

# Lab 05: Divide & Conquer Algorithms

24 กันยายน 2567

## Merge Sort หรือ ? แปลก ๆ นะ

ในคาบเรียนเราได้เรียนรู้การเขียน merge sort จาก pseudocode แล้ว ซึ่งเป็นวิธีการเขียนแบบ recursive ทำให้ได้การทำงานแบบ top-down แต่ในการเขียนโปรแกรมนั้นสามารถทำได้อีกวิธีคือการเขียน iterative โดยการเราลองประยุกต์ใช้เทคนิคนี้กับ merge sort จะได้รูปแบบเป็น bottom-up ซึ่ง 2 วิธีการนี้ใช้หลักการ merge เหมือนกัน

ดูตัวอย่างได้ดังรูป

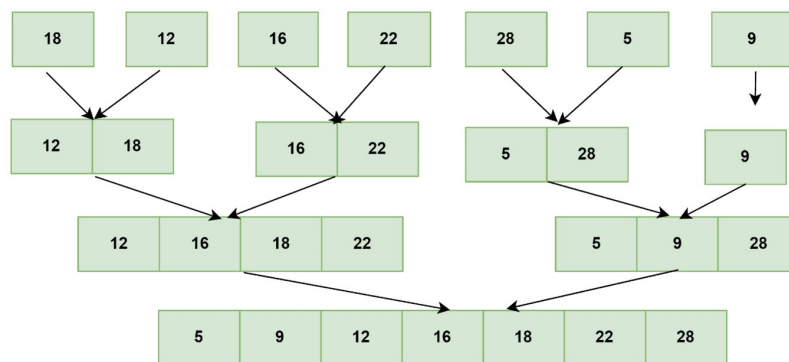


Figure 1: แผนภาพการทำงานของ Merge Sort

## งานของนักศึกษา

รับค่าจำนวนตัวเลขจากผู้ใช้ และตัวเลขแต่ละตัว จากนั้นทำการแสดงผลลำดับที่ถูกเรียงเรียบร้อยแล้ว

### ข้อมูลนำเข้า (Input)

บรรทัดที่ 1	รับค่าจำนวนข้อมูล ( $n$ )
บรรทัดที่ 2	ค่าใน Array จำนวน $n$ ตัว

### ข้อมูลส่งออก (Output)

บรรทัดที่ 1	ข้อมูลที่ได้รับเข้ามาที่ถูกเรียงลำดับจากน้อยไปมาก
-------------	---

**Algorithm 1** BottomUpMergeSort( $A, n$ )**Require:**  $A$  (a list of elements to be sorted),  $n$  (Number of elements in list)

```

if  $n < 2$  then
    Return
end if
 $i \leftarrow 1$  ▷ Initial subarray size
while  $i < n$  do
     $j \leftarrow 0$ 
    while  $j < n - i$  do
        if  $n < j + (2 \times i)$  then
            merge( $A, j, j + i, n$ )
            ▷ Merge subarray  $A[j \dots j + i - 1]$  with  $A[j + i \dots n - 1]$ 
        else
            merge( $A, j, j + i, j + (2 \times i)$ )
            ▷ Merge subarray  $A[j \dots j + i - 1]$  with  $A[j + i \dots j + 2 \times i - 1]$ 
        end if
         $j \leftarrow j + 2 \times i$ 
    end while
     $i \leftarrow i \times 2$ 
end while

```

**ตัวอย่างข้อมูลนำเข้า ส่งออก (Examples of Input & Output)**

Input	Output
6 -3 -8 3 -1 -2 -4	-8 -4 -3 -2 -1 3
10 152 -91 4 105 -15 46 173 166 14 163	-91 -15 4 14 46 105 152 163 166 173

## Quicksort

วันนี้เรามาลองเขียน Quicksort โดยเป็น Sorting Algorithms ที่มี Strategy แบบ Divide-and-Conquer ที่เราจะทำการแบ่งออกเป็น 2 Subarray โดยการหาตำแหน่งของ Pivot โดยใช้ **Hoare's Partition** โดยนำสิ่งที่เราเรียนจากอาจารย์เมื่อสักครู่นี้ มาเขียนโปรแกรมภาษา C หรือ C++ ดู

---

### Algorithm 2 Quicksort( $A$ )

---

**Require:**  $A[l \dots r]$  (An array or subarray)

if  $l < r$  then

$s \leftarrow \text{HoarePartition}(A[l \dots r])$

▷  $s$  is a split position

    Quicksort( $A[l \dots s - 1]$ )

    Quicksort( $A[s + 1 \dots r]$ )

end if

---



---

### Algorithm 3 HoarePartition( $A$ )

---

**Require:**  $A[l \dots r]$  (An array or subarray)

$p \leftarrow A[l]$

$i \leftarrow l$

$j \leftarrow r + 1$

while  $i < j$  do

    Repeat  $i \leftarrow i + 1$  until  $A[i] \geq p$

    Repeat  $j \leftarrow j - 1$  until  $A[j] \leq p$

    swap( $A[i], A[j]$ )

end while

swap( $A[i], A[j]$ )

▷ undo last swap when  $i \geq j$

swap( $A[l], A[j]$ )

return  $j$

---

## งานของนักศึกษา

จงใช้วิธีการ Quicksort ในการเรียงตัวเลขใน Array จากน้อยไปมาก

### ข้อมูลนำเข้า (Input)

บรรทัดที่ 1	รับค่าจำนวนข้อมูล ( $n$ )
บรรทัดที่ 2	ค่าใน Array จำนวน $n$ ตัว

### ข้อมูลส่งออก (Output)

บรรทัดที่ 1	ข้อมูลที่ได้รับเข้ามาที่ถูกเรียงลำดับจากน้อยไปมาก
-------------	---

## ตัวอย่างข้อมูลนำเข้า ส่งออก (Examples of Input &amp; Output)

Input	Output
6 -3 -8 3 -1 -2 -4	-8 -4 -3 -2 -1 3
10 152 -91 4 105 -15 46 173 166 14 163	-91 -15 4 14 46 105 152 163 166 173

## 2D Closest Pair

ในงานนี้จะให้ทุกคนได้ลองเขียน algorithm ที่ใช้ในการหาระยะทางของจุดที่สั้นที่สุดบนระนาบ 2 มิติ โดยที่มีเงื่อนไขว่าต้องใช้วิธี divide-and-conquer ในการคำนวณ

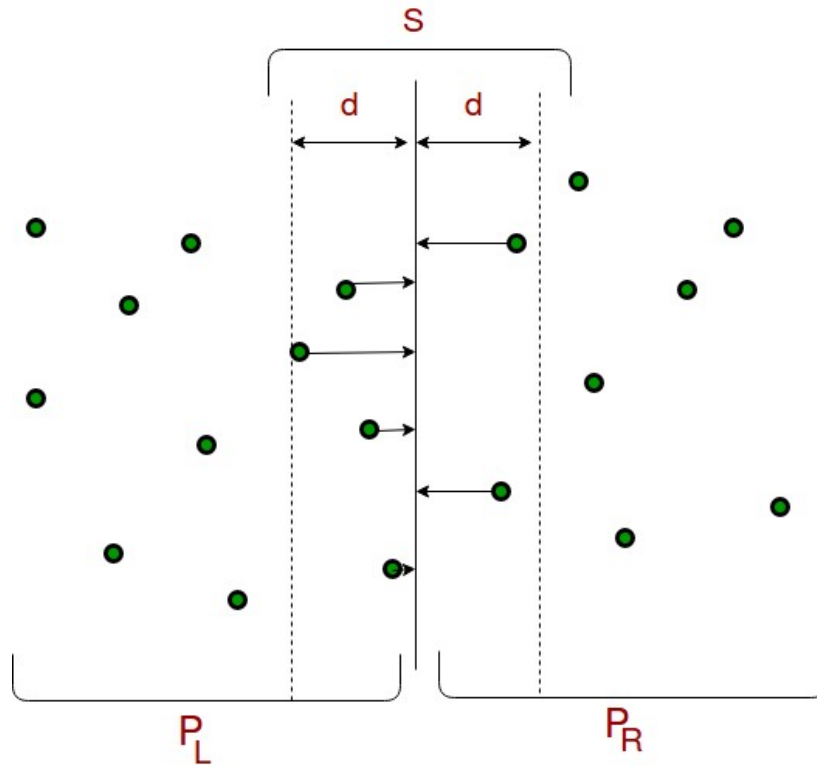


Figure 2: ตัวอย่างการทำงานของ 2D Closest Pair

### 1. แบ่งข้อมูล

- เรียงลำดับจุดทั้งหมดตามแกน  $X$
- แบ่งจุดออกเป็นสองส่วนเท่าๆ กัน โดยแบ่งตามเส้นแนวนตั้งที่อยู่กลางพื้นที่พิจารณา แบ่งจุดจุดให้ได้สองกลุ่มซ้าย (Left) และขวา (Right)

### 2. เรียกทำซ้ำ (Recursion)

- แก้ปัญหาในกลุ่มจุดด้านซ้าย (Left) และด้านขวา (Right) โดยหาคู่จุดที่ใกล้ที่สุดในแต่ละฝั่ง
- เปรียบเทียบค่าระยะทางใกล้ที่สุดที่ได้จากทั้งสองกลุ่มว่าระยะทางขั้นต่ำของแต่ละฝั่งคือเท่าใด (เรียกว่าระยะทาง  $d$ )

### 3. การรวมผล (Conquer)

- ตรวจสอบจุดที่อยู่ใกล้เส้นแบ่ง (Mid-line):** ตรวจสอบคู่จุดที่อยู่ใกล้กันแต่อยู่คนละฝั่งของเส้นแบ่ง โดยพิจารณาเฉพาะจุดที่มีระยะทางจากเส้นแบ่งไม่เกิน  $d$
- คำนวณระยะทางของคู่จุดในพื้นที่ตรงกลางระหว่างสองกลุ่มนี้ (Mid-region) และหาระยะทางที่สั้นที่สุดในพื้นที่นี้

### งานของนักศึกษา

การหาระยะทางที่สั้นที่สุดของ 2 จุดใดๆบนระนาบ 2 มิติ โดยรับค่าจำนวนจุดจากผู้ใช้และรับค่าพิกัดของจุด  $x$  และ  $y$  ใด ๆ จากนั้นแสดงผลระยะที่สั้นที่สุด

### ข้อมูลนำเข้า (Input)

บรรทัดที่ 1	รับค่าจำนวนจุดทั้งหมด ( $n$ )
บรรทัดที่ 2 ถึง ( $n + 1$ )	ค่าพิกัด $x, y$ ของแต่ละจุด $n$ จุด

### ข้อมูลส่งออก (Output)

บรรทัดที่ 1	ระยะทางที่สั้นที่สุดทศนิยม 3 ตำแหน่ง (แสดงโดยการพิมพ์รูปแบบ $\%.3f$ )
-------------	---

### ตัวอย่างข้อมูลนำเข้า ส่งออก (Examples of Input & Output)

Input	Output
5 10 1 0 6 8 3 2 0 9 4	1.414
6 10 1 0 6 8 2 2 0 4 4 6 6	2.236

## ยัณฑ์มงคล (Make a Wish)

หลังจากนักศึกษาได้ไปขอพรกับหลวงปู่ให้สอบครั้งที่ 1 สำเร็จไปได้ด้วยดี ผลปรากฏว่าผลการสอบครั้งที่ 1 นั้นดีอย่างไม่น่าเชื่อ นักศึกษาจึงเลื่อมใสในหลวงปู่มาก ๆ กระทั่งได้ชวนเพื่อน ๆ ไปขอพรในการสอบครั้งที่ 2 ต่อไป รวมไปถึงเรื่องอื่น ๆ ที่ต้องการ เช่น ความรัก การเงิน การงาน ฯลฯ

ทีนี้นักศึกษาได้ยกโขยงกันไปเป็นจำนวน 83 คน ไปที่วัดพุทธบูชาตามเคย ได้พบกับหลวงปู่เหมือนเดิม ที่นี้หลวงปู่ไม่ได้ให้สายลูกปัดเหมือนกับรอบที่แล้ว แต่ว่าให้กระด้ายันต์ที่ยาวมาก ๆ มาแผ่นหนึ่ง และยังสามารถบอกกับนักศึกษาทุกคนว่า

”หากโยมต้องการความเป็นสิริมงคลสูงสุด โยมต้องนำส่วนที่มีระดับความมงคลที่รวมกันได้มากที่สุดจำนวน 1 ผืน ตัดยังงี้ก็จะได้ผลรวมความมงคลได้มากที่สุด จึงจะได้ความโชคดี เจริญพร”

สมมติว่าหลวงปู่ให้ผ้ายันต์ความยาว 10 เมตรมา แต่ละส่วนมีเขียนเลขดังนี้

-59	76	3	69	-26	-54	-79	-88	72	43
-----	----	---	----	-----	-----	-----	-----	----	----

จะเห็นได้ว่า หากต้องการตัดผ้ายันต์ให้ได้ความมงคลสูงสุด นักศึกษาต้องตัดให้ได้เป็นรูปแบบดังนี้

-59	76	3	69	-26	-54	-79	-88	72	43
-----	----	---	----	-----	-----	-----	-----	----	----

และจะได้ผลรวมความสิริมงคลคือ  $76 + 3 + 69 = 148$  นั่นเอง

### งานของนักศึกษา

จงหาว่า หากหลวงปู่ให้ผ้ายันต์ที่มีความยาว  $n$  เมตร เราจะสามารถตัดอย่างไรให้ได้ผลรวมความเป็นสิริมงคลสูงที่สุดที่เป็นไปได้

### ข้อมูลนำเข้า (Input)

บรรทัดที่ 1	จำนวนเต็ม $n$ แทน ความยาว ผ้ายันต์ที่หลวงปู่ให้มา โดยที่ $1 \leq n \leq 1,000,000$
บรรทัดที่ 2	จำนวนเต็ม $l_i$ ทั้งหมด $n$ ตัว แทนระดับความสิริมงคลของผ้ายันต์แต่ละจุด แต่ละตัวคั่นด้วยช่องว่าง 1 ช่อง โดยที่ $-100,000 \leq l_i \leq 100,000$

### ข้อมูลส่งออก (Output)

บรรทัดที่ 1	ผลรวมความเป็นสิริมงคลที่มากที่สุด
-------------	-----------------------------------

## ตัวอย่างข้อมูลนำเข้า ส่งออก (Examples of Input &amp; Output)

Input	Output
10 -59 76 3 69 -26 -54 -79 -88 -72 43	148
25 -10 7 10 4 12 11 29 -8 15 18 -4 9 24 7 29 30 -8 1 20 14 -9 -2 -3 20 -5	226