

Señalizando un camino...

Contribución de Laura Rivero

Descripción del problema

Se ha señalado una ruta con **N** carteles para alertar a los automovilistas sobre las dificultades que presenta. Existen **T** tipos diferentes de dificultad, como por ejemplo curvas, calzada reducida, puente angosto, peligro de desmoronamiento o cruce de animales. Cada tipo se identifica con una letra del alfabeto inglés.

Unos obreros noveles encargados de señalar la ruta recibieron la cantidad exacta de carteles de cada tipo, de acuerdo a las dificultades registradas en los planos del trazado.

La inspectora vial ha recorrido la ruta, y vio con horror que los obreros no siempre acertaron con el lugar correcto de colocación de cada cartel.

Para remediar la situación, la inspectora ha ordenado que una experimentada cuadrilla repare la situación. Esta saldrá de un destacamento vial al comienzo de la ruta (en el kilómetro 0) y culminará su jornada en otro destacamento al final de la ruta (en el kilómetro 1.000, es decir a **1.000.000 de metros** del comienzo de la ruta).

Para facilitar el trabajo de la cuadrilla, la inspectora recopiló toda la información necesaria. Concretamente, elaboró un listado de **2N** ítems, cada uno de los cuales indica:

- La distancia **en metros** desde el comienzo de la ruta.
- Si en esa ubicación hay un cartel, o si debería haberlo.
- El tipo de dificultad y por lo tanto de cartel.

El costo de toda la operación se calcula sumando dos rubros:

- Remoción o instalación de carteles: costo de **K** pesos por cada operación.
- Movimiento del vehículo. El costo será de **C** pesos por cada metro recorrido.

La inspectora está atenta a todo, y para ayudar a disminuir el costo total de la maniobra, el vehículo no parte vacío sino que

lleva un suministro de carteles extra al salir del campamento inicial. Se sabe la cantidad exacta y los tipos de carteles extra con los que el vehículo inicia la travesía. Además, el vehículo tiene una capacidad ilimitada de almacenamiento de carteles.

La inspectora desea saber el costo total que este reacomodamiento de carteles ocasionará, haciéndolo del mejor modo posible.

Descripción de la función

Debes implementar la función `vial(K,C,tipo, cuantos, pos, queHay)`.

Sus parámetros son:

- `K, C`: Enteros que corresponden al costo de enterrar o remover cada cartel, y al costo por metro de los movimientos del vehículo, respectivamente.
- `tipo, cuantos`: Indican que la cuadrilla inicia el recorrido con `cuantos[i]` carteles del tipo `tipo[i]`, para $0 \leq i < T$. `tipo` es una cadena de longitud **T**, y `cuantos` es un arreglo de **T** enteros.
- `pos, queHay`: Indican que el par `pos[i], queHay[i]` forma un ítem del listado elaborado por la inspectora. Si `queHay[i]` es una letra **en mayúscula**, significa que en la ubicación `pos[i]` **debe haber un cartel** del tipo `queHay[i]`. Si en cambio `queHay[i]` es una letra **en minúscula**, significa que en esa ubicación **hay un cartel** de ese tipo. `queHay` es una cadena de longitud **2N**, y `pos` es un arreglo de **2N** enteros.

La función debe retornar el costo de la re-señalización de la ruta, eligiendo una política de reacomodamiento que lo minimice.

Evaluador

El evaluador local lee de la entrada estándar con el siguiente formato:

- Enteros **K, C, T, N**
- **T** líneas, una para cada tipo de dificultad, que contienen el tipo `tipo[i]` y la cantidad de carteles extra `cuantos[i]` con la que inicia su recorrido.
- **2N** líneas que contienen la descripción de la ruta que elaboró la inspectora. Cada línea con un entero `pos[i]` y una letra `queHay[i]`.

Escribe en la salida estándar el valor retornado por la función.

Restricciones

- $1 \leq N \leq 200.000$
- $1 \leq K \leq 1.000$
- $1 \leq C \leq 500$
- $1 \leq T \leq 26$
- $0 \leq \text{cuantos}[i] \leq 100$
- $0 < \text{pos}[i] < 1.000.000$
- `tipo` contiene solamente letras mayúsculas, sin repetidos.
- `queHay` contiene exactamente **N** letras mayúsculas y **N** minúsculas.
- `queHay` solo contiene letras que aparezcan en `tipo`.
- Por cada letra de `tipo`, `queHay` contiene la misma cantidad de apariciones en mayúscula y en minúscula de esa letra.
- No habrá inicialmente dos carteles en la misma posición.
- No se deberá colocar dos carteles en la misma posición.

Ejemplo

Si la entrada fuera:

```

340 40 8 7
C 0
L 0
G 2
A 0
H 1
J 1
P 4
Z 0
57606 A
10900 a
11650 A
50254 C
27592 G
27713 G
28306 g
50254 c
52010 g
58000 a
80000 J
80000 c
80001 j
80001 C

```

La salida correcta será:

```

40035600

```

Que corresponde al costo total de la siguiente maniobra óptima:

1. Desplazarse de **0** a **10900**
2. Remover un cartel A
3. Desplazarse de **10900** a **11650**
4. Colocar un cartel A
5. Desplazarse de **11650** a **27592**
6. Colocar un cartel G
7. Desplazarse de **27592** a **27713**
8. Colocar un cartel G
9. Desplazarse de **27713** a **28306**
10. Remover un cartel G
11. Desplazarse de **28306** a **52010**
12. Remover un cartel G
13. Desplazarse de **52010** a **58000**
14. Remover un cartel A
15. Desplazarse de **58000** a **57606**
16. Colocar un cartel A
17. Desplazarse de **57606** a **80000**
18. Colocar un cartel J
19. Remover un cartel C
20. Desplazarse de **80000** a **80001**
21. Colocar un cartel C
22. Remover un cartel J
23. Desplazarse de **80001** a **1000000**

Subtareas

1. **cuantos[i] = N** para todo **i** (7 puntos)
2. **T = 1**, **tipo[0] == 'A'** (22 puntos)
3. **N ≤ 1.000** (20 puntos)
4. **pos[i+1] ≥ pos[i]** para todo **i** (21 puntos)
5. Sin más restricción (30 puntos)