



International Olympiad in Informatics 2013

6-13 July 2013

Brisbane, Australia

Day 2 tasks

robots

Spanish — 1.0

El hermano chico de Marita ha dejado sus juguetes botados en el living! Por suerte, Marita ha construido pequeños robots para recoger los juguetes. Ella necesita tu ayuda para determinar qué robots deben recoger qué juguetes.

Hay T juguetes, cada uno con peso $W[i]$ y tamaño $S[i]$, ambos números enteros. Hay robots de dos tipos: débiles y pequeños.

- Hay A robots débiles. Cada uno de ellos tiene asociado un límite de peso $X[i]$ y puede transportar cualquier juguete de peso estrictamente menor que $X[i]$. Para los robots débiles, el tamaño del juguete no importa.
- Hay B robots pequeños. Cada uno de ellos tiene asociado un límite de tamaño $Y[i]$, y puede transportar cualquier juguete de tamaño estrictamente menor que $Y[i]$. Para los robots pequeños el peso del juguete no importa.

Cada uno de los robots de Marita tarda un minuto en encargarse de un juguete. Robots diferentes pueden encargarse de juguetes diferentes al mismo tiempo.

Tu tarea es determinar si los robots de Marita pueden ordenar todos los juguetes y, en el caso que puedan hacerlo, encontrar el menor tiempo en el que pueden completar su tarea.

Ejemplos

Suponga que hay $A=3$ robots débiles con límites de peso $X=[6,2,9]$, $B=2$ robots pequeños con límites de tamaño $Y=[4,7]$, y $T=10$ juguetes como se indica a continuación:

Juguete	0	1	2	3	4	5	6	7	8	9
Peso	4	8	2	7	1	5	3	8	7	10
Tamaño	6	5	3	9	8	1	3	7	6	5

El menor tiempo posible para ordenar todos los juguetes es tres minutos. La siguiente tabla muestra una forma de lograrlo:

	Robot débil 0	Robot débil 1	Robot débil 2	Robot pequeño 0	Robot pequeño 1
Minuto 1	Juguete 0	Juguete 4	Juguete 1	Juguete 6	Juguete 2
Minuto 2	Juguete 5		Juguete 3		Juguete 8
Minuto 3			Juguete 7		Juguete 9

Como un segundo ejemplo, suponga que hay $A=2$ robots débiles con límites de peso $X=[2,5]$, $B=1$ robots pequeños con límites de tamaño $Y=[2]$, y $T=3$ juguetes como se indica a continuación:

Juguete	0	1	2
Peso	3	5	2
Tamaño	1	3	2

Note que ningún robot puede tomar el juguete de peso 5 y tamaño 3, y por lo tanto, en este caso es imposible para los robots ordenar los juguetes.

Implementación

Debes enviar un archivo que implemente la función `putaway()` como sigue:

Tu Función: `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;
```

Descripción

Esta función debe calcular la mínima cantidad de minutos requeridos para que los robots ordenen todos los juguetes, o debe retornar `-1` si no es posible ordenarlos.

Parámetros

- A : Número de robots débiles.
- B : Número de robots pequeños.

- **T**: Número de juguetes.
- **X**: Arreglo de largo **A** que contiene enteros que especifican el peso límite para cada robot débil.
- **Y**: Arreglo de largo **B** que contiene enteros que especifican el tamaño límite para cada robot pequeño.
- **W**: Arreglo de largo **T** que contiene enteros que especifican el peso de cada juguete.
- **S**: Arreglo de largo **T** que contiene enteros que especifican el tamaño de cada juguete.
- *Retorna*: El mínimo número de minutos requerido para que los robots ordenen todos los juguetes, o **-1** si no es posible ordenarlos.

Sesión de Ejemplo

La siguiente sesión describe el primer ejemplo:

Parámetro	Valor
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]
Retorna	3

La siguiente sesión describe el segundo ejemplo:

Parámetro	Valor
A	2
B	1
T	3
X	[2, 5]
Y	[2]
W	[3, 5, 2]
S	[1, 3, 2]
Retorna	-1

Restricciones

- Time limit: 3 segundos
- Memory limit: 64 MiB
- $1 \leq T \leq 1,000,000$
- $0 \leq A, B \leq 50,000$ and $1 \leq A + B$
- $1 \leq X[i], Y[i], W[i], S[i] \leq 2,000,000,000$

Subtareas

Subtarea	Puntos	Restricciones Adicionales
1	14	$T = 2$ y $A + B = 2$ (exactamente dos juguetes y dos robots)
2	14	$B = 0$ (todos los robots son débiles)
3	25	$T \leq 50$ y $A + B \leq 50$
4	37	$T \leq 10,000$ y $A + B \leq 1,000$
5	10	(Ninguna)

Experimentación

El evaluador de ejemplo (sample grader) lee el input desde el archivo `robots.in`, que debe estar en el siguiente formato:

- línea 1: `A B T`
- línea 2: `X[0] ... X[A-1]`
- línea 3: `Y[0] ... Y[B-1]`
- siguientes `T` líneas: `W[i] S[i]`

Así, el primer ejemplo de arriba debe ser entregado en el siguiente formato:

```
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
```

Si `A = 0` o `B = 0` entonces la línea correspondiente (línea 2 o línea 3) debe estar vacía.

Notas del Lenguaje

C/C++ Debes incluir la línea `#include "robots.h"`.

Pascal Debes definir `unit Robots`. Todos los arreglos comienzan en `0` (no `1`).

Mira a modo de ejemplo las plantillas de solución en tu máquina.