

Universidade de Brasília

Maratona de Programação UnB/CIC 2016

24 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 Anas

Limite de Tempo: 4s

As irmãs Ana Clara, Ana Maria e Ana Regina leram o livro Poliana e viram o filme Corrente do Bem. Isto foi o suficiente para que elas resolvessem que todos os moradores do pequeno município em que residem deveriam ser amigos!

Empiricamente, elas determinaram que o tempo necessário para se estabelecer uma amizade entre duas pessoas depende de dois fatores: idade e sexo de cada um. O número de semanas necessárias para concretizar a amizade é igual à diferença entre as idades, em semanas. Se as pessoas forem de sexos opostos, o tempo é dobrado. Por exemplo, Pedro, de 25 anos, e Maria, de 18 anos, levariam 14 semanas para se tornarem amigos.

Contudo, por motivos diversos (distância, opinião, etc.), algumas pessoas não podem estabelecer uma amizade diretamente. Mas isso não preocupa às Anas pois a relação de amizade é transitiva: o amigo do meu amigo também é meu amigo!

Dadas as idades e sexos dos demais moradores do município, e os possíveis impedimentos de amizade direta, determine o tempo mínimo necessário para que todos se tornem amigos. Assuma que, inicialmente, apenas as Anas são amigas entre si e que uma mesma pessoa possa desenvolver várias relações de amizade em paralelo.

Entrada

A entrada consiste em uma série de, no máximo, 25 casos de teste. A primeira linha de um caso de teste contém o número N ($3 \leq N \leq 1.000$) de moradores do município (sem contar as Anas). A linha seguinte contém três inteiros I ($1 \leq I \leq 120$), separados por um espaço em branco, indicando as idades das três irmãs, respectivamente.

As N linhas seguintes contém, cada uma, um inteiro I ($1 \leq I \leq 120$) e um caractere S ($S \in \{F, M\}$), indicando a idade e o sexo de cada morador do município, respectivamente.

A linha seguinte contém o inteiro M ($0 \leq M \leq 2N$), que indica o número de possíveis impedimentos de amizade direta. Um mesmo cidadão pode ter, no máximo, dois impedimentos.

Por fim, as M linhas seguintes contém dois inteiros A e B ($1 \leq A, B \leq N$, $A \neq B$), indicando que o cidadão A não pode estabelecer uma amizade diretamente com o cidadão B .

Saída

Para cada caso de teste, a saída deve ser a mensagem “Caso t : T ”, onde t é o número do caso de teste e T o tempo necessário para que todos se tornem amigos, direta ou indiretamente. Atente à formatação de T , apresentada nos exemplos, e considere que todo ano tem 52 semanas. Assuma que T será sempre maior ou igual a uma semana.

Exemplos de entradas	Exemplos de saídas
4	Caso 1: 1 ano(s) e 2 semana(s)
16 18 21	Caso 2: 20 semana(s)
60 M	Caso 3: 2 ano(s)
70 F	
33 F	
89 M	
3	
1 2	
2 4	
4 3	
3	
10 10 10	
20 F	
10 M	
30 M	
0	
3	
10 10 10	
3 F	
62 M	
62 M	
2	
1 2	
1 3	
0	

B Balança mas não cai

Limite de Tempo: 3s

O *game show* **Balança mas não cai** premia os participantes pela coragem, sangue frio, equilíbrio, sorte e habilidade matemática.

As regras do jogo são simples: cada participante se posiciona, vendado, no centro de uma prancha sobre uma piscina. Do centro da prancha, ele consegue dar até 9 passos para a direita ou a esquerda: 10 ou mais passos em uma das direções resulta na queda na piscina e na alegria dos demais participantes.

O jogo dura, no máximo, 21 rodadas, onde a cada rodada são dadas ao participante duas opções de movimento dentre as descritas a seguir:

1. J : o participante deve pular e permanecer onde está;
2. M : o participante deve ir para o lado oposto do que se encontra, ficando à mesma distância da borda que estava antes (se estiver no centro, lá deve permanecer);
3. n : mover-se n unidades para a direita ($0 \leq n \leq 15$);
4. $-n$: mover-se n unidades para a esquerda ($15 \leq n < 0$).

Caso o participante dê mais de 3 pulos, a prancha cede e ele também vai para a piscina. Além disso, há um fator crucial para dar mais emoção ao jogo: nas rodadas 5, 10, 15 e 20, denominadas rodadas especiais, o movimento a ser feito é determinado pela produção!

Dados os movimentos das rodadas especiais e as opções de movimento das demais rodadas, determine qual é a rodada máxima que o participante consegue chegar antes de cair na água (ou vencer o jogo, caso atinja a 21ª e última rodada).

Entrada

A entrada consiste em uma série de T ($1 \leq T \leq 100$) casos de teste.

A primeira linha de um caso de teste contém quatro comandos de movimento S_i ($S_i \in \{M, J\} \cup [-15, 15]$, $1 \leq i \leq 4$), separados por um espaço em branco, que correspondem às rodadas 5, 10, 15 e 20, respectivamente. As 17 linhas seguintes contém dois comandos de movimento C_j ($C_j \in \{M, J\} \cup [-15, 15]$, $1 \leq j \leq 2$), separados por espaços em branco, com as opções disponíveis nas rodadas do jogo, em ordem crescente e excluídas as rodadas especiais.

Saída

Para cada caso de teste, deve ser impressa, em uma linha, a mensagem “Caso t : R ”, onde t é o número do caso de teste e R é a rodada mais distante possível que o participante pode atingir.

Exemplos de entradas	Exemplos de saídas
2	Caso 1: 19
J J J J	Caso 2: 13
3 -5	
2 1	
-3 -8	
12 4	
1 -4	
3 -6	
-11 7	
9 M	
J 10	
M -2	
5 -5	
8 -9	
1 -2	
J M	
J 9	
M 4	
-4 6	
3 13 -7 -11	
-5 15	
5 J	
-4 15	
6 3	
-15 -7	
9 -14	
-6 11	
0 0	
11 -7	
14 -11	
J -13	
-1 -6	
-2 2	
3 5	
M 13	
-7 5	
3 -14	

C Cebolinha

Limite de Tempo: 1s

Cebolinha, personagem clássico de Maurício de Souza, tem uma peculiaridade bastante conhecida: ele troca o 'R' por 'L' em suas falas. Contudo, há dois pontos importantes que podem passar despercebidos: ele fala o erre ele estiver no fim de uma palavra, e ele fala **elado**, e não **ellado**!

Para evitar que o roteirista esqueça alguma das regras da fala do Cebolinha, escreva um programa que receba o roteiro de uma história e faça as substituições adequadas. O roteiro tem a seguinte estrutura:

1. cada linha tem as falas de uma personagem, e o nome do personagem é dado logo no início da linha, seguido do caractere ':';
2. o nome da personagem começa sempre com letra maiúscula;
3. as falas são compostas por caracteres alfabéticos maiúsculos e minúsculos, e
 1. todas as linhas terminam com o caractere '!';
 2. todas as palavras são separadas por um único espaço em branco;
 3. não há palavras com mais de dois erres consecutivos;
 4. nenhuma palavra termina com dois ou mais erres.

Entrada

A entrada consiste em um roteiro de uma história com, no máximo, 1.000 linhas. Cada linha contém, no máximo, 200 caracteres.

Saída

O roteiro da entrada, com as falas do Cebolinha devidamente adaptadas para o seu jeito **diferente** de falar!

Exemplos de entradas	Exemplos de saídas
Cebolinha: Ei, Cascao!	Cebolinha: Ei, Cascao!
Cascao: Turma? Aonde voces vao?!	Cascao: Turma? Aonde voces vao?!
Cebolinha: Pra uma partida de futebol!	Cebolinha: Pla uma paltida de futebol!
Cebolinha: Quer vir com a gente?!	Cebolinha: Quer vir com a gente?!
Cascao: Legal! Vou tambem!	Cascao: Legal! Vou tambem!
Cebolinha: Rapido! Ja vai comecar!	Cebolinha: Lapido! Ja vai comecar!

D Dúvida

Limite de Tempo: 1s

Juca ficou surpreso quando descobriu que -40° Celsius e -40° Fahrenheit correspondiam exatamente a mesma temperatura! Curioso com o fato, indagou o seu professor que esclareceu que, em duas escalas lineares distintas, sempre haverá um valor tal que, em ambas escalas, corresponde a mesma medida.

Embora tenha acreditado no professor, Juca ficou na dúvida: como determinar este valor?

Ajude o jovem Juca em sua dúvida, determinando o ponto fixo entre duas escalas lineares $[a, b]$ e $[c, d]$.

Entrada

A entrada consiste em uma série de, no máximo, 1.000 casos de teste. Cada caso de teste é apresentado em uma linha com quatro inteiros a, b, c e d ($-1.000 \leq a, b, c, d \leq 1.000$, $a < b$, $c < d$, $b - a \neq d - c$), separados por um espaço em branco.

A entrada termina com os valores $a = b = c = d = 0$, os quais não devem ser processados.

Saída

Para cada caso de teste, deve ser impressa, em uma linha, a mensagem “Caso t : P ”, onde t é o número do caso de teste e P é o ponto fixo entre as duas escalas, com uma casa decimal de precisão.

Exemplos de entradas	Exemplos de saídas
0 100 32 212	Caso 1: -40.0
0 10 0 20	Caso 2: 0.0
-10 10 -100 100	Caso 3: 0.0
-100 0 0 200	Caso 4: -200.0
17 911 -538 672	Caso 5: 1587.2
0 0 0 0	

E Embalando brinquedos

Limite de Tempo: 2s

Caio trabalha em uma instituição de caridade que recebe doações de brinquedos para crianças abandonadas. Todo mês de dezembro ele e sua equipe tem a tarefa de embalar os brinquedos para presentear as crianças, mas o problema é que tanto os brinquedos quanto as embalagens são doações, então nem sempre é possível embalar um brinquedo corretamente.

Cada embalagem E tem uma capacidade de carga máxima de m kg, e cada brinquedo tem uma massa de b kg. Desprezando as dimensões da embalagem e do brinquedo, um brinquedo pode ser inserido em uma embalagem com capacidade igual ou maior do que sua massa.

Auxilie Caio e sua equipe determinando o número máximo de brinquedos que podem ser devidamente embalados.

Entrada

A entrada consiste em T ($1 \leq T \leq 100$) casos de teste, onde o valor de T é dado na primeira linha.

Cada caso de teste é composto por 3 linhas: a primeira contém os inteiros N e M ($1 \leq N, M \leq 100.000$), separados por um espaço em branco, que representam o número de embalagens e o número de brinquedos, respectivamente.

A segunda linha do caso de teste contém N inteiros E_i ($1 \leq E_i \leq 1.000, 1 \leq i \leq N$), que representam as capacidades de carga das embalagens, separados por espaços em branco. De forma semelhante, a terceira e última linha contém M inteiros B_j , ($1 \leq B_j \leq 1.000, 1 \leq j \leq M$), que representam as massas dos brinquedos.

Saída

Para cada caso de teste deve ser impressa, em uma linha, a mensagem “Caso t : X ”, onde t é o número do caso de teste e X é o número máximo de brinquedos que podem ser devidamente embalados.

Exemplos de entradas	Exemplos de saídas
3	Caso 1: 2
2 2	Caso 2: 1
4 5	Caso 3: 0
1 3	
3 2	
3 5 1	
4 4	
4 3	
4 2 3 1	
10 7 7	

F FizzBuzz

Limite de Tempo: 1s

FizzBuzz é um problema clássico de entrevistas de emprego nos Estados Unidos. O candidato tem que imprimir os números de 1 a 100 (ou a um certo N escolhido pelo entrevistador), um por linha, tomando o seguinte cuidado: se o número for múltiplo de três, deve ser impressa a palavra “fizz”, no lugar do número; se for múltiplo de cinco, a palavra a substituir o número é “buzz”; caso o número seja, simultaneamente, múltiplo de 3 e 5, deve ser impressa a mensagem “fizzbuzz”. Impressiona o fato de que 95% dos candidatos, formados na área de Computação, Engenharia e Matemática, não consigam resolver tal problema...

Neste problema temos uma variação simples do FizzBuzz: o problema é o mesmo, mas os números devem ser impressos por extenso, não em sua forma decimal.

Entrada

A entrada consiste em uma série de, no máximo, 100 casos de teste. Cada caso de testes é representado por uma única linha, com o valor de N ($1 \leq N \leq 1.000$).

Saída

Para cada caso de testes deve ser impressa, em uma linha, a mensagem “Caso t :", onde t é o número do caso de testes. Nas N linhas seguintes devem ser impressos ou os números por extenso, ou as expressões “fizz”, “buzz”, ou “fizzbuzz”, conforme indicado no problema. Os números deve estar escritos em letras minúsculas e sem acentos, com um único espaço em branco entre os termos e usando o conectivo “e”, quando necessário (por exemplo, 117 deve ser escrito como “cento e dezessete”).

Imprima uma linha em branco entre dois casos de teste consecutivos.

Exemplos de entradas	Exemplos de saídas
15	Caso 1:
0	um
	dois
	fizz
	quatro
	buzz
	fizz
	sete
	oito
	fizz
	buzz
	onze
	fizz
	treze
	quatorze
	fizzbuzz

G Grafos Conectados

Limite de Tempo: 1s

Um grafo G é dito conectado se, para qualquer par de vértices u, v de G , com $u \neq v$, existe uma sequência de vértices $u = m_0, m_1, m_2, \dots, m_k = v$ tal que as arestas $(m_i, m_{i+1}) \in G, i = 0, \dots, k-1$. Um grafo é dito não-direcionado se $(u, v) \in G$ implica em $(v, u) \in G$ (neste caso, (u, v) e (v, u) contam como uma única aresta).

Determine o número mínimo de arestas que um grafo G não-direcionado com n vértices deve ter para que exista a garantia que G seja conectado.

Entrada

A entrada consiste em T ($1 \leq T \leq 100$) casos de teste, onde o valor de T é dado na primeira linha da entrada.

Cada caso de teste é representado por uma única linha, com o valor de n ($1 \leq n \leq 10.000$).

Saída

Para cada caso de teste deve ser impressa, em uma linha, a mensagem “Caso t : A aresta(s)”, onde t é o número do caso de teste e A o número mínimo de arestas que G deve ter para que exista a garantia de que G seja conectado.

Exemplos de entradas	Exemplos de saídas
5	Caso 1: 1 aresta(s)
2	Caso 2: 2 aresta(s)
3	Caso 3: 4 aresta(s)
4	Caso 4: 11 aresta(s)
6	Caso 5: 37 aresta(s)
10	

H Hectares

Limite de Tempo: 2s

Os habitantes do planeta Vixvo têm como características a longevidade e a avareza. Um exemplo típico da raça é o clã Itsmine, que tem um histórico bastante peculiar.

O primeiro membro do clã, Oavo, decidiu sair da pobreza e, com muito trabalho, conseguiu sua primeira propriedade: um terreno quadrado, cujo lado media 1 hectare. Seu filho, Ojr, inspirado pela história do pai, trabalhou dobrado e conseguiu, como o pai, um terreno quadrado de lado um hectare, e foi além: obteve um segundo terreno, também quadrado, de 2 hectares de lado. O terceiro membro do clã, Oneto, consolidou a tradição, e esta que perdura até os dias de hoje.

Nela, o primogênito da N -ésima geração da família tem de reconstruir toda a história de seus ancestrais, obtendo terrenos quadrados de 1, 2, 3, ... hectares de lado, até chegar a um terreno quadrado de lado N . Importante notar que, avarentos que são, nunca vendem qualquer de suas aquisições, sendo que todos os terrenos, desde o primeiro (de Oavo), são propriedade da família.

O problema está na hora de declarar os impostos: fica, a cada geração, mais difícil computar a área total de todos os terrenos do clã Itsmine. Escreva um programa que determine a área total dos terrenos do clã, após o primogênito da geração N ter cumprido sua parte na tradição. Como este valor pode ser demasiadamente grande, imprima o resto da sua divisão por $10^9 + 7$.

Entrada

A entrada consiste em um série de, no máximo, 1.000 casos de teste. Cada caso de teste é representado por um único inteiro N ($1 \leq N \leq 10^9$), que indica a geração a ser considerada nos cálculos.

A entrada termina com o valor $N = 0$, o qual não deve ser processado.

Saída

Para cada caso de teste, imprima uma linha com a mensagem “Caso t : A ”, onde t é o número do caso de teste e A é o resto da divisão da área total (em hectares²) por $10^9 + 7$.

Exemplos de entradas	Exemplos de saídas
1	Caso 1: 1
2	Caso 2: 6
3	Caso 3: 20
10	Caso 4: 1210
100	Caso 5: 8670850
1000	Caso 6: 667082919
1000000	Caso 7: 670913771
0	

I Impossível

Limite de Tempo: 1s

Feijó, professor de matemática tradicionalista de uma turma de ensino médio, encontra-se em uma posição delicada. Em mais de uma oportunidade tentou aplicar um corretivo na turma, pedindo aos alunos que realizassem tarefas matemáticas tediosas e longas, com o intuito de promover o respeito e a disciplina. Contudo, a molecada, embora bangunceira, parecia ter tino afiado com os números.

A primeira tarefa foi a soma dos n primeiros inteiros consecutivos, tarefa que resolveram sem maiores dificuldades; depois, a soma dos quadrados dos n primeiros inteiros; em seguida, a soma dos cubos, e nada: sempre respondiam a questão em poucos minutos, e de forma correta!

Feijó preparou, então, sua última cartada: eles teriam que determinar o resto da soma dos quadrados dos n primeiros números de Fibonacci por $10^9 + 7$! A meninada ouviu a descrição da tarefa sem o sorriso habitual, e o professor distribuiu um valor de n diferente para cada um deles. “Finalmente!”, pensou em segredo o professor.

Passaram-se 20 minutos de tranquilidade e paz até que Juca se levantasse e declarasse: “A soma dos quadrados dos 1.000.000 primeiros números de Fibonacci, quando dividida por $10^9 + 7$, deixa resto 810.509.319”! “Impossível!”, pensou Feijó, “será que ele está certo”?

Auxilie Feijó, determinado o resto da divisão da soma dos quadrados dos n primeiros números de Fibonacci por $10^9 + 7$, sabendo que $F_0 = 0$, $F_1 = 1$ e $F_n = F_{n-1} + F_{n-2}$, $n \geq 2$.

Entrada

A entrada consiste em uma série de, no máximo, 100 casos de teste.

Cada caso de consiste em uma única linha, contendo o valor de n ($1 \leq n \leq 10^9$). A entrada termina com o valor $n = 0$, o qual não deve ser processado.

Saída

Para cada caso de teste deve ser impressa, em uma linha, a mensagem “Caso t : R ”, onde t é o número do caso de testes e R é o resto da divisão da soma dos quadrados dos n primeiros números de Fibonacci por $10^9 + 7$.

Exemplos de entradas	Exemplos de saídas
1	Caso 1: 1
10	Caso 2: 4895
100	Caso 3: 245606600
1000	Caso 4: 112019356
10000	Caso 5: 482686076
100000	Caso 6: 853821250
1000000	Caso 7: 810509319
10000000	Caso 8: 111597035
100000000	Caso 9: 964927951
1000000000	Caso 10: 999999734
0	

J JP Segurança 24 horas

Limite de Tempo: 3s

A JP Segurança 24 horas é uma empresa com longa tradição em instalação e manutenção de câmeras de segurança no interior de comércios, residências e prédios públicos. O serviço tem tamanho prestígio e aceitação que não é incomum que o cliente já queira contratar o serviço e fazer a instalação imediatamente, assim que a visita do técnico é concluída e o orçamento é apresentado.

Para minimizar seus gastos com locomoção, uma vez que o técnico tem de retornar a central para buscar as câmeras contratadas, a empresa precisa determinar o número mínimo de câmeras que seriam necessárias para fazer a segurança de um ambiente, de modo que o técnico possa levá-las em sua primeira viagem.

Auxilie a empresa a determinar esta quantidade mínima, sabendo que:

1. as câmeras são instaladas no interior o ambiente, no teto ou em uma de suas paredes;
2. cada câmera tem alcance de 360° , e pode monitorar qualquer espaço que não esteja obstruído por uma parede;
3. os ambientes onde as câmeras serão instaladas são polígonos de N lados, e as paredes (arestas ligando os vértices do polígono) são as únicas obstruções a serem consideradas.

Um polígono é uma área delimitada por N segmentos de reta que une os N vértices v_1, v_2, \dots, v_N por meio de N arestas $e_1 = (v_1, v_2), e_2 = (v_2, v_3), \dots, e_N = (v_N, v_1)$ tais que:

1. arestas adjacentes se interceptam em um único ponto, isto é, $e_i \cap e_{i+1} = v_{i+1}$;
2. arestas não-adjacentes não se interceptam, isto é, $e_i \cap e_j = \emptyset, j \neq i + 1$.

Entrada

A entrada consiste em uma série de, no máximo, 1.000 casos de teste.

Cada caso de teste é representado por um único inteiro N ($3 \leq N \leq 10.000$), que indica o número de vértices do polígono que delimita a região do ambiente a ser monitorado.

A entrada termina com o valor $N = 0$, que não deve ser processado.

Saída

Para cada caso de testes, imprima o número mínimo de câmeras que é suficiente para monitorar o ambiente do cliente.

Exemplos de entradas	Exemplos de saídas
3	1
7	2
0	

L Little Yathzee

Limite de Tempo: 4s

Little Yathzee é uma versão simplificada para apenas um jogador do jogo Yathzee. Como na versão original, o objetivo é obter a maior pontuação com dados. Esta versão utiliza 4 dados de 4 faces (as faces de cada dado são numeradas de 1 a 4), e é jogada em 4 turnos. Cada turno consiste dos seguintes passos:

1. Lançar os 4 dados.
2. Optar por uma das seguintes:
 - **Pontuar:** escolher uma dentre as quatro formas de pontuação possíveis, desde que ainda não escolhida em turnos anteriores, pontuar nela e encerrar o turno;
 - **Relançar:** escolher qualquer combinação possível dentre os 4 dados e os lançar novamente, voltando então ao passo 2.

A opção de relançar só pode ser escolhida, no máximo, 2 vezes por turno. As 4 categorias de pontuação correspondem aos números presentes nas faces dos dados: 1, 2, 3, 4. Quando o jogador escolhe uma destas quatro categorias, a pontuação obtida é igual ao número de faces dos dados que coincidem com a categoria escolhida multiplicado pelo valor da categoria. Por exemplo, se as faces dos dados apresentam os valores 2, 3, 3, 4, a pontuação do jogador em cada categoria seria:

- **Categoria 1:** 0 pontos (não há dados com face igual a 1);
- **Categoria 2:** 2 pontos (uma ocorrência do valor 2 nas faces dos dados);
- **Categoria 3:** 6 pontos (duas ocorrências do valor 3 nas faces dos dados);
- **Categoria 4:** 4 pontos (uma ocorrência do valor 4 nas faces dos dados).

O jogador pode pontuar apenas uma vez em cada categoria, mesmo que com o valor zero. A pontuação final do jogador será a soma das pontuações obtidas em cada uma das categorias após a conclusão do quarto e último turno.

Naturalmente, o desempenho do jogador fica condicionado à sorte (isto é, aos resultados dos lançamentos dos dados). Fica, portanto, uma questão em aberto: *qual seria a pontuação máxima que o jogador obteria se fossem conhecidos, de antemão, os resultados dos lançamentos dos dados?*

Entrada

A entrada consiste em T ($1 \leq T \leq 100$) casos de teste, onde o valor de T é dado na primeira linha da entrada.

Cada caso de teste é representado por 4 linhas, cada uma contém 12 inteiros, separados por um espaço em branco, que compõem uma sequência de 48 lançamentos de dados. Cada inteiro F

($1 \leq F \leq 4$) corresponde a face resultante do (re)lançamento de um dado, na mesma ordem em que são realizados.

Por exemplo, no primeiro caso de teste dado, ao fim do passo 1 os dados apresentariam em suas faces os números 3, 3, 4 e 1. Se no passo seguinte o jogador opta por relançar os primeiro, segundo e quarto dados, as faces resultantes mostrariam os números 4, 2, 4 e 2.

Saída

Para cada caso de testes deve ser impressa, em uma linha, a mensagem “Caso t : P ”, onde t é o número do caso de teste e P é a pontuação máxima que pode ser obtida a partir da sequência de lançamentos dada.

Exemplos de entradas	Exemplos de saídas
3	Caso 1: 32
3 3 4 1 4 2 2 4 4 2 3 2	Caso 2: 29
1 4 4 4 4 3 3 3 2 1 1 4	Caso 3: 34
1 2 4 3 2 2 1 4 4 4 1 2	
1 3 1 1 1 1 4 4 2 2 4 1	
2 4 1 3 2 1 4 4 1 1 4 3	
1 3 1 3 2 3 1 1 4 1 4 4	
2 1 1 3 4 2 2 4 3 4 4 1	
1 1 1 1 4 2 2 4 1 2 4 1	
4 4 1 1 1 2 1 2 3 2 1 2	
4 4 3 2 4 3 4 4 2 1 3 3	
3 3 1 2 3 3 4 3 1 1 4 3	
2 3 4 2 4 4 1 1 1 3 2 3	

M MDC

Limite de Tempo: 3s

Márcio está tentando fazer seu trabalho final de Matemática Discreta: escrever uma rotina que determine o maior divisor comum (MDC) de uma lista de N inteiros positivos. Contudo, como ele perdeu algumas aulas ao longo do semestre, ele só sabe computar o MDC do jeito que aprendeu no ensino fundamental: listar todos os números, separados por vírgulas, depois escrever uma barra vertical e, em seguida, testar se cada um dos números é ou não divisível por cada um dos números primos, em sequência.

A versão inicial de sua rotina é deveras lenta, e ele solicitou sua ajuda: você poderia escrever para Márcio uma subrotina que, dado um intervalo $[a, b]$ de índices dos número da lista, com $a \leq b$, determine se algum dos inteiros n_i , $a \leq i \leq b$, é ou não divisível por um primo p dado.

Entrada

A entrada consiste em uma série de, no máximo, 10 casos de teste. A primeira linha de cada caso de testes contém os inteiros N e Q ($1 \leq N \leq 10^6$, $1 \leq Q \leq 10^4$), separados por um espaço em branco, que representam o número de inteiros na lista e o número de vezes que a subrotina deve ser chamada.

A linha seguinte contém os N inteiros n_i ($1 \leq n_i \leq 700$, $1 \leq i \leq N$), separados por um espaço em branco. Por fim, as Q linhas seguintes contém, cada uma, os parâmetros de chamada da subrotina: a, b, p ($1 \leq a \leq b \leq N$, $1 \leq p \leq 700$), separados por um espaço em branco, onde a e b representam o intervalo em questão e p é um número primo.

Cuidado: a entrada é grande (quase 4MB), então é preciso ler os dados de forma eficiente!

Saída

Para cada caso de teste deve ser impressa, em uma linha, a mensagem “Caso t :", onde t é o número do caso de teste. Cada uma das Q linhas seguintes deve conter o resultado da chamada da subrotina, no formato “Q q : p ? R ”, onde q é o número da chamada da rotina, p é o primo utilizado na chamada e R é o retorno da rotina: “S”, para sim, “N”, para não. Imprima uma linha em branco entre dois casos de teste consecutivos.

Exemplos de entradas	Exemplos de saídas
10 4	Caso 1:
7 8 3 11 9 6 1 2 17 4	Q 1: 2 ? N
3 3 2	Q 2: 3 ? N
1 2 3	Q 3: 11 ? S
1 10 11	Q 4: 17 ? N
5 8 17	
3 3	Caso 2:
15 12 20	Q 1: 2 ? N
1 1 2	Q 2: 3 ? N
3 3 3	Q 3: 5 ? S
2 3 5	
0 0	

N Número de Pontos

Limite de Tempo: 4s

Uma curva elíptica é uma curva plana definida por uma equação de Weierstrass, isto é,

$$y^2 = ax^3 + bx + c,$$

com $\Delta = -16(4a^3 + 27b^2) \neq 0$. Um ponto (x, y) pertence a curva se satisfaz a equação acima. Além destes, a curva tem uma solução projetiva, denominada ponto no infinito, que serve como elemento neutro da adição e que não segue diretamente da equação na forma apresentada.

Neste problema você deve determinar se os parâmetros a , b e c dados formam ou não uma curva elíptica e, em caso afirmativo, determine quantos pontos incongruentes pertencem a curva. Toda a aritmética deve ser feita em no corpo finito \mathbb{F}_p , isto é, todas as operações de soma e produto devem ser feitas módulo p , onde p é primo.

Entrada

A entrada consiste em uma série de, no máximo, 10 casos de teste. Cada caso de teste é representado por uma única linha, com os valores de a , b , c e p ($1 \leq a, b, c < p$, $2 \leq p \leq 10^6$), separados por um espaço em branco.

Saída

Para cada caso de teste deve ser impressa, em uma linha, a mensagem “Caso t : P ponto(s)”, onde t é o número do caso de teste e P o número de pontos distintos da curva elíptica. Caso os parâmetros não formem um curva, a mensagem deve ser “Caso t : nao e curva”.

Exemplos de entradas	Exemplos de saídas
2 3 4 5	Caso 1: nao e curva
4 3 2 5	Caso 2: 5 ponto(s)
2 9 11 13	Caso 3: 11 ponto(s)
33 71 92 97	Caso 4: 85 ponto(s)