

414 - Encontro das Superfícies – 92,3%

Um dispositivo de aquisição de imagens provê imagens digitais de duas partes de superfície que eventualmente se juntarão ao entrar em contato uma com a outra. O momento deste contato final deverá ser estimado.

A imagem digital será composta por dois caracteres, “X” e “ ” (espaço em branco). Haverá sempre 25 colunas em cada imagem, mas o número de linhas, N, é variável. A primeira coluna (1) sempre terá um “X” e será parte da superfície a esquerda. A superfície a esquerda pode ser estendida a direita a partir da coluna 1 através de Xs contíguos.

Similarmente, a coluna 25 sempre terá um “X” e será parte da superfície a direita. A superfície a direita pode ser estendida para a esquerda através de X's contíguos.

A seguir a visão das superfícies através desse tipo de imagem digital.

Esquerda		Direita
XXXX		XXXXX ← 1
XXX		XXXXXXXX
XXXXX		XXXX
XX		XXXXXX
.		.
.		.
.		.
XXXX		XXXX
XXX		XXXXXXXX ← N
↑		↑
1		25

Em cada linha da imagem, haverá zero ou mais caracteres separando a superfície da esquerda da superfície da direita. Nunca haverá mais de uma região com espaços em branco por linha.

Para cada imagem dada, você deverá determinar o total de espaço vazio que existirá após a superfície a esquerda tocar a superfície a direita. O espaço vazio é o total dos espaços que permanece em branco entre as superfícies da esquerda e da direita após elas entrarem em contato.

As duas superfícies entrarão em contato através de movimentos estritamente na horizontal até que o X mais a direita de alguma linha da superfície da esquerda toque o X mais a esquerda daquela linha da superfície da direita. Não haverá nenhum tipo de rotação destas superfícies quando elas se tocarem; elas permanecerão rígidas e apenas se moverão horizontalmente.

Note: A imagem original pode apresentar duas superfícies que já estejam em contato, neste caso nenhum deslocamento será necessário para se calcular os espaços em branco.

Entrada

A entrada consistirá de uma série de imagens digitais. Cada imagem terá o seguinte formato:

Primeira linha -

Um inteiro positivo e menor que 13. Este valor será N.

N linhas seguintes -

Cada linha terá exatamente 25 caracteres; um ou mais X's, então zero ou mais espaços em branco; e por fim um ou mais X's.

O final do conjunto de dados é sinalizado por um conjunto em branco, tendo zero como valor de N e nenhum dado adicional.

Saída

Para cada imagem que você receberá como conjunto de entrada, você deverá retornar o número de espaços em branco restantes na imagem após as duas superfícies se tocarem. Imprima o total de espaços em branco (um valor por imagem de entrada).

Exemplo de Entrada

```
4
XXXX                XXXXX
XXX                 XXXXXXXX
XXXXX              XXXX
XX                 XXXXXXXX
2
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
1
XXXXXXXXXX        XX
0
```

Exemplo de Saída

```
4
0
0
```

573 – A Lesma – 90,7%

Uma lesma está na base (fundo) de um poço e deseja escalá-lo até o topo. A lesma pode escalar 3 pés durante o dia, mas desliza um pé enquanto dorme. A lesma tem um fator de fadiga de 10%, significando que a cada dia consecutivo de escalada ela escala $10\% \times 3 = 0,3$ pés a menos do que no dia anterior (os 10% são sempre calculados em relação à altura que ela consegue escalar no primeiro dia). Em que dia a lesma conseguirá escalar o poço? Por exemplo, quando a lesma conseguirá passar por um poço de 6 pés de altura? (um dia consiste de um período de sol mais o período da noite). Como pode ser visto na tabela a seguir, a lesma passa dos 6 pés durante o terceiro dia.

Dia	Altura Inicial	Distância Percorrida	Altura após Escalada
1	0'	3'	3'
2	2'	2.7'	4.7'
3	3.7'	2.4'	6.1'

Seu trabalho é resolver o problema geral de escalada da lesma. Dependendo dos parâmetros, a lesma conseguirá escalar o poço ou irá, eventualmente, escorregar até a base do poço. Isto é, a altura atingida pela lesma irá superar a altura do poço ou irá cair até ficar negativa. Você deverá identificar o que ocorrerá primeiro e em que dia.

Entrada

A entrada consistirá de um ou mais casos de teste. Cada caso em uma linha. Cada linha conterà quatro inteiros H, U, D e F, separados por um único espaço em branco. A entrada será encerrada com H recebendo o valor zero. Caso contrário, todos os números receberão valores entre 1 e 100. H é a altura do poço em pés, U é a distância em pés que a lesma consegue escalar durante o dia, D é a distância em pés que a lesma escorrega durante a noite, e F é o fator de fadiga expresso em porcentagem. A lesma nunca escala uma distância negativa. Se o fator de fadiga levar a distância escalada para um valor negativo então a lesma não escalará nada durante o dia. Independente do quanto a lesma escale em um dado dia, ela sempre escorregará D pés durante a noite.

Saída

Para cada caso de teste, imprima uma linha indicando se a lesma conseguiu deixar o poço ou falhou (escorregou de volta para o fundo) e em que dia isso aconteceu. O formato deve ser idêntico ao dos exemplos a seguir.

Exemplo de Entrada

```
6 3 1 10
10 2 1 50
50 5 3 14
50 6 4 1
50 6 3 1
1 1 1 1
0 0 0 0
```

Exemplo de Saída

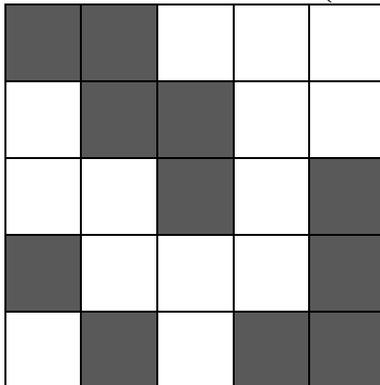
```
success on day 3
failure on day 4
failure on day 7
failure on day 68
success on day 20
failure on day 2
```

871 – Contando Células na Bolha – 90,8%

Conhecimento Geral

Considere um grid bidimensional de células, cada uma podendo estar vazia ou preenchida. Células preenchidas formam bolhas. Células preenchidas adjacentes formam uma bolha maior. Duas células são consideradas adjacentes se elas forem conectadas na horizontal, vertical ou diagonal. Poderá haver diversas bolhas no grid. Seu trabalho é encontrar a maior bolha (em termos de número de células) no grid.

A seguinte figura ilustra um grid com três bolhas (a maior contém cinco células)



Problema:

Escreva um programa que determine o tamanho da maior bolha para um dado conjunto de bolhas.

Entrada:

A entrada iniciará com um inteiro positivo indicando o número de casos de teste. Cada caso de teste é descrito a seguir. A linha inicial é seguida por uma linha em branco e também haverá uma linha em branco entre dois casos de teste consecutivos.

O grid é passado como um conjunto de strings, cada uma composta por zeros e uns. O valor 1 indica que a célula está preenchida e o valor 0 indica que a célula está vazia. O maior grid possível terá tamanho 25 x 25.

Saída

Para cada caso de teste, imprima o tamanho da maior bolha. Imprima uma linha em branco entre dois casos de teste.

Exemplo de Entrada

1

11000

01100

00101

10001

01011

Exemplo de Saída

5

872 – Ordenação – 92%

Ordenação é um conceito importante na matemática e ciência da computação. Por exemplo, o Lema de Zorns diz: um conjunto parcialmente ordenado no qual cada cadeia tem um limitante superior contém um elemento maximal.

O problema atual não envolve o Lema de Zorns nem semântica de ponto fixo, mas envolve ordenação.

Dada uma lista de restrições sobre variáveis do tipo $A < B$, você deverá escrever um programa que imprime todas as ordenações das variáveis que são consistentes com as restrições. Por exemplo, dadas as restrições $A < B$ e $A < C$, há duas ordenações das variáveis A, B e C possíveis que são consistentes com estas restrições: ABC e ACB.

Entrada

A entrada inicia com um único inteiro na primeira linha indicando o número de casos de teste. Cada caso é descrito como segue. Haverá uma linha em branco e também haverá uma linha em branco entre dois casos de teste.

Cada caso de teste terá duas linhas: uma lista de variáveis em uma linha seguida por uma linha com a lista de restrições do tipo $A < B$. Variáveis e restrições são separadas por um único espaço em branco.

Cada variável é representada por uma única letra maiúscula. Haverá ao menos duas variáveis e no máximo 20 variáveis. Não haverá mais do que 300 ordenações consistentes com as restrições em cada caso de teste.

Saída

Para cada caso de teste, a saída deve seguir a descrição abaixo. A saída de dois casos de teste seguidos deverá ser separada por uma linha em branco.

Todas as ordenações consistentes com as restrições devem ser impressas. As ordenações devem ser impressas em ordem alfabética, uma por linha. Caracteres em uma linha devem ser separados por um espaço em branco. Não imprima espaços em branco no final das linhas. Se não houver ordenação possível, imprima uma linha contendo apenas a palavra "NO".

Exemplo de Entrada

```
1  
  
A B F G  
A<B B<F
```

Exemplo de Saída

```
A B F G  
A B G F  
A G B F  
G A B F
```

1112 – Os Ratos e o Labirinto – 90,5%

Um conjunto de ratos de laboratório está sendo treinado para escapar de um labirinto. O labirinto é composto de células, e cada célula é conectada a algumas outras células. No entanto, existem obstáculos na passagem entre as células e, portanto, há uma penalidade de tempo para superar a passagem. Além disso, algumas passagens permitem aos ratos ir em uma direção, mas não o contrário.

Suponha que todos os ratos estão treinados e, quando colocados em uma célula arbitrária no labirinto, eles tomam o caminho que os leva para a célula de saída no tempo mínimo.

Vamos realizar a seguinte experiência: um rato é colocado em cada célula do labirinto e um temporizador de contagem regressiva é iniciado. Quando o temporizador para, contamos o número de ratos fora do labirinto.

Escreva um programa que, dada uma descrição do labirinto e o limite de tempo, prevê o número de ratos que sairão do labirinto. Suponha que não há estrangulamentos no labirinto, isto é, que todas as células têm espaço para um número arbitrário de ratos.

Entrada

A entrada começa com um único inteiro positivo na primeira linha, indicando o número de casos de teste, seguida por uma linha em branco. A seguir cada caso de teste é descrito. Entre dois casos de teste haverá uma linha em branco. As células do labirinto são numeradas de 1 a N , onde N é o número total de células (você pode assumir que N nunca será maior que 100). A primeira linha de um caso de teste contém o valor de N , o número de células do labirinto, a segunda contém E , o número da célula de saída do labirinto, e a terceira o valor T (valor inicial do temporizador). A quarta linha contém o número M de conexões no labirinto, e é seguida por M linhas, cada uma especificando uma conexão com três inteiros: o número das células a e b (entre 1 e N) e o tempo necessário para se viajar de a para b . Note que cada conexão é “mão única”, isto é, o rato só poderá viajar de b para a se houver outra conexão ligando b a a , neste caso é possível que o tempo para ir de a para b seja diferente daquele de se ir de b para a .

Saída

Para cada caso de teste, a saída deve ser de acordo com a seguinte descrição. As saídas de dois casos de teste consecutivos devem ser separadas por uma linha em branco. A saída consiste de uma única linha contendo o número de ratos que atingirão a célula de saída E em no máximo T unidades de tempo.

Exemplo de Entrada

```
1
4
2
1
8
1 2 1
1 3 1
2 1 1
2 4 1
3 1 1
3 4 1
4 2 1
4 3 1
```

Exemplo de Saída

```
3
```

1645 - Contagem – 94,5%

O Prof. Tigris é o chefe de uma equipe arqueológica que atualmente está encarregada de uma escavação em um local de relíquias antigas.

Este local contém as relíquias de uma vila onde uma civilização uma vez floresceu. Uma noite, examinando um registro escrito, você encontra um texto significativo para você. Diz o seguinte:

“Nossa aldeia é de glória e harmonia. Nossos relacionamentos são construídos de tal forma que todos, exceto o chefe da aldeia têm exatamente um chefe direto e ninguém será o chefe de si mesmo, o chefe do seu próprio chefe, etc. Todos exceto o chefe da aldeia é considerado como subordinado do seu chefe. Chamamos isso de configuração de relacionamento. O chefe da aldeia está no nível 0, seus subordinados estão no nível 1, e os subordinados de seus subordinados estão no nível 2, etc. Nossa configuração de relacionamento é harmoniosa porque todas as pessoas no mesmo nível têm o mesmo número de subordinados. Portanto nossa relação é ...”

O registro termina aqui. O Prof. Tigris agora se pergunta quantas diferentes configurações de relação harmoniosa podem existir. Ele só se preocupa com a forma holística da configuração, então duas configurações são consideradas idênticas se e somente se houver uma bijeção de n pessoas que transforma uma configuração em outra.

Por favor, veja as ilustrações abaixo para a explicação quando $n = 2$ e $n = 4$.

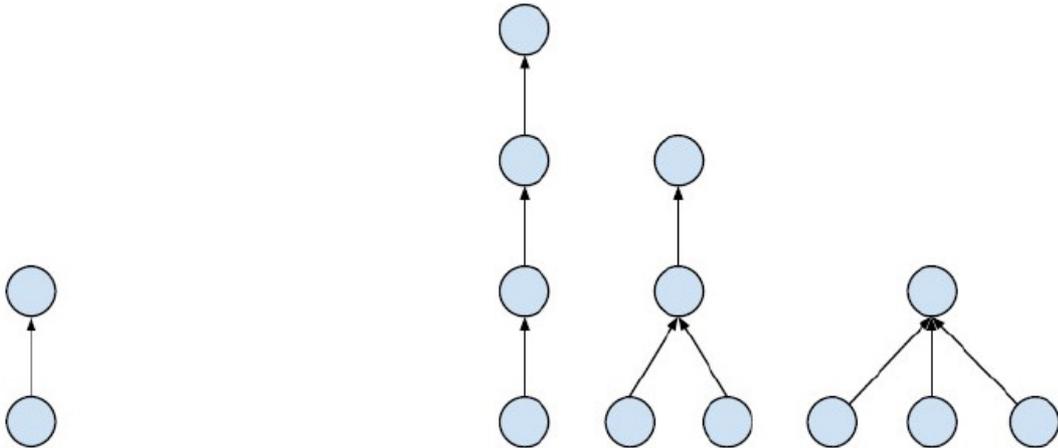


Figure 1: $n = 2$

Figure 2: $n = 4$

O resultado pode ser muito grande, então você deve tomar a operação de módulo, com o módulo de $10^9 + 7$ antes de imprimir sua resposta.

Entrada

Haverá vários casos de teste. Para cada caso de teste haverá uma única linha contendo um inteiro $1 \leq n \leq 1000$. A entrada será encerrada pelo fim de arquivo.

Saída

Para cada caso de teste imprima uma linha contendo: “Case X: Y”, sendo X o número do caso de teste começado por 1 e Y a resposta desejada.

Exemplo de Entrada

1
2
3
40
50
600
700

Exemplo de Saída

Case 1: 1
Case 2: 1
Case 3: 2
Case 4: 924
Case 5: 1998
Case 6: 315478277
Case 7: 825219749