



Problem A. Los Substrings Telefónicos

Source file name: Telefonicos.c, Telefonicos.cpp, Telefonicos.java, Telefonicos.py
Input: Standard
Output: Standard

Hay una compañía multinacional que tiene sede en más de 20 países, cada país tiene un código de área (de distinta longitud), dentro de cada país se tiene un código local (también de diferente longitud), dentro de cada área, se tiene edificios (también con códigos de diferente longitud) y cada edificio tiene oficinas con su código local (adivinen, también con diferente longitud) que su extensión telefónica directa.

Se cuenta con el directorio telefónico de la multinacional completa, pero se mezclaron los números de teléfono de país, edificios, extensiones a oficinas, etc. Se quiere saber cuántas oficinas hay en total. Su tarea consiste en indicar, dados todos los números de teléfono, cuántos de ellos ya tienen un prefijo (código de país, código de área) común a otros.

Input

El primer número T ($1 \leq T \leq 20000$) que debe leer es el número de teléfonos que hay en la guía. Las siguientes T líneas son números de a lo sumo 50 dígitos indicando un número de teléfono.

Output

Se debe imprimir una línea con un número entero con la cantidad de prefijos hallados en los números de teléfono indicados.

Example

| Input | Output |
|-----------------------------|--------|
| 4 1 1234 12 123 | 6 |

Explanation

El 1 es prefijo de 3 números, el 12 es prefijo de 2 números, el 123 es prefijo de 1 número.

Problem B. Los Edificios de Apartamentos

Source file name: Edificios.c, Edificios.cpp, Edificios.java, Edificios.py
Input: Standard
Output: Standard

En Nueva York, los apartamentos son la forma estándar para vivir en la ciudad, sin embargo, con la pandemia, se han generado muchos movimientos de persona que salen de los edificios para vivir en otras partes, lo que ha generado una serie de problemas para la empresa encargada de distribuir el gas a través de los ductos destinados para ello. Como medida de prevención se necesita que la cantidad de abastecimiento no exceda el que se puede consumir en promedio por persona y así, de haber una fuga, que el posible daño sea minimizado.



Ciudad de Nueva York

Para ello se han colocado en N edificios numerados del 1 a N , en ellos se indica la cantidad de personas que habitan por edificio, pero constantemente hay personas que entran (suman) o salen (restan) de cada edificio.

La empresa necesita saber además, cuál es la cantidad de personas que habitan del edificio i al edificio j en un momento dado para poder regular el gas y la presión que deben tener en diferentes puntos de la estación encargada de distribuir el valioso mineral.

Input

La primera línea contiene dos números separados por espacio: N $1 \leq N \leq 10^6$, indicando el número de edificios y Q $1 \leq Q \leq 10^5$, indicando el número de consultas o actualizaciones.

La siguiente línea contiene la secuencia E con N números enteros no negativos, indicando la cantidad de personas que viven en cada edificio. Siendo E_i la cantidad de personas que viven en el edificio i .

Las siguientes Q líneas son consultas o actualizaciones sobre el estado de los edificios.

Si se trata de una consulta la línea estará precedida por la letra q , siguiendo el formato: $q \ i \ j$ donde $1 \leq i \leq j \leq N$. Siendo i y j el intervalo de personas que habitan entre el edificio i y el j , incluidos y .

Si se tratara de una actualización, la línea estará precedida por la letra u , con el formato: $u \ i \ di$. Donde i es el número de edificio que se está modificando, y di ($e_i \leq |di| \wedge di \leq 10^5$) es la diferencia con que se está actualizando el i -ésimo edificio.

Output

Por cada consulta, debe imprimirse una línea, cada línea debe poseer el número de personas que habitan en el intervalo consultado.



Example

| Input | Output |
|-------|--------|
| 3 4 | 14 |
| 1 9 4 | 15 |
| q 1 3 | 4 |
| u 2 1 | |
| q 1 3 | |
| q 3 3 | |

Problem C. El HPF de Salud

Source file name: Salud.c, Salud.cpp, Salud.java, Salud.py
Input: Standard
Output: Standard

Debido a la pandemia actual los centros de vacunación tienen que atender a una población enorme, por lo que ha catalogado a las personas para poder atender a las personas con mayor riesgo primero. Pero algunas personas se han equivocado en cuándo debían de vacunarse, han fallado en identificar su verdadera prioridad y han atendido tarde a la cita de vacunación.

Sin embargo, la directriz de salud es clara, cualquier persona que se haya inscrito para vacunación, debe ser llamada acorde con su prioridad. Sólo se le debe aplicar una vacuna a cada persona y es posible que hayan varias personas con el mismo nombre.

Para ello debe hacer un programa que responda de la manera más eficiente posible sobre quién es el siguiente en ser llamado a vacunar.

Input

El caso de prueba consiste en a lo sumo 10^4 líneas, se debe leer hasta fin de archivo.

Los ingresos consiste en dos tokens, siendo el primero el nombre de la persona (de a lo sumo 20 chars), seguido de un espacio y luego un número que es su prioridad (cuánto más alto, más alta su prioridad). Los nombres son sólo letras [a-z] del alfabeto en inglés y en minúscula.

Las llamadas son con una línea que tiene la letra V (de vacuna).

Output

Por cada V, debe de mostrarse en una línea el nombre de la persona que debe ser vacunada, si hay vacunas, pero no hay personas en el registro, se debe dejar una línea en blanco.

Si quedaran personas sin vacunar, no se deben mostrar

Example

| Input | Output |
|--|----------------------------|
| lanudo 1 house 100 emilia 20 V V bragui 30 V | house emilia bragui |
| hugo 100 rpc 90 V V V niquefa 80 V | hugo rpc niquefa |

Problem D. Cantidad de Lux

Source file name: Lux.c, Lux.cpp, Lux.java, Lux.py
Input: Standard
Output: Standard

En términos simples, los lúmenes (lm) son una medida de la cantidad total de luz visible (a simple vista) de una lámpara o fuente de luz. Cuanto más alto el número de lúmenes la bombilla o lámpara es más brillante.

El lux es una unidad derivada, basada en el lumen, que a su vez es una unidad derivada basada en la candela.

Un lux equivale a un lumen por metro cuadrado, mientras que un lumen equivale a una candela x estereorradián. El flujo luminoso total de una fuente de una candela equivale a 4π lúmenes (puesto que una esfera comprende 4π estereorradianes).

Esta unidad es importante para poder valorar cuánta luz emite una estrella. o un conjunto de estrellas. Las alteraciones de luz, pueden indicar que una estrella nace (hace millones de años por el tiempo que tarda en llegar la luz hasta la Tierra o por el contrario, que una estrella muere, que hay varias formas en que puede morir, ya sea porque se ensancha y “explota” o porque es tan masivo que se convierte en un agujero negro. Esto nos lleva a la necesidad de poder identificar tanto el máximo nivel de luz como el mínimo en un intervalo del espacio.

Un telescopio dado, ha alineado las estrellas en una lista, a cada estrella le asigna la cantidad de lux que logra captar. Esta cantidad es modificada cada vez que se percibe un cambio en una estrella.

Con el fin de medir la cantidad de estrellas que mueren en un sector de la galaxia, un astrónomo, necesita saber cuál es el mínimo de lux y el máximo de lux que hay en un intervalo de estrellas, desde la posición i hasta la posición j (incluidos).

Para ello se le ha solicitado a usted el que pueda escribir un programa que le permita a nuestro amigo, responder eficientemente su consulta.

Input

La entrada consiste en un único caso de prueba.

La primera línea contiene dos números separados por espacio: N $1 \leq N \leq 2 \times 10^5$, indicando el número de estrellas visibles desde el telescopio y Q $1 \leq Q \leq 10^5$, indicando el número de consultas o actualizaciones.

La siguiente línea contiene la secuencia E con N números enteros no negativos, indicando la cantidad de lux que es recibida por cada estrella. Siendo E_i la cantidad de lux que se percibe de la estrella i .

Las siguientes Q líneas son consultas o actualizaciones sobre el estado de los edificios.

Si se trata de una consulta la línea estará precedida por la palabra MAX o MIN, siguiendo el formato: MAX i j ó MIN i j donde $1 \leq i \leq j \leq N$. Siendo i y j el intervalo de estrellas al que se le quiere consultar el máximo o mínimo.

Si se tratara de una actualización, la línea estará precedida por la palabra UPD, con el formato: UPD i ni . Donde i es el número de estrella que se está percibiendo distinto, y ni ($0 \leq ni \leq 10^5$) es la nueva cantidad de lux percibida por la i -ésima estrella.

Output

Por cada consulta de la entrada, se imprime en una línea independiente, el valor máximo o mínimo según corresponda del intervalo solicitado. (Con tanto el i como el j incluidos).

Example

| Input | Output |
|-------------|--------|
| 4 4 | 300 |
| 9 12 300 21 | 0 |
| MAX 2 3 | 21 |
| UPD 3 0 | |
| MIN 3 4 | |
| MAX 1 4 | |



Problem E. Los Amigos de mis Amigos

Source file name: Amigos.c, Amigos.cpp, Amigos.java, Amigos.py
Input: Standard
Output: Standard

En las universidades estadounidenses, se tiene como costumbre hacer fraternidades. Parte de la fraternidad consiste en que si dos personas pertenecen a la misma fraternidad, entonces se consideran “hermanos”.

Hay algunas personas, aún así, que no se unen a ninguna fraternidad pues tienen el extraño objetivo de ir a la universidad a estudiar (siempre hay alguien que no quiere encajar con verdadero propósito universitario ¿no?).

En la Universidad UGLI, hubo un problema con el registro de las fraternidades, pero todos los hermanos, recordaban al menos a un hermano de su fraternidad. Suponiendo que todas las personas de la UGLI son numeradas de 1 a N y se tienen todos los pares de personas que recordaban su hermandad. ¿Puede indicar cuántas fraternidades y cuántas personas no pertenecen a ninguna?

Input

La primer línea consiste en dos números enteros separados por espacio N ($1 \leq N \leq 10^4$) y H ($1 \leq H \leq 2 \times 10^4$) indicando el número de personas en UGLI y la cantidad de conocidos que recuerdan su hermandad respectivamente.

Las siguientes H líneas contienen dos números i j separados por espacio, indicando que i recuerda su hermandad con j .

Output

La salida deben ser dos enteros en una línea separados por espacio, el primero es el número de fraternidades y el segundo el número de personas que no pertenecían a ninguna fraternidad.

Example

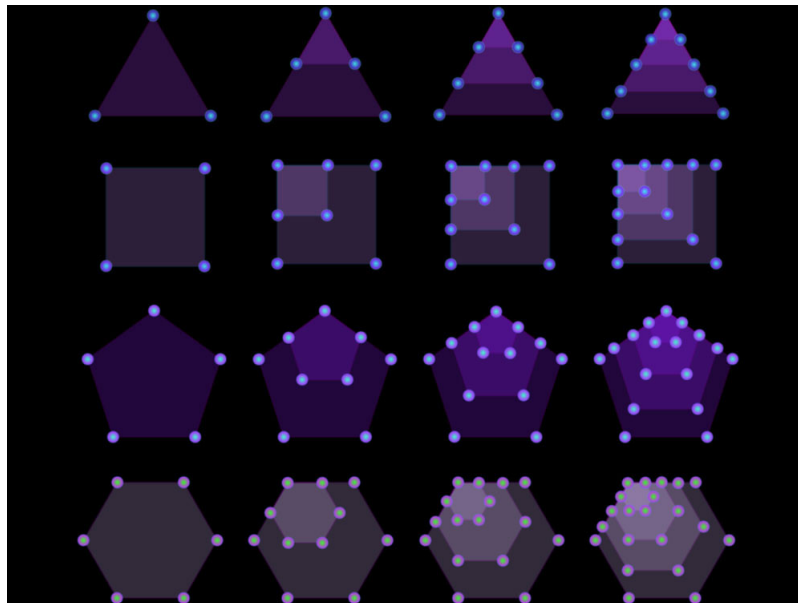
| Input | Output |
|-------------------|--------|
| 5 2 1 3 3 4 | 1 2 |

Problem F. Las Caras del Polígono

Source file name: Poligono.c, Poligono.cpp, Poligono.java, Poligono.py
Input: Standard
Output: Standard

Considere una figura poligonal¹ donde todas las líneas que lo conforman son siempre horizontales o verticales. Dada una figura, su tarea consiste en indicar el número de lados que posee la figura.

Deben de considerarse incluso aquellas caras que limitan con el borde de la figura.



Polígonos

Input

La primer línea consiste en dos números enteros separados por espacio L ($1 \leq N \leq 200$) y M ($1 \leq H \leq 200$) indicando el largo y ancho de la figura.

Las siguientes L líneas contienen N caracteres cada una, formando la figura, “.” para espacio vacío y “#” para indicar que forma parte de la figura. La figura siempre va a ser conexa.

Output

Una línea con un entero que corresponde con el número de lados que posee la figura (Considerar que los bordes de la figura también cuentan como uno de sus lados).

Example

| Input | Output |
|--|-----------------|
| <pre> 5 5###. ..#.. .###. </pre> | <pre> 12 </pre> |

¹Imagen tomada de: <https://www.flickr.com/photos/arquepoetica/2781771003>

Problem G. La Persecución de Link

Source file name: Persecucion.c, Persecucion.cpp, Persecucion.java, Persecucion.py
Input: Standard
Output: Standard

La calamidad Ganon ha vuelto a desatarse y ha sido contenida gracias a la magia de Zelda. Zelda esta vez ha logrado poner a Ganon en un baúl mágico, entretanto llega Link con la espada que vence la oscuridad y vuelve a sumir a Ganon en el submundo.

Pero ¡Oh calamidad!. Resulta que un Bokolin logró infiltrarse en el palacio y se ha robado el baúl. Link ha llegado justo a tiempo para darle persecución. Pero el Bokolin ha entrado al templo de la teletransportación y se ha ido a una ciudad custodiada por un centaleón.

Se sabe que el bokolin es muy supersticioso y sólo avanza a lugares por medio de caminos preestablecidos y además utiliza sólo aquellos caminos que llevan a un lugar con un centaleón, buscando que alguno de ellos pueda detener al intrépido Link. La mecánica de la situación consiste en que primero el bokolin llega a un lugar y huye hasta que Link aparece. (es decir, si Link no aparece, el bokolin no se mueve). Pero Link siempre se mueve hacia el bokolin.

Siempre que Link se enfrenta a un centaleón, gana y lo mata. Esto hace que ese lugar y todos los caminos que dan a ese lugar, sean inutilizables para el bokolin con el baúl.

Dado el mapa de Hyrule, ¿Cuál es el máximo número de centaleones que Link debe matar antes de que el Bokolin se quede sin lugares a dónde huir y por lo tanto Link podría enviar a Ganon de vuelta al submundo?.

Todos los lugares de Hyrule están numerados de 1 a n .

Input

La primer línea posee dos números, N el número de lugares del Reino de Hyrule ($1 \leq N \leq 16$) y C el número de caminos que hay en el Reino de Hyrule ($1 \leq C \leq 5 \times N$).

Output

Una línea con un único entero, correspondiente con el máximo número de centaleones que debe vencer Link antes de acorralar por completo al Bokolin.

Example

| Input | Output |
|--------------------------|--------|
| 4 3 1 2 2 3 3 4 | 4 |

Problem H. El Tráfico de Red

Source file name: Trafico.c, Trafico.cpp, Trafico.java, Trafico.py
Input: Standard
Output: Standard

Se cuenta con $N + 1$ computadoras en una red. Todas las computadoras se encuentran numeradas de 0 a N .

En la red es necesario enviar mensajes desde una computadora hacia todas las demás. Es posible que para ello se necesite que un mensaje pase por una o varias computadoras intermedias, si la fuente y el destino de un mensaje no se encuentran conectados. Al enviar un mensaje de una computadora a otra, se requiere que pase por el mínimo número de computadoras intermedias posible.

Dado el número de computadoras, la forma en cómo están conectadas y una computadora inicial, ¿Es posible indicar el número mínimo de computadoras por las que se debe enviar un mensaje?

Su tarea consiste en que dada la computadora inicial, indicar el número mínimo de computadoras por las que debe pasar un mensaje para todas las demás computadoras. (K no debe mostrarse en la solución).

Input

La primer línea consiste en tres números enteros positivos separados por espacio N ($1 \leq N \leq 10^5$) la cantidad de computadoras y C ($1 \leq C \leq 10^6$) indicando la cantidad de conexiones que tienen las computadoras y K siendo la computadora inicial.

Las siguientes C líneas tienen dos enteros i, j ($0 \leq i, j < N$), indicando que la computadora i y la computadora j están conectadas.

Output

En cada línea, se debe mostrar las computadoras ordenadas de menor a mayor. Cada línea debe iniciar con por el número de computadoras que se necesitan para llegar a ellas desde la computadora inicial (salvo la primera línea que indica aquellas a las que no se pueden llegar)

Para todas las computadoras que es imposible tener acceso, se deben de mostrar precedido por un 0:, también ordenadas de menor a mayor.

A partir de este punto, deben colocarse el número de computadoras por visitar para poder enviar un mensaje, iniciando con 1: para indicar las que están directamente conectadas, 2: para indicar las que necesitan de una computadora intermedia, etc.

La última línea sería aquella donde están las computadoras que requieren el máximo número de computadoras intermedias.

Example

| Input | Output |
|-------|--------|
| 5 3 1 | 0: 0 |
| 1 2 | 1: 2 3 |
| 1 3 | 2: 4 |
| 2 4 | |



Problem I. Hileras binarias

Source file name: Hileras.c, Hileras.cpp, Hileras.java, Hileras.py
Input: Standard
Output: Standard

El profesor Moralov tiene una fuerte obsesión con los números primos, aquellos naturales mayores que 1 y que no tienen divisores propios mayores que la unidad. Dado que es un profesor de computación e informática, está acostumbrado a trabajar con números binarios. Pero está cansado de encontrar la cantidad de primos en un intervalo, por lo que ha ideado un nuevo reto. Para ello ha ideado un nuevo problema, si tiene una hilera de largo arbitrario k , cuántas hileras tienen una cantidad prima de 1's y cantidad prima de 0's.

Por ejemplo, si se tiene una hilera de longitud 10, se quieren las hileras con 7 1's y 3 0's (y viceversa) o 5 0's y 5 1's.

Para hacer el problema más interesante, el profesor Moralov desea saber cuántas hileras binarias de largo entre X y Y (incluidos) existen las cuales tienen una cantidad prima de 0's y una cantidad prima de 1's.

Dado que este número puede ser muy grande, el profesor Moralov ha decidido tener el valor módulo 1000000007.

Input

En una primera línea viene un entero Q indicando el número de consultas, con $1 \leq Q \leq 10^5$. Las siguientes Q líneas tienen dos números enteros separados por espacio X, Y , con $1 \leq X \leq Y \leq 2000$.

Output

Por cada consulta, se debe imprimir una línea con el número (módulo 1000000007) de hileras con cantidad prima de 1's y 0's.

Example

| Input | Output |
|-------|--------|
| 3 | 0 |
| 2 2 | 6 |
| 4 4 | 492 |
| 10 11 | |

Problem J. El Problema del Parqueo

Source file name: Parqueo.c, Parqueo.cpp, Parqueo.java, Parqueo.py
Input: Standard
Output: Standard

En Japón se tienen puestos de parqueo que buscan aprovechar el espacio al máximo. Para ello se tienen espacios de parqueos dispuestos en cubículos de 7 carros, con una única entrada.

Cada uno de ellos, a lo interno se organiza tal como se muestra la siguiente figura:

| | | |
|----|----|----|
| | 31 | 43 |
| 15 | 35 | 31 |
| 64 | 43 | |

Dejando siempre la esquina inferior derecha vacía, así como la superior izquierda.

En ese tablero, se asume que cada número corresponde con el peso del vehículo.

El tema es que el orden en que son puestos por lo general, difiere del orden en que son sacados. Para ello se requiere ordenar diferente esa posición inicial, el problema es que el robot encargado de hacer esos cambios necesita energía para poder intercambiarse de lugar y esa energía es correspondiente con la suma de los pesos de los vehículos que está moviendo y por razones de amplitud, sólo se pueden intercambiar dos vehículos que estén contiguos (ya sea arriba-abajo o izquierda-derecha). Además, está garantizado que no hay dos vehículos con el mismo peso.

Dada la posición inicial (forma en como llegaron) y la posición final (forma en como deben salir), ¿podría usted indicar cuál es el mínimo gasto de energía del robot?

Input

Se trata de un sólo caso de prueba, las primeras 3 líneas indican la posición inicial, cada línea compuesta por enteros positivos (2-3-2 enteros respectivamente) separados por espacio y todos menores que 10^4 .

Las siguientes 3 líneas indican la posición final, cada línea compuesta por los mismos enteros que en las tres líneas anteriores, sólo que en una configuración diferente. En todos los casos, separados por espacio y en el mismo formato 2-3-2.

Output

La salida una línea que tiene un número indicando el mínimo costo de energía que necesitaría el robot para reacomodar los vehículos.

Example

| Input | Output |
|--|--------|
| 31 43 15 35 31 64 43 43 31 15 35 31 64 43 | 74 |



Problem K. Harry Potter y el Mensaje del Ministerio

Source file name: Mensaje.c, Mensaje.cpp, Mensaje.java, Mensaje.py
Input: Standard
Output: Standard

El Ministerio de Magia ha instalado un nuevo servicio de mensajería basado en aviones de papel encantado. El aurore Harry Potter ha podido hacer un hechizo para que los papeles se copien una vez que llegan a un destino, de modo que puede dejar una copia en un lugar y seguir su vuelo hacia otro destino. Esta copia puede hacerse tantas veces como se requiera.

Sin embargo, el hechizo requiere de savia de mandrágora, que como sabemos es muy costosa. Requiere tanta savia como tiempo tarda en trasladarse un avión de papel de un punto a otro (independientemente del número de copias).

Por otra parte, el Ministerio de Magia es como un laberinto, lleno de pasadizos y escondrijos por donde pueden ser enviados los mensajes. Nuestro aurore favorito desea poder enviar mensajes a todos los demás aurores, minimizando la cantidad de savia de mandrágora requerida, para ello debe encontrar el mínimo tiempo en que puede enviar un mensaje a todos los aurores.

Sabiendo que todos los aurores están numerados de 1 a n , y señalando los tiempos que hay de traslado entre los conductos (que van siempre en dos vías) de algunos de ellos (nótese que si A_1 está a 12 segundos de A_2 y A_2 está a 15 segundos de A_3 y no hubiera un conducto directo entre A_1 y A_3 , se puede asumir que un avión pasa por A_1 , dejando una copia, luego deja una copia en A_2 y finalmente sigue su vuelo hasta llegar a A_3 con un costo total de 27 segundos). ¿Puede ayudar a Harry Potter a determinar el tiempo mínimo en que todos los aurores son avisados del mensaje?

Input

Se trata de un sólo caso de prueba, la primer línea consiste en A , ($1 \leq N \leq 10^5$) indicando el número de aurores que hay en el ministerio y C , ($N - 1 \leq C \leq 5 \times N$) indicando el número de conductos que hay en el ministerio. Las siguientes C líneas consta cada una de 3 números enteros positivos separados por espacio: $i \ j \ t$, indicando que del *auror* _{i} al *auror* _{j} se requiere t segundos. Donde $i, j < A$ y $t < 10^4$

Output

La salida es una línea con un número entero, indicando el mínimo tiempo para que todos los aurores reciban su mensaje. (que es la suma de los tiempos de todos los aviones de papel)

Example

| Input | Output |
|---|--------|
| 5 7 1 2 1 1 3 8 2 3 1 3 4 2 3 5 10 4 5 2 2 5 8 | 6 |

Problem L. Un Metro Exhaustivo

Source file name: Metro.c, Metro.cpp, Metro.java, Metro.py
Input: Standard
Output: Standard

En la ciudad de Nueva York, se tiene un sistema de metro que cubre por completo la ciudad, cruza desde el llamado “uptown” hasta “downtown”, pasando desde luego por “midtown”. Pero aún más, se puede viajar casi de cualquier punto de la ciudad a cualquier punto de la ciudad. Se cuenta con más de 110 puntos del metro, sólo en la ciudad, sin tomar en cuenta las salidas de tren a otros estados.

Dejando de lado el horario de los trenes, se tienen los tiempos que tardan entre dos estaciones de la ruta del tren. Además, algunas estaciones se cruzan con otras, por lo que se pueden transbordar y dirigirse a otro lugar de la ciudad. Todas las estaciones están numeradas de 1 a n . Además, existe alguna manera de llegar desde cualquier estación i a cualquier estación j .

Considerando el mapa de metro de NYC con los tiempos entre diferentes estaciones, una serie de consultas entre las estaciones I y J , indicar el mínimo tiempo que se requiere para viajar desde I hasta J , asumiendo que siempre se está disponible un tren en cualquier estación a la que llegamos.

Input

La entrada es un único caso de prueba. La primera línea contiene tres números N ($1 \leq N \leq 110$) el número de estaciones de tren, L ($1 \leq L \leq 10^3$) el número de líneas de ferrocarril entre estaciones de tren y Q ($1 \leq Q \leq 10^3$) que serían las consultas sobre el tiempo que se necesita para ir de una estación a otra.

Las siguientes L líneas tienen tres enteros positivos, i , j y t , indicando que de la estación i hay una línea a la estación j y que el tren tarda t minutos en llegar de un punto a otro. En todos los casos, la relación es bidireccional. ($1 \leq i, j \leq N$) y $t < 30$

Las siguientes Q líneas tienen dos enteros, que son las consultas, desde el punto i al punto j . ($1 \leq i, j \leq N$).

Output

Por cada consulta, debe de imprimirse en una línea independiente el tiempo mínimo que se tardaría desde la estación i hasta la estación j .

Example

| Input | Output |
|--------|--------|
| 6 8 6 | 8 |
| 1 2 5 | 9 |
| 1 3 4 | 6 |
| 1 4 6 | 2 |
| 2 5 9 | 8 |
| 2 6 1 | 4 |
| 4 6 1 | |
| 5 4 2 | |
| 5 1 10 | |
| 1 5 | |
| 2 3 | |
| 1 6 | |
| 4 2 | |
| 3 3 | |
| 2 5 | |



Problem M. La Matriz de Múltiplos

Source file name: Multiplos.c, Multiplos.cpp, Multiplos.java, Multiplos.py

Input: Standard

Output: Standard

En una matriz A de tamaño $n \times m$ se colocan números naturales de forma que a_{ij} es el j -ésimo múltiplo de i . Un ejemplo de una matriz con $n = 2$ y $m = 3$ se puede apreciar en:

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 4 & 6 \end{pmatrix}$$

Dos estudiantes universitarios de la carrera de ingeniería compiten para poder indicar cuál es la suma de todos los números de la matriz, dado el largo (n) y el ancho (m).

Si el número resultara muy grande, se debe de indicar el valor módulo $10^9 + 7$

Input

La primer línea será un número T ($1 \leq T \leq 10^5$) indicando el número de casos de prueba. Las siguientes T líneas contienen dos números n y m indicando respectivamente el número de filas y columnas de la matriz de múltiplos. ($1 \leq n, m \leq 10^6$).

Output

Por cada caso de prueba se debe imprimir una línea con el resultado de la suma de la matriz de $n \times m$

Example

| Input | Output |
|-------|--------|
| 3 | 18 |
| 2 3 | 28 |
| 7 1 | 100 |
| 4 4 | |

Problem N. El Profesor Preocupado

Source file name: Profesor.c, Profesor.cpp, Profesor.java, Profesor.py
Input: Standard
Output: Standard

Un profesor universitario considera está teniendo problemas para conciliar el sueño. Desea saber si las notas que ha puesto su universidad siguen alguna distribución de probabilidad.

Las notas en su trabajo se colocan de la A a la F , (siempre es una letra en mayúscula del abecedario y sólo son esas 6 notas posibles).

Para ayudar al profesor a dormir mejor, usted debe realizar un programa que cuente la cantidad de notas ha puesto de cada tipo por año y finalmente por tipo.

Input

El primer número P ($1 \leq P \leq 3000$) es la cantidad de profesores que hay en la universidad. Por cada profesor se colocan 2 líneas, la primer línea indica la cantidad C ($1 \leq C \leq 10^4$) de estudiantes que ese profesor ha calificado en su trayectoria y la segunda línea contiene las C . Por tanto, las siguientes $2 \times P$ líneas contienen las notas que cada profesor ha colocado (en el orden en que las colocó).

Output

Por cada profesor, se debe indicar el número de notas A , B , C , D , E o F que ha colocado en una línea, de la forma $A:N_A$ $B:N_B$ $C:N_C$ $D:N_D$ $E:N_E$ $F:N_F$. La última línea debe estar precedida por el string **TOTAL:** y se debe indicar con el mismo formato, el número total de cada nota que se ha colocado en la universidad. Ninguna línea debe terminar por espacio.

Example

| Input | Output |
|-----------|--------------------------------|
| 4 | A:1 B:0 C:0 D:0 E:0 F:2 |
| 3 | A:1 B:2 C:1 D:0 E:1 F:0 |
| A F F | A:0 B:0 C:2 D:0 E:0 F:2 |
| 5 | A:5 B:0 C:0 D:0 E:0 F:0 |
| A B B E C | TOTAL: A:7 B:2 C:3 D:0 E:1 F:4 |
| 4 | |
| C C F F | |
| 5 | |
| A A A A A | |



Problem O. El Concurso Canino

Source file name: Concurso.c, Concurso.cpp, Concurso.java, Concurso.py
Input: Standard
Output: Standard

A lo largo de los tiempos el hombre ha utilizado a los perros como herramientas de trabajo, cuido o simple compañía. Uno de los usos más comunes en muchas partes es tener perro ovejeros, encargados de arrear ovejas. Tomás es un pastor que tiene n perros con un nombre distinto que le ayudan, cada perro tiene un número de ovejas que puede arrear individualmente. En un concurso para arreo de ovejas, en el que Tomás quiere participar, le permiten llevar hasta 3 perros y el ganador se indica por el máximo número de ovejas que los perros puedan arrear.

Tomás sabe que tres perros pueden arrear el producto de la cantidad de que pueden arrear individualmente. Buscando ganar el concurso, Tomás desea saber cuáles perros debe llevar para poder maximizar sus posibilidades de ganar el concurso, además desea saber cuál es el número de ovejas que debe pedir arrear. Tomás sabe que no tiene dos perros que puedan arrear la misma cantidad de ovejas, ni dos perros que se llamen igual.

Usted quiere ayudar a Tomás a ganar el concurso, por lo que ha decidido hacer un programa que le indique a Tomás lo que desea saber.

Input

La primer línea tiene el número de perros que tiene Tomás (T y $1 \leq T \leq 1000$). Las siguientes T líneas tienen un string con letras en minúscula y utilizando el alfabeto en inglés de no más de 20 chars. Seguido de un entero O ($1 \leq O \leq 1000$) que indica el número de ovejas que él o ella pueden arrear individualmente.

Output

La salida debe ser una única línea donde se debe indicar el nombre de los 3 perros ordenados alfabéticamente y separados por espacio, seguido del número total de ovejas que entre los 3 pueden arrear.

Example

| Input | Output |
|--|---------------------------|
| 7 negro 30 perlita 12 diana 70 bragui 2 nymeria 6 canela 8 pancho 5 | diana negro perlita 25200 |

Problem P. Observando Aves

Source file name: Aves.c, Aves.cpp, Aves.java, Aves.py
Input: Standard
Output: Standard

Uno de los quehaceres que está ganando popularidad es la observación de aves. Sin embargo, encontrar a algunas especies es un trabajo complicado, principalmente porque a veces su supervivencia se basa en poder pasar desapercibidos.



Quetzal

Si considera un mapa como si fuera una matriz (con la primera fila iniciando en 1, lo mismo que la columna), donde 1 indica que hay un ave y 0 indica que no hay un ave, un guía hábil es capaz de ver un ave si al fijarse en un punto arbitrario, hay un ave en algún cuadro contiguo. Se considera un cuadro contiguo a los que rodean, en la siguiente cuadrícula, las "X" están en las casillas contiguas a la casilla que tiene un 1.

| | | |
|---|---|---|
| X | X | X |
| X | 1 | X |
| X | X | X |

Dados la matriz de ubicación y los puntos que un guía mira, se desea saber la cantidad de aves diferentes que ese guía va a ver en su tour.

Input

La entrada consiste en un único caso de prueba, la primer línea consiste en tres números enteros positivos F , C y P ($F, C, P \leq 10^3$), denotando las filas, columnas y cantidad de puntos que el guía mira. Considere que si mira la misma ave desde dos puntos, sólo cuenta como una.

Las siguientes F líneas tienen C valores (1 para indicar la presencia de un ave y 0 para indicar su ausencia).

Las siguientes P líneas contienen dos enteros cada una i y j indicando la posición (i, j) que el observador mira.

Output

Una línea que tiene el número total de aves que se pudo observar.

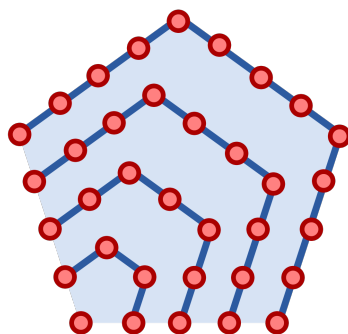
Example

| Input | Output |
|--|--------|
| 3 3 2 1 0 0 0 0 0 0 0 1 1 2 2 1 | 1 |

Problem Q. Número Pentagonal

Source file name: Pentagonal.c, Pentagonal.cpp, Pentagonal.java, Pentagonal.py
Input: Standard
Output: Standard

Un número pentagonal es aquel que sigue el patrón de la imagen:



1, 5, 12, 22, 35, ...

Primeros 5 números pentagonales

En general, un número pentagonal cumple con:

$$Pentagonal(n) = \frac{n(3n - 1)}{2}$$

Un profesor de matemática le ha solicitado, dado un número, encontrar el pentagonal más cercano a ese número. (Si estuviera equidistante con dos pentagonales, utilizar el menor).

Input

La primer línea contiene el número entero T ($1 \leq T \leq 10^5$) que es el número de valores que debe leer. Las siguientes T líneas tienen un número C ($1 \leq C \leq 10^{12}$).

Output

Por cada valor C , se debe imprimir en una línea independiente el número pentagonal más cercano a C .

Example

| Input | Output |
|-------|--------|
| 1 | 5 |
| 6 | |



Problem A. Telephone Substrings

Source file name: Telefonicos.c, Telefonicos.cpp, Telefonicos.java, Telefonicos.py
Input: Standard
Output: Standard

There is a multinational company that has headquarters in more than 20 countries, each country has an area code (of different length), within each country it has a local code (also of different length), within each area, it has buildings (also with codes of different length) and each building has offices with its local code (guess what, also with different length) than its direct telephone extension.

You have the complete multinational's phone directory, but you mixed up the phone numbers of country, buildings, office extensions, etc. You want to know how many offices there are in total. Your task is to indicate, given all the telephone numbers, how many of them already have a prefix (country code, area code) common to others.

Input

The first number T ($1 \leq T \leq 2000$) that you must read is the number of telephones in the directory. The next T lines are numbers of at most 50 digits indicating a telephone number.

Output

A whole number line should be printed with the number of prefixes found in the indicated telephone numbers.

Example

| Input | Output |
|-----------------------------|--------|
| 4 1 1234 12 123 | 6 |

Explanation

1 is a 3 number prefix, 12 is a 2 number prefix, 123 is a 1 number prefix.

Problem B. Apartment Buildings

Source file name: Edificios.c, Edificios.cpp, Edificios.java, Edificios.py
Input: Standard
Output: Standard

At New York, apartments are the standard way of living in the city, however, with the pandemic, there have been many movements of people leaving the buildings to live elsewhere, which has generated a series of problems for a company in charge of distributing gas through the pipelines intended for this purpose. As a preventive measure, it is necessary that the amount of supply does not exceed the average consumption per person and thus, in case of a leak, the possible damage will be minimized.



New York City

For this purpose, N buildings numbered from 1 to N , showing the number of people living in each building is indicated, but there are constantly people entering (adding) or leaving (subtracting) each building.

The company also needs to know the number of people living from building i to building j at any given time in order to be able to regulate the gas and the pressure at different points of the station in charge of distributing the worthy mineral.

Input

The first line contains two space-separated numbers: N $1 \leq N \leq 10^6$, indicating the number of buildings and Q $1 \leq Q \leq 10^5$, indicating the number of queries or updates.

The next line contains the sequence E with N non-negative integers, indicating the number of people living in each building. Being E_i the number of people living in building i .

The next Q lines are queries or updates on the status of the buildings.

If it is a query the line will be preceded by the letter q , following the format: $q \ i \ j$ where $1 \leq i \leq j \leq N$. Where i and j are the range of people living between building i and j , including and .

If it were an update, the line will be preceded by the letter u , with the format: $u \ i \ di$. Where i is the building number being modified, and di ($e_i \leq |di| \wedge di \leq 10^5$) is the difference with which the i -th building is being updated.

Output

For each query, one line must be printed, each line must contain the number of people living in the interval being queried.



Example

| Input | Output |
|-------|--------|
| 3 4 | 14 |
| 1 9 4 | 15 |
| q 1 3 | 4 |
| u 2 1 | |
| q 1 3 | |
| q 3 3 | |

Problem C. Health HPF

Source file name: Salud.c, Salud.cpp, Salud.java, Salud.py
Input: Standard
Output: Standard

Because of the current pandemic, vaccination centers have to serve a huge population, so they have categorized people in order to serve those most at risk first. But some people have been mistaken about when they should be vaccinated, have failed to identify their true priority, and have been late for their vaccination appointment.

However, the health guideline is clear, anyone who has registered for vaccination should be called according to their priority. Only one vaccination should be given to each person and it is possible that there may be several people with the same name.

To do this you must make a schedule that responds as efficiently as possible as to who is next to be called for vaccination.

Input

The test case consists of at most 10^4 lines, to be read until the end of the file.

The input consists of two tokens, the first one being the name of the person (of at most 20 chars), followed by a space and then a number which is its priority (the higher the higher its priority). The names are only letters [a-z] of the English alphabet and in lower case.

The calls are with a line that has the letter V (for vaccine).

Output

For each V, the name of the person to be vaccinated should be shown on one line. If there are vaccinations, but no persons in the record, a blank line should be left.

If there are unvaccinated persons, they should not be shown.

Example

| Input | Output |
|--|----------------------------|
| lanudo 1 house 100 emilia 20 V V bragui 30 V | house emilia bragui |
| hugo 100 rpc 90 V V V niquefa 80 V | hugo rpc niquefa |



Problem D. Lux quantity

Source file name: Lux.c, Lux.cpp, Lux.java, Lux.py
Input: Standard
Output: Standard

In simple terms, lumens (lm) is a measure of the total amount of light visible (to the naked eye) from a lamp or light source. The higher the number of lumens the brighter the bulb or lamp.

The lux is a derived unit, based on the lumen, which in turn is a derived unit based on the candela.

One lux equals one lumen per square meter, while one lumen equals one candela per steradian. The total luminous flux of a one candela source is equivalent to 4π lumens (since a sphere comprises 4π steradians).

This unit is important to be able to assess how much light a star or a set of stars emits. The alterations of light, can indicate that a star is born (millions of years ago by the time it takes for the light to reach the Earth or on the contrary, that a star dies, that there are several ways in which it can die, either because it widens and “explodes” or because it is so massive that it becomes a black hole. This leads to the need to be able to identify both the maximum light level and the minimum light level in an interval of space.

A given telescope, having lined up the stars in a list, assigns to each star the amount of lux it manages to capture. This amount is modified each time a change in a star is perceived.

In order to measure the number of stars dying in a sector of the galaxy, an astronomer needs to know what is the minimum lux and maximum lux in an interval of stars from position i to position j (inclusive).

For this purpose, you have been asked to write a program that will allow our friend to answer his query efficiently.

Input

The input consists of a single test case.

The first line contains two space-separated numbers: N $1 \leq N \leq 2 \times 10^5$, indicating the number of stars visible from the telescope and Q $1 \leq Q \leq 10^5$, indicating the number of queries or updates.

The next line contains the sequence E with N nonnegative integers, indicating the amount of lux that is received by each star. Being E_i the amount of lux that is perceived from star i .

The next Q lines are queries or updates on the status of the buildings.

If it is a query the line will be preceded by the word MAX or MIN, following the format: *MAX i j* or *MIN i j* where $1 \leq i \leq j \leq N$. Where i and j are the interval of stars for which the maximum or minimum is to be queried.

If it is an update, the line will be preceded by the word *UPD*, with the format: *UPD i ni*. Where i is the number of star being perceived differently, and ni ($0 \leq ni \leq 10^5$) is the new amount of lux perceived by the i -th star.

Output

For each query of the input, the maximum or minimum value as appropriate of the requested interval is printed on a separate line. (With both i and j included).

Example

| Input | Output |
|-------------|--------|
| 4 4 | 300 |
| 9 12 300 21 | 0 |
| MAX 2 3 | 21 |
| UPD 3 0 | |
| MIN 3 4 | |
| MAX 1 4 | |



Problem E. Friends' Friends

Source file name: Amigos.c, Amigos.cpp, Amigos.java, Amigos.py
Input: Standard
Output: Standard

At American universities, it is customary to have fraternities. Part of the fraternity is that if two people belong to the same fraternity, then they are considered “brothers”.

There are some people, even so, who do not join any fraternity because they have the odd goal of going to college to study (there is always someone who does not want to fit in with true college purpose right?).

At UGLI University, there was a problem with fraternity registration, but all the brothers, remembered at least one brother from their fraternity. Assuming that all UGLI people are numbered from 1 to N and you have all pairs of people who remembered their fraternity. Can you indicate how many fraternities and how many people do not belong to any fraternity?

Input

The first line consists of two space-separated integers N ($1 \leq N \leq 10^4$) and H ($1 \leq H \leq 2 \times 10^4$) indicating the number of people in UGLI and the number of acquaintances who remember their sisterhood respectively.

The next H lines contain two numbers i j separated by space, indicating that i remembers their sorority with j .

Output

The output should be two integers on one line separated by space, the first being the number of fraternities and the second the number of people who did not belong to any fraternity.

Example

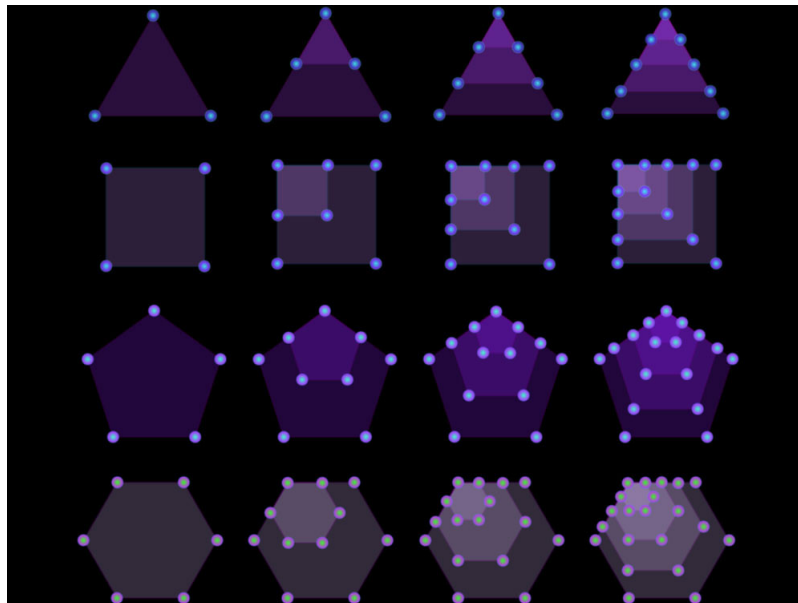
| Input | Output |
|-------------------|--------|
| 5 2 1 3 3 4 | 1 2 |

Problem F. The Faces of the Polygon

Source file name: Poligono.c, Poligono.cpp, Poligono.java, Poligono.py
Input: Standard
Output: Standard

Consider a polygonal figure¹ where all the lines that make it up are always horizontal or vertical. Given a figure, your task is to indicate the number of sides the figure has.

Even those faces on the border of the edge of the figure must be considered.



Polygon

Input

The first line consists of two space-separated integers L ($1 \leq L \leq 10^4$) and M ($1 \leq M \leq 10^4$) indicating the length and width of the figure.

The next L lines contain M characters each, forming the figure, “.” for empty space and “#” to indicate that it is part of the figure. The figure will always be connected.

Output

A line with an integer that corresponds to the number of sides the figure has (Consider that the edges of the figure also count as one of its sides).

Example

| Input | Output |
|--|-----------------|
| <pre> 5 5###. ..#.. .###. </pre> | <pre> 12 </pre> |

¹Image taken from: <https://www.flickr.com/photos/arquepoetica/2781771003>



Problem G. Link's Pursuit

Source file name: Persecucion.c, Persecucion.cpp, Persecucion.java, Persecucion.py
Input: Standard
Output: Standard

Ganon's calamity has once again been unleashed and has been contained thanks to Zelda's magic. Zelda this time has managed to put Ganon in a magic chest, meanwhile Link arrives with the sword that overcomes the darkness and plunges Ganon back into the underworld.

But oh calamity. It turns out that a Bokolin managed to infiltrate the palace and has stolen the chest. Link has arrived just in time to give chase. But the Bokolin has entered the temple of teleportation and gone to a city guarded by a centaleon.

It is known that the bokolin is very superstitious and only advances to places by means of preset paths and also uses only those paths that lead to a place with a centaleon, looking for any of them to stop the intrepid Link. The mechanics of the situation is that first the bokolin arrives at a place and flees until Link appears. (That is, if Link doesn't show up, the bokolin doesn't move). But Link always moves towards the bokolin.

Whenever Link faces a centaleon, he wins and kills it. This makes that place, and all paths leading to that place, unusable to the bokolin with the trunk.

Given the map of Hyrule, what is the maximum number of centaleons Link must kill before the Bokolin runs out of places to flee to and Link could therefore send Ganon back to the underworld.

All the locations in Hyrule are numbered from 1 to n .

Input

The first line has two numbers, N the number of places in the Hyrule Kingdom ($1 \leq N \leq 16$) and C the number of roads in the Hyrule Kingdom ($1 \leq C \leq 5 \times N$).

Output

A line with a single integer, corresponding to the maximum number of centaleons Link must defeat before completely cornering the Bokolin.

Example

| Input | Output |
|--------------------------|--------|
| 4 3 1 2 2 3 3 4 | 4 |

Problem H. Network Traffic

Source file name: Trafico.c, Trafico.cpp, Trafico.java, Trafico.py
Input: Standard
Output: Standard

There are N computers in a network. All computers are numbered from 0 to $N - 1$.

In the network it is necessary to send messages from one computer to all the others. This may require a message to pass through one or more intermediate computers if the source and destination of a message are not connected. When sending a message from one computer to another, it is required to pass through as few intermediate computers as possible.

Given the number of computers, the way they are connected, and an initial computer, is it possible to indicate the minimum number of computers through which a message must be sent?

Your task is that given the initial computer, indicate the minimum number of computers through which a message must pass for all other computers.

Input

The first line consists of three space-separated positive integers N ($1 \leq N \leq 5 \times 10^5$) the number of computers and C ($1 \leq C \leq 10^6$) indicating the number of connections the computers have and K being the initial computer.

The next C lines have two integers i, j ($0 \leq i, j < N$), indicating that computer i and computer j are connected.

Output

In each line, the computers should be shown in order from smallest to largest. Each line should start with the number of computers needed to reach them from the initial computer (except for the first line which indicates those that cannot be reached).

For all computers that are impossible to access, they should be shown preceded by a $+0:+$, also ordered from smallest to largest.

From this point, the number of computers to be visited to be able to send a message must be placed, starting with 1: to indicate those that are directly connected, 2: to indicate those that need an intermediate computer, etc.

The last line would be the one with the computers that require the maximum number of intermediate computers.

Example

| Input | Output |
|-------|--------|
| 5 3 1 | 0: 0 |
| 1 2 | 1: 2 3 |
| 1 3 | 2: 4 |
| 2 4 | |



Problem I. Binary strings

Source file name: Hileras.c, Hileras.cpp, Hileras.java, Hileras.py
Input: Standard
Output: Standard

Professor Moralov has a prime number obsession (those natural numbers greater than 1, that have no proper divisors greater than one). So as he is a computing and programming professor, he is used to work with binary numbers. But he is tired of finding primes in an interval, because he thinks it is too easy, so he has come up with a new challenge.

He got an string of an arbitrary length k , and he wants to compute the number of strings with that length that has a prime amount of 1's and a prime amount of 0's.

For example, if he has a 10 length string, he is looking for the strings with 7 1's and 3 0's (and vice-versa) or 5 0's and 5 1's.

In order to make the challenge even more interesting, professor Moralov wish to know how many binary strings of length between X and Y (included) exist, that each one of them has a prime amount of 0's and a prime amount of 1's. Since the amount can be really high, professor Moralov decided to have the amount mod 1000000007.

Input

In the first row comes an integer number Q with the number of queries. ($1 \leq Q \leq 10^5$). The next Q lines have two space separated integers X, Y , with $1 \leq X \leq Y \leq 2000$.

Output

For each query, it must be printed a line with the number of strings (mod 1000000007) with prime amount of 1's and 0's.

Example

| Input | Output |
|-------|--------|
| 3 | 0 |
| 2 2 | 6 |
| 4 4 | 492 |
| 10 11 | |

Problem J. The Parking Problem

Source file name: Parqueo.c, Parqueo.cpp, Parqueo.java, Parqueo.py
Input: Standard
Output: Standard

In Japan, there are parking lots that seek to make the best use of space. For this purpose, parking spaces are arranged in cubicles of 7 cars, with a single entrance.

Each one of them is organized internally as shown in the following figure:

| | | |
|----|----|----|
| | 31 | 43 |
| 15 | 35 | 31 |
| 64 | 43 | |

Always leaving the lower right corner and the top left corner empty.

On that board, it is assumed that each number corresponds to the vehicle's weight.

The issue is that the order in which they are placed usually differs from the order in which they are removed. The problem is that the robot in charge of making these changes needs energy to be able to change places and that energy corresponds to the sum of the weights of the vehicles it is moving and for reasons of amplitude, only two vehicles that are contiguous (either up-down or left-right) can be exchanged. In addition, it is guaranteed that no two vehicles have the same weight.

Given the initial position (how they arrived) and the final position (how they should leave), can you indicate what is the minimum energy expenditure of the robot?

Input

This is a single test case, the first 3 lines indicate the initial position, each line composed of positive integers (2-3-2 integers respectively) separated by space and all less than 10^4 .

The next 3 lines indicate the final position, each line composed of the same integers as in the previous three lines, just in a different configuration. In all cases, separated by space and in the same 2-3-2 format.

Output

The output a line that has a number indicating the minimum energy cost that the robot would need to rearrange the vehicles.

Example

| Input | Output |
|--|--------|
| 31 43 15 35 31 64 43 43 31 15 35 31 64 43 | 74 |



Problem K. Harry Potter and the Message in the Ministry

Source file name: Mensaje.c, Mensaje.cpp, Mensaje.java, Mensaje.py
Input: Standard
Output: Standard

The Ministry of Magic has installed a new courier system based on enchanted paper airplanes. Auror Harry Potter has been able to cast a spell so that the papers are copied once they reach a destination, so that he can leave a copy in one place and continue his flight to another destination. This copying can be done as many times as required.

However, the spell requires mandrake sap, which as we know is very expensive. It requires as much sap as the time it takes for a paper airplane to be transferred from one point to another (regardless of the number of copies).

On the other hand, the Ministry of Magic is like a labyrinth, full of passages and hiding places through which messages can be sent. Our favorite auror wants to be able to send messages to all the other aurors, minimizing the amount of mandrake sap required, to do this he must find the minimum time in which he can send a message to all the aurors.

Knowing that all aurors are numbered from 1 to n , and noting the times that there are transfer times between the conduits (which always go in two ways) of some of them (note that if A_1 is 12 seconds from A_2 and A_2 is 15 seconds from A_3 and there would be no direct conduit between A_1 and A_3 , it can be assumed that a plane passes through A_1 , leaving a copy, then leaves a copy at A_2 and finally continues its flight until it reaches A_3 with a total cost of 27 seconds). Can you help Harry Potter determine the minimum time in which all aurors are alerted with the message?

Input

This is a single test case, the first line consists of A , ($A, (N - 1 \leq N \leq 10^5)$) indicating the number of aurors in the ministry and C , ($N - 1 \leq C \leq 5 \times N$) indicating the number of conduits in the ministry. The next C lines each consist of 3 positive integers separated by space: $i \ j \ t$, indicating that from *auror* _{i} to *auror* _{j} takes t seconds. Where $i, j < A$ and $t < 10^4$.

Output

The output is a line with an integer, indicating the minimum time for all aurors to receive their message. (which is the sum of the times of all the paper planes).

Example

| Input | Output |
|---|--------|
| 5 7 1 2 1 1 3 8 2 3 1 3 4 2 3 5 10 4 5 2 2 5 8 | 6 |

Problem L. An Exhaustive Subway

Source file name: Metro.c, Metro.cpp, Metro.java, Metro.py
Input: Standard
Output: Standard

In New York City, you have a subway system that covers the entire city, crossing from the so-called “uptown” to “downtown”, passing of course through “midtown”. But even more, you can travel almost from any point in the city to any point in the city. There are more than 110 subway points in the city alone, not taking into account train departures to other states.

Leaving aside the train schedule, there are the times it takes between two stations on the train route. In addition, some stations intersect with others, so you can transfer and go to another part of the city. All stations are numbered from 1 to n .

Considering the NYC subway map with times between different stations, a series of queries between stations I and J , indicate the minimum time required to travel from I to J , assuming that a train is always available at any station we arrive at.

Input

The input is a single test case. The first line contains three numbers N ($1 \leq N \leq 110$) the number of train stations, L ($1 \leq L \leq 10^3$) the number of rail lines between train stations and Q ($1 \leq Q \leq 10^3$) which would be the queries on the time it takes to get from one station to another.

The following L lines have three positive integers, i , j and t , indicating that from station i there is a line to station j and that the train takes t minutes to get from one point to another. In all cases, the relationship is bidirectional. ($1 \leq i, j \leq N$) and $t < 30$.

The next Q lines have two integers, which are the queries, from point i to point j . ($1 \leq i, j \leq N$).

Output

For each query, the minimum time it would take from station i to station j should be printed on a separate line.

Example

| Input | Output |
|--------|--------|
| 6 8 6 | 8 |
| 1 2 5 | 9 |
| 1 3 4 | 6 |
| 1 4 6 | 2 |
| 2 5 9 | 8 |
| 2 6 1 | 4 |
| 4 6 1 | |
| 5 4 2 | |
| 5 1 10 | |
| 1 5 | |
| 2 3 | |
| 1 6 | |
| 4 2 | |
| 3 3 | |
| 2 5 | |



Problem M. The Matrix of Multiples

Source file name: Multiplos.c, Multiplos.cpp, Multiplos.java, Multiplos.py

Input: Standard

Output: Standard

In a matrix A of size $n \times m$ natural numbers are placed in such a way that a_{ij} is the j -th multiple of i . An example of a matrix with $n = 2$ and $m = 3$ can be seen in:

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 4 & 6 \end{pmatrix}$$

Two college engineering students compete to find the sum of all the numbers in the matrix, given the length (n) and width (m).

If the number turns out to be very large, the value modulo $10^9 + 7$ must be given.

Input

The first line will be a T number ($1 \leq T \leq 10^5$) indicating the number of test cases. The next T lines contain two numbers n and m indicating respectively the number of rows and columns of the matrix of multiples. ($1 \leq n, m \leq 10^6$).

Output

For each test case a line should be printed with the result of the sum of the $n \times m$ matrix's elements.

Example

| Input | Output |
|-------|--------|
| 3 | 18 |
| 2 3 | 28 |
| 7 1 | 100 |
| 4 4 | |

Problem N. The Concerned Professor

Source file name: Profesor.c, Profesor.cpp, Profesor.java, Profesor.py
Input: Standard
Output: Standard

A university professor is having trouble falling asleep. He wants to know if the grades his university has posted follow any probability distribution.

The grades in his paper are placed from *A* to *F*, (it is always a capital letter of the alphabet and only those 6 grades are possible).

To help the professor sleep better, you should make a program that counts the number of grades you have placed of each type by year and finally by type.

Input

The first number P ($1 \leq P \leq 3000$) is the number of professors in the university. For each professor 2 lines are placed, the first line indicates the number C ($1 \leq C \leq 10^4$) of students that professor has graded in his or her career and the second line contains the C . So, the next $2 \times P$ lines contain the grades that each teacher has placed (in the order in which he/she placed them).

Output

For each teacher, print in one line the number of grades *A*, *B*, *C*, *D*, *E* or *F* that he/she has placed in a line must be indicated, in the form $A:N_A B:N_B C:N_C D:N_D E:N_E F:N_F$. The last line must be preceded by the string **TOTAL:** and must indicate with the same format, the total number of each note that has been placed in the university.

No line should be ended with space.

Example

| Input | Output |
|-----------|--------------------------------|
| 4 | A:1 B:0 C:0 D:0 E:0 F:2 |
| 3 | A:1 B:2 C:1 D:0 E:1 F:0 |
| A F F | A:0 B:0 C:2 D:0 E:0 F:2 |
| 5 | A:5 B:0 C:0 D:0 E:0 F:0 |
| A B B E C | TOTAL: A:7 B:2 C:3 D:0 E:1 F:4 |
| 4 | |
| C C F F | |
| 5 | |
| A A A A A | |



Problem O. The Dog Show Contest

Source file name: Concurso.c, Concurso.cpp, Concurso.java, Concurso.py
Input: Standard
Output: Standard

Throughout the ages, man has used dogs as tools for work, care or simple companionship. One of the most common uses in many places is to have sheep dogs, in charge of herding sheep. Thomas is a shepherd who has n dogs with different names that help him, each dog has a number of sheep that he can herd individually. In a sheep herding contest, in which Thomas wants to participate, he is allowed to take up to 3 dogs and the winner is indicated by the maximum number of sheep the dogs can herd.

Tomas knows that three dogs can herd the product of the amount they can herd individually. Looking up to win the contest, Thomas wants to know which dogs he should take in order to maximize his chances of winning the contest, and he also wants to know the number of sheep he should ask to herd. Tomas knows that no two dogs can herd the same number of sheep, and no two dogs have the same name.

You want to help Thomas win the contest, so you have decided to make a program that tells Thomas what he wants to know.

Input

The first line has the number of dogs that Thomas has (T and $1 \leq T \leq 1000$). The next T lines have a string with lowercase letters using the English alphabet of no more than 20 chars. Followed by an integer O ($1 \leq O \leq 1000$) indicating the number of sheep he or she can herd individually.

Output

The output must be a single line where the name of the 3 dogs must be indicated in alphabetical order and separated by a space, followed by the total number of sheep that the 3 dogs can herd joined.

Example

| Input | Output |
|--|---------------------------|
| 7 negro 30 perlita 12 diana 70 bragui 2 nymeria 6 canela 8 pancho 5 | diana negro perlita 25200 |

Problem P. Bird watching

Source file name: Aves.c, Aves.cpp, Aves.java, Aves.py
Input: Standard
Output: Standard

One of the activities that is gaining popularity is bird watching. However, finding some species is a complicated task, mainly because sometimes their survival is based on being able to go unnoticed.



Quetzal

If you consider a map as if it were a matrix (with the first row starting at 1, the same as the column), where 1 indicates that there is a bird and 0 indicates that there is no bird, a skillful guide is able to see a bird if by looking at an arbitrary point, there is a bird in some contiguous square. A square is considered contiguous to the surrounding squares, in the following grid, the “X” are in the squares contiguous to the square that has a 1.

| | | |
|---|---|---|
| X | X | X |
| X | 1 | X |
| X | X | X |

Given the location matrix and the points a guide looks at, you want to know the number of different birds that guide will see on his or her tour.

Input

The input consists of a single test case, the first line consisting of three positive integers F , C and P ($F, C, P \leq 10^3$), denoting the rows, columns and number of points the guide looks at. Consider that if he looks at the same bird from two points, it only counts as one.

The following F lines have C values (1 to denote the presence of a bird and 0 to denote its absence).

The next P lines contain two integers each i and j indicating the position (i, j) that the observer looks at.

Output

A single line with the total number of birds that could be observed.

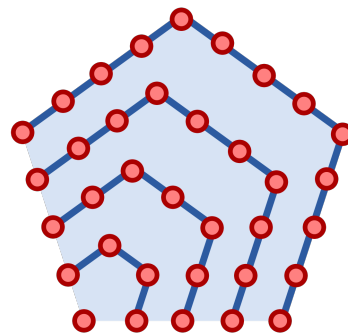
Example

| Input | Output |
|--|--------|
| 3 3 2 1 0 0 0 0 0 0 0 1 1 2 2 1 | 1 |

Problem Q. Pentagonal Number

Source file name: Pentagonal.c, Pentagonal.cpp, Pentagonal.java, Pentagonal.py
Input: Standard
Output: Standard

A pentagonal number is one that follows the pattern of the image:



1, 5, 12, 22, 35, ...

First 5 pentagonal numbers

In general, a pentagonal number satisfies:

$$P(n) = \frac{n(3n - 1)}{2}$$

A math teacher has asked you, given a number, to find the pentagonal closest to that number. (If it were equidistant with two pentagonal numbers, use the smaller one).

Input

The first line contains the integer T ($1 \leq T \leq 10^5$) which is the number of values to read. The next T lines have a number C ($1 \leq C \leq 10^{12}$).

Output

For each C value, the pentagonal number closest to C must be printed on a separate line.

Example

| Input | Output |
|-------|--------|
| 1 | 5 |
| 6 | |