



**INSTITUTO FEDERAL
DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA**
São Paulo

2024

Caderno de Problemas

Fase Final - VII Maratona de Programação INTERIF

Piracicaba - 30 de Novembro de 2024

A PROVA TEM DURAÇÃO DE **3:00** HORAS

APOIADORES



Informações gerais

Este caderno de tarefas é composto por 18 páginas (não contando a capa), numeradas de 1 a 18. Verifique se o caderno está completo.

Nome do programa

Sua solução deve ser nomeada *codename.c*, *codename.cpp*, *codename.java*, *codename.py*, onde *codename* identifica o nome do arquivo dado no enunciado de cada problema.

Sobre a entrada

- A entrada deve ser lida da entrada padrão.
- A entrada consiste em exatamente um caso de teste, que é descrito usando uma quantidade de linhas que depende do problema. O formato da entrada é como descrito em cada problema. A entrada não contém nenhum conteúdo extra.
- Todas as linhas da entrada, incluindo a última, terminam com o caractere de fim de linha (`\n`).
- A entrada não contém linhas vazias.
- Quando a entrada contém múltiplos valores separados por espaços, existe exatamente um espaço em branco entre dois valores consecutivos na mesma linha.

Sobre a saída

- A saída de seu programa deve ser escrita na saída padrão.
- Espaços em branco só devem ser colocados quando solicitado.
- Cada linha, incluindo a última, deve conter o caractere final-de-linha (`\n`).

Regras

- Só é permitida a comunicação entre os membros de um mesmo grupo.
- Não é permitida a comunicação com o técnico (coach) do time.
- Eventuais dúvidas sobre a prova utilizar o menu “clarification” do sistema de submissão.

Ambiente computacional

O sistema de correção das submissões e o Sistema Operacional do computador da fase final será executado utilizando a distribuição Ubuntu GNU/Linux Ubuntu 22.04, tendo os seguintes compiladores/interpretadores configurados:

	Sistema de correção	Computador da fase final
C	GCC 11.4.0	GCC 11.3.0
C++	G++ 11.4.0	G++ 11.3.0
Python	Python 3.10.12	Python 3.10.6
Java	Java 17.0.10	Java 17.0.5
C#	Mono JIT 6.12	Mono JIT 6.12

Problema A

Sem Dinheiro para Compras

Autor: Tiago Alexandre Dócusse (IFSP – Barretos)

Arquivo: mercado.(c|cpp|cs|java|py)

Timelimit: 1

Marina e Rafael foram fazer compras no mercado, e bem na hora de pagar, viram que fizeram algumas contas erradas e não possuem todo o dinheiro para pagar as compras que fizeram. Eles possuem uma mesada restrita, só recebendo cédulas de dinheiro, e nenhuma moeda. Dessa forma, meticolosos que são, apenas compram no mercado dos inteiros, onde os preços dos produtos não possuem casas decimais.

O operador do caixa informou a eles o valor que estava faltando, e como não possuem mais dinheiro, eles decidiram retirar produtos da compra em que a soma dos seus preços eram exatamente o valor que faltou. Você, o próximo da fila a ser atendido no caixa do mercado, não quer esperar que eles verifiquem item a item qual valor deve ser removido, e ofereceu uma solução para os irmãos: escolher dois produtos cuja soma de valores é o valor restante da compra, e retirar eles da lista de compra.

Rafael e Marina gostaram da ideia. No entanto, eles ainda não sabem programar, e pediram a você, bom programador que é, que faça um programa para informar os preços dos produtos a serem retirados no carrinho. Você consegue ajudar os irmãos?

Entrada

A entrada é formada inicialmente por um número inteiro x ($0 < x \leq 10^5$), indicado o valor que falta para completar a compra. Em seguida, informa-se um número inteiro y ($1 \leq y \leq 2^{32} - 1$), indicando o total de produtos comprados pelos irmãos. Logo após, seguem-se y valores inteiros z_i ($1 \leq z_i \leq 2^{31} - 2$), indicando o valor de cada produto. Garante-se que os preços dos produtos são únicos.

Saída

A saída deve exibir, em ordem crescente e separados por um espaço em branco, a ordem na esteira de compras de cada um dos produtos que serão removidos da compra, de forma que Marina e Rafael consigam pagar o valor do restante dos produtos. A ordem da saída é iniciada em zero e incrementada em uma unidade. Garante-se que a saída é a única possível.

Exemplos

Entrada	Saída
13 4 2 7 1 6	1 3

Problema B

Banco Comunitário

Autor: Carlos José de Almeida Pereira (IFSP – Campus São Carlos)

Arquivo: banco.(c|cpp|cs|java|py)

Timelimit: 1

Joilson organizou um banco comunitário no seu bairro, com uma moeda específica, o J\$, baseada nos serviços que os próprios moradores prestam e nos produtos artesanais que vendem, de forma que possam trocar esses serviços e produtos entre si de uma forma mais facilitada, e sem os custos extras que seriam cobrados pela intermediação de entidades financeiras. Estava indo tudo bem, até que o número de problemas devido a falta de fundos dos moradores começasse a atrapalhar a operação do banco. E não era nem por má-fé, mas sim porque os moradores não lembravam o total de seus valores armazenados, para poder calcular direitinho seus gastos. Joilson então teve uma ideia: ele começou a manter o histórico de operações de cada morador, e a cada x operações calculava o valor mediano dessas operações. Se a operação $x+1$ fosse igual ou maior a duas vezes essa mediana, ele enviava um aviso ao morador, para evitar a explosão das dívidas.

Por exemplo, seja o seguinte histórico de valores em J\$ nas operações realizadas por um certo morador:

2 3 4 2 3 6 8 4 5

Se o número de operações que vão ser avaliadas para emitir o aviso for 5, teremos:

[2 3 4 2 3] 6 8 4 5 \rightarrow Mediana = 3

Próxima operação: 6, que é igual a ($2 * mediana = 6$), então um aviso foi enviado.

2 [3 4 2 3 6] 8 4 5 \rightarrow Mediana = 3

Próxima operação: 8, que é maior que ($2 * mediana = 6$), então um aviso foi enviado.

2 3 [4 2 3 6 8] 4 5 \rightarrow Mediana = 4

Próxima operação: 4, que não é maior que ($2 * mediana = 8$), então não foi enviado aviso.

2 3 4 [2 3 6 8 4] 5 \rightarrow Mediana = 4

Próxima operação: 5, que não é maior que ($2 * mediana = 8$), então não foi enviado aviso.

Total de avisos emitidos: 2

Ajude Joilson a fazer um programa que, dado um histórico de valores de operações, e a quantidade de valores que serão avaliados por vez, diga quantos avisos foram enviados.

Entrada

A entrada é composta por 2 linhas. A primeira linha tem 2 números inteiros, X ($2 \leq X \leq 50$) e Y ($1 \leq Y < X$), Y ímpar, onde X representa o número total de operações no histórico do morador, e Y quantas operações serão consideradas para emitir ou não ou aviso. Na segunda linha temos X números inteiros Z ($1 \leq Z \leq 1000$) indicando o valor de cada operação no histórico do morador.

Saída

A saída é composta por uma linha, contendo apenas um número inteiro, indicando quantos avisos foram emitidos para este morador com base no seu histórico.

Exemplos

Entrada	Saída
9 5 2 3 4 2 3 6 8 4 5	2

Entrada	Saída
2 1 2 1	0

Entrada	Saída
10 9 1 2 3 4 5 6 7 8 9 9	0

Problema C

Chocolate Mineiro

Autor: Jones Mendonça de Souza (IFSP – Barretos)

Arquivo: chocolate.(c|cpp|cs|java|py)

Timelimit: 1



A pequena cidade de Planura, no interior de Minas Gerais, é cercada de fazendas e fábricas de queijo. Ao chegar na cidade, você é recebido no portal com um delicioso pão de queijo recém-assado, uma tradição turística que o prefeito Jurandir Leite guarda até os dias de hoje. Mas essa iguaria, que é apreciada pelos turistas, já se tornou uma rotina para os alunos da escola da cidade, que recebem diariamente queijo fresco no recreio e pão de queijo com linguiça na merenda escolar. Isso tem comprometido o coeficiente de rendimento escolar dos alunos da cidade, e tem sido motivo de preocupação para o prefeito Jurandir, que procura uma solução para o problema.

A diretora da escola, Professora Lucrécia Quesadilla, teve uma ideia para incentivar os alunos a se esforçarem em seus estudos. Ela decidiu que aqueles que obtivessem uma média final acima de 6 pontos nas disciplinas de matemática e português, receberiam uma recompensa especial: uma ou mais barras de chocolates Colombienses. A notícia se espalhou rapidamente pela escola, e os alunos ficaram motivados para se dedicarem aos estudos. Além de aprenderem sobre matemática e português, agora tinham a oportunidade de experimentar os tão famosos chocolates artesanais da cidade de Colômbia, localizada no interior de São Paulo.

No entanto, isso era apenas uma ideia que dependia do orçamento da prefeitura e da contribuição das fábricas de chocolate da cidade de Colômbia, já que estes chocolates são nobres e representam um alto custo para a prefeitura. Diante dessa situação, a diretora resolveu negociar o preço da barra de chocolate com as fábricas, mas tinha certeza que não seria uma tarefa fácil, já que estes chocolates são normalmente exportados para Itália com preços mais atrativos do que seria oferecido pela prefeitura de Planura. A grande preocupação da diretora, é se o prefeito não tiver orçamento suficiente para comprar os chocolates, ou pior ainda, se nenhum aluno alcançar a média necessária, o que deixaria as crianças com zero barras de chocolate. Na verdade, tanto o prefeito quanto a diretora, estão na expectativa de receber uma doação. Se isso acontecer, o custo da barra de chocolate será zero, o prefeito não irá gastar seu orçamento e as crianças que alcançaram a média ganharão 1 barra de chocolate cada.

Mas vamos ao que interessa, a diretora se emocionou ao ver a felicidade dos alunos, recebeu diversas ligações dos pais, apoiando sua ação e agradecendo por deixar nos olhos de seus filhos, a esperança de trocar o habitual pão de queijo pela descoberta de um novo prazer gastronômico. Como se não bastasse, a diretora ainda precisa de sua ajuda, pois dependendo do cenário a quantidade de chocolates poderá ser altíssima, o que pode inviabilizar a contagem e separação. Ela precisa de um programa de computador que seja capaz de fazer uma “divisão justa” das barras de chocolates, ou

seja, ela deseja que o chocolate comprado pelo prefeito Jurandir Leite seja dividido em quantidades iguais para cada aluno premiado. Se sobrar uma quantidade menor que não possa ser igualmente dividida entre os alunos premiados, estes deverão ser doados ao orfanato da cidade.

Entrada

A entrada é composta por uma única linha contendo três valores (a , b e c), separados por um espaço em branco, tais que $0 \leq a, b, c \leq 10^{19}$. O primeiro valor da esquerda para direita, representado por (a) indica a quantidade de alunos que serão premiados com barras de chocolate. O segundo valor da esquerda para direita, representado por (b), denota o orçamento liberado pelo prefeito Jurandir Leite para a compra das barras de chocolate. Por fim, representado por (c), o valor unitário da barra de chocolate, negociado pela diretora da escola Lucrécia Quesadilla.

Saída

Exiba a quantidade de barras de chocolate que cada aluno deverá receber depois da “divisão justa”, proposta pela diretora Lucrécia Quesadilla.

Exemplos

Entrada 8 502.01 13.90	Saída 4
Entrada 23 1700.08 22.30	Saída 3

Problema D

IFSP Binary Coin

Autor: Murilo Vargas da Silva (IFSP – Campus Birigui)

Arquivo: coin.(c|cpp|cs|java|py)

Timelimit: 1



Com o crescente interesse das pessoas por moedas digitais, estudantes do IFSP decidiram criar sua própria criptomoeda, o nome que escolheram é “IFSP Binary Coin”. A segurança dessa moeda é bem interessante, ela é baseada na combinação de informações binárias de todos os depósitos realizados em uma determinada conta para a geração da chave privada de acesso à conta de cada usuário.

Sua tarefa é desenvolver um algoritmo que gere a chave privada de cada carteira baseada nos depósitos realizados. Basicamente, cada conta poder ter N depósitos, esses depósitos sempre serão números inteiros. A chave privada de uma conta será gerada a partir da contagem e acumulação dos bits de cada posição dos inteiros que representam os depósitos, e a chave privada será uma expressão que se resolvida dará o saldo contido na conta, apenas as posições onde a contagem acumulada seja maior ou igual a 1 devem fazer parte da expressão da chave privada.

Exemplo 1, dois depósitos (8 e 1):

Inteiro	Binário
8	1 0 0 0
1	0 0 0 1
Contagem bits	1 0 0 1
chave: $1 * 2^3 + 1 * 2^0 = 9$	

Exemplo 2, quatro depósitos (1, 2, 4 e 8):

Inteiro	Binário
1	0 0 0 1
2	0 0 1 0
4	0 1 0 0
8	1 0 0 0
Contagem bits	1 1 1 1
chave: $1 * 2^3 + 1 * 2^2 + 1 * 2^1 + 1 * 2^0 = 15$	

Exemplo 3, quatro depósitos (55, 121, 200 e 255):

Inteiro	Binário
55	0 0 1 1 0 1 1 1
121	0 1 1 1 1 0 0 1
200	1 1 0 0 1 0 0 0
255	1 1 1 1 1 1 1 1
Contagem bits	2 3 3 3 3 2 2 3
chave: $2 * 2^7 + 3 * 2^6 + 3 * 2^5 + 3 * 2^4 + 3 * 2^3 + 2 * 2^2 + 2 * 2^1 + 3 * 2^0 = 631$	

Entrada

A primeira linha da entrada contém um inteiro $N (1 \leq N \leq 10^5)$ e a segunda linha contém N inteiros, separados por espaço, correspondentes à lista de depósitos realizados em uma determinada conta da moeda “IFSP Binary Coin”, sendo $a_1, \dots, a_N (0 \leq a_i \leq 10^9)$.

Saída

Seu programa deve imprimir uma única linha com a expressão que forma a chave privada da conta

Exemplos

Entrada 2 8 1	Saída chave: $1*2^3+1*2^0=9$
Entrada 4 1 2 4 8	Saída chave: $1*2^3+1*2^2+1*2^1+1*2^0=15$
Entrada 4 55 121 200 255	Saída chave: $2*2^7+3*2^6+3*2^5+3*2^4+3*2^3+2*2^2+2*2^1+3*2^0=631$

Problema E

Defeito no Teclado

Autor: João Paulo Lemos Escola (IFSP – Campus Barretos)

Arquivo: defeito.(c|cpp|cs|java|py)

Timelimit: 1

O DBA da sua equipe esteve analisando os registros de uma tabela do banco de dados e constatou que houve um problema de integridade no lançamento dos nomes de alguns produtos. Os nomes foram armazenados em um formato duvidoso, misturando letras e números. Um dos membros da sua equipe conseguiu descobrir que as letras, que aparecem em cada nome de produto, estão íntegras e que o problema, na verdade, é com os códigos numéricos que aparecem entre as letras. Esses códigos, segundo ele, são os códigos da tabela ASCII que não foram devidamente convertidos em caracteres durante a captura, provavelmente um defeito no teclado do computador do usuário que fez as inserções dos registros. Seu trabalho é criar um programa que receba esses nomes armazenados incorretamente e descubra qual é o nome correto, a fim de que a lista seja devidamente recuperada e posteriormente substituída no banco de dados.

Entrada

A entrada consiste de um valor inteiro N ($1 \leq N \leq 1000$) que representa a quantidade de nomes incorretos. Em seguida, há N linhas, em que cada linha contém um nome incorreto de produto (tamanho máximo de 255 caracteres). Considere que as frases são compostas apenas por números, caracteres minúsculos de “a” a “z” e caracteres de espaço.

Saída

A saída de seu programa deve conter a palavra resultante.

Exemplos

Entrada	Saída
2	
b97n97n97	banana
116101s116101	teste

Problema F

Radar em Interifnópolis

Autor: Jorge Francisco Cutigi (IFSP – Campus São Carlos)

Arquivo: radar.(c|cpp|cs|java|py)

Timelimit: 1

A cidade de Interifnópolis é conhecida por suas avenidas movimentadas e pela necessidade de manter a ordem no trânsito. Entre seus habitantes, estavam Otto, um jovem entusiasta da tecnologia, e seu pai, Guilherme, um experiente motorista que sempre respeitou as leis de trânsito.

Certa manhã, Otto e Guilherme decidiram dar uma volta de carro pela cidade. Enquanto cruzavam a avenida principal, Guilherme comentou: “Sabe, Otto, é sempre importante respeitar o limite de velocidade. Aqui na nossa cidade temos radares em todas as vias, e o limite de velocidade nessas vias é de L km/h.” Otto, curioso e sempre pronto para aprender, perguntou sobre as consequências de ultrapassar esse limite.

Guilherme explicou: “Se um motorista ultrapassa essa velocidade, o radar detecta e a cidade aplica uma multa de R\$ M . Além disso, para cada km/h que passar do limite, um adicional de R\$ A é cobrado”. Guilherme continuou sua explicação. “Por exemplo, considere que o valor da multa M é R\$ 150 e o valor A por km/h adicional é R\$ 10. Se o limite L da via é 60 km/h e passamos a 65 km/h, vamos pagar R\$ 150 mais 5 vezes R\$ 10, ou seja, R\$ 200 no total”. Otto, ouviu atentamente seu pai Guilherme. Como ele é fascinado por matemática e programação, decidiu que queria fazer um programa para calcular o valor total da multa caso alguém se excedesse.

Entrada

A entrada consiste de 4 valores inteiros, lidos na ordem apresentada abaixo em linhas separadas:

- L : Limite de velocidade da via ($0 < L < 1000$)
- M : Valor da multa ($0 < M < 1000$)
- A : Valor adicional por km/h acima do limite ($0 < A < 1000$)
- V : Velocidade registrada pelo radar ($0 < V < 1000$)

Saída

A saída consiste da impressão de um número inteiro que representa o valor da multa a ser paga. Se não houve multa, deve-se imprimir o valor 0.

Exemplos

Entrada	Saída
60 150 10 65	200

Entrada	Saída
60 150 10 50	0

Problema G

Que horas são?

Autor: Daniel Corrêa Lobato (IFSP – Campus São José do Rio Preto)

Arquivo: horas.(c|cpp|cs|java|py)

Timelimit: 1

“Em uma das duas mãos, ele segura um dispositivo para eu ver. É... eu não sei o que é. É um cilindro, com uns trinta centímetros de comprimento, e talvez uns quinze de largura. O cilindro tem cinco janelas quadradas alinhadas. Dentro de cada janela há uma forma. Acho que talvez sejam letras... O símbolo da direita gira e é substituído por um novo símbolo. Depois de alguns segundos, acontece de novo. E de novo.

– É um relógio!

Observo o relógio erídio por um tempo. Rocky o segura para que eu possa ver. O ciclo dos símbolos acontece na janela da direita. Estão em um rotor. Depois de um tempo, o rotor à esquerda muda de posição. Aha!

Até onde eu consegui entender, o rotor da direita muda a cada dois segundos. Pouco mais de dois segundos, eu acho. O ciclo passa por seis símbolos distintos antes de começar a repetir: ℓ , I , V , λ , $+$, \forall nessa ordem. Sempre que chega ao ℓ , o rotor seguinte à esquerda avança um passo.

Então, o relógio dele é bem intuitivo para mim, e funciona como um hodômetro. O zero deles claramente é ℓ . Até onde entendi, I é 1, V é 2, λ é 3, $+$ é 4 e \forall é 5. E quanto aos números 6 a 9? Não existem. Erídios usam base seis.

De todas as coisas que ensino aos meus alunos, bases numéricas são as mais difíceis para eles entenderem de verdade. Não há nada de especial no número dez. Temos dez dígitos únicos porque temos dez dedos. Simples assim. Rockies têm três dedos em cada mão.”

Os parágrafos acima são trechos do livro “Devoradores de Estrelas”, de Andy Weir (WEIR, A.. **Devoradores de estrelas**. Rio de Janeiro: Suma, 2021. 424p.). Nessa parte da história, o personagem principal acaba de descobrir como os seres do planeta Erídio fazem para contar o tempo. Assim como os erídios utilizam base 6, outros povos poderiam utilizar outras bases numéricas, e cada símbolo do primeiro rotor fica por um determinado tempo antes de ser alterado pelo próximo. Assim, no exemplo do livro, o relógio erídio tem seis símbolos, e cada símbolo é alterado no primeiro rotor a cada dois segundos. Assim, um relógio erídio que marque $\ell\ell VI$ corresponde a 26 segundos terrestres. O seu papel é construir um programa que converta um tempo indicado por um relógio alienígena em segundos terrestres.

Entrada

Cada caso de teste é composto por um conjunto de 4 linhas. A primeira linha contém um número inteiro $B(1 \leq B \leq 128)$ que indica a base numérica do relógio alienígena. A segunda linha contém uma cadeia com B caracteres diferentes indicando, cada um deles, o símbolo utilizado para representar os valores de 0 até $B - 1$, nesta ordem. A terceira linha contém uma cadeia com até 100 caracteres, cada um deles pertencente ao conjunto de caracteres da segunda linha, que corresponde ao tempo no relógio alienígena. A quarta linha contém, por fim, um número real $S(0.01 \leq S \leq 10)$ com a duração de cada símbolo no primeiro rotor, em segundos terrestres

Saída

Uma única linha contendo um valor real, com duas casas decimais, indicando o tempo terrestre equivalente ao tempo alienígena indicado.

Exemplos

Entrada	Saída
6 ABCDEF AAACB 2	26.00
2 AB BAAABAB 2.5	172.50

Problema H

Wi-Fi Boa Vizinhança

Autor: Carlos José de Almeida Pereira (IFSP – Campus São Carlos)

Arquivo: wifi.(c|cpp|cs|java|py)

Timelimit: 1

Na tranquila rua onde Joilson vive, a conexão com a Internet é uma bênção que poucos têm a sorte de desfrutar. Enquanto ele se deleita com uma largura de banda gigantesca, seus vizinhos enfrentam a frustração de não ter acesso à rede. Empático e generoso, Joilson decidiu que poderia fazer a diferença e compartilhar sua Internet com aqueles que moram ao seu redor.






No entanto, a tarefa não é tão simples quanto parece. Joilson sabe que, para compartilhar seu sinal de Internet, ele precisará investir em antenas repetidoras, que têm um alcance limitado e um custo alto. Para minimizar o seu custo e garantir que todos os seus vizinhos possam se conectar, Joilson precisa calcular o menor número de antenas necessárias para cobrir todos os lotes com casas da sua rua.

Dado que cada modelo de antena tem uma capacidade específica, que determina quantos lotes podem ser cobertos lateralmente (além da própria casa onde está instalada), e também ao fato que nem todos os lotes da rua são ocupados por casas, a pergunta que Joilson se faz é: qual o menor número de antenas ele deve comprar para garantir que todos na sua rua possam desfrutar de uma conexão à Internet?






Por exemplo, seja uma rua com 8 lotes:

lote 1	lote 2	lote 3	lote 4	lote 5	lote 6	lote 7	lote 8
--------	--------	--------	--------	--------	--------	--------	--------

Temos nessa rua, 5 lotes com casas, os lotes 1, 3, 6, 7 e 8:

lote 1	lote 2	lote 3	lote 4	lote 5	lote 6	lote 7	lote 8
							

Seja um modelo de antena que tenha capacidade de alcançar 2 lotes (pra direita ou esquerda, além da própria casa onde está instalada). Neste caso, o menor número de antenas que cobre todos os lotes são 2 antenas, colocadas, por exemplo, nos lotes 3 e 7:

lote 1	lote 2	lote 3	lote 4	lote 5	lote 6	lote 7	lote 8
							

Ajude Joilson a fazer um programa que, dada a configuração de casas de uma rua, e uma certa capacidade de cobertura lateral de um modelo de antena, calcule qual é o menor número de antenas para cobrir todas as casas.

Entrada

A primeira linha tem 2 números inteiros $X(1 \leq X \leq 20)$ e $Y(1 \leq Y \leq 20)$, onde X representa o número de casas que existem nesta rua, e Y a capacidade lateral da antena.

Na segunda linha temos X números inteiros $Z(1 \leq Z \leq 20)$ indicando o número de um lote ocupado por uma casa. Os números estão em ordem crescente.

Saída

A saída é composta por uma linha, contendo apenas um número inteiro $S(S \geq 1)$, indicando qual o menor número de antenas que são necessárias para cobrir de sinal todas as casas dessa rua.

Exemplos

Entrada 5 2 1 3 6 7 8	Saída 2
Entrada 6 2 1 3 5 7 10 20	Saída 4
Entrada 20 3 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20	Saída 3

Problema I

Foca em Mim

Autor: Daniel Corrêa Lobato (IFSP – São José do Rio Preto)

Arquivo: foca.(c|cpp|cs|java|py)

Timelimit: 1

Em um ambiente cada vez mais interconectado, as redes sociais desempenham um papel fundamental na comunicação e interação entre as pessoas. Um dos aspectos cruciais dessas redes é a noção de influência, medida muitas vezes pelo número de seguidores que um usuário possui. Mas só o número de seguidores não é uma boa métrica (diga isso para aquele seu amigo seguido por milhares de fakes). A importância, ou influência, de um usuário pode ser calculada de forma iterativa utilizando o conceito de PageRank.

O PageRank é um valor numérico atribuído a cada usuário, refletindo sua importância dentro da rede social com base na quantidade e qualidade dos seus seguidores. Em essência, um usuário é considerado mais influente se for seguido por outros usuários também considerados influentes. No início do algoritmo, todos os usuários têm a mesma importância, que é definida como 1. Durante cada iteração do algoritmo, o novo PageRank do usuário A é calculado como a soma das contribuições de todos os usuários que o seguem, excluindo o próprio usuário A. A contribuição de cada usuário (U) para o PageRank de A é calculada dividindo o PageRank de U pelo número total de usuários que U segue. Esse cálculo é realizado de forma atômica, o que significa que na iteração atual (i), os valores de PageRank utilizados são os resultantes da iteração anterior (i-1). O algoritmo para quando um determinado número de iterações seja alcançado ou quando os PageRanks convergem.

Dada uma lista de usuários e seus seguidores, seu objetivo é ordenar os usuários por sua importância dentro da rede utilizando uma implementação de PageRank que para depois de um determinado número de iterações.

Entrada

A entrada é composta por um conjunto de linhas. A primeira linha contém um número inteiro P ($0 \leq P \leq 10$) que indica o número de iterações que devem ser feitas para calcular o PageRank. A segunda linha contém um conjunto de N nomes (até 10 caracteres), sem espaço, representando os usuários da rede social ($1 \leq N \leq 50$), separadas por um ou mais espaços. As próximas N linhas contém um conjunto de N valores 0 ou 1, separados por um ou mais espaços, indicando os usuários seguidos pelo usuário correspondente. O valor 1 indica que uma pessoa segue a outra e 0 que uma pessoa não segue a outra.

Saída

A saída deve conter N linhas, onde cada linha contém o nome do usuário e o seu PageRank com 6 casas decimais. As linhas devem estar ordenadas de forma decrescente pelo PageRank do usuário depois de P iterações. No caso de usuários com o mesmo PageRank, a ordem de entrada dos usuários deve ser mantida (quem foi definido antes, aparece primeiro na lista)

Exemplos

Entrada	Saída
3 A B C 0 1 1 0 0 1 1 0 0	C: 1.250000 A: 1.000000 B: 0.750000

Entrada	Saída
2 A B C D 0 1 1 1 1 0 1 0 1 1 0 1 1 1 1 0	A: 1.111111 C: 1.111111 B: 1.000000 D: 0.777778

Entrada	Saída
1 Alice Benedito Carlos Dagoberto Evandro 0 1 1 0 0 1 0 0 1 0 1 0 0 0 0 1 1 0 0 1 1 0 0 1 0	Alice: 2.333333 Dagoberto: 1.000000 Benedito: 0.833333 Carlos: 0.500000 Evandro: 0.333333

Problema J

Torneio de Robótica

Autor: Cássio Agnaldo Onodera (IFSP – Campus Birigui)

Arquivo: `robotica.{c|cpp|cs|java|py}`

Timelimit: 1

Com o sucesso do último TRIF (Torneio de Robótica do Instituto Federal), o professor Ademir resolveu se antecipar e incentivou seus alunos a formarem suas equipes para já iniciarem a construção de seus robôs para a competição do próximo ano.

Cada equipe é formada por um ou mais alunos e cada aluno deve pertencer a exatamente uma equipe. Para que a equipe tenha sucesso, Ademir considera que seja importante que os alunos de uma equipe tenham afinidade entre si. Desta forma, se um aluno possui afinidade com outro aluno, estes alunos devem pertencer à mesma equipe.

Para que as equipes sejam formadas o mais rápido possível, Ademir solicitou que os alunos informassem suas relações de amizade.

Você como excelente programador, finalista da VII INTERIF, foi convocado para auxiliar a formação das equipes. Escreva um programa que receba o número total de alunos e a lista de amizades e informe o total de equipes formadas.

Entrada

A primeira linha contém dois inteiros N ($1 \leq N \leq 10^3$) e M ($1 \leq M \leq 10^4$) que representam o número de alunos e o número de linhas da lista de amizades respectivamente. Os alunos foram numerados de 1 a N . As próximas M linhas contém as relações de amizades indicados por dois inteiros I e J informando que I possui amizade com J e vice-versa ($1 \leq I, J \leq N, I \neq J$).

Saída

A saída é formada por uma única linha contendo um número inteiro Q indicando a quantidade máxima de equipes.

Exemplos

Entrada	Saída
5 3 1 2 4 1 2 5	2

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