



## International Olympiad in Informatics 2013

6-13 July 2013

Brisbane, Australia

Day 2 tasks

# robots

Croatian — 1.0

Maričin manji brat ostavio je igračke razbacane po sobi. Srećom, Marica ima specijalne robote koji će počistiti kaos. Vaš zadatak je pomoći Marici odlučiti kako će roboti pospremiti igračke.

Zadano je  $T$  igračaka, od kojih svaka ima svoju cjelobrojnu težinu  $W[i]$  i cjelobrojnu veličinu  $S[i]$ . Roboti mogu biti *mlohavi* i *mali*.

- Imamo  $A$  mlohavih robota. Svaki mlohavi robot ima ograničenje težine  $X[i]$ , i moći će nositi samo one igračke čija je težina strogo manja od  $X[i]$ . Veličina nije bitna.
- Imamo  $B$  malih robota. Svaki mali robot ima ograničenje veličine  $Y[i]$ , i može nositi samo one igračke čija je veličina strogo manja od  $Y[i]$ . Težina igračke nije bitna.

Svakom Maričinom robotu treba jedna minuta da pospremi igračku. Svaki robot u jednom trenutku može pospremati najviše jednu igračku. Različiti roboti mogu pospremati različite igračke odjednom.

Vaš zadatak je utvrditi mogu li Maričini roboti pospremiti sve igračke i u tom slučaju odrediti najmanje moguće vrijeme da to naprave.

## Primjer

Imamo  $A = 3$  mlohava robota s ograničenjima težine  $X = [6, 2, 9]$  i  $B = 2$  malih robota s ograničenjima veličine  $Y = [4, 7]$  i  $T = 10$  igračaka kako slijedi:

Oznaka igračke	0	1	2	3	4	5	6	7	8	9
Težina	4	8	2	7	1	5	3	8	7	10
Veličina	6	5	3	9	8	1	3	7	6	5

Najkraće vrijeme u kojem roboti mogu pospremiti sve igračke je tri minute:

	Mlohavi robot 0	Mlohavi robot 1	Mlohavi robot 2	Mali robot 0	Mali robot 1
1. min	Igračka 0	Igračka 4	Igračka 1	Igračka 6	Igračka 2
2. min	Igračka 5		Igračka 3		Igračka 8
3. min			Igračka 7		Igračka 9

U drugom primjeru imamo  $A = 2$  mlohava robota s ograničenjima težine  $X = [2, 5]$  i  $B = 1$  malih robota s ograničenjem veličine  $Y = [2]$  te  $T = 3$  igračke kako slijedi:

Oznaka igračke	0	1	2
Težina	3	5	2
Veličina	1	3	2

Niti jedan robot ne može podignuti igračku težine 5 i veličine 3 i stoga je nije moguće pospremiti sve igračke.

---

## Implementacija

Potrebno je priložiti datoteku koja implementira funkciju `putaway()` kako slijedi:

### Vaša 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;
```

### Opis

Ova funkcija računa najmanji broj minuta koji je potreban kako bi roboti pospremili sve igračke ili vraća `-1` ako je to nije moguće.

### Parametri

- $A$  : Broj mlohavih robota.
- $B$  : Broj malih robota.
- $T$  : Broj igračaka.
- $X$  : Niz duljine  $A$  koji sadrži cjelobrojna ograničenja težine za svakog mlohavog robota.
- $Y$  : Niz duljine  $B$  koji sadrži cjelobrojna ograničenja veličine za svakog malog robota.
- $W$  : Niz duljine  $T$  koji sadrži cjelobrojne težine svake igračke.
- $S$  : Niz duljine  $T$  koji sadrži cjelobrojne veličine svake igračke.
- *Povratna vrijednost*: Najmanji broj minuta koji je potreban kako bi roboti pospremili sve igračke ili `-1` ako to nije moguće.

---

## Testni podaci

Prvi primjer:

Parametar	Vrijednost
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]
Rješenje	3

Drugi primjer:

Parametar	Vrijednost
A	2
B	1
T	3
X	[2, 5]
Y	[2]
W	[3, 5, 2]
S	[1, 3, 2]
Rješenje	-1

---

## Ograničenja

- Vremensko ograničenje: 3 sekunde
- Memorijsko ograničenje: 64 MB
- $1 \leq T \leq 1,000,000$
- $0 \leq A, B \leq 50,000$  i  $1 \leq A + B$
- $1 \leq X[i], Y[i], W[i], S[i] \leq 2,000,000,000$

## Bodovanje

Podzadatak	Bodovi	Dodatna ograničenja
1	14	$T = 2$ i $A + B = 2$ (točno dvije igračke na dva robota)
2	14	$B = 0$ (svi su vam roboti mlohavi)
3	25	$T \leq 50$ i $A + B \leq 50$
4	37	$T \leq 10,000$ i $A + B \leq 1,000$
5	10	(n/a)

## Lokalno testiranje

Grejder na vašem računalu će pročitati datoteku `robots.in`, koja mora biti u sljedećem obliku:

- redak 1: `A B T`
- redak 2: `X[0] ... X[A-1]`
- redak 3: `Y[0] ... Y[B-1]`
- sljedećih `T` redaka: `W[i] S[i]`

Npr., prvi primjer bi bio predstavljen:

```
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
```

Ako je `A = 0` ili `B = 0` pripadajući retci u ulaznoj datoteci trebali bi biti prazni.

---

## Napomene

C/C++ Potrebno je dodati `#include "robots.h"`.

Pascal Potrebno je definirati `unit Robots`. Svi nizovi moraju biti indeksirani od `0` (a ne od `1`).

Pogledajte predloške rješenja na vašem računalu.