



International Olympiad in Informatics 2013

6-13 July 2013

Brisbane, Australia

Day 2 tasks

robots

Netherlands — 1.0

Al het speelgoed van Marita's kleine broertje ligt op de vloer!

Gelukkig heeft Marita speciale robots gebouwd om het speelgoed op te ruimen.

Marita heeft jouw hulp nodig om te bepalen welke robot welk speelgoed moet oppakken.

Er zijn T speeltjes die elk een gewicht (integer) $W[i]$ en omvang (integer) $S[i]$ hebben.

Er zijn twee soorten robots: *zwak* en *klein*.

- Er zijn A zwakke robots. Elke zwakke robot heeft een gewichtslimiet van $X[i]$, en kan alleen speeltjes tillen met een gewicht dat echt minder is dan $X[i]$. De grootte van het speeltje maakt niet uit.
- Er zijn B kleine robots. Elke kleine robot heeft een omvangslimiet van $Y[i]$, en kan alleen speeltjes tillen met een omvang die echt minder is dan $Y[i]$. Het gewicht van het speeltje maakt niet uit.

Het kost een robot 1 minuut om een speeltje op te ruimen. Verschillende robots kunnen tegelijkertijd verschillende speeltjes opruimen.

Bepaal of het mogelijk is voor de robots om alle speelgoed op te bergen. En, zo ja, hoeveel tijd ze hier minimaal voor nodig hebben.

Voorbeelden

Een eerste voorbeeld:

Stel dat er $A = 3$ zwakke robots zijn die de volgende gewichtslimieten hebben: $X = [6, 2, 9]$.
Stel dat er $B = 2$ kleine robots met omvangslimieten $Y = [4, 7]$ zijn, en dat er verder $T = 10$ speeltjes zijn.

Speeltje nummer	0	1	2	3	4	5	6	7	8	9
Gewicht	4	8	2	7	1	5	3	8	7	10
Omvang	6	5	3	9	8	1	3	7	6	5

Het kost minimaal drie minuten om alle speeltjes op te ruimen:

	Zwakke robot 0	Zwakke robot 1	Zwakke robot 2	Kleine robot 0	Kleine robot 1
Eerste minuut	Speeltje 0	Speeltje 4	Speeltje 1	Speeltje 6	Speeltje 2
Tweede minuut	Speeltje 5		Speeltje 3		Speeltje 8
Derde minuut			Speeltje 7		Speeltje 9

Een tweede voorbeeld:

Stel dat er $A = 2$ zwakke robots zijn die de volgende gewichtslimieten hebben: $X = [2, 5]$. Stel dat er $B = 1$ kleine robot met omvangslimiet $Y = [2]$ is, en dat er verder $T = 3$ speeltjes zijn.

Speeltje	0	1	2
Gewicht	3	5	2
Omvang	1	3	2

Geen enkele robot kan het speeltje met gewicht 5 en omvang 3 oppakken. Het is dus onmogelijk voor de robots om alle speeltjes op te ruimen.

Implementatie

Lever een bestand in dat de functie `putaway()` als volgt implementeert:

Jouw Functie: `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;
```

Beschrijving

Deze functie moet het kleinste aantal minuten bepalen waarin de robots alle speeltjes kunnen opruimen. Als dit niet mogelijk is, lever dan `-1` als resultaat.

Parameters

- `A`: Het aantal zwakke robots.
- `B`: Het aantal kleine robots.
- `T`: Het aantal speeltjes.
- `X`: Een array van lengte `A` dat integers bevat die voor elke zwakke robot de gewichtslimiet aangeven.
- `Y`: Een array van lengte `B` dat integers bevat die voor elke kleine robot de omvangslimiet aangeven.
- `W`: Een array van lengte `T` dat integers bevat die het gewicht van elk speeltje aangeven.
- `S`: Een array van lengte `T` dat integers bevat die de omvang van elke speeltje aangeven.
- *Resultaat*: Het kleinste aantal minuten dat nodig is om alle speeltjes op te ruimen, of `-1` als dit niet mogelijk is.

Voorbeeld

De volgende sessie beschrijft het eerste voorbeeld:

Parameter	Waarde
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]
Resultaat	3

De volgende sessie beschrijft het tweede voorbeeld:

Parameter	Waarde
A	2
B	1
T	3
X	[2, 5]
Y	[2]
W	[3, 5, 2]
S	[1, 3, 2]
Resultaat	-1

Randvoorwaarden

- Tijdslimiet: 3 seconden
- Geheugenlimiet: 64 Mb
- $1 \leq T \leq 1.000.000$
- $0 \leq A, B \leq 50.000$ en $1 \leq A + B$
- $1 \leq X[i], Y[i], W[i], S[i] \leq 2.000.000.000$

Subtasks

Subtask	Punten	Aanvullende randvoorwaarden
1	14	$T = 2$ and $A + B = 2$ (precies twee speeltjes en twee robots)
2	14	$B = 0$ (alle robots zijn zwak)
3	25	$T \leq 50$ en $A + B \leq 50$
4	37	$T \leq 10,000$ en $A + B \leq 1,000$
5	10	(Geen)

Stoeien

De voorbeeldgrader leest invoer uit `robots.in`. Dit bestand moet in het volgende format zijn:

- regel 1: `A B T`
- regel 2: `X[0] ... X[A-1]`
- regel 3: `Y[0] ... Y[B-1]`
- de volgende `T` regels: `W[i] S[i]`

Het eerste voorbeeld zou je in het volgende format moeten aangeven:

```
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
```

Als `A = 0` of `B = 0` dan moeten de bijhorende regels (regel 2 of regel 3) leeg zijn.

Language Notes

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

Pascal Definieer `unit Robots`. Alle arrays zijn genummerd vanaf `0` (niet `1`).

Op je computer staan templates van de oplossingen. Bekijk die voor voorbeelden.