



# Problema A

## Atirando em Ouros

Um jogo comum em Faceland é "Atirando em Ouros". O jogo é feito em um campo  $(2*N+1) \times (2*N+1)$ . Antes do jogo começar, a CJF (Comissão dos Jogos de Faceland) posiciona pinos de boliche nesse campo, de forma que o formato final se assemelha à figura do naipe de Ouros, em um jogo de cartas. Por exemplo, se  $N = 4$ , a configuração do campo seria:

```

. . . . * . . . .
. . . * * * . . .
. . * * * * * . .
. * * * * * * .
* * * * * * *
* * * * * * *
. * * * * * * .
. . * * * * * . .
. . . * * * . . .
. . . . * . . . .

```

Onde . representa uma célula do campo vazia, e \* representa uma célula com um pino. Depois que a CJF termina de configurar o campo dessa maneira, eles cobrem todo o campo, de forma que nenhum jogador pode ver a configuração do campo durante o jogo. Só há um time, e todos os participantes jogam juntos contra a CJF. No início do jogo, a CJF define um número  $R$  de rodadas. A partir desse momento, a cada rodada, um jogador é chamado para escolher uma linha ou uma coluna do campo em que deseja jogar. Escolhido o movimento, ele pega uma bola de boliche e joga na linha ou coluna escolhida, derrubando todos os pinos no caminho, mas sem ver o que está acontecendo (pois o campo está coberto).

Ao final das  $R$  rodadas, os jogadores devem informar o número de pinos que ainda estão em pé, ou seja, que não foram derrubados. Como pode haver muitas rodadas, eles acabam se perdendo na conta, e a CJF acaba ganhando. Por isso, eles decidiram chamar você para ajudá-los nessa tarefa, e sempre responder corretamente a quantidade de pinos não derrubados ao final do jogo.

### Entrada

A entrada é composta por vários casos de teste. A primeira linha contém o número  $T$  de casos de teste ( $1 \leq T \leq 1050$ ). Cada caso de teste começa com dois inteiros  $N$  ( $1 \leq N \leq 2*10^9$ ) e  $R$  ( $0 \leq R \leq 10^6$ ), que foram descritos na questão. A seguir, são dispostas  $R$  linhas. Cada linha  $i$  contém dois inteiros,  $T_i$  e  $P_i$ .  $T_i$  é 0 quando a jogada é na linha, e 1 quando a jogada é na coluna.  $P_i$  ( $1 \leq P_i \leq 2*N+1$ ) representa a linha ou coluna em que o jogador jogou a bola de boliche.

*Atenção: em apenas cerca de 1% dos casos o  $R$  será maior que  $10^4$ .*

### Saída

Para cada caso de teste, imprima uma única linha contendo a resposta esperada, isto é, o número de pinos que continuam em pé após todas as rodadas.



Exemplo de Entrada	Exemplo de Saída
2 1 0 4 4 1 5 0 2 0 6 1 9	5 23

*Autor: Gustavo Stor*



## Problema B

### Bom Amigo Secreto

Está perto do Natal, e o time de Eventos da empresa Koobecaf resolveu parar de tentar descobrir a senha de <http://goo.gl/forms/xoy0AQ83fP> e criar um evento para o amigo secreto. Porém, o amigo secreto que eles jogam é um pouco diferente. Você sempre compra dois presentes. Depois que o sorteio é feito, a brincadeira original é realizada, mas com um detalhe: depois que alguém lhe dá um presente, você dá de volta um dos dois presentes que você comprou.

Assim, se há três pessoas A, B e C, e A tira B, B tira C e C tira A, então:

- A dará um presente a B
- B dará um presente de agradecimento a A, e dará outro presente a C
- C dará um presente de agradecimento a B, e dará outro presente a A
- A dará um presente de agradecimento a C

O time se estressa com o sorteio quando um ciclo se fecha que não inclua todo mundo. Eles só gostam da brincadeira quando, a partir do momento que a primeira pessoa é escolhida para dar seu presente, a sequência se segue até que ela seja a última a receber, sem que nenhuma outra pessoa fique de fora. Como eles estão bastante ocupados com a preparação do evento, chamaram você para dizer de quantas formas cada pessoa pode ter recebido presentes no final, dado que o sorteio é feito de forma que não estresse o time. Dizemos que duas formas são diferentes quando há duas pessoas, A e B, que trocam presentes em uma delas e na outra não. Por exemplo, a configuração em que A *tira B*, B *tira C* e C *tira A* é igual a C *tira B*, B *tira A* e A *tira C*.

#### Entrada

A entrada é composta por vários casos de teste. A primeira linha contém o número T de casos de teste ( $1 \leq T \leq 10^3$ ). A seguir, para cada caso de teste, haverá um inteiro N ( $2 \leq N \leq 10^3$ ), representando o número de pessoas que jogarão o bom amigo secreto esse ano.

#### Saída

Para cada caso de teste, imprima uma única linha contendo o número de configurações distintas. A resposta deve ser dada módulo  $10^9 + 7$  (1000000007).

Exemplo de Entrada	Exemplo de Saída
2	1
3	878320 <u>7</u> 16
1000	

*Autor: Gustavo Stor*



## Problema C

### Comida de Graça é Bom ou Ruim?

João é um funcionário de uma empresa que tem comida de graça durante todo o dia, e ele gosta muito de comer. Na verdade, comer, para João, já virou uma questão de necessidade quando ele está trabalhando. João acredita bastante que cada comida encontrada nas cozinhas da empresa lhe dá uma certa quantidade de energia, medida em minutos. Isso é, que a comida  $x$  lhe dá  $f(x)$  minutos de energia para trabalhar. João gostaria muito de ser um funcionário bastante produtivo, tendo energia para trabalhar bem durante todo o seu expediente (8 horas de trabalho, ou 480 minutos). Porém, João começou a se preocupar com a concorrência para arrumar acompanhantes no Vale do Silício, e não quer comer mais do que o necessário para se manter em forma.

#### Entrada

Na primeira linha, a quantidade de casos  $T$  ( $1 \leq T \leq 100$ ). Para cada caso, 2 linhas. Na primeira linha, o inteiro  $N$ , que representa a quantidade de comidas diferentes que há na empresa ( $1 \leq N \leq 1000$ ).

Na segunda linha,  $N$  números inteiros  $f(i)$  ( $1 \leq f(i) \leq 480$ ) que representam a quantidade de energia que a comida  $i$  dá para João. Como João é um cara que gosta de diversificar, ele só consome cada comida  $i$  no máximo uma vez.

#### Saída

Para cada caso, imprima uma única linha: o máximo que João pode produzir no seu dia de trabalho, em minutos, sem comer mais do que o necessário, isso é, sem acumular mais do que 480 minutos de energia.

Exemplo de Entrada	Exemplo de Saída
2	480
10	440
10 20 30 40 50 60 70 80 90 100	
3	
420 90 350	

*Autor: Mateus Moury*



# Problema D

## Done Is Better Than Perfect

Mateus, um estagiário de uma grande empresa de tecnologia, resolveu implementar uma aplicação muito perversa e sem a autorização do seu gerente. As 00:00, ele iria lançar uma notificação para todos os usuários da empresa dizendo que foram detectadas fotos que incitavam a violência em seus álbuns *online* de fotos, só porque ele queria deixar todo mundo preocupado. Mas Mateus cometeu um erro grave, e as coisas não aconteceram do jeito que ele queria. Ele esqueceu que a empresa em que trabalha tem usuários no mundo todo (menos na China) e que, portanto, só porque na Califórnia era meia-noite, isso não quer dizer que no resto do mundo também era. Em Londres, por exemplo, já era 8h da manhã.

O gerente de Mateus ficou furioso com ele. Não só porque ele fez uma aplicação estúpida, mas também porque ele não a fez de uma maneira correta. Então para penalizá-lo, o gerente falou: “Eu quero que, dado um certo horário e uma quantidade de minutos, você calcule dois novos horários: o dado horário somado à quantidade de minutos, e o dado horário subtraído da quantidade de minutos. Se você não fizer isso hoje, você está demitido”.

Como Mateus está muito ocupado tentando descobrir porque na China não se pode usar o Facebook, ele pediu para você resolver esse problema para ele.

### Entrada

Na primeira linha a quantidade de casos  $T$  ( $1 \leq T \leq 1000$ ). Para cada caso, uma linha: um horário no formato hh:mm ( $0 \leq hh < 24$ ,  $0 \leq mm < 60$ ) e uma quantidade de minutos  $x$  ( $0 \leq x < 1440$ ).

### Saída

Para cada caso, imprima uma linha com os dois horários requisitados pelo gerente: primeiro, o horário que é resultado da subtração dos minutos, e depois o horário que é resultado da soma dos minutos. Os horários devem ser impressos na forma hh:mm ( $0 \leq hh < 24$ ,  $0 \leq mm < 60$ ).

Exemplo de Entrada	Exemplo de Saída
2	04:33 20:55
00:44 1211	03:52 04:24
16:08 736	

*Autor: Mateus Moury*



# Problema E

## Estudo de Sequências

Stavistor é um curioso estudioso de sequências de inteiros. O seu site favorito é <http://oeis.org/>, a enciclopédia online de sequências de inteiros. Sua curiosidade despertou quando tinha apenas 6 anos, e foi introduzido à famosa sequência de Fibonacci: 1, 1, 2, 3, 5, 8, 13...

Anos se passaram e Stavistor foi conhecendo e inventando mais e mais sequências. Seus dias variavam desde brincar com as expansões decimais da proporção áurea, a tentar provar a infinitude dos números primos. Até que o dia chegou em que Stavistor já não tinha a mesma disposição de antes para descobrir novas sequências de inteiros, e resolveu colocar a fé de novas descobertas matemáticas em uma nova geração, começando pelo seu filho.

Para introduzir as sequências de inteiros ao seu filho de apenas 3 anos, Stavistor resolveu ensinar uma das mais básicas sequências, em que a diferença de números consecutivos é uma constante: progressões aritméticas. Para testar se o seu filho tinha realmente aprendido o conceito de progressões aritméticas, Stavistor escreveu uma sequência A de N inteiros e pediu para que seu filho descobrisse qual a maior subsequência contínua de A que também é uma progressão aritmética. Seu filho deveria responder o índice do começo e do fim dessa subsequência. Caso houvesse duas ou mais subsequências contínuas com o mesmo tamanho máximo, ele deveria responder a que começa mais cedo na sequência.

Esperto, o filho de Stavistor contratou um programador para descobrir a resposta: você.

### Entrada

Na primeira linha a quantidade de casos T ( $1 \leq T \leq 100$ ). Para cada caso, duas linhas. A primeira linha contém o inteiro N ( $1 \leq N \leq 10^5$ ), representando a quantidade de elementos na sequência. Na linha seguinte, N inteiros  $A_i$  ( $-10^9 \leq A_i \leq 10^9$ ) são apresentados, representando a sequência A.

### Saída

Para cada caso, imprima uma linha com o índice do começo e do fim da maior subsequência contínua de A que também é uma progressão aritmética.

Exemplo de Entrada	Exemplo de Saída
3	2 4
5	3 6
10 2 3 4 20	1 2
6	
1 2 3 10 17 24	
4	
1 2 10 11	

Autor: Gustavo Stor



## Problema F

### Falta Um Nome Para Essa Questão

Uma prática comum entre grandes empresas de tecnologia é a de organizar competições de programação, parecidas com a Maratona de Programação, envolvendo problemas de raciocínio lógico, matemática, geometria, grafos e afins. Essas competições atraem anualmente dezenas de milhares de competidores dos mais diversos níveis espalhados ao redor do mundo. Os melhores colocados são convidados a participar da grande final na sede das empresas, e um número maior de participantes (normalmente entre 100 e 1000) é premiado com camisas da competição.

Após a competição, a empresa precisa distribuir as camisas para os vencedores. Para fazer a distribuição, agrupam-se os vencedores de acordo com seus países de origem. Alguns países possuem conexões entre si, que podem ser utilizadas pagando-se um custo definido. O processo de distribuição inicia-se na sede da empresa nos Estados Unidos, e as conexões são utilizadas para alcançar os demais países.

Infelizmente, a empresa contratada para confeccionar as camisas só atendeu a demanda de  $L$  camisas, de forma que alguns vencedores podem não receber suas camisas. Como os funcionários da empresa estão muito ocupados sugerindo nomes para essa questão em <http://goo.gl/forms/ukPibqgQxP>, seu trabalho é ajudar a empresa organizadora da competição a gastar o mínimo possível no custo de conexões para conseguir entregar essas camisas a  $L$  vencedores.

#### Entrada

A primeira linha da entrada contém um inteiro  $T$  ( $1 \leq T \leq 100$ ), o número de casos de teste. A primeira linha de cada caso de teste começa com três inteiros  $N$  ( $2 \leq N \leq 8$ ),  $M$  ( $N - 1 \leq M \leq N * (N - 1) / 2$ ) e  $L$  ( $1 \leq L \leq 300$ ), contendo o número de países contendo vencedores de camisa, o número de conexões entre países e o número de camisas disponíveis, respectivamente. A linha seguinte contém  $N$  inteiros  $V_i$  ( $0 \leq V_i \leq L$ ), contendo o número de vencedores de cada país. As próximas  $M$  linhas contém 3 inteiros  $A B C$ , que indicam uma conexão entre países  $A$  e  $B$  com custo  $C$  ( $1 \leq C \leq 1000$ ). Os países são representados por índices entre 0 e  $N - 1$ , e os Estados Unidos, sede da empresa, sempre são representados pelo índice 0. Há no máximo uma conexão entre dois países, e sempre há uma forma de conectar (diretamente ou indiretamente) os Estados Unidos a qualquer outro país.

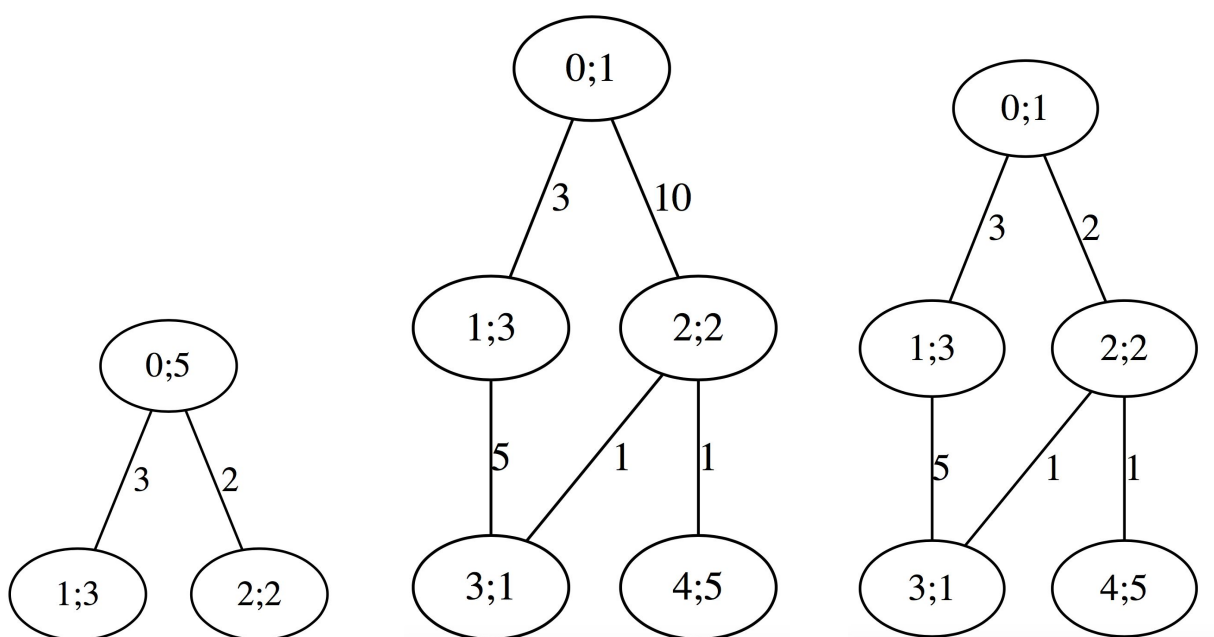
#### Saída

Para cada caso de teste imprima uma linha com o menor custo que é preciso pagar pra atender  $L$  dos vencedores de camisa. É garantido que sempre haverá uma resposta válida, ou seja, o número total de vencedores sempre será maior ou igual a  $L$ .

Exemplo de Entrada	Exemplo de Saída
3	3
3 2 8	9
5 3 2	6
0 1 3	
0 2 2	
5 5 6	
1 3 2 1 5	
0 1 3	



0 2 10	
1 3 5	
2 3 1	
2 4 1	
5 5 11	
1 3 2 1 5	
0 1 3	
0 2 2	
1 3 5	
2 3 1	
2 4 1	



*Ilustração dos grafos de entrada. Cada nó (país) está no formato  $(i; V_i)$*

*Autor: Lucas Lima*





## Problema G

### Grandes atalhos

Uma grande empresa de tecnologia percebeu que os usuários de seu chat poderiam alcançar uma maior velocidade de digitação se pudessem definir atalhos pessoais para palavras que utilizam bastante. Com o lançamento dessa nova feature, os usuários podem personalizar seu chat de forma que todas as vezes que eles escreverem  $*C$  (onde  $C$  é um caractere escolhido), a palavra  $A$  será colocada no lugar.

Dada uma string de entrada  $S$  e  $N$  atalhos da forma  $C \rightarrow A$ , onde  $C$  é um caractere e  $A$  é uma string, escreva um programa que imprime  $S'$ , uma string onde todas as ocorrências de  $*C$  em  $S$  são transformadas em  $A$ , para cada atalho.

#### Entrada

A primeira linha da entrada contém um inteiro  $T$  ( $1 \leq T \leq 100$ ), o número de casos de teste. Cada caso de teste começa com uma linha contendo  $S$  ( $1 \leq |S| \leq 10^5$ ), representando a string de entrada. A linha seguinte contém um inteiro  $N$  ( $0 \leq N \leq 26$ ), indicando o número de atalhos definidos. As  $N$  linhas seguintes contém os atalhos, da forma  $C A$ , onde  $C$  é um caractere e  $A$  é uma string ( $1 \leq |A| \leq 10$ ). As strings de entrada e dos atalhos são compostas apenas por letras maiúsculas e  $*$ , e todos os caracteres dos atalhos são letras maiúsculas. É garantido que cada caractere  $C$  aparecerá no máximo uma vez por caso de entrada.

#### Saída

Para cada caso de teste, imprima uma linha com a string resultante depois que os atalhos são aplicados à string de entrada.

Exemplo de Entrada	Exemplo de Saída
3	BBBBBAMARATONA
*A*ABA*C	*CB
2	*A
A BB	
C MARATONA	
**AB	
1	
A C	
*A	
0	

*Autor: Lucas Lima*



# Problema H

## High-primes

Lucas estava estudando matemática quando se deparou com as seguintes definições:

- Um número  $X$  é chamado de quadrado-perfeito quando sua raiz é um número inteiro.
- Um número  $Y$  é dito primo quando possui dois divisores positivos distintos, i.e. 1 e  $Y$ .
- Um número  $Z$  é chamado de high-prime quando ele é um quadrado-perfeito e a sua raiz é um número primo.

Ao ler isso, Lucas ficou se perguntando com que frequência ele poderia encontrar números high-primes caso considerasse apenas números em um dado intervalo  $[A,B]$ . Lucas está bastante ocupado no momento, você pode ajudá-lo?

### Entrada

A primeira linha da entrada contém  $T$  ( $1 \leq T \leq 10000$ ), o número de casos de teste. As próximas  $T$  linhas descrevem os casos de teste, cada caso de teste é composto por uma linha contendo dois inteiros:  $A$  e  $B$  ( $1 \leq A \leq B \leq 10^{12}$ ), que significam que Lucas só se interessa nos números do intervalo  $[A,B]$ .

### Saída

Para cada caso de teste imprima uma linha contendo um único inteiro, indicando quantos números high-primes existem no intervalo  $[A,B]$ .

Exemplo de Entrada	Exemplo de Saída
3	1
5 10	1
9 9	11
1 1000	

*Autor: Duhan Caraciolo*



# Problema I

Ih, errou de novo

Lucas está aprendendo a somar, mas ele ainda erra bastante, por isso ele quer que você o ajude. O problema é o seguinte: Lucas quer saber quantos pares  $(X,Y)$  existem tal que  $A \leq X \leq B$  e  $C \leq Y \leq D$ , e ainda  $X + Y > K$ .

## Entrada

A primeira linha da entrada contém  $T$  ( $1 \leq T \leq 30$ ), o número de casos de teste. As próximas  $T$  linhas descrevem os casos de teste, cada linha contém 5 inteiros:  $K A B C D$ , indicando os valores explicados anteriormente.

## Limites

$$-10^4 \leq A \leq B \leq 10^4$$

$$-10^9 \leq C \leq D \leq 10^9$$

$$-10^{15} \leq K \leq 10^{15}$$

## Saída

Para cada caso imprima uma linha com a resposta da questão.

Exemplo de Entrada	Exemplo de Saída
2	11
5 1 4 4 6	83367964786
100000000 -50 200 12341234 432143211	

Autor: Duhan Caraciolo

