



IX - Maratona POP Médio

26 de novembro de 2019

(Este caderno contém 9 problemas)

A PROVA TERÁ DURAÇÃO DE **DUAS HORAS E MEIA**

LEIA ATENTAMENTE AS INSTRUÇÕES ABAIXO ANTES DE INICIAR A PROVA

- A prova deve ser realizada em equipes de duas pessoas;
- Observem o nome do arquivo que deve ser enviado para cada problema;
- Cada questão tem um tempo limite para execução;
- É permitido consultar material impresso durante a prova;
- Não é permitida a consulta de qualquer material online;
- Este caderno de tarefas é composto de 10 páginas (incluindo esta);
- Verifique se o caderno está completo.

Criado por: @thiago.gouveia

Problema A – Antes do Reveillon

Nome do Programa: reveillon.(c|cpp|py|java)

Tempo: 1 segundo

Titico foi o nome que Tithânia e Thiryrica escolheram para seu filho mais novo. Os avós ficaram bastante desapontados pelas marcantes ausências do H e do Y. Mas é como dizem... Eles tiveram seu tempo, agora é o tempo dos jovens.

Revendo os vídeos do último dia 31 de dezembro, Tithânia lembrou que Titico adorou a contagem regressiva antes dos fogos do final do ano. Foi um momento mágico. Ela lembrou também que Titico, por ser muito novinho, esqueceu de falar dois números na hora de contar.

Como o áudio ficou bem ruim por causa do barulho dos fogos, Thiryrica pediu que você escrevesse um programa para recriar a contagem de Titico.

Entrada

A entrada contém três linhas. Na primeira linha, um número inteiro **N** ($3 \leq N \leq 100$) indicando o número de início da contagem. Na segunda linha, o primeiro número que Titico esqueceu de contar. Na terceira linha, o segundo número que Titico esqueceu de contar.

Saída

A saída deve ser a contagem regressiva feita por titico.

Casos de Teste

Entrada 1	Saída 1
5 3 1	5 4 2 0
Entrada 2	Saída 2
7 7 0	6 5 4 3 2 1

Criado por: @thiago.gouveia e @calebe_oliveira

Problema B – Boom!

Nome do Programa: boom.(c|cpp|py|java)

Tempo: 1 segundo

Há um pequeno planeta rochoso localizado nos ermos da Via Láctea. Seu nome: G787884. Aconteceu que no momento da concepção deste problema, seu nome foi G787884. O chefe do departamento de habitação deste planeta, o bom, velho e humilde senhor 9845321 está planejando o novo sistema de casas para os imigrantes de G787884. Chamou esse sistema de Boom!

9999999, o Imperador Celestial, cedeu um terreno montanhoso quadrado para que 9845321 construísse seu novo conjunto. As únicas restrições eram:

1- No centro do conjunto deverá haver um busto em homenagem ao senhor 9999999. 2- Uma vez escolhido o centro, as casas só poderiam ser construídas acima, abaixo, à direita ou à esquerda do centro. 3- Uma casa só pode ser construída em um terreno cuja altura seja múltipla da altura do centro.

Você poderia ajudar o humilde senhor 9845321 a escolher o melhor centro, de modo a construir o maior número de casas no terreno quadrado doado pelo Imperador Celestial?

Entrada

A primeira linha da entrada contém apenas um número inteiro **N** ($1 \leq N \leq 100$) indicando o tamanho da matriz, opz, do terreno quadrado doado por 9999999. Seguem **N** linhas, cada uma com **N** números, representando a altura de cada ponto do terreno.

Saída

A saída deve ser a coordenada **x y** do centro que maximiza o número de casas construídas. Em caso de empate, a menor linha vence. Se o empate persistir, a menor coluna vence.

Casos de Teste

Entrada 1	Saída 1
3 16 13 4 11 10 17 20 17 7	1 3
Entrada 2	Saída 2
5 12 17 7 11 6 8 10 6 13 13 5 13 21 12 8 15 17 18 20 11 10 8 13 7 6	1 5

Criado por: @thiago.gouveia

Problema C – Classificados

Nome do Programa: classificados.(c|cpp|py|java)

Tempo: 1 segundo

No ano de 2019, a iniciativa POP teve muitos inscritos. Além do limite do imaginável, para falar a verdade. Neste caso, foi necessário criar uma regra para classificar ou desclassificar os alunos.

A regra criada foi simples: se o aluno alcançou uma pontuação igual ou maior que a média de todas as pontuações, se classificou. Caso contrário, não se classificou.

Você poderia ajudar o pessoal do POP a calcular o número de alunos que se classificou nesse ano?.

Entrada

A primeira linha da entrada contém um número inteiro **N** ($1 \leq N \leq 1000$) indicando o número de alunos que fez a prova.

Seguem **N** linhas, cada uma com um número inteiro **X** ($0 \leq X \leq 1000$) indicando a nota que cada aluno obteve na prova.

Saída

A saída deve ser o número de usuários classificados, isto é, que obtiveram pontuação maior ou igual à média de todas as pontuações.

Casos de Teste

Entrada 1	Saída 1
4 200 200 400 400	2
Entrada 2	Saída 2
3 300 300 400	1
Entrada 3	Saída 3
3 1 2 3	2

Criado por: @thiago.gouveia e @victor_herberts

Revisado por: @kerven

Problema D – Desenhando Balões

Nome do Programa: baloes.(c|cpp|py|java)

Tempo: 1 segundo

O pequeno Titico, filho mais novo do ilustríssimo casal Thi-ti (Thiryrica e Tithânia) está aprendendo a desenhar. Balões! Balões são as únicas coisas que ele gosta de desenhar. Balões imensos! Balões gigantescos!

Percebendo que Titico ainda não aprendeu a colorir, Thi-ti resolveu adicionar um pouco de cor aos desenhos do pequeno. O único problema é que os balões são grandes demais. Por isso, cada balão gasta um giz de cera completo para ser colorido.

Curioso como todo futuro maratonista de programação, o pequeno Titico começou a se perguntar: dados os balões que já foram coloridos, os balões que ainda não foram coloridos e o conjunto de giz de cera que ainda restam na caixa: de quantas formas diferentes os balões restantes podem ser coloridos? Para titico, a ordem dos balões é importante!

Entrada

A entrada contém duas linhas. A primeira linha traz uma palavra composta apenas por letras maiúsculas e por símbolos '-'. Cada letra maiúscula representa um balão que foi colorido com a letra (cor) em questão, enquanto que cada símbolo '-' representa um balão que ainda não foi colorido.

A segunda linha traz uma palavra composta apenas por letras maiúsculas, indicando os bastões de giz de cera que ainda restam na caixa.

Observe que cada palavra (linha1) tem tamanho máximo 100 e que o número de símbolos '-' é **EXATAMENTE** o número de bastões que ainda resta na caixa menos 1.

Saída

A saída deve ser **X**, o número total de combinações diferentes para completar a coloração dos balões. Como este número pode ser muito grande, imprima-o em módulo **239**, isto é, o resto da divisão de **X** por **239**.

Casos de Teste

Entrada 1	Saída 1
A-B- CCD	3
Entrada 2	Saída 2
A--CC---D AABBEF	180

Criado por: @thiago.gouveia.da.silva

Problema E – Escolha os Classificados

Nome do Programa: escolha.(c|cpp|py|java)

Tempo: 1 segundo

No ano de 2018, a iniciativa POP teve poucos inscritos na categoria SAMSDGM (Super Avançado Master Sênior Destruidor de Galáxias do Mal). Neste caso, foi necessário criar uma regra bem simples para classificar ou desclassificar os candidatos.

A regra criada foi justa: se o aluno alcançou uma pontuação igual ou maior que **300**, se classificou. Caso contrário, não se classificou.

Dada a pontuação de um candidato, você poderia ajudar o pessoal do POP a verificar se o aluno conseguiu se classificar?

Entrada

A entrada contém apenas um número inteiro **N** ($0 \leq N \leq 1000$) indicando a nota obtida na prova.

Saída

A saída deve ser a palavra CLASSIFICADO caso a nota passe do ponto de corte e ELIMINADO caso contrário.

Casos de Teste

Entrada 1	Saída 1
300	CLASSIFICADO
Entrada 2	Saída 2
299	ELIMINADO
Entrada 3	Saída 3
301	CLASSIFICADO

Criado por: @thiago.gouveia e @kerven

Problema F – Festa no Bairro Quadrado

Nome do Programa: festa.(c|cpp|py|java)

Tempo: 2 segundos

A encantadora cidade de *Somsispaia*, no coração da região mais fria de *POPioPÓpolis*, é um destino turístico romântico, tranquilo e muito charmoso. É tudo o que você precisa neste inverno.

Uma das maiores atrações turísticas de *Somsispaia* é o Bairro Quadrado, onde todas as construções são quadradas, sendo que todas as paredes horizontais de todas as casas são paralelas, assim como todas as paredes verticais.

Sen-Barui, um dos moradores mais antigos deste bairro adora o silêncio e a paz. Sabendo que haverá uma festa em uma casa no Bairro Quadrado, *Sen-Barui* gostaria de saber se o som da festa chegará até sua casa.

O problema do Bairro Quadrado é a sua acústica. Por conta do modelo arquitetônico, o som se espalha facilmente entre as construções. Digamos que a festa será na casa 'A'. Se a distância entre a casa 'A' e a casa 'B' for no máximo k , o som se espalha de 'A' para 'B'. Neste caso, O som pode se espalhar também para qualquer casa que esteja a no máximo k de distância de 'B'. E assim sucessivamente.

Por fim, observe que uma casa pode estar dentro de outra, ou até mesmo começar dentro e terminar fora de outra casa.

Entrada

A primeira linha da entrada contém quatro inteiros **N a b k** ($1 \leq N, a, b, k \leq 100$) indicando, respectivamente, o número de quadrados, o quadrado onde acontece a festa, o quadrado onde *Sen-Barui* mora, e a distância máxima que o som se propaga de uma casa para outra.

Seguem **N** linhas, cada uma com os inteiros **id x1 y1 x2 y2** (todos no intervalo **[1, 200]**), representando respectivamente o ID do quadrado, a coordenada esquerda do quadrado, a coordenada inferior do quadrado, a coordenada direita do quadrado e a coordenada superior do quadrado.

Saída

A saída deve ser a palavra YEAP, caso seja possível ouvir o som da festa que acontece no quadrado a no quadrado b. Caso contrário, imprima NOPZ.

Casos de Teste

Entrada 1	Saída 1
5 1 5 2 1 1 7 3 9 2 1 4 3 6 3 9 3 11 5 4 1 2 3 4 5 7 5 8 6	NOPZ

Criado por: @thiago.gouveia e @kerven

Problema G – Grade do Caos

Nome do Programa: grade.(c|cpp|py|java)

Tempo: 2 segundos

O Projeto Olímpico de Programação (POP) do IFPB, Campus João Pessoa, está dividido em várias categorias. Temos o POP Fundamental, o POP Médio, O POP Médio Avançado, o POP Superior e o POP Superior Avançado.

Por incrível que pareça, até os dias de hoje, a grande maioria dos estudantes do POP são do curso de Engenharia Elétrica. Isso nos fez lembrar um desafio proposto por um professor da disciplina de Eletricidade: A Grade do Caos.

Ele montou uma grade assimétrica (as alturas das linhas e colunas são diferentes) com arames, colocou eletricidade na ponta inferior esquerda da grade e uma lâmpada na ponta superior direita. Logicamente, a lâmpada acendeu. Então o professor diminuiu um pouco a intensidade da corrente elétrica, apagando a luz. Contudo, o professor falou que se a menor distância entre a origem da eletricidade e a lâmpada fosse no máximo **D**, a lâmpada acenderia.

Com um sorriso sapeca o professor olhou a turma e soltou: qual o menor número de diagonais que devem ser adicionadas à grade de modo que a distância da origem à lâmpada seja menor que **D**? Uma diagonal é um fio que sai de uma junção linha-coluna e vai até uma junção vizinha, sem cruzar nenhum arame no caminho.

Entrada

A primeira linha da entrada traz dois inteiros **N** ($1 \leq N \leq 300$) e **D**, indicando o tamanho da grade (número de linhas e colunas) e a distância alvo.

A linha seguinte traz **N** inteiros, indicando a altura de cada linha da grade, de cima para baixo. A próxima linha traz **N** inteiros, indicando a largura de cada coluna da grade, da esquerda para a direita.

Saída

A saída deve ser o menor número de diagonais que deve ser adicionada à grade de modo que a distância entre o ponto inferior esquerdo e o ponto superior direito seja no máximo **D**. Caso não seja possível, o programa deve imprimir SEM CHANCE.

Casos de Teste

Entrada 1	Saída 1
4 14 1 2 3 4 3 2	1
Entrada 2	Saída 2
4 10 1 2 3 4 3 2	SEM CHANCE

Criado por: @thiago.gouveia e @calebe_oliveira

Revisado por: @kerven

Problema H – Hematoleogenese

Nome do Programa: hemato.(c|cpp|py|java)

Tempo: 2 segundos

O litoral do Brasil passou (passa?) por uma crise ecológica sem precedentes. Algum tipo de vazamento ou derramamento de óleo tem assolado várias praias, causando danos ao ecossistema como um todo, passando pelos animais, pela flora marinha e até pelos recifes de corais.

Até o momento não se descobriu a causa desse vazamento (derramamento?). Como não é possível parar o vazamento, os cientistas voltaram seus esforços para a descontaminação das áreas afetadas. Nesta linha, desenvolveram a Hematoleogênese, um novo descontaminante genético-biológico.

Uma vez aplicada a Hematoleogênese, esta leva um dia para fazer efeito na área de aplicação. No próximo dia, a Hematoleogênese se expande e destrói todo o óleo das áreas vizinhas. No dia seguinte, a Hematoleogênese se expande a partir de cada área que foi descontaminada, e descontamina todas as áreas vizinhas a esta ... e assim por diante até que não haja mais nenhuma área contaminada.

Como o processo é caro, os cientistas conseguiram apenas **3** descontaminantes (*crowdfunding* \o/). Eles agora pedem a sua ajuda para saber em que áreas aplicar os **3** descontaminantes de modo que todo o litoral seja descontaminado no menor número de dias possível.

Entrada

A primeira linha da entrada contém dois números inteiros **N** ($1 \leq N \leq 100$) e **M** ($0 \leq M \leq N \cdot N$). **N** indica o número de áreas contaminadas, enquanto **M** indica o número de ligações (pela correnteza) entre estes locais. Seguem **M** linhas, cada uma com dois inteiros **a** e **b** ($1 \leq a, b \leq N$) indicando que a correnteza pode levar o descontaminante de **a** para **b**, assim como pode levar o descontaminante de **b** para **a** (enchendo: $a \rightarrow b$, secando: $b \rightarrow a$).

Saída

A saída deve ser o menor número de dias necessário para descontaminar todo o litoral (todas as áreas apresentadas). Caso não seja possível, o programa deve imprimir FALHA.

Casos de Teste

Entrada 1	Saída 1
4 2 1 2 2 3	2
Entrada 2	Saída 2
4 0	FALHA

Criado por: @thiago.gouveia

Problema I – Imagina Isso numa Caixa

Nome do Programa: imagina.(c|cpp|py|java)

Tempo: 1 segundo

Paul OfTarso, um romântico incorrigível, estava passeando por Veneza (antes do alagamento, claro) quando viu um belíssimo item especial. Não pôde se conter e terminou por adquiri-lo como presente para sua amada Val WoudRead.

Após vislumbrar sua mais nova aquisição, nosso “*principeso*” pegou-se a refletir: Imagina isso numa Caixa!

Rumou para a única loja de caixas de Veneza que estava à altura de tanto amor e começou a procurar por alguma caixa que coubesse o presente.

Entrada

A primeira linha da entrada contém três números inteiros **pa**, **pl** e **pp** ($0 \leq pa, pl, pp \leq 1000$), representando a altura, a largura e a profundidade do presente.

A segunda linha da entrada contém um número inteiro **N** ($1 \leq N \leq 100$) indicando o número de caixas que existem na loja.

Seguem **N** linhas, cada uma com um três números inteiros cada: **ca**, **cl** e **cp** ($0 \leq ca, cl, cp \leq 100$), representando a altura, a largura e a profundidade de uma caixa.

Saída

Seu programa deve imprimir SIM, caso o presente caiba em alguma caixa e NAO caso contrário. Note que Paul é esperto, podendo girar o presente para que ele caiba na caixa.

Casos de Teste

Entrada 1	Saída 1
4 5 6 3 7 6 3 4 5 5 7 5 6	SIM
Entrada 2	Saída 2
5 3 4 3 10 1 2 20 9 2 3 4 4	NAO