
Longest Increasing Subsequence

การอบรมคอมพิวเตอร์โอลิมปิก สอวน. ค่ายที่ 2
ปีการศึกษา 2564

Pisit Makpaisit

Longest Increasing Subsequence

increasing subsequence คือ subsequence ที่ยาวที่สุดที่มีค่ามากขึ้นเรื่อยๆ
สำหรับอาร์เรย์ A ความยาว n

2	12	8	0	23	11	52	31	51	59	80	22
---	----	---	---	----	----	----	----	----	----	----	----

Longest Increasing Subsequence

ให้ $L(i)$ เป็นความยาวของ LIS ที่มี i เป็นตัวสุดท้ายใน subsequence

LIS ของอาร์เรย์ความยาว n หาได้จาก

$$\min_{1 \leq i \leq n} L(i)$$

Longest Increasing Subsequence

Base case

กรณีที่มีตัวเดียว (ตัวแรกเริ่มนับที่ 1)

$$L(1) = 1$$

General case

ไปดูว่า $L(k)$ เมื่อ $k < i$ มีตัวไหนที่มีค่ามากที่สุด ภายใต้เงื่อนไขว่า $A[k] < A[i]$

$$L(i) = 1 + \max(L(k)) \text{ where } 1 \leq k < i \text{ and } A[k] < A[i];$$

$$L(i) = 1, \text{ if no such } k \text{ exists.}$$

Longest Increasing Subsequence

เต็มตาราง n ช่อง

แต่ละช่องใช้เวลา n

Time complexity : $\Theta(n^2)$

Space complexity : $\Theta(n)$

ทำให้เร็วกว่านี้ได้หรือไม่?

Observations

กำหนดอาร์เรย์ [2, 5, 4, 8, 1, 9, 3]

(1) [2] - Candidate list

[2]

(2) [2, 5] - Candidate list

[2]

[5]

[2, 5]

(3) [2, 5, 4] - Candidate list

[2]

[4]

[5]

[2, 4]

[2, 5]

(4) [2, 5, 4, 8] - Candidate list

[2]

[4]

[5]

[8]

[2, 4]

[2, 5]

[2, 8]

[2, 4, 8]

[2, 5, 8]

...

Observations

ในชุดของ Candidate list มีบางอาร์เรย์ที่ dominate ตัวอื่น

(1) [2] - Candidate list
[2]

(2) [2, 5] - Candidate list
[2]
[5]
[2, 5]

(3) [2, 5, 4] - Candidate list
[2]
[4]
[5]
[2, 4]
[2, 5]

(4) [2, 5, 4, 8] - Candidate list

[2]
[4]
[5]
[8]
[2, 4]
[2, 5]
[2, 8]
[2, 4, 8]
[2, 5, 8]

- ใน candidate list ตัวที่มีความยาวเท่ากันจะมีตัวที่ดีที่สุดอยู่
- ในความยาวเท่ากันเก็บแค่ตัวเดียวที่ตัวสุดท้ายน้อยที่สุด

Observations

กำหนดอาร์เรย์ [2, 5, 4, 8, 1, 9, 3]

(1) [2] - Candidate list

[2]

(2) [2, 5] - Candidate list

[2]

[2, 5]

(3) [2, 5, 4] - Candidate list

[2]

[2, 4]

(4) [2, 5, 4, 8] - Candidate list

[2]

[2, 4]

[2, 4, 8]

(5) [2, 5, 4, 8, 1] - Candidate list

[1]

[2, 4]

[2, 4, 8]

(6) [2, 5, 4, 8, 1, 9] - Candidate list

[1]

[2, 4]

[2, 4, 8]

[2, 4, 8, 9]

Observations

กำหนดอาร์เรย์ [2, 5, 4, 8, 1, 9, 3]

(1) [2] - Candidate list
[2]

(2) [2, 5] - Candidate list
[2]
[2, 5]

(3) [2, 5, 4] - Candidate list
[2]
[2, 4]

(4) [2, 5, 4, 8] - Candidate list
[2]
[2, 4]
[2, 4, 8]

(5) [2, 5, 4, 8, 1] - Candidate list
[1]
[2, 4]
[2, 4, 8]

(6) [2, 5, 4, 8, 1, 9] - Candidate list
[1]
[2, 4]
[2, 4, 8]
[2, 4, 8, 9]

Case 1: $A[i]$ น้อยกว่าสมาชิกตัวที่มีความยาว 1
ให้สร้าง list ใหม่แทน list ความยาว 1 เดิม

Case 2: $A[i]$ มากกว่าตัวสุดท้ายของ list ที่ยาวที่สุด
สร้าง list ใหม่ที่ copy จาก list ที่ยาวที่สุด และนำ $A[i]$ เข้าไปต่อ

Case 3: กรณีอื่นๆ
หาว่า $A[i]$ น้อยกว่าตัวสุดท้ายของ list ไດ ให้ copy list ที่สั้นกว่าอยู่ 1 และนำ $A[i]$ เข้าไปต่อ

Observations

ไม่จำเป็นที่จะต้องเก็บ list ทั้งหมด
สามารถเก็บแค่ความยาว และค่าของ
ตัวสุดท้าย

(1) [2] - Candidate list

[2]	(1, 2)
-----	--------

(2) [2, 5] - Candidate list

[2]	(1, 2)
[2, 5]	(2, 5)

(3) [2, 5, 4] - Candidate list

[2]	(1, 2)
[2, 4]	(2, 4)

(4) [2, 5, 4, 8] - Candidate list

[2]	(1, 2)
[2, 4]	(2, 4)
[2, 4, 8]	(3, 8)

(5) [2, 5, 4, 8, 1] - Candidate list

[1]	(1, 1)
[2, 4]	(2, 4)
[2, 4, 8]	(3, 8)

(6) [2, 5, 4, 8, 1, 9] - Candidate list

[1]	(1, 1)
[2, 4]	(2, 4)
[2, 4, 8]	(3, 8)
[2, 4, 8, 9]	(4, 9)

ค่าของตัวสุดท้ายจะเรียงกันเสมอ (ใช้ binary search ในการหาได้)

Longest Increasing Subsequence

Algorithm 39 logLinearTimeLongestIncreasingSubsequence

```
1: function LIS( $A[1..n]$ )
2:    $L[1] \leftarrow A[1]$ 
3:    $LISlen \leftarrow 0$ 
4:   for  $i = 2$  to  $n + 1$  do
5:      $k \leftarrow \text{BinarySearch}(L, 1, i - 1, A[i])$       ▷ ค้นหา  $A[i]$  ใน  $L$  ที่ตำแหน่ง 1 ถึง  $i - 1$  (inclusion)
6:      $L[k] \leftarrow A[i]$ 
7:     if  $k > LISlen$  then
8:        $LISlen \leftarrow k$ 
9:   return  $LISlen$ 
```

Longest Increasing Subsequence

เวอร์ชันใหม่ สร้างอาร์เรย์ L ความยาว n สำหรับเก็บตัวสุดท้ายของ candidate list

มีการวนลูป n รอบ แต่ละรอบค้นหาค่าใน L ด้วย binary search ($O(\log n)$)

Time complexity : $O(n \log n)$

Space complexity : $\Theta(n)$