Poison

เวลา: 2 วินาที | หน่วยความจำ: 512 MB

ณ โรงเรียนกำเนิดคณิตวิทยาอันเลื่องชื่อ ซึ่งตั้งตระหง่านอยู่ท่ามกลางหุบเขาและสายหมอก ภายใน "วังอาคารวัลเลย์" อันศักดิ์สิทธิ์ มีตำนานของบุคคลผู้หนึ่งที่นักเรียนทุกคนต่างกล่าวขานถึง

เขาคือมหาเทพผู้ยิ่งใหญ่แห่งการนอนดึก การส่งงานก่อนเดดไลน์เพียง 3 วินาที และผู้สามารถเขียนโค้ดผ่านทุก test case ได้แม้คอมไพล์ไม่ผ่านในครั้งแรก...

นามของเขาคือ **GUGUGGUGUG**

ในคืนหนึ่ง ขณะที่ลมหนาวพัดผ่านระเบียงหอพัก และเสียง mechanical keyboard ดังก้องไปทั่วชั้น มหาเทพ GUGUGGUGUG ได้เกิดความคิดอันทะเยอทะยานขึ้นมา...

> "ถ้ามีเซิร์ฟเวอร์แรง ๆ... เราก็เปิดเซิร์ฟมายคราฟต์เล่นกันทั้งหอได้สิ"

แต่ปัญหาเดียวคือ...

เขาไม่มีเซิร์ฟเวอร์

ด้วยเหตุนี้ มหาเทพจึงเดินทางไปยังองค์กรลึกลับที่ไม่มีใครรู้ว่าทำอะไรจริง ๆ นอกจากก็อปงานรุ่นพี่ นั่นคือ

Institute of Copy-paste Technology (ICT)

สถาบันเทคโนโลยี Ctrl+C Ctrl+V อันโด่งดัง

ว่ากันว่า ICT มีห้องเซิร์ฟเวอร์ที่แรงพอจะรันทั้ง AI, Blockchain, Quantum Simulation และ Minecraft Modpack 500 mods พร้อมกันได้โดยไม่แลคแม้แต่นิดเดียว

มหาเทพ GUGUGGUGUG จึงแอบ "ยืม" เซิร์ฟเวอร์เครื่องหนึ่งออกมา โดยให้เหตุผลว่า

> "เอาไปทำ Distributed Computing Project ครับ"

แต่ความจริงนั้น...

เขาเอาไปเปิดเซิร์ฟมายคราฟต์ Hardcore พร้อมปลั๊กอินเศรษฐกิจ ระบบกาชา และสงครามแฟคชัน

ในช่วงแรก ทุกอย่างดูเหมือนจะเป็นไปด้วยดี นักเรียนในวังอาคารวัลเลย์ต่างเข้ามาเล่นกันอย่างสนุกสนาน

- บางคนสร้างบ้าน
- บางคนสร้างฟาร์มเหล็ก
- บางคนทำฟาร์มวิลเลเจอร์
- และบางคนใช้เวลา 14 ชั่วโมงเพื่อสร้างโลโก้โรงเรียนกลางแผนที่

แต่แล้ววันหนึ่ง...

ICT ตรวจพบว่า traffic ของเซิร์ฟเวอร์มี packet แปลก ๆ เต็มไปหมด ทั้งข้อมูล chunk generation, creeper explosion และข้อความ

> "tp หน่อย"

กว่าหมื่นครั้งต่อชั่วโมง

พวกเขาจึงค้นพบความจริงอันน่าตกใจว่า...

เซิร์ฟเวอร์วิจัยระดับองค์กร ถูกนำไปเปิดเซิร์ฟมายคราฟต์

สมาชิกระดับสูงของ ICT ต่างโกรธเป็นอย่างมาก บางคนถึงกับพูดว่า

> "นี่มัน misuse of computational resources ชัด ๆ!"

แต่ทว่า...

วันถัดไปคือวัน **Dorm Party**

วันแห่งมิตรภาพ อาหารบุฟเฟต์ และการเปิดเพลงเสียงดังจน RA ต้องเดินขึ้นมาดูทุก 15 นาที

ICT จึงไม่อยากทำเรื่องให้ใหญ่โต พวกเขาจึงเลือกใช้

"บทลงโทษเชิงวิชาการ"

แทน

ภายในงานเลี้ยง มีแก้วน้ำจำนวนทั้งหมด (N) แก้ว วางเรียงอยู่บนโต๊ะยาวกลางห้องโถง และ ICT ได้แอบใส่ยาพิษลงไปในแก้วเพียง **1 แก้ว**

อย่างไรก็ตาม พวกเขาก็ยังมีเมตตา...

ICT อนุญาตให้มหาเทพ GUGUGGUGUG สามารถถามได้ทั้งหมด **10 ครั้ง**

โดยในแต่ละครั้ง เขาสามารถเลือก "กลุ่มของแก้วน้ำ" ขึ้นมาหนึ่งกลุ่ม แล้วถามว่า

> "ในกลุ่มนี้ มีแก้วที่มียาพิษอยู่หรือไม่?"

ICT จะตอบเพียงแค่

- "มี"
- หรือ "ไม่มี"

เท่านั้น

หน้าที่ของคุณ คือช่วยมหาเทพ GUGUGGUGUG ใช้คำถามทั้ง 10 ครั้งอย่างชาญฉลาด เพื่อค้นหาแก้วน้ำที่มียาพิษให้ได้ ก่อนที่ Dorm Party จะเริ่มขึ้น...

และก่อนที่ใครบางคนจะเผลอดื่มมันเข้าไปจริง ๆ

เพราะหากหาไม่ทัน...

> เซิร์ฟมายคราฟต์อาจถูกปิดตลอดกาล

รายละเอียดการเขียนโปรแกรม

คุณต้องส่งโปรแกรมของฟังก์ชันตามเงื่อนไขต่อไปนี้

ฟังก์ชันที่ต้องเขียน

```
int glass(int N)
```

โดยที่

- (N) คือจำนวนแก้วน้ำทั้งหมด

ฟังก์ชันต้องคืนค่าเป็นหมายเลขของแก้วที่มียาพิษ

การถาม ICT

ภายในฟังก์ชัน `glass` คุณสามารถถาม ICT ผ่านฟังก์ชัน

```
void find_glass(vector a)
```

โดยที่

- a คือกลุ่มของแก้วน้ำที่ต้องการถาม

ICT จะเก็บคำถามนี้ไว้ และจะตอบทั้งหมดเมื่อเรียก `done_query()`

เงื่อนไขของการส่งกลุ่ม

สำหรับแต่ละคำถาม

- ห้ามมีเลขซ้ำภายใน vector เดียวกัน
- ทุกค่าต้องอยู่ในช่วง `0` ถึง `N-1`
- สามารถส่งแก้วน้ำ 1 แก้วลงไปหลายกลุ่มได้
- ห้ามเรียก `find_glass()` หลังจากเรียก `done_query()`
- ไม่สามารถส่ง vector เปล่าได้

จำนวนครั้งในการถาม

คุณสามารถเรียก `find_glass()` ได้ไม่เกิน $min(10, N)$ **ครั้ง**

หากเรียกเกิน จะได้รับคะแนนตามสัดส่วนที่กำหนด

การรับคำตอบจาก ICT

เมื่อส่งคำถามครบแล้ว สามารถเรียก

```
vector done_query()
```

โดย

- ค่าที่คืนกลับมาจะเรียงตามลำดับของคำถามที่ส่งไป
- ค่า `true` หมายถึง ในกลุ่มนั้นมีแก้วพิษ
- ค่า `false` หมายถึง ในกลุ่มนั้นไม่มีแก้วพิษ

ตัวอย่าง

สมมุติให้

```
N = 3
```

และแก้วพิษคือแก้วหมายเลข 2

grader เรียก

```
glass(3)
```

คุณส่งคำถามดังนี้

ครั้งที่ 1

```
find_glass({0,1})
```

ครั้งที่ 2

```
find_glass({0,2})
```

ครั้งที่ 3

```
find_glass({1,2})
```

หลังจากนั้นคุณเรียก

```
done_query()
```

และได้รับค่า

```
[false, true, true]
```

ดังนั้นคุณจึงสรุปได้ว่าแก้วพิษคือแก้วหมายเลข

```
2
```

และคืนค่าออกจากฟังก์ชัน

```
return 2;
```

ข้อกำหนด

- $2 \leq N \leq 1000$
- $0 \leq M \leq N - 1$

ปัญหาย่อย

ปัญหาย่อยที่ 1 (10 คะแนน)

- $N \leq 10$

ปัญหาย่อยที่ 2 (90 คะแนน)

- ไม่มีเงื่อนไขเพิ่มเติม
- หากเรียก `find_glass()` เกิน 10 ครั้ง จะได้รับคะแนนตามสัดส่วน $\frac{10}{q} \times 90$

โดยคะแนนจะไม่เกิน 90 คะแนน