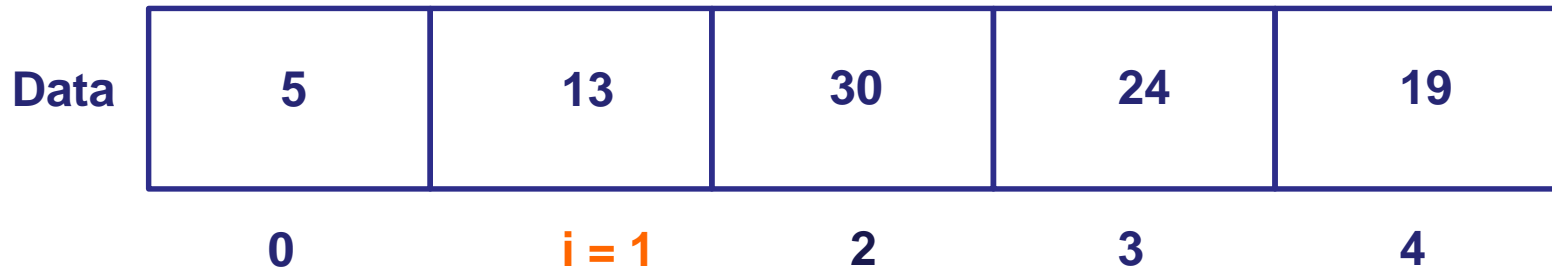
Selection Sort (การเรียงลำดับแบบเลือก) เป็นการเรียงลำดับแบบง่าย โดยเริ่มจากการหาตำแหน่งข้อมูลที่มีค่าน้อยที่สุดในกลุ่ม โดยเริ่มตั้งแต่ข้อมูลตัวแรกจนถึงข้อมูลตัวที่ n แล้วนำมาสลับกับข้อมูลตัวแรก ซึ่งจะทำให้ได้ข้อมูลตัวแรกที่มีค่าน้อยที่สุด แล้วไปหาตำแหน่งข้อมูลที่มีค่าน้อยที่สุดในกลุ่มตั้งแต่ข้อมูลตัวที่ 2 ถึงข้อมูลตัวที่ n เพื่อนำมาสลับกับข้อมูลในตัวที่ 2 และดำเนินการในลักษณะนี้ จนกระทั่งสลับข้อมูลตัวที่ $n - 1$ จึงจะได้ข้อมูลที่จัดเรียงลำดับที่ถูกต้อง

ลักษณะการดำเนินการของ Selection Sort

รอบที่ 1

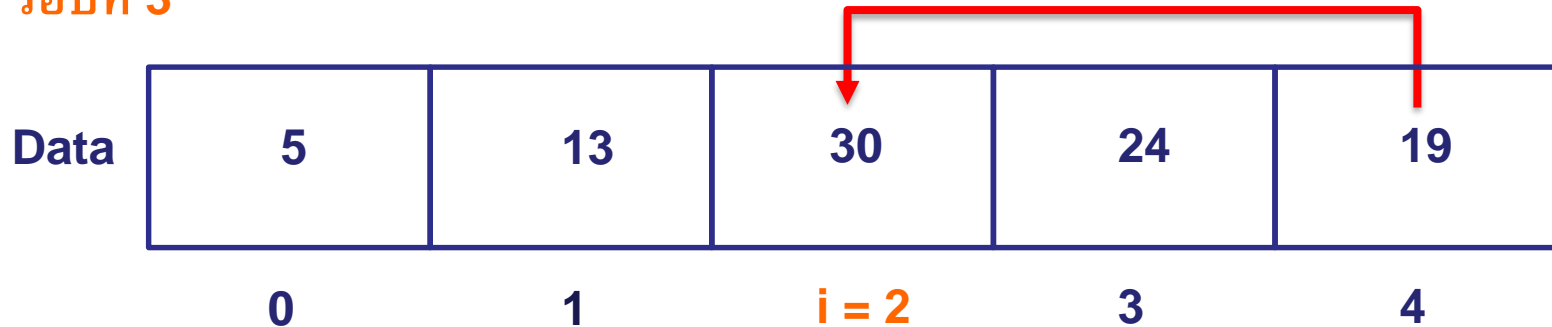


รอบที่ 2

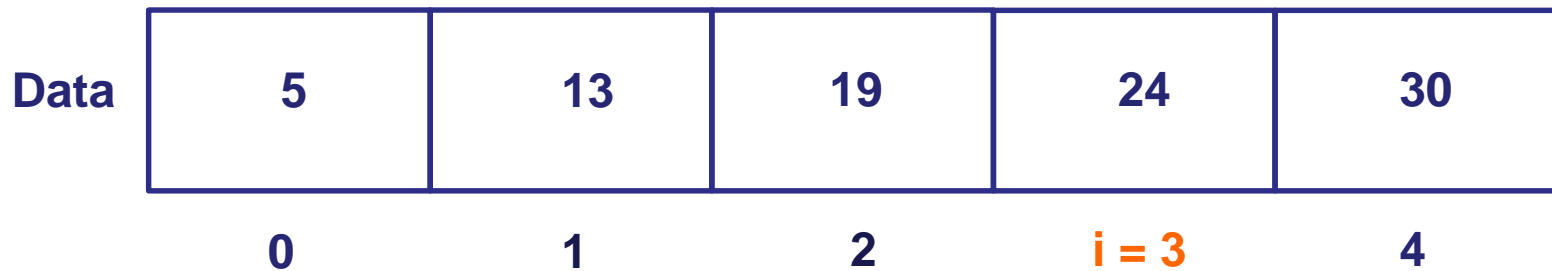


ลักษณะการดำเนินการของ Selection Sort [2]

รอบที่ 3



รอบที่ 4



ลักษณะการดำเนินการของ Selection Sort [3]

ผลลัพธ์

Data	5	13	19	24	30
	0	1	2	3	4

Time :: Best: $O(n^2)$ Worst: $O(n^2)$ Average: $O(n^2)$

บทที่ 9-1 Sorting

บทเรียนย่อย

9.1 Sorting Concept

9.2 Insertion Sort Algorithm

9.3 Insertion Sort Implementation

9.4 Selection Sort Algorithm

9.5 Selection Sort Implementation

9.6 Bubble Sort Algorithm

9.7 Bubble Sort Implementation

9.5 Selection Sort Implementation

Algorithm : Selection Sort

Pre : data[] (int) , size (int)

Post : ข้อมูลถูกจัดเรียงลำดับจากน้อยไปมาก

1. เริ่มต้น

2. ประกาศตัวแปรชื่อ temp, min, i และ j เป็นเลขจำนวนเต็ม

3. ทำซ้ำ ตั้งแต่ $i = 0$ จนมีค่าเท่ากับ $size - 2$

- 3.1 กำหนดให้ min มีค่าเท่ากับ i

- 3.2 ทำซ้ำ ตั้งแต่ $j = i + 1$ จนมีค่าเท่ากับ $size - 1$

.....

9.5 Selection Sort Implementation [2]

.....

3.2.1 ถ้า ข้อมูลตำแหน่งที่ j มีค่าน้อยกว่าข้อมูล
ตำแหน่งที่ min

3.2.1.1 หากเป็นจริง กำหนดให้ min มีค่า
เท่ากับ j

3.3 กำหนดให้ temp มีค่าเท่ากับ data ตำแหน่งที่ i

3.4 กำหนดให้ data ตำแหน่งที่ i มีค่าเท่ากับ data
ตำแหน่งที่ min

3.5 กำหนดให้ data ตำแหน่งที่ min มีค่าเท่ากับ temp

4. จบการทำงาน

การทำงานของ Selection Sort รอบที่ i มีค่าเท่ากับ 0

Data

13

5

30

24

19

temp = ? min, i = 0

j = 1

2

3

4

Data

13

5

30

24

19

temp = ? i = 0

min, j = 1

2

3

4



Data

13

5

30

24

19

temp = ? i = 0

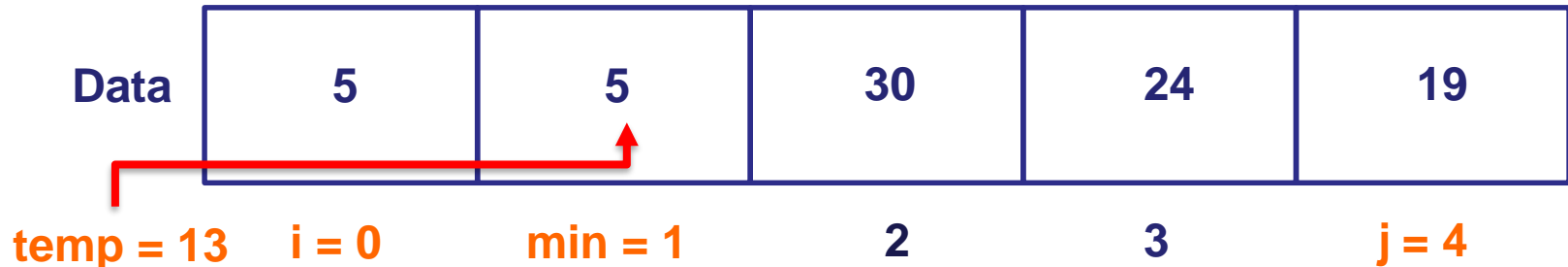
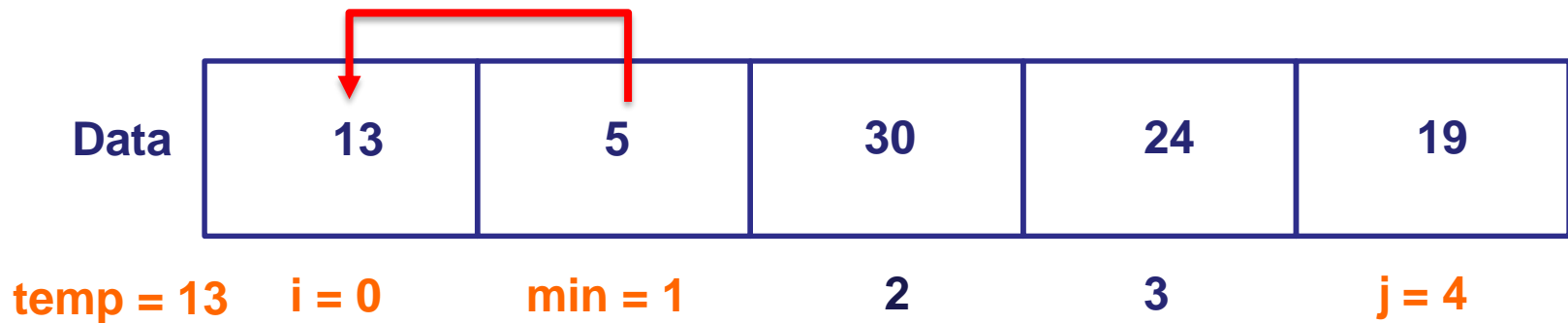
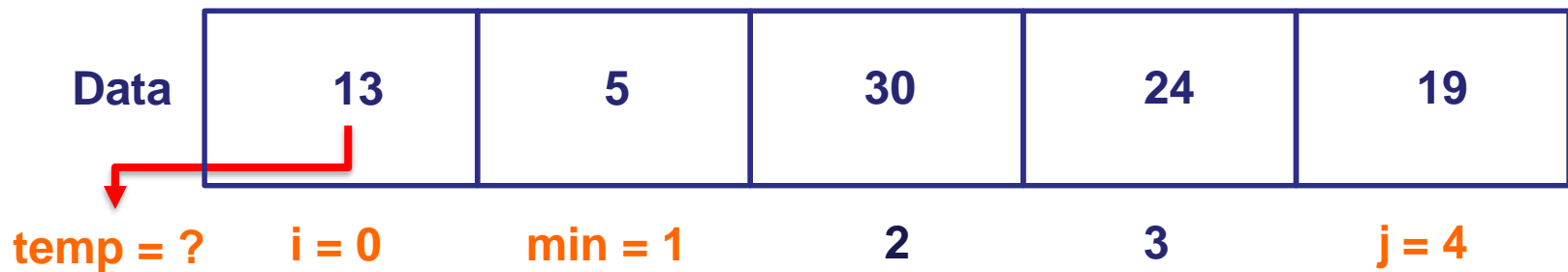
min = 1

2

3

j = 4

การทำงานของ Selection Sort รอบที่ i มีค่าเท่ากับ 0 [2]



การทำงานของ Selection Sort รอบที่ i มีค่าเท่ากับ 0 [3]

Data

5

13

30

24

19

temp = 13 i = 0

min = 1

2

3

j = 4

บทที่ 9-1 Sorting

บทเรียนย่อย

9.1 Sorting Concept

9.2 Insertion Sort Algorithm

9.3 Insertion Sort Implementation

9.4 Selection Sort Algorithm

9.5 Selection Sort Implementation

9.6 Bubble Sort Algorithm

9.7 Bubble Sort Implementation

9.6 Bubble Sort Algorithm

Bubble Sort (การเรียงลำดับแบบฟอง) เป็นการเรียงลำดับแบบการเปรียบเทียบเป็นคู่ ๆ โดยถ้าข้อมูลคู่นั้นยังเรียงลำดับไม่ถูกต้องให้ทำการสลับค่ากัน ซึ่งการเปรียบเทียบอาจจะเริ่มจากข้อมูลคู่แรกไปคู่สุดท้ายที่จัดเรียงแล้ว หรือเริ่มจากข้อมูลคู่สุดท้ายไปคู่แรกที่จัดเรียงแล้ว จากนั้นจะไปเริ่มต้นการเปรียบเทียบแบบเดียวกันอีกจนกระทั่งจัดเรียงลำดับเรียบร้อยแล้ว

ลักษณะการดำเนินการของ Bubble Sort รอบที่ 1

Data	13	5	30	24	19
	0	1	2	3	4

Data	13	5	30	19	24
	0	1	2	3	4

Data	13	5	19	30	24
	0	1	2	3	4

ลักษณะการดำเนินการของ Bubble Sort รอบที่ 1 [1]

Data	13	5	19	30	24
	0	1	2	3	4

Data	5	13	19	30	24
	0	1	2	3	4

ลักษณะการดำเนินการของ Bubble Sort รอบที่ 2

Data	5	13	19	30	24
	0	1	2	3	4

Data	5	13	19	24	30
	0	1	2	3	4

Data	5	13	19	24	30
	0	1	2	3	4

ลักษณะการดำเนินการของ Bubble Sort รอบที่ 3

Data	5	13	19	24	30
	0	1	2	3	4

Data	5	13	19	24	30
	0	1	2	3	4

Data	5	13	19	24	30
	0	1	2	3	4

ลักษณะการดำเนินการของ Bubble Sort รอบที่ 4

Data	5	13	19	24	30
	0	1	2	3	4

Data	5	13	19	24	30
	0	1	2	3	4

Time :: Best: $O(n^2)$ Worst: $O(n^2)$ Average: $O(n^2)$

บทที่ 9-1 Sorting

บทเรียนย่อย

9.1 Sorting Concept

9.2 Insertion Sort Algorithm

9.3 Insertion Sort Implementation

9.4 Selection Sort Algorithm

9.5 Selection Sort Implementation

9.6 Bubble Sort Algorithm

9.7 Bubble Sort Implementation

9.5 Bubble Sort Implementation

Algorithm : Bubble Sort

Pre : data[] (int) , size (int)

Post : ข้อมูลถูกจัดเรียงลำดับจากน้อยไปมาก

1. เริ่มต้น

2. ประกาศตัวแปรชื่อ temp, i และ j เป็นเลขจำนวนเต็ม

3. ทำซ้ำ ตั้งแต่ $i = 1$ จนมีค่าเท่ากับ $size - 1$

- 3.1 ทำซ้ำ ตั้งแต่ $j = size - 1$ จนมีค่าลดลงเท่ากับ i

- 3.1.1 ถ้า data ตำแหน่งที่ j มีค่าน้อยกว่า data ตำแหน่งที่ $j - 1$

.....

9.5 Bubble Sort Implementation [2]

Algorithm : Bubble Sort

Pre : data[] (int) , size (int)

Post : ข้อมูลถูกจัดเรียงลำดับจากน้อยไปมาก

.....

3.1.1.1 กำหนดให้ temp มีค่าเท่ากับ data
ตำแหน่งที่ j

3.1.1.2 กำหนดให้ data ตำแหน่งที่ j มีค่าเท่ากับ
data ตำแหน่งที่ j - 1

3.1.1.3 กำหนดให้ data ตำแหน่งที่ j -1 มีค่าเท่ากับ
temp

4. จบการทำงาน

แบบฝึกหัดที่ 1

1. สร้างคลาสชื่อ `SortingArray` เพื่อเรียงลำดับข้อมูลเลขจำนวนเต็มจากน้อยไปมาก โดยมีความสามารถในการจัดการข้อมูล ดังนี้

- สามารถสุ่มข้อมูลเลขจำนวนเต็มใส่อาร์เรย์
- สามารถจัดเรียงลำดับข้อมูลแบบ Insertion Sort
- สามารถจัดเรียงลำดับข้อมูลแบบ Selection Sort
- สามารถจัดเรียงลำดับข้อมูลแบบ Bubble Sort
- สามารถแสดงข้อมูลทั้งหมด

2. นำคลาสที่สร้างขึ้นไปทดสอบการใช้งานในฟังก์ชัน `main` โดยทำการสร้างเป็นลักษณะเมนูสำหรับทดลองทุกความสามารถที่มีในคลาส `SortingArray` ของตนเอง

แบบฝึกหัดที่ 2

1. นำคลาสที่ชื่อ DbLinkedList ใน example04 มาปรับปรุงเพื่อเพิ่มความสามารถในการเรียงลำดับข้อมูลเลขจำนวนเต็มจากน้อยไปมาก โดยเลือกรูปแบบการเรียงลำดับข้อมูล 1 แบบ จากรายการดังนี้

- จัดเรียงลำดับข้อมูลแบบ Insertion Sort
- จัดเรียงลำดับข้อมูลแบบ Selection Sort
- จัดเรียงลำดับข้อมูลแบบ Bubble Sort

2. นำคลาสที่สร้างขึ้นไปทดสอบการใช้งานในฟังก์ชัน main โดยทำการสร้างเป็นลักษณะเมนูสำหรับทดลองทุกความสามารถที่มีในคลาส DbLinkedList ของตนเอง