IX Maratona de Programação do IFB

Caderno de Problemas

27 de setembro de 2024

(Este caderno contém 13 problemas)

Comissão Organizadora:
Alberto Tavares Duarte Neto (UnB)
Bruno Vargas (UnB)
Caleb Martim (UnB)
Daniel Saad Nogueira Nunes (IFB)
Edson Alves da Costa Júnior (UnB/FGA)
Guilherme Novaes Ramos (UnB)
Jeremias Moreira Gomes (IDP)
Vinicius Ruela Pereira Borges (UnB)

Instituto Federal de Educação, Ciência e Tecnologia de Brasília, *campus* Taguatinga

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 – Atunda Batata

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

Autor: Guilherme Ramos

Atunda Batata soa como uma famosa frase em suaíli que foi absurdamente registrada por uma empresa de índole deveras duvidosa. A frase original pode ser traduzida como "sem preocupações". Fora da África, se tornou notória graças a uma canção cujo refrão remete a uma peculiar dupla/tripla de amigos ter esta abordagem para a vida.

Na canção, cada personagem tem uma resposta à mensagem de tranquilidade. O suricato acha que "*Atunda Batata! É lindo dizer!*", já o facochero cantarola que "*Atunda Batata! Sim, vai entender!*". Por fim, o leãozinho (novato da turma) sinaliza que "*Atunda Batata! Os seus problemas você deve esquecer!*"

Entrada

A entrada consiste de uma quantidade indeterminada de casos de teste, e termina com `EOF`. Cada teste é composto por duas linhas, a primeira apresenta o refrão que todos cantam, e a segunda apenas a parte de uma das personagens, conforme os exemplos.

Saída

Para cada performance listada, apresente uma linha indicando o cantor.

Exemplo

Entrada	Saída
`Atunda Batata,` `Sim, vai entender!`	`facochero`

`Atunda Batata,`
`E lindo dizer!`
`Atunda Batata,`
`Sim, vai entender!`
`Atunda Batata,`
`Os seus problemas voce deve esquecer!`
	`suricato`
	`facochero`
	`leao`

Problema B – Bombons

Limite de tempo: 3s
Limite de memória: 256MB

Autor: Edson Alves

João tem o hábito de, a cada ano, no Dia das Crianças, fazer doações de pacotes de bombons para as instituições de caridade. A criançada adora chocolates, mas sempre há brigas na divisão das guloseimas. Ciente do fato, João tem o cuidado de enviar, para uma instituição que atende C crianças, um pacote que tenha B bombons, sendo B um múltiplo de C, para garantir a igualdade na distribuição.

João adquire os bombons de um fornecedor que trabalha com N tipos de pacote, numerados de 1 a N, onde cada tipo de pacote tem uma certa quantidade de bombons. Conhecido o número de crianças atendidas por cada uma das M instituições que receberão as doações, ajude João a decidir qual tipo de pacote ele deve enviar para cada instituição. João é generoso, então se houver mais de um tipo de pacote que atenda o critério da divisão, opte pelo que tiver o maior número de bombons.

Entrada

A primeira linha da entrada contém o valor do inteiro N ($1 \leq N \leq 4 \times 10^5$).

A segunda linha da entrada contém N inteiros b_i ($1 \leq b_i \leq 10^7, 1 \leq i \leq N$), separados por um espaço em branco, indicando a quantidade de bombons no i-ésimo tipo de pacote.

A terceira linha da entrada contém o valor do inteiro M ($1 \leq M \leq 4 \times 10^5$).

As M linhas seguintes contém, cada uma, um inteiro m_j ($1 \leq m_j \leq 10^7, 1 \leq j \leq M$), indicando o número de crianças atendidas pela instituição j.

Saída

Para cada instituição imprima uma linha, na mesma ordem da entrada, contendo o tipo de pacote que deve ser enviado para a instituição, de modo a garantir uma divisão igualitária dos bombons entre as crianças. Caso não exista tal tipo, imprima o valor -1. Se existirem dois ou mais tipo que atenda todos os critérios estabelecidos, imprima qualquer um deles.

Exemplo

Entrada	Saída

`4`
`10 6 15 10`
`3`
`2`
`3`
`7`
	`4`
	`3`
	`-1`

Notas

A primeira instituição tem 2 crianças, então deve ser enviado um pacote do tipo 4, com 10 bombons (o pacote 2, com 6 bombons, atenderia o critério da divisão, mas tem menos bombons que o tipo 4). Observe que também seria igualmente válido enviar um pacote do tipo 1.

Como a terceira instituição atende 7 crianças, nenhum dos 3 pacotes permite uma divisão igualitária dos bombons.

Problema C – Cinuca Olímpica

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

Autor: Jeremias Moreira Gomes

Finalmente a sinuca foi reconhecida como um esporte olímpico! E para melhorar, o Brasil foi escolhido como referência para escrever as regras do esporte. Assim, após muita discussão, a Iniciativa de Fomento à Bilhar decidiu por criar uma versão mais interessante do esporte, chamada de Cinuca Olímpica.

As regras são as seguintes, jogam dois jogadores um contra o outro, a mesa possui uma quantidade N de bolas numeradas, com a possibilidade de repetição, e o jogo transcorre normalmente com os jogadores encaçapando bolas. A diferença para a regra tradicional é que a categoria determina a quantidade de bolas B que serão contabilizadas para a pontuação do jogador e, antes da partida começar, o juiz irá sortear um número G que é o objetivo central da partida. A partir desse número, a pontuação de cada jogador é composta pelas B bolas de número mais próximo de G que ele encaçapou.

Para apresentar as regras ao comitê olímpico, você foi contratado para criar um programa que, dadas as bolas encaçapadas de cada jogador, enumere as bolas consideradas para a pontuação de cada um.

Entrada

A entrada é composta por um único caso de testes. A primeira linha contém dois inteiros R e S ($B \leq R, S \leq 10^5$), separados por um espaço, que representam, respectivamente, o número de bolas encaçapadas pelo jogador 1 e o jogador 2.

A segunda linha possui R inteiros r_i ($1 \leq r_i \leq 10^6$), separados por espaço e em ordem não decrescente, que representam os números das bolas encaçapadas pelo jogador 1.

A terceira linha possui S inteiros s_i ($1 \leq s_i \leq 10^6$), separados por espaço e em ordem não decrescente, que representam os números das bolas encaçapadas pelo jogador 2.

A quarta linha possui dois inteiros B e G ($1 \leq B \leq 10^4$, $1 \leq G \leq 10^6$), separados por um espaço, que representam, respectivamente, a quantidade de bolas da categoria consideradas para a pontuação e o número sorteado pelo juiz.

Saída

A saída deve conter duas linhas. A primeira linha deve conter B inteiros separados por espaço, representando as bolas consideradas para a pontuação do jogador 1, e a segunda linha deve conter B inteiros separados por espaço, representando as bolas consideradas para a pontuação do jogador 2, conforme os exemplos. Para cada linha, as respostas podem estar em qualquer ordem.

Exemplo

Entrada	Saída

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

`5 6`
`1 2 3 8 9`
`4 5 6 7 8 9`
`3 5`
	`2 3 8`
	`4 5 6`

Problema D – Dupla de dois

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

Autor: Daniel Saad Nogueira Nunes

O prof. Jeremias resolveu adotar o método *Trezentos*, do prof. Fragelli, em suas disciplinas. Esse método consiste em parear um aluno com nota alta com outro com nota baixa, para que cooperem e progridam em conjunto. É claro que quando propôs isso, Jeremias ouviu um "pode ser dupla de três?", o qual respondeu com um sonoro "dupla de dois, somente!".

Jeremias adaptará esse método da seguinte forma: primeiramente ele calculará, para qualquer dupla que poderá ser formada, a soma das notas dessa dupla. Em seguida, ele utilizará essa métrica para atribuir as duplas de modo a minimizar a diferença absoluta das somas considerando qualquer par de duplas.

Não que o prof. Jeremias precise da sua ajuda para resolver esse problema, contudo, ajude-o mesmo assim.

Entrada

A primeira linha da entrada possui um inteiro n, que indica a quantidade de alunos.

As próximas n linhas descrevem, cada, um aluno. Especificamente, a i-ésima linha desse conjunto possui o nome de um aluno s_i, seguido por um inteiro, que representa sua nota x_i. Essas duas informações estão separadas por um espaço.

Restrições:

- $4 \leq n \leq 10^5$ e n é par.
- s_i consiste de letras maiúsculas ou minúsculas e $|s_i| \leq 10$, $1 \leq i \leq n$.
- $0 \leq x_i \leq 100$, $1 \leq i \leq n$.

Saída

Seu programa deverá imprimir na primeira linha a maior diferença das somas das notas de qualquer par de duplas, nas próximas $n/2$ linhas, as duplas escolhidas por Jeremias. Cada dupla deverá estar no formato: `<aluno1> <nota1> <aluno2> <nota2>`.

O juiz aceitará qualquer atribuição de duplas que obedeça os critérios do enunciado.

Exemplo

Entrada	Saída
4 Daniel 20 Edson 80 Jeremias 80 Guilherme 20	0 Guilherme 20 Edson 80 Daniel 20 Jeremias 80
6 Daniel 30 Edson 40 Jeremias 50 Guilherme 65 Vinicius 70 Alberto 80	5 Daniel 30 Alberto 80 Edson 40 Vinicius 70 Jeremias 50 Guilherme 65
4 Godofredo 100 Hortolina 80 Oliveira 10 Joao 0	10 Joao 0 Godofredo 100 Oliveira 10 Hortolina 80

Notas

No primeiro exemplo, se parearmos Guilherme com Edson e Daniel com Jeremias, ambas as duplas terão soma 100 cuja diferença é 0. Nesse caso, também seria possível parear Guilherme com Jeremias e Daniel com Edson.

No segundo exemplo, ao parear Daniel com Alberto (soma 110), Edson com Vinicius (soma 110) e Jeremias com Guilherme (115), temos a diferença de 5 entre as duplas de Daniel e Alberto, e, Jeremias e Guilherme. Qualquer outra atribuição de duplas causaria um aumento dessa diferença, fazendo com que ela deixasse de ser mínima.

Problema E – Encontros

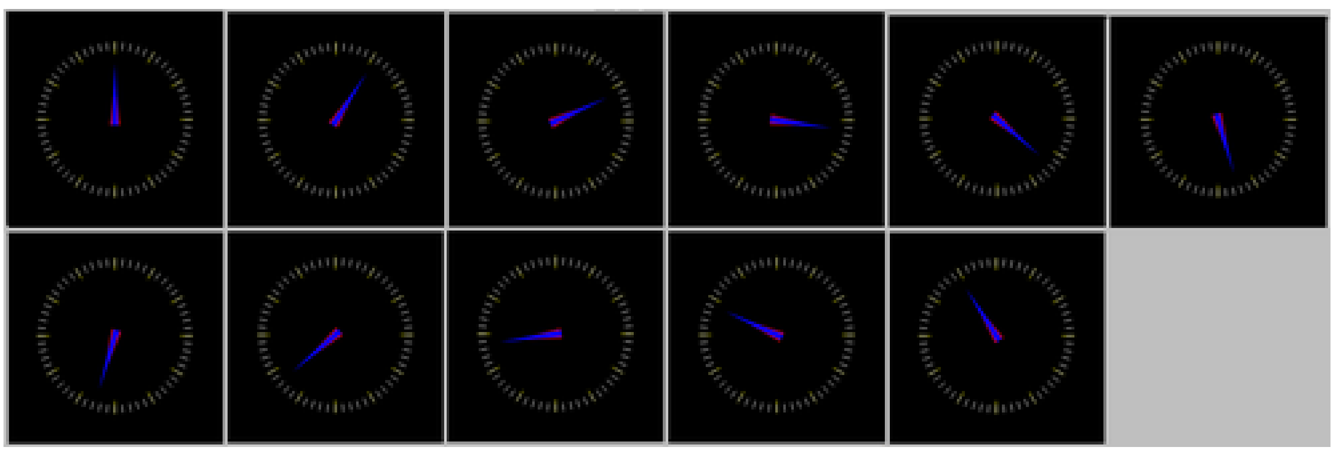
Limite de tempo: 1s
Limite de memória: 256MB

Autor: Jeremias Moreira Gomes

“Até um relógio quebrado está certo duas vezes por dia.”, mas ninguém pensa mais nisso, a partir do momento em que a humanidade está preparada para fazer o primeiro contato com seres de outro planeta.

O departamento de Investigação de Fenômenos Bizarros (IFB) é o responsável por qualquer tipo de contato com seres de outros planetas e outras realidades e, dessa vez, após anos de investigação eles finalmente conseguiram o inesperado, mas muito cobiçado: realizar o primeiro contato alienígena.

O único problema é que essa raça trabalha de uma forma um pouco diferente. Eles só podem fazer contato em horários bem específicos do dia, e isso demanda uma certa organização por parte dos terráqueos. A regra é a seguinte: o dia no planeta alienígena tem uma contagem apropriada para a quantidade de horas, minutos e segundos específicos; todos os relógios do planeta são analógicos e possuem apenas os ponteiros de horas e minutos; assim como na Terra, o relógio analógico alienígena dá duas voltas completas por dia; e por último, o contato só pode ser realizado nos momentos do dia em que os ponteiros dos relógios se alinham na mesma direção.



Entendendo as regras e horários de contato, o IFB pediu a sua ajuda para, dado um relógio analógico alienígena, determinar os horários em que o contato pode ser realizado.

Entrada

A entrada é composta por um único caso de testes. A única linha de entrada contém três inteiros H ($0 < H \leq 1000$), M ($0 < M \leq 1000$) e S ($0 < S \leq 5000$), separados por espaço, onde H representa a metade da quantidade de horas em um dia alienígena, M representa a quantidade de minutos em uma hora alienígena e S representa a quantidade de segundos em um minuto alienígena ($0 < H \leq M \leq S$).

Saída

A saída deve conter N linhas, ordenadas, onde cada uma deve conter um horário de contato, no formato $HH : MM$, onde HH representa a quantidade de horas e MM representa a quantidade de minutos do momento em que os ponteiros dos relógios se alinham na mesma direção.

Exemplo

Entrada	Saída

`8 100 200`	`00:00`
	`01:14`
	`02:28`
	`03:42`
	`04:57`
	`05:71`
	`06:85`

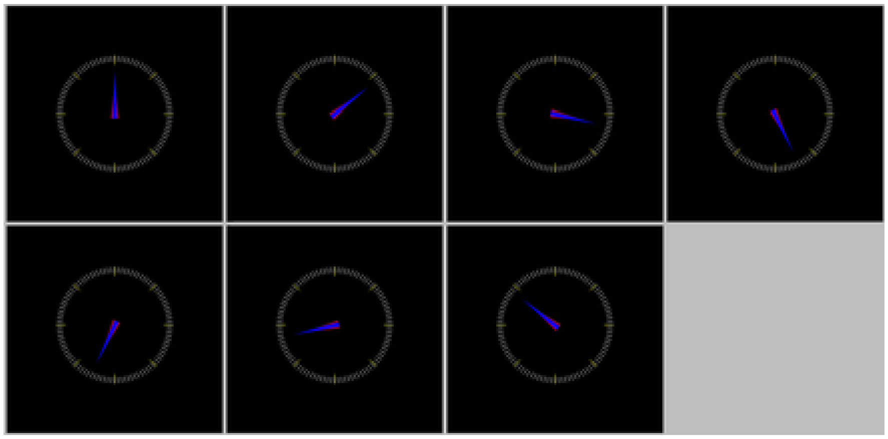
`12 60 60`	`00:00`
	`01:05`
	`02:10`
	`03:16`
	`04:21`
	`05:27`
	`06:32`
	`07:38`
	`08:43`
	`09:49`
	`10:54`

`14 50 100`	`00:00`
	`01:03`
	`02:07`
	`03:11`
	`04:15`
	`05:19`
	`06:23`
	`07:26`
	`08:30`
	`09:34`
	`10:38`
	`11:42`
	`12:46`

Notas

A figura da descrição representa o horário terráqueo, demonstrando os horários em que os ponteiros se alinham, cujo a entrada é dada por `12 60 60`, que é o segundo caso de testes.

Já o primeiro caso de testes, a imagem abaixo que representa os horários em que os ponteiros se encontram:



Problema F – Flores do Jardim

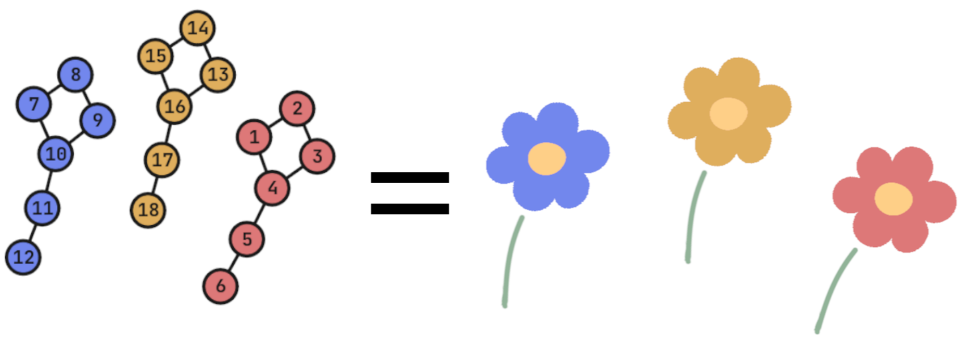
Limite de tempo: 1s
Limite de memória: 256MB

Autor: Caleb Martim

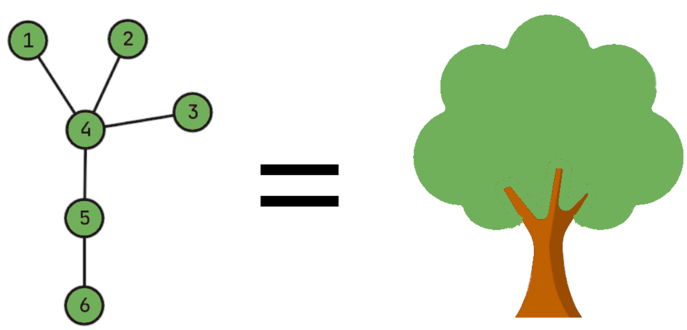
Nathália está cultivando um jardim e, para isso, comprou diversas sementes de flores. Porém, acidentalmente, algumas sementes que vieram em sua compra não eram de flores, mas sim de árvores! E ela só percebeu essa inconveniência quando todas as sementes já haviam sido plantadas.

O jardim de Nathália pode ser representado como um grafo não direcionado e não necessariamente conexo de N vértices e M arestas. Cada planta no jardim é um componente conexo diferente do grafo e cada vértice é unicamente identificado por algum número inteiro entre $1, 2, \ldots, N$.

Por exemplo, abaixo tem-se um exemplo de um grafo não direcionado e não conexo de 18 vértices e 18 arestas que pode representar um jardim contendo 3 plantas:



Vamos dizer que, se um componente composto por x vértices possui exatamente $x - 1$ arestas, este componente representa uma árvore. Por exemplo, o componente conexo abaixo contém 6 vértices e 5 arestas; portanto, ele representa uma árvore:



Nathália quer manter apenas flores em seu jardim; por isso, ela quer saber a quantidade de árvores que foram acidentalmente plantadas para poder removê-las com segurança e colocá-las em outro lugar (por ser uma ambientalista, ela não irá cortá-las).

Dada a representação do jardim de Nathália por um grafo de N vértices e M arestas, diga quantas árvores foram acidentalmente plantadas.

Entrada

A primeira linha contém dois inteiros N e M ($1 \leq N \leq 300,\ 0 \leq M \leq \frac{N \cdot (N-1)}{2}$) — o número de vértices e o número de aretas, respectivamente, que representam o jardim.

Cada uma das próximas M linhas contém dois inteiros A e B ($1 \leq A, B \leq N, A \neq B$) — indicando que há uma aresta entre os vértices A e B. É garantido há no máximo uma aresta entre qualquer par de vértices.

Saída

Imprima uma única linha contendo um número inteiro — a quantidade de árvores que foram plantadas no jardim.

Exemplo

Entrada	Saída
12 11 1 2 1 3 2 4 3 4 4 5 5 6 7 10 8 10 9 10 10 11 11 12	1
6 6 2 1 3 2 3 1 4 5 5 6 6 4	0
5 3 1 2 3 4 5 4	2
300 0	300
300 6 30 251 48 99 100 101 1 2 33 40 56 9	294

Problema G – Guloso da Cidade

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

Autor: Bruno Vargas

Ceilândia está comemorando mais um aniversário, e o prefeito preparou um bolo gigante que se estende por toda a praça principal, coberto por várias camadas de coberturas, cada uma com um intervalo específico $[L, R]$ ao longo do bolo. Cada cobertura possui um valor de "gostosura", que indica o quanto ela torna aquele ponto do bolo saboroso.

Emerson, o mais guloso da cidade, quer o pedaço mais gostoso, mas está de dieta e deve limitar a quantidade de coberturas que consome. Ao escolher um ponto para cortar o bolo, Emerson necessariamente pegará todas as coberturas associadas àquele ponto. Ele pode cortar o bolo em um único ponto para maximizar o valor total de "gostosura", respeitando o limite de K coberturas.

Ajude Emerson a encontrar o melhor ponto de corte que maximize a gostosura, respeitando a restrição de no máximo K coberturas. Se existir um ponto de corte que atenda a essas condições, imprima o maior valor de "gostosura" que ele pode obter.

Entrada

A primeira linha contém dois inteiros N e K ($1 \leq N, K \leq 10^5$), que representam respectivamente, a quantidade de coberturas no bolo e a quantidade máxima de cobertura que Emerson pode comer em um certo ponto. Depois seguem N linhas, contendo três inteiros L, R e G, ($1 \leq L \leq R \leq 10^6$), ($1 \leq G \leq 10^9$), que representam respectivamente, o segmento que a cobertura ocupa, e o seu valor de "gostosura".

Saída

A saída deve conter um número inteiro, indicando o maior valor de "gostosura" em um determinado ponto, respeitando a restrição que Emerson não pode comer mais que K coberturas naquele ponto. Se não for possível encontrar um ponto de corte que atenda a essas condições, imprima o valor "0".

Exemplo

Entrada	Saída
3 1 1 4 10 3 5 50 4 7 20	20
2 1 1 5 10 1 5 20	0

Problema H – Homiranho

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

Autor: Guilherme Ramos

O espetacular Homiranho, amigão da vizinhança, é a grande estrela da Carreta Furacão. Ele lidera o bando executando coreografias dançadas, malabarismos e acrobacias ao som de músicas animadas. E a galera pira quando toca sua música!

```
Homiranho, Homiranho,
Tem o duelo sempre ganho.
Gracioso, ele envolve
Muito mais que os Baryshnikov...
Cuidado! Lá vem Homiranho!

Capitão América, Mickey, Popeye
Todos ficam para trás.
Ele é mesmo tao bonzão?
Dança mais que o Fofão!
Cuidado! Lá vem Homiranho!
```

Entrada

A entrada consiste de duas linhas apresentando o contexto da Carreta Furacão. Nenhuma linha é vazia ou tem mais de 100 caracteres.

Saída

Para cada contexto, registre a reação da galera conforme os exemplos.

Exemplo

Entrada	Saída
`Ele e mesmo tao bonzao?` `Danca mais que o Fofao!`	`Cuidado! La vem Homiranho!`
`Lanca teias de qualquer tamanho` `Vence os duelos todo ano`	`Cuidado! La vem Homiranho!`
`Se sacode pelas teias?` `Danca pulsa em suas veias!`	`Cuidado! La vem Homiranho!`

Notas

Atenção, não use caracteres especiais ou com acentuação na saída.

Problema I – Inversões Mágicas de Botelho

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

Autor: Alberto Neto

O grande mago Botelho está realizando um experimento em seu laboratório mágico. Ele possui uma string binária s de tamanho n, que representa uma mensagem codificada. Para tornar as coisas mais interessantes, Botelho possui q pares de intervalos (l_i, r_i), cada um capaz de inverter todos os bits da string no intervalo especificado.

Botelho está intrigado: quantas strings diferentes ele pode criar a partir de sua string original usando qualquer combinação das operações mágicas disponíveis? Como o número de possibilidades é muito grande, ele pediu que você calculasse a resposta módulo 998244353.

Ajude o mago Botelho a descobrir a diversidade de mensagens que ele pode gerar com seus feitiços!

Entrada

A primeira linha de entrada contém dois inteiros n e q $(1 \leq n, q \leq 1000)$ — o tamanho da string binária s e a quantidade de pares, respectivamente.

Cada uma das próximas q linhas de entrada contém dois inteiros l_i, q_i cada $(1 \leq l_i \leq r_i \leq n)$ — os valores de cada intervalo.

Saída

Imprima um único inteiro — a quantidade de strings que podem ser obtidas, módulo 998244353.

Exemplo

Entrada	Saída
6 2 1 3 2 4	4
6 3 1 3 4 4 1 4	4
10 5 1 2 1 7 4 7 3 9 4 9	16

Problema J – Jogos Universitários

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

Autor: Vinicius Borges

Começaram os Jogos Universitários, uma competição nacional que conta com M modalidades esportivas! Saad é o chefe da delegação do IFB Taguatinga e definiu um objetivo ousado: conquistar exatamente N medalhas de ouro ao final dos Jogos Universitários.

Cada modalidade esportiva é numerada por um inteiro de 1 a M. As regras dos Jogos Universitários estabelecem que é possível conquistar em cada modalidade esportiva uma quantidade fixa de medalhas de ouro para cada vitória. No entanto, a instituição pode competir na i-ésima modalidade, no máximo, l_i vezes.

Saad recebeu a lista de modalidades do Comitê Organizador dos Jogos Universitários. Ele verificou que a vitória em cada modalidade i dá direito a o_i de medalhas de ouro. Determine a menor quantidade possível de competições que o IFB Taguatinga precisa vencer para conquistar exatamente N medalhas de ouro, sabendo-se das restrições de competição em cada modalidade esportiva.

Entrada

A primeira linha da entrada contém dois números inteiros N e M $(1 \leq N \leq 10^3, 1 \leq M \leq 100)$, representando o número de medalhas de ouro que Saad planeja ganhar e a quantidade de modalidades esportivas nos Jogos Universitários.

A segunda linha contém M números inteiros separados por espaço $o_1, \ldots, o_M$ $(1 \leq o_i \leq 10^3)$, representando a quantidade de medalhas de ouro obtidas no caso da vitória na i-ésima modalidade esportiva.

A terceira linha contém M inteiros $l_1, \ldots, l_M$ $(1 \leq l_i \leq 100)$, indicando a quantidade máxima de competições que pode ser realizada na i-ésima modalidade esportiva.

Saída

Imprima um único número inteiro indicando a menor quantidade possível de competições que o IFB Taguatinga precisa vencer para conquistar exatamente N medalhas de ouro, considerando que existe um limite de medalhas de ouro que podem ser conquistadas em cada modalidade. Se não for possível atingir esse objetivo, imprima -1.

Exemplo

Entrada	Saída

`3 2`
`1 2`
`1 1`
	`2`

`6 3`
`2 3 4`
`4 3 2`
	`2`

`10 2`
`3 4`
`1 1`
	`-1`

`12 4`
`1 5 1 3`
`1 2 5 3`
	`4`

Notas

No primeiro exemplo de teste, precisamos conquistar exatamente 3 medalhas de ouro e temos 2 modalidades esportivas. Pode-se competir na modalidade 1 uma vez para ganhar 1 medalha e competir na modalidade 2 uma vez para ganhar 2 medalhas, resultando em 3 medalhas de ouro. Por fim, a resposta é 2.

No segundo exemplo de teste, pode-se competir na modalidade 2 em duas competições para ganhar um total de 6 medalhas. Assim, o número mínimo de competições é 2.

No terceiro exemplo, não é possível conquistar exatamente 10 medalhas de ouro com as modalidades esportivas disponíveis e as restrições descritas.

Problema K – Kátia e os fatoriais

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

Autor: Edson Alves

Kátia estava resolvendo exercícios envolvendo fatoriais e observou que é bastante comum que a representação decimal do fatorial de um número positivo N tenha vários zeros à direita. De fato, para $N \geq 5$, há pelo menos um zero à direita na representação decimal de $N!$.

Mais curioso, porém, é que existem inteiros positivos M para os quais nenhum fatorial tem exatamente M zeros à direita! Por exemplo, não há inteiros positivos N tais que $N!$ tenha exatamente 5 zeros à direita em sua representação decimal.

Para continuar investigando estas questões, Kátia quer sua ajuda, e pediu que você escrevesse um programa que recebe um número inteiro M e que retorne, se existir, um inteiro positivo N tal que $N!$ tenha exatamente M zeros à direita em sua representação decimal.

Entrada

A entrada é composta por uma única linha, contendo o valor do inteiro positivo M ($1 \leq M \leq 10^9$).

Saída

Imprima, em uma linha, um inteiro positivo N tal que $N!$ tenha exatamente M zeros à direita em sua representação decimal. Caso não exista tal número, imprima o valor -1.

Exemplo

Entrada	Saída
2	13
5	-1
314159268	1256637102

Notas

No primeiro caso, $13! = 6227020800$. Observe que 12 seria uma resposta igualmente válida, pois $12! = 479001600$.

No segundo caso, não existe um inteiro positivo N tal que $N!$ tenha exatamente 5 zeros à direita em sua representação decimal.

Problema L – Lago Oeste

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

Autor: Daniel Saad Nogueira Nunes

O Lago Oeste é uma região de Brasília menor famosa que seus primos, Lago Norte e Lago Sul, mas é dono de uma beleza inestimável. Essa região possui diversas atividades ao ar livre, além de áreas de ecoturismo.

Adjacente ao Lago Oeste está a Floresta Nacional, sendo uma área de conservação do Distrito Federal do cerrado brasileiro. Infelizmente na época seca, a floresta nacional sofre com muitos incêndios. Para tratar das queimadas, o corpo de bombeiros do DF conta com a ajuda de drones, que identificam nas imagens aéreas as regiões em combustão para poderem jogar água, mitigando os incêndios em áreas de difícil acesso.

A imagem abaixo foi extraída de um drone. Cada símbolo dessa imagem representa um hectare de terra. As regiões representadas por "`#`" indicam áreas verdes, enquanto aquelas representadas por "`F`" indicam áreas em chamas.

```
##########
###FF#####
##FFFF####
##FFFF#FF#
###FF##FF#
#######FF#
```

No caso da imagem, existem duas regiões distintas em chamas na floresta, requerendo duas viagens para que o drone possa se abastecer de água. Dois hectares adjacentes, na vertical ou horizontal, marcados com `F`, são considerados parte do mesmo foco de incêndio.

Ajude o corpo de bombeiros a programar os drones e identificar o número de viagens necessárias para apagar todos os focos de incêndio.

Entrada

A primeira linha da entrada possui dois inteiros N e M, separados por um espaço, que indicam, respectivamente, o número de linhas e colunas da imagem.

Em seguida, há N linhas, com M caracteres, cada uma descrevendo uma linha da imagem.

Restrições:

- $1 \leq N, M \leq 300$
- Os únicos símbolos que podem aparecer na imagem são "`#`" e "`F`".

Saída

Imprima uma única linha com o número de viagens efetuadas pelos drones.

Exemplo

Entrada	Saída
6 10 ########## ###FF##### ##FFFF#### ##FFFF#FF# ###FF##FF# #######FF#	2

Notas

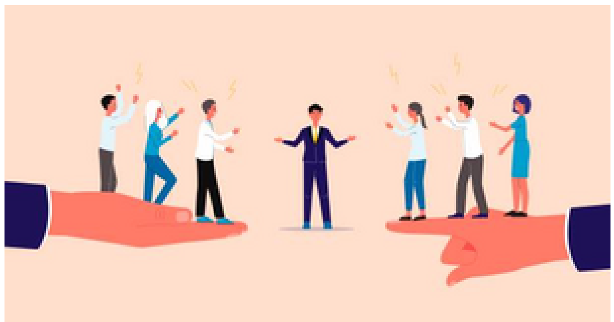
O primeiro exemplo reflete exatamente a situação do enunciado.

Problema M – Mediação

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

Autor: Jeremias Moreira Gomes

Os debates políticos não param, e as regras de cada debate, em cada emissora, estão cada vez mais malucas de acompanhar.



Dessa vez o canal Informação e Fatos Broadcast (IFB) resolveu adaptar as regras de mediação para que os próprios candidatos conseguissem se controlar durante as sabatinas, e para isso haverá um candidato mediador que será resposável por manter a ordem durante o debate.

Para escolher o candidato mediador, o IFB resolveu utilizar o seguinte critério: o valor do carisma de cada candidato. Como em cada debate os candidatos são sorteados previamente em suas posições, o objetivo é encontrar o mediador ideal separando os candidatos em dois grupos, da seguinte forma: cada grupo deverá ter a mesma quantidade total de carisma, independente do tamanho do grupo (mínimo de um candidato) e o mediador será o candidato que irá separar os dois grupos, sem que suas posições sejam alteradas.

Assim, o IFB resolveu pedir a sua ajuda para verificar se, dados os carimas de cada candidato em suas posições, há um candidato que possa ser o mediador ideal, para evitar confusões durante o debate.

Entrada

A entrada contém vários casos de teste. A primeira linha contém um inteiro D ($1 \leq D \leq 100$), que representa a quantidade de casos de teste. Cada caso de teste é composto por duas linhas. A primeira linha contém um inteiro N ($3 \leq N \leq 2 * 10^4$), que representa a quantidade de candidatos. A segunda linha contém N inteiros c_i ($1 \leq c_i \leq 1000$), separados por espaço, que representam o carisma de cada candidato.

Saída

Para cada caso de teste, imprima uma linha contendo o índice do candidato que pode ser o mediador ideal, ou a frase "A cadeira, usa a cadeira!" caso não haja um candidato que possa ser o mediador ideal.

Exemplo

Entrada	Saída
1 7 8 9 4 6 8 1 2	3
3 7 2 6 9 4 7 1 9 5 1 2 3 4 5 4 12 4 9 3	4 A cadeira, usa a cadeira! 2

Notas

No primeiro caso de testes, que só possui a representação de um único debate, é possivel dividir em dois grupos, nos quais o mediador é aquele que possui carisma 4, divindo os grupos em `[8, 9]` e `[6, 8, 1, 2]`.