

Problem A. Ainda Estou Comemorando!

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 256 megabytes



O que todos mais queríamos finalmente aconteceu: *Ainda Estou Aqui* venceu o *Oscar*! A nação brasileira entrou em êxtase, e o barulho da festa ecoou pelos quatro cantos do mundo. Fogos de artifício, gritos e panelas batendo marcaram esse momento histórico, e foi tão intenso que chegou a sacudir o planeta!

Mas nem todos compartilharam dessa alegria. Emília, uma autodeclarada inimiga da diversão, ficou furiosa com tanto barulho. Armada com um medidor de decibéis e um olhar de reprovação, ela passou o dia inteiro registrando o nível médio do som minuto a minuto. O resultado foi uma lista $1 \leq A_1, A_2, \dots, A_N \leq M$, onde cada valor A_k representa a amplitude média do som no minuto k .

Determinada a acabar com a festa, Emília quer provar que a celebração foi um escândalo. Para isso, ela planeja postar gráficos dramáticos nas redes sociais que “comprovem” que o barulho atingiu níveis “inaceitáveis”. No entanto, o que ela não quer que ninguém saiba é que pretende manipular os dados para espalhar *fake news*!

Ajude Ada a proteger o povo contra essa vilã! Descubra os verdadeiros valores de quantos minutos tiveram uma amplitude média maior ou igual a k , para todo $1 \leq k \leq M$, antes que Emília consiga distorcer os números!

Input

A primeira linha da entrada possui dois inteiros $1 \leq N, M \leq 10^5$.

A segunda linha contém N inteiros A_1, A_2, \dots, A_N ($1 \leq A_i \leq M$).

Output

Imprima M inteiros B_1, B_2, \dots, B_M , onde B_k representa quantos minutos tiveram uma amplitude média maior ou igual a k .

Examples

standard input	standard output
1 5 5	1 1 1 1 1
3 4 1 2 3	3 2 1 0
5 4 2 2 2 1 4	5 4 1 1

Problem B. Mapa da Cidade

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 256 megabytes



A cidade de MFPólis foi escolhida para sediar uma das etapas mais esperadas da temporada de Fórmula 1. Com a chegada iminente de equipes, pilotos e, é claro, milhares de fãs apaixonados pelo esporte, a prefeitura teve uma ideia ousada: criar um circuito turístico especial, conectando os principais pontos da cidade para entreter os visitantes nos dias em que não houver treinos ou corridas.

A proposta parecia simples: construir ruas exclusivas entre os pontos turísticos, formando um trajeto que os fãs pudessem percorrer com facilidade, como se estivessem seguindo um verdadeiro circuito de corrida pelas paisagens da cidade. No entanto, fãs de Fórmula 1 não são turistas comuns. Eles são estratégicos, analíticos e extremamente exigentes. Para eles, um circuito turístico só é considerado **BOM** se atender a dois critérios fundamentais:

1. É possível chegar a qualquer ponto turístico a partir de qualquer outro.
2. Existe exatamente um único caminho entre cada par de pontos turísticos.

Infelizmente, adequar o circuito para atender a essas exigências não é tarefa fácil. Por isso, o prefeito de MFPólis quer aproveitar que você ainda está aqui para contar com a sua ajuda.

Com o mapa da cidade, que possui N pontos turísticos numerados de 1 a N e uma lista de M ruas bidirecionais ligando esses pontos, sua tarefa é determinar se o circuito atual é **BOM**. Caso não seja, você deve calcular o número mínimo de ruas que devem ser destruídas e construídas, respectivamente, para que o circuito atenda aos critérios e se torne **BOM**.

Input

A primeira linha contém dois inteiros N e M ($1 \leq N \leq 10^5$, $0 \leq M \leq \min(2 \cdot 10^5, \frac{N(N-1)}{2})$) — o número de pontos turísticos e o número de ruas construídas.

As próximas M linhas contêm dois inteiros a_i e b_i ($1 \leq a_i, b_i \leq N$, $a_i \neq b_i$), indicando que existe uma rua ligando o ponto a_i ao ponto b_i .

É garantido que não existe mais do que uma rua ligando diretamente dois pontos.

Output

Imprima a palavra **BOM** se o circuito atender aos dois critérios.

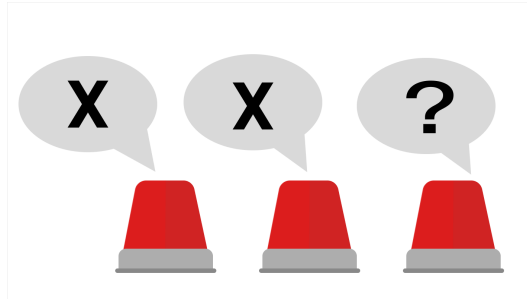
Caso contrário, imprima em uma única linha a palavra **RUIM** seguido de dois inteiros, representando, respectivamente, o número mínimo de ruas que devem ser destruídas e o número mínimo de ruas que devem ser construídas para que o circuito se torne **BOM**.

Examples

standard input	standard output
3 2 1 2 2 3	BOM
5 4 1 2 2 3 3 1 4 5	RUIM 1 1
4 6 1 2 1 3 1 4 2 3 2 4 3 4	RUIM 3 0

Problem C. O sensor diferente

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 256 megabytes



Marina acabou de ser promovida a engenheira líder no departamento de Controle de Qualidade de uma renomada fabricante de componentes automotivos. Após anos dedicados à empresa e superando desafios técnicos, ela agora enfrenta sua primeira grande missão: resolver um problema crítico na linha de produção de sensores de alta precisão.

A fábrica recebeu um alerta urgente de um importante cliente: veículos equipados com sensores da última remessa estão apresentando comportamentos inesperados. "Os carros estão mostrando leituras inconsistentes e isso está afetando o sistema de frenagem autônoma informou o cliente, preocupado com a segurança dos motoristas.

Marina foi chamada à sala de controle central, onde os engenheiros monitoram o processo de calibração tripla — o orgulho da empresa. Este sistema utiliza três sensores independentes para medir a mesma variável, garantindo máxima confiabilidade através de redundância. Em condições normais, todos deveriam apresentar leituras idênticas ou muito próximas. Quando há discrepância significativa, significa que um dos sensores está com defeito.

O terminal de diagnóstico exibe três valores inteiros K_1 , K_2 e K_3 . O tempo é crucial — cada minuto perdido pode significar mais sensores defeituosos sendo despachados e potencialmente colocando vidas em risco. Marina sabe que, em cada conjunto de medições, dois sensores sempre mostrarão o mesmo valor correto, enquanto o terceiro apresentará uma leitura divergente — este é o sensor com defeito que precisa ser identificado rapidamente.

Ajude Marina a identificar qual valor está incorreto para que a equipe possa ajustar o processo de fabricação imediatamente. A reputação da empresa como líder em tecnologia de sensores confiáveis está em suas mãos!

Input

A primeira linha da entrada possui um inteiro ($1 \leq T \leq 10^4$), o número de casos de teste.

Cada uma das T linhas seguintes possui 3 inteiros, ($-10^9 \leq K_1, K_2, K_3 \leq 10^9$), representando os valores medidos por cada um dos sensores.

É garantido que exatamente dois desses valores serão iguais.

Output

Imprima um único número inteiro: o valor do sensor que apresenta leitura diferente dos outros dois.

Example

standard input	standard output
5	10
7 10 7	2
2 1 1	100
100 59 59	13
-42 -42 13	21
21 3 3	

Problem D. Círculos da UNICAMP

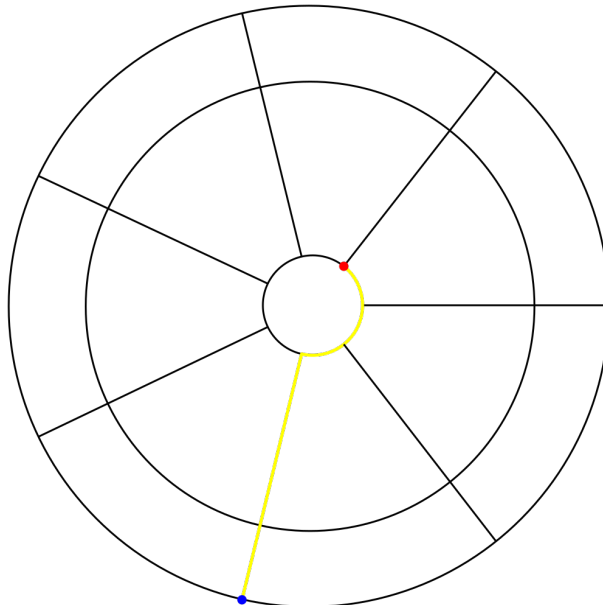
Input file: standard input
Output file: standard output
Time limit: 4 seconds
Memory limit: 256 megabytes



Ada, a arara, chegou à UNICAMP para fazer a segunda fase da MFP. Infelizmente, ela está atrasada e não sabe como chegar ao Instituto de Computação para fazer a prova.

As ruas da UNICAMP podem ser modeladas como três circunferências, de raios R_1 , R_2 e R_3 , conectadas por N linhas equidistantes (veja a figura). Note que as circunferências da UNICAMP são perfeitas e, portanto, seus comprimentos divididos pelos diâmetros correspondem a $\pi \approx 3.14159265359$.

Ada está na interseção entre uma dessas circunferências e alguma dessas linhas, e o Instituto de Computação também está. Entretanto, ela não tem certeza de qual interseção está. Sendo assim, Ada fará a você Q perguntas da seguinte forma: "Se eu estiver na interseção entre o círculo C_1 e a linha L_1 , e o Instituto de Computação estiver na interseção entre o círculo C_2 e a linha L_2 , qual é o menor caminho do meu ponto até o Instituto de Computação?". Ajude Ada a resolver essas dúvidas!



Exemplo de configuração da UNICAMP, com $R_1 = 2$, $R_2 = 7$, $R_3 = 9$ e $N = 7$. Ada começa na interseção entre o círculo 1 e a linha 2, e precisa chegar à interseção entre o círculo 3 e a linha 5. O menor caminho é representado pela linha amarela, com comprimento aproximado de 14.3855874062.

Input

A primeira linha de entrada contém 3 inteiros R_1 , R_2 e R_3 ($1 \leq R_1 < R_2 < R_3 \leq 10^4$), os raios dos

círculos da UNICAMP.

A segunda linha da entrada contém 2 inteiros N e Q ($1 \leq N, Q \leq 5 \cdot 10^4$), a quantidade de linhas que dividem a UNICAMP e a quantidade de perguntas, respectivamente.

As próximas Q linhas da entrada contêm 4 inteiros: C_1, L_1, C_2, L_2 ($1 \leq C_1, C_2 \leq 3, 1 \leq L_1, L_2 \leq N$), indicando que você deve imprimir o caminho mínimo entre a interseção do C_1 -ésimo círculo com a L_1 -ésima linha e da interseção do C_2 -ésimo círculo com a L_2 -ésima linha.

Output

Para cada pergunta, imprima um único número real: a resposta para a pergunta correspondente. A resposta será aceita se o erro absoluto ou relativo for menor ou igual a 10^{-6} .

Example

standard input	standard output
2 9 11	14.3855874062
7 5	0.0000000000
1 2 3 6	10.0783811092
1 2 1 2	2.0000000000
2 5 3 6	12.3855874062
3 5 2 5	
2 7 1 3	

Problem E. Engenharia Reversa

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 256 megabytes



Ada é uma arara feliz que vive em uma floresta com $N \cdot M$ árvores, dispostas no formato de uma matriz de N linhas por M colunas, em que cada árvore tem uma certa quantidade de maçãs.

Ada frequentemente voa de uma árvore para outra adjacente a ela (em um dos quatro sentidos na matriz), mas como ela gosta de surpresas, ela garantiu que todas as árvores adjacentes entre si tivessem um número diferente de maçãs.

Porém, ela possui uma arqui-inimiga: Py, uma serpente que vive na mesma floresta. Py, insatisfeita com a felicidade de Ada, escolheu um subconjunto das árvores e comeu exatamente uma maçã de cada uma das árvores desse subconjunto.

Agora Ada quer repor as maçãs, mas não lembra quais eram antes as quantidades de maçãs em cada árvore, e nem sabe qual foi o subconjunto escolhido por Py. Dessa forma, dada a atual quantidade de maçãs em cada árvore, ajude Ada imprimindo uma possível configuração de quantidades de maçãs em cada árvore antes da ação de Py.

Input

A primeira linha de entrada contém dois inteiros N e M ($1 \leq N, M \leq 100$), as dimensões da matriz da floresta.

Seguem N linhas, cada uma com M inteiros, representando a quantidade de maçãs em cada árvore da floresta, após a ação de Py.

Output

Imprima N linhas, cada uma com M inteiros, representando uma possível configuração de quantidades de maçãs em cada árvore antes da ação de Py.

Se houver múltiplas respostas possíveis, imprima qualquer uma.

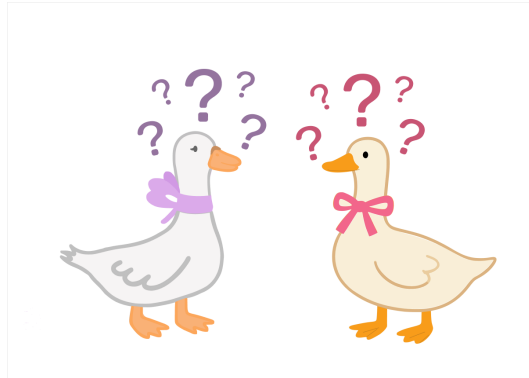
Para os casos de teste deste problema, é garantido que sempre haverá resposta.

Examples

standard input	standard output
2 2 1 1 2 3	2 1 3 4
2 3 10 11 12 13 14 15	10 11 12 13 14 15

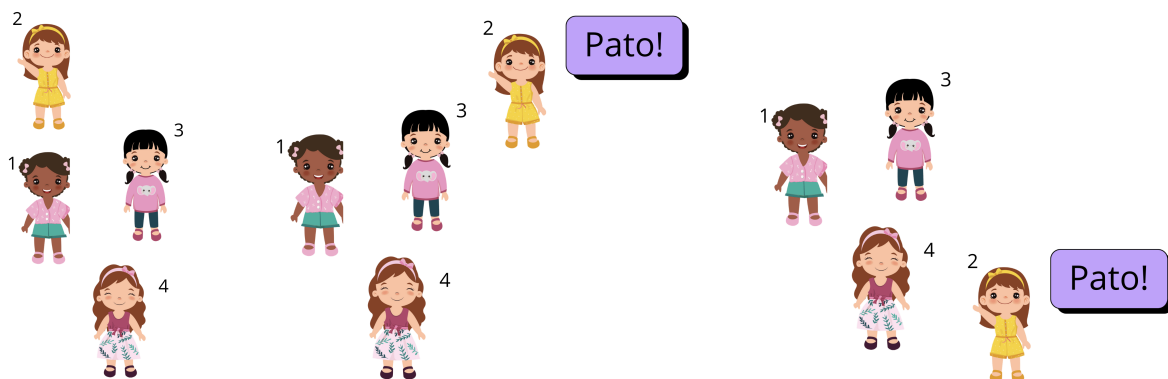
Problem F. Pato ganso

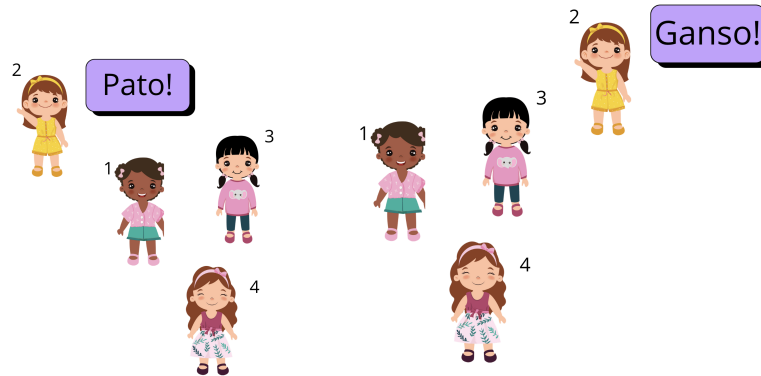
Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 256 megabytes



No recreio da escola, algumas crianças se juntam em uma roda para brincar de “pato-ganso”. O jogo funciona da seguinte forma:

- Na roda há N crianças, que identificamos com os inteiros de 1 a N de tal forma que a criança i está do lado da $i + 1$, para $i < N$, e a criança 1 está do lado da N . Por exemplo, a criança 1 está do lado da N e da 2, a criança 2 do lado da 1 e da 3, etc.
- Inicialmente, todas as crianças estão sentadas, olhando para o centro.
- Através de um jogo de “Uni duni tê”, as crianças apontam para uma criança P , que se levanta do círculo. Então, ela começa a andar ao redor da roda e, para cada criança que passa, fala “Pato”, com um leve toque na cabeça.
- Por fim, após falar “Pato” K vezes, ela grita “Ganso” e encosta na criança seguinte, que deve se levantar e pegar a criança de número P .





Exemplo de jogo com $N = 4$ crianças, no qual a criança $P = 2$ começa fora da roda e fala “Pato” $K = 3$ vezes.

Note que, como a criança de número P se levantou da roda, restam apenas $N - 1$ crianças sentadas: $1, 2, \dots, P - 1, P + 1, \dots, N - 1, N$.

Após muitos recreios jogando, as crianças observaram que umas falavam “Pato” muito mais do que as outras, que ficavam um tempo desproporcional sentadas esperando. Para lidar com isso, decidiram que o número K de vezes em que uma criança proclama “Pato” será fixo.

Em contrapartida, isso cria um novo problema: agora o jogo é previsível. Como grande entusiasta do clube das Monitoras Femininas do Pato-ganso (MFP), você foi encarregada de dizer exatamente qual será a criança que irá escutar “Ganso”.

Input

A entrada consiste de três linhas, cada uma com um único inteiro.

A primeira linha da entrada possui um inteiro N ($2 \leq N \leq 1000$), o número de crianças na roda.

A segunda linha da entrada possui um inteiro P ($1 \leq P \leq N$), o identificador da criança que se levanta da roda.

A terceira linha da entrada possui um inteiro K ($1 \leq K \leq 1000$), representando o número de vezes que “Pato” é dito.

Output

Imprima uma única linha contendo um único inteiro: a posição da criança que escuta “Ganso”.

Examples

standard input	standard output
3 1 3	3
2 2 100	1
4 2 3	3

Problem G. Misturador Frase-Palavra

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 256 megabytes



Nossa amiga Ada está trabalhando no desenvolvimento de um novo programa: o Misturador Frase-Palavra (MFP). Ele transforma duas palavras de mesmo tamanho em uma única palavra, que contém as letras de cada uma alternadas e em ordem.

Mais formalmente, ele recebe duas palavras A e B , ambas com N letras cada. Então, ele devolve uma sequência S , com $2N$ letras, cujas letras são, em ordem, $A_1B_1A_2B_2 \dots A_NB_N$.

Ada terminou de escrever o programa e o executou, mas ela quer verificar se ele está funcionando corretamente. Por isso, ela pede que você, dada a sequência S devolvida pelo programa, descubra e imprima quais devem ser as palavras A e B que ele recebeu, considerando que foi executado corretamente.

Input

A entrada contém uma única linha com a palavra S .

É garantido que S é uma palavra com $|S|$ letras minúsculas ($2 \leq |S| \leq 10$, $|S|$ é par).

Output

Imprima duas linhas: a primeira deve conter a palavra A e a segunda a palavra B .

Examples

standard input	standard output
hweolrllod	hello world
oi	o i
qtuael	que tal