Seletiva UnB 2019

Caderno de Problemas

18 de maio

(Este caderno contém 12 problemas)

Comissão Organizadora:
Prof. Edson Alves da Costa Júnior (UnB/FGA)
Prof. Guilherme Novaes Ramos (UnB)
Prof. Vinicius Ruela Pereira Borges (UnB)
Prof. Daniel Saad Nogueira Nunes (IFB)
José Marcos da Silva Leite
Lucas Vasconcelos Mattioli
Pedro Henrique Lima Ferreira

Universidade de Brasília
http://www.unb.br

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 conveção 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
Radares

Limite de tempo: 1s

Autor: Vinícius Ruela Pereira Borges

Uma rodovia está sendo preparada para receber radares móveis para flagrar condutores que estiverem dirigindo acima do limite de velocidade. A tecnologia desses radares permite identificar os condutores infratores ao calcular a velocidade média de um veículo em relação a uma distância percorrida, dada pelo posicionamento de dois radares na rodovia. Os agentes de trânsito pretendem realizar um estudo do tráfego desta rodovia e consideraram empregar inicialmente apenas um par de radares móveis para analisarem as velocidades dos condutores na rodovia.

Uma questão relevante que os agentes precisam resolver é determinar o melhor posicionamento dos radares pela rodovia de extensão total N. A estratégia consiste em posicionar aleatoriamente os dois radares pela rodovia, sendo que o primeiro radar é posicionado no quilômetro l da rodovia, enquanto que o segundo radar é posicionado no quilômetro r, em que $l < r$. Um veículo tem sua velocidade média m calculada de acordo com as velocidades registradas nas passagens pelos radares e a distância entre eles. Para ampliar a amostragem e a representatividade das aferições, os agentes se deslocam pela rodovia, alterando o posicionamento dos dois radares durante um determinado período de tempo.

Os agentes de trânsito estão interessados em identificar o comprimento do trecho da rodovia que apresentou a maior quantidade de infrações, para que eles possam realizar *blitzes* educativas ou de fiscalização. Como essa tarefa é muito trabalhosa, ajude os agentes e escreva um programa que determine o comprimento (em quilômetros) do maior trecho da rodovia que registrou a maior quantidade infrações por excesso de velocidade.

Entrada

A primeira linha da entrada consiste em três inteiros $2 \leq N \leq 10^4$, $1 \leq Q \leq 10^6$ e $60 \leq L \leq 170$, que representam a extensão total da rodovia (em quilômetros), a quantidade total de aferições pelo par de radares e o limite de velocidade da rodovia, respectivamente. Em seguida são definidas Q linhas, em que cada linha i descreve três inteiros $1 \leq l_i \leq N-1$, $l_i < r_i \leq N$ e $1 \leq m_i \leq 220$ separados por um espaço branco, associados aos quilômetro da rodovia de posicionamento do radar l, quilômetro da rodovia de posicionamento do radar r e a velocidade média calculada.

Saída

Se existirem motoristas infratores durante o período de tempo da fiscalização, seu programa deve imprimir um inteiro indicando o comprimento (em quilômetros) do maior trecho da rodovia em que se cometeu a maior quantidade de infrações. Caso não existam motoristas infratores, seu programa deve imprimir -1.

Exemplo

Entrada	Saída

`10 4 100`
`3 6 111`
`4 8 109`
`1 5 88`
`2 5 105`
	`2`

`5 2 120`
`1 3 95`
`2 5 119`
	`-1`

`10 4 100`
`1 3 107`
`8 10 130`
`5 7 90`
`4 6 91`
	`3`

Notas

No primeiro exemplo, a maior quantidade de infrações (no total, 3 infrações) foi cometida no trecho compreendido pelos quilômetros 4 e 5, portanto o comprimento deste trecho é 2. No segundo exemplo, nenhum condutor excedeu o limite de velocidade da rodovia em todas as aferições, sendo a resposta igual a -1. No terceiro exemplo, os trechos compreendidos pelos quilômetros 1 e 3 e 8 e 10 foram aqueles que registraram a maior quantidade de infrações (no total, 1 infração apenas) e ambos os trechos possuem comprimento 3.

Problema B
Presente de Dia das Mães

Limite de tempo: 1s

Autor: Daniel Saad Nogueira Nunes

No dia das mães, Cristiano resolveu fazer uma surpresa para todas as mães de sua família. Ele comprou várias caixas de chocolate para presentear cada mãe com uma quantidade de chocolates de forma que:

1. cada mãe receba a mesma quantidade de chocolates,
2. uma mãe não pode receber chocolates de caixas diferentes, por conta da diferença de sabores, e
3. a quantidade de chocolates presenteados a cada mãe é a maior possível.

Como Cristiano quer ser o mais justo possível e não consegue fazer esta distribuição sozinho devido à grande quantidade de mães e a complexidade dos critérios, você ficou responsável por ajudá-lo.

Entrada

A primeira linha da entrada possui dois números inteiros N ($1 \leq N \leq 10^5$) e M ($1 \leq M \leq 10^9$), separados por um espaço em branco, representando a quantidade de caixas de chocolate e o número de mães, respectivamente.

A próxima linha contém N inteiros $V_0, V_1, \ldots, V_i, \ldots, V_{N-1}$, separados por um espaço em branco, de modo que V_i ($0 \leq V_i \leq 10^9$) indica a quantidade de chocolates presentes na i-ésima caixa.

Saída

Seu programa deverá imprimir a maior quantidade de chocolate possível distribuída a cada mãe de acordo com os critérios apresentados.

Exemplo

Entrada	Saída
`3 3` `1 2 3`	`1`
`3 4` `8 2 1`	`2`
`3 3` `5 9 2`	`4`

Notas

No primeiro caso de teste, exitem três caixas de chocolate com 1, 2 e 3 chocolates respectivamente e 3 mães. Nesse caso, o máximo de chocolates presenteados a cada mãe é 1. Um chocolate de cada caixa é uma possível configuração que atende os critérios.

No segundo caso de teste, existem três caixas de chocolate com 8, 2 e 1 chocolates respectivamente e 4 mães. Nesse caso, o máximo de chocolates presenteados a cada mãe é 2. Uma possibilidade de configuração é distribuir 2 chocolates da primeira caixa a todas as mães.

No terceiro caso de teste, existem três caixas de chocolate com 5, 9 e 2 chocolates respectivamente e 3 mães. Nesse caso, o máximo de chocolates presenteados a cada mãe é 4. Uma possibilidade de configuração é distribuir 4 chocolates da primeira caixa a uma mãe e dividir 8 chocolates da segunda caixa entre as duas mães restantes.

Problema C
Boca de Urna

Limite de tempo: 1s

Autor: Edson Alves da Costa Junior

Énoislândia é um pequeno município brasileiro com N habitantes. É de conhecimento de todos os moradores do município que, devido à relações de parentesco, amizade, etc., o eleitor a votará no mesmo candidato que o eleitor b.

Um instituto de pesquisa conseguiu apurar, logo após o término da votação, os votos de M eleitores. Este instituto deseja saber o mínimo de votos que ainda devem ser verificados para que seja possível determinar o resultado da eleição com 100% de acurácia, isto é, para determinar o voto individual de todos os eleitores.

Entrada

A primeira linha da entrada contém o inteiro N ($1 \leq N \leq 10^5$) e outro inteiro Q ($0 \leq Q \leq \min(10^5, N(N-1)/2)$), separados por um espaço em branco, onde Q é a quantidade de relações conhecidas.

Cada uma das Q linhas seguintes contém dois inteiros a ($1 \leq a \leq N$) e b ($1 \leq b \leq N, b \neq a$), separados por um espaço em branco, indicando que o eleitor a votará no mesmo candidato que o eleitor b.

A linha seguinte contém o inteiro M ($0 \leq M \leq N$), que indica o número de votos que foram apurados logo após o término da eleição.

Por fim, a última linha contém M inteiros e_i ($1 \leq e_i \leq N, e_i \neq e_j$ se $i \neq j$), separados por um espaço em branco, indicado os eleitores que declararam seus votos ao instituto de pesquisa.

Saída

Imprima, em uma linha, o número mínimo de votos que ainda devem ser conhecidos para que o resultado da eleição possa ser determinado com 100% de acerto.

Exemplo

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

Notas

No primeiro caso, é sabido que 2 e 3 votam no mesmo candidato, mas o instituto apurou apenas o voto de 1. Assim, é preciso saber em quem 2 ou 3 votou: conhecido o voto de um dos dois já é o suficiente para saber os votos de todos os três eleitores.

No segundo caso, o instituto apurou o voto do eleitor 2, e a partir dele pode deduzir os votos de 4 e de 5. Ao apurar o voto de 3, também determinou o voto de 1. Logo o instituto já tem informações para saber os votos de todos os cinco eleitores.

Problema D
Lista de Exercícios

Limite de tempo: 1s

Autor: Edson Alves da Costa Junior

O professor de matemática precisa gerar uma lista de exercícios sobre matrizes e determinantes. Ele quer criar uma matriz $A_{2\times 2}$ cujos coeficientes sejam inteiros distintos, diferentes de zero e cujos valores absolutos sejam menores ou iguais a 1.000. Além disso, o determinante da matriz deve ser um inteiro D.

Auxilie o professor nesta tarefa, exibindo uma matriz que atenda os requisitos dados.

Entrada

A entrada consiste em uma única linha, contendo o inteiro D ($-1.000 \leq D \leq 1.000$).

Saída

A imprima, na primeira linha da saída, os coeficientes a_{11} e a_{12} da matriz A, separados por um espaço em branco.

Na segunda linha imprima os coeficientes a_{21} e a_{22} da matriz A, separados por um espaço em branco.

Se houver mais de uma matriz A que atenda os requisitos apresentados, imprima qualquer uma delas.

Exemplo

Entrada	Saída
1	3 4 5 7
3	3 -2 9 -5
4	3 2 7 6

Notas

No primeiro caso, o determinante da matriz é igual a $(3 \times 7) - (4 \times 5) = 21 - 20 = 1$. Note que os quatro coeficientes da matriz são distintos e diferentes de zero.

No segundo caso, $(3 \times (-5)) - (9 \times (-2)) = -15 + 18 = 3$.

No terceiro caso, $(3 \times 6) - (2 \times 7) = 18 - 14 = 4$.

Problema E
Construindo Estradas

Limite de tempo: 1s

Autor: Pedro Henrique Lima Ferreira e José Marcos da Silva Leite

Existem **N** cidades no país que José governa. Cada cidade tem um índice diferente de 1 a **N**, e duas cidades são consideradas vizinhas se a diferença de seus índices é 1. Deste modo, todas as cidades tem 2 cidades vizinhas, com exceção da $cidade_1$ e da $cidade_N$. Em outras palavras, a $cidade_1$ é vizinha da $cidade_2$, a $cidade_2$ é vizinha das $cidade_1$ e $cidade_3$ e assim por diante; exceto pela $cidade_N$ que tem somente a $cidade_{N-1}$ como vizinha.

José deseja construir uma estrada entre estas cidades. Para isso, tem a sua disposição os **N** melhores funcionários do país! Ele pretende colocar um em cada cidade, onde funcionários de cidades vizinhas irão trabalhar juntos para construir as estradas. Por exemplo:

- o funcionário da $cidade_1$ irá trabalhar junto com o funcionário da $cidade_2$ na construção da estrada que liga a $cidade_1$ à $cidade_2$;

- o funcionário da $cidade_2$ ajudará na construção da estrada que liga a $cidade_1$ à $cidade_2$ (junto com o funcionário da $cidade_1$) e na estrada que liga a $cidade_2$ à $cidade_3$ (junto com o funcionário da $cidade_3$), e assim por diante.

Cada estrada é feita somente com um tipo de asfalto e há somente 2 tipos de asfalto disponíveis para a construção das estradas, o tipo **A** e o tipo **B**. Um $funcionário_i$ cobra um preço a_i para construir uma estrada com asfalto do tipo **A** e cobra um preço b_i para construir uma estrada com asfalto do tipo **B**. Logo, considerando que o $funcionário_i$ está na cidade **K** e o $funcionário_j$ está na cidade **K+1**, para construir uma estrada do tipo **A** entre estas cidades é gasto $a_i + a_j$.

José não se importa com o tipo dos asfaltos das estradas, ele apenas deseja pagar o mínimo possível aos funcionários para ter estradas entre todas as cidades. Isto significa que entre as cidades **K** e **K+1** (considerando novamente que o $funcionário_i$ está na cidade **K** e o $funcionário_j$ está na cidade **K+1**), José pagará o valor de $min(a_i + a_j, b_i + b_j)$.

Dados os **N** funcionários e o preço que cada um cobra para construir estradas com diferentes tipos de asfaltos, ajude José a descobrir qual o menor preço que ele pode pagar para construir estradas entre todas as cidades.

Entrada

A primeira linha contém um número **N** ($2 \leq N \leq 1000$), indicando o número de funcionários. As próximas **N** linhas contém dois números, a_i e b_i ($1 \leq a_i, b_i \leq 10^5$), separados por um espaço, indicando os preços que o i-ésimo funcionário cobra para construir estradas com asfalto do tipo **A** e do tipo **B**, respectivamente.

Saída

Imprima um único inteiro indicando qual o menor preço que José irá pagar para ter estradas entre todas as cidades vizinhas.

Exemplo

Entrada	Saída
4 4 3 6 8 6 8 5 4	28
5 3 4 2 4 5 3 4 3 2 5	21
5 2 2 3 3 4 4 5 5 6 6	29

Notas

No primeiro exemplo, José tem 4 funcionários:

$funcionário_1$, que cobra 4 e 3 dinheiros para construir estradas com asfalto do tipo a e tipo b, respectivamente:

- $funcionário_2$, que cobra 6 e 8.
- $funcionário_3$, que cobra 6 e 8.
- $funcionário_4$, que cobra 5 e 4.

Neste caso, José vai colocar os funcionários nas N cidades na seguinte ordem: 2, 1, 4, 3. Ou seja:

- $cidade_1$: $funcionário_2$
- $cidade_2$: $funcionário_1$
- $cidade_3$: $funcionário_4$
- $cidade_4$: $funcionário_3$.

Deste modo, José pagará:

- 10 para fazer uma estrada (tipo a) entre $cidade_1$ e $cidade_2$.
- 7 dinheiros para fazer uma estrada (tipo b) entre $cidade_2$ e $cidade_3$.
- 11 dinheiros para fazer uma estrada (tipo a) entre $cidade_3$ e $cidade_4$.

Logo, no total, José pagará 28 para ter todas as estradas.

No terceiro exemplo, a ordem dos funcionários nas 5 cidades será 5, 1, 2, 3, 4 e o preço final será de 29.

Problema F
Espetinho do Barbosinha

Limite de tempo: 1s

Autor: Daniel Saad Nogueira Nunes

O *Espetinho do Barbosinha* se tornou bastante popular na cidade do Gama devido ao baixo preço e alta qualidade das "jantinhas" que são oferecidas.

Em um certo momento, Barbosinha se deparou com um problema. A procura por suas "jantinhas" era tão grande que ele ficou preocupado em não conseguir atender todos os seus clientes de maneira adequada devido à falta de cadeiras. Neste esquema, cada cliente poderia, através de um aplicativo, inserir o horário estimado de chegada e saída. Dessa forma, Barbosinha saberia de antemão o número de cadeiras que ele precisaria deixar disponível e tomar as devidas providências.

Como o formulário preenchido pelos clientes obteve um grande número de respostas, Barbosinha não estava conseguindo mensurar corretamente o número mínimo de cadeiras que ele necessitaria e pediu ajuda a você para automatizar esta tarefa. Dado os horários estimados de chegada e saída dos clientes em um determinado dia, você deverá construir um programa que compute o número mínimo de cadeiras que Barbosinha precisará providenciar aos seus clientes. Considere que uma cadeira está reservada a cada cliente durante a sua estadia e portanto não pode ser usada por outro cliente neste período.

Entrada

A primeira linha da entrada possui um inteiro N ($1 \leq N \leq 10^5$) contendo o número de clientes que preencheram o formulário.

As N linhas seguintes contém, cada uma, duas strings, separadas por espaço, no formato `HH:MM:SS`, indicando respectivamente o tempo de chegada e saída de cada cliente. Considere que as horas seguem o padrão 24h e que um cliente não pode preencher o tempo de saída em uma dia diferente do dia de chegada.

Saída

Seu programa deverá imprimir o número mínimo de cadeiras que deverão ser disponibilizadas para atender todos os clientes.

Exemplo

Entrada	Saída
2 12:00:00 12:30:00 13:00:00 13:30:00	1
3 10:00:00 10:30:00 10:30:01 11:00:00 10:00:00 12:00:00	2
6 08:00:00 10:00:00 12:00:00 16:00:00 09:00:00 13:00:00 18:00:00 23:00:00 15:00:00 20:00:00 08:00:00 22:00:00	3

Notas

No primeiro caso de teste, os dois clientes não estão no espetinho do Barbosinha no mesmo momento, portanto apenas 1 cadeira é necessária.

No segundo caso de teste, o primeiro e segundo clientes não se topam, mas o terceiro cliente estará no espetinho do Barbosinha em um momento em comum com o primeiro e segundo clientes.

No terceiro caso de teste, existirão momentos em que 3 clientes estarão no espetinho do Barbosinha simultaneamente: especificamente das 9h às 10h, ou das 12h às 13h e das 15h às 16h, ou das 18h às 20h. Qualquer outro horário possuirá uma quantidade menor de clientes.

Problema G
Propagação de Worms

Limite de tempo: 1s

Autor: Daniel Saad Nogueira Nunes

Um *worm* é uma categoria de *malware* (programa intencionalmente nocivo) que é capaz de se propagar autonomamente via algum meio de comunicação ao explorar vulnerabilidades das máquinas hospedeiras. Para dificultar ou impedir a sua identificação por meio de *softwares* de detecção e prevenção de intrusão ou antivírus, *worms* utilizam diversas estratégias, e uma delas é agir furtivamente.

Um *worm* em especial, conhecido como `zapzap.w32`, atua furtivamente pela rede ao se propagar apenas para as máquinas vizinhas à máquina infectada. Feito isso, ele interrompe sua propagação sem infectar os vizinhos dos vizinhos.

Você, como futuro analista de segurança, está preocupado com este *worm* e quer saber qual o mínimo número de máquinas que ele deve infectar para se propagar a todas as outras. Seu objetivo com esse estudo é justamente fortalecer a segurança nestes pontos críticos.

Entrada

A primeira linha da entrada possui dois inteiros N ($1 \leq N \leq 18$) e M ($0 \leq M \leq (N \cdot (N-1)/2)$), separados por um espaço, que representam, respectivamente, o número de máquinas e o número de conexões entre essas máquinas.

Cada uma das próximas N linhas compreende uma única palavra que possui apenas letras minúsculas entre '`a`' e '`z`', com no máximo 20 caracteres, que identifica cada máquina na rede.

As próximas M linhas descrevem, cada, um par de palavras, A e B, separadas por espaço, indicando que existe um enlace *full-duplex* entre as máquinas A e B.

É garantido que cada máquina tem um identificador único. Assuma que uma máquina não pode estar conectada a ela mesma e que existe apenas uma conexão direta entre duas máquinas quaisquer.

Saída

Seu programa deverá imprimir duas linhas. A primeira linha deve possuir um inteiro que representa o número mínimo de máquinas a serem infectadas para o `zapzap.w32` se propagar a todas as outras de acordo com o seu *modus operandi*. Já a segunda linha deverá possuir os nomes dos hospedeiros, separados por espaço, das máquinas infectadas inicialmente pelo *worm*.

Na possibilidade de múltiplas soluções atenderem as restrições do enunciado, qualquer uma pode ser utilizada como resposta.

Exemplo

Entrada	Saída

`2 1`
`asterix`
`fiona`
`asterix fiona`
	`1`
	`asterix`

`4 4`
`phi`
`psi`
`chi`
`mu`
`phi psi`
`psi chi`
`chi mu`
`mu phi`
	`2`
	`phi psi`

`6 4`
`sun`
`mars`
`jupyter`
`saturn`
`neptune`
`pluto`
`sun saturn`
`sun mars`
`sun jupyter`
`sun neptune`
	`2`
	`sun pluto`

Notas

No primeiro caso de teste, se o *worm* infectar qualquer uma das duas máquinas ele conseguirá se propagar a outra.

No segundo caso de teste, basta o *worm* infectar inicialmente as máquinas `phi` e `psi` para que ele consiga infectar toda a rede. Outra possibilidade seria o *worm* infectar as máquinas `phi` e `chi`.

No último caso de teste, se o *worm* infectar inicialmente apenas o hospedeiro `sun`, ele conseguirá infectar `mars`, `jupyter`, `saturn` e `neptune`, entretanto, não conseguirá infectar `pluto`. Logo, para que todos os hospedeiros sejam infectados, é necessário que o *worm* contamine inicialmente `sun` e `pluto`.

Problema H
Quantos Caminhos?

Limite de tempo: 2s

Autor: Lucas Vasconcelos Mattioli

Falcãozinho, seu amigo, quer saber quantos caminhos que não passam por buracos existem no gramado entre o ponto de ônibus e sua escola. Para tanto ele fornece um *grid* com N linhas, M colunas e K buracos. O caminho é sempre da parada de ônibus, o ponto $(1,1)$, para a entrada da escola, ponto (N,M) - portanto Falcãozinho se move somente para a direita e para baixo.

O ponto (i,j) representa a intersecção entre a linha i e a coluna j. Ir para a direita a partir de um ponto (i,j) significa ir para o ponto $(i,j+1)$; ir para baixo a partir de um ponto (i,j) significa ir para o ponto $(i+1,j)$. Cada buraco está localizado em um ponto (i,j) distinto. Dois caminhos A e B são ditos diferentes caso suas listas de pontos visitados sejam diferentes.

Entrada

A primeira linha contém dois inteiros separados por um espaço N e M $(1 \leq N, M \leq 10^5, N+M > 2)$, representando a quantidade de linhas e a quantidade de colunas do *grid*, respectivamente. A segunda linha contém um inteiro K $(0 \leq K \leq 1000)$ representando a quantidade de buracos no *grid*. Cada uma das próximas K linhas contém dois inteiros x e y $(1 \leq x \leq N,\ 1 \leq y \leq M)$, representando, respectivamente, a linha e a coluna do ponto em que um buraco se encontra. É garantido que não existem dois buracos num mesmo ponto e que não há buracos nos pontos $(1,1)$ e (N,M).

Saída

A saída deve ser composta por um inteiro representando a quantidade de caminhos do ponto $(1,1)$ ao ponto (N,M) que não passam por qualquer buraco. Como a resposta pode ser muito grande, imprima o resto da divisão deste valor por 10^9+7.

Exemplo

Entrada	Saída
2 2 0	2
3 3 1 1 2	3
888 347 6 9 10 1 88 776 334 551 292 888 222 347 347	686682579

Notas

No primeiro caso, os dois possíveis caminhos são: (1, 1) $>$ (1, 2) $>$ (2, 2) e (1, 1) $>$ (2, 1) $>$ (2, 2).

No segundo caso, nenhum caminho pode passar por (1, 2). Assim, a resposta são todos os 3 caminhos começando em (2, 1), porque é a única opção a partir do ponto (1, 1).

Problema I
Sequência Binomial Central

Limite de tempo: 1s

Autor: Edson Alves da Costa Junior

Seja $a_n = \{a_1, a_2, \ldots\}$ a sequência dos elementos centrais do triângulo de Pascal, isto é, $a_i = \binom{i}{\lfloor i/2 \rfloor}$, com $i = 1, 2, \ldots$. Os dez primeiros termos desta sequência são $1, 2, 3, 6, 10, 20, 35, 70, 126, 252$.

Dado um primo p, determine quantos, dentre os N primeiros elementos desta sequência, que são múltiplos de p.

Entrada

A entrada consiste em uma única linha contendo os inteiros N e p ($1 \leq N \leq 2\times 10^5, 2 \leq p \leq 10^9 + 7$), separados por um espaço em branco.

Saída

Imprima, em uma linha, o número de múltiplos de p entre os N primeiros termos da sequência a_n.

Exemplo

Entrada	Saída
`3 2`	`1`
`5 3`	`2`
`10 7`	`4`

Notas

No primeiro caso temos apenas um múltiplo de 2 entre os três primeiros termos: $a_2 = 2 = 2 \times 1$.

No segundo caso temos dois múltiplos de 3 entre os cinco primeiros termos: $a_3 = 3 = 3 \times 1$ e $a_4 = 6 = 3 \times 2$.

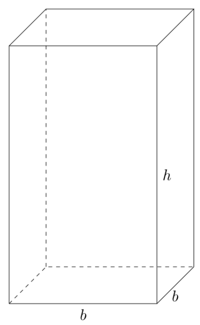
No terceiro caso temos $a_7 = 35 = 7 \times 5$, $a_8 = 70 = 7 \times 10$, $a_9 = 126 = 7 \times 18$ e $a_{10} = 252 = 7 \times 36$.

Problema J
Fabricação de Caixas

Limite de tempo: 1s

Autor: Edson Alves da Costa Junior

Uma empresa do ramo de embalagens deseja construir uma caixa no formato de um prisma retangular cuja base é um quadrado de lado b cm e com altura h cm. Esta caixa deve ter um volume igual V cm^3 e a área da superfície, contando a tampa e a base e as laterais, deve ser igual a A cm^2.



Determine dois inteiros positivos b e h de modo que a caixa tenha volume V e área superficial A. É garantido que há pelo menos uma solução para cada entrada.

Entrada

A entrada consiste em uma única linha, contendo os valores dos inteiros V e A ($1 \leq V \leq 10^{14}, 1 \leq A \leq 10^{15}$), separados por um espaço em branco.

Saída

Imprima, em uma linha, os inteiros positivos b e h, separados por um espaço em branco. Se houver mais de um par que satisfaça as condições apresentadas, imprima qualquer um deles.

Exemplo

Entrada	Saída
`20 48`	`2 5`
`216 216`	`6 6`
`147 182`	`7 3`

Problema K
Máquina de Refrigerante

Limite de tempo: 1s

Autor: Daniel Saad Nogueira Nunes

Próximo a residência de Epaminondas existe uma máquina de refrigerantes. Certo dia, Epaminondas estava com bastante sede e colocou uma moeda nesta máquina, que despejou um refrigerante que Epaminondas não havia pedido. Depois de várias tentativas e de muito dinheiro perdido, Epaminondas percebeu que esta máquina escolhia um refrigerante ao acaso, de acordo com uma distribuição de probabilidade baseada apenas no último refrigerante despejado pela máquina.

Não querendo perder mais dinheiro e sendo um menino um tanto quanto pessimista, Epaminondas deseja saber qual a probabilidade do seu refrigerante favorito **não** ser despejado nenhuma vez pela máquina após a inserção de um número fixo de moedas. Ajude o pobre garoto a economizar as suas moedas e faça um programa para calcular essa probabilidade.

Entrada

A primeira linha da entrada contém 4 inteiros, separados por espaço, N $(1 \leq N \leq 100)$, $U(0 \leq U < N)$, F $(0 \leq F < N)$ e M $(0 \leq M \leq 100)$, indicando respectivamente: o número de tipos distintos de refrigerantes oferecidos pela máquina, o tipo do último refrigerante despejado pela máquina, o tipo do refrigerante preferido de Epaminondas e o número de moedas que Epaminondas está disposto a gastar.

As próximas N linhas contém, cada uma, N valores $a_{i,0}, a_{i,1}, \ldots a_{i,j}, \ldots a_{i,n-1}$, onde $a_{i,j}$ é a probabilidade da máquina despejar o refrigerante de tipo j dado que o último despejado foi do tipo i.

Saída

Seu programa deverá imprimir a probabilidade de o refrigerante favorito de Epaminondas não ser despejado nenhuma vez após a inserção de M moedas na máquina.

Para cada caso de teste, se sua resposta é um valor y e a resposta do juiz é o valor z, sua resposta será considerada correta se $\dfrac{|y-z|}{\max(1, z)} \leq 10^{-3}$.

Exemplo

Entrada	Saída

`2 0 1 1`
`0.25 0.75`
`0.5 0.5`
	`0.25`

`3 0 2 2`
`0.25 0.5 0.25`
`0.5 0.25 0.25`
`0.25 0.25 0.5`
	`0.5625`

`3 1 2 3`
`0.25 0 0.75`
`0 0.5 0.5`
`0.75 0.25 0`
	`0.125`

Notas

No primeiro caso de teste existem dois tipos de refrigerante. O último tipo despejado pela máquina foi o 0 e o refrigerante favorito de Epaminondas é do tipo 1. Ao todo, Epaminondas dispõe de uma moeda. A probabilidade, após a inserção dessa moeda, da máquina não despejar o refrigerante preferido de Epaminondas é: 0.25.

No segundo caso de teste existem três tipos de refrigerante. O último tipo despejado pela máquina foi o 0 e o refrigerante favorito de Epaminondas é do tipo 2. Ao todo, Epaminondas dispõe de duas moedas. A probabilidade, após a inserção dessas duas moedas, da máquina não despejar o refrigerante preferido de Epaminondas é: $(0.25)^2 + (0.25 \cdot 0.5) + (0.5)^2 + (0.5 \cdot 0.25) = 0.5625$.

Problema L
Almoço em Manhattan

Limite de tempo: 1s

Autor: José Marcos da Silva Leite

Chico está de férias em Manhattan e quer almoçar, mas pretende gastar pouco com sua refeição. Após realizar uma rápida pesquisa na Internet, ele conseguiu uma lista com a localização (x, y) e o preço médio da refeição de diversos restaurantes. O que ele ainda não levou em consideração foi o custo de transporte!

Ele sabe sua atual localização, e que a distância entre duas localizações (x_a, y_a) e (x_b, y_b) é dada por $|x_a - x_b| + |y_a - y_b|$, conhecida como distância de Manhattan (ou métrica do táxi). Chico também disse que, de táxi, gasta 2 dinheiros para cada unidade de distância. Este custo do transporte deve ser levado em consideração para o cálculo do custo total do almoço.

Chico sabe que você é bom em informática e quer sua ajuda para economizar! Ele vai te mandar, pelo zap, sua localização e uma lista de restaurantes com localização e preço médio da refeição.

Manhattan é uma cidade cheia de prédios, então é possível pode haver mais de um restaurante em uma mesma localização.

Entrada

A primeira linha da entrada contém três inteiros, separados por espaço, N $(1 \leq N \leq 10^5)$, x_c e y_c $(-10^3 \leq x_c, y_c \leq 10^3)$, os quais representam a quantidade de restaurantes na lista e a localização de Chico, respectivamente.

Em seguida, são descritas N linhas. Cada linha contém três inteiros x_i, y_i $(-10^3 \leq x_i, y_i \leq 10^3)$ e p_i $(1 \leq p_i \leq 30)$, os quais descrevem a posição do i-ésimo restaurante e seu preço $(1 \leq i \leq N)$, respectivamente.

Saída

Em uma única linha de saída, imprima dois inteiros V e L, separados por um espaço em branco.

O inteiro V representa o gasto total, e deve ser o menor valor que Chico pagará para almoçar, considerando o custo do transporte. O inteiro L $(1 \leq L \leq N)$ é o índice, de acordo com a ordem de leitura, do restaurante onde Chico deve ir.

Caso exista vários restaurantes empatados com menor preço, você deve imprimir o menor índice.

Exemplo

Entrada	Saída

`3 3 3`
`1000 1000 2`
`4 4 10`
`10 -17 8`
	`14 2`

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

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

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

Notas

No primeiro caso de teste, Chico está na posição $(3, 3)$ e o restaurante que gerará menor despesa é o segundo, que encontra-se na posição $(4, 4)$ e tem preço médio de refeição 10. Para viajar até esse restaurante, Chico gasta 4 reais e, com os 10 reais da refeição, o custo total é 14. Portanto deverá ser impresso `14 2`.

No segundo caso de teste, Chico encontra-se na posição $(2, 2)$ e ambos os restaurantes dão a mesma despesa: 6 dinheiros. Portanto deve ser impresso o índice do primeiro restaurante. O terceiro caso de teste é similar ao segundo.

No último caso de teste, o restaurante mais vantajoso é o segundo, com custo 5.