



## XII MARATONA DE PROGRAMAÇÃO INTERNA UERJ - 18/10/2018

### Informações:

1. Usaremos o sistema URI Online Judge, da Universidade de Erechim, na maratona. Para se logar os alunos já devem estar cadastrados e devem usar a senha MaratonaXII para acesso à competição, que é privada.
2. Cada equipe utilizará um único micro, designado pela organização. APENAS um dos alunos da dupla deve se logar no sistema.
3. Pode ser consultado qualquer material escrito. Não se pode usar pen-drives, celulares, Internet na competição. Isso desclassifica a equipe.
4. Os problemas serão checados para verificar sua originalidade.
5. Durante a competição as duplas podem solicitar a ajuda da organização apenas para checar se o tratamento de entrada/saída está correto.
6. A pontuação e classificação é dada pelo número de problemas corretos com desempate na penalidade. A penalidade é a soma dos tempos de submissão dos problemas certos acrescida de 20 minutos por cada submissão errada.
7. Este caderno contém a descrição dos 8 problemas da maratona.
8. As linguagens permitidas são aquelas aceitas no URI: Python, Pascal, C, C++. JAVA, ...
9. Haverá duas categorias de equipes: Veteranos e Iniciantes.
10. Ao final da maratona será sorteado um brinde entre os participantes.
11. A premiação para os dois primeiros colocados em cada categoria será feita na sexta, 19/10, no encerramento da Semana do IME.

Boa sorte.

URI Online Judge | 1

# O Tasco é Sempre Vice!

Por Fabiano S. Oliveira, UERJ  Brazil**Timelimit: 1**

É bem conhecida a provocação que afirma que o Tasco, um dos melhores times de futebol do Brasil, é sempre vice-campeão das competições nas quais participa. Embora isso seja um elogio (estar em tal elevada colocação com regularidade sem dúvida é um mérito!), a mania do brasileiro de valorizar apenas a primeira colocação dá o efeito de provocação a esta afirmativa. É claro que isto não passa de mais uma das incontáveis brincadeiras que fazem parte do folclore futebolístico, mas Eurico, um fanático pelo Tasco, decidiu verificar se há algum fundamento nela de uma vez por todas, analisando o registro histórico de todas as competições em que o Tasco participou.

## Entrada

A primeira linha da entrada é composta do inteiro  $2 \leq N \leq 10^6$  que constitui o número de times. Cada time recebe um código distinto entre 1 e  $N$ , sendo o código 1 reservado para o Tasco. Em seguida, há dados de vários campeonatos. A primeira linha de cada campeonato é formada pelo inteiro  $1 \leq M \leq 10^5$ , que representa o número de partidas que ocorreram em tal campeonato. Cada uma das  $M$  linhas seguintes possui os dados de uma partida, no formato "**Time1 Placar1 Time2 Placar2**", onde  $1 \leq \text{Time1}, \text{Time2} \leq N$  são os códigos dos times que disputaram a partida e  $0 \leq \text{Placar1}, \text{Placar2} \leq 100$  são seus respectivos números de gols. A pontuação de cada time em um campeonato é a soma dos pontos obtidos em cada partida do campeonato. Em uma partida, a pontuação para o time que consegue a vitória é de 3 pontos, o empate é de 1 ponto e a derrota é de 0 pontos. Um time fica à frente de outro no campeonato se possui mais pontos ou, no caso de mesma pontuação, se possui uma maior quantidade de gols ou, em caso de mesma quantidade de gols, se seu código é menor do que o do outro. O fim da entrada de campeonatos se caracteriza pela leitura de um valor de  $M$  igual a 0.

## Saída

Em uma única linha, deve-se escrever "Y" se de fato o Tasco foi um dos times que foram mais vezes vice-campeão considerando-se todos os campeonatos, ou "N" caso contrário.

---

**Exemplos de Entrada****Exemplos de Saída**

---

Exemplos de Entrada	Exemplos de Saída
3 4 1 1 2 1 2 2 3 0 3 2 1 0 2 1 1 4 2 3 1 2 0 1 1 3 1 0	Y
2 1 1 1 2 1 1 2 1 1 0 2 1 1 2 0 2 2 1 2 0	N

URI Online Judge | 2

# Colorindo Tetraedros

Por Paulo E. D. Pinto, UERJ-Universidade do Estado do Rio de Janeiro  Brazil**Timelimit: 1**

Um designer inventou uma marca para uma empresa em forma de tetraedro. Ele tem várias cores disponíveis para pintar e quer saber de quantas maneiras diferentes o tetraedro pode ser colorido usando qualquer combinação de cores nas faces do mesmo. Note que, se mediante rotações convenientes, a coloração de dois tetraedros coincidir, então trata-se da mesma coloração.

Ajude esse designer a fazer esse cálculo.

## Entrada

Cada linha de entrada contém um inteiro  $N$ ,  $1 \leq N \leq 10^4$ , o número de cores disponíveis. A entrada termina com um valor 0, que não deve ser processado.

## Saída

Para cada entrada, imprimir o número de colorações distintas do tetraedro, com o número de cores dado. Como o resultado pode ser muito grande, apresentá-lo como o resto da divisão por 1000007.

Exemplo de Entrada	Exemplo de Saída
1	1
8	400
2250	878008
0	

# Um Problema com Final Feliz

Por Paulo E. D. Pinto, UERJ- Universidade do Estado do Rio de Janeiro  Brazil

**Timelimit: 1**

Paul Erdős foi um notável matemático húngaro que publicou 1475 artigos em diversas áreas. Ele deu o nome de Teorema do Final Feliz ao seguinte teorema provado por ele e alguns amigos:

"Dados 5 pontos no plano, não colineares três a três, quatro deles sempre formam um quadrilátero convexo".

O nome do teorema deve-se ao fato de que o casal amigo que trabalhou no mesmo acabou se casando. Muitos matemáticos trabalharam na generalização desse problema, o que ainda não foi conseguido. Neste problema você vai constatar o teorema e, esperamos, com total felicidade! São dados 5 pontos, não colineares três a três, e você deve encontrar a área do maior quadrilátero convexo envolvendo 4 dos pontos dados.

## Entrada

Cada linha de entrada contém 10 inteiros pares  $X_1, Y_1, X_2, Y_2, X_3, Y_3, X_4, Y_4, X_5, Y_5$ ,  $0 \leq X_i, Y_i \leq 10^3$ ,  $1 \leq i \leq 5$ , onde cada par  $X_i, Y_i$  representa as coordenadas de um dos 5 pontos. A entrada termina com 10 valores 0, que não devem ser processados.

## Saída

Para cada entrada, imprima a área do maior quadrilátero convexo que pode ser formado com 4 dos 5 pontos dados.

Exemplo de Entrada	Exemplo de Saída
0 100 0 0 100 100 50 90 100 0	10000
20 10 30 50 80 20 10 70 40 30	800
0 0 0 0 0 0 0 0 0 0	

URI Online Judge | 4

# Treino No Velódromo

Por Paulo E. D. Pinto, UERJ-Universidade do Estado do Rio de Janeiro  Brazil**Timelimit: 1**

Três ciclistas estão treinando no velódromo para a próxima Olimpíada. Eles dão a partida juntos e fazem milhares de voltas a cada dia, de forma regular. Cada um leva dado tempo para completar a volta e corre sempre com a mesma velocidade.

O técnico anotou o tempo de uma volta para os dois primeiros ciclistas e só sabe, em relação ao terceiro o tempo que leva para os três se alinharem novamente na linha de partida.

Você vai ajudar o técnico calculando todos os possíveis tempos que o terceiro ciclista leva para uma volta.

## Entrada

Cada entrada contém três inteiros:  $T$ ,  $1 \leq T \leq 10^6$ , o tempo que os ciclistas levam para se encontrar novamente na linha de partida,  $A, B$ ,  $1 \leq A, B \leq 10^2$ , os tempos respectivos que os ciclistas 1 e 2 levam para dar uma volta. A entrada termina com 0 0 0, que não devem ser processados.

## Saída

Para cada entrada, imprimir, de forma ordenada, os possíveis tempos que o terceiro ciclista leva para completar uma volta, de forma a ocorrer a coincidência mencionada.

Exemplo de Entrada	Exemplo de Saída
42 6 7	1 2 3 6 7 14 21 42
40 5 4	8 40
58652 11 62	172 1892 5332 58652
0 0 0	

URI Online Judge | 5

# $\text{Fibonacci}^k$

Por Fabiano S. Oliveira, UERJ  Brazil**Timelimit: 1**

Néchi é um garoto muito esperto. Ele percebeu logo nos primeiros anos de sua vida acadêmica que é uma boa ideia procurar padrões contidos na descrição dos problemas para memorizá-los. É curioso observar a maneira como ele memorizou a sequência de Fibonacci; ele pensou assim: seja  $k$  o número de letras "c" contidas em "Fibonacci". Inicie a sequência por  $0, 1, \dots, k-1$  e obtenha o próximo elemento da sequência pela soma dos  $k$  elementos imediatamente anteriores. Elabore um programa para computar um elemento arbitrário da sequência de  $\text{Fibonacci}^k$ , como Néchi gosta de chamar o problema.

## Entrada

A primeira linha da entrada é composta do inteiro  $1 \leq T \leq 10$  que constitui o número de casos de teste. Cada uma das próximas  $T$  linhas consistem dos casos de teste, que vem um por linha. Um caso de teste é composto por um inteiro  $2 \leq k \leq 10^5$  seguido de um inteiro  $1 \leq N \leq 2 \times 10^5$ .

## Saída

Deve-se escrever uma linha por caso de teste com o valor do resto da divisão do  $N$ -ésimo elemento da sequência de  $\text{Fibonacci}^k$  por **1000007**.

Exemplo de Entrada	Exemplo de Saída
5	3
2 5	4
7 5	83
7 10	56310
8 64	851025
10000 20000	

URI Online Judge | 6

# Virus

Por Paulo E. D. Pinto, UERJ- Universidade do Estado do Rio de Janeiro  Brazil**Timelimit: 1**

Estamos em um país da África, onde um vírus perigosíssimo acaba de chegar em uma grande cidade. O país vai ter que tomar medidas extremas para que ele não se propague. A única saída será bloquear estradas que chegam nas cidades infectadas. Mas o país só tem dinheiro para bloquear um único trecho de estrada. Nesse país existe sempre pelo menos um caminho entre duas cidades. A Vigilância Sanitária quer descobrir se existe um trecho de estrada que, quando bloqueado, separa o resto do país da área infectada.

Você vai ajudar, fazendo um programa para identificar se existe um tal trecho e indicar qual o número máximo de cidades que podem ser isoladas do vírus. Observe que pode haver mais de um trecho de estrada entre duas cidades.

## Entrada

A primeira linha da entrada é um inteiro  $T$  ( $1 \leq T \leq 100$ ) que indica quantos serão os casos de teste. A seguir são descritos  $T$  testes. Cada teste é descrito em várias linhas. A primeira contém 3 inteiros:  $N$  ( $1 \leq N \leq 1000$ ), o número de cidades do país,  $M$  ( $N-1 \leq M \leq 10000$ ), o número de interligações entre as cidades e  $C$  ( $1 \leq C \leq N$ ), o número da cidade infectada. A seguir vêm  $M$  linhas com 2 inteiros indicando todos os pares de cidades interligadas. As cidades são numeradas de 1 a  $N$ .

## Saída

Para cada caso de teste imprima o número máximo de cidades que podem ser isoladas com o bloqueio de um trecho de estrada. Se não houver um trecho com as características procuradas, responda 0.

---

**Exemplo de Entrada**

---

**Exemplo de Saída**

---



**Exemplo de Entrada****Exemplo de Saída**

```
2
9 10 3
1 3
4 2
3 4
5 3
4 5
6 5
7 6
6 5
8 7
8 9
4 4 1
1 2
2 3
3 4
1 4
```

```
3
0
```

# Velhas Balanças

Por Paulo E. D. Pinto, UERJ- Universidade do Estado do Rio de Janeiro  Brazil

**Timelimit: 1**

Velhas balanças de dois pratos ainda podem ser encontradas em feiras, laboratórios e museus. Em um dos pratos coloca-se o objeto a ser pesado e, no outro, os pesos, até equilibrar a balança. Somam-se os pesos e obtém-se o peso do objeto. Mas também podem ser colocados alguns pesos junto com o objeto a ser pesado e, nesse caso, esses pesos têm que ser subtraídos da soma de pesos do lado oposto. Por exemplo, um objeto de peso 15 pode ser pesado colocando, do outro lado da balança, os pesos 8, 4, 2, 1, pois  $8+4+2+1 = 15$ . Mas também pode ser pesado colocando-se o peso 16 do lado oposto e o peso 1 junto com o objeto, pois  $16-1 = 15$ .

Neste problema, os pesos são todos as potências de 2 e há somente um peso de cada valor. Você deve descobrir o esquema que pesa dado objeto com peso inteiro, usando o menor número de pesos do tipo descrito.

## Entrada

A primeira linha de entrada contém inteiro  $T$  ( $1 \leq T \leq 10000$ ), que indica o número de casos de teste. Em cada uma das  $T$  linhas a seguir vem um inteiro  $N$  ( $1 \leq N \leq 2048$ ), indicando o peso a ser usado.

## Saída

Para cada caso de teste imprima o número mínimo de pesos necessário para pesar o objeto dado.

Exemplo de Entrada	Exemplo de Saída
4	1
1	2
10	3
100	3
1000	

URI Online Judge | 8

# Aproveite a Oferta

Por Paulo E. D. Pinto, UERJ- Universidade do Estado do Rio de Janeiro  Brazil**Timelimit: 1**

Um supermercado está fazendo uma promoção de venda de refrigerantes. Se um dia você comprar refrigerantes e levar os cascos vazios no dia seguinte, ela troca cada conjunto de  $K$  garrafas vazias por uma garrafa cheia. Um cliente quer aproveitar ao máximo essa oferta e por isso comprou várias garrafas no primeiro dia da promoção. Agora ele quer saber quantas garrafas terá ao final do segundo dia da promoção, se usá-la ao máximo.

Faça um programa para calcular isso.

## Entrada

A primeira linha de entrada contém inteiro  $T$  ( $1 \leq T \leq 10000$ ), que indica o número de casos de teste. Em cada uma das  $T$  linhas a seguir vêm dois inteiros  $N$  e  $K$  ( $1 \leq K, N \leq 10000$ ), respectivamente o número de refrigerantes comprados e o número de garrafas vazias para ganhar uma cheia.

## Saída

Para cada caso de teste imprima o número de garrafas que o cliente terá no segundo dia, se aproveitar ao máximo a oferta.

Exemplo de Entrada	Exemplo de Saída
3	4
7 4	4
4 7	574
4000 7	