

I Maratona de Programação do IFB

Maratona

26 de agosto de 2017



(Este caderno contém 13 problemas)

Comissão Organizadora:

Prof. Daniel Saad Nogueira Nunes (IFB)
Prof. Edson Alves da Costa Júnior (UnB/FGA)
Jeremias Moreira Gomes (UnB)

Apoio:

Caio César Félix Caldeira (Discente – IFB)
Danyelle da Silva Oliveira Angelo (Discente – IFB)
Flávia Dias Campos (Discente – IFB)
Gregory Henrique Moreira dos Santos Pereira (Discente – IFB)
Italo de Souza Moraes (Discente – IFB)
Jerônimo Hermano Neves Cunha (Discente – IFB)
Marcus Vinícius de Souza Almeida (Discente – IFB)
Rafael Passos (Discente – IFB)

Problema A

Pizza Quadrada

Limite de tempo: 1s

Autor: Edson Alves da Costa Júnior

Com o aumento da demanda popular, a pizzaria Na Hora passou a oferecer, aos seus clientes, pizzas em formato quadrado. Contudo, ela possui um grande estoque de embalagens circulares, e o gerente gostaria de utilizá-las provisoriamente até que a encomenda de embalagens quadradas seja entregue.

Dado o raio ‘R’ da embalagem circular, determine o lado ‘L’ da maior pizza quadrada que pode ser transportada nesta embalagem, e também a área interna ‘A’ da embalagem que ficará desocupada quando for utilizada para transportar uma pizza quadrada.

Entrada

A entrada consiste em T ($1 \leq T \leq 1.000$) 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 contém o valor do inteiro R ($1 \leq R \leq 1.000$), que corresponde ao raio da embalagem circular.

Saída

Para cada caso de testes devem ser impressos, em uma linha e separados por um espaço em branco, os valores de L e A , com 3 casas decimais de precisão. Para cada um destes valores, se sua resposta é x e a resposta do juiz é y , sua resposta será considerada correta se $\frac{|x-y|}{\max(1,y)} \leq 10^{-3}$.

Exemplo de entrada

Exemplo de entrada	Saída para o exemplo de entrada
3	2.828 4.566
2	7.071 28.540
5	56.569 1826.548
40	

Problema B

Números de Thabit

Limite de tempo: 1s

Autor: Edson Alves da Costa Júnior

Um número natural T_k é denominado um número de Thabit se ele pode ser escrito na forma $T_k = 3 \cdot 2^k - 1$, onde k é um número inteiro não-negativo. Os primeiros quatro números de Thabit são 2, 5, 11, 23. Um número natural maior do que 1 é primo se os únicos divisores deste número são 1 e ele próprio.

Dado um número natural N , classifique-o como primo, número de Thabit, primo de Thabit (número de Thabit que também é primo) ou composto e não Thabit (nem primo, nem Thabit).

Entrada

A entrada consiste em uma série de casos de teste. A primeira linha da entrada contém o número T ($1 \leq T \leq 1.000$) de casos de teste. Cada caso de testes é representado por uma única linha, contendo um número inteiro N ($1 \leq N \leq 1.000.000$).

Saída

Para cada caso de testes deverá ser impressa a mensagem “Caso # t : C ”, seguida de uma quebra de linha, onde t é o número do caso de teste (cuja contagem se inicia no número 1) e C é a classificação do número, conforme apresentado no texto, sem acentos.

Exemplo de entrada

Exemplo de entrada	Saída para o exemplo de entrada
4	Caso #1: primo
7	Caso #2: numero de Thabit
3071	Caso #3: primo de Thabit
2	Caso #4: composto e nao Thabit
6	

Problema C

Ajude Raio Negro

Limite de tempo: 1s

Autor: Jeremias Moreira Gomes

Os Inumanos são uma raça guerreira que viviam em harmonia com os humanos no planeta Terra, até o governo mundial decidir lhes aplicar um golpe militar, aniquilando todos aqueles não-humanos.

Na tentativa de salvar seu povo, o rei do Inumanos, Raio Negro, tomou a linha de frente da batalha, enquanto todos fugiam para a Lua.

Raio Negro, assim como toda a inumanidade, foi exposto a névoa terrigênise que provê distintos poderes para cada ser. Dentre os poderes adquiridos por Raio Negro, está o seu grito semi-sônico poligonal-convexo, que é um fenômeno baseado na interação entre partículas e sua voz, permitindo a ele gerar polígonos convexos com alto poder de destruição, sempre que emana qualquer som da sua boca.

O grito semi-sônico poligonal-convexo é tão perigoso que Raio Negro só o utiliza como último recurso (motivo pelo qual ele nunca conversa), e agora, nesse golpe militar, ele foi alvejado por muitos tiros ao mesmo tempo e precisa utilizá-lo para parar todos os tiros, porém com o menor efeito colateral possível. Ou seja, gerando um grito semi-sônico poligonal-convexo de menor área possível.

Acontece que sua alteza, Raio Negro, sabe gerar os polígonos convexo com alta precisão se lhe forem informados as coordenadas, mas não consegue calcular, apenas olhando para a posição de todos os tiros. Dessa forma, você, Programmerboy, resolveu ajudar Raio Negro com seu cérebro-processador quântico, que permite fazer cálculos mais rápidos que tudo que existe, e a sua projeção mental, para dizer a Raio Negro qual é a área mínima do polígono convexo que ele deverá gerar com o grito semi-sônico poligonal-convexo.

Assim, dado um conjunto de balas, que foram atiradas ao mesmo tempo, ajude Raio Negro a executar o seu grito semi-sônico poligonal-convexo com a menor área possível que pare todos os tiros de uma só vez.

Entrada

A entrada consiste de vários casos de teste. Cada caso de teste recebe, inicialmente, um inteiro N , ($3 \leq N \leq 5.000$), que representa a quantidade de balas que estão vindo na direção de Raio Negro ao mesmo tempo. Nas N linhas seguintes, são informados valores X_i e Y_i ($-5.000 \leq X_i, Y_i \leq 5.000$), que indicam as coordenadas de cada bala que deverá ser parada por Raio Negro.



Os casos de teste devem parar de ser processados quando $N = 0$.

Saída

Para cada caso de teste, deve ser respondido um valor com duas casas decimais de precisão, representando a área mínima do polígono convexo que Raio Negro deverá criar.

Exemplo de entrada

Exemplo de entrada	Saída para o exemplo de entrada
4	1.50
1 1	9.00
2 2	
1 3	
2 3	
5	
4 4	
1 1	
2 3	
1 4	
4 1	
0	

Problema D

Máximos e Mínimos

Limite de tempo: 1s

Autor: Edson Alves da Costa Júnior

De acordo com o Cálculo, uma função contínua $f(x)$ em um intervalo fechado $[a, b]$ admite um máximo e um mínimo global. Os valores x onde podem ocorrer este máximo ou mínimo são os extremos deste intervalo (a e b) e os pontos críticos (isto é, pontos onde a derivada $f'(x)$ é igual a zero ou não está definida dentro deste intervalo). O máximo ou mínimo globais correspondem aos valores $f(x)$ que são os maiores ou os menores possíveis considerados todos os x pertencentes ao intervalo $[a, b]$.

Considere o polinômio de terceiro grau $p(x) = Cx^3 + Dx^2 + Ex + F$, com $C \neq 0$. Dados os valores dos coeficientes de $p(x)$ e o intervalo $[a, b]$, determine o mínimo e o máximo global de $p(x)$ neste intervalo.

Entrada

A primeira linha da entrada contém os valores a e b que delimitam o intervalo fechado. A segunda linha contém os coeficientes C, D, E, F do polinômio $p(x)$.

Todos os valores estão separados por espaços em branco, e são números reais, com uma casa decimal de precisão, com $a \leq b$ e $C \neq 0$.

Saída

A primeira linha da saída deve ser a mensagem “Mínimo local: m ”, onde m é o valor mínimo global de $p(x)$ no intervalo $[a, b]$.

De forma similar, a segunda linha da saída deve ser a mensagem “Máximo local: M ”, onde M é o valor máximo global de $p(x)$ no intervalo $[a, b]$. Ao final desta mensagem deve ser impressa uma quebra de linha.

Os valores m e M devem ser apresentados com 3 casas decimais de precisão.

Exemplo de entrada

Exemplo de entrada	Saída para o exemplo de entrada
-2.0 1.0	Mínimo local: -8.000
1.0 0.0 0.0 0.0	Máximo local: 1.000
1.0 4.0	Mínimo local: -64.000
-1.0 0.0 0.0 0.0	Máximo local: -1.000
1.9 3.1	Mínimo local: 13.500
1.0 -7.5 18.0 0.0	Máximo local: 14.000
1.9 2.3	Mínimo local: 2.999
1.0 -6.0 12.0 -5.0	Máximo local: 3.027

Problema E

Color Combo

Limite de tempo: 1s

Autor: Edson Alves da Costa Júnior

No puzzle *Color Combo!* o jogador deverá organizar as peças coloridas que caem do alto da tela de acordo com suas respectivas cores. Quando três ou mais peças de cada cor estiverem conectadas (uma peça está conectada à outra se estiver em contato direto em uma das quatro direções cardeais com alguma outra peça de mesma cor), as peças são destruídas, e o jogador obtém uma pontuação P .

Esta pontuação depende do número de peças destruídas, e é igual a um valor base B , acrescido de 100 pontos para cada peça acima de três. O total é acrescido em 25% se foram destruídas mais de três de peças, ou 50%, se o total de peças destruídas for superior a sete.

Dado o valor base B e a quantidade N de peças destruídas, determine a pontuação obtida pelo jogador.

Entrada

A entrada consiste em uma série de casos de teste. A primeira linha da entrada contém o inteiro T ($1 \leq T \leq 1.000$), que determina o número de casos de teste. Cada caso de teste é representado por uma única linha com os inteiros B ($1 \leq B \leq 100.000$) e N ($3 \leq N \leq 1.000$), separados por um espaço em branco.

Saída

Para cada caso de teste deverá ser impressa a mensagem “Caso t : P pontos”, seguida de uma quebra de linha, onde t é o número do caso de teste (cuja contagem inicia com o número 1) e P é a pontuação obtida pela destruição de N peças.

O valor de P deve ser um número inteiro: despreze a parte decimal, se houver. Se P for igual a 1, o texto deve ser “ponto”, ao invés de “pontos”.

Exemplo de entrada

Exemplo de entrada	Saída para o exemplo de entrada
4	Caso 1: 1 ponto
1 3	Caso 2: 200 pontos
60 4	Caso 3: 343 pontos
75 5	Caso 4: 2250 pontos
1000 8	

Problema F

Numero de Série Tabajara

Limite de tempo: 1s

Autor: Daniel Saad Nogueira Nunes

A corporação Tabajara começou a produzir eletrônicos em geral para os consumidores. Cada produto possui um número de série específico contendo apenas dígitos.

A lógica de um número de série de um produto Tabajara novo é a seguinte: utiliza-se o número imediatamente maior que o último número de série produzido com a limitação de ater-se aos mesmos dígitos. Assim, dado um número de série do último produto produzido, deve ser determinado o próximo número de série a ser utilizado seguindo estes critérios.

Entrada

A entrada consiste de vários casos de teste. Cada caso de teste possui uma única linha contendo um número positivo N com no máximo 100 dígitos. A entrada termina quando o $N = -1$.

Saída

Para cada caso de teste, deverá ser impressa uma linha contendo o próximo número de série ou -1 , caso não haja um próximo número de série.

Exemplo de entrada

Exemplo de entrada	Saída para o exemplo de entrada
123	132
17654	41567
321	-1
-1	

Problema G

Empilhando Caixas

Limite de tempo: 1s

Autor: Edson Alves da Costa Júnior

O chefe do almoxarifado de uma pequena empresa deu a seguinte instrução aos seus funcionários: uma caixa só pode ser colocada em cima de uma caixa que contém massa maior ou igual a da caixa a ser empilhada.

As caixas a serem empilhadas tem mesmas dimensões e são fabricadas com mesmo material, de modo que são diferenciadas apenas pelas etiquetas que indicam a massa total armazenada na caixa.

Com a chegada de uma nova remessa de material, os funcionários reservaram o espaço necessário para uma nova pilha de caixas. Um funcionário trazia as caixas do caminhão para a porta do almoxarifado, enquanto um segundo funcionário consultava a etiqueta da caixa recém-chegada: se a massa contida nela permitisse o empilhamento, o funcionário a colocava na pilha; caso contrário, separava a caixa numa sala ao lado.

Sabendo que a pilha pode ter, no máximo, seis caixas, e conhecida a ordem em que as caixas foram retiradas do caminhão e analisadas, determine a massa total que o funcionário conseguiu empilhar.

Entrada

A entrada consiste em uma série de, no máximo, 200 casos de teste e termina em EOF.

A primeira linha de um caso de teste contém a quantidade N ($1 \leq N \leq 20$) de caixas que serão retiradas do caminhão. As N linhas seguintes contém as massas m_i ($1 \leq m_i \leq 100$), em ordem de retirada, de cada caixa i ($1 \leq i \leq N$). A massa de cada caixa é dada em kilogramas.

Saída

Para cada caso de testes deve ser impressa, em uma linha, a mensagem “Massa empilhada: M kg”, onde M é a massa total do material contido nas caixas que foram empilhadas.

Exemplo de entrada

Exemplo de entrada	Saída para o exemplo de entrada
2	Massa empilhada: 10 kg
10	Massa empilhada: 120 kg
15	
10	
40	
25	
30	
25	
20	
8	
10	
2	
2	
1	

Problema H

Tamanduás vs Cupins

Limite de tempo: 1s

Autor: Edson Alves da Costa Júnior

No *tower defense* **Tamanduás vs Cupins**, os tamanduás comem os cupins que tentam avançar para uma casa de madeira. Cada cupim tem uma massa, em miligramas, e o Tamanduá Bandeira, uma das torres aliadas, consegue consumir um determinado número de miligramas (mg) de cupim antes de ficar cheio e não conseguir comer mais cupins, deixando os demais atravessarem seu caminho.

Há três tipos de cupim no jogo: o Filhote, o Adulto e o Rei, cada um pesando a mg, b mg e c mg, respectivamente. Em uma *wave*, os cupins são consumidos na ordem que chegam ao tamanduá, e um tamanduá não pode comer apenas parte de um cupim: se ele não conseguir consumir o cupim inteiro, ele deixará o mesmo passar para o próximo tamanduá.

Dada a massa dos diferentes tipos de cupim, capacidade de consumo do Tamanduá Bandeira, o número de inimigos e a composição da *wave*, determine o número mínimo de tamanduás que devem ser enfileirados para consumir toda a *wave*.

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.

A primeira linha de um caso de teste contém as massas a, b, c ($1 \leq a, b, c \leq 1.000$), em miligramas, dos três tipos de cupim do jogo, conforme descrito anteriormente. Estes valores são números inteiros separados por um espaço em branco.

A segunda e a terceira linhas do caso de teste informam, respectivamente, dois inteiros: a capacidade de consumo do Tamanduá Bandeira M ($a, b, c \leq M \leq 10.000$), em mg, e o número N ($1 \leq N \leq 1.000$) de inimigos na *wave*.

As próximas N linhas da entrada contém os cupins que compõem a *wave*, na ordem que chegarão aos tamanduás.

Saída

Para cada caso de teste deverá ser impressa a mensagem “Caso $\#t$: Q tamandua(s)”, onde t é o número do caso de teste (cuja contagem tem início no número um) e Q é o número mínimo de tamanduás que devem ser enfileirados para consumir toda a *wave*. Esta mensagem deve ser seguida de uma quebra de linha.

Exemplo de entrada

Exemplo de entrada	Saída para o exemplo de entrada
1	Caso #1: 2 tamandua(s)
3 8 10	
20	
3	
Adulto	
Filhote	
Rei	
3	Caso #1: 1 tamandua(s)
1 2 3	Caso #2: 3 tamandua(s)
10	Caso #3: 4 tamandua(s)
1	
Rei	
3 4 5	
6	
4	
Rei	
Filhote	
Adulto	
Filhote	
3 8 10	
20	
8	
Adulto	
Rei	
Rei	
Filhote	
Rei	
Rei	
Rei	
Filhote	

Problema I

Divisão de Doces

Limite de tempo: 1s

Autor: Daniel Saad Nogueira Nunes

Regina é uma mãe muito cuidadosa com relação aos seus filhos, Pedro e Paulo.

Sempre que ela compra doces, ela certifica-se de dividir igualmente os doces aos seus filhos. Mas Regina é uma mãe esperta, ela não leva em conta a quantidade, mas a qualidade do doce para realizar a divisão de maneira mais equânime possível.

Além disso, Regina é muito justa. Se ela não consegue dividir os doces igualmente aos seus dois filhos, ela prefere não dar os doces, assim nenhum sai no prejuízo.



Dado vários doces com suas qualidades, determine se Regina consegue dividi-los igualmente entre os seus filhos.

Entrada

A entrada consiste de vários casos de teste. Cada caso de teste possui duas linhas. A primeira linha contém um inteiro N ($0 \leq N \leq 1.000$) denotando o número de doces. A segunda linha possui N inteiros, em que cada inteiro V_i ($0 \leq V_i \leq 100$) representa a qualidade do i -ésimo doce.

Os casos de teste devem parar de ser processados quando $N = 0$.

Saída

Para cada caso de teste, deve ser respondido em uma linha “sim”, caso Regina consiga dividir os doces igualmente entre seus filhos e “nao”, caso contrário.

Exemplo de entrada

Exemplo de entrada	Saída para o exemplo de entrada
5	sim
2 2 1 3 2	nao
5	nao
2 2 2 3 2	
6	
1 2 1 2 1 2	
0	

Problema J

Famíliaes Russos

Limite de tempo: 1s

Autor: Edson Alves da Costa Júnior

Na Rússia, os nomes dos cidadãos são formados por três partes: nome, patronímico e família. Por exemplo, Yuri (nome) Constantinovitch (patronímico, filho de Constantin) Romanov (família).

De forma simplificada, o patronímico é formado a partir do nome do pai mais um sufixo, que depende do sexo do indivíduo e da terminação do nome do pai: “evich”, “ovich”, “ich”, para homens e “evna”, “ovna”, “ichna”, para mulheres. Se Ivan e Sonia são filhos de Petr, então eles tem patronímico Petrovich e Petrovna, respectivamente.

Já a família deriva do nome do patriarca, adicionado de um sufixo de forma semelhante ao patronímico: “ev”, “in”, “ov”, para homens e “ina”, “eva”, “ova”, para mulheres. Por exemplo, Petrov significa “clã de Petr”.

Dado o nome de um indivíduo russo e uma lista de cidadãos, identifique quantos quantos familiares (mesma família) e quantos irmãos (mesma família, mesmo pai) deste indivíduo há dentre os listados.

Nota: Existem outros sufixos e exceções, tanto para o patronímico quanto para a família. Para efeitos do problema, considere apenas os sufixos citados.

Entrada

A entrada consiste em uma série de, no máximo, 100 casos de teste. A primeira linha de um caso de teste contém o nome de indivíduo russo. A segunda linha contém um natural N ($1 \leq N \leq 250$) que indica a quantidade de cidadãos na lista. As próximas N linhas contém os nomes contidos na lista, um por linha. Os nomes são composto por, no máximo, 100 caracteres alfabéticos maiúsculos, minúsculos ou espaços em branco.

Saída

Para cada caso de teste deverá ser impressa a mensagem “Caso # t : P parente(s), I irmao(s)”, onde t é o número do caso de teste e P e I correspondem ao número de parentes e irmãos do indivíduo citado na primeira linha do caso de teste.

Exemplo de entrada

Exemplo de entrada	Saída para o exemplo de entrada
Igor Ivanovich Dmitriev	Caso #1: 1 parente(s), 0 irmao(s)
3	Caso #2: 3 parente(s), 2 irmao(s)
Tania Yurievna Vladimirova	
Petr Vladimirovich Dmitriev	
Raissa Igorevna Ivanova	
Ivan Petrovich Yuriev	
5	
Catia Yurievna Dmitrieva	
Petr Petrovich Vladimirov	
Katia Petrovna Yurieva	
Feodor Petrovich Yuriev	
Natasha Ivanovna Yurieva	

Problema K

Distribuidora de Calçados

Limite de tempo: 1s

Autor: Edson Alves da Costa Júnior

Uma fábrica de calçados produz mensalmente pares de sapatos de diversos números (formas), atendendo os pedidos dos comerciantes da região. A demanda está cada vez maior, de modo que a fábrica está precisando de ajuda na automação dos processos do pedidos.

Dada a quantidade de pares de sapatos produzidos durante o mês, determine se é ou não possível atender os pedidos dos comerciantes.

Entrada

A primeira linha da entrada contém o valor de N ($1 \leq N \leq 50$), que representa a quantidade de formas distintas que a fábrica produziu no mês. Nas próximas N linhas são compostas de dois números inteiros F_i ($1 \leq F_i \leq 60$) e Q_i ($1 \leq Q_i \leq 200$), os quais indicam o número da forma e a quantidade de pares produzidos da forma indicada.

Na linha seguinte é apresentado o número de pedidos P ($1 \leq P \leq 10$) dos comerciantes locais. Para cada pedido é informado, em uma linha, a quantidade E ($1 \leq E \leq 6$) de formas distintas a serem encomendadas, seguida de E linhas com os valores T_i ($1 \leq T_i \leq 60$) e M_i ($1 \leq M_i \leq 20$), que indicam o número da forma encomendada e a quantidade de pares desta forma, respectivamente.

Saída

A saída deve ser a mensagem “A fabrica consegue atender todos os pedidos” ou “A fabrica nao consegue atender todos os pedidos”, conforme for o caso. Esta mensagem deve ser seguida de uma quebra de linha.

Exemplo de entrada

Exemplo de entrada	Saída para o exemplo de entrada
1 38 100	A fabrica consegue atender todos os pedidos
2 1 38 50	
1 38 40	
2 37 50	A fabrica nao consegue atender todos os pedidos
38 25	
1	
2	
37 30	
38 30	
3	A fabrica nao consegue atender todos os pedidos
38 30	
40 20	
42 15	
2	
2	
38 10	
40 10	
1	
39 10	

Problema L

Backup

Limite de tempo: 1s

Autor: Edson Alves da Costa Júnior

A equipe da CPD (Central de Processamento de Dados) de uma empresa é responsável pelo backup semanal dos servidores. Além da cópia física, em máquinas locais e remotas, a equipe entendeu que também seria prudente fazer cópias em mídias físicas e guardá-las em um cofre.

Embora não concorde com tal procedimento, por considerá-lo desnecessário, o departamento de finanças disse que liberaria os recursos para a compra dos equipamentos, desde que o número de equipamentos adquiridos fosse o menor possível e que os equipamentos fossem utilizados em sua totalidade, isto é, cada equipamento deveria armazenar dados em quantidade igual a sua capacidade máxima.

O tamanho do *backup* é dado em *gibibytes* (GiB), e os equipamentos que podem vir a ser adquiridos são: pendrives de 1, 2, 4, 8, 16, 32 ou 64GiB; unidades de estado sólido de 128, 256 ou 512GiB; e discos rígidos convencionais de 1, 2, 4 ou 8 *tebibytes* (TiB, onde 1TiB = 1024GiB).

Dado o tamanho do *backup* semanal, em GiB, e considerando que a CPD consegue dividir este *backup* em pacotes do tamanho necessário e que possam ser gravados utilizando a íntegra do espaço disponível nos dispositivos escolhidos, determine o número mínimo de equipamentos a serem adquiridos.

Entrada

A entrada consistem em uma série de casos de teste. Cada caso de teste é representado por uma única linha da entrada, com um inteiro indicando o tamanho T ($1 \leq T \leq 16.000$), em GiB, do *backup* semanal a ser gravado nos equipamentos.

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

Saída

Para cada caso de teste deve ser impresso o número mínimo de equipamentos que devem ser adquiridos para armazenar o *backup* semanal. Por exemplo, no primeiro caso de teste, um *backup* de 38 GiB, embora um único pendrive de 64 GiB (ou uma unidade de qualquer outro equipamento de maior capacidade) fosse suficiente para armazenar esta quantidade dados, haveria espaço inutilizado ao final da cópia, contrariando a determinação do departamento de finanças. Por isso são necessários 3 dispositivos: um pendrive de 32 GiB, um pendrive de 4 GiB e um pendrive de 2 GiB (veja que todos serão utilizados em 100 % ao final da cópia).

Exemplo de entrada

Exemplo de entrada	Saída para o exemplo de entrada
38	3
1024	1
2	1
15	4
80	2
12437	6
0	

Problema M

Ajude Superman

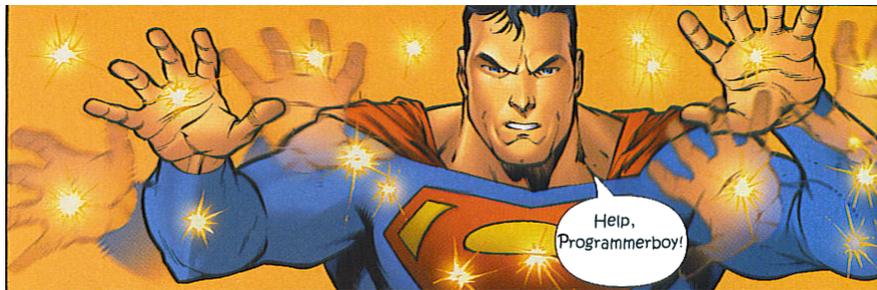
Limite de tempo: 2s

Autor: Jeremias Moreira Gomes

Face ao perigo, Superman está enfrentando um dos seus piores inimigos, o Metallo. Metallo é um ciborgue que construiu uma arma para causar estragos e incitar uma briga com o superman: a metalladora.

A metalladora é uma arma que atira balas a uma velocidade surpreendentemente rápida. Suas munições não machucam o superman, mas a velocidade dos tiros é tão alta que Superman só consegue utilizar uma mão para pegar as balas (ele precisa concentrar toda a sua supervelocidade em uma mão e isso o deixa intangível em todo o resto do corpo), e ainda assim tem dificuldades em escolher quais balas ele deve pegar.

O que Metallo não sabia é que Superman pode contar com a ajuda do nosso mais novo super-herói: você, o Programmerboy!



O Programmerboy não possui a mesma velocidade corporal que a metalladora ou o Superman alcançam, mas possui dois poderes extremamente úteis: o cérebro-processador quântico, que permite fazer cálculos mais rápidos que tudo que existe, e a projeção mental que, quando combinada com o seu cérebro, permite instruir o Superman sobre quais balas ele deverá pegar, para minimizar o estrago causado pela metalladora.

Assim, dado um conjunto de balas atiradas e o momento em que cada uma é disparada pela metalladora, ajude Superman a pegar a maior quantidade de munições possível.

Entrada

A entrada consiste de vários casos de teste. Cada caso de teste recebe, inicialmente, dois inteiros I, J ($3 \leq I, J \leq 25$), que representam a grade de possíveis posições para cada tiro dado pela arma do vilão. Na linha seguinte são informados dois valores X e Y ($1 \leq X \leq I, 1 \leq Y \leq J$), que indicam a posição inicial da mão do superman. Em seguida, é passado um valor K ($1 \leq K \leq 1.000$), que informa a quantidade de tiros que foi disparado pela arma. Nas K linhas seguintes, são informados valores X_i, Y_i, M_i ($1 \leq X_i \leq I, 1 \leq Y_i \leq J, 1 \leq M_i \leq 5 \cdot K$), que representam a linha, a coluna e o momento em que a mão do superman deve estar, para ser possível pegar a bala.

É garantido que nenhuma bala alcançará superman no mesmo momento.

Os casos de teste devem parar de ser processados quando $I = J = 0$.

Saída

Para cada caso de teste, deve ser respondido com um inteiro, representando a quantidade máxima de balas que o superman conseguirá pegar.

Exemplo de entrada

Exemplo de entrada	Saída para o exemplo de entrada
5 5	3
1 1	2
3	
1 2 1	
2 3 2	
3 3 3	
5 5	
5 5	
4	
1 1 4	
2 4 2	
4 4 1	
2 3 3	
0 0	
