

TECHJam

DEEP

CODE

TechJam 2019 Deep Code · First Round Programming Task

โปรดอ่าน

เอกสารฉบับนี้รวมถึงข้อความและรูปภาพที่ปรากฏในเอกสารฉบับนี้ เป็นทรัพย์สินทางปัญญาของกสิกร มิชินส์-เทคโนโลยี กรุ๊ป (KBTG) ตามพระราชบัญญัติลิขสิทธิ์ พุทธศักราช 2537 ไม่อนุญาตให้ผู้ใดนำส่วนใดส่วนหนึ่งหรือทั้งหมดของเอกสารฉบับนี้ไปคัดลอก ทำซ้ำ ดัดแปลง เผยแพร่ หรือส่งต่อเอกสารฉบับนี้แก่ผู้อื่นโดยเด็ดขาด โปรดเก็บรักษาข้อมูลในเอกสารนี้เป็นความลับ

Emergency Response Stress Test

Problem Statement

องค์การบริการกู้ภัยฉุกเฉิน (Emergency Response Service Authority; ERSA) กำลังจะจัดงานซ้อมทดสอบกู้ภัยประจำปี ซึ่งจัดขึ้น ณ ถนนทางหลวงทางตรงสายหนึ่ง เพื่อทดสอบขีดความสามารถในการตอบสนองต่อภัยที่เกิดขึ้นที่อาจเกิดขึ้นอย่างฉับพลันและพร้อมกันในหลาย ๆ จุด สมาชิกหน่วยกู้ภัยฉุกเฉินแต่ละคนจากทั้งหมด n คน จะประจำการอยู่ที่จุดต่าง ๆ บนทางหลวงสายนี้ ทางหลวงสายนี้เปรียบเสมือนเส้นจำนวนในแกน X และจุดต่าง ๆ บนทางหลวงสามารถแทนได้ด้วยพิกัดบนแกน X นอกจากนี้ระยะทาง 1 เมตรบนทางหลวงสายนี้มีค่าเท่ากับระยะ 1 หน่วยบนแกน X

การทดสอบจะเริ่มขึ้นเมื่อมีเสียงนกหวีด เมื่อเสียงนกหวีดดังขึ้น นาฬิกาจับเวลา m เรือนจะปรากฏขึ้นตามสถานที่ตำแหน่งต่าง ๆ บนทางหลวง นอกจากนั้นนาฬิกาแต่ละเรือนจะเริ่มนับเวลาทอยหลังจากค่าเริ่มต้นที่แตกต่างกัน เป้าหมายของทีมหน่วยกู้ภัยโดยรวมคือจะต้องส่งสมาชิกของทีมตัวเองไปเคลมนาฬิกาจับเวลาให้ได้จำนวนมากที่สุดก่อนนาฬิกาเรือนนั้น ๆ หมดเวลาลงเสียก่อน ตลอดเวลาทำการทดสอบนี้ สมาชิกทีมกู้ภัยแต่ละคนสามารถเคลมนาฬิกาได้อย่างมากที่สุด 1 เรือน และนาฬิกาเรือนดังกล่าวจะถูกเคลมได้หากสมาชิกคนนั้นวิ่งไปถึงนาฬิกาเรือนดังกล่าวก่อนหมดเวลาหรือทันเวลาฉิวเฉียดพอดีเท่านั้น

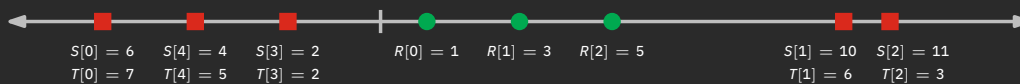
หมายเหตุ กฎความปลอดภัยของทีมหน่วยกู้ภัยคือ สมาชิกแต่ละคนสามารถเคลื่อนไหวดำวยอัตราเร็วสูงสุด 1 เมตรต่อวินาทีเท่านั้น

Example #1

ลองพิจารณาตัวอย่างสถานการณ์ดังนี้ สมมติว่า ทีมหน่วยกู้ภัยมีสมาชิกทั้งสิ้น $n = 3$ คน แต่ละคนประจำการอยู่ที่ตำแหน่ง $x = 1$, $x = 3$ และ $x = 5$ ตามลำดับ

เมื่อเสียงนกหวีดดังขึ้น นาฬิกาจับเวลา $m = 5$ เรือนปรากฏขึ้นดังต่อไปนี้

- นาฬิกาเรือนที่ 1 ปรากฏที่ตำแหน่ง $x = -6$ และเริ่มต้นนับเวลากอยหลังที่ 7 วินาที
- นาฬิกาเรือนที่ 2 ปรากฏที่ตำแหน่ง $x = 10$ และเริ่มต้นนับเวลากอยหลังที่ 6 วินาที
- นาฬิกาเรือนที่ 3 ปรากฏที่ตำแหน่ง $x = 11$ และเริ่มต้นนับเวลากอยหลังที่ 3 วินาที
- นาฬิกาเรือนที่ 4 ปรากฏที่ตำแหน่ง $x = -2$ และเริ่มต้นนับเวลากอยหลังที่ 2 วินาที
- นาฬิกาเรือนที่ 5 ปรากฏที่ตำแหน่ง $x = -4$ และเริ่มต้นนับเวลากอยหลังที่ 5 วินาที



ในกรณีนี้ ทีมหน่วยกู้ภัยจะสามารถเคลมนาฬิกาจับเวลาได้มากที่สุดเพียง 2 เรือนเท่านั้น นั่นคือ

- สมาชิกที่ประจำอยู่ที่ตำแหน่ง $x = 1$ เคลมนาฬิกาเรือนที่ 1 อย่างทันฉิวเฉียด
- สมาชิกที่ประจำอยู่ที่ตำแหน่ง $x = 5$ เคลมนาฬิกาเรือนที่ 2 ก่อนหมดเวลา 1 วินาทีพอดี

Objectives

จงเขียนโปรแกรมเพื่อรับ Input Data ต่อไปนี้

- ข้อมูลเกี่ยวกับสมาชิกทีมหน่วยกู้ภัยทั้งสิ้น n ราย (โดยที่ $1 \leq n \leq 200,000$) กล่าวคือสมาชิกกู้ภัยคนที่ i สำหรับ $i = 0, 1, \dots, n-1$ จะมีข้อมูลดังต่อไปนี้
 - $R[i]$ คือตำแหน่งเริ่มต้นบนแกน X ของสมาชิกคนที่ i (โดยที่ $-1,000,000,000 \leq R[i] \leq 1,000,000,000$)
- ข้อมูลของนาฬิกาทั้งสิ้น m เรือนที่ปรากฏขึ้นเมื่อเริ่มเสียงสัญญาณนกหวีด (โดยที่ $1 \leq m \leq 200,000$) กล่าวคือนาฬิกาเรือนที่ j สำหรับ $j = 0, 1, \dots, m-1$ จะมีข้อมูลดังต่อไปนี้
 - $S[j]$ คือตำแหน่งที่นาฬิกาเรือนที่ j ปรากฏบนแกน X (โดยที่ $-1,000,000,000 \leq S[j] \leq 1,000,000,000$)
 - $T[j]$ คือเวลาเริ่มต้นของนาฬิกาเรือนที่ j ในหน่วยวินาที (โดยที่ $0 \leq T[j] \leq 1,000,000,000$)

แล้วจึงคำนวณจำนวนนาฬิกาที่สามารถเคลมได้มากที่สุดเท่าที่เป็นไปได้ และคืนค่าคำตอบดังกล่าวเป็น Output Data ของโปรแกรม

Interfaces and Data Format

โปรแกรมที่เขียนขึ้นจะต้องรับ Input Data ผ่าน Standard Input ซึ่งมีรูปแบบดังต่อไปนี้

- บรรทัดแรกมีจำนวนเต็มสองจำนวน n และ m
- บรรทัดที่ $i+2$ สำหรับ $i = 0, 1, \dots, n-1$ จะมีจำนวนเต็มหนึ่งจำนวน ซึ่งก็คือ $R[i]$
- บรรทัดที่ $n+j+2$ สำหรับ $j = 0, 1, \dots, m-1$ จะมีจำนวนเต็มสองจำนวนที่ถูกคั่นด้วยช่องว่าง ซึ่งก็คือ $S[j]$ และ $T[j]$

```
1  n m
2  R[0]
3  R[1]
...
n+1 R[n-1]
n+2 S[0] T[0]
n+3 S[1] T[1]
...
n+m+1 S[m-1] T[m-1]
```

โปรแกรมที่เขียนขึ้นจะต้องคืน Output Data ผ่าน Standard Output เป็นจำนวนเต็ม 1 จำนวน ซึ่งเป็นคำตอบของโจทย์ตามที่ระบุไว้ในหัวข้อ Objectives ข้างต้น

Example Input	Example Output
3 5 1 3 5 -6 7 10 6 11 3 -2 2 -4 5	2

Scoring

โปรแกรมของคุณจะถูกทดสอบกับ Test Cases ที่มีเงื่อนไขต่าง ๆ ดังนี้

SMALL (คะแนน 20%)

รับประกันว่าจำนวนของสมาชิกกู้ภัยและจำนวนของนาฬิกาจะสอดคล้องกับเงื่อนไข $1 \leq n, m \leq 1,000$ และค่าพิทัดแกน X จะอยู่ในช่วงตั้งแต่ $-100,000$ จนถึง $100,000$

MEDIUM (คะแนน 35%)

รับประกันว่าค่าพิทัดแกน X จะอยู่ในช่วงตั้งแต่ $-100,000$ จนถึง $100,000$

LARGE (คะแนน 45%)

ไม่มีเงื่อนไขเพิ่มเติม

Limitations

โปรแกรมจะถูกจำกัดเวลาอยู่ที่ 1.0 วินาทีต่อ Test Case (baseline) และถูกจำกัดหน่วยความจำอยู่ที่ 512 MB

- สำหรับโปรแกรมที่เขียนด้วยภาษา C หรือ C++ จะถูกจำกัดเวลาเท่ากับค่า baseline ข้างต้น
- สำหรับโปรแกรมที่เขียนด้วยภาษา Go หรือ Java จะถูกจำกัดเวลาอยู่ที่ 1.5 เท่าของ baseline ข้างต้น
- สำหรับโปรแกรมที่เขียนด้วยภาษา JavaScript หรือ Python จะถูกจำกัดเวลาอยู่ที่ 2.5 เท่าของ baseline ข้างต้น