



International Olympiad in Informatics 2013

6-13 July 2013
Brisbane, Australia
Day 2 tasks

robots

Thai — 1.0

น้องชายของมาริต้ากำลังของเล่นกระจายทั่วทั้งห้องนั่งเล่น! โชคดีที่มาริต้าได้พัฒนาหุ่นยนต์พิเศษที่ใช้เก็บของเล่นเอาไว้ เธอต้องการความช่วยเหลือจากคุณในการเลือกว่าจะให้หุ่นยนต์ตัวใดเก็บของเล่นชิ้นไหน

มีของเล่น T ชิ้น แต่ละชิ้นระบุน้ำหนัก $W[i]$ เป็นจำนวนเต็ม และระบุขนาด $S[i]$ เป็นจำนวนเต็มเช่นกัน หุ่นยนต์มีสองแบบ: หุ่นยนต์อ่อนแอ และ หุ่นยนต์เล็ก

- มีหุ่นยนต์อ่อนแอ A ตัว หุ่นยนต์อ่อนแอแต่ละตัวจะมีขีดจำกัดด้านน้ำหนักเป็น $X[i]$ และสามารถยกของเล่นใด ๆ ที่มีน้ำหนักน้อยกว่า $X[i]$ ได้ หุ่นยนต์อ่อนแอสามารถยกของเล่นขนาดใดก็ได้
- มีหุ่นยนต์เล็ก B ตัว หุ่นยนต์เล็กแต่ละตัวจะมีขีดจำกัดด้านขนาดเป็น $Y[i]$ และสามารถยกของเล่นใด ๆ ที่มีขนาดเล็กกว่า $Y[i]$ ได้ หุ่นยนต์เล็กสามารถยกของเล่นน้ำหนักเท่าใดก็ได้

หุ่นยนต์แต่ละตัวของมาริต้าใช้เวลาหนึ่งนาทีในการเก็บของเล่นแต่ละชิ้น หุ่นแต่ละตัวจะยกของเล่นได้ที่ละหนึ่งชิ้น แต่หุ่นยนต์หลายตัวสามารถทำงานพร้อม ๆ กันได้ กล่าวคือ ในเวลาหนึ่ง ๆ หุ่นหลายตัวสามารถเก็บของเล่นหลายชิ้นที่แตกต่างกันได้

งานของคุณคือตัดสินใจว่าหุ่นยนต์ของมาริต้าสามารถเก็บของเล่นทั้งหมดได้หรือไม่ ถ้าได้ให้คำนวณหาเวลาที่สั้นที่สุดที่สามารถทำได้

ตัวอย่าง

สำหรับตัวอย่างแรก สมมติว่ามีหุ่นยนต์อ่อนแอ $A=3$ ตัว ที่มีขีดจำกัดด้านน้ำหนัก $X=[6,2,9]$ และหุ่นยนต์เล็กจำนวน $B=2$ ตัว ที่มีขีดจำกัดด้านขนาด $Y=[4,7]$ และมีของเล่น $T=10$ ชิ้น ดังระบุด้านล่าง

หมายเลขของเล่น	0	1	2	3	4	5	6	7	8	9
น้ำหนัก	4	8	2	7	1	5	3	8	7	10
ขนาด	6	5	3	9	8	1	3	7	6	5

เวลาที่น้อยที่สุดที่สามารถเก็บของเล่นได้คือสามนาที โดยมีขั้นตอนในการเก็บดังนี้

	หุ่นอ่อนแอ 0	หุ่นอ่อนแอ 1	หุ่นอ่อนแอ 2	หุ่นเล็ก 0	หุ่นเล็ก 1
นาทีที่ 1	ของเล่น 0	ของเล่น 4	ของเล่น 1	ของเล่น 6	ของเล่น 2
นาทีที่ 2	ของเล่น 5		ของเล่น 3		ของเล่น 8
นาทีที่ 3			ของเล่น 7		ของเล่น 9

สำหรับตัวอย่างที่สอง สมมติว่ามีหุ่นยนต์อ่อนแอ $A=2$ ตัว ที่มีขีดจำกัดด้านน้ำหนัก $X=[2,5]$ และหุ่นยนต์เล็กจำนวน $B=1$ ตัว ที่มีขีดจำกัดด้านขนาด $Y=[2]$ และมีของเล่น $T=3$ ชิ้น ดังระบุด้านล่าง

หมายเลขของเล่น	0	1	2
น้ำหนัก	3	5	2
ขนาด	1	3	2

เนื่องจากไม่มีหุ่นยนต์ตัวใดที่สามารถยกของเล่นน้ำหนัก 5 และขนาด 3 ได้ ดังนั้นจึงเป็นไปได้ที่บรรดาหุ่นยนต์จะเก็บของเล่นทั้งหมดได้

การเขียนโปรแกรม

คุณจะต้องส่งแฟ้มโปรแกรมที่เขียนฟังก์ชัน `putaway()` ดังที่ระบุด้านล่าง

ฟังก์ชัน `putaway()` ของคุณ

C/C++

```
int putaway(int A, int B, int T,
            int X[], int Y[], int W[], int S[]);
```

Pascal

```
function putaway(A, B, T : LongInt;
                 var X, Y, W, S : array of LongInt) : LongInt;
```

คำอธิบาย

ฟังก์ชันนี้จะคำนวณเวลาที่น้อยที่สุด (เป็นนาที) ที่หุ่นยนต์จะต้องใช้ในการเก็บของเล่นทั้งหมด หรือคืนค่า `-1` ถ้าไม่สามารถทำได้

พารามิเตอร์

- A : จำนวนหุ่นยนต์อ่อนแอ
- B : จำนวนหุ่นยนต์เล็ก
- T : จำนวนของเล่น
- X : อาร์เรย์ขนาด A ที่เก็บลำดับของจำนวนเต็มระบุขีดจำกัดด้านน้ำหนักสำหรับหุ่นยนต์อ่อนแอแต่ละตัว
- Y : อาร์เรย์ขนาด B ที่เก็บลำดับของจำนวนเต็มระบุขีดจำกัดด้านขนาดสำหรับหุ่นยนต์เล็กแต่ละตัว
- W : อาร์เรย์ขนาด T ที่เก็บลำดับของจำนวนเต็มระบุน้ำหนักของของเล่นแต่ละชิ้น
- S : อาร์เรย์ขนาด T ที่เก็บลำดับของจำนวนเต็มระบุขนาดของของเล่นแต่ละชิ้น
- *คืนค่า*: เวลาเป็นนาทีที่น้อยที่สุดที่ต้องใช้ในการเก็บของเล่นทั้งหมด, หรือ `-1` ถ้าไม่สามารถทำได้

ตัวอย่างการติดต่อ

ด้านล่างเป็นการติดต่อของตัวอย่างแรก

พารามิเตอร์	ค่า
A	3
B	2
T	10
X	[6, 2, 9]
Y	[4, 7]
W	[4, 8, 2, 7, 1, 5, 3, 8, 7, 10]
S	[6, 5, 3, 9, 8, 1, 3, 7, 6, 5]
คืนค่า	3

ด้านล่างเป็นการติดต่อในตัวอย่างที่สอง

พารามิเตอร์	ค่า
A	2
B	1
T	3
X	[2, 5]
Y	[2]
W	[3, 5, 2]
S	[1, 3, 2]
คืนค่า	-1

เงื่อนไขบังคับ

- ข้อจำกัดด้านเวลา 3 วินาที
- ข้อจำกัดด้านหน่วยความจำ 64 MiB
- $1 \leq T \leq 1,000,000$
- $0 \leq A, B \leq 50,000$ และ $1 \leq A + B$
- $1 \leq X[i], Y[i], W[i], S[i] \leq 2,000,000,000$

ปัญหาย่อย

ปัญหาย่อย	คะแนน	เงื่อนไขเพิ่มเติมของข้อมูลนำเข้า
1	14	$T = 2$ และ $A + B = 2$ (มีของเล่นสองชิ้นและมีหุ่นยนต์สองตัว)
2	14	$B = 0$ (หุ่นยนต์ทุกตัวเป็นหุ่นยนต์อ่อนแอ)
3	25	$T \leq 50$ และ $A + B \leq 50$
4	37	$T \leq 10,000$ และ $A + B \leq 1,000$
5	10	(ไม่มี)

การทดลอง

เกรดเตอร์ตัวอย่างในเครื่องของคุณจะอ่านข้อมูลนำเข้าจากแฟ้ม `robots.in`, ซึ่งจะต้องอยู่ในรูปแบบต่อไปนี้:

- บรรทัด 1: `A B T`
- บรรทัด 2: `X[0] ... X[A-1]`
- บรรทัด 3: `Y[0] ... Y[B-1]`
- อีก T บรรทัดถัดไป: `W[i] S[i]`

ยกตัวอย่างเช่น ตัวอย่างแรก จะระบุในรูปแบบดังนี้

```
3 2 10
6 2 9
4 7
4 6
8 5
2 3
7 9
1 8
5 1
3 3
8 7
7 6
10 5
```

ถ้า $A = 0$ หรือ $B = 0$ บรรทัดในแฟ้มที่ระบุข้อมูลที่สอดคล้องกับข้อมูลดังกล่าว (บรรทัดที่ 2 หรือ บรรทัดที่ 3) จะต้องเป็นบรรทัดว่าง

หมายเหตุด้านภาษา

C/C++ คุณจะต้องประกาศ `#include "robots.h"` ที่ต้นโปรแกรม

Pascal คุณจะต้องนิยาม `unit Robots` อาเรย์ทั้งหมดจะเริ่มนับที่ 0 (ไม่ใช่ 1)

ดูตัวอย่างเทมเพลตในเครื่องของคุณ