

● The Overcomplicated Merging Madness

100 milliseconds, 1 megabyte

————๑๙————

หลังจากสร้างของเล่นให้เซเรน่าจนเสร็จ มิเอเนะและลูน่าจึงได้นำของเล่นชิ้นนี้ไปให้กับเซเรน่าในวันเกิดของเธอ ซึ่งนั่นทำให้เซเรน่าดีใจมาก ๆ ทำให้มิเอเนะและลูน่าได้รับคะแนนในงานนี้เยอะมาก ๆ ทั้งจากงานคุ้มกันที่ยอดเยี่ยม งานดูแลเซเรน่าที่ทำได้อย่างทั่วถึงและน่าประทับใจ รวมถึงของเล่นที่น่าพึงพอใจ

หลังจากวันนั้น มิเอเนะและลูน่าออกไปผจญภัย ทำงาน เพื่อเก็บสะสมคะแนนรวมถึงเรียนไปด้วยอยู่เป็นระยะเวลาหนึ่ง จนในขณะนี้มีเอเนะที่อยู่ในฐานนักเรียนปีที่ 4 แล้วนั้น ก็ได้มาทำหน้าที่เป็นผู้ช่วยของศาสตราจารย์ที่เธอนับถือ เพื่อศึกษาค้นคว้า เรียนรู้ รวมถึงช่วยดูแลรุ่นน้องไปด้วย โดยศาสตราจารย์ได้สั่งให้นักเรียนปี 2 ในชั้นของเขาทำการสอบปฏิบัติโดยการสร้าง ต้นไม้เวทมนตร์ โดยการใช้หลักการรายแบบตัวเลข (Numeral Incantation) โดนจะสอบทีละคนตามลำดับภายในคาบเรียน ซึ่งศาสตราจารย์สอนวิชานี้อยู่หลายชั้นมาก ๆ และวิชาดังกล่าวเป็นวิชาเลือกที่ค่อนข้างยากมาก ๆ ทำให้แม้จะมีประโยชน์มาก ๆ แต่ก็มีจำนวนคนลงเรียนที่ค่อนข้างน้อยในแต่ละชั้นเรียน ทำให้มีจำนวนผู้เข้าสอบกระจาย ๆ ไปเป็นกลุ่ม ๆ ย่อยหลายกลุ่ม เนื่องจากศาสตราจารย์มีงานเยอะมาก ๆ ทำให้มิเอเนะในฐานะผู้ช่วยต้องนำคะแนนของนักเรียนแต่ละคนในทุก ๆ ชั้นเรียน มารวมกันเพื่อให้ศาสตราจารย์สามารถตัดเกรดแบบรวมทุกชั้นเรียนได้ ซึ่งตัวคะแนนนั้นศาสตราจารย์ใช้เวทมนตร์เก็บเป็น ต้นไม้ค้นหาแบบทวิภาค หนึ่งชั้นเรียนจะมีต้นไม้ค้นหาแบบทวิภาคหนึ่งต้น และศาสตราจารย์ต้องการให้มิเอเนะนำต้นไม้ค้นหาแบบทวิภาคที่เก็บคะแนนของแต่ละชั้นเรียนมาเปลี่ยนให้เป็นโครงสร้างแบบ Max Heap เพียงอันเดียวเท่านั้น เพื่อให้ศาสตราจารย์สามารถเห็นภาพคะแนนโดยรวมแบบเรียงลำดับคะแนนได้ เนื่องจากศาสตราจารย์ใจดีมาก ๆ ท่านจึงอาจเปลี่ยนการตัดเกรดไปเป็นรูปแบบอื่นแทนหากจะทำให้นักเรียนได้เกรดโดยรวมที่ดีกว่า



ข้อมูลนำเข้า (Input)

บรรทัดแรก

รับจำนวนเต็มบวก n แทนจำนวนต้นไม้ค้นหาแบบทวิภาคจะรับโดยที่ $1 \leq n \leq 10$

บรรทัดที่สองเป็นต้นไป

รับจำนวนเต็มบวก a โดยที่ในกรณีปกติ $1 \leq a \leq 100$ เป็นค่าที่จะใส่ไปในต้นไม้ค้นหาแบบทวิภาค จนกว่า จะรับค่า a มาเป็น -111 ให้ถือว่าเป็นการสิ้นสุดต้นไม้ค้นหาแบบทวิภาคนั้น แล้วให้รับค่าต่อไปเพื่อนำไปใส่ในต้นไม้ค้นหาแบบทวิภาคอันต่อไป จนกว่าจำนวนต้นไม้จะครบตาม n

**คำอธิบายเพื่อความชัดเจนมากยิ่งขึ้น

ข้อมูลส่งออก (Output)

มีบรรทัดเดียว แสดงค่าทั้งหมดใน Max Heap ที่เป็นผลลัพธ์ของการทำงาน ทุกค่าคั่นด้วย , (comma)

ตัวอย่างข้อมูลนำเข้าและข้อมูลส่งออก (Input/Output Examples)

ข้อมูลนำเข้า	ข้อมูลส่งออก
3 9 7 12 20 15 5 2 4 -111 1 6 8 13 19 -111 24 14 28 11 3 -111	28, 24, 20, 19, 13, 6, 15, 12, 5, 7, 8, 4, 1, 9, 14, 2, 11, 3

จำนวนชุดทดสอบ: 7 ชุด

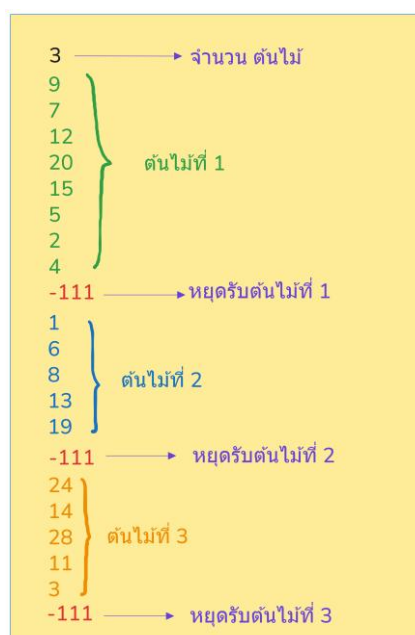
ข้อจำกัด

เนื่องจากคำตอบจริง ๆ แล้วสามารถออกมาได้หลากหลายรูปแบบ เปลี่ยนแปลงไปตามลำดับการทำงาน และการสร้าง Max Heap จึงขอกำหนดขั้นตอนการทำงานดังนี้

- รับค่า input เข้ามาเพื่อสร้าง Binary Search Tree และเก็บ Binary Search Tree ทั้งหมดไว้ใน collection ใดก็ได้ที่เลือก โดยให้คงลำดับการเข้ามาของ Tree เอาไว้
- ทำการนำ Binary Search Tree ที่เก็บไว้มาแปลงเป็น Max Heap โดยใช้หลักการ inorder postorder และใช้หลักการ Breadth-First Search ในการเก็บค่าแต่ละ Tree ลงใน collection ที่เลือกเช่น array list vector เป็นต้น เพื่อให้ Max Heap ที่ได้มาเป็น Complete Binary Tree โดยให้ทำแบบนี้กับทุกต้นไม้และให้คงลำดับเอาไว้เหมือนเดิม
- ทำการนำ collection แต่ละอันที่แปลงมา merge กันทีละคู่ตามลำดับ โดยการ merge ให้นำ collection มารวมกันโดยการต่อกันเลย เช่น [7,6,3,1] กับ [8,4,2] ให้รวมกันให้เป็น [7,6,3,1,8,4,2] จากนั้นสร้าง Max Heap จาก collection นั้น เช่นหากมี tree1,tree2,tree3
 - > maxHeap1,maxHeap2,maxHeap3 (ทำการแปลง tree เป็น max heap)
 - > collection1, collection2, collection3 (bfs เพื่อเก็บค่าลงใน collection ที่เลือก เช่น array, vector)
 - > resultHeap, collection2, collection3 (loop เพื่อ merge กัน loop รอบแรก build แค่ collection แรก)
 - > resultHeap, collection3 (loop รอบต่อ ๆ ไป เอา result ต่อกับ collection ต่อไปแล้ว buildMaxHeap)
 - > resultHeap (loop รอบต่อ ๆ ไป เอา result ต่อกับ collection ต่อไปแล้ว buildMaxHeap)

คำอธิบายตัวอย่างข้อมูลนำเข้า

อธิบายการอ่านข้อมูลนำเข้า



เกณฑ์การให้คะแนนและขอบเขตปัญหาย่อย (Scoring criteria's for subproblems)

การให้คะแนนจะพิจารณาจากเวลาและหน่วยความจำที่โปรแกรมใช้ในการประมวลผล

ระดับ	เงื่อนไข	Runtime และ Memory	ชุดทดสอบ	คะแนน
1	ครึ่งหนึ่งของชุดทดสอบทั้งหมด	100 milliseconds, 1 megabyte	7 ชุด	100%
2	ไม่มีเงื่อนไขเพิ่มเติม	5 milliseconds, 512 kilobytes	7 ชุด	200%