

E-Bidding

การประมูลเป็นการขายสินค้า โดยขายให้แก่ผู้ที่เสนอราคาสูงสุด โปรแกรมที่จะพัฒนานี้ อนุญาตให้เปลี่ยนราคาประมูลได้ (ทั้งเพิ่มหรือลดราคาเสนอ) หรือจะถอนประมูลก็ได้



สิ่งที่ต้องทำ

เขียนโปรแกรมที่รับรายการการเสนอราคาประมูล และการถอนการประมูลของผู้เข้าประมูลต่าง ๆ เพื่อสรุปว่า ใครประมูลได้สินค้าอะไร รวมเป็นเงินทั้งหมดเท่าไร ดังตัวอย่างข้างล่างนี้

Input	คำอธิบาย
8 B b3 p1 11 B b1 p2 10 B b2 p2 9 B b2 p4 1 W b1 p2 W b1 p4 B b1 p1 11 B b3 p4 5	บรรทัดแรกเป็น 8 ระบุว่ามีย่อยรายการที่เกี่ยวกับการประมูลตามมาอีก 8 รายการ 8 บรรทัดต่อมา แต่ละบรรทัดมีอักขระแรกเป็น B หรือ W ถ้าเป็นตัวอื่น ก็ข้ามบรรทัดนั้นไป <ul style="list-style-type: none"> ถ้าเป็น B แทนการเสนอราคาประมูล (Bid) จะตามด้วย รหัสผู้ประมูล รหัสสินค้า และราคาที่เสนอ การเสนอราคาประมูลในบรรทัดใด จะถือว่า เป็นการเสนอราคาประมูลล่าสุด คือ จะแทนราคาที่เคยเสนอไปก่อนหน้านี้ (ถ้ามี) ถ้าเป็น W แทนการถอนการประมูล (Withdraw) จะตามด้วย รหัสผู้ประมูล และรหัสสินค้า (ถ้าไม่พบรายการการประมูลที่มีรหัสผู้ประมูลและรหัสสินค้าที่จะถอน ก็ข้ามไป ไม่ต้องสนใจ) ผู้ชนะประมูลคือ ผู้ที่เสนอราคาสูงสุด ในกรณีที่เสนอราคาเท่ากัน ให้ผู้ที่เสนอก่อนเป็นผู้ชนะ

วิเคราะห์จากตัวอย่างข้างบนนี้ ได้การเปลี่ยนแปลงการประมูลต่าง ๆ แสดงดังตารางข้างล่างนี้

		สินค้าที่ถูกเสนอราคาประมูล		
input		p1	p2	p4
B b3 p1 11		(b3, 11)		
B b1 p2 10		(b3, 11)	(b1, 10)	
B b2 p2 9		(b3, 11)	(b1, 10) , (b2, 9)	
B b2 p4 1		(b3, 11)	(b1, 10) , (b2, 9)	(b2, 1)
W b1 p2		(b3, 11)	(b2, 9)	(b2, 1)
W b1 p4		(b3, 11)	(b2, 9)	(b2, 1)
B b1 p1 11		(b3, 11) , (b1, 11)	(b2, 9)	(b2, 1)
B b3 p4 5		(b3, 11) , (b1, 11)	(b2, 9)	(b2, 1) , (b3, 5)
ผู้ชนะประมูล		(b3, 11) b1 และ b3 เสนอเท่ากัน แต่ b3 เสนอก่อน	(b2, 9) b2 เสนอราคามากสุด	(b3, 5) b3 เสนอราคามากสุด

สรุปว่า **b1, b2 และ b3** ต้องชำระเงินรวมเป็น \$0, \$9 และ \$16 ตามลำดับ

ดังนั้น โปรแกรมจะแสดงผลลัพธ์ของผู้ประมูล รายละเอียดบรรทัด **เรียงตามรหัสผู้ประมูลจากน้อยไปมาก** ในรูปแบบ

รหัสผู้ประมูล: \$จำนวนเงินรวมที่ต้องชำระ -> รายการสินค้าที่ประมูลได้ (เรียงรหัสสินค้าจากน้อยไปมาก)

จากตัวอย่างข้างบนนี้ จะได้ผลที่แสดงคือ

b1: \$0

b2: \$9 -> p2

b3: \$16 -> p1 p4

ข้อมูลนำเข้า

เป็นไปตามลักษณะที่อธิบายไว้ข้างต้น

ข้อมูลส่งออก

เป็นไปตามลักษณะที่อธิบายไว้ข้างต้น

ตัวอย่าง

input (จากแป้นพิมพ์)	output (ทางจอภาพ)	input (จากแป้นพิมพ์)	output (ทางจอภาพ)
3 B b1 p4 4 B b2 p3 3 B b3 p2 2	แบบที่ 1 b1: \$4 -> p4 b2: \$3 -> p3 b3: \$2 -> p2	4 B b1 p4 4 B b2 p3 3 W b1 p4 B b3 p2 2	แบบที่ 5 b1: \$0 b2: \$3 -> p3 b3: \$2 -> p2
3 B b1 p4 4 B b1 p3 3 B b2 p1 1	แบบที่ 2 b1: \$7 -> p3 p4 b2: \$1 -> p1	4 B b1 p4 4 B b1 p3 3 B b2 p1 1 W b2 p1	แบบที่ 6 b1: \$7 -> p3 p4 b2: \$0
3 B b2 p1 1 B b1 p1 2 B b3 p1 3	แบบที่ 3 b1: \$0 b2: \$0 b3: \$3 -> p1	4 B b2 p1 1 B b1 p1 2 B b3 p1 3 W b3 p1	แบบที่ 7 b1: \$2 -> p1 b2: \$0 b3: \$0
3 B b3 p3 1 B b1 p3 2 B b1 p2 3	แบบที่ 4 b1: \$5 -> p2 p3 b3: \$0	5 B b3 p3 1 B b1 p3 2 B b1 p2 3 B b1 p1 1 W b1 p2	แบบที่ 8 b1: \$3 -> p1 p3 b3: \$0
		8 B b1 p1 2 B b2 p1 2 B b3 p1 2 B b3 p2 99 B b2 p2 2 B b1 p2 2 W b3 p2 B b1 p4 4	แบบที่ 9 b1: \$6 -> p1 p4 b2: \$2 -> p2 b3: \$0

ข้อแนะนำ

ข้อกำหนดการประมูลมีหลายเงื่อนไข ข้อมูลขาเข้าที่ใช้ทดสอบจึงมีความง่ายยากหลายแบบ ดังนี้

รูปแบบข้อมูลที่ใช้ทดสอบ การทำงานของโปรแกรม	การถอนประมูล	สินค้าแต่ละชิ้น มีผู้เสนอ ราคาประมูล N คน	ผู้ประมูลแต่ละคนเสนอ ราคาสินค้า M ชิ้น
แบบที่ 1	ไม่มี	$N \leq 1$	$M \leq 1$
แบบที่ 2	ไม่มี	$N \leq 1$	$M \geq 1$
แบบที่ 3	ไม่มี	$N \geq 1$	$M \leq 1$
แบบที่ 4	ไม่มี	$N \geq 1$	$M \geq 1$
แบบที่ 5	มี	$N \leq 1$	$M \leq 1$
แบบที่ 6	มี	$N \leq 1$	$M \geq 1$
แบบที่ 7	มี	$N \geq 1$	$M \leq 1$
แบบที่ 8	มี	$N \geq 1$	$M \geq 1$
แบบที่ 9	แบบที่ 8 + ผู้เสนอราคาสูงสุดมีหลายราย		

เขียนโปรแกรมให้รองรับแบบที่ 9 จึงมีสิทธิ์ได้คะแนนเต็ม แต่อาจยากหน่อย

ถ้าเขียนในแบบอื่น ถึงแม้จะไม่ได้คะแนนเต็ม แต่ได้คะแนนบ้างและง่ายกว่า ลองพิจารณาเองว่า แบบไหนง่าย ก็ทำแบบนั้นก่อน