



## International Olympiad in Informatics 2013

6-13 July 2013

Brisbane, Australia

Day 2 tasks

# robots

Norwegian — 1.0

Maritas lillebror har lagt igjen leker over hele stuegulvet! Heldigvis har Marita utviklet spesialroboter for å rydde bort lekene. Hun trenger din hjelp med å avgjøre hvilke roboter som burde plukke opp hvilke leker.

Det finnes  $T$  leker, hver med en heltallsvekt  $W[i]$  og heltallsstørrelse  $S[i]$ . Det finnes to typer roboter: *svake* (*weak*) og *små* (*small*).

- Det finnes  $A$  svake roboter. Hver svake robot har en vektbegrensning  $X[i]$ , og kan bære en hvilken som helst leke som veier strengt mindre enn  $X[i]$ . Lekens størrelse har ingen betydning.
- Det finnes  $B$  små roboter. Hver lille robot har en størrelsesbegrensning  $Y[i]$ , og kan bære en hvilken som helst leke som har størrelse strengt mindre enn  $Y[i]$ . Lekens vekt har ingen betydning.

Hver av Maritas roboter bruker ett minutt på å rydde bort en leke. Forskjellige roboter kan rydde bort forskjellige leker samtidig.

Oppgaven din er å avgjøre om Maritas roboter kan rydde vekk alle lekene, og i så fall, hva som er den minste tiden de trenger for å gjøre det.

## Eksempler

Som et første eksempel, anta at det er  $A = 3$  svake roboter med vektbegrensninger  $X = [6, 2, 9]$ ,  $B = 2$  små roboter med størrelsesbegrensninger  $Y = [4, 7]$ , og  $T = 10$  leker som følger:

Lekenummer	0	1	2	3	4	5	6	7	8	9
Vekt	4	8	2	7	1	5	3	8	7	10
Størrelse	6	5	3	9	8	1	3	7	6	5

Den minste tiden som trengs for å rydde bort alle lekene er tre minutter:

	<b>Svak robot 0</b>	<b>Svak robot 1</b>	<b>Svak robot 2</b>	<b>Liten robot 0</b>	<b>Liten robot 1</b>
<b>Første minutt</b>	Leke 0	Leke 4	Leke 1	Leke 6	Leke 2
<b>Andre minutt</b>	Leke 5		Leke 3		Leke 8
<b>Tredje minutt</b>			Leke 7		Leke 9

Som et andre eksempel, anta at det er  $A = 2$  svake roboter med vektbegrensninger  $X = [2, 5]$ ,  $B = 1$  liten robot med størrelsesbegrensning  $Y = [2]$ , og  $T = 3$  leker som følger:

<b>Lekenummer</b>	<b>0</b>	<b>1</b>	<b>2</b>
<b>Vekt</b>	3	5	2
<b>Størrelse</b>	1	3	2

Ingen robot er i stand til å plukke opp leken med vekt 5 og størrelse 3, så det er umulig for robotene å rydde bort alle lekene.

---

## Implementasjon

Du skal sende inn en fil som implementerer funksjonen `putaway()` som følger:

### Din function: `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;
```

### Beskrivelse

Denne funksjonen skal beregne det minste antallet minutter som robotene trenger for å rydde bort alle lekene, eller så skal den returnere `-1` hvis dette ikke er mulig.

### Parametre

- `A`: Antallet svake roboter.
- `B`: Antallet små roboter.
- `T`: Antallet leker.
- `X`: Et array med lengde `A` som inneholder heltall som spesifiserer vektgrensen for hver svake robot.
- `Y`: Et array med lengde `B` som inneholder heltall som spesifiserer størrelsesgrensen for hver lille robot.
- `W`: Et array med lengde `T` som inneholder heltall som gir vekten av hver leke.
- `S`: Et array med lengde `T` som inneholder heltall som gir størrelsen til hver leke.
- *Returnerer*: Det minste antallet minutter som kreves for å rydde bort alle lekene, eller `-1` hvis dette ikke er mulig.

---

## Sample Session

The following session describes the first example above:

Parameter	Value
<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>Returns</b>	3

The following session describes the second example above:

Parameter	Value
<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>Returns</b>	-1

---

## Constraints

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

---

## Subtasks

Subtask	Points	Additional Input Constraints
1	14	$T = 2$ and $A + B = 2$ (exactly two toys and two robots)
2	14	$B = 0$ (all robots are weak)
3	25	$T \leq 50$ and $A + B \leq 50$
4	37	$T \leq 10,000$ and $A + B \leq 1,000$
5	10	(None)

---

## Experimentation

The sample grader on your computer will read input from the file `robots.in`, which must be in the following format:

- line 1: `A B T`
- line 2: `X[0] ... X[A-1]`
- line 3: `Y[0] ... Y[B-1]`
- the next `T` lines: `W[i] S[i]`

For instance, the first example above should be provided in the following format:

```
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
```

If `A = 0` or `B = 0` then the corresponding line (line 2 or line 3) should be empty.

---

## Language Notes

**C/C++** You must `#include "robots.h"`.

**Pascal** You must define the `unit Robots`. All arrays are numbered beginning at `0` (not `1`).

See the solution templates on your machine for examples.