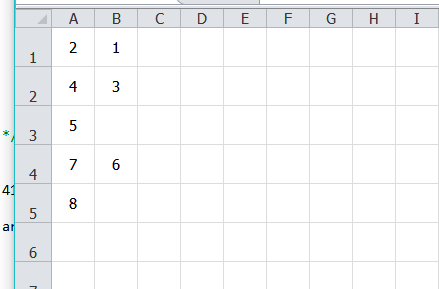
**Jump World**

เราจะทำ LIS แบบข้อ Tarn Sequence กัน หรือก็คือทำ LIS แบบเก็บลง vector ด้วยนั่นแหละ

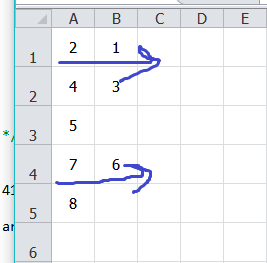
เรามาดูข้อสังเกตบางอย่างของเลขใน vector กัน

สมมุติว่าเลขเป็น 2,4,1,5,3,7,6,8

Vector ที่ได้จะเป็น



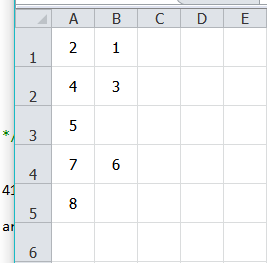
ข้อสังเกต : ในแต่ละแถว เลขจะมีค่าลดลงจากซ้ายไปขวาเสมอ



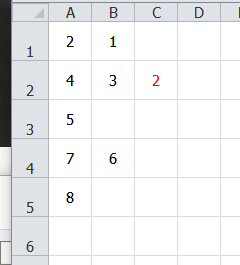
วิธีการของข้อนี้คือ เราจะมี vector แบบนี้ กับ vector อีกอันสำหรับเก็บจำนวนวิธี ที่มีหน้าตาเหมือนกัน

1. เราจะทำ LIS ตามปกติ ก็คือ lower\_bound บนอาเรย์ เปลี่ยนค่าในอาเรย์, ยัดลง vector เหมือน Tarn Sequence

2. เราจะทำการยัดจำนวนวิธีลงไปด้วย สมมุติว่าเรามีเลข 2 เพิ่มขึ้นมาจากเดิม

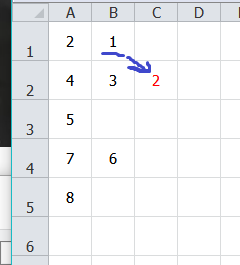


มันจะเป็นแบบนี้ใช่ไหมครับ



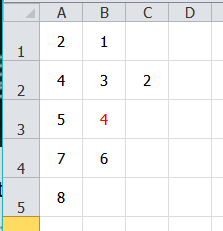
แล้วทีนี้ เราดูว่าเจ้าเลข 2 ตัวนี้ ไปต่อท้ายใคร แล้วได้ “ลำดับเพิ่มความยาว 2 ตัวที่ลงท้ายด้วยเลข 2 ตัวนี้”

สังเกตว่า เราต้องเอาเลข 2 ตัวนี้ไปต่อท้าย พวกลำดับเพิ่มความยาว 1 เท่านั้น และต้องลงท้ายด้วยเลขที่น้อยกว่า 2 ซึ่งตัวที่เราจะเอาไปต่อท้ายได้ก็คือ



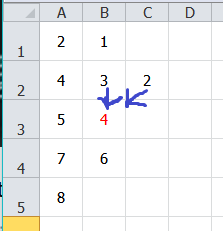
ดังนั้น เราก็เอาจำนวนวิธีในการสร้างลำดับเพิ่มความยาว 1 ที่ลงท้ายด้วย 1 มานั่นเอง

ทีนี้สมมุติว่า มีเลข 4 มาต่อท้ายอีก จะกลายเป็น



เราก็ดูว่าเจ้าเลข 4 ตัวนี้ ไปต่อท้ายใคร แล้วได้ “ลำดับเพิ่มความยาว 3 ตัวที่ลงท้ายด้วยเลข 4 ตัวนี้”

ซึ่งตัวที่เราเอาไปต่อแล้วได้ตามนี้ก็คือ



ก็คือ เอาลำดับเพิ่มความยาว 2 ตัวที่ลงท้ายด้วย 3 หรือ 2 มาต่อท้ายนั่นเองงง (เราเอาลงท้ายด้วย 4 มาต่อไม่ได้ เพราะมันจะไม่ใช่ลำดับเพิ่ม)

ดังนั้น จำนวนวิธีจะเท่ากับวิธีสร้างลำดับเพิ่มความยาว 2 ที่ลงท้ายด้วย 3 หรือ 2 นั่นเอง

ทีนี้ เราจะคำนวณจำนวนวิธีพวกนี้เร็วๆอย่างไรล่ะ?

1. เราจะทำ quicksum จำนวนวิธีเอาไว้ จากซ้ายไปขวา

2. เราจะใช้ข้อสังเกตที่ว่า เลขในแถวๆนึงจะเรียงจากมากไปน้อยเสมอ

ดังนั้น เราสามารถ binary search หาตัวแรกที่มีค่า <= เลขล่าสุดที่ใส่เข้ามา แล้วหัก quicksum ของเลขก่อนหน้าตัวนั้นออก หรือก็คือ

= qs[idx-1][ช่องสุดท้ายของ vector แถวที่ idx-1] – qs[idx-1][ช่องแรกที่มีค่าน้อยกว่าเรา-1]

= จำนวนวิธีการสร้าง LIS ความยาว i-1 ที่ลงด้วยด้วยเลขใน vector แถวที่ i-1 ตัวที่น้อยกว่าเราเป็นต้นไป