



## International Olympiad in Informatics

2013

6-13 July 2013

Brisbane, Australia

Day 2 tasks

# robots

French — 1.0

Le petit frère de Marita a laissé traîner plein de jouets dans le salon ! Heureusement, Marita a développé des robots pour ranger les jouets. Elle a besoin de votre aide pour déterminer quel robot doit ramasser quel jouet.

Il y a  $T$  jouets, chacun ayant un poids entier  $W[i]$  et une taille entière  $S[i]$ . Il y a deux types de robots : les *faibles* et les *petits*.

- Il y a  $A$  robots faibles. Chaque robot faible a une limite de poids  $X[i]$  et ne peut transporter que des jouets de poids strictement inférieur à  $X[i]$ , peu importe leur taille.
- Il y a  $B$  petits robots. Chaque petit robot a une limite de taille  $Y[i]$  et ne peut transporter que des jouets de taille strictement inférieure à  $Y[i]$ , peu importe leur poids.

Chaque robot prend une minute pour ranger un jouet. Différents robots peuvent ranger différents jouets en même temps.

Vous devez déterminer si les robots de Marita peuvent ranger tous les jouets et, si c'est le cas, le temps minimal dont ils ont besoin pour le faire.

## Exemples

Premier exemple : supposons qu'il y a  $A = 3$  robots faibles de limites de poids  $X = [6, 2, 9]$ ,  $B = 2$  petits robots de limites de taille  $Y = [4, 7]$  et les  $T = 10$  jouets suivants :

Numéro du jouet	0	1	2	3	4	5	6	7	8	9
Poids	4	8	2	7	1	5	3	8	7	10
Taille	6	5	3	9	8	1	3	7	6	5

Il leur faut au minimum trois minutes pour ranger tous les jouets :

	<b>Robot faible 0</b>	<b>Robot faible 1</b>	<b>Robot faible 2</b>	<b>Petit robot 0</b>	<b>Petit robot 1</b>
<b>Première minute</b>	Jouet 0	Jouet 4	Jouet 1	Jouet 6	Jouet 2
<b>Deuxième minute</b>	Jouet 5		Jouet 3		Jouet 8
<b>Troisième minute</b>			Jouet 7		Jouet 9

Second exemple : supposons qu'il y a  $A = 2$  robots faibles de limites de poids  $X = [2, 5]$ ,  $B = 1$  petits robots de limites de taille  $Y = [2]$  et les  $T = 3$  jouets suivants :

<b>Numéro du jouet</b>	<b>0</b>	<b>1</b>	<b>2</b>
<b>Poids</b>	3	5	2
<b>Taille</b>	1	3	2

Aucun robot ne peut ramasser le jouet de poids 5 et taille 3, il est donc impossible que les robots rangent tous les jouets.

---

## Implémentation

Vous devez soumettre un fichier qui implémente la fonction `putaway()` comme suit :

### Votre fonction : `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;
```

### Description

Cette fonction doit calculer le nombre minimal de minutes dont les robots ont besoin pour ranger tous les jouets ou retourner `-1` si c'est impossible.

### Paramètres

- `A` : le nombre de robots faibles.
- `B` : le nombre de petits robots.
- `T` : le nombre de jouets.
- `X` : un tableau de taille `A` contenant des entiers spécifiant la limite de poids de chaque robot.
- `Y` : un tableau de taille `B` contenant des entiers spécifiant la limite de taille de chaque robot.
- `W` : un tableau de taille `T` contenant des entiers spécifiant le poids de chaque jouet.
- `S` : un tableau de taille `T` contenant des entiers spécifiant la taille de chaque jouet.
- *Retourne* : le nombre minimal de minutes nécessaires pour ranger tous les jouets ou `-1` si c'est impossible.

---

## Exemple de session

La session suivante décrit le premier exemple :

Paramètre	Valeur
<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>Retourne</b>	3

La session suivante décrit le second exemple :

Paramètre	Valeur
<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>Retourne</b>	-1

---

## Contraintes

- Limite de temps : 3 secondes
- Limite de mémoire : 64 Mio
- $1 \leq T \leq 1\,000\,000$
- $0 \leq A, B \leq 50\,000$  et  $1 \leq A + B$
- $1 \leq X[i], Y[i], W[i], S[i] \leq 2\,000\,000\,000$

---

## Sous-tâches

Sous-tâche	Points	Contraintes d'entrée supplémentaires
1	14	$T = 2$ and $A + B = 2$ (exactement deux jouets et deux robots)
2	14	$B = 0$ (que des robots faibles)
3	25	$T \leq 50$ et $A + B \leq 50$
4	37	$T \leq 10\,000$ et $A + B \leq 1\,000$
5	10	(Aucune)

---

## Expérimentation

L'évaluateur fourni sur votre ordinateur lira en entrée le fichier `robots.in`, qui doit être au format suivant :

- ligne 1 : `A B T`
- ligne 2 : `X[0]` à `X[A-1]`
- ligne 3 : `Y[0]` à `Y[B-1]`
- les `T` lignes suivantes : `W[i] S[i]`

Par exemple, le premier exemple doit être fourni au format suivant :

```
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` ou `B = 0` alors la ligne correspondante (ligne 2 ou ligne 3) doit être vide.

---

## Remarques pour chaque langage

C/C++ You devez utiliser `#include "robots.h"`.

Pascal Vous devez utiliser l'unité `unit Robots`. Tous les tableaux sont numérotés à partir de `0` (et non `1`).

Prenez exemple sur les modèles de solution fournis sur votre machine.