

## O23C1P1 สะสมเหรียญ

การแข่งขันเกมออนไลน์เกมหนึ่งมีรอบสะสมเหรียญ ผู้จัดการแข่งขันได้ให้ผู้ร่วมแข่งขันแต่ละคนสามารถเข้าไปเล่นมินิเกมที่เราเตรียมไว้จำนวน  $N$  เกม โดยแต่ละคนจะได้รับเวลาที่ใช้ในการสะสมเหรียญ (Time) จากนั้นผู้จัดแข่งขันจะนำเอาเวลา ( $t_i$ ) ที่ผู้แข่งขันทำไว้ในแต่ละมินิเกมที่  $i$  มาหักจากเวลาที่ใช้ในการสะสมเหรียญ และคำนวณหาจำนวนเหรียญที่สะสมได้ การเล่นเกมจะเล่นตามลำดับที่ละเกม แต่ละมินิเกมมีสถานะ (Status) ของเงื่อนไขการได้รับเหรียญดังนี้

- สถานะ Silver ( $S t_i$ ) จะได้ 1 เหรียญและหักเวลา  $t_i$  จากเวลาที่ใช้ในการสะสมเหรียญ
- สถานะ Gold ( $G t_i$ ) จะได้ 1 เหรียญและหักเวลา  $t_i$  จากเวลาที่ใช้ในการสะสมเหรียญ
  - แต่เพิ่มเงื่อนไขพิเศษ คือ หากพบว่าเวลาที่ใช้สะสมเหรียญ (Time) เหลือไม่เกินครึ่งหนึ่งของเวลาที่ใช้ในการสะสมเหรียญเริ่มต้นที่ได้รับ จะได้รับเพิ่มอีก 1 เหรียญ รวมเป็น 2 เหรียญ
- สถานะ Exchange ( $C t_i$ ) จะไม่ได้เหรียญเพิ่ม
  - แต่จะนำเหรียญไปแลกเวลาเพิ่ม โดยใช้เหรียญ 3 เหรียญ เพื่อรับเวลา  $t_i$  เพิ่มให้กับค่า Time ที่เป็นเวลาที่ใช้สะสมเหรียญ

ทั้งนี้หากเวลาที่ใช้ในการสะสมเหรียญหมด ก็จะไม่สามารถเล่นเกมต่อไปได้

ให้เขียนโปรแกรมคำนวณหาจำนวนเหรียญสะสม โดยการคำนวณขึ้นอยู่กับเวลา ( $t_i$ ) ที่ผู้แข่งขันใช้ในการเล่นเกมแต่ละรายการ และสถานะของมินิเกม (Status) โดยการคำนวณจะต้องเป็นไปตามลำดับเกมที่คุณจัดแข่งขันกำหนด

### รูปแบบข้อมูลนำเข้า

บรรทัดแรก เป็น เลขจำนวนเต็ม 2 จำนวน Time และ  $N$  คั่นด้วยช่องว่าง โดย

- Time แทน จำนวนหน่วยเวลาที่ใช้สะสมเหรียญ ซึ่งผู้จัดการแข่งขันกำหนดค่าเริ่มต้นให้ โดยที่  $1 \leq \text{Time} \leq 10000$
- $N$  แทนจำนวนเกมที่เข้าร่วม โดยที่  $1 \leq N \leq 10000$

อีก  $N$  บรรทัด แต่ละบรรทัดเป็นข้อมูลมินิเกมแต่ละเกม ประกอบด้วยข้อมูล 2 ค่า คั่นด้วยช่องว่างได้แก่

- สถานะ (Status) มี 3 ค่า คืออักขระ S G และ C แทนเงื่อนไขการได้รับเหรียญตามที่ระบุข้างต้น
- จำนวนเต็ม  $t_i$  โดยที่  $1 \leq t_i \leq 100$  แทนเวลาที่ผู้แข่งขันได้ใช้ในการเล่นเกม โดยที่  $1 \leq i \leq N$  และเป็นลำดับการคำนวณเหรียญ

### รูปแบบข้อมูลส่งออก

เลขจำนวนเต็ม 2 ค่า จำนวน 1 บรรทัด ประกอบด้วย จำนวนเหรียญ และจำนวนเกมที่ใช้สะสมเหรียญ

ตัวอย่างการคำนวณ (จากข้อมูลตัวอย่าง ที่ 2)

Input	ความหมาย	การทำงาน/การคำนวณ	Output
10 8	เวลาดังต้น	รับค่าเวลาดังต้น (Time) = 10	4 6
	จำนวนรายการเกม	รับค่าจำนวนรายการเกม (N) = 8	
S 1	ประเภทเกม และเวลาที่ใช้รายการที่ 1	เกมสถานะ S และผู้แข่งขันใช้เวลาสำหรับเกมนี้ไป 1 หน่วยเวลา ดังนั้นเวลาที่เหลือสำหรับสะสมเกมถัดไปเท่ากับ $\text{Time} = 10 - 1 = 9$ หน่วย และสะสมได้ 1 เหรียญ	
S 2	ประเภทเกม และเวลาที่ใช้รายการที่ 2	เกมสถานะ S และใช้เวลาสำหรับเกมนี้ 2 หน่วยเวลา ดังนั้น $\text{Time} = 9 - 2 = 7$ หน่วย โดยสะสมเพิ่มได้อีก 1 เหรียญ (รวม 2 เหรียญ)	
S 1	ประเภทเกม และเวลาที่ใช้รายการที่ 3	เกมสถานะ S และใช้เวลาสำหรับเกมนี้ 1 หน่วยเวลา ดังนั้น $\text{Time} = 7 - 1 = 6$ หน่วย โดยสะสมเพิ่มได้อีก 1 เหรียญ (รวม 3 เหรียญ)	
G 3	ประเภทเกม และเวลาที่ใช้รายการที่ 4	เกมสถานะ G และใช้เวลาสำหรับเกมนี้ 3 หน่วยเวลา ดังนั้น $\text{Time} = 6 - 3 = 3$ หน่วย เวลาไม่ถึงครึ่งหนึ่งของเวลาดังต้น (10) จึงสะสมเพิ่มได้อีก 2 เหรียญ (รวม 5 เหรียญ)	
C 5	ประเภทเกม และเวลาที่ใช้รายการที่ 5	สถานะ C เป็นการใช้เหรียญขอแลกเวลาเพิ่ม ใช้เหรียญ 3 เหรียญ ได้เวลาเพิ่ม 5 หน่วย ดังนั้น $\text{Time} = 3 + 5 = 8$ หน่วย	
S 2	ประเภทเกม และเวลาที่ใช้รายการที่ 6	เกมสถานะ S และใช้เวลาสำหรับเกมนี้ 2 หน่วยเวลา ดังนั้น $\text{Time} = 8 - 2 = 6$ หน่วย โดยสะสมเพิ่มได้อีก 1 เหรียญ (รวม 3 เหรียญ )	
S 4	ประเภทเกม และเวลาที่ใช้รายการที่ 7	เกมสถานะ S และใช้เวลาสำหรับเกมนี้ 4 หน่วยเวลา ดังนั้น $\text{Time} = 6 - 4 = 2$ หน่วย โดยสะสมเพิ่มได้อีก 1 เหรียญ (รวม 4 เหรียญ)	
S 3	ประเภทเกม และเวลาที่ใช้รายการที่ 8	เกมสถานะ S และใช้เวลาสำหรับเกมนี้ 3 หน่วยเวลา ใช้เวลาจน $\text{Time}$ น้อยกว่า 0 เสมือนเล่นเกมไม่จบ จึงไม่สามารถสะสมเหรียญในรายการนี้ได้ (รวม 4 เหรียญ)	
		<p><u>ความหมายของผลลัพธ์ (Output)</u></p> <p>4 6</p> <p>สะสมเหรียญได้ 4 เหรียญ</p> <p>โดยสะสมได้ทั้งหมด 6 เกม</p>	

ข้อสอบครั้งที่ 1 ข้อสอบปฏิบัติโปรแกรม  
วันที่ 14 ต.ค. 66 เวลา 9:00-12:00 น.

โครงการโอลิมปิกวิชาการ ค่าย 1 สาขาวิชาคอมพิวเตอร์ ปี 2566  
ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยเชียงใหม่

ตัวอย่างข้อมูลนำเข้า (Input) และส่งออก (Output)

ตัวอย่างที่ 1		ตัวอย่างที่ 2		ตัวอย่างที่ 3		ตัวอย่างที่ 4	
Input	Output	Input	Output	Input	Output	Input	Output
15 6	8 6	10 8	4 6	20 8	0 5	20 8	1 4
S 1		S 1		S 3		S 4	
G 3		S 2		S 4		S 2	
G 4		S 1		G 2		S 8	
G 2		G 3		C 5		C 3	
S 2		C 5		S 6		S 6	
G 2		S 2		G 8		C 2	
		S 4		C 2		S 5	
		S 3		S 5		S 5	