

หนักหรือเบา (heavy or light)

1 second, 256 MB

มีเหรียญ N เหรียญ ($3 \leq N \leq 300$) ทุกเหรียญมีน้ำหนักเท่ากันหมด ยกเว้นหนึ่งเหรียญเป็นเหรียญปลอม ซึ่งอาจจะมีน้ำหนักมากกว่าหรือน้อยกว่าเหรียญอื่นก็ได้

มีตาชั่งอยู่หนึ่งอัน สามารถใช้ชั่งน้ำหนักเหรียญหรือเซตของเหรียญได้ ให้คุณหาว่าเหรียญปลอมที่เพิ่มเข้ามานั้นมีน้ำหนักน้อยกว่า หรือมากกว่าเหรียญอื่น ๆ

ถ้าคุณใช้จำนวนครั้งในการชั่งน้อยที่สุดเท่าที่เป็นไปได้ในทุกชุดข้อมูลทดสอบ คุณจะได้คะแนนเต็ม ถ้ามีบางชุดข้อมูลทดสอบคุณใช้มากกว่านั้นคุณจะได้คะแนนเท่ากับ

$$20 + 80 * (\text{จำนวนชั่งน้อยสุด/จำนวนชั่งของคุณ})^2$$

โดยคะแนนที่คุณได้จะพิจารณาจากชุดข้อมูลทดสอบที่คุณได้คะแนนน้อยที่สุด

การใช้ Library

ให้ include "hollib.h" มีฟังก์ชันให้เรียกดังนี้

- `int hol_init_all()` จะคืนจำนวนข้อมูลชุดทดสอบในการเรียกทำงานครั้งนี้ สำหรับการ ทำงานหนึ่งครั้ง จำนวนชุดข้อมูลทดสอบจะไม่เกิน 100 ชุด ฟังก์ชันนี้จะเรียกครั้งเดียวเมื่อเริ่ม ต้นการทำงาน

ในแต่ละข้อมูลชุดทดสอบ คุณจะใช้ฟังก์ชันต่อไปนี้

- `int hol_init()` จะคืนจำนวนเต็ม N ($3 \leq N \leq 300$)
- `int hol_weight(int lcount, int llist[], int rcount, int rlist[])` จะ เป็นการสั่งให้ชั่งน้ำหนัก โดยที่ `lcount` และ `rcount` แทนจำนวนเหรียญที่อยู่ตาชั่งด้านซ้ายและ ขวา และ `llist` กับ `rlist` เป็นอาร์เรย์เก็บหมายเลขของเหรียญที่ต้องการใส่ในตาชั่งซ้ายและขวา หมายเลขของเหรียญมีค่า 0 ถึง $N-1$ ค่าที่คืนของฟังก์ชันนี้จะเป็น -1 (ด้านซ้ายหนักกว่า), 0 (เท่ากัน), และ 1 (ขวานหนักกว่า)
- `int hol_answer(int result)` เรียกเมื่อจบการทำงานในทุกข้อมูลทดสอบ ให้ `result = 1` ถ้าเหรียญแปลกปลอมหนักกว่าปกติ และ `result = 2` ถ้าเหรียญแปลกปลอมเบากว่าปกติ

การทดสอบโปรแกรมของคุณ

สามารถดาวน์โหลดไลบรารีสำหรับทดสอบได้ที่

<https://theory.cpe.ku.ac.th/~jittat/loi/2016/hfiles/fgti3hd9-heavyorlight/>

ไลบรารีจะอ่านอินพุตจาก standard input ในรูปแบบดังนี้

- บรรทัดแรกระบุจำนวนข้อมูลทดสอบ T อีก T บรรทัดถัดไประบุข้อมูลทดสอบแต่ละชุด
- แต่ละบรรทัดของข้อมูลทดสอบจะมี string หนึ่งตัว ค่า N จะถูกคำนวณจากความยาวของสตริง นี้ สตริงดังกล่าวระบุน้ำหนักของเหรียญแต่ละเหรียญโดยที่ตัวอักษร N แทนเหรียญน้ำหนักปกติ H แทนเหรียญหนักกว่าปกติ และ L แทนเหรียญเบากว่าปกติ

เมื่อทำงานเสร็จ หลักเรียก `hol_answer` โปรแกรมจะพิมพ์ว่า Correct หรือ Incorrect และมีวงเล็บ จำนวนครั้งที่ชั่งระบุด้วย

(ตัวอย่างอยู่หน้าถัดไป)

ตัวอย่างการทำงานของไลบรารีตัวอย่าง

สมมติว่าข้อมูลนำเข้าที่ standard library เป็นดังนี้

2

NNHN

LNNNNN

คำสั่งที่	คำสั่ง	คำตอบ
1	hol_init_all()	2
2	hol_init()	4
3	hol_weight(1,{0},2,{2,3})	1
4	hol_weight(1,{2},1,{1})	-1
5	hol_answer(1)	ไม่มีการคืนค่า หน้าจอจะพิมพ์ Correct (2)
6	hol_init()	6
7	hol_weight(3,{0,1,2},3,{3,4,5})	1
8	hol_answer(2)	ไม่มีการคืนค่า หน้าจอจะพิมพ์ Correct (1)

การให้คะแนน

มีข้อมูลหลายชุดทดสอบ ในแต่ละชุดทดสอบก็มีหลายข้อมูลทดสอบ ถ้าในทุกชุดทดสอบตอบถูกหมดจะได้คะแนนดังนี้

- 100% ถ้าในทุก ๆ การทำงาน คุณสั่ง hol_weight เป็นจำนวนครั้งเท่ากับจำนวนคำถามที่น้อยที่สุด
- พิจารณาคะแนนเท่ากับ $20 + 80 * (\text{จำนวนชั่งน้อยสุด} / \text{จำนวนชั่งของคุณ})^2$ โดยคิดในกรณีทดสอบที่คุณได้คะแนนน้อยที่สุด

ถ้ามีการตอบผิด จะได้ 0 คะแนน