

แบบฝึกหัดภาคปฏิบัติการ สัปดาห์ที่เก้า

อาเรย์หนึ่งมิติ

อ.ดร.ภิญโญ แท้ประสาทสิทธิ์ ภาควิชาคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยศิลปากร

คำสั่ง

1. ให้เขียนโปรแกรมภาษาซีสำหรับปัญหาที่ให้ไป
2. ระเบียบการส่งงานเขียนโปรแกรมก็คือนักศึกษาจะส่งโค้ดเข้าไปโปรแกรมตรวจงาน

จุดประสงค์การเรียนรู้ การใช้อาเรย์หนึ่งมิติคู่กับการใช้ลูปทั้งแบบชั้นเดียวและสองชั้น

ปัญหา 1 พิมพ์เลขย้อนกลับ [ReversePrint]

จงเขียนโปรแกรมที่รับค่า N มาค่าหนึ่ง จากนั้นโปรแกรมจะรับเลขจำนวนเต็มมาอีก N ค่า เมื่อรับเสร็จแล้วโปรแกรมจะพิมพ์เลขทั้งหมดออกมาย้อนหลังไปหน้า โดยเลขแต่ละตัวเว้นด้วยช่องว่างหนึ่งช่อง

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
5 1 2 3 4 5	5 4 3 2 1
6 -2 5 7 0 -1 4	4 -1 0 7 5 -2

หมายเหตุ ข้อมูลเข้าตัวแรกจะเป็นจำนวนเต็มบวกแน่ ๆ ดังนั้นไม่ต้องคอยตรวจว่า N เป็นจำนวนเต็มบวกมีค่าไม่เกิน 10,000

ปัญหา 2 การแข่งขันประเภททีม [TeamCompetition]

ในการแข่งขันเกมประเภททีมเกมหนึ่ง ผู้เล่นจะถูกแบ่งออกเป็นสองทีม ทีมละ N คน คนที่ 1 ของทีมแรกจะแข่งกับคนที่ 1 ของทีมที่สอง ในทำนองเดียวกัน คนที่ 2 ของแต่ละทีมก็จะมาแข่งกัน และเป็นเช่นนี้จนครบทั้งหมด N คู่ กติกามีอยู่ว่าผู้เข้าแข่งขันแต่ละคนจะทำแต้มตั้งแต่ 0 ถึง 10 คะแนน ถ้าใครทำคะแนนมากกว่าคู่แข่งของตัวเองจะถือว่าชนะในเกมนั้น ถ้าทำได้น้อยกว่าถือว่าแพ้ และถ้าทำได้เท่ากันถือว่าเสมอ

การชนะในแต่ละเกมจะทำคะแนนให้กับทีมของผู้ชนะ 2 คะแนน การเสมอได้ 1 คะแนน และการแพ้จะไม่ได้คะแนน ทีมที่มีคะแนนรวมมากกว่าเป็นฝ่ายชนะ ทีมที่มีคะแนนรมน้อยกว่าแพ้ และ หากทั้งสองทีมได้คะแนนรวมเท่ากันถือว่าเป็นผลเสมอ

จงเขียนโปรแกรมที่รับแต้มของผู้เข้าแข่งขันแต่ละคนและสรุปผลการแข่งขันประเภททีมออกมาให้ได้ตามข้อกำหนดเรื่องข้อมูลเข้าและผลลัพธ์ต่อไปนี้

ข้อมูลเข้า

1. บรรทัดแรกเป็นตัวเลขจำนวนเต็ม N ซึ่งระบุจำนวนผู้เข้าแข่งขันของแต่ละทีม ($N \leq 1000$)
2. บรรทัดที่สองคือแต้มที่ผู้เข้าแข่งขันในทีมที่หนึ่งทำได้ เรียงจากคนที่ 1 ไปคนที่ 2, 3, ..., N โดยแต้มเป็นจำนวนเต็ม แต้มของแต่ละคนถูกคั่นด้วยช่องว่าง
3. บรรทัดที่สามคือแต้มที่ผู้เข้าแข่งขันทีมที่สองทำได้ เรียงจากคนแรกไปคนสุดท้ายในลักษณะเดียวกับคะแนนของทีมที่หนึ่ง

ผลลัพธ์

1. บรรทัดแรกระบุทีมที่ชนะ [ดูรูปแบบผลลัพธ์ในตัวอย่างทางด้านใต้]
2. บรรทัดที่สองระบุคะแนนของทั้งสองทีม โดยนำคะแนนของทีมที่ชนะขึ้นก่อน ในกรณีที่เสมอให้แสดงคะแนนของทีมไหนก่อนก็ได้ (เพราะคะแนนเท่ากันลำดับจึงไม่มีผล)

ตัวอย่าง

ตัวอย่าง	ข้อมูลเข้า	ผลลัพธ์
1	5 1 2 3 1 4 3 9 3 5 6	Team 2 wins Score 9 to 1 [คำอธิบาย: ทีมที่สอง ชนะ 4 ครั้ง เสมอ 1 ครั้ง จึงมี 9 แต้ม ส่วนทีมแรก เสมอครั้งเดียวนอกนั้นแพ้หมดจึงมีเพียง 1 แต้ม]
2	5 7 9 3 5 6 3 8 7 8 5	Team 1 wins Score 6 to 4 [คำอธิบาย: ทีมที่หนึ่ง ชนะ 3 ครั้ง แพ้ 2 ครั้ง จึงมี 6 แต้ม ส่วนทีมที่สอง ชนะ 2 ครั้ง แพ้ 3 ครั้ง จึงมี 4 แต้ม]
3	7 7 8 7 8 5 5 9 9 9 7 4 4 6 6	Draw game Score 7 to 7 [คำอธิบาย: ทั้งสองทีมชนะ 3 แพ้ 3 เสมอ 1 มี 7 แต้มเท่ากัน ผลการแข่งขันจึงออกมาเป็นเสมอ]

คำแนะนำ ข้อนี้จะใช้อาเรย์หนึ่งมิติหนึ่งหรือสองอันก็ได้ สำหรับหลายคนอาจจะรู้สึกว่าการใช้อาเรย์หนึ่งมิติสองอันดูเป็นธรรมชาติมากกว่า ในที่นี้จะอธิบายวิธีคิดที่มีอาเรย์สองอันคือ A และ B

1. อ่านค่า N เข้ามา
2. วนรับค่าแถวแรกเข้าไปเก็บไว้ใน A
3. วนรับค่าแถวที่สองเข้าไปเก็บไว้ใน B (คือตอนนี้จะมีรูปหนึ่งชั้นสองอันแยกกัน)
4. วนลูปอีกครั้งอ่านค่าจาก $A[i]$ และ $B[i]$ และนำค่าทั้งสองมาเปรียบเทียบกัน
5. นำผลการเปรียบเทียบที่ได้ไปคิดเป็นคะแนนของแต่ละทีม

ปัญหา 3 ตำแหน่งของเลขที่สนใจ [NumberOccurrence]

กำหนดชุดตัวเลข $x_1, x_2, x_3, \dots, x_N$ เช่น 7, 9, 2, 9, 7, 10, 2, 9, 3, 9 ในปัญหานี้ผู้ใช้สนใจที่จะหาว่า ตัวเลขที่ตนสนใจปรากฏอยู่ลำดับที่เท่าใดบ้าง เช่น ถ้าสนใจเลข 9 จากตัวอย่างเราก็จะได้ว่าเลข 9 ปรากฏเป็นลำดับที่ 2, 4, 8 และ 10 ส่วนถ้าสนใจเลข 7 ลำดับการปรากฏของมันก็จะเป็ลำดับที่ 1 และ 5

จงเขียนโปรแกรมที่รับชุดตัวเลข เมื่อรับชุดตัวเลขเสร็จแล้ว ให้รับตัวเลขที่ผู้ใช้สนใจ จากนั้นโปรแกรมจะต้องพิมพ์ลำดับการปรากฏของตัวเลขที่ผู้ใช้สนใจออกมาทั้งหมด ในกรณีที่ตัวเลขที่ผู้ใช้สนใจไม่ปรากฏอยู่ในชุดตัวเลขเลย ให้พิมพ์เลข 0 ออกมา

ข้อมูลเข้า

- บรรทัดแรกคือจำนวนเต็ม N ซึ่งเป็นตัวระบุความยาวของชุดตัวเลข โดยที่ $1 < N \leq 2,500,000$ (ความยาวคือจำนวนค่าตัวเลข เช่น ในตัวอย่างข้างบนจะมีค่า N เป็น 10)
- บรรทัดที่สองเป็นเลขจำนวนเต็มในชุดตัวเลขทั้งหมด โดยที่ $0 < x_i < 1,000$ เลขแต่ละค่าถูกคั่นด้วยช่องว่างหนึ่งช่อง
- บรรทัดที่สามเป็นเลขจำนวนเต็มหนึ่งตัวที่ผู้ใช้สนใจจะตรวจสอบดูว่ามันปรากฏอยู่ที่ลำดับเท่าใดบ้าง

ผลลัพธ์

มีหนึ่งบรรทัดแสดงลำดับการปรากฏของตัวเลขที่สนใจ ค่าลำดับแต่ละค่าจะถูกคั่นด้วยช่องว่างหนึ่งช่อง แต่ถ้าเลขที่ผู้ใช้สนใจไม่ปรากฏในลำดับเลย โปรแกรมจะพิมพ์เลขศูนย์ออกมาเพียงตัวเดียว

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
10 7 9 2 9 7 10 2 9 3 9 7	1 5
10 7 9 2 9 7 10 2 9 3 9 9	2 4 8 10
15 3 5 7 9 2 3 4 7 3 2 1 100 333 777 100 100	12 15
15 3 5 7 9 2 3 4 7 3 2 1 100 333 777 100 6	0

คำแนะนำ เนื่องจากอาเรย์มีขนาดใหญ่ เราควรสร้างมันไว้ภายนอกฟังก์ชัน main ไม่เช่นนั้นพื้นที่หน่วยความจำภายใน main จะไม่เพียงพอที่จะสร้างอาเรย์ได้

ปัญหา 4 บันไดตัวเลข (NumberStairs2)

ในปัญหานี้ เราต้องการพิมพ์ข้อความออกมาเป็นรูปบันได โดยบันไดแต่ละชั้นเป็นตัวเลขจำนวนเต็มหนึ่งหลัก เรียงต่อกันไป [ดูตัวอย่างท้ายโจทย์ประกอบ] จำนวนชั้นถูกกำหนดด้วยจำนวนเลขโดดที่รับเข้ามา โดยโปรแกรมของเราจะต้องรับเลขโดด¹ ที่มีค่าตั้งแต่ 0 ถึง 9 เข้ามาเรื่อย ๆ จนกว่าจะพบเลขติดลบหรือเลขที่มีค่าตั้งแต่ 10 ขึ้นไป จากนั้นจึงพิมพ์บันไดตัวเลขออกมาเป็นผลลัพธ์จากเลขโดดที่ผู้ใช้ใส่เข้ามา

ข้อมูลเข้า

มีบรรทัดเดียว เป็นจำนวนเต็มไม่ทราบจำนวนที่แน่นอนล่วงหน้า ทราบแต่เพียงว่ามีไม่เกิน 501 ตัว และมีไม่น้อยกว่า 2 ตัว โดยตัวสุดท้ายจะเป็นเลขติดลบหรือไม่ก็เลขที่มีค่าตั้งแต่ 10 ขึ้นไป ส่วนเลขก่อนถึงตัวสุดท้ายเป็นเลขโดดมีค่าตั้งแต่ 0 ถึง 9 [รวมเลข 0 และ 9 ด้วย] เลขแต่ละตัวคั่นด้วยช่องว่างหนึ่งช่อง

ผลลัพธ์

เป็นบันไดตัวเลขที่พิมพ์ออกมาเป็นข้อความหลายบรรทัด โดยบรรทัดชั้นบนสุดมีตัวเลขเพียงตัวเดียว ส่วนชั้นต่ำลงมาจะมีจำนวนตัวเลขเพิ่มขึ้นเรื่อย ๆ ชั้นละหนึ่งตัว มีลักษณะดังแสดงในตัวอย่างด้านใต้ **หมายเหตุ** การพิมพ์ช่องว่างเป็นสิ่งที่สำคัญต้องมีจำนวนที่ถูกต้องพอดี และในผลลัพธ์ที่ถูกต้องนั้น บรรทัดสุดท้ายจะไม่มีช่องว่างหน้าตัวเลขเลย

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
3 5 1 0 -2	3 55 111 0000
0 1 5 3 10	0 11 555 3333
7 8 2 7 3 0 2 7 0 523	7 88 222 7777 33333 000000 2222222 77777777 000000000

คำแนะนำ ข้อนี้ใช้อาเรย์หนึ่งมิติก็จริง แต่ต้องใช้ลูปสองชั้นในการคำนวณ

1 เลขโดดคือจำนวนที่มีเลขอยู่แค่ตัวเดียว

ปัญหา 5 พิมพ์กราฟแท่งแนวตั้ง เวอร์ชันพื้นฐาน (vertical_bar_graph_basic)

การวาดกราฟเป็นการแสดงความมากน้อยของจำนวนที่มีความสำคัญมาก แม้ในปัจจุบันการแสดงผลแบบกราฟฟิกจะเป็นแนวทางหลักในการวาดกราฟ แต่แนวคิดของการวาดหลายประการก็ยังมีลักษณะเดียวกันกับการวาดด้วยตัวอักษร

จงเขียนโปรแกรมที่พิมพ์กราฟแท่งจำนวน N แท่ง โดยความสูงของกราฟแต่ละแท่งจะเป็นไปตามความมากน้อยของค่าที่ต้องการแสดง เช่น หากค่าที่ต้องการแสดงคือ 12 กราฟแท่งของเลข 12 นี้ก็จะมีจำนวนเครื่องหมายดอกจันทั้งหมด 12 อัน แต่ถ้าหากค่าที่ต้องการแสดงคือ 8 ก็จะมีเครื่องหมายดอกจันของกราฟแท่งดังกล่าวเพียง 8 อัน เป็นต้น

ข้อมูลเข้า

- บรรทัดแรกคือจำนวนเต็ม N โดยที่ $N \leq 100$
- บรรทัดที่สองคือจำนวนเต็มบวก N ตัว แต่ละตัวแทนความสูงของกราฟแต่ละแท่ง เรียงจากซ้ายไปขวา เลขแต่ละตัวคั่นด้วยช่องว่างหนึ่งช่อง นอกจากนี้เลขแต่ละตัวมีค่าไม่เกิน 200

ผลลัพธ์

เป็นกราฟแท่งที่ความสูงของแต่ละแท่งถูกแทนด้วยจำนวนเครื่องหมายดอกจันในแท่งนั้น ๆ นอกจากนี้กราฟแท่งสองอันที่อยู่ติดจากกันจะไม่มีช่องว่างมาคั่น (ดูตัวอย่างประกอบ)

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
5 4 5 8 1 3	* * * ** *** *** * *** * *****
6 1 3 2 4 3 3	* * *** ***** *****

หมายเหตุ ตัวอย่างชี้ให้เห็นว่ากราฟทุกแท่งจะมีฐานอยู่บรรทัดเดียวกันทั้งหมด และบรรทัดแรกของผลลัพธ์ที่พิมพ์ออกมาจะต้องมีเครื่องหมายดอกจันของกราฟที่สูงที่สุดด้วย

คำแนะนำ ข้อนี้ควรใช้อาเรย์หนึ่งมิติ ส่วนลูปนั้นจะมีอยู่สองชุด คือชุดที่ใช้อ่านข้อมูลเข้าจะเป็นลูปชั้นเดียว แต่ตอนพิมพ์กราฟจะเป็นลูปอีกชุดซึ่งเป็นลูปสองชั้น

ปัญหา 6 หาที่ตั้งร้านขายของริมทาง (road_side_shop)

นักลงทุนรายหนึ่งต้องการเปิดร้านขายของบนถนนที่มีความยาวทั้งหมด N ช่วง แต่ละช่วงจะมีประชากรอยู่เป็นจำนวนต่าง ๆ กันไป เขามีสมมติฐานว่า โดยปรกติแล้วลูกค้าที่จะมาซื้อของที่ร้านจะอยู่ไม่ไกลจากร้านเกินระยะ K ช่วงถนน และเพื่อให้ร้านมีลูกค้าได้มากที่สุด เขาจึงพยายามหาตำแหน่งของร้านที่จะครอบคลุมช่วงถนนให้ได้จำนวนประชากรรวมมากที่สุดเท่าที่เป็นได้ และสิ่งแรกที่เขาต้องการทราบให้ได้ก่อนก็คือว่า จำนวนลูกค้าที่มากที่สุดที่จะมาซื้อของมีกี่คน

จงเขียนโปรแกรมที่คำนวณหาจำนวนลูกค้าที่มากที่สุดที่จะมาซื้อของที่ร้าน โดยจำนวนลูกค้านับจากจำนวนประชากรบนช่วงถนนที่อยู่ห่างจากร้านไม่เกิน K ช่วงถนน [มีตัวอย่างและคำอธิบายด้านล่างประกอบความเข้าใจ]

ข้อมูลเข้า

- บรรทัดแรกคือจำนวนเต็ม N และ K โดยที่ $1 \leq N \leq 10,000$ และ $1 \leq K \leq 100$
- บรรทัดที่สองเป็นจำนวนเต็มบวกหรือศูนย์แสดงจำนวนประชากรในถนนแต่ละช่วง เรียงจากช่วงแรกไปช่วงสุดท้ายตามลำดับ เลขแต่ละตัวคั่นด้วยช่องว่างหนึ่งช่อง และมีค่าไม่เกิน 500

ผลลัพธ์

เป็นเลขจำนวนเต็มแสดงจำนวนประชากรรวมสูงสุดที่เป็นไปได้ ซึ่งอยู่ในบริเวณที่จะมาซื้อของที่ร้าน

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
5 1 3 2 1 0 3	6
8 1 0 4 4 1 0 5 0 5	10
8 2 0 4 4 1 0 5 0 5	14

อธิบายตัวอย่างที่หนึ่ง ถนนมีอยู่ทั้งหมด 5 ช่วง ขอบเขตการเดินทางมาใช้บริการที่ร้านของลูกค้าจะไม่เกิน 1 ช่วงถนน ดังนั้นถ้าตั้งร้านไว้ที่ช่วงถนนที่สอง ประชากรที่อยู่ในช่วงที่หนึ่ง สอง และสามจะมาซื้อของได้ ดังนั้นจำนวนประชากรรวมในกรณีที่ตั้งร้านไว้ ณ ช่วงถนนที่สองจึงเป็น $3 + 2 + 1 = 6$ แต่ถ้าตั้งร้านในช่วงถนนที่สามจะได้จำนวนลูกค้าเป็น $2 + 1 + 0 = 3$ ถ้าตั้งในช่วงถนนที่สี่จะได้จำนวนลูกค้าเป็น $1 + 0 + 3 = 4$ ดังนั้นการตั้งร้านในช่วงถนนที่สองจะได้จำนวนลูกค้ารวมสูงสุดคือ 6 คน ผลลัพธ์จึงเป็น 6 ทั้งนี้ขอให้สังเกตด้วยว่าการตั้งร้านที่ติดขอบหรือใกล้ช่วงถนนด้านปลายทั้งสองมากเกินไปจะไม่มีทางให้ผลลัพธ์ที่ดีกว่าเดิม เราจึงไม่จำเป็นที่จะต้องนำผลลัพธ์เหล่านั้นมาพิจารณาด้วย

คำแนะนำ ควรใช้อาเรย์หนึ่งมิติและลูปสองชั้นในการแก้ปัญหา และให้ระมัดระวังที่ชี้ช่องอาเรย์มีค่าติดลบหรือเกินกว่าขอบเขตของอาเรย์ ความผิดพลาดของค่าดัชนีจะทำให้โปรแกรมค้างและเกิดเป็น time out ได้