



## International Olympiad in Informatics 2013

6-13 July 2013

Brisbane, Australia

Day 2 tasks

# robots

Spanish — 1.0

¡El hermano menor de Marita ha dejado unos juguetes en todo el piso de la sala! Afortunadamente, Marita ha desarrollado unos robots especiales para recoger los juguetes. Ella necesita tu ayuda para determinar cuales robots deben recoger cuales juguetes.

Hay  $T$  juguetes, cada uno con un entero peso  $W[i]$  y un entero tamaño  $S[i]$ . Los robots vienen en dos tipos: "débiles" y "pequeños".

- Hay  $A$  robots débiles. Cada robot débil tiene un peso límite  $X[i]$ , y puede cargar cualquier juguete de peso estrictamente menor que  $X[i]$ . El tamaño del juguete no importa.
- Hay  $B$  robots pequeños. Cada robot pequeño tiene un límite de tamaño  $Y[i]$ , y puede cargar cualquier juguete de tamaño estrictamente menor que  $Y[i]$ . El peso del juguete no importa.

A cada uno de los robots de Marita le toma un minuto para recoger un juguete. Robots diferentes pueden recoger diferentes juguetes al mismo tiempo.

Tu tarea es determinar si los robots de Marita pueden recoger todos los juguetes, y de ser así, el tiempo más corto en el que pueden hacer esto.

## Ejemplos

Como primer ejemplo, supón 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 sigue:

Número del 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 tiempo más corto para poner todos los juegos fuera es tres minutos:

	Robot débil 0	Robot débil 1	Robot débil 2	Robot pequeño 0	Robot pequeño 1
Primer minuto	Juguete 0	Juguete 4	Juguete 1	Juguete 6	Juguete 2
Segundo minuto	Juguete 5		Juguete 3		Juguete 8
Tercer minuto			Juguete 7		Juguete 9

Como segundo ejemplo, supón que hay "A = 2" robots débiles con límites de peso "X = [2, 5]", "B = 1" robots pequeños con límite de tamaño "Y = [2]", y "T = 3" juguetes como sigue:

Número del juguete	0	1	2
Peso	3	5	2
Tamaño	1	3	2

Ningún robot puede levantar el juguete de peso 5 y tamaño 3, así que es imposible para los robots sacar todos los juguetes.

## Implementación

Tu 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 el número mínimo de minutos requeridos para que los robots recojan todos los juguetes, o debe retornar `-1` si esto no es posible.

### Parámetros

- `A`: El número de robots débiles.
- `B`: El número de robots pequeños.
- `T`: El número de juguetes.
- `X`: Un arreglo de longitud `A` que contiene enteros que especifican el límite de peso de cada robot débil.

- **Y**: Un arreglo de longitud **B** que contiene enteros que especifican el límite de tamaño de cada robot pequeño.
- **W**: Un arreglo de tamaño **T** que contiene enteros que dan el peso de cada juguete.
- **S**: Un arreglo de tamaño **T** que contiene enteros que dan el tamaño de cada juguete.
- *Retorna*: La cantidad mínima de minutos requerida para sacar todos los juguetes, ó **-1** si esto no es posible.

## Sesión de Ejemplo

La siguiente sesión describe el primer ejemplo seguidamente:

Parámetro	Valor
<b>A</b>	3
<b>B</b>	2
<b>T</b>	10
<b>X</b>	[6, 2, 9]
<b>Y</b>	[4, 7]
<b>W</b>	[4, 8, 2, 7, 1, 5, 3, 8, 7, 10]
<b>S</b>	[6, 5, 3, 9, 8, 1, 3, 7, 6, 5]
<b>Retorna</b>	3

La siguiente sesión describe el segundo ejemplo:

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

## Restricciones

- Tiempo Límite: 3 segundos
- Límite de memoria: 64 MiB (Megabytes)

- $1 \leq T \leq 1,000,000$
- $0 \leq A, B \leq 50,000$  y  $1 \leq A + B$
- $1 \leq X[i], Y[i], W[i], S[i] \leq 2,000,000,000$

## Subtareas

Subtareas	Puntos	Restricciones de entrada 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 que se encuentra en tu computadora leerá la entrada del archivo `robots.in`, que debe estar en el siguiente formato.

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

En instancia, el primer ejemplo anterior debe ser provisto 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 " $A = 0$ " ó " $B = 0$ " entonces la línea correspondiente (línea 2 o 3) debe estar vacía.

---

## Notas del lenguaje

C/C++ Tu debe `#include "robots.h"`.

Pascal Tu debes definir `unit Robots`. Todos los arreglos son numerados empezando en `0` (no `1`).

Ve las plantillas de solución en tu máquina para ejemplos.