

## Eligiendo Carreras

Contribución de Lautaro Lasorsa

### Descripción del problema

Ana y Joe son una pareja y ambos están pensando en escoger su carrera universitaria (saben que irán a la misma universidad). Y necesitan tu ayuda.

Cada uno tiene varias carreras que le interesan, y le es indistinto cursar cualquiera de ellas. Además, tienen la intención de cursar juntos la mayor parte de las carreras que sea posible.

Como aún están estudiando los planes de estudio, las carreras que les interesan pueden cambiar. Por eso quieren, después de cada vez que uno de ellos cambia sus intereses, saber cual es la mayor cantidad de materias que pueden cursar juntos, y las carreras con las que ocurriría esto. Si hay más de una posibilidad, les es indistinto cuál sea.

La universidad a la que piensan asistir es muy particular, ya que en ella los alumnos de cada carrera cursan una sola materia por cuatrimestre. Por lo tanto, cada carrera puede describirse como una lista de enteros, donde el  $i$  –esimo entero es la materia cursada en el  $i$  –esimo cuatrimestre.

Cada materia se identifica con un número entre 1 y 1.000.000.000 . Además, se sabe que si en un cuatrimestre las dos carreras no tienen la misma materia, no tendrán la misma materia en ningún cuatrimestre posterior.

### Detalles de implementación

Debes implementar las funciones

- `Inicializar(N)`, una función que será llamada una unica vez al inicio del programa, siendo el ENTERO  $N$  la cantidad de veces que Ana o Joe actualizarán sus intereses. No debe retornar nada.

- `Actualizar(car, num, per, op, maxp, ca, cj)`, recibe *car*, que es un VECTOR DE ENTEROS que contiene describe una carrera.

Además recibe los ENTERO *num*, *per* y *op*, el primero es el número que identifica a la carrera, el segundo indica cuál de los 2 realiza la actualización (0 para Ana y 1 para Juan) y el segundo indica que lo que hace es agregarla (1) o quitarla (0) de su lista de intereses. Obviamente solo quitan una carrera que les interesaba y no agregan una carrera que ya estaba entre sus intereses.

Por último, recibe 3 variables que son REFERENCIA A ENTERO, en donde debe guardar la respuesta a la consulta. *maxp* es la máxima cantidad de cuatrimestres que pueden cursar juntos (puede ser 0), y *ca* y *cj* son las carreras que Ana y Joe, respectivamente, deberen elegir para lograr ese objetivo. Si alguno de los 2 no está interesado en ninguna carrera, los 3 valores deben colocarse en -1. No debe retornar nada.

Inicialmente ninguno está interesado en ninguna carrera.

### Evaluable local

El Evaluador Local primero leera una línea con el valor  $N$ . Luego llamará a `Inicializar(N)`.

Después leera las  $N$  actualizaciones. En cada actualización leera primero una línea con 4 enteros : *num*, *per*, *op* y  $M$ , siendo  $M$  la cantidad de cuatrimestres que dura esa carrera.

La segunda línea contendrá los  $M$  enteros que describen el plan de estudios de la carrera.

Luego llamará a `Actualizar(car, num, per, op, maxp, ca, cj)` y mostrará en una línea los valores *maxp*, *ca* y *cj*.

### Cotas

- Siendo  $T$  la suma de los largos de los valores  $M$  de todas las actualizaciones.  $1 \leq T \leq 200.000$
- $1 \leq car_i, num \leq 1.000.000.000$

### Ejemplos

Si el evaluador local recibe la siguiente entrada:

```
10
1 0 1 5
1 2 3 4 5
1 1 1 5
1 2 3 4 5
5 1 1 6
1 2 3 4 5 9
1 1 0 5
1 2 3 4 5
1 0 0 5
1 2 3 4 5
1000 0 1 7
1 2 3 6 7 8 10
50 0 1 3
100 101 102
1000 0 0 7
1 2 3 6 7 8 10
10000 1 1 4
100 103 105 1100
10000 1 0 4
100 103 105 1100
```

Una implementación correcta podrá devolver:

```
-1 -1 -1
5 1 1
5 1 1
5 1 5
-1 -1 -1
3 1000 5
3 1000 5
0 50 5
1 50 10000
0 50 5
```

En cambio, si recibe:

```
6
1 0 1 10
1 2 3 4 5 6 7 8 9 10
2 0 1 5
11 12 13 14 15
3 0 1 7
100 99 98 97 96 95 94
4 0 1 1
100
5 1 1 1
314
3 1 1 7
100 99 98 97 96 95 94
```

Podrá devolver:

```
-1 -1 -1
-1 -1 -1
-1 -1 -1
-1 -1 -1
0 1 5
7 3 3
```

### Subtareas

- $1 \leq T \leq 500$ . (10 puntos)
- $1 \leq T \leq 3000$ . (20 puntos)
- Nunca ninguno de los 2 elimina una carrera de sus intereses (25 puntos)
- Sin restricciones adicionales. (45 puntos)