



International Olympiad in Informatics 2013

6-13 July 2013

Brisbane, Australia

Day 2 tasks

robots

Bulgarian — 1.0

Малкият брат на Марита е разхвърлял играчките си по пода на стаята. За щастие Марита е направила специални роботи, които могат да прибират играчките. Тя се нуждае от помощта ви, за определянето на това кои роботи трябва да свършат работата.

Има T играчки, всяка от които има целочислено тегло $W[i]$ и целочислена големина $S[i]$. Роботите са два вида: *слаби* и *малки*.

- Има A слаби роботи. Всеки слаб робот има ограничение $X[i]$ за тежестта на играчката, която може да пренася, като може да пренася само играчка с тегло строго по-малко от $X[i]$. Големината на играчката няма значение.
- Има B малки роботи. Всеки малък робот има ограничение $Y[i]$ за големината на играчката, която може да пренася, като може да пренася само играчка с големина строго по-малка от $Y[i]$. Теглото на играчката няма значение.

На всеки робот му трябва една минута, за да пренесе една играчка. Различните роботи могат да пренасят различните играчки за едно и също време.

Вашата задача е да определите дали роботите на Марита могат да разчистят всички играчки и ако това е възможно, да се намери най-краткото време, за което това може да стане.

Примери

Като първи пример да предположим, че има $A = 3$ слаби работа с ограничения за теглата $X = [6, 2, 9]$, $B = 2$ малки работа с ограничения за големините на играчките $Y = [4, 7]$ и $T = 10$ играчки, за които:

Номер на играчката	0	1	2	3	4	5	6	7	8	9
Тегло	4	8	2	7	1	5	3	8	7	10
Големина	6	5	3	9	8	1	3	7	6	5

Най-краткото време за разчистване на играчките е три минути:

	Слаб робот 0	Слаб робот 1	Слаб робот 2	Малък робот 0	Малък робот 1
Първа минута	Играчка 0	Играчка 4	Играчка 1	Играчка 6	Играчка 2
Втора минута	Играчка 5		Играчка 3		Играчка 8
Трета минута			Играчка 7		Играчка 9

Като втори пример да предположим, че има $A = 2$ слаби робота с ограничения за теглата $X = [2, 5]$, $B = 1$ малък робот с ограничение за големините на играчките $Y = [2]$ и $T = 3$ играчки, за които:

Играчка номер	0	1	2
Тегло	3	5	2
Големина	1	3	2

Никой от роботите не може да пренесе играчката с тегло 5 и големина 3, затова е невъзможно всичките играчки да бъдат разчистени.

Имплементация

Трябва да изпратите файл, който реализира функцията `putaway()` както следва:

Вашата функция: `putaway()`

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;
```

Описание

Функцията трябва да пресмята най-малкия брой минути, необходими за пренасянето на всички играчки или трябва да върне резултат `-1`, ако това е невъзможно.

Параметри

- A : брой на слабите работи.
- B : брой на малките работи.
- T : брой на играчките.

- **X**: Масив от **A** елемента, съдържащ цели числа, задаващи ограниченията за теглата за всеки от слабите работи.
- **Y**: масив от **B** елемента, съдържащ цели числа, който задава ограниченията за големините за всеки от малките работи.
- **W**: масив от **T** елемента, съдържащ цели числа, който задава теглата на всяка от играчките.
- **S**: масив от **T** елемента, съдържащ цели числа, който задава големините на всяка от играчките.
- *Резултат*: Най-малкият брой минути, необходими за разчистването на всички играчки или **-1**, ако това е невъзможно.

Примерна сесия

Следната сесия описва първия пример:

Параметър	Стойност
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]
Резултат	3

Следващата сесия описва втория пример:

Параметър	Стойност
A	2
B	1
T	3
X	[2, 5]
Y	[2]
W	[3, 5, 2]
S	[1, 3, 2]
Резултат	-1

Ограничения

- Време: 3 секунди
- Памет: 64 MiB
- $1 \leq T \leq 1,000,000$
- $0 \leq A, B \leq 50,000$ и $1 \leq A + B$
- $1 \leq X[i], Y[i], W[i], S[i] \leq 2,000,000,000$

Подзадачи

Подзадача	Точки	Допълнителни входни ограничения
1	14	$T = 2$ и $A + B = 2$ (точно две играчки и два робота)
2	14	$B = 0$ (всички работи са слаби)
3	25	$T \leq 50$ и $A + B \leq 50$
4	37	$T \leq 10,000$ и $A + B \leq 1,000$
5	10	(Няма)

Експериментиране

Опростеният грейдер от вашия компютър ще чете входните данни от файл `robots.in`, който трябва да бъде в следния формат:

- ред 1: `A B T`
- ред 2: `X[0] ... X[A-1]`
- ред 3: `Y[0] ... Y[B-1]`
- следващите `T` реда: `W[i] S[i]`

Например първият от горните примери трябва да бъде описан по следния начин:

```
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
```

Ако `A = 0` или `B = 0`, тогава съответният ред (ред 2 или ред 3) трябва да бъде празен.

Бележки по езика за програмиране

C/C++ Трябва да използвате `#include "robots.h"`.

Pascal Трябва да дефинирате `unit Robots`. Всички масиви са номерирани от `0` (не от `1`).

Виж шаблоните на решения от вашата машина за примери.