



## International Olympiad in Informatics 2013

6-13 July 2013

Brisbane, Australia

Day 2 tasks

# robots

Lithuanian — 1.0

Marytės broliukas paskleidė žaislus po visą kambarį! Laimei, Marytės išrasti robotai sugeba juos nešioti į vietą.

Padėkite jai suplanuoti, kurie robotai kuriuos žaislus turi padėti į vietą.

Iš viso yra  $T$  žaislų, kurių svoriai  $W[i]$  ir dydžiai  $S[i]$ . Robotai yra dviejų rūšių: *silpni* ir *maži*.

- Silpnų robotų yra  $A$ . Toks robotas turi svorio ribojimą  $X[i]$ , reiškiantį, kad jis gali paimti žaislą, kurio svoris griežtai mažesnis nei  $X[i]$ . Žaislo dydis nesvarbus.
- Mažų robotų yra  $B$ . Toks robotas turi dydžio ribojimą  $Y[i]$ , reiškiantį, kad jis gali paimti žaislą, kurio dydis griežtai mažesnis nei  $Y[i]$ . Žaislo svoris nesvarbus.

Bet kuris Marytės robotas per vieną minutę gali paimti vieną žaislą ir jį padėti į vietą. Robotai gali dirbti nepriklausomai vienas nuo kito, todėl vienu metu gali būti nešami keli žaislai.

Jūsų užduotis – nustatyti, ar Marytės robotai gali sutvarkyti visus žaislus, ir jei taip, kiek trumpiausiai jiems reiktų dirbti.

## Pavyzdžiai

Kaip pirmą pavyzdį nagrinėkime situaciją, kai yra  $A = 3$  silpni robotai, kurių svorio ribojimai yra  $X = [6, 2, 9]$ , ir  $B = 2$  maži robotai, kurių dydžio ribojimai yra  $Y = [4, 7]$ . Taip pat yra  $T = 10$  žaislų:

Žaislo nr.	0	1	2	3	4	5	6	7	8	9
Svoris	4	8	2	7	1	5	3	8	7	10
Dydis	6	5	3	9	8	1	3	7	6	5

Trumpiausias tvarkymosi laikas yra trys minutės:

	Silpnas robotas nr. 0	Silpnas robotas nr. 1	Silpnas robotas nr. 2	Mažas robotas nr. 0	Mažas robotas nr. 1
Pirma minutė	Žaislas nr. 0	Žaislas nr. 4	Žaislas nr. 1	Žaislas nr. 6	Žaislas nr. 2
Antra minutė	Žaislas nr. 5		Žaislas nr. 3		Žaislas nr. 8
Trečia minutė			Žaislas nr. 7		Žaislas nr. 9

Kaip antrą pavyzdį nagrinėkime situaciją, kai yra  $A = 2$  silpni robotai, kurių svorio ribojimai yra  $X = [2, 5]$ , ir  $B = 1$  mažas robotas, kurio dydžio ribojimas yra  $Y = [2]$ . Taip pat yra  $T = 3$  žaislai:

Žaislo nr.	0	1	2
Svoris	3	5	2
Dydis	1	3	2

Joks robotas negali paimti dydžio 5 ir svorio 3 žaislo, todėl robotai negali sutvarkyti kambario.

## Realizavimas

Pateikite failą, realizuojantį funkciją `putaway()`:

### Jūsų funkcija: `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;
```

### Veikimas

Ši funkcija turi suskaičiuoti, kiek mažiausiai minučių pakaktų robotams padėti visus žaislus į vietą, arba grąžinti `-1`, jei tai neįmanoma.

### Parametrai

- `A`: silpnų robotų skaičius.
- `B`: mažų robotų skaičius.

- **T** : žaislų skaičius.
- **X** : **A** ilgio sveikų skaičių masyvas, pateikiantis silpnų robotų svorio ribojimus.
- **Y** : **A** ilgio sveikų skaičių masyvas, pateikiantis mažų robotų dydžio ribojimus.
- **W** : **T** ilgio sveikų skaičių masyvas, pateikiantis žaislų svorius.
- **S** : **T** ilgio sveikų skaičių masyvas, pateikiantis žaislų dydžius.
- *return*: mažiausias minučių skaičius, kurio pakaktų visus žaislus padėti į vieta; jei to padaryti neįmanoma, **-1**.

## Pavyzdinis veikimas

Tokia sesija atitinka pirmą pavyzdį.

Parametras	Vertė
<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>return</b>	3

Tokia sesija atitinka antrą pavyzdį.

Parametras	Vertė
<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>return</b>	-1

## Ribojimai

- Laiko ribojimas: 3 sekundės
- Atminties ribojimas: 64 MiB
- $1 \leq T \leq 1,000,000$
- $0 \leq A, B \leq 50,000$  ir  $1 \leq A + B$
- $1 \leq X[i], Y[i], W[i], S[i] \leq 2,000,000,000$

## Dalinės užduotys

Dalinė užduotis	Taškai	Papildomi ribojimai
1	14	$T = 2$ ir $A + B = 2$ (du žaislai ir du robotai)
2	14	$B = 0$ (visi robotai silpni)
3	25	$T \leq 50$ ir $A + B \leq 50$
4	37	$T \leq 10,000$ ir $A + B \leq 1,000$
5	10	( <i>Nėra</i> )

## Eksperimentavimas

Jums pateiktas pavyzdinis vertintojas skaitys failą `robots.in`, kurio formatas toks:

- Pirma eilutė: `A B T`
- Antra eilutė: `X[0] ... X[A-1]`
- Trečia eilutė: `Y[0] ... Y[B-1]`
- Kitos `T` eilučių: `W[i] S[i]`

Pirmas anksčiau aprašytas pavyzdys būtų pateiktas tokiu formatu:

```
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
```

Jei `A = 0` arba `B = 0`, tai atitinkama eilutė (antra arba trečia) turi būti tuščia.

---

## Pastabos apie programavimo kalbas

C/C++ Įterpkite `#include "robots.h"`.

Pascal Apibrėžkite `unit Robots`. Visi masyvai numeruojami nuo `0` (ne nuo `1`).

Naudokite pateiktą sprendimo šabloną kaip pavyzdį.