

VI Competição Feminina de Programação da UnB

28 de junho de 2025



VI COMPETIÇÃO
FEMININA DE
PROGRAMAÇÃO
DA UnB



Caderno de Problemas - Informações Gerais

Este caderno contém 09 problemas; as páginas estão numeradas de 1 a 16, não contando as páginas de capa e contra-capas. Verifique se o caderno está completo.

- 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.java` onde `codigo` é a letra maiúscula que identifica o problema. Lembre-se que em Java o nome da classe principal deve ser igual ao nome do arquivo.
- Sobre a entrada:
 1. A entrada de seu programa deve ser lida da entrada padrão;

2. 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.
 3. Cada linha, incluindo a última, contém exatamente um caractere final-de-linha.
 4. O final da entrada coincide com o final do arquivo.
- 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.
 - Sobre dúvidas
 1. Utilize a aba clarification para dúvidas da prova. Os juízes podem opcionalmente atendê-lo com respostas acessíveis a todos;

Organização

Coordenadora: Prof^ª. Dr^ª. Maristela Terto de Holanda (CIC/UnB)

Professores Colaboradores: Prof^ª. Dr^ª. Alba Cristina Magalhães Alves de Melo (CIC/UnB), Prof. Edson Alves Da Costa Junior (FCTE/UnB), Prof. Daniel de Paula Porto (CIC/UnB), Prof. Daniel Saad Nogueira Nunes (IFB) , Prof^ª. Aleteia Patrícia Favacho de Araujo Von Paumgartten (CIC/UnB), Prof. Dr. Vinícius Ruela Pereira Borges (CIC/UnB), Prof.^a. Carla Maria Chagas e Cavalcante Koike e Prof^ª. Maria Emilia T. Walter (CIC/UnB).

Juíza-chefe: Isabela Souza Sisnando de Araujo (CIC/UnB)

Alunas Colaboradoras

Adrielly Vitória Costa de Lima (CIC/UnB)	Laryssa de Oliveira Ferreira (CIC/UnB)
Ana Clara Barbosa Borges (FCTE/UnB)	Luana Cruz Silva (CIC/UnB)
Ana Carolina Kinoshita Uehara (CIC/UnB)	Mayara Marques da Silva (CIC/UnB)
Bianca Patrocínio Castro (FCTE/UnB)	Maria Vitória Matos Mourão (FT/UnB)
Brenda Beatriz Maciel da Silva (CIC/UnB)	Vanessa Paixão Costa (CIC/UnB)
Danielly Reis dos Santos (FT/UnB)	Emilly Victoria Bernardes Damiao (Ene/UnB)
Hanne Gabriel Curado (FT/UnB)	Manuela Ferreira Mattos (CIC/UnB)

Setters

Isabela Souza Sisnando de Araujo (CIC/UnB), Adne Moretti Moreira (FCTE/UnB), Arthur Menezes Botelho (CIC/UnB), Belpy Pierre Starling (CIC/UnB), Bruno Vargas de Souza (CIC/UnB), Emerson Luiz Cruz Junior (CIC/UnB), Iasmim de Queiroz Freitas (CIC/UnB), Lucas Gabriel de Oliveira Lima (CIC/UnB), Maxwell Oliveira dos Reis (CIC/UnB) e Pedro Ávila Beneveli (CIC/UnB).

Revisores

Pedro de Carvalho Gallo Pereira (Ene/UnB), João Carlos Gonçalves de Oliveira Filho (CIC/Unb), Prof. Daniel Saad Nogueira Nunes (IFB) e Prof. Dr. Vinícius Ruela Pereira Borges (CIC/UnB).

Agradecimentos

- Neospace
- Grupo Mulheres do Brasil - Brasília
- Grupo UnBalloon

Problema A – A arte da mímica

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

Autor: Belpy Pierre Starling

Mímica Mavi, também conhecida como M.M., é uma artista famosa por um truque peculiar: ela consegue imitar com perfeição qualquer reação das pessoas ao seu redor. Seja uma careta engraçada, uma expressão de espanto ou um simples sorriso, M.M. sempre copia **exatamente** o que vê.



Durante uma de suas apresentações, alguém fez uma reação — e sua missão é ajudar M.M. a imitá-la instantaneamente. Para isso, dado o texto da reação feita pela pessoa, imprima a reação imitada por Mavi.

Entrada

A entrada consiste em apenas uma linha com uma string, contendo a reação R feita pela pessoa ($1 \leq |R| \leq 20$). A reação **não contém espaço**, porém pode conter letras, números e símbolos.

Saída

Imprima a mesma string de reação recebida como entrada, representando a imitação perfeita de M.M..

Exemplo

Entrada	Saída
:)	:)
:3	:3
o(>w<)o	o(>w<)o

Problema B – Batendo Asas

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

Autor: Bruno Vargas de Souza

No município de Patos, existe uma tradição curiosa: durante o Festival das Aves Voadoras, todas as famílias da região treinam suas galinhas para o grande “Duelo do Voo”. Há décadas, uma galinha chamada **Rafinha** vem sendo celebrada como a campeã invicta, pois conseguiu ficar **13 segundos** no ar, um feito praticamente lendário. Dizem que, quando Rafinha voa, o povo todo se reúne, acende fogueiras e canta versos em sua homenagem.



Neste ano, a organização do Festival decidiu criar um sistema automatizado para avaliar, em tempo real, as performances das concorrentes. Cada nova galinha é solta em um campo aberto cheio de bandeirolas coloridas, e um cronômetro registra exatamente por quantos segundos ela permanece voando (ou “flutuando”, conforme as galinhas preferirem). Sua missão é escrever um programa que, dado o tempo de voo X (em segundos) de uma galinha participante, informe se ela já tem um “voo no patamar da rafinha” ou se ainda “precisa de mais asas” para entrar para o hall dos campeões.

Entrada

Um único número inteiro X ($0 \leq X \leq 100$), que indica quantos segundos a galinha ficou voando durante sua apresentação.

Saída

Imprima “voo no patamar da rafinha” se a galinha participante atingiu ou superou a marca de **13 segundos** do recorde de Rafinha, ou “precisa de mais asas”, caso contrário. Escreva tudo em minúsculo e em uma única linha.

Exemplo

Entrada	Saída
8	precisa de mais asas
14	voo no patamar da rafinha

Problema C – Competição de Golf

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

Autor: Lucas Gabriel de Oliveira Lima

Às sextas-feiras, Ana Borges e Bianca costumam se reunir no LINF para estudar, fazer gyms e upsolvings juntas.

Porém, na última sexta-feira, devido a um feriado, o LINF estava fechado, impedindo que elas seguissem sua rotina habitual. Para contornar esse problema, decidiram se reunir no Discord da UnBalloon e fazer o upsolving de algumas questões.

Após terminarem todas as questões separadas, elas ainda queriam passar um tempo juntas na call fazendo algo divertido. Então, consultaram o catálogo de jogos e decidiram jogar uma partida de **Lary Party**.



Lary Party é um jogo de minigolfe, no qual cada mapa possui exatamente n pistas. Em cada pista, a penalidade de uma jogadora é equivalente ao número de jogadas necessárias para acertar a bola no buraco — quanto menos jogadas, menor a penalidade. A vencedora da partida é quem tiver a menor soma de penalidades após percorrer as n pistas.

No entanto, apesar do jogo ser muito divertido, ele possui alguns bugs. Após uma das partidas, embora o jogo tenha registrado corretamente as penalidades de Ana e Bianca em cada pista, ele não exibiu quem venceu. Assim, elas precisam de um programa que determine quem foi a vencedora da partida, ou se houve um empate.

Entrada

A entrada é composta por três linhas:

A primeira contém um número n , que representa o número de pistas do mapa que estão jogando.

A segunda linha contém n inteiros $a_1, a_2, a_3, \dots, a_n$, onde a_i representa a penalidade que **Ana** tomou para completar a pista i .

A terceira linha contém n inteiros $b_1, b_2, b_3, \dots, b_n$, onde b_i representa a penalidade que **Bianca** tomou para completar a pista i .

Restrições:

$$1 \leq n \leq 100$$

$$1 \leq a_i, b_i \leq 100$$

Saída

Seu programa deve imprimir uma única linha, contendo uma das seguintes mensagens (sem as aspas):

“Ana” caso Ana tenha a menor soma de penalidades.

“Bianca” caso Bianca tenha a menor soma de penalidades.

“Empate” caso ambas tenham a mesma soma de penalidades.

Exemplo

Entrada	Saída
6 2 4 3 1 1 2 1 3 5 2 1 2	Ana
5 1 1 2 3 2 2 4 1 1 1	Empate
8 2 3 2 3 2 3 3 4 2 2 2 3 2 3 3 3	Bianca

Problema D – Duda e Sushis!

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

Autor: Isabela Souza Sisnado de Araujo

Duda acabou de sair da Maratona SBC de Programação, onde seu time “Lenhadoras de Segtree” foi eleito o melhor time feminino da América Latina!

Depois de tanto programar, Duda está com muita fome e resolveu jantar em um restaurante japonês! Nesse restaurante, cada peça i de sushi tem um valor A_i , só tem um problema ... as peças passam rápido em uma esteira, por isso, dado uma sequência de sushis, Duda só consegue pegar ou as peças em posições ímpares ou as peças em posições pares nessa sequência.



Por exemplo, dada a sequência $A = [10, 27, 24, 7, 11, 5]$, Duda conseguiria pegar as seguintes combinações de peças:

- Posições ímpares (A_1, A_3 e A_5) : 10, 24, 11
- Posições pares (A_2, A_4 e A_6) : 27, 7, 5

Duda quer obter a maior soma possível dos valores das peças que ela selecionar. No exemplo acima, as duas possibilidades de soma são :

- 45 ($10 + 24 + 11$, se ela pegar os sushis nas posições ímpares);
- 39 ($27 + 7 + 5$, se ela pegar os sushis na posições pares).

Portanto, nesse caso, a melhor opção para Duda é selecionar as peças nas posições ímpares e obter a soma 45.

Agora Duda precisa de sua ajuda para saber, dentre as duas possibilidades, qual fornecerá a maior soma de valores das peças de sushi!

Você pode ajudá-la informando qual a maior soma que ela consegue obter?

Entrada

A primeira linha de entrada contém o inteiro N ($2 \leq N \leq 10^5$), a quantidade total de sushis na sequência.

A segunda linha de entrada contém N inteiros A_i ($0 \leq A_i \leq 100$), cada A_i indicando o valor da peça de sushi na posição i ($1 \leq i \leq N$).

Saída

Imprima um único inteiro: a maior soma dos valores de sushi que Duda consegue atingir, seguindo a restrição definida acima em relação à paridade das posições dos sushis escolhidos.

Se, para as duas possibilidades, a soma for igual, imprima o valor desta soma.

Exemplo

Entrada	Saída
6	45
10 27 24 7 11 5	
3	10
5 10 5	
7	20
7 3 9 1 2 1 2	

Problema E – Este ou leste?

Limite de tempo: 1.5s
Limite de memória: 256MB

Autor: Emerson Luiz Cruz Junior

No Laboratório de Informações Super Secretas (**LINFSS**), Isa, Iasmim e Adrielly trabalham analisando strings compostas por letras minúsculas do alfabeto para descobrir segredos ocultos. Um segredo é revelado pela quantidade de vezes que a **palavra da semana** aparece como substring na string analisada.

Uma **substring** é qualquer sequência de caracteres que pode ser obtida removendo-se zero ou mais caracteres do início e do final da string original.

Adne, a chefe do departamento, decidiu usar um método criativo para revelar a palavra da semana. Ela deu a seguinte dica:

“Onde o sol nasce”



Isa, lembrando-se de uma difícil palavra cruzada que havia resolvido no dia anterior, perguntou se a palavra seria *“este”* ou *“leste”*. Diante dessa dúvida, Adne estabeleceu as seguintes regras para a escolha:

- A palavra correta será aquela que aparecer mais vezes como substring na string analisada S
- Em caso de empate, a palavra *“leste”* será a escolhida
- Se nenhuma palavra aparecer como substring, então a palavra escolhida será *“leste”*

Sua tarefa é ajudar as agentes do LINFSS a determinar qual palavra deve ser escolhida: *“leste”* ou *“este”*, seguindo as regras estabelecidas por Adne.

Entrada

A entrada consiste em duas linhas. A primeira linha apresenta um inteiro n ($4 \leq n \leq 2 \times 10^5$), representando o comprimento da string S . A segunda linha contém a string S , composta por n caracteres minúsculos do alfabeto (a-z).

Saída

Imprima, em uma única linha, a palavra escolhida: “este” ou “leste”.

Exemplo

Entrada	Saída
11 esteouleste	este
7 esteira	este
12 grafodebotas	leste

Problema F – Festival de talentos

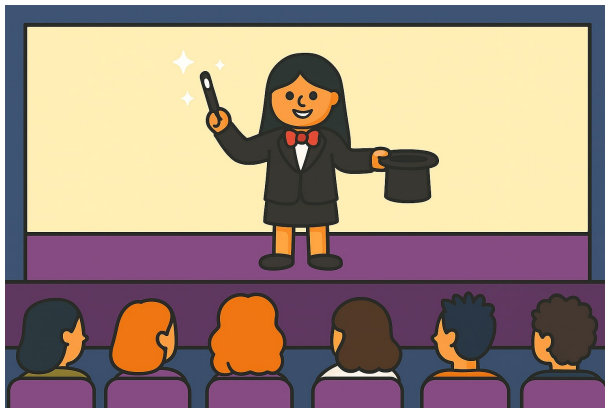
Limite de tempo: 1s
Limite de memória: 256MB

Autor: Adne Moretti Moreira

Chegou o dia do Festival de Talentos da Universidade Nacional de Balloons (UNB), um evento muito esperado pelos estudantes e familiares! É o momento de enaltecer os talentos dos estudantes!

Entre os destaques da noite está Ana Komatsu, uma estudante apaixonada por mágica e famosa por seus truques inesperados. Ela vai se apresentar na Grande Apresentação Final, que é realizada no auditório principal da universidade, esse espetáculo atrai uma enorme plateia, cheia de familiares e amigos. Toda a família de Ana Komatsu está empolgada para vê-la brilhar no palco, porém... Ana Komatsu acordou atrasada, e como todos queriam ir juntos, sua família acabou se atrasando também.

Quando finalmente chegam ao auditório, várias famílias já estão acomodadas (sempre sentadas em grupos consecutivos de cadeiras). Agora, eles precisam da sua ajuda para saber se ainda há cadeiras livres para toda a família Komatsu no auditório, permitindo que eles se sentem juntos e assistam ao grande show de Ana Komatsu.



Considere que o auditório tem N cadeiras alinhadas, representadas por números de 1 a N , que K pessoas na família de Komatsu (sem contar com ela) vão assistir ao show, e que M é a quantidade de famílias de estudantes que já estão sentados no auditório.

Para cada estudante i , a primeira pessoa da família se sentou na cadeira x_i e, dessa família, y_i pessoas foram assistir e se acomodaram consecutivamente a partir da primeira pessoa.

Responda com “SIM”, caso seja possível que toda a família de Ana Komatsu assista ao Festival sentados juntos, ou “NAO”, caso contrário, para que se organizem.

Entrada

A primeira linha contém três inteiros N , M e K ($1 \leq N \leq 10^9, 0 \leq M \leq 10^5, 1 \leq K \leq 10^9$) – Onde N é o número de cadeiras no auditório, M é o número de famílias que já estão sentadas no auditório e K é o número de pessoas da família de Komatsu, sem considerar ela.

Cada uma das próximas M linhas contém dois inteiros x_i e y_i ($1 \leq x_i, y_i \leq 10^9$), referentes à primeira cadeira que a família do estudante i ocupou e a quantas pessoas dessa família irão se sentar. Cada família que chega ao auditório se senta em cadeiras livres, ou seja, que ainda não foram ocupadas por nenhuma outra família.

Saída

Imprima uma única linha contendo “SIM”, se toda a família de Komatsu consegue se sentar de forma consecutiva, ou “NAO”, se tiverem que se separar ou ficar em pé.

Exemplo

Entrada	Saída
5 2 3 1 1 2 1	SIM
6 2 2 1 3 4 2	NAO

Notas

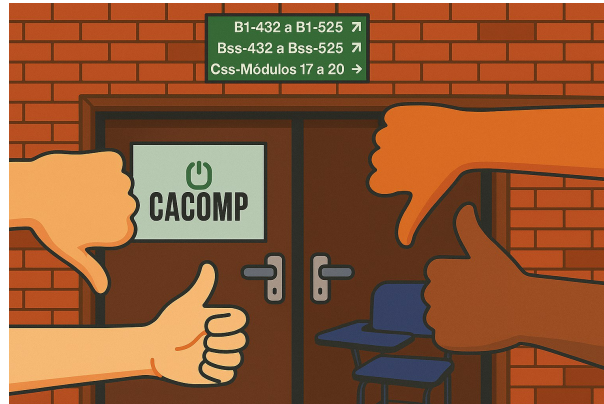
Para o primeiro caso de teste, existem 5 cadeiras no auditório, 2 famílias já acomodadas e 3 pessoas da família de Komatsu que vão se sentar. A primeira família apenas se sentou na cadeira 1, e a segunda família apenas na cadeira 2, então ainda estão vazias as cadeiras 3, 4 e 5 e a família de Komatsu poderá sentar junto.

Problema G – Gerenciando o CA

Limite de tempo: 2s
Limite de memória: 256MB

Autor: Pedro Ávila Beneveli

O Clube Aplicado de Computação (CAComp) é um espaço muito popular entre os alunos dos cursos de tecnologia, já que é um espaço onde eles não só podem aplicar o que aprendem nas aulas, mas também descansar entre elas.



Sara, a nova presidente do CA, percebeu que, devido ao grande fluxo de pessoas, a gestão não estava conseguindo controlar se todos os alunos já tinham saído do CA, ocasionando em algumas pessoas ficando presas. Por isso, ela decidiu anotar exatamente quando algum aluno entrou e quando algum saiu do CA, simbolizando pelos símbolos ‘(’ e ‘)’, respectivamente. Além disso, ela percebeu um comportamento curioso entre os frequentadores do CA: antes de sair, um aluno sempre vai esperar todos os alunos que chegaram depois dele sair, ou seja, quem saiu do CA em um dado instante sempre foi o último aluno a entrar, dentre todos os estudantes dentro do CA.

Como Sara está muito ocupada, ela decidiu pedir sua ajuda. Dada as anotações de tamanho N de Sara $s_1s_2s_3\dots s_N$, onde s_i indica se alguém entrou ou saiu no instante i , Sara irá fazer Q perguntas: Em que instante o aluno que entrou no instante X saiu do CA?

Entrada

A primeira linha da entrada consiste em dois inteiros N e Q ($2 \leq N \leq 2 \cdot 10^5$, $1 \leq Q \leq 2 \cdot 10^5$), o tamanho das anotações da Sara e o número de perguntas, respectivamente.

A segunda linha da entrada consiste de uma string de N caracteres. É garantido que:

- Todos os caracteres são ‘(’ ou ‘)’, simbolizando a entrada ou a saída de um aluno, respectivamente.
- Para todo instante de tempo, há uma quantidade maior ou igual de caracteres ‘(’ do que a quantidade de caracteres ‘)’.

As próximas Q linhas da entrada consistem de um único inteiro X_i cada ($1 \leq X_i \leq N$), o X correspondente à i -ésima pergunta da Sara. É garantido que um aluno entra no CA no instante X_i .

Saída

A saída deve consistir de Q linhas, onde a saída na i -ésima linha deve ser a resposta à i -ésima pergunta da Sara:

- Se o aluno não saiu do CA, responda “Ficou Preso” (sem as aspas).
- Se o aluno saiu do CA, responda com um inteiro correspondendo ao instante que ele saiu do CA.

Exemplo

Entrada	Saída
5 3	4
((()))	Ficou Preso
3	5
1	
2	
10 4	2
()()((()))	9
1	10
6	5
3	
4	

Problema H – Cabos do LINF

Limite de tempo: 4s
Limite de memória: 256MB

Autor: Iasmim de Queiroz Freitas

Durante os preparativos para a Competição Feminina de Programação da UnB, a professora Maristela percebeu que os cabos de rede do laboratório de informática (LINF) estavam uma bagunça. Muitos cabos estavam conectados de forma incorreta — alguns entre dois computadores, outros entre dois pontos de rede ou até com uma ponta solta. Isso fez com que muitos computadores ficassem sem acesso à internet.

Para resolver o problema, a professora decidiu desconectar todos os cabos e pediu para o Lucas Sala ajudá-la com isso. Sem querer, ele acabou quebrando a ponta de todos os cabos ao removê-los, sendo necessário comprar novos.

Sabe-se que o metro do cabo custa 1 real e que, para que seja possível comprar os cabos a tempo da competição, o valor de cada um deles não pode ultrapassar um limite x . Caso contrário, seria necessário abrir um processo formal de compra com licitação — algo demorado, inviável antes do evento. Felizmente, como todas as participantes estão aqui hoje, a professora conseguiu realizar a compra de forma que o valor do cabo mais longo adquirido foi exatamente igual ao limite x .

O LINF é um laboratório singular, com um comprimento incomensurável. A cada metro a partir da porta pode existir um ponto de rede e /ou uma entrada de rede de um computador. Portanto, o comprimento de um cabo conectado a um ponto de rede na posição i e uma entrada de rede na posição j deve ser $|i - j| + 1$ metros — o metro extra é necessário para garantir a flexibilidade do cabo.

A professora Maristela comprou e organizou os cabos de forma ótima, conectando o maior número possível de computadores a pontos de rede de forma a minimizar o comprimento do maior cabo utilizado. Com base nisso, você deve descobrir o valor do limite x , sabendo que ele é igual ao valor do maior cabo adquirido.

Entrada

A primeira linha contém dois inteiros n e m ($1 \leq n, m \leq 5 \cdot 10^5$), representando, respectivamente, o número de pontos de rede disponíveis e o número de computadores funcionando.

A segunda linha contém n inteiros $R_1, R_2, \dots, R_n$ em ordem crescente ($R_i < R_{i+1}$), onde cada R_i ($1 \leq R_i \leq 10^9$) representa a posição em metros do i -ésimo ponto de rede.

A terceira linha contém m inteiros $C_1, C_2, \dots, C_m$ em ordem crescente ($C_j < C_{j+1}$), onde cada C_j ($1 \leq C_j \leq 10^9$) representa a posição em metros da j -ésima entrada de rede de um computador.

Saída

Imprima um único inteiro x , representando o menor valor possível para o maior cabo adquirido, sendo que a maior quantidade de computadores possível foi conectada a pontos de rede.

Exemplo

Entrada	Saída
2 4	2
1 10	
2 3 4 11	
4 2	5
1 10 20 30	
5 28	
3 3	1
1 2 3	
1 2 3	

Notas

No primeiro caso de teste, como há dois pontos de rede e quatro entradas de rede de computadores, a maior quantidade possível de computadores conectados é dois. Podemos conectar primeiro ponto e a primeira entrada de rede com um cabo de $|1 - 2| + 1 = 1 + 1 = 2$ metros e o segundo ponto e a quarta entrada com um cabo de $|10 - 11| + 1 = 2$ metros. Em qualquer outra configuração o valor do maior cabo excede 2 reais, então o menor valor possível para o maior cabo e o limite x são de fato 2.

No segundo caso de teste, a configuração que minimiza o tamanho do maior cabo é conectando de novo o primeiro ponto e primeira entrada (5 metros) e o segundo ponto e a quarta entrada (3 metros), de forma que o valor do limite x é 5 reais.

No terceiro caso de teste, devemos conectar todos os pontos e entradas no mesmo metro com cabos de tamanho 1 para conectar o maior número de computadores à rede de forma a minimizar o valor do maior cabo adquirido.

Problema I – O Multiverso do 110

Limite de tempo: 2s
Limite de memória: 512MB

Autor: Maxwell Oliveira dos Reis



Transição entre multiversos

Finalmente chegou o grande dia da Maratona Feminina de Programação! Há várias pessoas se encaminhando para o local de prova neste exato momento. Em especial, a Nikolle está chegando na rodoviária de Brasília e se prepara para pegar o 110, o ônibus que a levará para a UnB.

Se preparando para embarcar, a Nikolle é avisada de que o 110 não cobraria a passagem. Porém, nem tudo são flores... O cobrador avisa que a partir de agora apenas as pessoas escolhidas poderão embarcar e, para que isso aconteça, é necessário derrotar o cobrador no seu próprio jogo:

O Multiverso do 110!

O jogo funciona da seguinte forma:

- O cobrador possui um mapa da cidade com N marcações em pontos (x, y) distintos.
- O jogador atua como motorista do 110 e seu objetivo é passar por todos os pontos marcados ao menos uma vez, usando o menor tempo possível.
- O jogador pode escolher qualquer um dos N pontos para iniciar sua jornada.
- A viagem de um ponto A , localizada em (x_A, y_A) para um ponto B , localizada em (x_B, y_B) leva $\sqrt{abs(x_A - x_B)^2 + abs(y_A - y_B)^2}$ minutos.
- O jogador inicia uma contagem $Q = 0$ e o incrementa sempre que chega pela primeira vez em um ponto, incluindo o ponto inicial.
- Quando o contador Q chega a K , o jogador recebe um bilhete especial e reinicia sua contagem.
- Ao usar o bilhete, o jogador deve escolher um ponto qualquer. O bilhete leva o jogador para outro universo, onde seu 110 possui um itinerário diferente e está, nesse momento, exatamente no ponto desejado. Porém, a transição entre universos leva T minutos.
- Um bilhete possui uso único e pode ser utilizado a qualquer momento a partir do seu recebimento. Além disso, o uso dos bilhetes não é obrigatório.

A Nikolle não pode pensar nesse problema agora, visto que isso pode deixá-la cansada e atrapalhar seu desempenho na MFP. Por isso, ela pediu sua ajuda para conseguir chegar ao local de prova. Será que você consegue zerar **O Multiverso do 110** e ajudá-la?

Entrada

A primeira linha do input consiste dos inteiros N , K ($1 \leq K \leq N \leq 16$) e T ($1 \leq T \leq 10^9$), sendo respectivamente o número de pontos marcados no mapa, o número de visitas necessárias para adquirir um bilhete e o tempo necessário para completar a viagem após o uso de um bilhete.

As próximas N linhas contêm dois inteiros, x e y ($-10^9 \leq x, y \leq 10^9$), indicando as posições de cada ponto.

Saída

Imprima um único número real que representa a menor quantidade de minutos necessária para zerar o jogo.

Sua resposta será considerada correta se o erro relativo ou absoluto não ultrapassar 10^{-6} . Formalmente, seja a a sua resposta e b a resposta do juiz. Sua resposta será aceita se, e somente se, $\frac{|a-b|}{\max(1, |b|)} \leq 10^{-6}$

Exemplo

Entrada	Saída
1 1 100 1 1	0.0000000000
2 2 99 1 1 10000 10000	14140.7214101686
4 3 1 1 1 2 2 3 3 -10000000000 10000000000	3.8284271247
8 2 3638 7038 2215 -7228 -8341 6940 -3108 3178 -5244 -895 -2767 9095 9176 -700 8032 3833 -980	28298.5259189741

Notas

Para o primeiro exemplo, a jogadora deve iniciar no único ponto existente, totalizando 0 minutos.

No segundo exemplo a jogadora só recebe um bilhete após visitar dois pontos, onde ela já teria zerado o jogo.

No terceiro exemplo é ótimo visitar os pontos na ordem 1, 2, 3, receber o bilhete e usar para chegar no último ponto.