



Maratona de Programação

Caderno de Problemas
UFV, 31 de agosto de 2023

Instruções:

- Este caderno contém 16 problemas: com páginas numeradas de 1 a 16, não contando esta página de rosto.
- Em todos os problemas, a entrada deve ser lida da entrada padrão e a saída deve ser escrita na saída padrão.

UFV

Universidade Federal de Viçosa

Informações gerais

- A entrada deve ser lida da entrada padrão e a saída escrita na saída padrão
- Todas as linhas, inclusive a última, tanto da entrada quanto da saída, devem ter fim-de-linha
- Sempre que uma linha contém vários valores, eles são separados por um único espaço em branco e nenhum outro espaço deve aparecer (nem linha em branco), tanto na entrada quanto na saída
- O alfabeto inglês é sempre usado, não deve haver acentos, cedilhas, etc, nem na entrada nem na saída

Tempos limites de execução

- G - Capybarbie – 2,0 s
- J - Capivaras cochilando – 0,5 s
- K - Capivaras dormindo – 0,5 s
- P - Ilhas na UFV – 1,0 s
- Nas demais – 0,2 s
- ATENÇÃO! Os tempos foram calibrados para C++ e são os mesmos para todas as linguagens; não há garantia que códigos em outras linguagens executem dentro destes tempos.

“Esta é uma obra de ficção, qualquer semelhança com nomes, pessoas, fatos ou situações da vida real terá sido mera coincidência.”

Fontes das figuras ilustrativas usadas nos problemas da prova:

- Questões ausentes: criada especialmente para essa prova com assistência do DALL-E 2
- Números unários: criada e editada especialmente para essa prova com assistência do DALL-E 2
- Febre Maculosa na UFV: criada especialmente para essa prova com assistência do DALL-E 2
- Capyheimer: <https://www.youtube.com/watch?v=yvBlyC7x0aI>
- Capybarbie: <https://www.instagram.com/p/Cq5SAxYS5u3/>
- Capialok: <https://www.teepublic.com/t-shirt/40720245-dj-capybara-art>
- Capivaras dormindo: criada especialmente para essa prova
- Capivaras dormindo 2: criada especialmente para essa prova
- Fome de pizza: https://www.reddit.com/r/capybara/comments/xck1fb/capybara_pogging_over_a_new_york_style_pizza/
- Filó e os primos gêmeos: criada especialmente para essa prova com assistência do DALL-E 2
- atan: onde está?: arquivo pessoal
- Ilhas na UFV: arquivo pessoal

Problema A. Questão ausente

Arquivo-fonte: ".x", onde x deve ser c, cpp, java ou py

Esta questão foi removida, você pode usar esta página como rascunho.

Problema B. Questões ausentes

Arquivo-fonte: "questoes.x", onde x deve ser c, cpp, java ou py

O comitê de prova da Maratona da SI preparou várias questões, imprimiu e enviou para a gráfica preparar os cadernos de questões. Na hora da competição, percebeu que faltavam algumas questões!

Uma investigação foi iniciada para descobrir se alguém retirou as questões intencionalmente ou se foi apenas desatenção do funcionário, que deixou uma capivara comer algumas páginas.

Independentemente do motivo, algo chamou a atenção do comitê: as questões ausentes eram justamente as indexadas por vogais!

Sua tarefa é descobrir quantas questões há na prova, descontando as questões ausentes. As questões seguem a ordem do alfabeto.

Entrada

A entrada contém apenas uma consoante maiúscula, indicando a última questão da prova.

Saída

Escreva o número de questões presentes na prova.



Cena do crime

Exemplos

Entrada	Saída
F	4
Entrada	Saída
P	12
Entrada	Saída
B	1

- No 1º caso as questões seriam A, B, C, D, E e F. As questões A e E estão ausentes, então há 4 questões na prova.
- O 2º caso de teste é o caso real deste caderno de prova...

Problema C. Números unários

Arquivo-fonte: "unarios.x", onde x deve ser c, cpp, java ou py

Pesquisadores da UFV estavam notando algo estranho na borda das lagoas: frequentemente apareciam vários riscos. Um dia, após soltarem 20 jacarés na lagoa apareceram 20 riscos. A seguir, uma capivara comeu dois jacarés e, então, o número de riscos virou 18.

Após alguns estudos, descobriram algo surpreendente: capivaras sabem contar, mas utilizam o sistema unário de numeração!

No sistema decimal, os dígitos vão de 0 até 9. O número 314, por exemplo, significa $3 \times 10^2 + 1 \times 10^1 + 4 \times 10^0$. Em unário, só há o dígito 1. O número 111 (3, em decimal) significa $1 \times 1^2 + 1 \times 1^1 + 1 \times 1^0$. No sistema unário, o número 0 é representado por uma string vazia.

Para estudar a numeração das capivaras, a UFV precisará de um programa capaz de fazer operações aritméticas com esses números. Você foi contratado para implementar a operação de subtração.

Entrada

A entrada contém duas linhas, cada uma contendo um número positivo em unário (contendo de 1 a 10^6 dígitos).

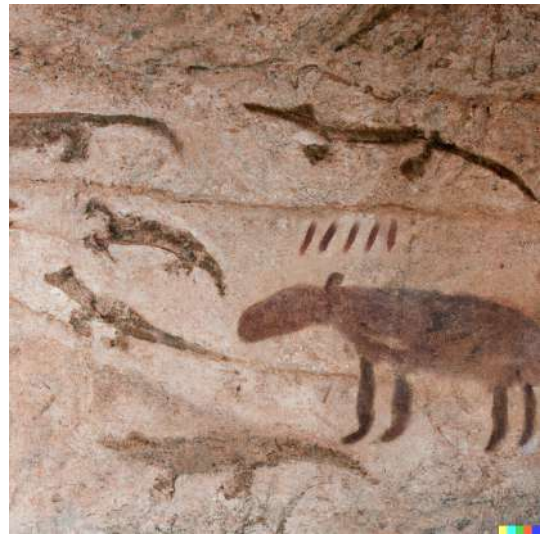
Saída

Escreva uma linha contendo a diferença, em unário, entre o primeiro e o segundo número.

Exemplos

Entrada	Saída
1111 11	11
11 1111	-11

No primeiro exemplo: o primeiro número é 4 em decimal e o segundo é 2. A diferença é 2, representada por 11, em unário.



Número unário em pintura rupestre na parede do PVA

Problema D. Febre Maculosa na UFV

Arquivo-fonte: "febre.x", onde x deve ser c, cpp, java ou py

Recentemente descobriram que uma capivara da UFV estava com febre maculosa! Infelizmente, muitos calouros ainda não conhecem os perigos dessa doença e, portanto, não se preveniram ao andar perto dos lagos da nossa universidade.

Um dos sintomas dessa temível doença é que o roedor fica andando em zig-zag e espalhando carrapato estrela (o vetor (não daqueles de MAT135/INF390) da doença) por onde passa.

Sabe-se que Q calouros da UFV estavam passeando descalços perto da lagoa (o que aumenta consideravelmente a chance de contaminação) e podem ter cruzado várias vezes o caminho por onde a capivara contaminada passou. Por precaução, o reitor lhe contratou para desenvolver um software capaz de contar quantas vezes cada um dos calouros cruzou a rota da capivara.

Por sorte (sua), tanto a capivara quanto os calouros percorreram rotas bem simples: o roedor andou em um padrão zig-zag percorrendo os pontos $i = 1 \dots N$ de coordenadas $C_i = (i, y_{c_i})$, formando $N - 1$ segmentos de reta, cada um conectando um ponto C_i ao ponto C_{i+1} .

Cada aluno j , por sua vez, sempre fez um percurso (descalço) na horizontal no sentido da esquerda para a direita, tendo tirado o calçado em um ponto com coordenada $(x_{a_{j1}}, y_{a_j})$ e terminando o percurso no ponto $(x_{a_{j2}}, y_{a_j})$.

Vamos considerar que um aluno j cruzou um dos segmentos s da rota da capivara se o percurso dele interceptou s ou se tocou s , exceto se o ponto mais à esquerda de s for $(x_{a_{j2}}, y_{a_j})$ ou o mais à direita for $(x_{a_{j1}}, y_{a_j})$, ou seja, exceto se o percurso do aluno começa onde o segmento termina ou termina onde o segmento começa.

Entrada

A entrada começa com dois inteiros N ($2 \leq N \leq 10^5$) e Q ($1 \leq Q \leq 10^5$).

A seguir, há N inteiros y_{c_i} , representando a rota da capivara ($1 \leq y_{c_i} \leq 10^9$).

Por fim, há Q linhas, cada uma com 3 inteiros $x_{a_{j1}}, x_{a_{j2}}, y_{a_j}$, representando a rota de um aluno descalço ($1 \leq x_{a_{j1}}, x_{a_{j2}} \leq N$, $x_{a_{j1}} < x_{a_{j2}}$ e $1 \leq y_{a_j} \leq 10^9$).

Saída

Para cada aluno, escreva uma linha informando quantas vezes sua rota cruzou algum segmento da rota da capivara.

Exemplos

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

- O primeiro aluno cruzou o segundo e o terceiro segmento.
- O segundo aluno cruzou o primeiro e o terceiro segmento.
- O terceiro aluno cruzou o primeiro segmento. Note que o aluno percorre a rota de $(1, 3)$ até $(2, 3)$, mas consideramos que ele não intercepta o segmento s de $(2, 3)$ até $(3, 5)$, pois o final do trajeto do aluno é igual ao início de s .



Capivara com mácula de febre suspeitosa

Problema E. Questão ausente

Arquivo-fonte: `-.x`, onde `x` deve ser `c`, `cpp`, `java` ou `py`

Esta questão foi removida, você pode usar esta página como rascunho.

Problema F. Capyheimer

Arquivo-fonte: "capyheimer.x", onde x deve ser c, cpp, java ou py

Recentemente, dois filmes causaram muita discussão na internet: Capybarbie e Capyheimer. Com o objetivo de alcançar a maior bilheteria e gerar mais marketing, Capyheimer criou um plano muito engenhoso: ele percebeu que, se deixar as capivaras bem ricas, quando uma delas assistir a seu filme, ela convencerá outras duas capivaras a assistirem no dia seguinte (a não ser que existam menos do que duas). Obviamente, apenas as que ainda não assistiram e que são neutras (ou seja, não são fãs da Capybarbie) assistem ao filme.



A grande procura por seu filme gerará um sucesso explosivo nos cinemas. O problema é que atualmente muitas salas estão reservadas para outros filmes e, portanto, Capyheimer precisa convencer os donos dos cinemas a reservarem salas para o seu.

Dadas a quantidade total de capivaras neutras N no planeta, a de fãs da Capybarbie B e o número I de capivaras neutras assistindo ao filme do Capyheimer no dia do lançamento, o objetivo é calcular quantas capivaras irão ao cinema assistir ao filme dele no dia de maior demanda.

Capyheimer planejou fazer um teste para obter a resposta a essa pergunta, mas atualmente tais testes estão proibidos. Portanto, ele achou melhor lhe contratar para desenvolver um software capaz de fazer esse tipo de cálculo.

Entrada

A entrada começa com um inteiro T indicando o número de casos de teste ($1 \leq T \leq 100000$). A seguir, há T linhas, cada uma contendo um teste. Cada linha possui 3 inteiros: N, B e I ($1 \leq N, B, I \leq 10^{15}, I \leq N$).

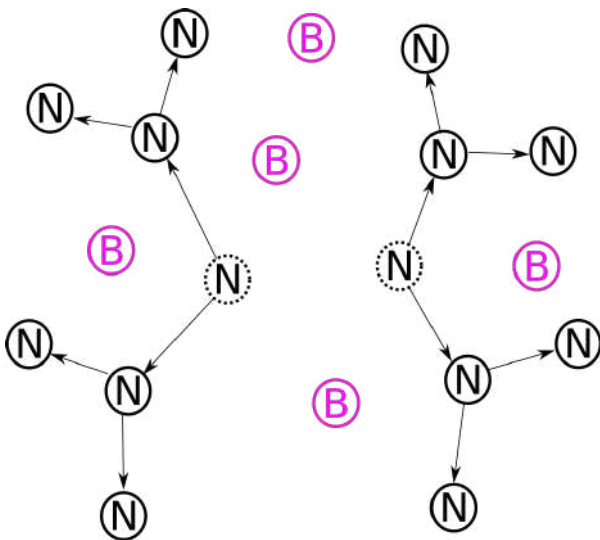
Saída

Para cada caso de teste, escreva uma linha contendo o número de capivaras assistindo ao filme de Capyheimer no cinema no dia de maior demanda.

Exemplos

Entrada	Saída
2	8
14 5 2	8
15 5 1	

A figura abaixo ilustra a cadeia de eventos do primeiro caso de teste. Os círculos representam as capivaras (B para fãs da Capybarbie e N neutras). No dia de lançamento há 2 capivaras assistindo ao filme (representadas por círculo tracejado). No terceiro dia, teremos 8 assistindo ao filme.



Problema G. Capybarbie

Arquivo-fonte: "capybarbie.x", onde x deve ser c, cpp, java ou py

Capybarbie frequentemente viaja de Viçosa para visitar seu namorado Capyken, em Tangamandápio. Porém, seu pai, Capyheimer, quer que ela pare de perder tempo namorando e se dedique mais aos estudos na UFV e às maratonas de programação.

Para resolver esse problema, ele planeja explodir bombas em estradas, fazendo com que seja impossível haver alguma rota entre Viçosa e Tangamandápio.

Sabe-se que Capyheimer possui uma quantidade B de bombas e que cada estrada i (sempre conectando duas cidades) exige uma determinada quantidade B_i de bombas para se tornar intrafegável (as que ficam perto de Manhuaçu nem precisam de bombas, já que estão completamente esburacadas).

Capyken pretende investir em melhorias nas estradas para permitir que a Capybarbie o visite. Cada valor X de Capicoins que o Capyken investe faz com que cada estrada i passe a necessitar de $B_i + X$ bombas para ser desativada. Capyken consegue investir apenas valores inteiros.

Capyken deseja investir o mínimo possível de dinheiro (para sobrar para rodízios de piza com a Capybarbie) de modo a permitir que sua namorada o visite. Desenvolva um programa para ajudá-lo a calcular quanto irá gastar.



Entrada

A entrada começa com 3 inteiros: B, C, E indicando, respectivamente, o número de bombas, cidades e estradas que existem ($0 \leq B \leq 10^6, 2 \leq C \leq 10^3, 0 \leq E \leq 3 \times 10^5$).

A seguir, há E linhas, cada uma descrevendo uma estrada. Cada linha contém 3 inteiros: C_1, C_2, B_i indicando, respectivamente, que ela liga a cidade C_1 à cidade C_2 ($1 \leq C_1, C_2 \leq C$) e inicialmente precisa de B_i ($0 \leq B_i \leq 10^6$) bombas para ser desativada. Cada estrada é de mão única.

Assuma que Viçosa é a cidade 1 e Tangamandápio é a C .

Saída

Escreva uma linha contendo o mínimo de Capicoins que o Capyken precisa investir para permitir o tráfego de Viçosa até Tangamandápio. Se não for possível que a Capybarbie visite o Capyken, mesmo com ele disposto a fortalecer as estradas, escreva -1.

Exemplos

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

Se Capyheimer destruir a estrada ligando as cidades 1 e 2 e a ligando 1 e 4, ele gastaria $1 + 2 = 3$ de suas bombas (isso seria o mínimo que ele gastaria) e isso bloquearia completamente o tráfego entre Viçosa (cidade 1) e Tangamandápio (6). Se Capyken investir 3 Capicoins, o número de bombas necessárias em cada estrada aumentaria em 3 unidades e o número mínimo que Capyheimer gastaria para impedir o tráfego seria 9, mas ele possui apenas 7.

Problema H. DJ Capialok

Arquivo-fonte: "capialok.x", onde x deve ser c, cpp, java ou py

Capialok está se formando na UFV e foi convidado para trabalhar como DJ no baile de formatura. Mas há um problema! A maioria dos formandos se empolga com ritmos musicas que Capialok nem considera música. Então, para agradar os formandos, ele fez um evento teste com um repertório musical completo, incluindo, no seu conceito, músicas boas e músicas de gosto duvidoso.

Ocupado no serviço de DJ, ele não teve como anotar de forma precisa a empolgação dos ouvintes. Mas anotou, para cada música tocada, se a empolgação diminuiu ou aumentou em relação à música anterior. Mais tarde, Capialok criou um sistema de pontuação das músicas:

- Cada música começa valendo 100 pontos;
- Se a empolgação de uma música aumentou em relação à música anterior, ela ganha tantos pontos quantos forem os aumentos seguidos de empolgação que ocorreram logo antes dela;
- Se a empolgação de uma música diminuiu em relação à música anterior, ela perde 2 pontos para cada diminuição seguida de empolgação logo antes dela;
- A primeira música sempre vale 100.



Capialok preparando-se para o baile de formatura

Por exemplo, suponha que a anotação de Capialok seja: ++++---+---+:

- A música 1 vale 100 (sempre)
- As músicas 2, 3 e 4 valem respectivamente 101, 102 e 103
- A música 5 continua valendo 100 (não houve diminuição logo antes dela)
- As músicas 6, 7 e 8 valem respectivamente 100, 101 e 102
- As músicas 9, 10 e 11 valem respectivamente 100, 98 e 96
- A música 12 continua valendo 100 (não houve aumento logo antes dela)

O sistema de pontuação de Capialok pode não ser muito preciso, mas é tudo o que ele tem. Ajude-o a escolher as músicas!

Entrada

A entrada começa com uma linha contendo dois inteiros, o número N de músicas que Capialok tocou no evento teste e o tempo total T_i , em segundos, que ele tem para tocar músicas no baile de formatura ($1 \leq N \leq 200$ e $1 \leq T \leq 7200$).

Em seguida, há uma linha contendo N valores inteiros D_i , indicando a duração de cada música em segundos ($1 \leq D_i \leq 600$).

Por fim, uma linha contendo N caracteres '+' ou '-', indicando respectivamente se a empolgação dos ouvintes aumentou ou diminuiu em relação à música anterior (note que o primeiro caractere é irrelevante para a música 1, que sempre vale 100).

Saída

Escreva a pontuação total máxima que Capialok pode conseguir escolhendo de forma adequada as músicas para o baile.

Exemplos

Entrada	Saída
12 800 120 130 100 90 80 110 120 110 150 100 80 120 ++++---+---+	802
3 30 16 15 15 ---	194
3 30 16 16 15 -++	101

Problema I. Questão ausente

Arquivo-fonte: `-.x`, onde `x` deve ser `c`, `cpp`, `java` ou `py`

Esta questão foi removida, você pode usar esta página como rascunho.

Problema J. Capivaras cochilando

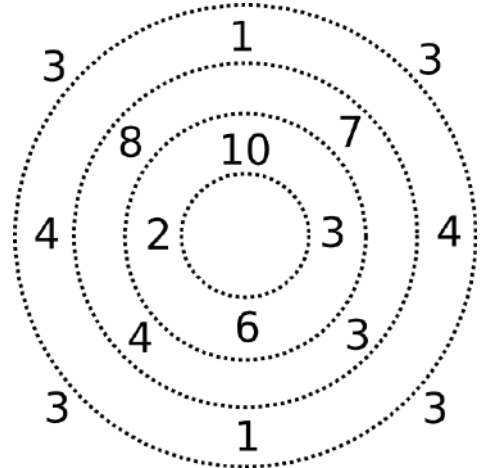
Arquivo-fonte: "cochilando.x", onde x deve ser c, cpp, java ou py

Os calouros da UFV notaram algo interessante sobre as capivaras da universidade: ao dormir, elas se organizam em um padrão concêntrico muito peculiar.

Esse padrão se forma da seguinte maneira:

- No círculo mais interno, dormem N capivaras, cada uma contendo uma quantidade de carrapatos.
- Para cada par de capivaras adjacentes no círculo anterior (contendo a e b carrapatos), há uma capivara entre elas em um círculo mais externo cujo número de carrapatos é $|a - b|$.
- Esse processo se repete recursivamente até que se forme um círculo onde todas capivaras possuem a mesma quantidade de carrapatos.

Veja o exemplo ao lado: no primeiro círculo há 4 capivaras com, respectivamente, 10, 3, 6 e 2 carrapatos. No seguinte, há outras 4 capivaras com $|10 - 3| = 7$, $|10 - 2| = 8$, $|2 - 6| = 4$ e $|6 - 3| = 3$ carrapatos. O processo termina quando a quantidade de carrapatos em todas fica igual. No exemplo, todas têm 3 carrapatos no último círculo.



O reitor lhe contratou para desenvolver um software de Visão Computacional capaz de contar o número de carrapatos em um determinado grupo. A versão inicial do seu software consegue apenas contar o número de capivaras e carrapatos nas que estão no círculo mais interno do grupo.

Termine o software para que, dadas essas quantidades, conte o número total de carrapatos do grupo.

Entrada

A entrada começa com um inteiro N , o número de capivaras no círculo mais interno ($1 \leq N \leq 200$).

A seguir, uma linha com N inteiros não-negativos x_i , representando a quantidade de carrapatos em cada uma ($0 \leq x_i \leq 10^9$).

Saída

Escreva uma linha contendo o total de carrapatos no grupo. É garantido que, nos casos de teste, o processo termina.

Exemplos

Entrada	Saída
4 2 10 3 6	65
2 1 5	14

Problema K. Capivaras dormindo

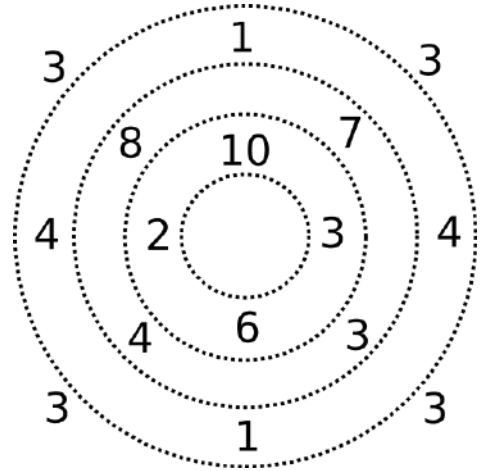
Arquivo-fonte: "dormindo.x", onde x deve ser c, cpp, java ou py

Obs.: este texto é igual ao da questão anterior, a única diferença é a última frase (seção Saída).

Os calouros da UFV notaram algo interessante sobre as capivaras da universidade: ao dormir, elas se organizam em um padrão concêntrico muito peculiar.

Esse padrão se forma da seguinte maneira:

- No círculo mais interno, dormem N capivaras, cada uma contendo uma quantidade de carrapatos.
- Para cada par de capivaras adjacentes no círculo anterior (contendo a e b carrapatos), há uma capivara entre elas em um círculo mais externo cujo número de carrapatos é $|a - b|$.
- Esse processo se repete recursivamente até que se forme um círculo onde todas capivaras possuem a mesma quantidade de carrapatos.



Veja o exemplo ao lado: no primeiro círculo há 4 capivaras com, respectivamente, 10, 3, 6 e 2 carrapatos. No seguinte, há outras 4 capivaras com $|10 - 3| = 7$, $|10 - 2| = 8$, $|2 - 6| = 4$ e $|6 - 3| = 3$ carrapatos. O processo termina quando a quantidade de carrapatos em todas fica igual. No exemplo, todas têm 3 carrapatos no último círculo.

O reitor lhe contratou para desenvolver um software de Visão Computacional capaz de contar o número de carrapatos em um determinado grupo. A versão inicial do seu software consegue apenas contar o número de capivaras e carrapatos nas que estão no círculo mais interno do grupo.

Termine o software para que, dadas essas quantidades, conte o número total de carrapatos do grupo.

Entrada

A entrada começa com um inteiro N , o número de capivaras no círculo mais interno ($1 \leq N \leq 200$).

A seguir, uma linha com N inteiros não-negativos x_i , representando a quantidade de carrapatos em cada uma ($0 \leq x_i \leq 10^9$).

Saída

Escreva uma linha contendo o total de carrapatos no grupo. Nos casos em que o processo nunca termina, escreva -1 .

Exemplos

Entrada	Saída
4 2 10 3 6	65
2 1 5	14

Problema L. Fome de pizza

Arquivo-fonte: "fome.x", onde x deve ser c, cpp, java ou py

Os competidores da UFV sempre vão a uma pizzaria após as maratonas. Desta vez pediram várias pizzas, de diferentes tamanhos. Mas, como o tempo está curto, cada um terá direito a apenas um pedaço. Com a fome de pizza que você está, já está calculando qual pedaço vai pegar: o de maior área, obviamente!

Entrada

A entrada começa com um inteiro N indicando o número de pizzas compradas ($1 \leq N \leq 1000$). Em seguida N linhas, cada uma com um inteiro P_i e um número real R_i , respectivamente o número de pedaços e o raio da i -ésima pizza ($1 \leq P_i \leq 20$ e $1 \leq R_i \leq 1000$).

Saída

Seu programa deve escrever o índice da pizza com o pedaço de maior área. Se houver empate, o de menor índice entre eles.

Considere as pizzas numeradas a partir de 1.

Observações

Assuma que todas as pizzas são perfeitamente redondas e que todos os pedaços de uma mesma pizza têm exatamente a mesma área, ou seja, a área de cada pizza i é igualmente dividida entre seus P_i pedaços.

Exemplos

Entrada	Saída
3 6 10 8 11 10 12	1
3 3 3 4 4 5 5	3
4 8 5 20 8 5 4 1 1.7	2



Garantindo seu pedaço de pizza

Problema M. Filó e os primos gêmeos

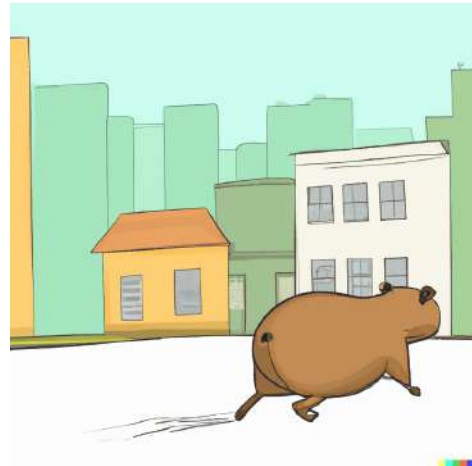
Arquivo-fonte: "filo.x", onde x deve ser c, cpp, java ou py

Filó é uma capivara que parecia viver feliz com um estudante de agronomia da UFV. Um dia, entretanto, ele ficou com receio de perder a Filó quando viu fiscais à porta de seu quarto no alojamento. Ele então gritou: "Filó, corra e se esconda entre os primos gêmeos!"

Filó saiu correndo pela P. H. Rolfs e entrou numa casa sem que os fiscais percebessem onde ela tinha entrado. Eles verificaram o nome de todos os moradores, mas não encontraram nenhum primo nem parente do estudante, muito menos algum gêmeo.

O que eles não sabiam é que Filó frequentava as aulas no PVB e sabia muito bem o que são primos gêmeos: dois números primos cuja diferença é igual a 2.

Por exemplo, se as casas são numeradas a partir de 1, Filó se escondeu na casa 4, que é a primeira entre dois primos gêmeos (no caso, 3 e 5). Se as casas são numeradas a partir de 7, então Filó se escondeu na casa 12, que é a primeira entre dois primos gêmeos (nesse caso, 11 e 13).



Última vez que Filó foi vista pelos fiscais

Entrada

A entrada começa com um inteiro N indicando o número da primeira casa da P. H. Rolfs ($1 \leq N \leq 10^6$).

Saída

Seu programa deve escrever o número da casa em que Filó se escondeu.

Exemplos

Entrada	Saída
1	4
Entrada	Saída
7	12
Entrada	Saída
4	6
Entrada	Saída
110	138
Entrada	Saída
200	228

- No terceiro caso de teste, Filó não se escondeu na casa 4 porque 4 não estava entre primos gêmeos (não havia casa 3).

Problema N. atan: onde está?

Arquivo-fonte: "atan.x", onde x deve ser c, cpp, java ou py

Por algum motivo, os compiladores da maratona mineira não possuem a função atan, da cmath, para cálculo de arco tangente. Em uma das questões, era muito importante calcular o atan de π^i e muitos times implementaram códigos supondo que a função existia.

Felizmente os cálculos podem ser feitos de outra forma, sem utilizar essa função trigonométrica. O problema é que um dos times da UFV nunca consegue encontrar o atan!

Para piorar, eles utilizaram N atan ao longo do código e, portanto, o problema ficou ainda maior! O Gabriel e o Matheus, integrantes do time, pediram para você desenvolver um código capaz de encontrar todas as ocorrências dessa função no programa que desenvolveram.



Times da UFV na Maratona Mineira: cadê atan?

Entrada

A entrada é composta por várias linhas de código (até 1000 linhas, cada uma contendo até 1000 colunas), não necessariamente em C++.

Saída

Escreva as linhas e colunas onde a string "atan" aparece (não importa se ela aparece como parte de um comentário, uma chamada de função, etc). Cada ocorrência da palavra deverá gerar uma linha na saída, contendo dois inteiros (respectivamente, o número da linha e da coluna onde o primeiro caractere se encontra) separados por um espaço. Suponha que a contagem comece de 1. A saída deve estar na ordem em que as strings aparecem.

Exemplos

Entrada	Saída
<pre>double temp = atan(5); int main() { int n = 5; cout << n*atan(3.14); //precisamos calcular n*atan(pow(pi, i)) //onde esta o atan???? //ATAN //capivara at an return 0; }</pre>	<pre>1 15 4 11 5 25 6 15</pre>

Problema O. Questão ausente

Arquivo-fonte: ".x", onde x deve ser c, cpp, java ou py

Esta questão foi removida, você pode usar esta página como rascunho.

Problema P. Ilhas na UFV

Arquivo-fonte: "ilhas.x", onde x deve ser c, cpp, java ou py

Como todos sabem, há muitas capivaras na UFV. Como muitos alunos as estavam incomodando na hora de dormir, a reitoria decidiu construir ilhas na lagoa para que elas durmam com mais tranquilidade, longe da borda.

Felizmente, a lagoa em que habitam possui uma geografia propícia, que permite que ilhas sejam criadas simplesmente alterando o nível da água (dispositivo em funcionamento na foto ao lado).

Como há muitas famílias de capivaras (e cada uma gostaria de uma ilha particular), o reitor lhe contratou para desenvolver um software que, dada a elevação do relevo de uma lagoa (também conhecida por batimetria) e o nível da água, calcula quantas ilhas há nela.

Uma lagoa será sempre quadrada e representado por uma matriz de elevação, onde cada célula aproxima a elevação do pedaço correspondente da lagoa. Uma ilha é um conjunto contíguo de células que estão acima do nível da água (cada célula tem, no máximo, 4 vizinhos – ou seja, considera-se a vizinhança apenas de células na mesma linha ou na mesma coluna). Assuma que células na borda da lagoa também podem formar ilhas.



Família de capivaras aguardando a formação de sua ilha

Entrada

A entrada começa com dois inteiros N e Q indicando, respectivamente, a largura da lagoa, em células ($1 \leq N \leq 1000$), e o número de consultas ($1 \leq Q \leq 10000$).

A seguir, há N linhas, cada uma contendo N inteiros não-negativos (entre 0 e 10^9) representando a elevação, em metros, do relevo na célula correspondente da lagoa.

Então, há Q linhas, cada uma representando uma consulta. Cada linha terá um número inteiro E ($0 \leq E \leq 10^9$) representando uma possível elevação da lagoa.

Saída

Para cada consulta (correspondente a uma elevação E), escreva quantas ilhas haverá na lagoa se o nível da água for E metros.

Exemplos

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

Na primeira consulta, há apenas a célula com elevação 6 acima da água. Na segunda, há as ilhas com elevações (5), (6), (5, 5, 5, 4, 5). Na terceira, as ilhas possuem células com elevações (5), (6), (5, 5, 5), (5)