



II Maratona de Programação UCDB

Nível Médio

27 de Maio de 2019

Este caderno contém 10 problemas; as páginas estão numeradas de 1 a 11.

Informações Gerais

Estas condições são válidas para todos os problemas, exceto quando explicitamente dito.

Nome do programa

1. Sua solução deve ser chamada *problema.c*, *problema.cpp* ou *problema.java*; onde *problema* é a letra maiúscula que identifica o problema.

Entrada

1. A entrada de seu programa deve ser lida da entrada padrão.
2. A entrada consiste de um único caso de teste, que é descrito com uma quantidade de linhas definida no problema. Não há outras informações na entrada.
3. Quando uma linha contém diversos valores, eles são separados por *um único espaço*. Não há outros espaços na entrada. Não há linhas vazias.
4. Apenas o alfabeto em Inglês é utilizado. Não há caracteres com til, acento, trema ou outros símbolos (ñ, Ã, é, Ì, ô, Ü, ç, etcetera).
5. Todas as linhas, incluindo a última, contém o tradicional caractere de quebra-de-linha.

Saída

1. A saída de seu programa deve ser escrita na saída padrão.
2. O resultado de um caso de teste deve ser escrito com uma quantidade de linhas definida no problema. Não deve haver outras informações na saída.
3. Quando uma linha contém diversos valores, eles devem ser separados por *um único espaço*. Não deve haver outros espaços na entrada. Não deve haver linhas vazias.
4. Apenas o alfabeto em Inglês deve ser utilizado. Não deve haver caracteres com til, acento, trema ou outros símbolos (ñ, Ã, é, Ì, ô, Ü, ç, etcetera).
5. Todas as linhas, incluindo a última, devem conter o tradicional caractere de quebra-de-linha.
6. Para resultados de números reais, arredonde para o número racional mais próximo com a quantidade equivalente de dígitos de precisão. O caso de teste é feito de modo que não haja ambiguidade neste tipo de arredondamento.

Problema A - Lendo Livros

Você começou a competir com seu amigo para ver quem consegue ler mais livros em menos tempo. Seu amigo lia muito mais que você, até o dia que você percebeu que ele lia somente livros muito finos.

Então você resolveu contar as páginas dos livros, aumentando também a quantidade de páginas lidas por dia. Agora você lê 5 páginas por dia e termina 16 dias antes do que se estivesse lendo 3 páginas por dia. Neste cenário, quantas páginas tem o livro?

Entrada

A entrada é composta de vários casos de testes. Cada caso de teste é composto de três números Q ($0 < Q < 20$), D ($0 < D < 20$) e P ($0 < P < 20$) separados por um espaço. Sendo que Q é a quantidade de páginas lidas por dia. D é o número de dias que você adiantaria a leitura caso estivesse lendo a quantidade de páginas informada pelo número P . Um único valor zero indica o fim da entrada.

Saída

Para cada caso de teste deverá ser impresso a quantidade de páginas do livro. (Utilize o plural corretamente e não use acentos). Este número deverá ser um inteiro, o qual representa a quantidade de página. Este valor deverá ser truncado caso necessário.

Entrada 1	Saída 1
3 16 5 0	120 paginas

Problema B - Cartas

Beatriz gosta muito de jogar cartas com as amigas. Para treinar memória e raciocínio lógico, ela inventou um pequeno passatempo com cartas. Ela retira as cinco primeiras cartas do topo de um baralho bem embaralhado, e as coloca em sequência, da esquerda para a direita, na mesa, com as faces voltadas para baixo.

Então ela olha, por um breve instante, cada uma das cartas da sequência (e logo as recoloca na mesa, com a face para baixo). Usando apenas a sua memória, Beatriz deve agora dizer se a sequência de cartas está ordenada crescentemente, decrescentemente, ou não está ordenada.

De tanto jogar, ela está ficando cansada, e não confia em seu próprio julgamento para saber se acertou ou errou. Por isso, ela pediu para você fazer um programa que, dada uma sequência de cinco cartas, determine se a sequência dada está ordenada crescentemente, decrescentemente, ou não está ordenada.

Entrada

A entrada consiste de uma única linha que contém as cinco cartas da sequência. Os valores das cartas são representados por inteiros entre 1 e 13. As cinco cartas têm valores distintos.

Saída

Seu programa deve produzir uma única linha, contendo um único caractere maiúsculo: ‘C’ caso a sequência dada esteja ordenada crescentemente, ‘D’ se estiver ordenada decrescentemente, ou ‘N’ caso contrário.

Entrada 1	Saída 1
1 2 3 5 6	C

Problema C - Diamantes e Areia

João está trabalhando em uma mina, tentando retirar o máximo que consegue de diamantes «>». Ele deve excluir todas as partículas de areia "." do processo e a cada retirada de diamante, novos diamantes poderão se formar. Se ele tem como uma entrada .<...«..»....>....»>., três diamantes são formados. O primeiro é retirado de <..>, resultando .<...<>....>....»>. Em seguida o segundo diamante é retirado, restando .<.....>....»>. O terceiro diamante é então retirado, restando no final»>., sem possibilidade de extração de novo diamante.

Entrada

Deve ser lido um valor inteiro N que representa a quantidade de casos de teste. Cada linha a seguir é um caso de teste que contém até 1000 caracteres, incluindo "<,>,".

Saída

Você deve imprimir a quantidade de diamantes possíveis de serem extraídos em cada caso de entrada.

Entrada 1	Saída 1
2	3
<..><..>>	1
<<<..<.....<<<<....>	

Problema D - Frequência de Números

Neste problema sua tarefa será ler vários números e em seguida dizer quantas vezes cada número aparece na entrada de dados, ou seja, deve-se escrever cada um dos valores distintos que aparecem na entrada por ordem crescente de valor.

Entrada

A entrada contém apenas 1 caso de teste. A primeira linha de entrada contém um único inteiro N , que indica a quantidade de valores que serão lidos para X ($1 \leq X \leq 2000$) logo em seguida. Com certeza cada número não aparecerá mais do que 20 vezes na entrada de dados.

Saída

Imprima a saída de acordo com o exemplo fornecido abaixo, indicando quantas vezes cada um deles aparece na entrada por ordem crescente de valor.

Entrada 1	Saída 1
7	4 aparece 1 vez(es)
8	8 aparece 2 vez(es)
10	10 aparece 3 vez(es)
8	260 aparece 1 vez(es)
260	
4	
10	
10	

Problema E - Número Perfeito

Na matemática, um número perfeito é um número inteiro para o qual a soma de todos os seus divisores positivos próprios (excluindo ele mesmo) é igual ao próprio número. Por exemplo o número 6 é perfeito, pois $1+2+3$ é igual a 6. Sua tarefa é escrever um programa que imprima se um determinado número é perfeito ou não.

Entrada

A entrada contém vários casos de teste. A primeira linha da entrada contém um inteiro N ($1 \leq N \leq 20$), indicando o número de casos de teste da entrada. Cada uma das N linhas seguintes contém um valor inteiro X ($1 \leq X \leq 10^8$), que pode ser ou não, um número perfeito.

A entrada contém vários casos de teste. A primeira linha de um caso de teste contém um número inteiro N indicando o número de questões da folha de respostas ($1 \leq N \leq 255$). Cada uma das N linhas seguintes descreve a resposta a uma questão e contém cinco números inteiros A , B , C , D e E , indicando os valores de nível de cinza médio para cada uma das alternativas da resposta ($0 \leq A, B, C, D, E \leq 255$).

Saída

Para cada caso de teste de entrada, imprima a mensagem “ X eh perfeito” ou “ X nao eh perfeito”, de acordo com a especificação fornecida.

Entrada 1	Saída 1
3	6 eh perfeito
6	5 nao eh perfeito
5	28 eh perfeito
28	

Problema F - Nove

Paulo Bruno é um menino que adora Pokemons, porém odeia matemática, ele detesta exponenciação e por algum motivo não calcula corretamente operações que envolvam o número 9. Sabendo disso, seu amigo Werlesson decidiu fazer um desafio, ele quer que Paulo Bruno calcule a N -ésima potência de 9 e diga o último dígito dessa potência. Por exemplo, sendo $N=2$, o resultado seria 1, pois $9^2=81$. O problema é que, dependendo do valor de N , o resultado da exponenciação pode ser um número muito grande. Sem ideias, Paulo Bruno decidiu pedir sua ajuda.

Entrada

A entrada é composta por diversas instâncias. A primeira linha da entrada contém um inteiro T indicando o número de instâncias. Cada instância é composta por apenas uma linha, que contém o número inteiro N . ($0 \leq N \leq 10^9$).

Saída

Para cada instância na entrada, imprima uma linha contendo um inteiro, o último dígito de 9^N .

Entrada 1	Saída 1
2	9
1	1
2	

Problema G - Leitura Ótica

O professor João decidiu aplicar somente provas de múltipla escolha, para facilitar a correção. Em cada prova, cada questão terá cinco alternativas (A, B, C, D e E), e o professor vai distribuir uma folha de resposta para cada aluno. Ao final da prova, as folhas de resposta serão escaneadas e processadas digitalmente para se obter a nota de cada aluno. Inicialmente, ele pediu ajuda a um sobrinho, que sabe programar muito bem, para escrever um programa para extrair as alternativas marcadas pelos alunos nas folhas de resposta. O sobrinho escreveu uma boa parte do software, mas não pode terminá-lo, pois precisava treinar para a Maratona de Programação.

Durante o processamento, a prova é escaneada usando tons de cinza entre 0 (preto total) e 255 (branco total). Após detectar os cinco retângulos correspondentes a cada uma das alternativas, ele calcula a média dos tons de cinza de cada pixel, retornando um valor inteiro correspondente àquela alternativa. Se o quadrado foi preenchido corretamente o valor da média é zero (preto total). Se o quadrado foi deixado em branco o valor da média é 255 (branco total). Assim, idealmente, se os valores de cada quadrado de uma questão são (255, 0, 255, 255, 255), sabemos que o aluno marcou a alternativa B para essa questão. No entanto, como as folhas são processadas individualmente, o valor médio de nível de cinza para o quadrado totalmente preenchido não é necessariamente 0 (pode ser maior); da mesma forma, o valor para o quadrado não preenchido não é necessariamente 255 (pode ser menor). O prof. João determinou que os quadrados seriam divididos em duas classes: aqueles com média menor ou igual a 127 serão considerados pretos e aqueles com média maior a 127 serão considerados brancos.

Obviamente, nem todas as questões das folhas de resposta são marcadas de maneira correta. Pode acontecer de um aluno se enganar e marcar mais de uma alternativa na mesma questão, ou não marcar nenhuma alternativa. Nesses casos, a resposta deve ser desconsiderada.

O professor João necessita agora de um voluntário para escrever um programa que, dados os valores dos cinco retângulos correspondentes às alternativas de uma questão determine qual a alternativa corretamente marcada, ou se a resposta à questão deve ser desconsiderada.

Entrada

A entrada contém vários casos de teste. A primeira linha de um caso de teste contém um número inteiro N indicando o número de questões da folha de respostas ($1 \leq N \leq 255$). Cada uma das N linhas seguintes descreve a resposta a uma questão e contém cinco números inteiros A, B, C, D e E , indicando os valores de nível de cinza médio para cada uma das alternativas da resposta ($0 \leq A, B, C, D, E \leq 255$).

O ultimo caso de teste é seguido por uma linha que contém apenas um número zero.

Saída

Para cada caso de teste da entrada seu programa deve imprimir N linhas, cada linha correspondendo a uma questão. Se a resposta à questão foi corretamente preenchida na folha de resposta, a linha deve conter a alternativa marcada ('A', 'B', 'C', 'D' ou 'E'). Caso contrário, a linha deve conter o caractere '*' (asterisco).

Entrada 1	Saída 1
3	A
0 255 255 255 255	E
255 255 255 255 0	C
255 255 127 255 255	D
4	*
200 200 200 0 200	*
200 1 200 200 1	B
1 2 3 4 5	
255 5 200 130 205	
0	

Problema H - Batalha de Pomekons

Depois de capturar muitos Pomekons, Dabriel e Guarte resolveram batalhar. A forma de duelo é simples, cada treinador coloca um Pomekon na batalha e vence quem tem o Pomekon com maior valor de golpe, que é definido da seguinte maneira:

$$\text{ValorGolpe} = \text{Bonus} + (\text{Ataque} + \text{Defesa})/2$$

O Bônus será dado ao Pomekon do treinador que estiver em um level de valor par.

Neste problema será dado a você o valor do bônus aplicado, os valores de ataque e defesa do Pomekon de Dabriel e Guarte e seus respectivos níveis, cabe a você informar o ganhador da batalha.

Entrada

A entrada é composta por diversas instâncias. A primeira linha da entrada contém um inteiro T indicando o número de instâncias. Cada instância começa com um inteiro B ($0 \leq B \leq 100$), que indica o valor do bônus aplicado. Nas duas linhas seguintes terão três inteiros A_i , D_i e L_i ($1 \leq A_i, D_i \leq 100$, $1 \leq L_i \leq 50$), representando o valor de ataque do Pomekon, o valor de defesa e o level do treinador. A primeira linha representa o Pomekon de Dabriel e a segunda o de Guarte.

Saída

Para instância na entrada você deverá imprimir o nome do treinador que irá vencer a batalha, em caso de empate imprima: "Empate", sem aspas.

Entrada 1	Saída 1
3	Guarte
5	Empate
12 23 15	Dabriel
42 12 20	
2	
52 1 11	
1 52 1	
3	
95 12 22	
5 51 21	

Problema I - Número Primo

Na matemática, um Número Primo é aquele que pode ser dividido somente por 1 (um) e por ele mesmo. Por exemplo, o número 7 é primo, pois pode ser dividido apenas pelo número 1 e pelo número 7.

Entrada

A entrada contém vários casos de teste. A primeira linha da entrada contém um inteiro N ($1 \leq N \leq 100$), indicando o número de casos de teste da entrada. Cada uma das N linhas seguintes contém um valor inteiro X ($1 < X \leq 10^7$), que pode ser ou não, um número primo.

Saída

Para cada caso de teste de entrada, imprima a mensagem “ X eh primo” ou “ X nao eh primo”, de acordo com a especificação fornecida.

Entrada 1	Saída 1
3	8 nao eh primo
8	51 nao eh primo
51	7 eh primo
7	

Problema J - Indecisão das Renas

Esse ano as Renas do papai Noel decidiram que Rudolph não seria mais aquele que sempre ficaria à frente. Elas escolheriam de forma justa entre elas quem iria encabeçar o trenó. E nada é mais justo que o acaso.

Então optaram pela seguinte forma para escolher: Cada Rena faria a quantidade que quisesse de bolas de neve, sem as outras verem. Depois, todas as bolas de neve de todas as Renas seriam reunidas em uma única e grande pilha. Por último, as bolas de neve seriam tiradas dessa pilha, uma a uma, e distribuídas entre elas sempre seguindo a ordem: Dasher, Dancer, Prancer, Vixen, Comet, Cupid, Donner, Blitzen e Rudolph. Até que se acabassem as bolas de neve. A rena que ficasse com a última bola de neve seria declarada vencedora e ficaria na posição principal do trenó este ano.

Dado o número de bolas de neve feitas por cada Rena, determine qual Rena ganhou o sorteio.

Entrada

A entrada é composta por uma única linha contendo 9 números inteiros A_i ($1 \leq A_i \leq 10^4$).

Saída

A saída é composta por uma única linha contendo o nome da Rena vencedora.

Entrada 1 1 2 3 4 5 6 7 8 9	Saída 1 Rudolph
Entrada 2 9 9 9 9 9 9 9 9 7	Saída 2 Donner
Entrada 3 1 2 1 2 1 2 1 2 1	Saída 3 Vixen