

Cartilla de Problemas

UNSA-ACM-ICPC Competencia de Programación

Fecha de la Competencia: Noviembre 2024

Problema A

Estatura

Carlitos es un entusiasta de las aventuras con un amor insaciable por los parques de diversiones. A pesar de su pasión vibrante, Carlitos enfrenta un desafío único: su estatura limitada. Mientras planea ansiosamente su aventura de fin de semana, se da cuenta de que su altura podría afectar su experiencia en el parque de diversiones. No se trata solo de elegir un parque; se trata de encontrar uno en el que pueda disfrutar de la emoción de los juegos.

Imagina el caleidoscopio de colores, las risas alegres y la adrenalina de las atracciones. Carlitos siempre se ha sentido atraído por la energía de los parques de diversiones. Con el fin de semana acercándose, estudia los folletos del parque, revisando los requisitos de altura de cada juego. Su objetivo es maximizar su diversión, y aquí es donde entras en escena.

Tu tarea es ayudar a Carlitos a determinar el número de atracciones que puede disfrutar en un parque específico. Considerando su altura y los requisitos mínimos de altura de cada juego, oriéntalo para aprovechar al máximo su aventura en el parque de diversiones.

Entrada

La primera línea contendrá el número de casos de prueba, las siguientes líneas por cada caso de prueba primero habrán dos números enteros, N y H ($1 \leq N \leq 6$ y $90 \leq H \leq 200$), que representan la cantidad de juegos en un parque y la altura de Carlitos en centímetros, respectivamente. En la segunda línea de la entrada, se darán las alturas mínimas A_1, \dots, A_N ($90 \leq A_i \leq 200$) de cada uno de los juegos del parque.

Salida

Tu programa debe imprimir una sola línea que contenga la cantidad de juegos en los que Carlitos puede subirse, es decir, la cantidad de juegos para los que su altura es al menos tan grande como la altura mínima requerida.

Ejemplo de Entrada

```
2
1 100
100
6 120
200 90 100 123 120 169
```

Ejemplo de Salida

```
1
3
```

Problema B

Vacaciones Agotadoras

William está planificando sus próximas vacaciones. Un tema recurrente en todas ellas es la necesidad de lidiar con el cansancio. Algunos días ni siquiera logra disfrutar mucho, ya que después de varias actividades el cansancio se vuelve mayor de lo que puede soportar.

Esta vez, William tuvo una idea. Va a estimar el impacto de cada una de las actividades turísticas en su nivel de energía. Ha notado que algunas actividades, como deportes y caminatas, son agotadoras y consumen su energía, mientras que otras actividades, como obras de teatro y musicales, son revitalizantes y le ayudan a recuperar energía.

Más precisamente, William empieza con D unidades de energía y divide sus actividades en dos grupos: C actividades agotadoras y R actividades revitalizantes. Cada actividad agotadora requiere una cierta cantidad de energía y, por lo tanto, consume esa cantidad al realizarse. Cada actividad revitalizante le otorga una cantidad de energía cuando la realiza. Además, organiza las actividades en cada grupo según sus preferencias, ya que hay actividades que está más ansioso por realizar. Es importante notar que las actividades de ambos conjuntos pueden alternarse, pero William siempre realizará las actividades en orden de preferencia dentro de cada grupo, sin saltarse ninguna.

A lo largo de sus vacaciones, al decidir qué actividad realizar a continuación, elegirá la primera actividad agotadora aún no realizada, si tiene suficiente energía para llevarla a cabo. De lo contrario, realizará la próxima actividad revitalizante aún no realizada, si queda alguna, recuperando así algo de energía. Naturalmente, si en algún momento no quedan actividades agotadoras, simplemente podrá realizar todas las actividades revitalizantes restantes.

Ahora, considerando este proceso, él te ha pedido ayuda para determinar cuántas actividades (incluyendo agotadoras y revitalizantes) podrá realizar.

Entrada

Habrán varios casos de prueba, la primera línea de cada caso de prueba tendrá 3 enteros D , C y R , que indican respectivamente la cantidad de energía inicial, el número de actividades agotadoras, y el número de actividades revitalizantes ($1 \leq D \leq 10^5$, $1 \leq C \leq 10^4$ y $1 \leq R \leq 10^4$). Cada una de las siguientes C líneas contiene un entero C_i ($1 \leq C_i \leq 10^5$ para $1 \leq i \leq C$), indicando el consumo de energía requerido para una actividad agotadora, en orden de preferencia. Finalmente, cada una de las R líneas siguientes contiene un entero R_i ($1 \leq R_i \leq 10^5$ para $1 \leq i \leq R$), indicando la cantidad de energía que recupera una actividad revitalizante, también en orden de preferencia; todo esto para cada caso de prueba.

Salida

Imprime una única línea que contenga un solo entero, el número total de actividades (incluyendo agotadoras y revitalizantes) que William podrá realizar.

Ejemplo de Entrada

Handwritten input example:

```

40 3 3
30 10
20 10
10 100
5 5
5 5
5 5
40 2 2
60 120
80 120
5
10
100 3 1
60
50
10
  
```

Handwritten output example:

```

5
2
2
  
```

Handwritten calculations:

$$100 - 10 = 90 \quad (1)$$

$$90 - 60 = 30 \quad (2)$$

60
60
50
10

Ejemplo de Salida

5
2
2

Problema C

Lexicográficamente Agradable

En la pintoresca aldea de Lexiconia, vivían dos amigos, Lily y Ethan. Un día, una misteriosa carta llegó a la casa de los amigos, sellada con una encantadora insignia. Dentro del sobre, encontraron un acertijo demasiado complejo incluso para las mentes más sabias de la aldea.

En este acertijo, se da un número entero K y una cadena S que contiene solo letras minúsculas, la cual puede ser alterada según una regla curiosa. En cada momento, los aldeanos tienen la libertad de elegir un índice i y, mágicamente, los caracteres S_i y S_{i+K} intercambian sus posiciones. El acertijo se resuelve cuando se encuentra la cadena lexicográficamente mínima, usando solo operaciones que respeten la regla impuesta por el acertijo.

La aldea quedó muy curiosa y entusiasmada con el acertijo. Lily y Ethan, siempre hambrientos de aventura, decidieron lanzarse de lleno a este desafío. Sin embargo, a medida que observaban la cadena, notaron que había innumerables intercambios posibles.

Con la cadena del acertijo presente en sus mentes, los amigos se preguntaron: ¿cómo navegar en este inmenso mar de posibilidades y determinar la cadena lexicográficamente mínima? Cada intercambio realizado parecía como hojear un libro mágico, que revelaba nuevos secretos y misterios.

El tiempo pasó y hasta ahora Lily y Ethan no han terminado de resolver este acertijo. ¿Podrás ayudarles?

Entrada

Habrán varios casos de prueba y la primera línea de cada caso contendrá la cadena S ($1 \leq |S| \leq 10^5$). La segunda línea de la entrada contendrá el entero K ($1 \leq K < |S|$).

Salida

Por cada caso de prueba, imprimir una única línea que contenga la cadena lexicográficamente mínima posible usando estas operaciones.

Ejemplo de Entrada

zaaab

4

njoab

2

Ejemplo de Salida

baaaz

banjo

zaaab

0 1 2 3 4

baaaz

njoab

njoab

0 1 2 3 4

o j n a b

Solved

Problema E

Malla Aérea

En el reino de Cuadradonia, el monarca desea revisar todas las tarifas aéreas. Para ello, solicitó a su contador una tabla con las propuestas de nuevos precios.

Sin embargo, el monarca estudió en el Instituto de Computación y Programación de Chapecó (ICPC) y tiene suficiente conocimiento para exigir coherencia en la tabla. La tabla es coherente si ninguna ruta con escalas es más barata que el vuelo directo.

Verificada la coherencia de la tabla, al monarca le gustaría reducir el número de vuelos directos, sin aumentar los costos de los viajes.

El problema es verificar la coherencia de la tabla y, si esta es coherente, informar al monarca cuántos vuelos directos pueden ser eliminados sin incrementar el costo de ningún viaje.

Entrada

Habrán varios casos de prueba y para cada uno de ellos:

La primera línea contiene N ($1 \leq N \leq 100$), que es el número de ciudades en Cuadradonia servidas por vuelos. Existen entonces N líneas adicionales, L_1, L_2, \dots, L_N . La línea L_i contiene N enteros, $C_{i1}, C_{i2}, \dots, C_{iN}$, donde C_{ij} es el costo del vuelo directo entre las ciudades i y j .

El costo de ida y vuelta entre dos ciudades es siempre igual, es decir, $C_{ij} = C_{ji}$, para todos los pares $\{i, j\}$ tales que $1 \leq i \leq N$ y $1 \leq j \leq N$. Cuando $i = j$, $C_{ij} = 0$. Cuando $i \neq j$, $1 \leq C_{ij} \leq 10^3$.

Salida

Para cada caso de prueba, imprimir una única línea que contenga un entero. Si la tabla es incoherente, el entero debe ser igual a -1. Si la tabla es coherente, el entero debe ser igual al mayor número de vuelos directos que pueden ser eliminados sin incrementar los costos de los viajes para los pasajeros.

Ejemplo de Entrada

```

3
0 1 2
1 0 1
2 1 0
3
0 2 2
2 0 2
2 2 0
3
0 2 9
2 0 2
9 2 0

```

Ejemplo de Salida

```

1
0
-1

```