517 311 การวิเคราะห์และการออกแบบขั้นตอนวิธี

Divide and Conquer

12 September 2023

เนื้อหา

- การออกแบบอัลกอริทึม
  - บรู๊ซฟอร์ท (Brute Force)
  - การแบ่งแยกและเอาชนะ (Divide and Conquer)
  - กำหนดการพลวัติ (Dynamic Programming)
  - อัลกอริทึมเชิงละโมบ (Greedy Algorithm)
- อัลกอริทึมในกราฟ
- ปัญหากลุ่มเอ็นพีบริบรูณ์

วัตถุประสงค์ (การออกแบบอัลกอริทึม)

- เทคนิคที่ใช้ในการออกแบบอัลกอริทึม
  - การแบ่งแยกและเอาชนะ (Divide and Conquer)
  - กำหนดการพลวัต (Dynamic Programming)
  - อัลกอริทึมเชิงละโมบ (Greedy Algorithm)
- จำหลักการของแต่ละวิธี
- เปรียบเทียบความเหมือนและแตกต่างกันของแต่ละเทคนิค
- เข้าใจปัญหาที่ใช้เป็นตัวอย่างของการออกแบบอัลกอริทึมด้วยเทคนิคที่นำเสนอ
- การออกแบบการแก้ปัญหาด้วยวิธีการออกแบบที่ศึกษาไปได้ (รวมถึงการพิสูจน์ความถูกต้องของอัลกอริทึม)
- วัดประสิทธิภาพของอัลกอริทึมในการแก้ปัญหา

ปัญหา

- คำอธิบายปัญหาในรูปทั่วไป
  - Input
  - Output
  - สมการที่เกี่ยวข้อง (ถ้ามี)
- กรณีตัวอย่าง (Instance)
- การแก้ปัญหา (การหาคำตอบ)
- วัดประสิทธิภาพของอัลกอริทึมในการแก้ปัญหา (เทียบกับบรู๊ซฟอร์ท หรือวิธีการอื่น)

ความรู้ที่จำเป็น

- ความรู้เรื่องการเขียนโปรแกรมและโครงสร้างข้อมูล เช่น การเรียงลำดับ (Sorting)
- ความรู้ทางคณิตศาสตร์และวิธีการพิสูจน์
- ฟังก์ชันเรียกตัวเอง
- การวิเคราะห์ประสิทธิภาพของอัลกอริทึม
- การคิดวิเคราะห์ในเชิงของการเขียนโปรแกรม (อัลกอริทึม)
- การแสดงอัลกอริทึมในรูปรหัสเทียม (Pseudocode)
- การพิสูจน์ความถูกต้องของอัลกอริทึม

จงแสดงอัลกอริทึมในการหาค่าเฉลี่ย (Mean)

คนที่ 1

$$\frac{1+2+2+5+7+11+15}{7}$$

คนที่ 2

$$\frac{x_1+x_2+\ldots+x_n}{n}$$

คนที่ 3

$$\frac{\sum_{i=1}^{n} x_i}{n}$$

คนที่ 4 เมื่อข้อมูลเก็บใน $x_1, x_2, \ldots, x_n$

$$\frac{\sum_{i=1}^{n} x_i}{n}$$

ข้อสังเกต

1. สมการไม่ใช่อัลกอริทึม
2. การเขียนสมการต้องอธิบายตัวแปรที่ใช้ในสมการด้วย
3. การเขียนอัลกอริทึมต้องเขียนในรูปทั่วไป
4. การแสดงตัวอย่างไม่ใช่อัลกอริทึม

วัตถุประสงค์

บรู๊ซฟอร์ท (Brute Force) และ การแบ่งแยกและเอาชนะ (Divide and Conquer)

- หลักการของบรู๊ซฟอร์ท
- หลักการออกแบบโดยวิธีการแบ่งแยกและเอาชนะ
- ปัญหาตัวอย่างที่ใช้หลักการออกแบบโดยวิธีการแบ่งแยกและเอาชนะ
- การวิเคราะห์ประสิทธิภาพของอัลกอริทึมที่ใช้หลักการออกแบบโดยวิธีการแบ่งแยกและเอาชนะ
- ตัวอย่างอัลกอริทึมที่ใช้หลักการออกแบบโดยวิธีการแบ่งแยกและเอาชนะ
  - Binary Search
  - Merge sort
  - Quick sort

ปัญหาการหาค่าเลขชี้กำลัง

ปัญหาการหาค่าเลขชี้กำลัง (Exponentiation Problem) เป็นการคำนวณหาค่าของ a^n สำหรับค่าของ a ที่เป็นจำนวนจริงและไม่เท่ากับศูนย์ และ n เป็นเลขจำนวนเต็มบวกหรือศูนย์

```
p ← 1
for i ← 1 to n do
    p ← p × a
end for
return p
```

ความซับซ้อนเชิงเวลา ?

การคูณเมทริกซ์

ให้ A และ B เป็นเมทริกซ์ขนาดเท่ากับ $n \times n$ ผลของการคูณเมทริกซ์ $C = AB$

$$c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$$

ตัวอย่างการคูณเมทริกซ์ที่มีขนาด 2×2

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \cdot \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} = \begin{bmatrix} a_{11} \cdot b_{11} + a_{12} \cdot b_{21} & a_{11} \cdot b_{12} + a_{12} \cdot b_{22} \\ a_{21} \cdot b_{11} + a_{22} \cdot b_{21} & a_{21} \cdot b_{12} + a_{22} \cdot b_{22} \end{bmatrix}$$

- การหาค่าของสมาชิก c_{ij} ในเมทริกซ์ 1 ตัว ใช้ความซับซ้อนเชิงเวลา ?
- การคูณเมทริกซ์ทั้งหมดใช้ความซับซ้อนเชิงเวลา ?

การแบ่งแยกและเอาชนะ (Divide and Conquer)

1. การแบ่งปัญหา (Divide) แบ่งกรณีตัวอย่างของปัญหา (Instance) ออกเป็นส่วน ๆ โดยแต่ส่วนมีขนาดเล็กลง
2. การแก้ปัญหาย่อย (Conquer) หาคำตอบสำหรับกรณีตัวอย่างของปัญหาที่มีขนาดเล็กพอ แต่ถ้ากรณีตัวอย่างของปัญหายังมีขนาดใหญ่ก็ต้องทำการแบ่งปัญหาอีกครั้ง
3. การรวมคำตอบ (Combine) ทำการประมวลผลคำตอบย่อย ๆ มาเป็นคำตอบของปัญหาที่ต้องการหาคำตอบในระดับบน (ขั้นตอนนี้ในบางปัญหาอาจจะไม่มีก็ได้)

การค้นหาทวิภาค

```
function BinarySearchRec(A, low, high, key)
    mid ← (low + high)/2
    if high < low then
        return −1
    end if
    if A[i] == key then
        return mid
    else if key < A[mid] then
        BinarySearchRec(A, low, mid − 1, key)
    else
        BinarySearchRec(A, mid + 1, high, key)
    end if
end function
```

การลดและเอาชนะ (Decrease and Conquer)

การหาค่าเลขชี้กำลัง

$$a^n = \begin{cases} a^{n/2} \cdot a^{n/2} & \text{เมื่อ } n \text{ เป็นจำนวนคู่} \\ a^{(n-1)/2} \cdot a^{(n-1)/2} \cdot a & \text{เมื่อ } n \text{ เป็นจำนวนคี่} \\ 1 & \text{เมื่อ } n = 0 \end{cases}$$

- เขียนเป็นอัลกอริทึม

function *power*(a, n)

- การแบ่งแยกและเอาชนะ
- ความซับซ้อนเชิงเวลา

การคูณเมทริกซ์

กำหนดให้เมทริกซ์ A และ B มีขนาดเท่ากับ $n \times n$ ซึ่ง $n = 2^k$ โดย $k \geq 0$

- ในกรณีพื้นฐาน เมื่อ $n = 1$ เป็นการคูณตัวเลข 2 จำนวนตามปกติ
- ในกรณีที่ $n \geq 2$ แล้ว จะมีการแบ่งปัญหาใหญ่ออกเป็นปัญหาย่อย โดยเมทริกซ์ A, B และเมทริกซ์ผลคูณ C สามารถแบ่งออกเป็นเมทริกซ์ย่อย 4 เมทริกซ์ที่มีขนาดเป็น $n/2 \times n/2$

$$A = \left[\begin{array}{c|c} A_{11} & A_{12} \\ \hline A_{21} & A_{22} \end{array}\right], B = \left[\begin{array}{c|c} B_{11} & B_{12} \\ \hline B_{21} & B_{22} \end{array}\right] \quad \text{และ} \; C = \left[\begin{array}{c|c} C_{11} & C_{12} \\ \hline C_{21} & C_{22} \end{array}\right]$$

- ผลคูณของ C_{11} มาได้อย่างไร
- อัลกอริทึมอยู่ในหน้าถัดไป
- ความซับซ้อนเชิงเวลาเป็นเท่าไหร่ จงแสดงการคำนวณ

อัลกอริทึม

function $MatMultiply(A, B, dim)$
    if $dim == 1$ **then**
        return $A \times B$
    else
        $P_1 \leftarrow MatMultiply(A_{11}, B_{11}, dim/2)$
        $P_2 \leftarrow MatMultiply(A_{12}, B_{21}, dim/2)$
        $P_3 \leftarrow MatMultiply(A_{11}, B_{12}, dim/2)$
        $P_4 \leftarrow MatMultiply(A_{12}, B_{22}, dim/2)$
        $P_5 \leftarrow MatMultiply(A_{21}, B_{11}, dim/2)$
        $P_6 \leftarrow MatMultiply(A_{22}, B_{21}, dim/2)$
        $P_7 \leftarrow MatMultiply(A_{21}, B_{12}, dim/2)$
        $P_8 \leftarrow MatMultiply(A_{22}, B_{22}, dim/2)$
        $C_{11} \leftarrow P_1 + P_2$; $C_{12} \leftarrow P_3 + P_4$
        $C_{21} \leftarrow P_5 + P_6$; $C_{22} \leftarrow P_7 + P_8$
        return C
    end if
end function

Strassen's Algorithm

$$D_1 = (A_{11} + A_{22})(B_{11} + B_{22}) = A_{11}B_{11} + A_{11}B_{22} + A_{22}B_{11} + A_{22}B_{22}$$
$$D_2 = (A_{21} + A_{22})B_{11} = A_{21}B_{11} + A_{22}B_{11}$$
$$D_3 = A_{11}(B_{12} - B_{22}) = A_{11}B_{12} - A_{11}B_{22}$$
$$D_4 = A_{22}(B_{21} - B_{11}) = A_{22}B_{21} - A_{22}B_{11}$$
$$D_5 = (A_{11} + A_{12})B_{22} = A_{11}B_{22} + A_{12}B_{22}$$
$$D_6 = (A_{21} - A_{11})(B_{11} + B_{12}) = A_{21}B_{11} + A_{21}B_{12} - A_{11}B_{11} - A_{11}B_{12}$$
$$D_7 = (A_{12} - A_{22})(B_{21} + B_{22}) = A_{12}B_{21} + A_{12}B_{22} - A_{22}B_{21} - A_{22}B_{22}$$

$$C = \begin{bmatrix} D_1 + D_4 - D_5 + D_7 & D_3 + D_5 \\ D_2 + D_4 & D_1 + D_3 - D_2 + D_6 \end{bmatrix}$$

Strassen's Algorithm (วิเคราะห์)

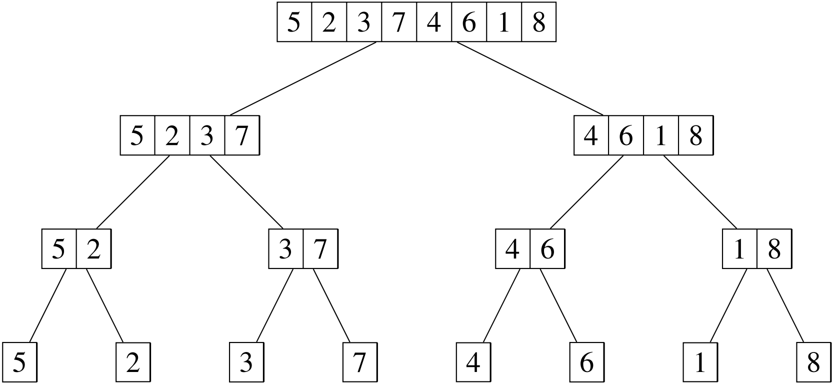
- ความซับซ้อนเชิงเวลาคืออะไร ? คำนวณอย่างไร
- ทำเป็นโปรแกรมภาษา Java ได้อย่างไร

การเรียงลำดับแบบผสาน (Mergesort)

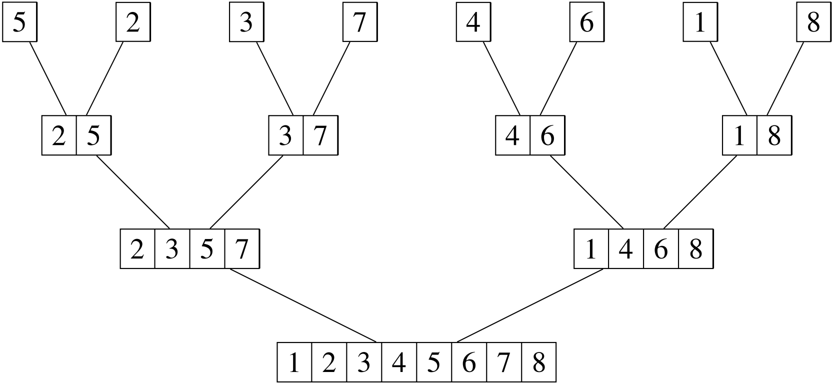
การเรียงลำดับแบบผสานสามารถอธิบายในรูปของการแบ่งแยกและเอาชนะได้ดังนี้

1. การแบ่งปัญหาดำเนินการแบ่งสมาชิกของอาเรย์ A ซึ่งจะถูกแบ่งเป็น 2 ส่วน ด้วยจำนวนที่เท่า ๆ กัน
2. การแก้ปัญหาย่อย เมื่อจำนวนสมาชิกของอาเรย์ไม่มากก็ทำการเรียงลำดับ แต่ถ้าจำนวนสมาชิกของอาเรย์ยังมีจำนวนมากอยู่ก็จะเรียกการแบ่งปัญหาซ้ำ
3. การรวมคำตอบเป็นการรวมอาเรย์ย่อย 2 อาเรย์ที่มีการเรียงลำดับของค่าเป็นอาเรย์เดียวที่เรียงลำดับแล้ว

- ตัวอย่างอยู่ในสไลด์ถัดไป
- อัลกอริทึมอยู่ในไฟล์ pdf แยกไว้
- ความซับซ้อนเท่ากับเท่าใด



รูปที่ 1: การแบ่งของการเรียงลำดับแบบผสาน



รูปที่ 2: การรวมของการเรียงลำดับแบบผสาน

การนับความผกผัน (Counting Inversion)

การนับความผกผันเป็นการเปรียบเทียบลำดับความสำคัญของสิ่งของ (มีลักษณะของการจัดลำดับ) กำหนดให้รายการ A มีสมาชิก $a_1, a_2, \ldots, a_n$ ที่มีค่าเป็นจำนวนเต็มตั้งแต่ 1 ถึง n การนับความผกผันคือการหาจำนวนคู่ของ (i, j) ที่ $1 \leq i < j \leq n$ ที่ $a_i > a_j$

6	3	5	2	8	1	4	7

สำหรับจำนวนของความผกผันในลำดับที่ให้มาเท่ากับ 14 โดยสามารถแสดงความผกผันได้ดังนี้

(6,3)	(6,5)	(6,2)	(6,1)	(6,4)	(3,2)	(3,1)
(5,2)	(5,1)	(5,4)	(2,1)	(8,1)	(8,4)	(8,7)

ถ้า A มีสมาชิก n จำนวนความผกผันมากที่สุดเท่ากับเท่าใด ?

การนับความผกผัน (บรู๊ซฟอร์ท)

```
counter ← 0
for i ← 1 to n − 1 do
    for j ← i + 1 to n do
        if A[i] > A[j] then
            counter ← counter + 1
        end if
    end for
end for
```

- ความซับซ้อนเท่ากับเท่าใด

Divide and Conquer

function $CountInversion(A)$
    $cl \leftarrow 0; cr \leftarrow 0; cc \leftarrow 0$
    $n \leftarrow length(A)$
    if $n > 1$ **then**
        $m \leftarrow n/2$
        $B \leftarrow A[1..m]$
        $C \leftarrow A[m+1..n]$
        $cl \leftarrow CountInversion(B)$
        $cr \leftarrow CountInversion(C)$
        $cc \leftarrow MergeCount(A, B, C)$
    end if
    return $cl + cr + cc$
end function

- หน้าที่ของ $MergeCount(A, B, C)$ คือ การนับจำนวนความผกผันระหว่างการรวมอาร์เรย์ของB และ C แล้วเก็บผลการรวมที่ A

ตัวอย่าง 1

จงแสดงการนับความผกผันของรายการที่กำหนดให้ต่อไปนี้ ด้วยวิธีการแบ่งแยกและเอาชนะที่มีพื้นฐานจากการเรียงลำดับแบบผสาน

6	3	5	2	8	1	4	7

- รอบแรกแบ่งรายการออกเป็น 2 ส่วน คือ รายการ $6, 3, 5, 2$ และ $8, 1, 4, 7$
- รอบถัดมาแบ่งออกเป็น 2 รายการ 2 ชุด คือ รายการ $6, 3, 5, 2$ แบ่งออกเป็น $6, 3$ และ $5, 2$ และอีกรายการ $8, 1, 4, 7$ แบ่งออกเป็น $8, 1$ และ $4, 7$
- รอบต่อมา แบ่งรายการที่มีสมาชิก 1 ค่า

ในกรณีที่รายการที่มีสมาชิก 1 ค่า ไม่มีความผกผัน หรือความผกผันเท่ากับศูนย์

- การรวมรายการ $B = (6)$ และ $C = (3)$ จะได้ $A = (3, 6)$ โดย $cl + cr + cc = 0 + 0 + 1 = 1$
- การรวมรายการ $B = (5)$ และ $C = (2)$ จะได้ $A = (2, 5)$ โดย $cl + cr + cc = 0 + 0 + 1 = 1$
- การรวมรายการ $B = (8)$ และ $C = (1)$ จะได้ $A = (1, 8)$ โดย $cl + cr + cc = 0 + 0 + 1 = 1$
- การรวมรายการ $B = (4)$ และ $C = (7)$ จะได้ $A = (4, 7)$ โดย $cl + cr + cc = 0 + 0 + 0 = 0$

ในกระบวนการ *MergeCount* ในรอบต่อมา

- การรวมรายการ $B = (3, 6)$ และ $C = (2, 5)$ จะได้ $A = (2, 3, 5, 6)$ โดย $cl = 1$ และ $cr = 1$ สำหรับ cc คำนวณได้ดังนี้
  - การรวมรายการ $B = (3, 6)$ และ $C = (2, 5)$ จะได้รายการรวม $A = (2)$ เหลือรายการ $B = (3, 6)$ และ $C = (5)$ จำนวนความผกผันเท่ากับ 2
  - การรวมรายการ $B = (3, 6)$ และ $C = (5)$ จะได้รายการรวม $A = (2, 3)$ เหลือรายการ $B = (6)$ และ $C = (5)$ จำนวนความผกผันเท่ากับ 0
  - การรวมรายการ $B = (6)$ และ $C = (5)$ จะได้รายการรวม $A = (2, 3, 5)$ เหลือรายการ $B = (6)$ จำนวนความผกผันเท่ากับ 1
  - รายการรวม $A = (2, 3, 5, 6)$ ความผกผันจากการรวมรายการทั้งสอง คือ $cc = 2 + 0 + 1 = 3$

ดังนั้นผลรวม $cl + cr + cc = 1 + 1 + 3 = 5$

- การรวมรายการ $B = (1, 8)$ และ $C = (4, 7)$ จะได้ $A = (1, 4, 7, 8)$ โดย $cl = 1$ และ $cr = 0$
  - การรวมรายการ $B = (1, 8)$ และ $C = (4, 7)$ จะได้รายการรวม $A = (1)$ เหลือรายการ $B = (8)$ และ $C = (4, 7)$ จำนวนความผกผันเท่ากับ 0
  - การรวมรายการ $B = (8)$ และ $C = (4, 7)$ จะได้รายการรวม $A = (1, 4)$ เหลือรายการ $B = (8)$ และ $C = (7)$ จำนวนความผกผันเท่ากับ 1
  - การรวมรายการ $B = (8)$ และ $C = (7)$ จะได้รายการรวม $A = (1, 4, 7)$ เหลือรายการ $B = (8)$ จำนวนความผกผันเท่ากับ 1
  - รายการรวม $A = (1, 4, 7, 8)$ ความผกผันจากการรวมรายการทั้งสองคือ $cc = 0 + 1 + 1 = 2$

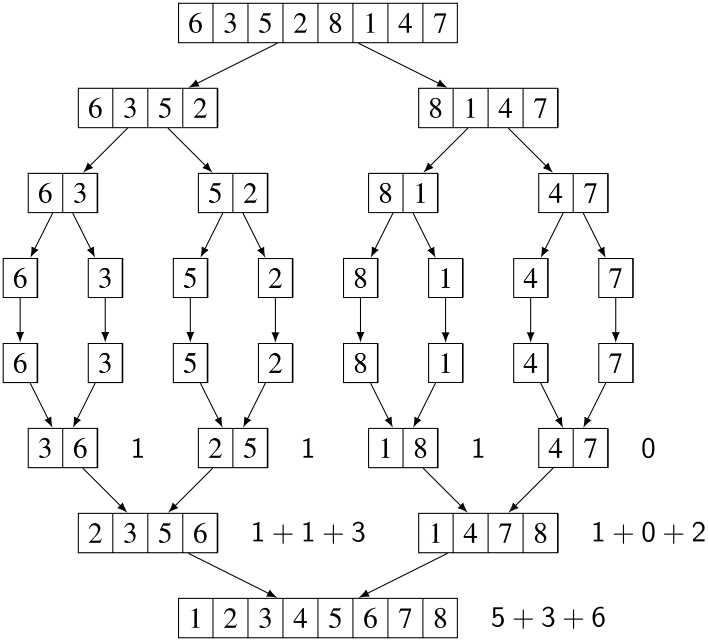
  ดังนั้นผลรวม $cl + cr + cc = 1 + 0 + 2 = 3$

การรวมรายการ $B = (2, 3, 5, 6)$ และ $C = (1, 4, 7, 8)$ จะได้ $A = (1, 2, 3, 4, 5, 6, 7, 8)$ โดย $cl = 5$ และ $cr = 3$

- การรวมรายการ $B = (2, 3, 5, 6)$ และ $C = (1, 4, 7, 8)$ จะได้ $A = (1)$ เหลือรายการ $B = (2, 3, 5, 6)$ และ $C = (4, 7, 8)$ จำนวนความผกผันเท่ากับ 4
- การรวมรายการ $B = (2, 3, 5, 6)$ และ $C = (4, 7, 8)$ จะได้ $A = (1, 2)$ เหลือรายการ $B = (3, 5, 6)$ และ $C = (4, 7, 8)$ จำนวนความผกผันเท่ากับ 0
- การรวมรายการ $B = (3, 5, 6)$ และ $C = (4, 7, 8)$ จะได้ $A = (1, 2, 3)$ เหลือรายการ $B = (5, 6)$ และ $C = (4, 7, 8)$ จำนวนความผกผันเท่ากับ 0
- การรวมรายการ $B = (5, 6)$ และ $C = (4, 7, 8)$ จะได้ $A = (1, 2, 3, 4)$ เหลือรายการ $B = (5, 6)$ และ $C = (7, 8)$ จำนวนความผกผันเท่ากับ 2

- การรวมรายการ $B = (5, 6)$ และ $C = (7, 8)$ จะได้ $A = (1, 2, 3, 4, 5)$ เหลือรายการ $B = (6)$ และ $C = (7, 8)$ จำนวนความผกผันเท่ากับ 0
- การรวมรายการ $B = (6)$ และ $C = (7, 8)$ จะได้ $A = (1, 2, 3, 4, 5, 6)$ เหลือรายการ $C = (7, 8)$ จำนวนความผกผันเท่ากับ 0
- การรวมรายการ $C = (7, 8)$ จะได้ $A = (1, 2, 3, 4, 5, 6, 7)$ เหลือรายการ $C = (8)$ จำนวนความผกผันเท่ากับ 0
- รายการรวม $A = (1, 2, 3, 4, 5, 6, 7, 8)$ ความผกผันจากการรวมรายการทั้งสองคือ $cc = 4 + 0 + 0 + 2 + 0 + 0 + 0 = 6$

ดังนั้นผลรวม $cl + cr + cc = 5 + 3 + 6 = 14$


6 3 5 2 8 1 4 7
6 3 5 2
8 1 4 7
6 3
5 2
8 1
4 7
6
3
5
2
8
1
4
7
3 6
1
2 5
1
1 8
1
4 7
0
2 3 5 6
1 + 1 + 3
1 4 7 8
1 + 0 + 2
1 2 3 4 5 6 7 8
5 + 3 + 6

ปัญหาค่าผลรวมที่มากที่สุดของอาเรย์ย่อย

นิยาม 1

ปัญหาคำนวณหาค่าผลรวมที่มากที่สุดของอาเรย์ย่อย (Maximum Subarray) กำหนดอาเรย์ของตัวเลข แสดงผลรวมที่มากที่สุดของอาเรย์ย่อยที่มีลำดับต่อเนื่องกัน (อาเรย์ย่อยต้องมีจำนวนสมาชิกอย่างน้อย 1 ตัว)

วิธีการแก้ปัญหา

- บรู๊ซฟอร์ท (Brute Force)
- การแบ่งแยกและเอาชนะ (Divide and Conquer)

บรู๊ซฟอร์ท

```
1: maxsum = −∞
2: for i ← 1 to n do
3:     sum = 0
4:     for j ← i to n do
5:         sum = sum + a[j]
6:         if sum > maxsum then
7:             maxsum ← sum
8:         end if
9:     end for
10: end for
11: return maxsum
```

โดยที่ a เป็นอาเรย์ที่มีจำนวนสมาชิกเท่ากับ n

คำถาม เมื่อ $a = \{2, 4, -6, -2, 3, 8, -4, 1\}$ บรรทัดที่ 5 มีการคำนวณกี่ครั้ง

ตัวอย่าง 2

จงแสดงค่าผลรวมที่มากที่สุดของอาเรย์ย่อย โดยมีอาเรย์ให้ต่อไปนี้ ด้วยวิธีการแบ่งแยกและเอาชนะ

อัลกอริทึมอยู่ในอีกไฟล์

2	-6	4	-1	3	8	-4	1

1. รอบแรกแบ่งอาเรย์ออกเป็น 2 ส่วน คือ อาเรย์ $2, -6, 4, -1$ และ $3, 8, -4, 1$
2. รอบถัดมาแบ่งออกเป็น 2 อาเรย์ย่อย จำนวน 2 ชุด คือ อาเรย์ $2, -6, 4, -1$ แบ่งออกเป็น $2, -6$ และ $4, -1$ และอีกอาเรย์ $3, 8, -4, 1$ แบ่งออกเป็น $3, 8$ และ $-4, 1$
3. รอบต่อมา แบ่งอาเรย์ที่มีสมาชิก 1 ค่า

ในกรณีที่อาเรย์ที่มีสมาชิก 1 ค่า ค่าผลรวมที่มากที่สุดของอาเรย์ย่อยคือของสมาชิกนั้นเอง

การรวมอาเรย์ที่มีสมาชิก 1 ตัวและ 2 ตัว

4. การรวมอาเรย์ 2 และ -6 จะได้ $(2, -6)$ โดย lmax = 2, rmax = -6, cmax = -4 ดังนั้น maxsum = 2
5. การรวมอาเรย์ 4 และ -1 จะได้ $(4, -1)$ โดย lmax = 4, rmax = -1, cmax = 3 ดังนั้น maxsum = 4
6. การรวมอาเรย์ 3 และ 8 จะได้ $(3, 8)$ โดย lmax = 3, rmax = 8, cmax = 11 ดังนั้น maxsum = 11
7. การรวมอาเรย์ -4 และ 1 จะได้ $(-4, 1)$ โดย lmax = -4, rmax = 1, cmax = -3 ดังนั้น maxsum = 1
8. การรวมอาเรย์ $(2, -6)$ และ $(4, -1)$ จะได้ $(2, -6, 4, -1)$ โดย lmax = 2, rmax = 4, cmax = 0 (-4+4) ดังนั้น maxsum = 4
9. การรวมอาเรย์ $(3, 8)$ และ $(-4, 1)$ จะได้ $(3, 8, -4, 1)$ โดย lmax = 11, rmax = 1, cmax = 8 (11-3) ดังนั้น maxsum = 11

การรวมอาเรย์ที่มีสมาชิก 4 ตัว

10. การรวมอาเรย์ $(2, -6, 4, -1)$ และ $(3, 8, -4, 1)$ จะได้ $(2, -6, 4, -1, 3, 8, -4, 1)$ โดย lmax = 4, rmax = 11, cmax = 14 (3+11) ดังนั้น maxsum = 14

ลำดับ	อาเรย์ย่อย 1	อาเรย์ย่อย 2	ผลรวมของอาเรย์	lmax	rmax	cmax	maxsum
1	2	-6	(2,-6)	2	-6	-4	2
2	4	-1	(4,-1)	4	-1	3	4
3	3	8	(3,8)	3	8	11	11
4	-4	1	(-4,1)	-4	1	-3	1
5	(2,-6)	(4,-1)	(2,-6,4,-1)	2	4	0	4
6	(3,8)	(-4,1)	(3,8,-4,1)	11	1	8	11
7	(2,-6,4,-1)	(3,8,-4,1)	(2,-6,4,-1,3,8,-4,1)	4	11	14	14

ตัวอย่าง 3

จงแสดงค่าผลรวมที่มากที่สุดของอาเรย์ย่อย โดยมีอาเรย์ให้ต่อไปนี้ ด้วยวิธีการแบ่งแยกและเอาชนะ

$$(1, -6, 7, -5, 4, -1, 3, 8, -4, 1, -2, 5, -3, 6, -2, 1)$$

1. รอบแรกแบ่งอาเรย์ออกเป็น 2 ส่วน คือ อาเรย์ $1, -6, 7, -5, 4, -1, 3, 8$ และ $-4, 1, -2, 5, -3, 6, -2, 1$
2. รอบถัดมาแบ่งออกเป็น 2 อาเรย์ย่อย จำนวน 2 ชุด คือ อาเรย์ $1, -6, 7, -5, 4, -1, 3, 8$ แบ่งออกเป็น $1, -6, 7, -5$ และ $4, -1, 3, 8$ และอีกอาเรย์ $-4, 1, -2, 5, -3, 6, -2, 1$ แบ่งออกเป็น $-4, 1, -2, 5$ และ $-3, 6, -2, 1$

3. รอบถัดมาแบ่งออกเป็น 2 อาเรย์ย่อย จำนวน 4 ชุด
    - $1, -6, 7, -5$ แบ่งออกเป็น $1, -6$ และ $7, -5$
    - $4, -1, 3, 8$ แบ่งออกเป็น $4, -1,$ และ $3, 8$
    - $-4, 1, -2, 5$ แบ่งออกเป็น $-4, 1$ และ $-2, 5$
    - $-3, 6, -2, 1$ แบ่งออกเป็น $-3, 6$ และ $-2, 1$
4. รอบต่อมา แบ่งอาเรย์ที่มีสมาชิก 1 ค่า

ในกรณีที่อาเรย์ที่มีสมาชิก 1 ค่า ค่าผลรวมที่มากที่สุดของอาเรย์ย่อยคือของสมาชิกนั่นเอง

การรวมอาเรย์ย่อยที่มีสมาชิก 1 ตัว

5. การรวมอาเรย์ 1 และ -6 จะได้ $(1, -6)$ โดย lmax = 1, rmax = -6, cmax = -5 ดังนั้น maxsum = 1
6. การรวมอาเรย์ 7 และ -5 จะได้ $(7, -5)$ โดย lmax = 7, rmax = -5, cmax = 2 ดังนั้น maxsum = 7
7. การรวมอาเรย์ 4 และ -1 จะได้ $(4, -1)$ โดย lmax = 4, rmax = -1, cmax = 3 ดังนั้น maxsum = 4
8. การรวมอาเรย์ 3 และ 8 จะได้ $(3, 8)$ โดย lmax = 3, rmax = 8, cmax = 11 ดังนั้น maxsum = 11

การรวมอาเรย์ย่อยที่มีสมาชิก 1 ตัว (2)

9. การรวมอาเรย์ -4 และ 1 จะได้ $(-4, 1)$ โดย lmax = -4, rmax = 1, cmax = -3 ดังนั้น maxsum = 1
10. การรวมอาเรย์ -2 และ 5 จะได้ $(-2, 5)$ โดย lmax = -2, rmax = 5, cmax = 3 ดังนั้น maxsum = 5
11. การรวมอาเรย์ -3 และ 6 จะได้ $(-3, 6)$ โดย lmax = -3, rmax = 6, cmax = 3 ดังนั้น maxsum = 6
12. การรวมอาเรย์ -2 และ 1 จะได้ $(-2, 1)$ โดย lmax = -2, rmax = 1, cmax = -1 ดังนั้น maxsum = 1

การรวมอาเรย์ย่อยที่มีสมาชิก 2 ตัว

13. การรวมอาเรย์ $(1, -6)$ และ $(7, -5)$ จะได้ $(1, -6, 7, -5)$ โดย lmax = 1, rmax = 7, cmax = 2 (-5+7) ดังนั้น maxsum = 7
14. การรวมอาเรย์ $(4, -1)$ และ $(3, 8)$ จะได้ $(4, -1, 3, 8)$ โดย lmax = 4, rmax = 11, cmax = 14 (3+11) ดังนั้น maxsum = 14
15. การรวมอาเรย์ $(-4, 1)$ และ $(-2, 5)$ จะได้ $(-4, 1, -2, 5)$ โดย lmax = 1, rmax = 5, cmax = 4 (1+3) ดังนั้น maxsum = 5
16. การรวมอาเรย์ $(-3, 6)$ และ $(-2, 1)$ จะได้ $(-3, 6, -2, 1)$ โดย lmax = 6, rmax = 1, cmax = 5 (6-1) ดังนั้น maxsum = 6

การรวมอาเรย์ย่อยที่มีสมาชิก 4 ตัวและ 8 ตัว

17. การรวมอาเรย์ $(1, -6, 7, -5)$ และ $(4, -1, 3, 8)$ จะได้ $(1, -6, 7, -5, 4, -1, 3, 8)$ โดย lmax = 7, rmax = 14, cmax = 16 (2+14) ดังนั้น maxsum = 16
18. การรวมอาเรย์ $(-4, 1, -2, 5)$ และ $(-3, 6, -2, 1)$ จะได้ $(-4, 1, -2, 5, -3, 6, -2, 1)$ โดย lmax = 5, rmax = 6, cmax = 8 (5+3) ดังนั้น maxsum = 8
19. การรวมอาเรย์ $(1, -6, 7, -5, 4, -1, 3, 8)$ และ $(-4, 1, -2, 5, -3, 6, -2, 1)$ จะได้ $(1, -6, 7, -5, 4, -1, 3, 8, -4, 1, -2, 5, -3, 6, -2, 1)$ โดย lmax = 16, rmax = 8, cmax =19 (16+3) ดังนั้น maxsum = 19