Maratona de Programação da SBC 2024

Sub-Regional Brasil do ICPC

31 de Agosto de 2024

Caderno de Problemas

Informações Gerais

Este caderno contém 12 problemas; as páginas estão numeradas de 1 a **??**, não contando esta página de rosto. Verifique se o caderno está completo.

Este conjunto de problemas também está sendo utilizado simultaneamente nas seguintes competições: The 2024 ICPC Gran Premio de Centroamerica, The 2024 ICPC Gran Premio de Mexico e Competencia Boliviana de Programación.

A) Sobre os nomes dos programas

1) Para soluções em C/C++ e Python, o nome do arquivo-fonte não é significativo, pode ser qualquer nome.
2) Se sua solução é em Java, ela deve ser chamada *codigo_de_problema*`.java` onde *codigo_de_problema* é a letra maiúscula que identifica o problema. Lembre que em Java o nome da classe principal deve ser igual ao nome do arquivo.
3) Se sua solução é em Kotlin, ela deve ser chamada *codigo_de_problema*`.kt` onde *codigo_de_problema* é a letra maiúscula que identifica o problema. Lembre que em Kotlin o nome da classe principal deve ser igual ao nome do arquivo.

B) Sobre a entrada

1) A entrada de seu programa deve ser lida da *entrada padrão*.
2) A entrada é composta de um único caso de teste, descrito em um número de linhas que depende do problema.
3) Quando uma linha da entrada contém vários valores, estes são separados por um único espaço em branco; a entrada não contém nenhum outro espaço em branco.
4) Cada linha, incluindo a última, contém exatamente um caractere final-de-linha.
5) O final da entrada coincide com o final do arquivo.

C) Sobre a saída

1) A saída de seu programa deve ser escrita na *saída padrão*.
2) Quando uma linha da saída contém vários valores, estes devem ser separados por um único espaço em branco; a saída não deve conter nenhum outro espaço em branco.
3) Cada linha, incluindo a última, deve conter exatamente um caractere final-de-linha.

Promoção:

Sociedade Brasileira de Computação

v1.0

Problema A
Atenção à Reunião

Vinicius está em uma reunião da diretoria do Instituto de Consultoria de Palestras e Comentários (ICPC) pensando que seria muito bom se os membros da diretoria fossem mais concisos, e mantivessem suas falas dentro do tempo definido para cada diretor, para que a reunião terminasse antes do almoço. Infelizmente, talvez devido à natureza da instituição, todos gostam muito de falar.

Sabendo que

- há N diretores que irão falar na reunião;
- cada diretor irá falar pelo mesmo tempo;
- e entre duas falas consecutivas há um intervalo de 1 minuto,

determine a duração máxima de cada fala, em minutos, para que a reunião dure no máximo K minutos.

Entrada

A primeira linha contém um inteiro N ($1 \leq N \leq 100$), o número de diretores. A segunda linha contém um inteiro K ($1 \leq K \leq 1000$ e $K \geq N$), a duração máxima da reunião em minutos. Para todos os casos de entrada, a fala de cada diretor tem duração de pelo menos 1 minuto.

Saída

Seu programa deve produzir uma única linha, contendo um único inteiro, indicando a duração da fala de cada membro da diretoria, em minutos.

Exemplo de entrada 1	**Exemplo de saída 1**
7 120	16

Explicação do exemplo 1:

Há 7 diretores e a duração máxima da reunião é 120 minutos. Se cada diretor falar por 16 minutos, temos $16 \times 7 = 112$ minutos. Como há seis intervalos entre as falas, e cada intervalo tem duração de um minuto, ao todos temos 118 minutos. Note que, nesse caso, dois minutos do tempo da reunião não são utilizados, e que, se as falas fossem superiores a 16 minutos, o tempo total superaria o limite de 120 minutos.

Exemplo de entrada 2	**Exemplo de saída 2**
1 10	10

Explicação do exemplo 2:

Há apenas um diretor e a duração da reunião é 10 minutos. Assim, a duração máxima da fala do diretor é 10 minutos.

Exemplo de entrada 3	**Exemplo de saída 3**
100 1000	9

Problema B
Bacon Number

Carlinhos adora filmes, e recentemente tem estado fascinado com o número de Bacon, mais conhecido como *Bacon Number*, que é definido da seguinte forma.

- O número de Bacon do ator Kevin Bacon é igual a 0;
- Se o menor número de Bacon de um ator com quem X tenha aparecido em um mesmo filme for b, o número de bacon do ator X é $b + 1$.

Ou seja, o número de Bacon mede o menor caminho entre qualquer ator e o ator Kevin Bacon, em que dois atores são conectados se eles apareceram juntos em um mesmo filme.

Carlinhos está interessado em um problema mais geral: dados dois atores, como conectá-los através de filmes e atores intermediários? São dados N filmes, e, para cada filme, quais dos M atores existentes atuaram nele. Carlinhos quer responder Q consultas: na i-ésima delas, queremos computar alguma forma de conectar o ator x_i com o ator y_i. Devemos achar alguma sequência $x_i = a_1, f_1, a_2, f_2, \ldots, f_{k-1}, a_k = y_i$, em que $1 \leq a_j \leq N$ são atores e $1 \leq f_j \leq M$ são filmes, e o ator a_j atuou nos filmes f_{j-1} e f_j, ou indicar que não existe tal sequência.

Entrada

Na primeira linha da entrada, são fornecidos dois inteiros N ($1 \leq N \leq 100$) e M ($1 \leq M \leq 10^4$), o número de filmes e o número de atores. Seguem N linhas. Na i-ésima delas, o primeiro inteiro n_i ($1 \leq n_i \leq M$) denota o número de atores no filme i. Seguem n_i números em ordem crescente separados por espaço: os índices, de 1 a M, dos atores que atuaram no filme i. Na próxima linha, leia um número Q ($1 \leq Q \leq 10^4$): o número de consultas. As próximas Q linhas descrevem as consultas. Na i-ésima delas, leia dois números x_i, y_i ($1 \leq x_i \neq y_i \leq M$), os atores que queremos conectar.

Saída

Para cada uma das consultas, se não existe sequência, imprima -1. Caso contrário, imprima duas linhas. Na primeira, o número de atores k_i ($2 \leq k_i \leq 10^5$) em alguma maneira de conectar x_i e y_i. Na segunda, imprima a sequência como descrita, com k_i atores e $k_i - 1$ filmes, de maneira alternada. Se houver mais de uma maneira de conectar os atores, imprima qualquer uma delas.

Exemplo de entrada 1	**Exemplo de saída 1**

`4 6`
`3 1 2 5`
`3 1 3 5`
`2 2 4`
`1 6`
`4`
`1 5`
`1 4`
`3 4`
`1 6`
	`2`
	`1 1 5`
	`3`
	`1 1 2 3 4`
	`4`
	`3 2 1 1 2 3 4`
	`-1`

Problema C

Casal do BipBop

É hora de Bob e Charlie entrarem em um novo hiperfoco de casal: dancinhas de BipBop. Essa rede social especializada em vídeos curtos está viralizando mais do que nunca. Como consequência, agora os casais medem o quanto se amam em termos do quão bem conseguem dançar juntos. Em teoria, o estilo de dança BipBop é simples e pode ser usado para praticamente qualquer música. Geralmente, consiste em uma sequência de movimentos, um para cada verso, representado por um número inteiro, já que, francamente, seus movimentos são bem genéricos.

Sempre atrasados, os namorados acabaram de chegar a uma festa. A música já está tocando, mas eles ainda querem impressionar e mostrar que conseguem dançar BipBop mesmo sem saber em qual verso a música realmente está. Então, cada um começa a dançar em um verso aleatório e continua seguindo a coreografia até que algum deles atinja o final da coreografia ou até que descoordenem algum movimento (cada um execute um movimento diferente).

Não existe uma música popular que Bob e Charlie não saibam dançar, então, dada uma música representada por uma sequência de movimentos, um para cada verso, calcule o número esperado de versos em que eles estarão dançando em sincronia, se cada um deles inicialmente acha que a música está tocando em um verso aleatório com probabilidade uniforme.

Entrada

Na primeira linha, leia um inteiro N ($1 \leq N \leq 10^5$), o número de movimentos da sequência dada. Na segunda linha, leia N inteiros $V_1, V_2, \ldots, V_N$ ($1 \leq V_i \leq N$), correspondentes ao movimento associado a cada um dos versos na sequência .

Saída

Imprima o número esperado de versos (movimentos) que o casal dançará em sincronia, se cada um deles escolher uma posição uniformemente aleatória. Imprima a resposta como uma fração irredutível P/Q, em que $\gcd(P, Q) = 1$. Pode ser provado que sempre é possível expressar a resposta dessa forma.

Exemplo de entrada 1	Exemplo de saída 1
2 1 1	5/4

Explicação do exemplo 1:

Note que há 4 maneiras equiprováveis da coreografia ocorrer: tanto Bob quando Charlie podem começar no primeiro ou no segundo verso, com probabilidade $1/2$ de cada um começar em cada um dos versos e portanto probabilidade $1/4$ para cada uma das combinações. Caso ambos comecem no primeiro verso, eles dançarão 2 versos em sincronia. Nas outras três possibilidades, dançarão apenas um verso em sincronia. Assim, temos em média, $2 \times 1/4 + 1 \times 1/4 + 1 \times 1/4 + 1 \times 1/4 = 5/4$ versos em sincronia.

Exemplo de entrada 2	Exemplo de saída 2
4 1 1 1 1	15/8

Exemplo de entrada 3	Exemplo de saída 3
7 1 2 1 3 1 2 1	48/49

Problema D
Desafio do Chefão

Fulano, um jogador ávido, se deparou com um desafio épico no jogo online "Desafio do Chefão". O objetivo é derrotar um chefão poderoso, cujo poder é descrito por um conjunto de runas ancestrais. Essas runas representam um número binário gigante N, indicando a força total do inimigo.

Para vencer o chefão, Fulano dispõe de M magias diferentes, e o objetivo é reduzir a força total do inimigo a zero utilizando essas magias. A i-ésima magia está associada a dois inteiros a_i e b_i. Ao ser lançada, a magia i reduz o valor de N em a_i unidades. Tal magia pode ser lançada quantas vezes o jogador quiser, desde que duas condições específicas sejam atendidas:

- O valor de a_i deve ser menor ou igual ao valor atual de N.
- O valor atual de N deve ser divisível por 2^{b_i}. Em outras palavras, a magia i só pode ser utilizada se os últimos b_i dígitos de N forem zeros.

Fulano está fascinado pelo jogo e quer descobrir de quantas maneiras diferentes ele pode combinar as magias para reduzir o número binário N a exatamente zero e, assim, derrotar o chefão. Duas combinações são consideradas diferentes se a sequência em que as magias são lançadas for diferente.

Como o número de combinações possíveis pode ser muito grande, a resposta deve ser dada módulo $10^9 + 7$.

Ajude Fulano a encontrar a resposta!

Entrada

A primeira linha contém um único número inteiro N ($1 \leq N \leq 10^{18}$), representando o poder do chefão.

A segunda linha contém um único número inteiro M ($1 \leq M \leq 10^5$), denotando o número de magias disponíveis.

As próximas M linhas contêm as descrições das magias: a i-ésima destas linhas contém dois números a_i ($1 \leq a_i \leq 100$) e b_i ($0 \leq b_i \leq 60$).

Saída

Imprima um único número inteiro: o número de sequências de usos de magias admissíveis (tomados módulo $10^9 + 7$), que reduzem o poder do chefão de N para 0.

Exemplo de entrada 1	**Exemplo de saída 1**

`6 2`
`1 0`
`2 1`
	`8`

Exemplo de entrada 2	**Exemplo de saída 2**

`9 5`
`1 0`
`1 1`
`4 3`
`1 1`
`8 0`
	`92`

Problema E

Estojo de Joias

A princesa da Nlogônia guarda sua coleção de pérolas em um estojo de jóias quadrado composto de N colunas, cada coluna com N caixinhas. Ela coloca um número diferente de pérolas em cada caixinha, e organiza o estojo de forma que em cada coluna, de cima para baixo, as caixinhas contenham números crescentes de pérolas e que em cada linha, da esquerda para a direita, as caixinhas também contenham números crescentes de pérolas.

A princesa desconfia que sua irmã pequena, que é muito sapeca, esteja mexendo nas suas coisas em suas brincadeiras. Em particular, a princesa desconfia que seu estojo de jóias tenha sido rotacionado de 90 graus no sentido horário, possivelmente múltiplas vezes.

A figura (a) abaixo mostra um exemplo da organização original de um estojo 4×4. A figura (b) mostra o estojo rotacionado no sentido horário, de 90 graus, uma vez.

3	4	6	12
7	10	11	19
9	14	17	22
15	16	20	25

(a)

90°

15	9	7	3
16	14	10	4
20	17	11	6
25	22	19	12

(b)

Dados os números de pérolas em cada caixinha, escreva um programa para determinar qual o menor número de rotações de 90 graus no sentido anti-horário que são necessárias para fazer com que o estojo de jóias volte para o estado original.

Entrada

A primeira linha da entrada contém um inteiro N, o número de linhas e colunas do estojo ($2 \leq N \leq 50$). Cada uma das N linhas seguintes contém N inteiros $K_{i,j}$, a quantidade de pérolas da caixinha na linha i e coluna j ($0 \leq K_{i,j} \leq 10^5$, para $1 \leq i \leq N$ e $1 \leq j \leq N$). Na entrada, as linhas são dadas de cima para baixo, e as colunas são dadas da esquerda para a direita.

Saída

Seu programa deve produzir uma única linha contendo apenas um inteiro R (que pode ser 0, 1, 2 ou 3), o menor número de vezes que o estojo de jóias deve ser rotacionado no sentido anti-horário para retornar ao estado original.

Exemplo de entrada 1	Exemplo de saída 1
4 15 9 7 3 16 14 10 4 20 17 11 6 25 22 19 12	1

Explicação do exemplo 1:

Este exemplo corresponde ao exemplo do enunciado. É necessário rotacionar o estojo no sentido anti-horário uma vez.

Exemplo de entrada 2	**Exemplo de saída 2**

`3`
`300 250 150`
`280 200 140`
`240 190 130`
	`2`

Explicação do exemplo 2:

É necessário rotacionar o estojo no sentido anti-horário duas vezes.

Exemplo de entrada 3	**Exemplo de saída 3**

`2`
`2 4`
`1 3`
	`3`

Explicação do exemplo 3:

É necessário rotacionar o estojo no sentido anti-horário três vezes.

Problema F
Frações Contínuas

O pequeno Charles era um dos melhores programadores competitivos do mundo. No entanto, ele nunca gostou muito de programar. Agora que está aposentado, ele pode dedicar seus estudos ao que realmente ama: frações contínuas.

Para se preparar para a próxima Imensa Competição de Phrações Contínuas (ICPC), ele precisa resolver o seguinte problema:

Defina $p_0 = 1$ como a fração de nível 0. Em seguida, defina:

$$p_1 = \frac{1}{1+1}$$

como a fração de nível 1, p_1. Além disso, defina

$$p_2 = \frac{1}{1 + \frac{1}{1+1}}$$

como a fração de nível 2, p_2, e assim por diante.

Dado um valor inteiro N, ajude Charles a determine o valor do numerador da fração p_N.

Entrada

A primeira e única linha contém um inteiro N ($1 \leq N \leq 40$).

Saída

O valor p_N pode ser escrito como uma fração da forma $\frac{a}{b}$, onde a e b são coprimos. Imprima uma linha contendo o valor de a.

Exemplo de entrada 1	**Exemplo de saída 1**
2	2

Exemplo de entrada 2	**Exemplo de saída 2**
10	89

Problema G
Geografia dos Rios

Ao estudar a geografia dos rios do mundo, você se pergunta: quando dois rios se unem, quem escolhe qual vai ser o nome do rio a partir dessa junção? De fato, a resposta é simples: com a união de dois rios, o nome passa a ser o nome daquele que tinha maior volume de água. Dado que todos os rios eventualmente se unem e desaguam no mar, um problema interessante é calcular, dado o nome de cada nascente, o nome do rio final que desagua no mar.

Formalmente, são dadas N nascentes de rios. Para cada nascente, você tem uma quantidade de litros de água l_i que nasce dela. Além disso, pares de rios se encontram (como uma árvore binária), até todos se juntarem e desaguarem no mar. Quando dois rios se encontram, a quantidade de litros de água é somada, e o nome do rio passa a ser o nome do rio que tinha mais água, ou, em caso de empate, o de menor índice. O nome inicial de cada nascente é seu índice.

O que você quer saber é o nome do rio que por fim deságua no mar. Porém, estamos em época de chuvas! Você precisa processar Q operações. Em cada uma delas, uma chuva aconteceu que causa com que em uma nascente n_i agora nasce q_i litros **a mais** de água (e isso será mantido para as operações futuras). Depois de cada operação, calcule o nome do rio que desagua no mar.

Entrada

A primeira linha contém um inteiro N ($1 \leq N \leq 10^5$): o número de nascentes.

A segunda linha contém N inteiros l_i ($1 \leq l_i \leq 10^9$): o número de litros de água que nascem na nascente i.

As $N - 1$ linhas seguintes descrevem como os rios se unem. Na i-ésima delas, dois inteiros a_i, b_i ($1 \leq a_i, b_i < N + i$) indicam que os rios a_i e b_i se unem para formar o rio $N + i$ (cujo volume de água será a soma dos volumes de a_i e b_i, e nome será o nome do que tiver mais volume de água). É garantido que os valores a_i e b_i são válidos, isto é, $a_i \neq b_i$ e nenhum deles já foi unido previamente na entrada.

A próxima linha contém um inteiro Q ($1 \leq Q \leq 10^5$), o número de operações.

Seguem Q linhas com as operações: a i-ésima linha contém dois inteiros n_i e q_i ($1 \leq n_i \leq N$ e $1 \leq q_i \leq 10^9$), significando que na nascente representada pelo ponto n_i agora nasce q_i litros **a mais** de água.

Saída

Imprima, na primeira linha, o nome do rio que inicialmente desagua no mar. Em seguida, imprima Q linhas: após cada operação, o nome do rio que desagua no mar.

Exemplo de entrada 1	**Exemplo de saída 1**
3 1 4 4 1 2 4 3 2 3 2 1 2	2 3 2

Problema H
Harmônicos Interferentes

A transmissão de mensagens por meios eletromagnéticos apresenta diversos desafios, como interferência de outros sinais, naturais ou artificiais, que podem corromper uma transmissão.

Uma estratégia comum é o envio de informações adicionais que permitam validar uma mensagem recebida. Alguns protocolos mais robustos permitem até mesmo corrigir alguns erros da mensagem enviada.

Arthur e Bruna estão testando um novo protocolo de transmissão em um dispositivo que eles desenvolveram. Uma mensagem M, que é uma sequência de bits, é enviada de Arthur para Bruna, juntamente com uma sequência de controle N, também representada como uma sequência de bits. Ao compor a mensagem M e escolher os bits de N, Arthur se certifica que o inteiro codificado por M seja divisível pelo inteiro representado por N.

Para cada bit recebido por Bruna, caso o bit tenha sido transmitido sem problemas, o valor `0` ou `1` será armazenado no dispositivo receptor. Caso tenha havido alguma interferência, o símbolo `*` é armazenado no lugar do bit. O resultado da transmissão será armazenado no par (M', N').

Após a comunicação, caso a mensagem tenha sido enviada com sucesso, Bruna consegue decodificar a mensagem M original (pois $M = M'$). Caso tenha havido algum problema, por conta da forma como o protocolo funciona, pode ainda ser possível decodificar a mensagem. Caso muitos bits tenham sido perdidos, Bruna simplesmente descarta a mensagem. Mas para transmissões onde no máximo 16 bits do par (M, N) original tenham sido perdidos, Bruna gostaria de tentar recuperar a mensagem, evitando restransmissões. Ela precisa de sua ajuda para recuperar uma das possiveis mensagens codificadas pelo par (M', N') recebido.

Por exemplo, suponha que Bruna tenha recebido M'=`111*` e N'=`1*`. Duas transmissões poderiam ter sido realizadas:

1. M=`1111` com N=`11`. Neste caso, os números 15 e 3 estão representados em M e N, respectivamente.
2. M=`1110` com N=`10`. Neste caso, os números 14 e 2 estão representados em M e N, respectivamente.

Sua tarefa é: dadas as representações das informações recebidas, encontrar uma mensagem M que possa ter sido enviada por Arthur. Caso mais de uma mensagem exista, você pode imprimir qualquer mensagem que possa ter sido transmitida por Arthur.

Entrada

A primeira linha da entrada conterá uma sequência de caracteres representando M', com $1 \leq |M'| \leq 500$. A segunda linha da entrada conterá uma sequência de caracteres representando N', com $1 \leq |N'| \leq 16$. Todos os caracteres de N' e M' serão `0`, `1` ou `*`. No total, nunca haverá mais de 16 caracteres `*` na entrada.

Saída

Uma única linha deve ser impressa, contendo uma mensagem M, compatível com a informação recebida por Bruna.

Exemplo de entrada 1	Exemplo de saída 1
`111*` `1*`	`1111`

Explicação do exemplo 1:

Este caso corresponde ao exemplo fornecido no enunciado.

Exemplo de entrada 2	**Exemplo de saída 2**
`101**` `11`	`10101`

Explicação do exemplo 2:

Neste caso, as diferentes formas de se escolher os bits desconhecidos resultariam em mensagens correspondentes aos inteiros 20, 21, 22 e 23, e apenas 21, representado por `10110`, é divisível por 3.

Problema I
Ingredientes Alergênicos

Na Nlogonia, as comidas são identificadas por números. Os números primos identificam os ingredientes básicos, e o número que identifica cada comida é dado pelo produto dos números associados aos ingredientes que a compõem, respeitando multiplicidades. Por exemplo, uma comida de número 12 contém duas unidades do ingrediente 2, e uma unidade do ingrediente 3, já que $12 = 2 \cdot 2 \cdot 3$.

Você mora na Nlogonia, e é dono de um restaurante self-service, isto é, em que as pessoas montam seus próprios pratos com as comidas disponíveis no restaurante. Você está aguardando atender Q pessoas em seu restaurante hoje.

Cada pessoa tem um conjunto de alergias, que são identificadas por um número inteiro da mesma maneira: cada número primo que divide o número da pessoa indica que ela é alérgica ao ingrediente associado a esse número primo.

Dados os números associados a cada comida presente no seu restaurante, calcule, para cada uma das Q pessoas, quantos pratos diferentes ela pode montar de forma que não haja nenhum ingrediente no prato ao qual ela seja alérgica.

Entrada

A primeira linha da entrada contém um inteiro N ($1 \leq N \leq 10^5$), o número de comidas no seu restaurante. A próxima linha contém os números associados a cada comida V_i ($1 \leq V_i \leq 10^6$). A próxima linha contém um inteiro Q ($1 \leq Q \leq 10^5$), o número de pessoas que vai comer no seu restaurante. Seguem Q linhas; a i-ésima delas contém um número X_i ($1 \leq X_i \leq 10^6$), o número que representa as alergias da pessoa i.

Saída

Para cada uma das Q pessoas, imprima um número: a quantidade de pratos que podem ser montados com os ingredientes do restaurante, de forma que não esteja presente nenhum ingrediente ao qual a pessoa é alérgica. Como a resposta pode ser muito grande, imprima o resto de sua divisão por $10^9 + 7$.

Exemplo de entrada 1	Exemplo de saída 1
6 1 2 3 4 5 6 4 1 2 4 6	64 8 8 4

Explicação do exemplo 1:

A primeira pessoa não tem nenhuma alergia, então todas as 64 possibilidades de pratos são válidas para ela. Por outro lado, a última pessoa é alérgica às comidas que contêm os ingredientes associados aos números primos 2 e 3. Dessa forma, apenas 4 pratos são possíveis para ela: o prato vazio (sem nenhuma comida), o prato apenas com a comida 1 (que não tem nenhum ingrediente), o prato com a comida 5, e o prato com as comidas 1 e 5.

Problema J

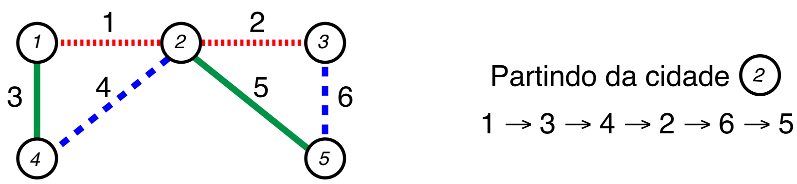
Jornada Colorida

No país de Oz as estradas são pavimentadas com pedras coloridas. Cada estrada conecta exatamente duas cidades, pode ser percorrida nos dois sentidos e é colorida com pedras de uma única cor.

Dorotéia está visitando Oz pela primeira vez e deseja realizar um passeio pelo país, atendendo às seguintes condições:

- O passeio deve iniciar e terminar numa mesma cidade.
- O passeio deve passar por cada estrada do país exatamente uma vez e não pode usar duas estradas consecutivas (ou seja, uma imediatamente após a outra) que tenham a mesma cor.
- A primeira e última estrada do passeio devem ter cores diferentes.

A figura (a) abaixo ilustra um exemplo com cinco cidades e seis estradas. A figura (b) mostra um possível passeio que inicia e termina na cidade 2 e satisfaz as restrições de cores das estradas. Na figura (b), o passeio inicia da cidade 2 e percorre, em sequência, as estradas 1 (vermelha), 3 (verde), 4 (azul), 2 (vermelha), 6 (azul) e, finalmente, 5 (verde).



(a) (b)

Ajude Dorotéia a encontrar tal passeio ou, se não for possível, indique que não existe.

Entrada

A primeira linha da entrada contém três inteiros, N, M e K, representando respectivamente o número de cidades ($2 \leq N \leq 1000$), o número de estradas ($1 \leq M \leq 1000$) e o número de cores ($1 \leq K \leq 1000$). As cidades são identificadas por inteiros de 1 a N, as estradas são identificadas por inteiros de 1 a M e as cores são identificadas por inteiros de 1 a K. Cada uma das M linhas seguintes descreve uma estrada e contém três inteiros I, J e C, onde I e J representam cidades ($1 \leq I, J \leq N$ e $I \neq J$), e C indica a cor da estrada $1 \leq C \leq K$. As estradas são dadas na ordem de sua identificação, ou seja, a primeira estrada da entrada é a de número 1, a segunda estrada é a de número 2, e assim por diante.

Saída

Caso não exista passeio que satisfaça as restrições, imprima um único inteiro -1. Caso contrário, seu programa deve produzir duas linhas descrevendo um passeio válido. A primeira linha deve conter o identificador da cidade inicial do passeio. A segunda linha deve conter M inteiros distintos, cada um identificando uma estrada, na ordem do passeio. Se houver mais de um passeio possível, imprima qualquer um deles.

Exemplo de entrada 1	**Exemplo de saída 1**

`5 6 3`
`1 2 1`
`2 3 1`
`1 4 2`
`2 4 3`
`2 5 2`
`3 5 3`
	`2`
	`1 3 4 2 6 5`

Explicação do exemplo 1:

Este é o exemplo do enunciado. São cinco cidades, seis estradas e três cores (1 = vermelha, 2 = verde, 3 = azul). Note também que há outros passeios possíveis, por exemplo partido da cidade 1: $3 \to 4 \to 2 \to 6 \to 5 \to 1$.

Exemplo de entrada 2	**Exemplo de saída 2**

`2 4 2`
`1 2 1`
`1 2 1`
`1 2 2`
`1 2 2`
	`1`
	`1 3 2 4`

Exemplo de entrada 3	**Exemplo de saída 3**

`6 6 3`
`1 2 1`
`2 3 2`
`3 1 3`
`4 5 1`
`5 6 2`
`6 4 3`
	`-1`

Exemplo de entrada 4	**Exemplo de saída 4**

`3 2 2`
`1 2 1`
`1 2 2`
	`1`
	`1 2`

Exemplo de entrada 5	**Exemplo de saída 5**

`3 3 1`
`1 2 1`
`2 3 1`
`3 1 1`
	`-1`

Problema K
Karamell

Karamell, Caramel, Caramello ou Caramelo. Diferentes idiomas, mas você sabe do que eu estou falando. Alice e Bob são irmãos gêmeos e também adoram caramelos! Por isso, como presente de aniversário, pediram caramelos para todos convidados da festa que estão organizando.

No dia da festa, Alice e Bob receberam seus presentes: N sacolas de caramelos. A i-ésima sacola continha a_i caramelos.

Como os aniversariantes não querem abrir as sacolas imediatamente, decidiram distribuir os caramelos da seguinte maneira: as sacolas serão consideradas em ordem e, no i-ésimo passo, os a_i caramelos da i-ésima sacola são dados à pessoa que tiver menos caramelos naquele momento. Em caso de empate, Alice recebe os caramelos (afinal, "primeiro as damas").

Uma coisa que eles não gostaram é que, dependendo da ordem em que as sacolas são consideradas, a quantidade final de caramelos que cada um recebe pode ser diferente. Por exemplo, se as sacolas estivessem ordenadas com as quantidades descritas pela sequência $[1, 2, 2, 3]$, Alice terminaria com 3 e Bob com 5 caramelos. Por outro lado, caso fossem consideradas na ordem $[1, 2, 3, 2]$, ambos terminariam com 4.

Você esqueceu de comprar caramelos para os aniversariantes, mas resolveu dar um presente ainda mais interessante: um programa que determina uma maneira de ordenar as sacolas de forma que Alice e Bob recebam a mesma quantidade de caramelos, se possível.

Entrada

A primeira linha contém um único inteiro N ($1 \le N \le 100$), indicando o número de sacolas. A segunda linha contém N inteiros $a_1, \ldots, a_N$ ($1 \le a_i \le 100$), onde a_i indica o número de caramelos na sacola i.

Saída

Caso seja impossível achar uma ordem como pedida, imprima -1. Caso contrário, imprima uma linha com N inteiros separados por espaço, indicando uma ordenação válida dos valores a_i que garanta que os caramelos serão divididos igualmente entre os irmãos.

Exemplo de entrada 1	Exemplo de saída 1
4 1 2 2 3	1 2 3 2

Exemplo de entrada 2	Exemplo de saída 2
5 1 2 2 3 6	3 6 2 2 1

Exemplo de entrada 3	Exemplo de saída 3
6 1 12 21 23 33 34	-1

Problema L
Lexicograficamente Máximo

Uma lista de N inteiros $a_1, \ldots, a_N$ está armazenada na memória de um dispositivo eletrônico. Este dispositivo possui uma operação muito peculiar disponível: a de troca de bits entre números. Mais precisamente, dados inteiros i, j e k, tal operação troca o k-ésimo bit do inteiro a_i pelo k-ésimo bit do inteiro a_j (e vice-versa).

Fenômenos muito interessantes podem acontecer ao se realizar tal operação uma ou mais vezes, como a obtenção de números que nem mesmo pertenciam à lista original, ou mesmo números maiores ou menores que todos os elementos originais.

Neste problema, estamos interessados em utilizar a operação tantas vezes quanto necessário para alterar a lista de números de forma que a lista resultante seja a lexicograficamente máxima, isto é, que a_1 seja o maior possível, que a_2 seja o maior possível dentre as possíveis soluções que maximizam a_1, e assim por diante.

Entrada

A primeira linha da entrada contém um inteiro N ($1 \leq N \leq 10^5$) e a segunda linha contém N inteiros, separados por espaço, correspontes à lista $a_1, \ldots, a_N$ ($0 \leq a_i \leq 10^9$).

Saída

Seu programa deve imprimir uma única linha contendo N inteiros separados por espaço, correspondentes à sequência lexicograficamente máxima que pode ser obtida.

Exemplo de entrada 1	Exemplo de saída 1
4 8 4 2 1	15 0 0 0

Exemplo de entrada 2	Exemplo de saída 2
4 12 15 1 20	31 13 4 0