

Torneo Argentino Interuniversitario de Programación

- Facultad de Ciencias Exactas y Naturales — Universidad de Buenos Aires
- Facultad de Informática — Universidad Nacional de La Plata
- Facultad de Ingeniería y Ciencias Hídricas — Universidad Nacional del Litoral
- Facultad de Matemática, Astronomía y Física — Universidad Nacional de Córdoba

SESIÓN DE COMPETENCIA

24 de septiembre de 2011

Este conjunto contiene 10 problemas; las páginas están numeradas de 1 a 17.

Información General

Salvo indicación en contrario, lo siguiente vale para todos los problemas.

Entrada

1. La entrada se debe leer de la entrada estándar (standard input).
2. La entrada contiene varios casos de prueba. Cada caso se describe utilizando una cantidad de líneas que depende del problema.
3. Cuando una línea de datos contiene varios valores, éstos se separan utilizando exactamente *un* espacio entre ellos. Ningún otro espacio aparece en la entrada. No hay líneas en blanco.
4. Todas las líneas, incluyendo la última, tienen la marca usual de fin de línea.
5. El final de la entrada se indica con una línea que contiene ciertos valores que dependen del problema. Dicha línea no se debe procesar como un caso de prueba.

Salida

1. La salida se debe escribir en la salida estándar (standard output).
2. El resultado de cada caso de prueba debe aparecer en la salida utilizando una cantidad de líneas que depende del problema.
3. Cuando una línea de resultados contiene varios valores, éstos se deben separar utilizando exactamente *un* espacio entre ellos. Ningún otro espacio debe aparecer en la salida. No debe haber líneas en blanco.
4. Todas las líneas, incluyendo la última, deben tener la marca usual de fin de línea.
5. No se debe utilizar ninguna marca especial para indicar el final de la salida.

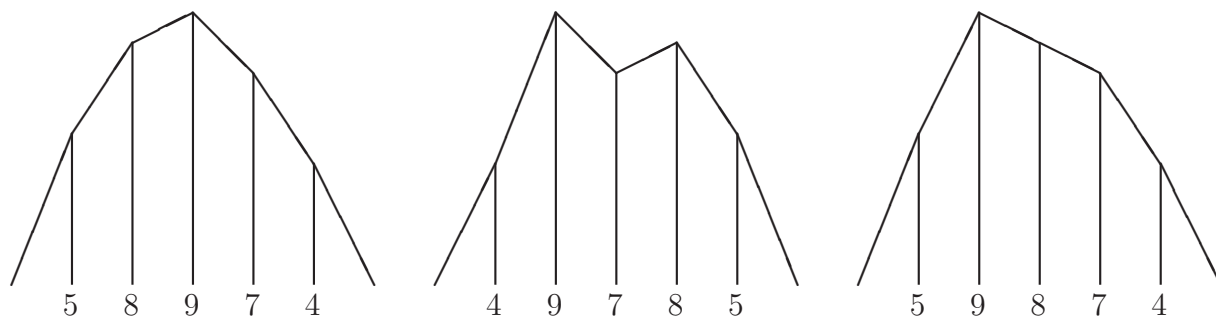
Problema A

Armado de Carpas

El grupo Instaladores de Carpas Para Circo (ICPC) tiene una forma particular de armar sus carpas, que depende básicamente de una vista bidimensional de las mismas.

Una carpa dada se arma utilizando una cantidad N de pilotes de diversas alturas. El procedimiento utilizado por el ICPC para armar la carpa es el siguiente. Primero se marcan en el suelo $N + 2$ puntos alineados, donde la distancia entre cada par de puntos consecutivos es de exactamente 2 metros. Luego se coloca verticalmente un pilote en cada uno de los N puntos centrales. Finalmente, se coloca la tela de la carpa uniendo el extremo (superior) de cada pilote con los extremos de los pilotes vecinos, y en el caso del primer y el último pilotes, se los une con los puntos libres que quedaron en el suelo.

La siguiente figura muestra 3 posibles maneras de armar una carpa utilizando las instrucciones anteriores, con pilotes de alturas 4, 5, 7, 8 y 9.



Luego de años de trabajo, el ICPC llegó a la conclusión de que para obtener carpas útiles y resistentes, es necesario que el ángulo que forman dos porciones consecutivas de tela en el extremo de cada pilote, medido hacia adentro de la carpa, sea *estrictamente* menor a 180 grados. En la figura mostrada, sólo la carpa de la izquierda cumple esa condición. En la carpa del centro el ángulo es mayor a 180 grados en los pilotes de alturas 4 y 7, mientras que en la carpa de la derecha el ángulo tiene exactamente 180 grados en el pilote de altura 8. Llamamos carpas válidas a aquellas que cumplen la recomendación del ICPC.

Por supuesto, dada la cantidad de pilotes y sus alturas, hay muchas formas de colocarlos y de ese modo obtener carpas diferentes, algunas de las cuales van a ser válidas y otras no. La tarea en este caso es, dados esos datos, contar la cantidad de carpas válidas diferentes que es posible armar. Dos carpas válidas se consideran diferentes si la secuencia de alturas de los pilotes de una de ellas, leída de izquierda a derecha, es distinta de la secuencia de alturas de la otra, leída de igual manera.

Entrada

Cada caso de prueba se describe utilizando dos líneas. La primera línea contiene un entero N que indica la cantidad de pilotes ($1 \leq N \leq 60$). La segunda línea contiene N enteros H_i que representan las alturas de los pilotes en metros ($1 \leq H_i \leq 10^9$ para $1 \leq i \leq N$). El final de la entrada se indica con una línea que contiene el número -1 .

Salida

Para cada caso de prueba, imprimir en la salida una línea conteniendo un entero que representa la cantidad de carpas válidas diferentes que se pueden armar con los pilotes dados.

Entrada de ejemplo	Salida para la entrada de ejemplo
5	2
4 5 7 8 9	16
7	1
33 65 57 64 63 61 49	1
1	0
1000000000	0
3	
2 2 3	
3	
1 3 1	
4	
2 2 2 2	
-1	

Problema B

Blanco y Negro

El famoso juego Blanco y Negro es un solitario que utiliza fichas iguales entre sí. Cada ficha tiene dos caras de colores diferentes. Sorprendentemente, esos colores son blanco y negro.

Para jugar se comienza ubicando N fichas formando una sola fila. Cada ficha tiene una de sus caras hacia arriba, elegida al azar. También al azar se elige un patrón objetivo. Dicho patrón es una secuencia ordenada de N colores blanco o negro. En cada jugada se elige un conjunto de fichas consecutivas, y todas ellas se invierten, de manera tal que en cada una de ellas el color que estaba arriba pasa a estar abajo y viceversa. El objetivo del juego es lograr que las caras superiores de las fichas muestren el patrón que se eligió al comienzo.

Barby descubrió este solitario hace poco tiempo, y rápidamente se dió cuenta de que siempre se puede ganar, ya que cada ficha se puede invertir individualmente. Para hacer el juego más interesante, Barby decidió intentar formar el patrón en la menor cantidad posible de jugadas. A Barby sólo le interesa la cantidad de jugadas, independientemente de cuántas fichas se inviertan en cada jugada. Para saber qué tan bien lo está haciendo, Barby les pide que hagan un programa que a partir de la disposición inicial de las fichas y el patrón objetivo, indique la mínima cantidad de jugadas necesarias para ganar el solitario. ¿Le van a decir que no?

Entrada

Cada caso de prueba se describe utilizando una línea. La línea contiene dos cadenas S y T no vacías y de igual longitud. La cadena S representa la disposición inicial de las fichas, mientras que la cadena T representa el patrón objetivo. Ambas cadenas utilizan solamente las letras mayúsculas “B” y “N”, que representan respectivamente al color blanco y al color negro. Las cadenas tienen a lo sumo 500 letras. El final de la entrada se indica con una línea que contiene dos asteriscos (“*”).

Salida

Para cada caso de prueba, imprimir en la salida una línea conteniendo un entero que representa la mínima cantidad de jugadas necesarias para pasar de la disposición de las fichas representada por S a la disposición representada por T .

Entrada de ejemplo	Salida para la entrada de ejemplo
BBNBBNBBBB NNNNNBBNNB	3
BNBNB NBNNB	1
BNBN NBNNB	1
B B	0
* *	

Problema C

Circuitos Turísticos

Todo el mundo conoce el archipiélago de Nordenskjold del ártico siberiano, el cual depende del Krai de Krasnoyarsk, integrante de la Federación de Rusia. El archipiélago está formado por N islas y M rutas acuáticas bidireccionales. Cada ruta conecta directamente un determinado par de islas, y para cada par de islas existe a lo sumo una ruta que las conecta.

Dada la popularidad que tiene hoy en día el archipiélago de Nordenskjold, el gobierno de Krasnoyarsk está preocupado por el valor turístico del mismo. El valor turístico del archipiélago está dado por la cantidad de islas que pertenecen al menos a un “circuito turístico”. Un circuito turístico es un recorrido que comienza y termina en la misma isla, tiene al menos 3 islas, no llega más de una vez a la misma isla, y utiliza las rutas acuáticas para pasar de una isla a otra.

El gobierno de Krasnoyarsk quiere saber cuál es la mínima cantidad de rutas adicionales que es necesario construir para que cada isla del archipiélago de Nordenskjold pertenezca al menos a un circuito turístico. Ustedes fueron contratados para responder esa pregunta. Dado que el clima en el archipiélago es extremadamente severo, nuestra recomendación es que hagan ahora mismo un programa que resuelva el problema, y de ese modo eviten viajar hasta allá.

Entrada

Cada caso de prueba se describe utilizando varias líneas. La primera línea contiene dos enteros N y M que indican respectivamente la cantidad de islas y la cantidad de rutas entre ellas ($3 \leq N \leq 100$, $1 \leq M \leq 1000$). Las islas son identificadas por enteros diferentes entre 1 y N . Cada una de las M líneas siguientes describe una ruta distinta utilizando dos enteros diferentes U y V que identifican a las islas conectadas por esa ruta ($1 \leq U < V \leq N$). Asumir que en cada caso de prueba no hay más de una ruta que conecte el mismo par de islas. El final de la entrada se indica con una línea que contiene dos veces el número -1 .

Salida

Para cada caso de prueba, imprimir en la salida una línea conteniendo un entero que representa la mínima cantidad de rutas que es necesario agregar para que cada isla pertenezca al menos a un circuito turístico.

Entrada de ejemplo	Salida para la entrada de ejemplo
3 1	2
1 3	0
9 10	1
1 2	4
2 3	
1 3	
7 9	
5 9	
5 7	
6 8	
4 6	
4 8	
8 9	
4 4	
1 2	
1 4	
1 3	
2 3	
12 9	
1 7	
2 6	
4 9	
9 10	
8 12	
1 5	
1 8	
8 11	
4 10	
-1 -1	

Problema D

Desperdiciando Tiempo

Cierta pareja de jóvenes suele aprovechar el tiempo al máximo. Esto produce que se sobrecarguen de estrés, por lo que decidieron “desperdiciar” un poco de tiempo viendo su serie de televisión favorita.

La serie tiene N temporadas. Cada temporada tiene una cantidad de capítulos que puede variar dependiendo de su éxito, de la disponibilidad de los actores, del equipo de producción y de otros factores externos. Todos los capítulos tienen una duración fija de M minutos.

Para no perder el hilo, antes del comienzo de cada temporada la pareja vuelve a ver compulsivamente todos los capítulos de todas las temporadas anteriores. La preocupación que tienen ahora es si van a desperdiciar demasiado tiempo con este hobby que debería tranquilizarlos. Necesitan la ayuda de ustedes para no volver a caer en el estrés.

Entrada

Cada caso de prueba se describe utilizando dos líneas. La primera línea contiene dos enteros N y M que indican respectivamente la cantidad de temporadas de la serie y la duración en minutos de cada capítulo ($1 \leq N \leq 10^5$, $1 \leq M \leq 10^6$). La segunda línea contiene N enteros C_i que representan la cantidad de capítulos de las sucesivas temporadas en el orden en que son emitidas ($1 \leq C_i \leq 100$ para $1 \leq i \leq N$). El final de la entrada se indica con una línea que contiene dos veces el número -1 .

Salida

Para cada caso de prueba, imprimir en la salida una línea conteniendo un entero que representa la cantidad de minutos que va a utilizar la pareja para ver la serie completa.

Entrada de ejemplo	Salida para la entrada de ejemplo
6 20	9000
24 23 15 22 24 17	10000
1 100	5445000000
100	
10 1000000	
99 99 99 99 99 99 99 99 99 99	
-1 -1	

Problema E

Equilibrio

La media y la mediana muchas veces confunden a los estudiantes por sus nombres similares, pero son dos cosas muy diferentes. En este problema vamos a trabajar con la media y la mediana de un conjunto formado por una cantidad impar de enteros, todos distintos entre sí. En este caso, si N es la cantidad de elementos del conjunto, la media es la suma de los mismos dividido N , mientras que la mediana es aquel elemento del conjunto que tiene $(N - 1)/2$ elementos menores que él y $(N - 1)/2$ mayores. Por ejemplo, si el conjunto es $\{0, 2, 6, 4, 13\}$, la media es 5 mientras que la mediana es 4.

Queremos simplificar la tarea de los estudiantes generando conjuntos “equilibrados”, donde no haya diferencia entre media y mediana. Decimos que un conjunto es equilibrado si está formado por una cantidad impar de enteros, todos distintos entre sí, y la media coincide con la mediana. Por ejemplo, el conjunto $\{0, 2, 6, 4, -2\}$ es equilibrado, ya que tiene $N = 5$ enteros distintos, y tanto la media como la mediana valen 2.

Se sugirió el siguiente procedimiento para obtener conjuntos equilibrados. Se elige cualquier conjunto con una cantidad par de enteros distintos entre sí, y se le agrega otro entero diferente a los que tenía, de manera tal que en el conjunto obtenido la media coincida con la mediana. Para ver si el procedimiento funciona, necesitamos un programa que dados $N - 1$ enteros distintos entre sí, con N impar, nos diga cuántos conjuntos equilibrados de N elementos se pueden obtener usando el procedimiento descripto.

Entrada

Cada caso de prueba se describe utilizando dos líneas. La primera línea contiene un entero impar N que indica la cantidad de elementos que debe tener el conjunto equilibrado ($3 \leq N \leq 499$). La segunda línea contiene $N - 1$ enteros distintos Z_i que representan a los $N - 1$ elementos ya prefijados ($-10^{14} \leq Z_i \leq 10^{14}$ para $1 \leq i \leq N - 1$). El final de la entrada se indica con una línea que contiene el número -1 .

Salida

Para cada caso de prueba, imprimir en la salida una línea conteniendo un entero que representa la cantidad de conjuntos equilibrados que se pueden obtener agregando un elemento al conjunto de entrada.

Entrada de ejemplo	Salida para la entrada de ejemplo
5	3
0 2 6 4	1
7	3
1 2 3 4 5 8	
3	
-1000000000000000 1000000000000000	
-1	

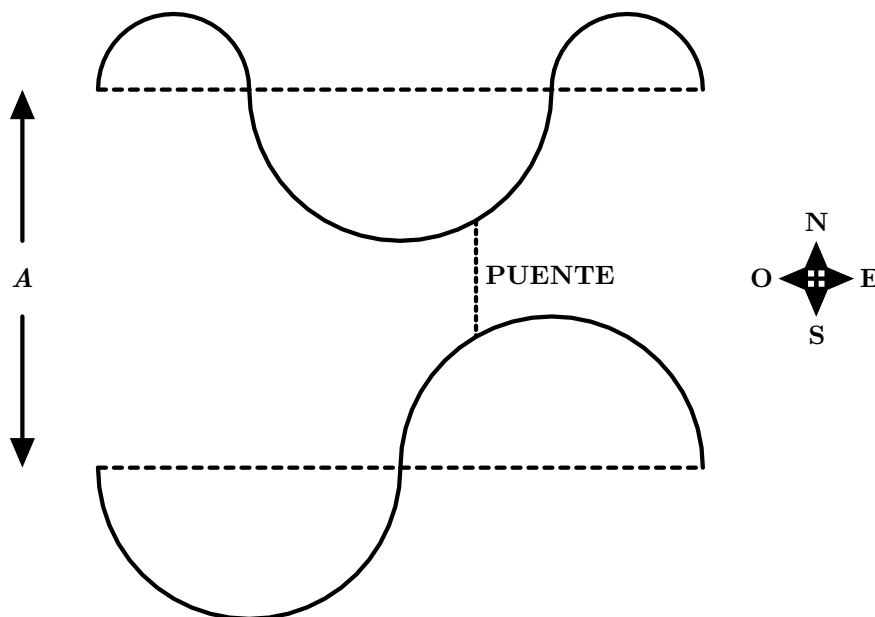
Problema F

Fábrica de Puentes

En el Área Central de Macedonia (ACM) hay un río que corre en dirección oeste-este. Los habitantes de la región quieren construir un puente que atraviese el río en dirección norte-sur, para lo cual contrataron a la fábrica de puentes más barata que conocen, la Industria Constructora de Puentes Cortos (ICPC). Como su nombre lo indica, la constructora se especializa en fabricar los puentes más cortos posibles para cada río.

Lo primero que hizo la ICPC fue modelar las costas (norte y sur) del río. Cada una de ellas se considera formada por semicircunferencias de radios diversos, orientadas alternadamente hacia un lado y hacia el opuesto, con sus centros todos colineales. Las dos rectas que quedan definidas por los centros de las semicircunferencias de cada costa tienen dirección oeste-este, y están a una distancia A una de la otra. Este valor A recibe el nombre de ancho del río. Las costas están alineadas entre sí, es decir, la costa norte no comienza ni termina más al oeste ni más al este que la costa sur.

En la siguiente figura aparece el modelo de cierto río. En la costa norte la semicircunferencia que está más hacia el oeste está orientada hacia el norte, mientras que en la costa sur la semicircunferencia que está más hacia el oeste está orientada hacia el sur. En cada costa las orientaciones de las sucesivas semicircunferencias se alternan. El puente más corto posible en dirección norte-sur es el mostrado.



La tarea de ustedes es ayudar a la constructora determinando la longitud del puente más corto que conecta ambas costas del río del ACM en la dirección norte-sur.

Entrada

Cada caso de prueba se describe utilizando tres líneas. La primera línea contiene un entero A que indica el ancho en metros del río ($3 \leq A \leq 10^4$). La segunda línea describe la costa

norte del río, mientras que la tercera describe la costa sur del mismo.

La descripción de cada costa comienza con un entero C que representa la cantidad de semicircunferencias que modelan esa costa ($1 \leq C \leq 1000$). Luego sigue un caracter “N” o “S” indicando respectivamente que la semicircunferencia que está más hacia el oeste en esa costa está orientada hacia el norte o hacia el sur. La descripción termina con C enteros R_i que representan los radios en metros de las semicircunferencias de esa costa ($1 \leq R_i < A/2$ para $1 \leq i \leq C$). Las semicircunferencias se describen ordenadas desde la que está más hacia el oeste en esa costa (radio R_1) hasta la que está más hacia el este (radio R_C). Las orientaciones de las semicircunferencias se alternan entre norte y sur a partir de la orientación que se indica para la semicircunferencia que está más hacia el oeste. Asumir que en cada caso de prueba las costas están alineadas entre sí, lo que implica que la suma de los radios de las semicircunferencias de cada costa es la misma.

El final de la entrada se indica con una línea que contiene el número -1 .

Salida

Para cada caso de prueba, imprimir en la salida una línea conteniendo un racional que representa la longitud en metros del puente más corto que conecta ambas costas del río en la dirección norte-sur. Redondear el resultado al racional más cercano con 2 dígitos decimales. En caso de empates, redondear hacia arriba. Siempre utilizar exactamente 2 dígitos luego del punto decimal, incluso si eso significara terminar con un cero.

Entrada de ejemplo	Salida para la entrada de ejemplo
5	1.54
3 N 1 2 1	3.00
2 S 2 2	1.00
3	
1 N 1	
1 S 1	
3	
1 S 1	
1 N 1	
-1	

Problema G

Ganando Tiempo

Graciela maneja un micro escolar. Lamentablemente, día a día es objeto de burlas por ser una de las pocas mujeres que se dedica a esa profesión. Para mejorar su imagen ha decidido agregar a su prudencia en el manejo, eficiencia en el tiempo de viaje. La idea de Graciela es realizar su recorrido en el menor tiempo posible, pero sin transgredir las normas de tránsito.

El micro que maneja Graciela tiene un sistema muy moderno que le permite ajustar la aceleración a cualquier número real instantáneamente. Por lo tanto, la aceleración es constante por intervalos, saltando a otra aceleración cuando Graciela así lo decide. Si v es la velocidad del micro en un momento dado y a su aceleración que se mantiene constante durante un intervalo de tiempo t , entonces la velocidad al final del intervalo será $v + at$. Además, el micro habrá avanzado una distancia $at^2/2 + vt$ durante ese intervalo de tiempo.

Las normas del Tratado Argentino de Interacción con Pedales (TAIP) impiden que un micro utilice una aceleración mayor a A o una desaceleración mayor a D , es decir, que los valores permitidos para la aceleración son únicamente los números reales entre $-D$ y A , inclusive. Además, el tratado establece la existencia de puntos de control, en los cuales la velocidad de un micro no puede estar por debajo de cierta velocidad mínima ni por arriba de cierta velocidad máxima.

Graciela conoce los puntos de control que aparecen en el recorrido que realiza, la longitud del mismo, y las constantes A y D . Al inicio del recorrido, tanto la velocidad como la aceleración del micro son 0. No hay restricciones adicionales respecto de la velocidad ni de la aceleración que puede tener el micro al finalizar su recorrido (en particular, no es obligatorio que frene). El trabajo de ustedes es utilizar esos datos para determinar el mínimo tiempo que necesita Graciela para realizar su recorrido apegándose a las normas del TAIP.

Entrada

Cada caso de prueba se describe utilizando varias líneas. La primera línea contiene cuatro enteros N , L , A y D . El valor N representa la cantidad de puntos de control que hay en el recorrido ($1 \leq N \leq 10^5$). El valor L indica la longitud del recorrido ($2 \leq L \leq 10^7$). Los valores A y D representan respectivamente la aceleración y desaceleración máximas permitidas según fue explicado antes ($1 \leq A, D \leq 100$). Cada una de las N líneas siguientes describe un punto de control diferente utilizando tres enteros X , V y W que representan respectivamente la distancia entre el punto de control y el inicio del recorrido ($1 \leq X \leq L - 1$), la velocidad mínima, y la velocidad máxima de un micro que pasa por ese punto ($1 \leq V \leq W \leq 100$). Asumir que en cada caso de prueba los puntos de control aparecen ordenados de manera estrictamente creciente de acuerdo a su distancia desde el inicio del recorrido. Las longitudes están expresadas en m, las velocidades en m/s, y las aceleraciones en m/s². El final de la entrada se indica con una línea que contiene cuatro veces el número -1 .

Salida

Para cada caso de prueba, imprimir en la salida una línea conteniendo un racional que representa el mínimo tiempo (en segundos) que se necesita para realizar el recorrido sin infringir ninguna de las reglas, o un asterisco (“*”) si es imposible lograrlo. Redondear el resultado al racional más cercano con 2 dígitos decimales. En caso de empates, redondear hacia arriba. Siempre utilizar exactamente 2 dígitos luego del punto decimal, incluso si eso significara terminar con un cero.

Entrada de ejemplo	Salida para la entrada de ejemplo
1 40 10 1	*
20 21 21	2.83
1 40 10 5	2.00
20 20 20	35.96
1 20 10 50	
10 14 15	
5 1000 2 5	
400 30 80	
600 35 50	
700 10 30	
900 30 40	
950 10 30	
-1 -1 -1 -1	

Problema H

Holgazán

Humberto es muy holgazán. Es tan perezoso que hasta le da fiaca ser vago. No es extraño que casi no haya trabajado durante toda su vida, pero tuvo suerte y ganó mucho dinero en la lotería. Con parte de ese dinero compró un campo vacío e hizo construir varias casas dentro del mismo, a donde se mudaron Humberto junto con sus familiares y amigos. Es así que Humberto puede ir en línea recta de cualquier punto a cualquier otro del campo, sin tener que caminar de más por culpa del diseño urbano (consideramos insignificante rodear un árbol, una casa o cualquier otro elemento que se interponga en el camino de Humberto).

Lamentablemente, la situación de Humberto es de una holgazanería tal que esto no le bastó. Para caminar aún menos, hizo instalar catapultas en ciertos lugares estratégicos del campo. Cada catapulta puede enviarlo desde el punto donde está instalada a cualquier punto a una cierta distancia fija que depende de la fuerza de la catapulta.

La fortuna por fin parecía sonreírle a Humberto, hasta que se le presentó un nuevo problema. Caminar en línea recta era muy fácil, mientras que ahora con las catapultas los recorridos son más complicados. Humberto ya está cansado de tantos problemas. Como todavía tiene mucho dinero, los contrató a ustedes para que le digan la mínima distancia que tiene que caminar para movilizarse de un punto a otro del campo.

Entrada

Cada caso de prueba se describe utilizando varias líneas. La primera línea contiene cinco enteros N , P_X , P_Y , L_X y L_Y . El valor N representa la cantidad de catapultas que hay en el campo ($1 \leq N \leq 100$). El par (P_X, P_Y) indica el punto de partida en el plano XY del recorrido de Humberto, mientras que el par (L_X, L_Y) indica análogamente el punto de llegada ($1 \leq P_X, P_Y, L_X, L_Y \leq 10^9$). Cada una de las N líneas siguientes describe una catapulta diferente utilizando tres enteros C_X , C_Y y F , los cuales indican que en el punto (C_X, C_Y) hay instalada una catapulta de fuerza F ($1 \leq C_X, C_Y, F \leq 10^9$). Esto significa que Humberto se puede transportar, sin necesidad de caminar, desde el punto (C_X, C_Y) hasta cualquier punto que esté a una distancia de *exactamente* F del punto (C_X, C_Y) . Asumir que en cada caso de prueba todos los puntos informados son distintos (el punto de llegada es distinto al punto de partida, no hay catapultas en ninguno de los dos, y no hay dos catapultas en el mismo punto). El final de la entrada se indica con una línea que contiene cinco veces el número -1 .

Salida

Para cada caso de prueba, imprimir en la salida una línea conteniendo un racional que representa la mínima distancia que Humberto tiene que caminar para ir desde el punto de partida hasta el punto de llegada, utilizando eventualmente las catapultas. Redondear el resultado al racional más cercano con 2 dígitos decimales. En caso de empates, redondear hacia arriba. Siempre utilizar exactamente 2 dígitos luego del punto decimal, incluso si eso significara terminar con un cero.

Entrada de ejemplo	Salida para la entrada de ejemplo
1 10 10 20 10	1.00
11 10 9	10.00
1 20 10 10 10	4.00
11 10 9	10.41
2 1 1 12 1	5.11
3 1 6	
8 1 5	
1 12 12 1 1	
6 6 9	
5 10 10 1 1	
3 3 7	
8 3 7	
8 8 7	
3 8 7	
5 5 5	
-1 -1 -1 -1 -1	

Problema I

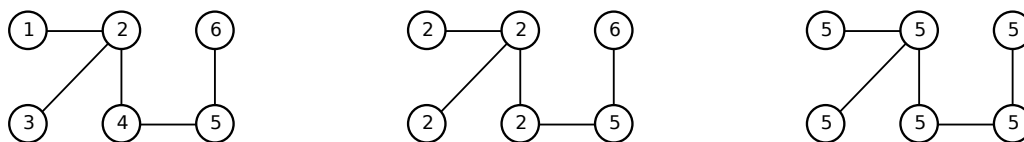
Imperialismo

La sed de conquista es algo común en la historia de la humanidad. . . y del universo entero.

En el planeta Imperius hay actualmente N fortalezas que fueron contruídas de a una por vez. Cada fortaleza en el momento de ser construída recibió un número diferente entre 1 y N , según su orden cronológico de construcción. Además, cada fortaleza (excepto la primera) fue conectada a exactamente una de las fortalezas ya existentes por un camino directo que servía de vía comercial. Esto era suficiente para que cualquier fortaleza pudiera comerciar con cualquier otra, eventualmente pasando por otras fortalezas intermedias.

El planeta era cada vez más próspero y pacífico, hasta que se dejaron de construir fortalezas. En ese momento, N imperios diferentes se formaron, y cada uno dominó una fortaleza diferente. De manera previsible, la sed de conquista se apoderó de Imperius. Es así que cada año, exactamente uno de los imperios sobrevivientes conquista a los imperios vecinos a él, y pasa a dominar todas las fortalezas que eran dominadas por ellos. Dos imperios son vecinos si existen al menos dos fortalezas, una dominada por un imperio y la otra por el otro, tales que se construyó un camino directo entre ambas fortalezas.

En la parte izquierda de la siguiente figura aparece un posible escenario en el cual 6 fortalezas están dominadas inicialmente por 6 imperios diferentes. Para cada fortaleza se indica con un número entre 1 y 6 el imperio que la domina. Los caminos construídos entre fortalezas están representados por líneas. Si el primer año el imperio 2 conquista a sus vecinos, la situación sería la mostrada en la parte central de la figura. Si luego de eso, en el segundo año el imperio 5 conquista a sus vecinos, todas las fortalezas pasarían a estar dominadas por este imperio, como se ve en la parte derecha de la figura.



Notar que los imperios conquistados desaparecen. Como cada año desaparece al menos un imperio, al cabo de cierta cantidad de años un único imperio dominará todas las fortalezas. Se pide encontrar la mínima cantidad de años en que eso puede ocurrir.

Entrada

Cada caso de prueba se describe utilizando dos líneas. La primera línea contiene un entero N que representa la cantidad de fortalezas ($2 \leq N \leq 10^4$). La segunda línea contiene $N - 1$ enteros P_i indicando que al construir la $(i + 1)$ -ésima fortaleza, ésta fue conectada por un camino directo a la fortaleza P_i ($1 \leq P_i \leq i$ para $1 \leq i \leq N - 1$). El final de la entrada se indica con una línea que contiene el número -1 .

Salida

Para cada caso de prueba, imprimir en la salida una línea conteniendo un entero que representa la mínima cantidad de años luego de los cuales es posible que un único imperio domine todas las fortalezas.

Entrada de ejemplo	Salida para la entrada de ejemplo
6	2
1 2 2 4 5	2
7	1
1 1 3 3 4 4	
6	
1 2 2 2 2	
-1	

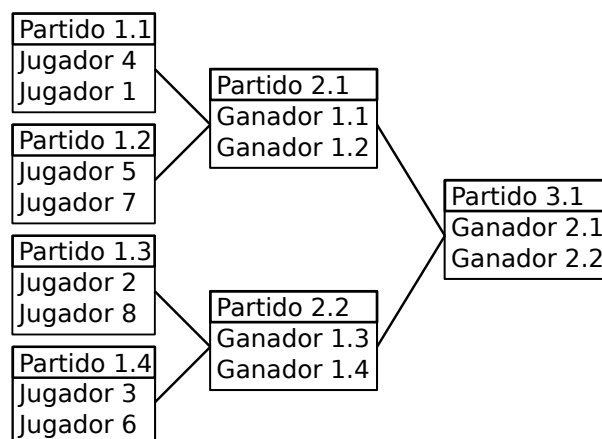
Problema J

Juego, Set y Partido

La Asociación de Clubes Modernos (ACM) organiza todos los años un torneo de CompuTenis, un deporte especialmente adaptado a un público sin estado físico mensurable. Las reglas del CompuTenis son muy complejas (basta decir que involucran programar con el código pegado a la oreja), pero afortunadamente no es necesario conocer sus detalles para resolver este problema. Sólo hace falta saber que en un partido de CompuTenis compiten dos jugadores, ganando el partido aquel que gane primero S sets; a su vez, cada set está formado por varios juegos, y para ganar un set se requiere vencer en al menos J juegos, con una diferencia de al menos D juegos ganados más que el oponente.

El torneo que organiza la ACM tiene K rondas, y en él participan $N = 2^K$ jugadores. Todos los jugadores participan de la primera ronda. En cada ronda cada jugador que todavía está en competencia se enfrenta en un único partido contra otro jugador que también sigue en carrera. El ganador de cada partido pasa a la siguiente ronda, mientras que el perdedor queda automáticamente descalificado. El ganador del único partido de la K -ésima ronda es el ganador del torneo.

La ACM desea hacer un torneo lo más largo posible, ya que los partidos son televisados y la ACM cobra por cada minuto de emisión. Dado cualquier par de jugadores distintos, la ACM conoce la probabilidad de que uno de los jugadores le gane un juego al otro. Ustedes son miembros del comité organizador de la ACM. La tarea que tienen es organizar los partidos de cada ronda a fin de maximizar la cantidad esperada de juegos en el torneo. Organizar los partidos de la primera ronda significa indicar los pares de jugadores que van a enfrentarse en cada partido. Organizar los partidos de cada una de las otras rondas significa indicar los pares de partidos de la ronda anterior cuyos ganadores van a enfrentarse en cada partido de la nueva ronda; esto sólo puede depender de la identificación de los partidos de la ronda anterior. La siguiente figura muestra una posible organización para $K = 3$ rondas con $N = 2^K = 8$ jugadores.



Por el momento, la ACM únicamente quiere estimar su ganancia, de modo que basta que ustedes digan la máxima cantidad esperada de juegos que puede tener el torneo.

Entrada

Cada caso de prueba se describe utilizando varias líneas. La primera línea contiene cuatro enteros K , S , J y D . El valor K representa la cantidad de rondas en el torneo ($1 \leq K \leq 3$). El valor S indica la cantidad de sets que es necesario ganar para ganar un partido ($1 \leq S \leq 10$). El valor J es la cantidad mínima de juegos que es necesario ganar para ganar un set, mientras que el valor D indica que un jugador debe ganar esa cantidad de juegos más que su oponente para ganar el set ($1 \leq D \leq J \leq 100$). Los jugadores son identificados por enteros diferentes entre 1 y $N = 2^K$. Cada una de las N líneas siguientes contiene N valores. En la i -ésima línea, el j -ésimo valor es un entero P_{ij} tal que $p_{ij} = P_{ij}/100$ es la probabilidad de que el jugador i le gane un juego al jugador j ($0 \leq P_{ij} \leq 100$ para $1 \leq i, j \leq N$). Asumir que $P_{ii} = 0$ (ya que un jugador no juega contra sí mismo), mientras que $P_{ij} + P_{ji} = 100$ para $i \neq j$. El final de la entrada se indica con una línea que contiene cuatro veces el número -1 .

Salida

Para cada caso de prueba, imprimir en la salida una línea conteniendo un racional que representa la máxima cantidad esperada de juegos que puede tener el torneo correspondiente. Redondear el resultado al racional más cercano con 2 dígitos decimales. En caso de empates, redondear hacia arriba. Siempre utilizar exactamente 2 dígitos luego del punto decimal, incluso si eso significara terminar con un cero.

Entrada de ejemplo	Salida para la entrada de ejemplo
1 1 2 2	4.00
0 50	224.08
50 0	
3 3 6 2	
0 88 2 76 71 24 50 4	
12 0 54 37 84 95 88 98	
98 46 0 66 36 13 33 33	
24 63 34 0 29 21 96 63	
29 16 64 71 0 0 47 13	
76 5 87 79 100 0 56 89	
50 12 67 4 53 44 0 23	
96 2 67 37 87 11 77 0	
-1 -1 -1 -1	