

Universidade de Brasília

Seletiva da Maratona de Programação UnB/CIC 2016

22 de outubro de 2016

Coordenação:

Prof. Edson Alves da Costa Júnior (UnB/FGA)

Prof. Guilherme Novaes Ramos (UnB/CIC)

A) Sobre a entrada

1. A entrada de seu programa deve ser lida da *entrada padrão*.
2. Quando uma linha da entrada contém vários valores, estes são separados por um único espaço em branco; a entrada não contém nenhum outro espaço em branco.
3. Cada linha, incluindo a última, contém o caractere final-de-linha.
4. Quando não indicada outra forma, o final da entrada coincide com o final do arquivo.

B) Sobre a saída

1. A saída de seu programa deve ser escrita na *saída padrão*.
2. Quando uma linha da saída contém vários valores, estes devem ser separados por um único espaço em branco; a saída não deve conter nenhum outro espaço em branco.
3. Cada linha, incluindo a última, deve conter o caractere final-de-linha.

C) Sobre os problemas

As situações retratadas nos problemas são inteiramente fictícias e não correspondem à realidade. Nada escrito nos enunciados tem a intenção de desrespeitar o leitor. Tudo foi escrito de maneira a se adequar às situações hipotéticas da melhor maneira possível.

A Música a é qual?

Limite de Tempo: 3s

Fulano gosta muito de cantar, mas já está entediado de cantar sempre as mesmas músicas, então ele teve a ideia de cantar a letra ao contrário da seguinte forma:

1. para cada frase da música a primeira palavra se torna a última, a segunda se torna a penúltima e assim por diante;
2. as vírgulas também seguem a primeira regra, mas devem respeitar o espaçamento da língua portuguesa;
3. somente a primeira letra de cada frase deve ser maiúscula.

Por exemplo, “Vem comigo, amor, eu vou levar pra passear” vira “Passear pra levar vou eu, amor, comigo vem”.

Sua tarefa é ajudá-lo reescrevendo as letras conforme as regras apresentadas.

Entrada

A entrada consiste de várias linhas compostas por palavras de, no máximo, 20 caracteres alfabéticos, maiúsculos e minúsculos, e os caracteres , e '. Cada linha conta como uma frase, e cada frase contém, no máximo, 20 palavras, separadas por um único espaço em branco. A entrada não excede o limite de 1000 frases.

Saída

Para cada frase deve ser impressa a frase que será cantada por Fulano, segundo as regras apresentadas.

Exemplos de entradas	Exemplos de saídas
Hey now you're an all star	Star all an you're now hey
Get your game on, go play	Play go, on game your get
Hey now you're a rock star	Star rock a you're now hey
Get the show on, get paid	Paid get, on show the get

B Batalha Naval

Limite de Tempo: 3s

Fulano e Beltrano estão jogando batalha naval. Como já jogam há bastante tempo, decidiram dificultar o jogo e passaram a jogar usando somente submarinos, que ocupam somente uma posição do tabuleiro. Decidiram também jogar em um tabuleiro maior, com as coordenadas indo desde $(-10^9, -10^9)$ até $(10^9, 10^9)$.

Depois que eles posicionaram suas embarcações, Ciclano, o juiz, teve uma ideia para dificultar um pouco mais o jogo:

1. cada jogador deve escrever todas as possíveis coordenadas X em cartões tipo A e escrever todas as possíveis coordenadas Y em cartões tipo B . Assim, cada jogador terá dois decks de cartões, um A e um B ;
2. para jogar uma bomba, o jogador deve escolher um cartão tipo A de seu deck e um cartão tipo B de seu deck, jogando a bomba no ponto (a, b) , sendo a o valor escrito no cartão tipo A e b o valor escrito no cartão tipo B ;
3. logo após jogar uma bomba, o jogador descarta os cartões que usou, não podendo usá-los novamente.

Agora, nem sempre é possível acertar todas as embarcações. Fulano, ao analisar seu campo, ficou curioso com a quantidade máxima de submarinos que Beltrano pode afundar. Sabendo que você sabe programar, pediu sua ajuda pelo celular.

Entrada

A entrada é composta por vários casos de teste. A primeira linha de cada caso de teste contém a quantidade n ($2 \leq n \leq 10^5$) de submarinos que ele tem. Cada uma das próximas n linhas contém dois inteiros x_i e y_i ($|x_i|, |y_i| \leq 10^9$) descrevendo a coordenada do i -ésimo submarino.

Saída

Para cada caso de teste você deve imprimir uma linha com um único inteiro: a quantidade máxima de submarinos que Beltrano pode afundar.

Exemplos de entradas	Exemplos de saídas
4	3
1 2	
2 3	
1 3	
3 4	

C Contando Triângulos

Limite de Tempo: 3s

Dado um conjunto de pontos no plano 2D, conte a quantidade X de triângulos diferentes formados a partir de qualquer 3 pontos do conjunto.

Define-se um triângulo como um conjunto de três pontos distintos não colineares (não existe nenhuma reta que passe por estes três pontos). Dois triângulos são considerados distintos se os seus conjuntos são distintos. Observe que em conjuntos a ordem não importa, logo $\{(0, 0), (1, 1), (2, 3)\}$ e $\{(1, 1), (0, 0), (2, 3)\}$ representam o mesmo triângulo.

Entrada

O input consiste de T ($1 \leq T \leq 100$) casos de testes. A primeira linha de cada caso de teste consiste de um inteiro N ($3 \leq N \leq 1000$), que indica a quantidade de pontos. As N linhas seguintes contém as coordenadas dos pontos, compostas por dois inteiros X_i e Y_i ($0 \leq X_i, Y_i \leq 10^9$). É garantido que cada ponto é único.

Saída

Para cada caso de teste imprima, em uma linha, a quantidade de triângulos possíveis, seguida de uma quebra de linha.

Exemplos de entradas	Exemplos de saídas
3	0
3	1
0 0	4
1 1	
2 2	
3	
0 0	
1 1	
2 3	
4	
0 0	
2 2	
2 0	
3 1	

D Fast Fourier Transform

Limite de Tempo: 1s

Ciclano, um estudante de Ciência de Computação, acabou de aprender uma matéria muito interessante, Transformada Rápida de Fourier (*Fast Fourier Transform* - FFT). Quando ele contou isso a seu amigo Matheus, esse lhe apresentou uma aplicação interessantíssima, porém não tão usual, da transformada: multiplicação de polinômios.

Seja $A(x) = a_0 + a_1x_1 + a_2x_2 + \dots + a_nx_n$, para a_i pertencente aos reais. A FFT permite transformar $A(x)$ para a **forma ponto valor**, isto é ela permite descrever $A(x)$ através dos pontos $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$, tais que $y_i = A(x_i)$ e $x_i \neq x_j$ se $i \neq j$. Além disso, na Transformada de Fourier, os pontos x_i são fixos para representações com mesmo número de pontos, isto é, se dois polinômios A e B são, cada um, representados por k pontos, então $x_i^A = x_i^B$ ($1 \leq i \leq k$).

Ou seja, por exemplo, para $A(x) = x$ e $B(x) = x^2$, uma possibilidade para a **forma ponto valor** é $A(x) = \{(-2, -2), (0, 0), (2, 2)\}$ e $B(x) = \{(-2, 4), (0, 0), (2, 4)\}$ (note que os valores de x são iguais para ambos os polinômios). É possível provar que se há pelo menos $n + 1$ pontos na **forma ponto valor**, sendo n o grau do polinômio, essa forma descreve unicamente $A(x)$, devido à interpolação de polinômios.

Interessado, Ciclano decidiu contar isso à sua irmã mais nova Beltrana e propôs um desafio: dado dois polinômios $A(x)$ e $B(x)$ em **forma ponto valor**, ele quer que ela lhe diga qual a **forma ponto valor** de $C(x) = A(x)B(x)$ (C é igual à A vezes B). Por exemplo, se $A(x) = x$ e $B(x) = 1 + x^3$, $C(x) = A(x)B(x) = x(1 + x^3) = x + x^4$. Infelizmente Beltrana não entendeu uma palavra do que o seu irmão disse. Ajude-a a resolver a questão e vencer o desafio.

Entrada

A primeira linha da entrada contém um inteiro T ($1 \leq T \leq 10^2$) que indica o número de casos de teste. A primeira linha de cada caso de teste contém um inteiro N ($1 \leq N < 200000$, $N = 2k$, onde k é um inteiro positivo), que representa o dobro do grau dos polinômios A e B . A segunda linha contém N inteiros y_i^A ($1 \leq i \leq N$), onde $y_i^A = A(x_i)$ e (x_i, y_i^A) pertence à **forma ponto valor** de A . De forma semelhante, a linha seguinte contém N inteiros y_i^B ($1 \leq i \leq N$), onde $y_i^B = B(x_i)$ e (x_i, y_i^B) pertence à **forma ponto valor** de B . Pode-se assumir que $1 \leq y_i^A, y_i^B \leq 10^9$ e que $x_1 < x_2 < x_3 < \dots < x_{n-1} < x_n$.

Saída

Para cada caso de teste, a saída deve ser uma linha com N números y_i^C , onde $y_i^C = C(x_i)$ e (x_i, y_i^C) pertence à **forma ponto valor** de C . Deve-se imprimir C em ordem estritamente crescente de x_i .

Exemplos de entradas	Exemplos de saídas
1	0 1
2	
1 1	
0 1	

E Maranha, o Ameba

Limite de Tempo: 1s

Uma garrafa de cloro na mão esquerda, uma caixa de nuggets na mão direita e um mendigo ao lado: todos que conhecem Maranha sabem que esse é seu estilo. Maranha é um garoto muito especial que adora programar em Zava enquanto se aventura pelo mundo da internet, especialmente no sítio Reddit. Maranha é um garoto muito procrastinador, condição esta que não o deixa programar em Zava da maneira que gostaria e que lhe rendeu o apelido de Ameba.

Triste por não conseguir alcançar seus objetivos devido à dita condição, Maranha decidiu procurar na internet formas de ter mais força de vontade e energia ao longo do dia; foi, então, que encontrou o sítio do famosíssimo astrólogo, psicólogo e filantropo Pedrinho ‘Uchiha’ de Lyra. Neste sítio, é possível encontrar todos os tipos de dicas imagináveis (dadas pelo próprio doutor Pedrinho), variando desde como comer três pratos de arroz com frango sem desmaiar até como fechar a passagem de outros carros no estacionamento.

Navegando pelo sítio, Maranha encontrou a dica que poderia mudar sua vida: “Flexões: um novo método para vencer a procrastinação”. Na página da dica, Pedrinho explica passo a passo o porquê do método funcionar, mas Maranha, como um bom procrastinador preguiçoso, pula logo para a descrição do método. A descrição dada por doutor Pedrinho é a seguinte: “A realização do método é feita durante N dias, tendo cada dia i um valor a_i associado. No i -ésimo dia, o praticante deve realizar $a_1 + a_2 + \dots + a_i$ flexões; ou seja, o praticante deve realizar o número de flexões atribuídas àquele dia e a soma do número de flexões de todos os dias anteriores àquele (...).”

Depois de ler a dica e pensar um pouco, Maranha percebeu que o número total de flexões que teria que fazer caso realizasse tal método seria demasiadamente grande. Apesar de querer mudar seu estilo de vida, Maranha é preguiçoso por natureza; por esse motivo, ele quer reorganizar a sequência sugerida por doutor Pedrinho de tal forma que o número total de flexões feitas seja o menor possível. Maranha pode rearranjar os números da sequência da forma que quiser. Por exemplo: se a sequência dada for $\{4, 3, 7, 4\}$, o número total de flexões será 43; mas alguns elementos podem ser trocados de posição para obter a sequência $\{3, 4, 4, 7\}$, que tem 39 como número total de flexões.

Maranha, como o bom preguiçoso que é, pediu sua ajuda para encontrar a sequência que minimiza o número total de flexões a serem feitas. Ajude-o!

Entrada

A primeira linha contém um inteiro T ($1 \leq T \leq 100$), que representa a quantidade de casos de testes. A primeira linha de cada caso de teste contém um inteiro N ($1 \leq N \leq 20$), o qual representa a quantidade de dias para realização do método de doutor Pedrinho. A segunda linha de cada caso de teste contém N inteiros a_i ($1 \leq a_i \leq 10^9$), que correspondem aos valores descritos no método acima.

Saída

A saída é composta por T linhas, uma para cada caso de teste. A i -ésima linha deve conter a sequência que minimiza o número total de flexões do i -ésimo caso de teste. Os inteiros de uma sequência devem ser separados por um espaço, não havendo um espaço após o último inteiro da sequência.

Exemplos de entradas	Exemplos de saídas
2	3 8
2	2 3 9 10
8 3	
4	
9 10 2 3	

F Múltiplo Perfeito

Limite de Tempo: 1s

Dados dois inteiros positivos A e B ($A < B$), encontre o menor inteiro X que seja múltiplo de A , seja um quadrado perfeito e seja maior que B . Em outras palavras, encontre o menor inteiro X tal que $X = kA = Y^2 > B$, para k e Y inteiros.

Observação: um quadrado perfeito é um inteiro Q que tem a forma $Q = F^2 = F \cdot F$, para algum F inteiro.

Entrada

A primeira linha do input é composta por um inteiro T ($1 \leq T \leq 100$), o qual representa a quantidade de casos de testes. As próximas T linhas são compostas pelos dois inteiros A ($1 \leq A \leq 10^9$) e B ($2 \leq B \leq 10^{15}, B > A$), respectivamente.

Saída

Para cada um dos T testes, imprima uma linha contendo o inteiro X , como descrito acima. Assuma que o valor de X é sempre menor do que 2^{63} .

Exemplos de entradas	Exemplos de saídas
3	4
1 2	49
7 8	443389415830596
379022652 384108813026771	

G O Bom, O Mau e Os Mortos

Limite de Tempo: 1s

Gerações descendentes da família do vilão Darth Vader alternam entre O Bem e O Mal. Estamos interessados no guerreiro mais forte de sua geração. Considere por exemplo que Darth Vader seja o mais forte de sua geração, que é Do Mal. Seu filho Luke, Do Bem, o mais forte da segunda geração. Kylo, neto de Vader, é o mais forte da terceira geração, Do Mal, e assim sucessivamente. Cada geração possui o seu indivíduo mais forte.



Após N gerações de paz, O Bem escolherá o seu guerreiro mais forte entre as gerações Do Bem para lutar com o guerreiro mais forte entre as gerações Do Mal. Dadas as forças do guerreiro mais forte de cada uma das N gerações, determine a força de cada um destes dois guerreiros que irão duelar.

Entrada

A primeira linha de entrada é um inteiro T ($1 \leq T \leq 10^2$) indicando o número de casos de teste. A primeira linha de cada caso de teste contém um inteiro par N ($2 \leq N \leq 10^3$). A segunda linha de cada caso de teste é composta por N inteiros A_i ($1 \leq A_i \leq 10^9$), separados por um espaço em branco, onde A_i é a força do guerreiro mais forte da i -ésima geração. A primeira geração é Do Mal, a segunda Do Bem, a terceira Do Mal, a quarta Do Bem, e assim sucessivamente.

Saída

A saída é composta por T linhas, cada uma contendo dois inteiros B e M , as forças dos guerreiros mais fortes Do Bem e Do Mal, respectivamente.

Exemplos de entradas	Exemplos de saídas
2	6 7
4	7 6
7 4 5 6	
4	
6 7 5 4	

H Pimenta e seu teclado

Limite de Tempo: 1s

Um belo dia, Matheus Pimenta, um tecladista mundialmente famoso, decidiu que iria compôr uma música para comemorar um grande acontecimento em sua vida. Em seguida, ele chamou um grupo de amigos para ajudá-lo.

Para isso, eles decidiram usar o teclado de Pimenta, no qual há k notas descritas cada uma por uma string composta apenas por letras minúsculas do alfabeto inglês. Para ajudar seus amigos, Matheus criou uma lista com todas as notas do teclado, tal que dois elementos consecutivos variam em 1 tom, sendo o primeiro 1 tom abaixo do segundo, ou o segundo 1 tom acima do primeiro

Ou seja, se a lista for:

do re mi

De **do** à **re** aumenta-se 1 tom, de **re** à **mi** aumenta-se 1 tom e de **do** à **mi** aumenta-se 2 tons. Além disso, de **re** à **do** abaixa-se 1 tom, de **mi** à **re** abaixa-se 1 tom e de **mi** à **do** abaixa-se 2 tons. Ao final de uma tarde, eles chegaram à uma partitura com N notas, porém Pimenta não estava feliz com o resultado. Devido à seu vasto conhecimento Pimenta formulou um conjunto de t sequências harmoniosas e dividiu-as em dois tipos:

1. $2\ a\ v$, uma sequência de 2 notas tal que da 1ª nota para a 2ª há uma diferença de a tons;
2. $3\ b\ c\ v$, uma sequência de 3 notas tal que da 1ª para a 2ª há uma diferença de b tons e entre a 2ª e a 3ª há uma diferença de c tons.

Em ambos os tipos, v é o valor harmônico da sequência. Note que a ordem das notas é importante: **do re** pode ser uma sequência harmoniosa sem que **re do** o seja.

Pimenta irá escolher quais notas devem permanecer na partitura e quais devem ser retiradas. Ajude-o a encontrar o maior valor harmônico da partitura, após as mudanças, tal que o valor harmonioso de uma partitura é a soma do valor harmônico de todas as subsequências contínuas de 2 e de 3 notas da partitura.

Considere que toda sequência que não foi mencionada por Pimenta possui valor harmônico 0.

Entrada

A primeira linha de entrada é um inteiro T ($1 \leq T \leq 10^2$) indicando o número de casos de teste.

Para cada caso de teste a formatação do input é a seguinte:

A primeira linha da entrada contém um inteiro k ($1 \leq k \leq 101$). A próxima linha contém k strings, cada uma com até 50 caracteres alfabéticos minúsculos, representando a lista de notas, conforme descrito anteriormente.

A terceira linha contém um inteiro t ($1 \leq t \leq 200^2$). As próximas t linhas contém as informações das por Pimenta, em uma das duas formas abaixo:

1. $2\ a\ v$, ($-100 \leq a \leq 100, 1 \leq v \leq 10^9$);

2. $3\ b\ c\ v$, $(-100 \leq b, c \leq 100, 1 \leq v \leq 10^9)$.

Assuma que, para duas linhas i e j , com $i \neq j$, tem-se $a_i \neq a_j$ e $(b_i, c_i) \neq (b_j, c_j)$.

A próxima linha contém um inteiro N ($1 \leq N \leq 100$). A linha seguinte contém N strings pertencentes à lista de notas feitas por Pimenta.

Saída

Para cada caso de teste imprima em uma linha o valor harmônico máximo da partitura após a possível remoção de elementos.

Para o primeiro caso de teste a melhor partitura é *programacao competitiva eh perfeita*, com 20 pontos. Alguns exemplos de partituras que não possuem valor harmônico máximo são: *programacao eh perfeita*, 10 pontos e *programacao competitiva eh programacao perfeita*, 13 pontos.

Exemplos de entradas	Exemplos de saídas
1	20
4	
competitiva perfeita	
programacao eh	
2	
2 -2 10	
3 -1 -1 3	
6	
programacao eh competitiva	
eh programacao perfeita	

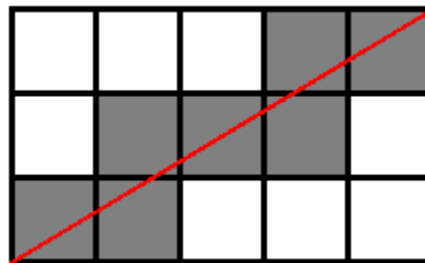
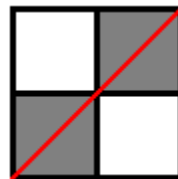
I Retângulo

Limite de Tempo: 1s

Fulano quer comprar um notebook novo. Como os preços dos notebooks estão muito altos, Fulano decidiu trabalhar.

Em seu trabalho, Fulano se deparou com o seguinte problema. Considere um retângulo de A metros de altura e L metros de largura preenchido por quadrados de 1 metro de lado. Trace uma diagonal de uma ponta do retângulo a outra. Quantos quadrados de 1 metro de lado são atravessados por essa diagonal?

Um quadrado só é considerado atravessado pela diagonal, se esta passar por dentro desse quadrado. Isto é, um quadrado não é considerado atravessado se a diagonal bater em sua quina. No retângulo 2×2 , por exemplo, somente dois quadrados são considerados como atravessados, pois a diagonal só atravessa a parte interior de dois quadrados.



Entrada

A primeira linha da entrada é um inteiro T ($1 \leq T \leq 10^4$), indicando o número de casos de teste. Cada caso de teste é composto por dois inteiros A, L ($1 \leq A, L \leq 10^9$) separados por espaço.

Saída

A saída é composta por T linhas, uma para cada caso de teste, contendo um número correspondendo à quantidade de quadrados de 1 metro de lado atravessados pela diagonal do retângulo.

Exemplos de entradas	Exemplos de saídas
2	2
2 2	7
5 3	

J Street Fighter

Limite de Tempo: 3s

Fulano está cansado de perder para seus amigos no *Street Fighter* – um jogo de luta entre duas pessoas, onde cada jogador escolhe um personagem para batalhar entre si. Fulano está jogando *Street Fighter* com o personagem Ryu e deseja criar um combo com o máximo de dano possível. Um combo é feito acertando vários golpes em sequência, onde um golpe dá uma quantidade D de dano e, após ele ser desferido, somente alguns golpes podem ser escolhidos para dar continuidade ao combo. A cada dois golpes dados dentro do combo, o dano dos próximos golpes sofre uma redução cumulativa de 10%, até atingir o valor mínimo de 0%, o que significa que os golpes não causarão mais dano.

Por exemplo, numa sequência de 50 golpes, os 2 primeiros golpes causam 100% de dano; os dois próximos, 90% de dano; os 2 seguintes causam apenas 80%, e assim por diante. A partir do 21º golpe, qualquer golpe adicional não causa mais dano ao personagem do adversário.

Seu objetivo é ajudar Fulano a criar um combo com o maior dano possível.

Entrada

A primeira linha da entrada contém o inteiro T ($1 \leq T \leq 10$), que indica o número de casos de teste.

A primeira linha de cada caso de teste contém um inteiro N ($1 \leq N \leq 1000$), indicando quantos golpes Ryu possui. A segunda linha contém N inteiros, separados por espaços em branco, indicando o dano D_i ($1 \leq D_i \leq 10^7, 1 \leq i \leq N$) que cada golpe causa ao adversário. As N linhas contém, cada uma, um golpe. A linha correspondente ao golpe i começa com um inteiro L ($0 \leq L \leq N$), que indica a quantidade de golpes que podem dar continuidade ao combo após o golpe i ser desferido, seguido de L números, os identificadores desses golpes, respectivamente.

Saída

Para cada caso de teste imprima o valor do combo de maior dano possível, com duas casas decimais de precisão, seguido de uma quebra de linha.

Exemplos de entradas	Exemplos de saídas
1	140.00
5	
40 1 1 1 100	
2 2 5	
1 3	
1 4	
1 5	
0	

K String Top

Limite de Tempo: 2s

Dada uma string S , seja S_i uma substring de S . Entende-se por substring uma sequência contígua de caracteres de S . Uma substring S_i é chamada de top se é possível construir uma string A dada utilizando somente caracteres de S_i .

Por exemplo, considere A a string a ser construída e seja A' uma string que pode vir a se tornar A . Cada caractere de S_i pode ser utilizado uma única vez ou ser descartado; quando usado, este caractere pode ser adicionado em qualquer posição de A' .

Dada as strings S e A , determine o tamanho $|S_i|$ da menor substring top, se existir, onde $|S_i|$ é a quantidade de caracteres da substring S_i .

Entrada

A primeira linha de entrada contém um inteiro T ($1 \leq T \leq 100$), que indica o número de casos de teste. A primeira linha de cada caso de teste contém a string S ($1 \leq |S| \leq 10^4$). A segunda linha de cada caso de teste contém a string A ($1 \leq |A| \leq 10^4$). Todos os caracteres de S e A são letras maiúsculas do alfabeto inglês.

Saída

Para cada caso de teste, a saída deve ser uma única linha, contendo o tamanho da menor substring top de S . Caso o problema não tenha solução, imprima o valor -1.

Exemplos de entradas	Exemplos de saídas
3	4
AABBCC	2
ABC	-1
TOPPER	
PP	
XXYZW	
XYA	

L SoloQ Fulaninho Vrau RX

Limite de Tempo: 3s

Fulano, Ciclano e Beltrano jogam LoL e são bronze. No LoL, dentre várias outras coisas, é possível obter RPs e usá-los para comprar *skins* (ou roupinhas) para você ou algum amigo.

Certo dia, os três jogaram por horas na casa de Beltrano. Fulano esqueceu de sair de sua conta quando foi embora. Obviamente, Ciclano e Beltrano não podiam deixar essa oportunidade de *trollar* e pensaram na seguinte brincadeira:

1. vão gastar **todos** os RPs de Fulano comprando *skins* para pessoas aleatórias de sua lista de amigos;
2. vão comprar somente *skins* de 675 e 1350 RPs;

Depois de terminarem, viram o histórico e se perguntaram: sabendo quantas *skins* de 675 e 1350 RPs tem no LoL, de quantas formas diferentes o histórico de compras poderia ficar?

Considere que neste jogo existe um histórico de compras, onde é possível ver toda a movimentação de compra de itens, e que em compras de *skins* de presente, apareça somente as *skins* compradas. Considere também que Fulano é bem conhecido no bronze e sua lista de amigos é tão grande que pode ser considerada infinita.

Por exemplo:

Considerando skins de 675 RP como: “SKIN A”, “SKIN B”. Ou seja, B=2.

Considerando skins de 1350 RP como: “SKIN C”, “SKIN D”, “SKIN E”. Ou seja C=3.

E com fulano tendo 1350 RP os históricos possíveis são:

Histórico 1		Histórico 2		Histórico 3		Histórico 4	
Skin	RP	Skin	RP	Skin	RP	Skin	RP
SKIN A	675	SKIN A	675	SKIN B	675	SKIN B	675
SKIN A	675	SKIN B	675	SKIN A	675	SKIN B	675

Histórico 5		Histórico 6		Histórico 7	
Skin	RP	Skin	RP	Skin	RP
SKIN C	1350	SKIN D	1350	SKIN E	1350

São possíveis 7 históricos distintos. Vale notar que nos históricos 1 e 4 uma mesma skin aparece mais de uma vez, mas só quer dizer que foram dadas de presente para pessoas distintas.

Entrada

A entrada é composta por vários casos de teste.

Cada caso de teste é representado por uma linha contendo três inteiros $B, C (2 < B, C < 1000)$ e $N (675 \leq N < 10^{15})$, onde B é a quantidade de *skins* de 675 RPs, C é a quantidade de *skins* de 1350 RPs e N a quantidade de RP que o Fulano tinha. É garantido que sempre é possível gastar exatamente todos os RPs de Fulano.

Saída

A quantidade de formas diferentes que histórico de compras de Fulano pode assumir, de acordo com a brincadeira de seus amigos. Como esse número pode ser muito grande, imprima o resto da divisão por $10^9 + 7$.

Exemplos de entradas	Exemplos de saídas
2 3 1350	7
2 5 4050	1189