

Universidad Mayor de San Andrés - Carrera de Informática

International Collegiate Programming Contest



**Club de Programación
Competitiva
{UMSA}**

Competencia Interna Primera División 2026

La Paz, 15 de agosto de 2026

Las paginas están numeradas desde el 1 al 21 (sin contar caratula e indice).

Setters:

*Alba Murillo, Nicolas Armando
Gutierrez Sardinias, Rossie Jashiel
Juarez Careaga, Andres Leonel
Navia Paiva, Kevin Arturo
Quispe Mamani, Miguel Angel
Rengel Sempertegui, Addis
Vargas Quelca, Alexis Jaime*

Judges:

*Carvajal Yucra, Oscar Gauss
Loza Ramirez, Samuel
Troche Venegas Rolando*

Sponsors:



Índice

A. El mosaico de Oridion	1
B. La Autopista de la Hormiga	3
C. Las direcciones perfectas del maestro	5
D. Elecciones en el País de las Maravillas	7
E. BANANITAS FC road to Cabo Verde	9
F. La Alcaldía de Puercoburgo	11
G. La Nueva Máquina de Alasitas I	13
H. La Nueva Máquina de Alasitas II	15
I. Bananitas FC vs Yapapa FC	17
J. Ene por Ene más Uno entre Dos	19
K. Cara o Escudo	21

Problema A

El mosaico de Oridion

Tiempo Límite: 1 seg, Memoria Límite: 256 MB

Autor: Who knows, could've been anyone

En unas ruinas recién descubiertas, la arqueóloga Liora encontró el mosaico de Oridion. Originalmente era una única franja de piedra de longitud n , pero el paso del tiempo la dejó dividida en k fragmentos. El fragmento i tiene longitud a_i .

Para conservar el hallazgo, Liora debe restaurar la franja usando un antiguo cemento de unión. El cemento solo reacciona con teselas de longitud uno, y las ruinas cuentan con una herramienta que puede desprender una tesela de ese mismo tamaño.

En un movimiento puede realizar una de estas acciones:

1. Elegir un fragmento de longitud $a_i \geq 2$ y desprender de él una tesela de longitud 1. El fragmento se reemplaza por dos fragmentos de longitudes 1 y $a_i - 1$.
2. Elegir dos fragmentos distintos, uno de longitud a_i y otro de longitud 1, y unirlos con el cemento. Ambos se reemplazan por un único fragmento de longitud $a_i + 1$.

Después de cada acción, todos los fragmentos que no participaron permanecen intactos. ¿Cuál es la menor cantidad de movimientos con la que Liora puede restaurar el mosaico como una sola franja de longitud n ?

Entrada

La primera línea contiene un entero t ($1 \leq t \leq 10^4$), el número de casos de prueba.

Cada caso de prueba contiene dos líneas:

- La primera contiene dos enteros n y k ($2 \leq n \leq 10^9$, $2 \leq k \leq 10^5$): la longitud original de la franja y la cantidad de fragmentos encontrados.
- La segunda contiene k enteros $a_1, a_2, \dots, a_k$ ($1 \leq a_i \leq n - 1$), las longitudes de los fragmentos. Se garantiza que $a_1 + a_2 + \dots + a_k = n$.

Se garantiza que la suma de k entre todos los casos de prueba de una entrada no supera 2×10^5 .

Salida

Para cada caso de prueba, imprime el mínimo número de movimientos necesarios para restaurar el mosaico.

Ejemplos

Ejemplo de entrada 1	Ejemplo de salida 1
4 5 3 3 1 1 5 2 3 2 11 4 2 3 1 5 16 6 1 6 1 1 1 6	2 3 9 15

Problema B

La Autopista de la Hormiga

Tiempo Límite: 1 seg, Memoria Límite: 256 MB

Autor: Who knows, could've been anyone

Sobre una cuadrícula infinita, todas las casillas están inicialmente en blanco. Una hormiga comienza en la casilla $(0, 0)$, mirando hacia el **Norte**. En cada paso aplica la misma regla de siempre:

- Si la casilla es **blanca**: gira 90° a la **derecha**, la pinta de **negro**, y avanza.
- Si la casilla es **negra**: gira 90° a la **izquierda**, la pinta de **blanco**, y avanza.

Te van a hacer Q **preguntas independientes**. Cada pregunta da un número de pasos X_k , y para cada una debes decir, asumiendo que la hormiga arranca siempre desde cero en $(0, 0)$ mirando al Norte:

1. Las coordenadas (x, y) de la casilla donde queda parada después de exactamente X_k pasos.
2. La **dirección** hacia la que queda mirando (NORTE, ESTE, SUR u OESTE).
3. El **color** de la casilla donde queda parada (BLANCO o NEGRO).
4. La **cantidad de casillas que están pintadas de negro** en ese momento, sobre todo el tablero (no solo las que la hormiga visitó — el total de casillas negras existentes tras esos X_k pasos).

Cada pregunta es independiente entre sí (no hay estado compartido: para cada X_k se responde “si la hormiga hubiera dado exactamente X_k pasos desde el inicio”).

Entrada

- Una línea con un entero Q ($1 \leq Q \leq 10^5$), el número de preguntas.
- Q líneas siguientes, cada una con un entero X ($0 \leq X \leq 10^{18}$).

Salida

Para cada pregunta, una línea con cinco valores separados por espacios:

`x y DIRECCION COLOR CASILLAS_NEGRAS`

en el mismo orden en que fueron consultadas.

Ejemplos

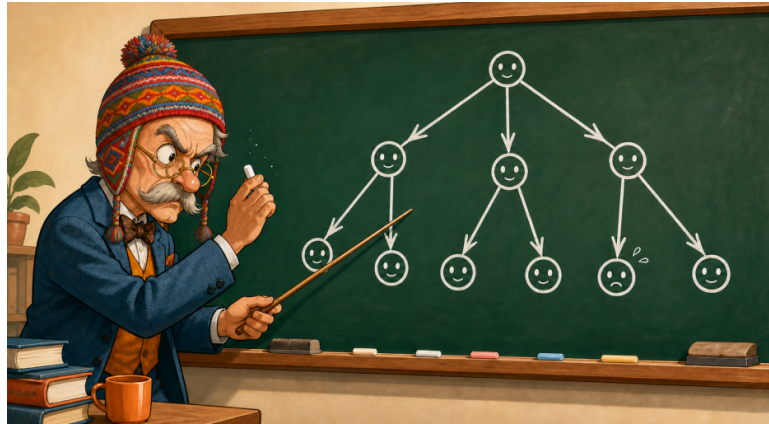
Ejemplo de entrada 1
6 0 1 5 9977 20000 10000000000000000000
Ejemplo de salida 1
0 0 NORTE BLANCO 0 1 0 ESTE BLANCO 1 -1 0 OESTE BLANCO 3 -15 10 OESTE NEGRO 715 -210 -184 NORTE BLANCO 1872 -19230769230769056 -19230769230769033 OESTE BLANCO 115384615384614951

Problema C

Las direcciones perfectas del maestro

Tiempo Límite: 1 seg, Memoria Límite: 256 MB

Autor: Who knows, could've been anyone



El maestro ha dibujado un árbol de N vértices en la pizarra. Por desgracia, se niega a borrarlo hasta que cada arista esté orientada. Nadie sabe por qué, pero el maestrillo lo llama «investigación importante».

Cada vértice v tiene dos valores, k_v y w_v . Después de orientar todas las aristas, el maestro otorga w_v puntos al vértice v si su grado de salida es exactamente k_v :

$$\text{outdeg}(v) = k_v$$

En caso contrario, el vértice obtiene 0 puntos y recibe una mirada de decepción del maestro.

Orienta todas las aristas para maximizar la suma total de puntos y, más importante aún, mantener feliz al maestro.

Entrada

La primera línea contiene un entero N , el número de vértices del árbol.

La segunda línea contiene N enteros:

$$k_1, k_2, \dots, k_N$$

La tercera línea contiene N enteros:

$$w_1, w_2, \dots, w_N$$

Cada una de las siguientes $N - 1$ líneas contiene dos enteros u y v , que indican una arista entre los vértices u y v .

Restricciones

- $1 \leq N \leq 2 \cdot 10^5$
- $0 \leq k_v \leq \deg(v)$
- $1 \leq w_v \leq 10^9$
- El grafo dado es un árbol.

Salida

Imprime un único entero: la máxima suma de puntos que puede obtenerse orientando todas las aristas.

Ejemplos

Ejemplo de entrada 1	Ejemplo de salida 1
1 0 7	7

Ejemplo de entrada 2	Ejemplo de salida 2
2 1 0 5 8 1 2	13

Ejemplo de entrada 3	Ejemplo de salida 3
3 0 1 0 4 10 7 1 2 2 3	17

Explicación

En el primer ejemplo, el único vértice tiene grado de salida 0, por lo que aporta 7 puntos.

En el segundo ejemplo, al orientar la arista como $1 \rightarrow 2$, ambos vértices alcanzan sus grados de salida requeridos y aportan un total de $5 + 8 = 13$ puntos.

En el tercer ejemplo, se orientan las aristas como $1 \rightarrow 2$ y $2 \rightarrow 3$. Los vértices 2 y 3 alcanzan sus grados de salida requeridos y aportan $10 + 7 = 17$ puntos.

Problema D

Elecciones en el País de las Maravillas

Tiempo Límite: 1 seg, Memoria Límite: 256 MB

Autor: Who knows, could've been anyone

En el **País de las Maravillas**, se aproximan las elecciones presidenciales y el ambiente está que arde. Varios partidos políticos se han lanzado a la contienda, cada uno con estrategias... particulares para conseguir votos:

- **PDC** (*Partido De las Casas*): promete una casa de 2 pisos a cada elector (nadie sabe con qué presupuesto).
- **UN** (*Unidos Nunca-jamás*): en cada mitin cantan a todo pulmón “Ojo del Tigre” para “motivar” al electorado.
- **MAS** (*Movimiento Alegre del Sombrero*): en vez de dar discursos, simplemente bailan sin parar frente a las urnas.
- **ICPC** (*Incredible Coalición de Personajes Curiosos*): reparte té y pastelitos “come-me” a todo el que se acerque.
- **LIBRE** (*La Ilusión Bien Repartida y Eterna*): promete que nadie tendrá que trabajar nunca más (spoiler: nadie sabe cómo).

Ante tanto caos, la **ONPE** (*Organización Nacional de Procesos Electorales*) — la única entidad que mantiene la cordura en todo este manicomio — te ha contratado para programar el sistema que determine, apenas cerradas las urnas, si hay un **ganador en primera vuelta** o si el país deberá prepararse para una **segunda vuelta**.

Reglas de la ONPE

Un partido gana en primera vuelta si se cumple **al menos una** de estas condiciones:

1. Obtiene una votación **mayor al 50 %**.
2. Obtiene una votación **mayor al 40 %**, y además le saca **10 o más puntos porcentuales de diferencia** al partido que queda en segundo lugar.

Si ningún partido cumple lo anterior, **habrá segunda vuelta**.

Entrada

- La primera línea contiene un entero N ($2 \leq N \leq 100$): la cantidad de partidos.
- La segunda línea contiene N enteros $a_1, a_2, \dots, a_N$ ($1 \leq a_i \leq 100$): el porcentaje de votación de cada partido. Se garantiza que $a_1 + a_2 + \dots + a_N = 100$.

Salida

Imprime una sola línea:

- NO HABRA SEGUNDA VUELTA si hay un ganador en primera vuelta.
- HABRA SEGUNDA VUELTA si ningún partido cumple las condiciones de victoria.

Ejemplos

Ejemplo de entrada 1	Ejemplo de salida 1
3 55 30 15	NO HABRA SEGUNDA VUELTA

Ejemplo de entrada 2	Ejemplo de salida 2
3 35 33 32	HABRA SEGUNDA VUELTA

Ejemplo de entrada 3	Ejemplo de salida 3
3 45 45 10	HABRA SEGUNDA VUELTA

Problema E

BANANITAS FC road to Cabo Verde

Tiempo Límite: 1 seg, Memoria Límite: 256 MB

Autor: Who knows, could've been anyone



El famoso club **Bananitas FC**, cansado de perder partidos y sin lograr su primera victoria, decidió viajar a Cabo Verde para mejorar su nivel de juego. Sin embargo, el dinero es un problema y deben arreglárselas solos como siempre, así que los jugadores optaron por adentrarse en el mundo de la programación para financiar el viaje.

Como parte de una entrevista técnica para conseguir su primer trabajo, les plantearon el siguiente problema:

Dado un array de enteros positivos (A). En una operación puedes elegir un índice i y reemplazar a_i por a_i^2 o por $\lfloor \sqrt{a_i} \rfloor$. Puedes aplicar estas operaciones tantas veces como desees. Calcula el mínimo número de operaciones necesarias para que el arreglo quede ordenado de forma no decreciente y que se cumpla $1 \leq a_i \leq 10^{18}$ en todo momento.

Lamentablemente, ninguno de los jugadores tiene el nivel necesario para resolverlo. Aun así, mantienen la esperanza de que algún participante de la competencia interna de la UMSA pueda ayudarlos. ¿Podrás resolver el problema y acercar a Bananitas FC a su sueño Cabo Verdiano?

Entrada

- La primera línea contiene un entero N ($1 \leq N \leq 10^5$): la cantidad de elementos del array.
- La segunda línea contiene N enteros a_i ($1 \leq a_i \leq 10^{18}$).

Salida

La cantidad mínima de operaciones necesarias para que el arreglo quede ordenado de forma no decreciente, cumpliendo en todo momento que $1 \leq a_i \leq 10^{18}$.

Ejemplos

Ejemplo de entrada 1	Ejemplo de salida 1
3 5 2 1	3

Ejemplo de entrada 2	Ejemplo de salida 2
4 1 5 5 5	0

Ejemplo de entrada 3	Ejemplo de salida 3
4 1 16 4 16	1

Problema F

La Alcaldía de Puercoburgo

Tiempo Límite: 1 seg, Memoria Límite: 256 MB

Autor: Who knows, could've been anyone

Tras las últimas elecciones municipales en la próspera ciudad de **Puercoburgo**, el Lobo Feroz fue elegido sorpresivamente como el nuevo alcalde. Para ganarse la confianza de sus tres ciudadanos y disipar viejos rencores, su principal promesa de campaña fue construir el nuevo edificio de la alcaldía en un punto $P = (X, Y)$ del plano tal que **la suma de las distancias que deben recorrer los tres cerditos desde sus casas hasta la alcaldía sea mínima**.

El mapa de Puercoburgo está representado mediante un plano bidimensional y las coordenadas son enteras. La casa de cada uno de los tres cerditos se encuentra ubicada en una coordenada entera conocida $C_i = (x_i, y_i)$ para $i \in \{1, 2, 3\}$.

Debido al trazado urbano en cuadrícula estricta de la ciudad, los cerditos únicamente pueden desplazarse a través de las calles de forma paralela a los ejes coordenados. Por lo tanto, la distancia entre una casa $C_i = (x_i, y_i)$ y la alcaldía $P = (X, Y)$ está dada por la **distancia Manhattan**:

$$d(C_i, P) = |x_i - X| + |y_i - Y|$$

El Lobo quiere demostrar la eficiencia de su gestión publicando de inmediato la planificación del proyecto. Tu tarea es ayudarlo calculando el mínimo de la suma de las distancias Manhattan desde las tres casas hasta la posición óptima del edificio de la alcaldía.

Entrada

La primera línea contiene un único entero T ($1 \leq T \leq 10^4$), que representa el número de casos de prueba.

A continuación, cada caso de prueba consta de 3 líneas. Cada una de estas líneas contiene dos enteros x_i, y_i ($-10^9 \leq x_i, y_i \leq 10^9$), que indican las coordenadas rectangulares de la casa del i -ésimo cerdito.

Salida

Para cada caso de prueba, imprime una sola línea con un entero: la menor suma total posible de las distancias Manhattan que recorrerán los tres cerditos desde sus respectivas casas hasta la alcaldía.

Note que el Lobo puede construir su alcaldía sobre la casa de algunos cerditos.

Ejemplos

Ejemplo de entrada 1	Ejemplo de salida 1
1 0 0 1 1 2 0	3

Problema G

La Nueva Máquina de Alasitas I

Tiempo Límite: 1 seg, Memoria Límite: 256 MB

Autor: Who knows, could've been anyone

Este año apareció una máquina nueva en las Alasitas, y Oscar la tiene entre ojos: cree que se le puede sacar mucha plata.

La máquina es completamente electrónica. Por dentro arma un mazo virtual con a **tréboles negros** y b **corazones rojos**, y lo baraja uniformemente al azar: todos los órdenes posibles del mazo son igual de probables. Jugar una partida cuesta c bolivianos.

Una partida funciona así. Las cartas del mazo se reparten una por una hasta agotarlo. Antes de que salga cada carta, la pantalla te pide anunciar un palo, y tú aprietas uno de los dos botones: **trébol** o **corazón**. Por cada carta cuyo palo hayas anunciado correctamente, la máquina te paga **1 boliviano**.

La pantalla revela las cartas todas juntas recién al terminar la partida. Por eso, mientras juegas, lo único que conoces es la composición inicial del mazo.

La máquina se recarga cada día con valores distintos de a , b y c , que anuncia en la pantalla de inicio antes de que pagues. Oscar quiere evaluar q días.

Oscar es excompetidor de ICPC, así que la estrategia óptima ya la tiene clarísima. El problema es que le da pereza hacer las cuentas, y como confía en tu potencial te pide a ti que le digas, para cada día, cuál sería el valor esperado de su **ganancia neta**: los bolivianos que cobra menos los c que pagó. Así decide si vale la pena caminar hasta la feria.

Ese valor puede ser negativo, y es un número racional P/Q . Imprímelo módulo $10^9 + 7$.

Entrada

La primera línea contiene un entero q : la cantidad de días a evaluar.

Cada una de las siguientes q líneas contiene tres enteros a , b y c : los tréboles negros del mazo de ese día, los corazones rojos, y el costo de la partida en bolivianos.

$$1 \leq q \leq 10^5$$

$$0 \leq a, b \quad 1 \leq a + b \leq 10^9 \quad 0 \leq c \leq 10^9$$

Se garantiza que la suma de $\min(a, b)$ sobre todos los días no supera 10^6 .

Salida

Imprime q líneas. En la i -ésima, el valor $P \cdot Q^{-1} \bmod (10^9 + 7)$, donde P/Q es la ganancia neta esperada del día i bajo juego óptimo.

Formalmente, sea $M = 10^9 + 7$. Se puede demostrar que la respuesta se escribe como P/Q con P y Q enteros coprimos y $Q \not\equiv 0 \pmod{M}$. Debes imprimir el único entero x con $0 \leq x < M$ tal que $x \cdot Q \equiv P \pmod{M}$.

Ejemplos

Ejemplo de entrada 1	Ejemplo de salida 1
3 1 1 1 4 0 2 2 2 3	0 2 1000000006

Nota

Día 1. El mazo tiene un trébol y un corazón, y la partida cuesta 1 boliviano. Como Oscar no ve ninguna carta mientras juega, en ambos turnos su información es exactamente la misma, y cualquier botón que apriete acierta con probabilidad $\frac{1}{2}$. Cobra $\frac{1}{2} + \frac{1}{2} = 1$ boliviano en promedio, así que su ganancia neta esperada es 0: ese día la máquina es justa.

Día 2. Las cuatro cartas son tréboles y la pantalla lo dice desde el inicio, así que Oscar aprieta siempre trébol y acierta las cuatro. Gana $4 - 2 = 2$.

Problema H

La Nueva Máquina de Alasitas II

Tiempo Límite: 1 seg, Memoria Límite: 256 MB

Autor: Who knows, could've been anyone

*Este problema es una versión diferente de **La Nueva Máquina de Alasitas I**. El enunciado es idéntico salvo por lo que aparece en negrita.*

Este año apareció una máquina nueva en las Alasitas, y Oscar la tiene entre ojos: cree que se le puede sacar mucha plata.

La máquina es completamente electrónica. Por dentro arma un mazo virtual con a tréboles negros y b corazones rojos, y lo baraja uniformemente al azar: todos los órdenes posibles del mazo son igual de probables. Jugar una partida cuesta c bolivianos.

Una partida funciona así. Las cartas del mazo se reparten una por una hasta agotarlo. Antes de que salga cada carta, la pantalla te pide anunciar un palo, y tú aprietas uno de los dos botones: trébol o corazón. Por cada carta cuyo palo hayas anunciado correctamente, la máquina te paga 1 boliviano.

La pantalla revela **cada carta apenas aprietas el botón**. Por eso, mientras juegas, **sabes en todo momento qué cartas siguen dentro del mazo**.

La máquina se recarga cada día con valores distintos de a , b y c , que anuncia en la pantalla de inicio antes de que pagues. Oscar quiere evaluar q días.

Oscar es excompetidor de ICPC, así que la estrategia óptima ya la tiene clarísima. El problema es que le da pereza hacer las cuentas, y como confía en tu potencial te pide a ti que le digas, para cada día, cuál sería el valor esperado de su **ganancia neta**: los bolivianos que cobra menos los c que pagó. Así decide si vale la pena caminar hasta la feria.

Ese valor puede ser negativo, y es un número racional P/Q . Imprímelo módulo $10^9 + 7$.

Entrada

La primera línea contiene un entero q : la cantidad de días a evaluar.

Cada una de las siguientes q líneas contiene tres enteros a , b y c : los tréboles negros del mazo de ese día, los corazones rojos, y el costo de la partida en bolivianos.

$$1 \leq q \leq 10^5$$

$$0 \leq a, b \quad 1 \leq a + b \leq 10^9 \quad 0 \leq c \leq 10^9$$

Se garantiza que la suma de $\min(a, b)$ sobre todos los días no supera 10^6 .

Salida

Imprime q líneas. En la i -ésima, el valor $P \cdot Q^{-1} \bmod (10^9 + 7)$, donde P/Q es la ganancia neta esperada del día i bajo juego óptimo.

Formalmente, sea $M = 10^9 + 7$. Se puede demostrar que la respuesta se escribe como P/Q con P y Q enteros coprimos y $Q \not\equiv 0 \pmod{M}$. Debes imprimir el único entero x con $0 \leq x < M$ tal que $x \cdot Q \equiv P \pmod{M}$.

Ejemplos

Ejemplo de entrada 1	Ejemplo de salida 1
3	5000000004
1 1 1	2
4 0 2	833333339
2 2 3	

Día 1. El mazo tiene un trébol y un corazón, y la partida cuesta 1 boliviano. En el primer turno Oscar acierta con probabilidad $\frac{1}{2}$. Pero al ver esa carta ya sabe con certeza cuál es la que queda, así que el segundo turno lo acierta siempre. Cobra $\frac{1}{2} + 1 = \frac{3}{2}$ en promedio, de modo que su ganancia neta esperada es $\frac{1}{2}$, y $\frac{1}{2} \equiv 5000000004 \pmod{10^9 + 7}$.

Día 2. Las cuatro cartas son tréboles y la pantalla lo dice desde el inicio, así que Oscar aprieta siempre trébol y acierta las cuatro. Gana $4 - 2 = 2$.

Problema I

Bananitas FC vs Yapapa FC

Tiempo Límite: 1 seg, Memoria Límite: 256 MB

Autor: Who knows, could've been anyone



En el estadio más grande de la ciudad hay N jugadores distribuidos en distintas zonas. Las zonas están conectadas mediante caminos bidireccionales, y se sabe que cualquier par de zonas está conectado por un único recorrido simple; es decir, la red de caminos forma un árbol.

Cada jugador pertenece a uno de dos equipos:

- **Bananitas FC**
- **Yapapa FC**

Para cualquier par de zonas (u, v) con $u \leq v$, consideremos el camino simple que las conecta. Diremos que este camino es **favorable para Bananitas FC** si el número de jugadores de Bananitas FC presentes en las zonas del camino es estrictamente mayor que el número de jugadores de Yapapa FC.

Tu tarea es determinar cuántos pares de zonas (u, v) generan un camino favorable para Bananitas FC.

Entrada

La primera línea contiene un entero N , el número de zonas del estadio.

Las siguientes $N - 1$ líneas contienen dos enteros u y v , indicando que existe un camino entre las zonas u y v .

La segunda línea contiene una cadena de longitud N . El carácter i -ésimo es:

- B si la zona i contiene un jugador de Bananitas FC.
- Y si la zona i contiene un jugador de Yapapa FC.

Salida

Imprime un único entero: la cantidad de pares (u, v) con $u \leq v$ tales que, en el camino simple entre u y v , la cantidad de jugadores de Bananitas FC es estrictamente mayor que la cantidad de jugadores de Yapapa FC.

Ejemplos

Ejemplo de entrada 1	Ejemplo de salida 1
5 1 2 1 3 3 4 3 5 BBYYB	7

Ejemplo de entrada 2	Ejemplo de salida 2
10 2 8 4 9 1 6 3 7 1 8 5 8 1 10 1 7 7 9 YBBBBBYBY	16

Problema J

Ene por Ene más Uno entre Dos

Tiempo Límite: 1 seg, Memoria Límite: 256 MB

Autor: Who knows, could've been anyone

Los antiguos jueces Ent de la UMSA tenían varias tradiciones al organizar concursos: incluir un problema que se resolviera usando $N(N + 1)/2$, mencionar al Botas en algún enunciado, lograr que Chuck Norris ganara el contest, olvidarse de probar algún caso, ir a comer una mega burger al terminar, o sufrir algún evento desafortunado que mantuviera a todos preocupados durante toda la competencia.

Cuenta la leyenda que llegó el día en que a nuestro querido competidor Botas le tocó organizar su primera DIV 2. Entre recibir problemas, escribir casos de prueba, revisar soluciones y comprar los globos para el concurso, apenas tenía tiempo para respirar.

Desesperado, fue al legendario laboratorio Patito a preguntarle a Addis cuántos globos debía comprar.

Addis, que por aquella época no estaba precisamente cuerdo, respondió:

- Tengo una lista de los participantes.
- Los sentaremos en una cuadrícula de N filas y M columnas dentro del laboratorio.
- A cada participante le he asignado un número $B_{i,j}$ que representa qué tan bueno es compitiendo.
- Frente a cada participante colocaremos una cuerda con globos.
- Cada participante debe tener al menos un globo.
- Si un participante es mejor que otro participante sentado en una posición adyacente, entonces su cuerda debe verse más impresionante.
- Dos posiciones son adyacentes si comparten un lado (arriba, abajo, izquierda o derecha).
- Una cuerda se considera más impresionante si tiene estrictamente más globos.
- Deberías comprar la menor cantidad posible de globos para poder cumplir estas condiciones.

Botas estaba seguro de que Addis estaba bromeando o simplemente había perdido la cordura, pero prefirió hacerle caso.

Ayuda a Botas a determinar cuántos globos debe comprar.

Entrada

La primera línea contiene dos enteros N y M ($1 \leq N \cdot M \leq 10^6$), que representan el número de filas y columnas de participantes.

Las siguientes N líneas contienen M enteros cada una. El entero $B_{i,j}$ ($0 \leq B_{i,j} \leq 10^5$) representa qué tan bueno es el participante sentado en la fila i y columna j .

Salida

Imprime un único entero: la cantidad mínima de globos que Botas debe comprar.

Ejemplos

Ejemplo de entrada 1	Ejemplo de salida 1
1 3 1 0 2	5

Puedes darle 2 globos al primer participante, 1 globo al segundo y 2 globos al tercero.

Ejemplo de entrada 2	Ejemplo de salida 2
1 3 1 2 2	4

Puedes darle 1 globo al primer participante, 2 globos al segundo y 1 globo al tercero. Nota que el segundo participante es mejor que el primero, por lo que debe tener más globos.

Ejemplo de entrada 3	Ejemplo de salida 3
2 2 1 2 3 4	8

Una distribución válida es:

1 2
2 3

Cada participante tiene al menos un globo, y cualquier participante con un valor mayor que el de un vecino adyacente recibe más globos que dicho vecino. No existe una distribución válida con menos de 8 globos.

Problema K

Cara o Escudo

Tiempo Límite: 1 seg, Memoria Límite: 256 MB

Autor: Who knows, could've been anyone

Dilan acaba de volver de su viaje al Campamento de Programación que se llevó a cabo hace dos semanas en la ciudad de La Plata, Argentina. Volvió con la cabeza llena de problemas de lógica y con muchas ganas de trolearse... digo, de poner a prueba a sus amigos en Bolivia. Así que reunió a la banda y les planteó el siguiente problema:

Tienes n monedas puestas en fila, y todas están mostrando **cara**. En un movimiento, puedes elegir dos monedas **adyacentes** y voltearlas ambas al mismo tiempo (si una moneda muestra cara pasa a mostrar escudo, y si muestra escudo pasa a mostrar cara).

Dilan quiere saber si, usando cualquier cantidad de movimientos (incluso cero), es posible dejar **todas** las monedas mostrando **escudo**.

Ayuda a los amigos de Dilan a responder esta pregunta antes de que él se ponga insoportable presumiendo lo que aprendió en Argentina.

Entrada

La entrada contiene un único entero n ($1 \leq n \leq 10^9$), la cantidad de monedas.

Salida

Imprime SI si es posible lograr que todas las monedas muestren escudo, o NO en caso contrario.

Ejemplos

Ejemplo de entrada 1	Ejemplo de salida 1
1	NO

Ejemplo de entrada 2	Ejemplo de salida 2
2	SI