

XI Maratona UnB de Programação

Caderno de Problemas

27 de setembro de 2023



(Este caderno contém 14 problemas)

Comissão Organizadora:

Alberto Tavares Duarte Neto (UnB)

Daniel Porto (UnB)

Edson Alves da Costa Júnior (UnB/FGA)

Guilherme Novaes Ramos (UnB)

Vinicius Ruela Pereira Borges (UnB)

Daniel Saad Nogueira Nunes (IFB)

Jeremias Moreira Gomes

Lembretes

- É permitido consultar livros, anotações ou qualquer outro material impresso durante a prova, entretanto, o mesmo não vale para materiais dispostos eletronicamente.
- A correção é automatizada, portanto, siga atentamente as exigências da tarefa quanto ao formato da entrada e saída conforme as amostras dos exemplos. Deve-se considerar entradas e saídas padrão;
- Para cada problema, além dos testes públicos, o juiz executará a sua submissão contra uma série de testes secretos para fornecer um parecer sobre a correção do programa.
- Procure resolver o problema de maneira eficiente. Se o tempo superar o limite pré-definido, a solução não é aceita. Lembre-se que as soluções são testadas com outras entradas além das apresentadas como exemplo dos problemas;
- Utilize a aba *clarification* para dúvidas da prova. Os juízes podem opcionalmente atendê-lo com respostas acessíveis a todos;

C/C++

- Seu programa deve retornar zero, executando, como último comando, `return 0` ou `exit 0`.

Java

- Não declare ‘`package`’ no seu programa Java.
- Note que a convenção para o nome do arquivo fonte deve ser obedecida, o que significa que o nome de sua classe pública deve ser uma letra maiúscula igual a letra que identifica o problema.

Python

- Tenha cuidado ao selecionar a versão correta na submissão.

Problema A

Abdução Alienígena

Limite de tempo: 1s
Limite de memória: 256MB

Autor: Vinicius Borges

Aristênio é um renomado ufologista da Nlogonia e, certo dia, observou um disco voador pairando sobre a zona rural da cidade onde mora. Com base nos seus conhecimentos, ele verificou que a raça alienígena é muito perigosa e, com medo de ser abduzido, decide fugir para a cachoeira dos Azares. Nesse local, ele consegue se esconder, impossibilitando a abdução.

Inicialmente, Aristênio está em sua residência. Para chegar até a cachoeira dos Azares, ele precisará passar pelos abrigos existentes na zona rural por meio das trilhas que os ligam. Obviamente o disco voador percebe a fuga de Aristênio e tentará perseguir Aristênio até essa cachoeira.

A tarefa de fuga de Aristênio é mais difícil, pois cada trilha possui uma dificuldade (terreno, rios, animais selvagens) e a travessia por ela pode ser demorada, sendo feita em vários minutos. Por outro lado, não existem obstáculos para o rápido disco voador, que sempre voa sobre uma trilha em um único minuto. Visando dificultar uma iminente abdução e conhecendo as trilhas da região, Aristênio corre pelo menor caminho possível até a cachoeira dos Azares para se salvar. Da mesma maneira, o disco voador fará o menor caminho possível para tentar abduzir Aristênio.

Considere que os abrigos sejam enumerados por inteiros de 1 a N , em que 1 é o abrigo da residência de Aristênio e N é o abrigo subterrâneo na cachoeira dos Azares. Determine se Aristênio consegue chegar ao abrigo da cachoeira dos Azares com segurança, isto é, antes do disco voador.

Entrada

A primeira linha da entrada contém três números inteiros positivos N ($3 \leq N \leq 5 \cdot 10^4$), M ($2 \leq M \leq 10^4$) e D ($1 \leq D \leq N$, $2 \leq D \leq N - 1$), que indica a quantidade de abrigos, a quantidade de trilhas que ligam esses abrigos e a posição do disco voador nas trilhas da zona rural, respectivamente.

Em seguida, existem M linhas descrevendo os abrigos e as trilhas que os ligam. A i -ésima linha contém três inteiros x_i , y_i e t_i ($1 \leq x_i, y_i \leq N$, $1 \leq t_i \leq 10^5$), representando o tempo t_i gasto para percorrer a trilha desde o abrigo x_i até y_i . Aristênio pode percorrer uma trilha que liga dois abrigos em qualquer um dos dois sentidos.

Saída

Imprima “SIM” (sem aspas duplas) caso Aristênio consiga escapar da abdução. Imprima “NAO” (sem aspas duplas) caso contrário.

Exemplo

XI Maratona UnB de Programação

Entrada	Saída
5 5 3	NAO
1 2 7	
2 3 2	
2 4 3	
3 5 1	
4 5 5	
7 6 6	SIM
4 5 6	
1 2 1	
2 7 2	
5 6 4	
2 3 2	
3 4 7	
8 11 6	NAO
6 7 7	
2 5 6	
1 2 13	
7 8 11	
5 4 4	
4 6 11	
3 4 9	
5 7 8	
2 4 7	
3 6 7	
2 3 9	

Problema B

Boate Azul

Limite de tempo: 1s
Limite de memória: 256MB

Autor: Daniel Porto

A Boate Azul é a boate mais famosa da cidade. Reduto de pessoas bonitas e apaixonadas. O dono do estabelecimento resolveu contemplar alguns de seus clientes mais assíduos com um cartão de desconto. Para enviar o cartão de desconto, o dono da boate possui como entrada uma lista com os últimos acessos dos clientes. Essa lista está na forma de string e , por medidas de segurança, apresenta somente a inicial do nome de cada cliente.

A lista dos contemplados deve ser construída também com as iniciais, mas conforme as seguintes regras:

1. Ao incluir uma pessoa de inicial i na lista, a pessoa imediatamente posterior na lista de últimos acessos também deve ser incluída.
2. As inclusões são feitas em ordem alfabética, exceto quando em conflito com a regra anterior.
3. Cada cliente só é contemplado uma vez.

Caso a lista dos contemplados não possa ser montada com todos os clientes, um erro deve ser indicado.

Entrada

A entrada é descrita por uma única linha contendo uma string s composta apenas por n letras minúsculas ($1 \leq n \leq 100$), em que n é par.

Saída

Imprima a lista de clientes contemplados com o cartão de desconto, caso ela seja possível seguindo as regras definidas; ou imprima -1 caso contrário.

Exemplo

Entrada	Saída
beazcd	azbecd
abadcd	abcd
abad	-1

Problema C

Carimbador Maluco

Limite de tempo: 1s
Limite de memória: 256MB

Autor: Guilherme Ramos

A situação está cada vez mais complicada, e o governo pseudo-ditatorial impôs uma série de restrições ao seu direito de ir e vir. Agora, todos têm de ter uma permissão especial com o carimbo do Marechal Ukko.

Para viajar, é preciso encaminhar um pedido (em 3 vias) e aguardar a resposta. O voo só é liberado se o pedido voltar selado, registrado, carimbado, avaliado e rotulado. Nessa ordem!

Faça um programa para verificação dos pedidos de modo que o Mal. Ukko não seja preso por balburdiar os processos!

Entrada

A entrada consiste em uma string contendo uma sequência de até 1000 caracteres, sem espaços. É garantido que os processos necessários, se ocorrerem, são únicos.

Saída

Caso todas as condições para viajar sejam atendidas, apresente a mensagem “Sim, sim, sim, sim!”. Caso contrário, informe que “Nao vai a lugar nenhum!”.

Exemplo

Entrada	Saída
selaDoregistradOcaRImbadoavaliadorotulado	Sim, sim, sim, sim.
marcadoseladocoladoregistradolambidocarimbadoavaliadorotulado	Sim, sim, sim, sim.
praLuaataxaeAlta	Nao vai a lugar nenhum!

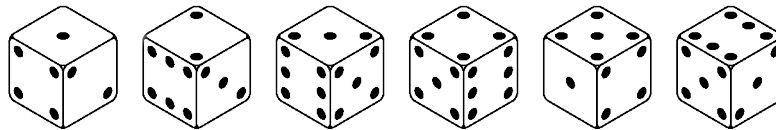
Problema D

Dados

Limite de tempo: 1s
Limite de memória: 256MB

Autor: Edson Alves

Uma possível representação de um dado de seis faces exhibe 3 de suas faces, conforme ilustra a figura abaixo:



Naturalmente, as outras 3 faces ficam ocultas. Você conseguiria identificar os números que não foram representados?

Entrada

A primeira linha da entrada contém um inteiro T ($1 \leq T \leq 20$), representando o número de casos de teste.

Cada caso de teste é representado por uma única linha, indicado os três números x, y, z ($1 \leq x, y, z \leq 6, x \neq y, x \neq z, y \neq z$) visíveis na representação, separados por um espaço em branco.

Saída

Para cada caso de teste imprima, em uma linha, a soma dos números que ficaram ocultos na representação do dado.

Exemplo

Entrada	Saída
6	14
1 4 2	10
2 6 3	7
3 6 5	6
4 5 6	11
5 1 4	7
6 5 3	

Notas

A figura ilustra os seis casos de teste listados. No primeiro deles, a representação exhibe os números 1, 4 e 2, e o omite os números 3, 5 e 6, de modo que a resposta é $3 + 5 + 6 = 14$.

Problema E

Entra no meu Time

Limite de tempo: 3s
Limite de memória: 4MB

Autor: Jeremias Moreira Gomes

Pessoal alguém quer participar da seletiva e tá sem time ainda?	22:33	Tem alguém sem time aí pra seletiva?	16:09
queríamos juntar o pessoal que ta sem time pra gente organizar	15:16	Tbm to sem time	19:40
alguém aí sem time para a maraatona da semana universitária?	19:41	To sem time :S	20:18
Galera, algum iniciante aí sem time ? Tá faltando um pra fechar o meu	20:06	Aí galera alguém tá sem time pra maratona?	10:02
		peessoas alguem ta sem time? tem uma vaga aqui	09:58
		Alguém tá sem time pra seletiva da unb?	15:38

“Alguém sem time?”

Não tem coisa pior do que um maratonista querendo participar de um maratona e não conseguindo encontrar os amigos para fecharem um time.

O professor Thomas, pensando nos seus alunos, pediu para que todos a participarem da próxima maratona escrevessem um número no papel e o colocassem em uma urna. Os alunos que formam um time deveriam escrever o mesmo número no papel.

Infelizmente, Marvin não conseguiu formar seu time, mas mesmo assim escreveu um número no papel e o colocou na urna. O professor conseguiu achar outros dois alunos para participarem da maratona junto com Marvin mas, como este competidor esqueceu o que escreveu no papel, agora o professor precisa localizar o número de Marvin para registrar o time. Ele pediu a sua ajuda para localizar esse número, sabendo que, exceto por Marvin, todos os times estão completos com três participantes.

Entrada

A entrada é composta por um único caso de teste. A primeira linha da entrada possui um inteiro N ($4 \leq N \leq 18 \cdot 10^5 + 1$), que indica a quantidade de papéis na urna. As N linhas seguintes possuem um inteiro V cada ($0 \leq V \leq 10^{12}$) indicando o número no papel que cada competidor preencheu.

Saída

A saída deve conter um único inteiro, que é o número depositado por Marvin na urna.

Exemplo

XI Maratona UnB de Programação

Entrada	Saída
7	7
2	
1	
2	
1	
7	
1	
2	
10	1
2	
2	
4	
2	
2	
2	
4	
4	
2	
1	

Problema F

Formiguinhas

Limite de tempo: 1s
Limite de memória: 256MB

Autor: Guilherme Ramos

Ir ao mercado não é tarefa para os fracos: as formigas estão vindo com tudo!

A cada compra, os insetos avançam sobre as pessoas sem motivo aparente. A situação piorou desde que a esse tipo específico de formiga entrou para a lista de animais silvestres, proibindo a sua utilização, perseguição, destruição, caça ou apanha (Lei 5.197/67, agora conhecida como “Lei Kent Brockman”).

Numa tentativa de retirar essa espécie da lista protegida, você está registrando as ocorrências de ataques para apresentar às autoridades.

Entrada

A entrada consiste de duas linhas contendo um string cada, indicando o contexto da compra no mercado. Nenhum contexto é vazio ou tem mais de 100 caracteres.

Saída

Para cada contexto, registre a reação conforme os exemplos.

Exemplo

Entrada	Saída
Fui ao mercado comprar cafe	Eu sacudi, sacudi, sacudi.
E a formiguinha subiu no meu pe	Mas a formiguinha nao parava de subir!
Fui ao mercado comprar batata roxa	Eu sacudi, sacudi, sacudi.
E a formiguinha subiu na minha coxa	Mas a formiguinha nao parava de subir!

Problema G

G de StrinG?

Limite de tempo: 1s
Limite de memória: 256MB

Autor: Alberto Neto

Você recebe uma string misteriosa s de presente de aniversário, mas você não gosta da string s ! Você claramente prefere a string t .

Então você decide transformar a string s usando a seguinte operação quantas vezes quiser.

- Escolha duas letras c_1 e c_2 . Transforme **todas** as ocorrências de c_1 de s em c_2 .

Diga se é possível transformar s em t e, se for, a menor quantidade de operações necessárias para isso.

Entrada

A primeira linha contém um único inteiro n ($1 \leq n \leq 100$) — o tamanho das strings s e t .

A segunda linha contém uma cadeia de caracteres s ($|s| = n$) — a string a ser modificada.

A terceira linha contém uma cadeia de caracteres t ($|t| = n$) — a string desejada.

Saída

Se for impossível transformar s em t , imprima -1. Caso contrário, imprima um único inteiro representando a menor quantidade de operações necessárias para transformar s em t .

Exemplo

Entrada	Saída
4 bcde aaaa	4
4 aaaa bcde	-1
5 bzjck afzaf	5

Problema H

Hora de Ao Mossar

Limite de tempo: 1s
Limite de memória: 256MB

Autor: Alberto Neto

As lenhadoras de segtree viajaram para Chapecó, para a Final Brasileira da Maratona de Programação da SBC.

Há n opções de ao mosso no restaurante. Duda Carvalho fica bastante incomodada quando o dinheiro da sua conta não termina com .00, isto é, quando não é possível expressar sua fortuna apenas com unidades de real. Sabendo que inicialmente a conta bancária dela tem d centavos (e infinitas unidades de real), suas companheiras Duda Holanda e Nathália decidiram investigar a situação.

Elas calculam, para cada ao mosso, seu fator de gostosura p_i e a quantidade de centavos c_i no preço. Se s é a soma da gostosura de todos os n ao mossos, elas afirmam que a probabilidade de Duda Carvalho comer o ao mosso i é p_i/s .

As meninas precisam de Duda bem para fazer a prova, mas sabem que se sua conta bancária não estiver redonda ela vai ficar pensando nisso o tempo todo. O seu trabalho é calcular o valor esperado de ao mossos que Duda Carvalho comerá até que os centavos de sua conta bancária fique exatamente .00, para que as meninas possam planejar a viagem.

Entrada

A primeira linha de entrada contém dois inteiros n e d ($1 \leq n \leq 100$, $1 \leq d \leq 99$) — a quantidade de ao mossos diferentes e os centavos na conta bancária de Duda, respectivamente.

Cada uma das próximas n linhas contém dois inteiros c_i e p_i ($0 \leq c_i \leq 99$, $1 \leq p_i \leq 10^5$) — os centavos do preço e o fator de gostosura do ao mosso i , respectivamente. **é garantido que a soma de todos os p_i é menor ou igual a 10^5 .**

Saída

Imprima um único número real v — o valor esperado de ao mossos que Duda Carvalho comerá até que sua conta bancária fique redonda, isto é, tenha 00 nas casas dos centavos. Sua resposta será considerada se o erro absoluto não exceder 10^{-6} .

Exemplo

XI Maratona UnB de Programação

Entrada	Saída
1 10	9.0000000
10 1	
3 10	9.3431085
10 1	
10 1	
20 2	
2 33	1.4285714
0 3	
67 7	

Problema I

Impostor

Limite de tempo: 1s
Limite de memória: 256MB

Autor: Edson Alves

Os matemáticos foram contagiados pelo sucesso do jogo “Among Us” e propuseram uma nova versão do jogo. Nesta nova variante, o jogo é composto por 5 números inteiros positivos distintos: 4 membros da tripulação e o impostor. Os membros da tripulação compartilham um divisor comum D que não divide o impostor. O curioso é que com estas regras, a depender dos valores dos inteiros, pode ser que exista mais de um impostor!

Dados os valores dos 5 inteiros, determine o impostor e o divisor D que o caracterize como impostor. Se existir mais de um impostor possível, acuse qualquer um deles!

Entrada

A entrada é composta por uma única linha, com 5 inteiros positivos x_i ($1 \leq x_i \leq 10^9, 1 \leq i \leq N$), separados por um espaço em branco.

Saída

Imprima, em uma linha, o impostor x e o divisor D que o identifica, separados por um espaço em branco. Caso exista mais de um impostor, ou mais de um divisor D , imprima qualquer um deles.

Exemplo

Entrada	Saída
10 20 30 42 50	42 5
12 30 15 48 16	15 2

Notas

No primeiro caso, $D = 5$ divide 10, 20, 30 e 50, porém não divide 42, de modo que 42 é o impostor. Observe que $D = 10$ também poderia ser usado para identificar o impostor.

No segundo caso, 15 é impostor (comprovado por $D = 2$), mas 16 também é impostor ($D = 3$).

Problema J

Jiraiya-ja-ya

Limite de tempo: 1s
Limite de memória: 256MB

Autor: Daniel Saad Nogueira Nunes

O incrível ninja Jiraiya, dentre muitas das habilidades, também era conhecido por ser um mestre na arte dos palíndromos. Ele conseguia codificar e decodificar mensagens secretas utilizando esses tipos especiais de palavras para trabalhar com seus aliados.

Seu poderoso inimigo, chamado Oninin Dokusai, em uma de suas tentativas de dominação mundial, resolveu atrapalhar a comunicação de Jiraiya com seus aliados, impedindo o ninja de ser auxiliado, e assim, chegar em seu objetivo. As mensagens que Jiraiya recebia, agora estavam corrompidas. Para tentar obter as mensagens originais, Jiraiya resolveu tentar transformar a mensagem recebida em um palíndromo, de modo a continuar a sua comunicação com a equipe. Sua estratégia foi a de modificar a mensagem original utilizando as operações de modificar um caractere, inserir ou remover um caractere em uma posição arbitrária. Além disso, Jiraiya procurou minimizar o número de operações realizadas.

Ajude nosso ninja a obter uma mensagem palindrômica ao utilizar o número mínimo de operações possíveis na mensagem original.

Entrada

A entrada possui uma única linha com uma string S , composta de caracteres minúsculos sobre o alfabeto $\{a, b, \dots, z\}$.

Restrições:

- $1 \leq |S| \leq 1000$.

Saída

Imprima uma linha com um inteiro, que indica o número mínimo de operações a ser utilizado para transformar a mensagem original em um palíndromo.

Exemplo

Entrada	Saída
abracadabra	3
ovo	0
abcde	2

Notas

No primeiro exemplo, basta remover o **r**, trocar o **d** por **c** e inserir um **r** antes do último **b**, obtendo o palíndromo **abracacarba**.

A palavra do segundo exemplo já é um palíndromo, não é necessário fazer nada.

Já no terceiro exemplo, podemos substituir **d** por **b** e **e** por **a**.

Problema K

Kaskata

Limite de tempo: 1s
Limite de memória: 256MB

Autor: Alberto Neto

Arthur_9548 e MagePetrus, em sua empresa de jogos É Só Fazer[®], estão criando um jogo chamado Kaskata, no qual o jogador deve jogar uma bola numa posição e fazer pontos.

As regras do jogo funcionam da seguinte forma. Existem n *bumpers* numerados de 1 até n de forma ordenada, e o jogador solta uma bola com m de energia no bumper x . Quando a bola acerta o bumper x , se ela possui energia **positiva** então uma bola é criada nos bumpers $x - 1$ e $x + 1$ (se estes existirem) com $m - 1$ de energia cada. Toda vez que uma bola acerta um bumper, a pontuação dele é aumentada em 1.

Dudu foi chamado para testar o jogo. Ele vai soltar uma bola com m de energia no bumper x . Ajude no *debugging* do jogo e encontre a pontuação final de cada bumper. Como esse valor pode ser muito grande, calcule-o módulo 998244353.

Entrada

A primeira linha de entrada contém três inteiros n , x e m ($1 \leq x \leq n \leq 100$; $0 \leq m \leq 10^9$) — a quantidade de bumpers, a posição inicial e a energia inicial da bola, respectivamente.

Saída

Imprima uma única linha com n inteiros p_1, \dots, p_n — a pontuação de cada um dos n bumpers módulo 998244353.

Exemplo

Entrada	Saída
5 2 0	0 1 0 0 0
5 2 1	1 1 1 0 0
5 2 2	1 3 1 1 0
5 2 3	3 3 4 1 1
1 1 100000	1

Problema L

Labuta Diária

Limite de tempo: 1s
Limite de memória: 256MB

Autor: Daniel Saad Nogueira Nunes

Toupeiras são mamíferos inteiramente adaptados a ambientes subterrâneos e se alimentam, principalmente, de animais invertebrados, como as minhocas. Esses mamíferos conseguem cavar extensos túneis e se alojam em tocas subterrâneas. São animais conhecidos por serem solitários devido a diversos fatores, como: territorialidade, escassez de recursos, melhor eficiência na escavação e melhor capacidade de caça. Outras espécies de toupeira, como a toupeira-nariz-de-estrela, exibem traços sociais ao compartilhar túneis e recursos.

Marco, em sua labuta diária, está estudando as toupeiras sociais e percebeu duas características interessantes nos túneis e tocas cavados por elas:

- É possível sair de uma toca e chegar em outra através de uma sequência de túneis.
- Só existe uma possível rota de uma toca a outra.

Ajude Marco a estudar o comportamento das toupeiras nos deslocamentos de uma toca a outra ao calcular a distância desses deslocamentos.

Entrada

A primeira linha da entrada possui dois inteiros n e q , que indicam, respectivamente, o número de tocas e o número de análises de Marco. As próximas $n - 1$ linhas, descrevem cada, uma tripla de inteiros, (u, v, c) , que informa que existe um túnel da toca identificada por u até a toca identificada por v com comprimento de c metros. As últimas q linhas, contém cada, um par de inteiros (a, b) , descrevendo o objeto de análise de marco.

Restrições:

- $2 \leq n \leq 10^5$
- $1 \leq q \leq 10^5$
- $1 \leq u, v \leq n, u \neq v$.
- $1 \leq c \leq 10^4$.

Saída

Para cada uma das análises de Marco, imprima uma linha com a distância percorrida pelas toupeiras no trajeto da toca identificada por u até a toca identificada por v .

Exemplo

Entrada	Saída
11 4	27
10 1 5	2
6 10 3	24
3 6 8	12
9 3 5	
2 3 7	
11 3 10	
11 7 2	
11 5 1	
11 4 4	
8 11 6	
1 5	
11 7	
6 8	
9 2	

Problema M

Maximizando o ataque

Limite de tempo: 1s
Limite de memória: 256MB

Autor: Edson Alves

Pedro está jogando um RPG tático e, no início de cada missão, ele dispõe de C moedas de ouro para contratar o seu exército. Há N unidades disponíveis para contratação: cada uma delas cobra c moedas de ouro e tem um ataque igual a a .

Contudo, o ataque de cada unidade pode ser aumentado mediante a compra de melhores equipamentos. Há 3 níveis de equipamento disponíveis para cada unidade: para ser contratada usando um equipamento de nível j , a unidade cobra e_j moedas de ouro e o seu o ataque se torna k_j .

Dados os valores de N e C , os custos e_j e os novos ataques k_j de cada unidade, determine o melhor exército que Pedro pode contratar, se o objetivo é maximizar a soma dos ataques das unidades contratadas.

Entrada

A primeira linha da entrada contém os valores dos inteiros N ($1 \leq N \leq 100$) e C ($1 \leq C \leq 10^5$), respectivamente, separados por um espaço em branco.

As $2N$ linhas seguintes contém informações sobre as unidades. Cada unidade é representada por duas linhas: a primeira delas contém os valores dos inteiros a, k_1, k_2 e k_3 ($1 \leq a < k_1 < k_2 < k_3 \leq 10^9$), separados por um espaço em branco. A segunda linha contém os valores dos inteiros c, e_1, e_2 e e_3 ($1 \leq c < e_1 < e_2 < e_3 \leq 10^5$), separados por um espaço em branco.

Saída

Imprima, em uma linha, o valor máximo da soma do ataque das unidades contratadas por Pedro e o número de unidades contratadas, separados por um espaço em branco.

Caso Pedro tenha contratado uma ou mais unidades, imprima N linhas. Cada linha deve contém o identificador i da unidade contratada ($1 \leq i \leq N$) e o nível do equipamento que ela estará usando, separados por um espaço em branco. Caso não seja pago nenhum equipamento para o unidade, imprima o valor zero; caso contrário, imprima o nível L do equipamento ($1 \leq L \leq 3$).

Cada unidade pode ser contratada uma única vez. Se houver mais de uma forma de maximizar a soma dos ataques das unidades, imprima qualquer uma delas.

Exemplo

Entrada	Saída
3 10	10 2
1 3 5 9	2 0
1 4 7 10	3 0
5 9 12 15	
4 10 15 20	
5 10 15 20	
6 12 18 24	
1 1	0 0
1 2 3 4	
2 3 4 5	
5 100	35 2
1 2 4 8	3 3
10 20 30 40	5 1
1 2 3 10	
10 20 30 40	
2 3 4 5	
5 10 15 20	
5 8 12 15	
30 50 80 90	
20 30 50 100	
50 80 120 150	

Notas

No primeiro caso, Pedro consegue um ataque total de 10 se contratar as unidades 2 e 3 sem equipamentos: a unidade 2 custa 4 moedas e tem ataque igual a 5; a unidade 3 custa 6 moedas e também tem ataque igual a 5.

No segundo caso, Pedro não consegue pagar a única unidade disponível.

Problema N

Número no display

Limite de tempo: 1s
Limite de memória: 256MB

Autor: Edson Alves

Daniel está trabalhando no código de um display luminoso, capaz de exibir k caracteres. Para testar o dispositivo, ele escreveu uma rotina que exibe no painel, inicialmente, os k primeiros dígitos da expansão decimal infinita da fração $1/N$, onde N é um inteiro escolhido por Daniel a cada teste. Após a exibição inicial, a cada segundo todos os dígitos se movem uma posição para a esquerda, de modo que o primeiro dígito da expansão deixa o display e o $(k + 1)$ -ésimo entra na posição vaga à direita, e a cada segundo o processo se repete.

Acompanhando o teste, Daniel teve uma curiosidade: para um certo valor N , quanto tempo ele precisaria esperar, em segundos, para que o painel exibisse o número M , composto por k dígitos decimais? Ou M jamais seria exibido?

Auxilie Daniel a tirar esta dúvida computando o tempo, em segundos, ou indicando que M jamais será exibido.

Entrada

A primeira linha da entrada contém os valores dos inteiros N ($2 \leq N \leq 10^8$) e k ($1 \leq k \leq 10^6$), separados por um espaço em branco.

A segunda linha da entrada contém um número inteiro M , composto por exatamente k dígitos decimais. O valor de M pode conter zeros à esquerda.

Saída

Imprima, em uma linha, a mensagem “Sim”, caso N eventualmente seja exibido no display, ou a mensagem “Nao”, caso contrário.

Caso N venha a ser exibido imprima, na linha seguinte, o tempo mínimo, em segundos, que Daniel deve esperar para que N seja exibido pela primeira vez.

Exemplo

Entrada	Saída
7 2	Sim
28	2
2 5	Sim
50000	0
12345 16	Nao
8100445524503848	

Notas

No primeiro caso, observe que $1/7 = 0.14285714285714285\dots$, de modo que, inicialmente, o display exibe o número 14. Passado um segundo, o display exibirá o número 42 e, um segundo depois, o número 28, de modo que Daniel deve esperar, no mínimo, 2 segundos.

No segundo caso, $1/2 = 0.50000000\dots$, de modo que o 50000 é o valor exibido inicialmente pelo display. Assim, Daniel não precisa esperar para visualizar 50000 no display.

No terceiro caso, por mais que Daniel espere, o número 8100445524503848 jamais será exibido pelo display.