



International Olympiad in Informatics 2013

6-13 July 2013

Brisbane, Australia

Day 2 tasks

robots

Português — 1.0

O irmão mais novo de Marita largou brinquedos no chão da sala! Felizmente, Marita desenvolveu robôs especiais para guardar os brinquedos. Ela precisa da sua ajuda para determinar quais robôs devem pegar quais brinquedos.

Existem T brinquedos, cada um com peso inteiro $W[i]$ e um tamanho inteiro $S[i]$. Os robôs são de dois tipos: "fracos" e "pequenos".

- Existem A robôs fracos. Cada robô fraco possui um limite de peso $X[i]$, e pode carregar qualquer brinquedo com peso estritamente menor do que $X[i]$. O tamanho do brinquedo não importa.
- Existem B robôs pequenos. Cada robô pequeno possui um limite de tamanho $Y[i]$, e pode carregar qualquer brinquedo com tamanho estritamente menor do que $Y[i]$. O peso do brinquedo não importa.

Cada robô de Marita leva um minuto para colocar cada brinquedo no lugar. Diferentes robôs podem organizar diferentes brinquedos ao mesmo tempo.

Sua tarefa é determinar se os robôs de Marita podem guardar todos os brinquedos, e se puderem, o menor tempo no qual eles podem fazer isso.

Exemplos

Como um primeiro exemplo, suponha que existem $A=3$ robôs fracos com limites de pesos $X=[6,2,9]$, $B=2$ robôs pequenos com limite de tamanhos $Y=[4,7]$, e $T=10$ brinquedos como a seguir:

Número do brinquedo	0	1	2	3	4	5	6	7	8	9
Peso	4	8	2	7	1	5	3	8	7	10
Tamanho	6	5	3	9	8	1	3	7	6	5

O menor tempo para guardar todos os brinquedos é três minutos:

	Robô fraco 0	Robô fraco 1	Robô fraco 2	Robô pequeno 0	Robô pequeno 1
Primeiro minuto	Brinquedo 0	Brinquedo 4	Brinquedo 1	Brinquedo 6	Brinquedo 2
Segundo minuto	Brinquedo 5		Brinquedo 3		Brinquedo 8
Terceiro minuto			Brinquedo 7		Brinquedo 9

Como um segundo exemplo, suponha que existem $A=2$ robôs fracos com limites de peso $X=[2,5]$, $B=1$ robôs pequenos com limite de tamanhos $Y=[2]$, e $T=3$ brinquedos como a seguir:

Número do brinquedo	0	1	2
Peso	3	5	2
Tamanho	1	3	2

Nenhum robô pode pegar o brinquedo de peso 5 e tamanho 3, assim é impossível para estes robôs guardarem todos os brinquedos.

Implementação

Você deve submeter um arquivo que implemente a função `putaway()` como a seguir:

Sua função: `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;
```

Descrição

Esta função deve calcular o número mínimo de minutos requeridos para que os robôs guardem todos os brinquedos, ou deve devolver `-1` se isto não for possível.

Parâmetros

- A : O número de robôs fracos.
- B : O número de robôs pequenos.
- T : O número de brinquedos.

- **X**: Um vetor de tamanho **A** contendo inteiros que especificam os limites de peso para cada robô fracos.
- **Y**: Um vetor de tamanho **B** contendo inteiros que especificam os limites de tamanho para cada robô pequeno.
- **W**: Um vetor de tamanho **T** contendo inteiros que especificam o peso de cada brinquedo.
- **S**: Um vetor de tamanho **T** contendo inteiros que especificam o tamanho de cada brinquedo.
- *Devolve*: O menor número de minutos necessários para guardar todos os brinquedos, ou **-1** se isto não for possível.

Sessão de exemplos

A sessão seguinte descreve o primeiro exemplo acima:

Parâmetro	Valor
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]
Devolve	3

A sessão seguinte descreve o segundo exemplo acima:

Parâmetro	Valor
A	2
B	1
T	3
X	[2, 5]
Y	[2]
W	[3, 5, 2]
S	[1, 3, 2]
Devolve	-1

Restrições

- Limite de tempo: 3 segundos
- Limite de memória: 64 MiB
- $1 \leq T \leq 1,000,000$
- $0 \leq A, B \leq 50,000$ e $1 \leq A + B$
- $1 \leq X[i], Y[i], W[i], S[i] \leq 2,000,000,000$

Subtarefa

Subtarefa	Pontos	Restrições adicionais
1	14	$T = 2$ e $A + B = 2$ (exatamente dois brinquedos e dois robôs)
2	14	$B = 0$ (todos os robôs são fracos)
3	25	$T \leq 50$ e $A + B \leq 50$
4	37	$T \leq 10,000$ e $A + B \leq 1,000$
5	10	(Nenhuma)

Experimentação

O validador de exemplo no seu computador lerá a entrada do arquivo `robots.in`, qual deve estar no seguinte formato:

- linha 1: `A B T`
- linha 2: `X[0] ... X[A-1]`
- linha 3: `Y[0] ... Y[B-1]`
- as próximas `T` linhas: `W[i] S[i]`

Por exemplo, o primeiro exemplo acima deve ser fornecido no seguinte formato:

```
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
```

Se `A=0` ou `B=0` então as linhas correspondentes (linha 2 ou linha 3) deve ser vazia.

Notas da linguagem

C/C++ Você deve incluir `#include "robots.h"`.

Pascal Você deve definir a `unit Robots`. Todos os vetores são numerados iniciando por `0` (e não `1`).

Veja os templates de solução na sua máquina como exemplos.