

## แบบฝึกหัดอัลกอริทึมเชิงละโมภ

1. จากปัญหา fractional knapsack จงเขียนโปรแกรมแบบ greedy เพื่อค้นหาเซตของ item ที่มีน้ำหนัก (weight) รวม ไม่เกินขนาดของถุง (weight)  $W$  และมีมูลค่ารวม (value) สูงสุด

### ข้อมูลนำเข้า

บรรทัดที่ 1 จำนวนเต็ม  $n$   $W$  แทนจำนวน item และน้ำหนักของถุง โดยที่  $1 < n \leq 10$  และ  $1.0 \leq W \leq 3,000.0$  คั่นด้วยช่องว่าง

บรรทัดที่ 2 รายการจำนวนเต็ม  $n$  รายการ แทนน้ำหนักของ item แต่ละชิ้น

บรรทัดที่ 3 รายการจำนวนเต็ม  $n$  รายการ แทนมูลค่าของ item แต่ละชิ้น

### ข้อมูลส่งออก

เซต  $n$  แสดงสัดส่วนการเลือก item แต่ละรายการเป็นคำตอบ โดยที่ 0.0 หมายถึง item ไม่ถูกเลือก และ 1.0 หมายถึงเลือกทั้งชิ้น ทศนิยม 2 ตำแหน่ง

บรรทัดถัดไปมูลค่ารวมสูงสุดของ item ที่ถูกเลือก ทศนิยม 2 ตำแหน่ง

ตัวอย่างข้อมูลนำเข้า	ตัวอย่างข้อมูลส่งออก
4 25 18 15 10 5 25 24 5 8	0.28 1.00 0.00 1.00 38.94
3 5 1 2 3 1 2 2	1.00 1.00 0.67 4.33

2. ต้องการจัดเก็บไฟล์ข้อมูลจำนวน  $n$  ไฟล์ ซึ่งมีความยาวของแต่ละไฟล์แทนด้วย  $l_1, l_2, l_3, \dots, l_n$  ลงบนเทป (tape) บันทึกข้อมูลความยาวไม่จำกัด โชคดีที่ว่าการเข้าถึงไฟล์ข้อมูลแต่ละไฟล์จะต้องเริ่มจากต้นเทปเสมอ ซึ่งจะส่งผลกระทบต่อเวลารวมเฉลี่ยในการเข้าถึงไฟล์ (mean retrieval time) ทุกไฟล์บนเทป ตัวอย่างเช่น สมมติให้  $f_1, f_2$  และ  $f_3$  มีความยาวของไฟล์ 10, 5 และ 13 ตามลำดับ

- หากจัดเก็บไฟล์  $f_1$  ตามด้วย  $f_2$  และ  $f_3$  จะได้เวลารวมเฉลี่ยของการเข้าถึงทั้ง 3 ไฟล์ คือ  $(10 + (10+5) + (10+5+13))/3 = 17.67$
- หากจัดเก็บไฟล์  $f_2$  ก่อนตามด้วย  $f_1$  และ  $f_3$  จะทำให้เวลารวมเฉลี่ยมีค่าเท่ากับ 16.00

จงเขียนโปรแกรม greedy เพื่อหาวิธีจัดเก็บไฟล์เหล่านี้ให้เวลารวมเฉลี่ยมีค่าน้อยที่สุด

### ข้อมูลนำเข้า

บรรทัดแรก จำนวนเต็ม  $n$  แทนจำนวนไฟล์ที่ต้องการจัดเก็บ  $1 < n \leq 100$

## แบบฝึกหัดอัลกอริทึมเชิงละโมภ

บรรทัดสอง รายการจำนวนเต็ม  $n$  รายการ แทนความยาวของแต่ละไฟล์  $1 \leq f_i \leq 1,000$  คั่นด้วยช่องว่าง

### ข้อมูลส่งออก

เวลารวมเฉลี่ยที่น้อยที่สุดในการเข้าถึงทุกไฟล์บนเทป ทศนิยม 2 ตำแหน่ง

ตัวอย่างข้อมูลนำเข้า	ตัวอย่างข้อมูลส่งออก
3 10 5 13	16.00
6 7 8 15 6 12 9	28.17

3. ตารางเวลารถไฟ  $n$  ขบวน แต่ละขบวนจะประกอบไปด้วย เวลามาถึง (arrival time) และเวลาออก (departure time) หากต้องการสร้างชานชลา (platform) เพื่อให้รถไฟแต่ละขบวนสามารถเข้าออกได้ตามเวลาที่ระบุโดยไม่มีการ delay จงเขียนโปรแกรมแบบ greedy เพื่อค้นหาจำนวนชานชลาที่น้อยที่สุดจากตารางเวลาของรถไฟที่กำหนดให้

### ข้อมูลนำเข้า

บรรทัดที่ 1 จำนวนเต็ม  $n$  แทนจำนวนรถไฟที่เข้าออกสถานี โดยที่  $1 \leq n \leq 500$

$n$  บรรทัดถัดไป แต่ละบรรทัดแสดงทศนิยม  $t_a$   $t_b$  แทนเวลาเข้าและออกของรถไฟแต่ละขบวน คั่นด้วยช่องว่าง โดยที่  $t_a < t_b$

### ข้อมูลส่งออก

จำนวนชานชลาที่น้อยที่สุดที่รถไฟแต่ละขบวนสามารถเข้าออกได้โดยไม่เกิดการ delay

ตัวอย่างข้อมูลนำเข้า	ตัวอย่างข้อมูลส่งออก
6 2.00 2.30 2.10 3.40 3.00 3.20 3.20 4.30 3.50 4.00 5.00 5.20	2
3 9.00 9.40	1

แบบฝึกหัดอัลกอริทึมเชิงโมบ

9.10 12.00	
12.01 13.50	

4. เส้นตรง (line) ประกอบไปด้วยจุดเริ่มต้น  $x_i$  และจุดสิ้นสุด  $x_j$  แทนด้วย  $(x_i, x_j)$  โดยที่  $x_i \leq x_j$  หากมีเส้นตรงทั้งหมด  $n$  เส้น และต้องการรวมเส้นตรงเหล่านี้เป็นเส้นเดียวเป็น  $(x_a, x_b)$  โดยใช้เส้นตรงเหล่านี้ให้น้อยที่สุด ตัวอย่างเช่น สมมติมีเส้นตรง 10 เส้น ได้แก่  $(1,2), (3,5), (1,5), (2,4), (4,5), (3,6), (2,7), (7,9), (4,8), (1,3)$  หากต้องการรวมเป็นเส้นตรง  $(1,9)$  คำตอบที่น้อยที่สุดคือ 3 เส้น ได้แก่  $(1,5), (4,8), (7,9)$  จงเขียนโปรแกรม greedy เพื่อหาคำตอบดังกล่าว

**ข้อมูลนำเข้า**

บรรทัดที่ 1 จำนวนเต็ม  $n$  แทนจำนวนเส้นตรง โดยที่  $1 \leq n \leq 500$

$n$  บรรทัดถัดไป แต่ละบรรทัดแสดงเส้นตรงแทนด้วยคู่ลำดับของจุดเริ่มต้นและจุดสิ้นสุด  $x_i$  และ  $x_j$

คั่นด้วยช่องว่าง โดยที่  $-50,000 \leq x_i, x_j \leq 50,000$  และ  $x_i \leq x_j$

บรรทัดสุดท้าย แทนด้วยคู่ลำดับ  $x_a x_b$  แทนเส้นตรงที่ต้องการ

**ข้อมูลส่งออก**

จำนวนเส้นตรงที่น้อยที่สุดซึ่งถูกเลือกเพื่อสร้างเส้นตรง  $(x_a, x_b)$

บรรทัดถัดไป แต่ละบรรทัด แทนรายการเส้นตรง  $(x_i, x_j)$  ที่ถูกเลือก เรียงลำดับตาม  $x_i$  และ  $x_j$  น้อยไปมาก

ตัวอย่างข้อมูลนำเข้า	ตัวอย่างข้อมูลส่งออก
10	3
1 2	1 5
3 5	4 8
1 5	7 9
2 4	
4 5	
3 6	
2 7	
7 9	
4 8	
1 3	
1 9	

## แบบฝึกหัดอัลกอริทึมเชิงละโมภ

5. ตารางด้านล่างแสดงกิจกรรมที่ต้องการใช้ห้องประชุมในช่วงเวลาต่างๆ

กิจกรรม	1	2	3	4	5	6	7	8	9	10
เวลาเริ่ม	1	3	0	5	3	5	6	8	8	2
เวลาสิ้นสุด	4	5	6	7	8	9	10	11	12	13

หากมีห้องประชุมว่างทั้งหมด  $k$  ห้อง จงเขียนโปรแกรมแบบ greedy เพื่อค้นหาจำนวนกิจกรรมสูงสุดที่สามารถจัดได้ กำหนดให้ greedy criterion จะเลือกกิจกรรมที่ใช้เวลานานที่สุดก่อน (longest first strategy)

**ข้อมูลนำเข้า**

บรรทัดที่ 1 จำนวนเต็ม  $n$  และ  $k$  แทนจำนวนกิจกรรมและจำนวนห้องประชุม

$n+1$  บรรทัดถัดไป คู่ลำดับจำนวนเต็ม  $s$  และ  $t$  แทนเวลาเริ่มต้นและเวลาสิ้นสุดของแต่ละกิจกรรม

โดย  $1 \leq s, t \leq 20$  และ  $s < t$

**ข้อมูลส่งออก**

จำนวนกิจกรรมสูงสุดที่สามารถจัดได้

ตัวอย่างข้อมูลนำเข้า	ตัวอย่างข้อมูลส่งออก
10 3 1 4 3 5 0 6 5 7 3 8 5 9 6 10 8 11 8 12 2 13	5