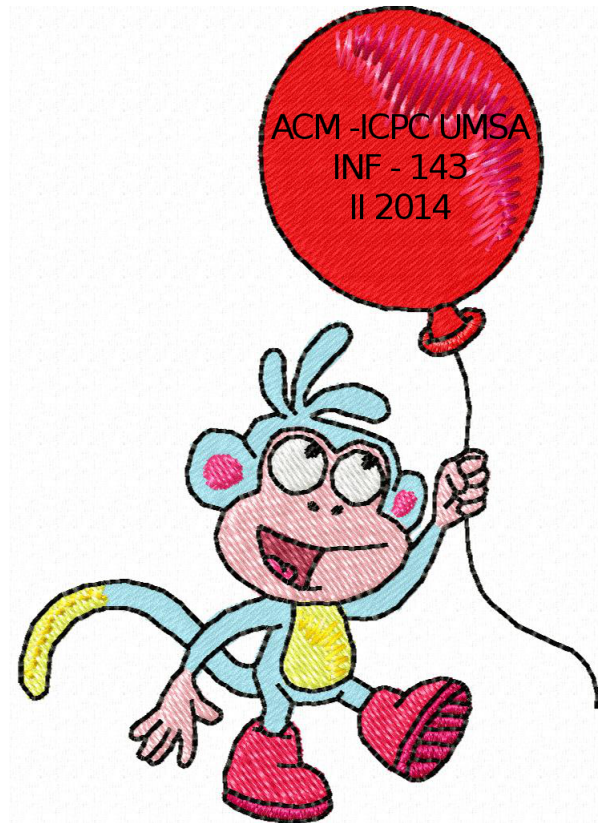


Universidad Mayor de San Andrés
Competencia de Programación de INF-143 II
2014



Competencia de Programación de INF-143 II-2014

29 de noviembre de 2014

Las páginas están numeradas desde el 1 al 16 .

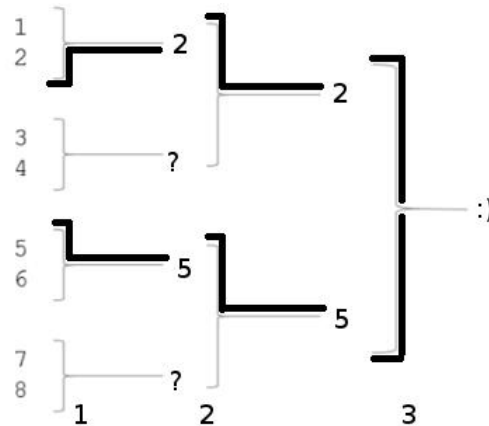
Índice

Al reality show	1
Botas no eres picasso	2
Ciencias de la Computación	4
Don botas y sus robots	5
El niño mimado	7
Foquitos	8
Girl	10
Homero	11
I Love You XOR	12
Jackiado	13
Kuadragonia	14
Loco por el uno	15
Modulo	16

Problema A

Al reality show

Una vacación Willmar decidió hacer un Reality Show con los concursantes de la ICPC, y los reunió a todos en una casa en donde a los competidores con números de 1 a N , cada viernes ocurre una ronda clasificatoria, donde se enfrentan los programadores en “un épico srm de topcoder” 1vs1. (El programador 1 se enfrenta con el 2, el 3 con el 4, el 5 con el 6... y así sucesivamente) El perdedor de cada *vs* debe retirarse del Reality show. :(Botas uno de los competidores tiene el sueño de concursar con Rolo porque ellos dos comenzaron a programar juntos, y además piensa que es el más fácil de eliminar en el Reality show. El ahora quiere saber en qué número de ronda se enfrentaría con Rolo. Puedes ayudar a Botas a saber en qué ronda se enfrentará con Rolo? Por ejemplo si fuera un reality show de 8 participantes y Botas fuera el participante 2 y Rolo el 5 se encontrarían en la ronda 3. (ver el gráfico).



Input

La entrada consiste de un número R (el número de rondas del concurso, cada ronda tiene $2R$ participantes) ($1 \leq R \leq 60$) y dos números a, b que son los números Willmar le asignó a Botas y Rolo. ($1 \leq a, b \leq 2^R$).

Output

Se debe imprimir el número de ronda en la que se enfrentaran Rolo y Botas.

Ejemplos de entrada	Ejemplos de salida
3 2 5	3

Problema B

Botas no eres picasso

Botas es un muchacho que le gusta mucho pintar su cuaderno, el pinta pequeñas celdas de su cuaderno llenándolo con un color hasta formar una región $N * N$ de diferentes colores. Ahora botas considera una "frecuencia" de un color como el número de veces que aparece dicho color. A botas le gustaría mucho hacer que todas las frecuencia de cada color sean lo menos diferente posible, para hacer esto el hará una sola operación que sera el cambio de color por otro color diferente que haya en la región de colores, pero considere que botas no cambiara un color que ya fue cambiado pues el considera que es una perdida de tiempo. Ahora mientras botas estaba haciendo esto, sus amigos vinieron y le dijeron "vamos a jugar dota" y a botas le gusta mucho jugar dota por lo que el quiere ir con ellos, pero por otro lado el quiere terminar esto, pero como no tiene mucho tiempo el quiere hacerlo rápido con el menor número de cambios de color, pero botas no trajo su computadora así que pide su ayuda para hacer esto. Así que si quiere que botas vaya tranquilo a jugar debe ayudarlo. A Botas no le gusta prestar su cuaderno por lo que él solo le dará el número de veces con el que aparece cada color pero como botas maneja solo 10 colores, entonces el los denotara con números entre 0-9. SE considera una menor diferencia si la diferencia de dos colores es de 0 o 1. Ejemplo:

frecuencia	1 4 4 2 1 2 1 0 0 1
color	0 1 2 3 4 5 6 7 8 9

Botas no quiere que cuenten los 0 pues el le da a entender que si hay 0 es por que no aparece en la región. Ahora el hará 4 movimientos para que las frecuencias queden lo menos diferente posible quedando así:

frecuencia	2 2 2 2 2 2 2 0 0 2
color	0 1 2 3 4 5 6 7 8 9

Como observa lo menos diferente que puede ser es 0.

Input

La primera linea sera T que simboliza el número de regiones. Las siguientes T lineas serán las frecuencias con las que aparecen los 10 colores denotadas por 10 enteros F_i donde ($0 \leq F_i \leq 10000$).

Output

Ahora como botas es un muchacho muy ordenado el quiere que muestre tanto la dimensión de su región y el mínimo número de cambios que tendrá que hacer para que su región tenga sus frecuencias lo menos distinto posible.

Ejemplos de entrada	Ejemplos de salida
2	4 4
1 4 4 2 1 2 1 0 0 1	4 3
0 0 4 5 1 6 0 0 0 0	

Problema C

Ciencias de la Computación

Botas estudia Ciencias de la Computación y le quedan por cursar n materias. Él sabe que si hace una sola materia le va a costar 1 unidad de esfuerzo aprobarla, pero si hace una segunda materia le cuesta 2 unidades de esfuerzo para aprobar la segunda materia, y en general, si hace i materias le cuesta i unidades de esfuerzo aprobar la i -ésima materia. Él sabe que hacer un esfuerzo de E unidades o más lo va a estresar y por lo tanto va a dejar la carrera. Cuántas materias como máximo puede tomar por semestre?

Input

La entrada esta compuesta por un numero T ($1 \leq T \leq 10^5$) el numero de casos de prueba, seguido por T lineas compuestas por dos numeros N y E ($1 \leq N \leq 4 * 10000$) ($1 \leq E \leq 1000000000$), el numero de materias y esfuerzo maximo respectivamente.

Output

Por cada caso de prueba imprimir una linea que contiene el numero máximo de materias que puede tomar en un semestre.

Ejemplos de entrada	Ejemplos de salida
2	3
10 10	4
10 11	

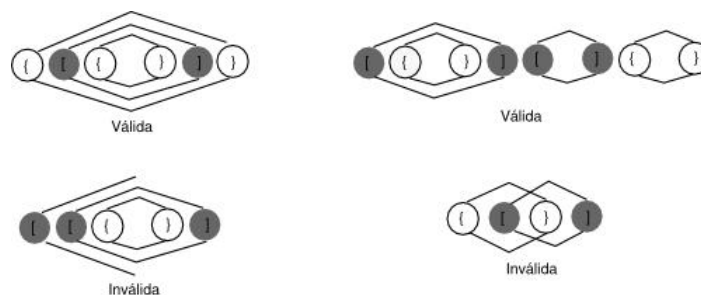
Problema D

Don botas y sus robots

Los gatos de nuestro amigo Botas desaparecieron, Botas se encuentra muy triste y para llenar ese vacío que sus gatos dejaron, Botas decidió aprender electrónica y comenzó a crear robots que lo acompañen en su soledad, lo interesante de estos robots además de ser autónomos, era la capacidad que tenían para aumentar la longitud de sus brazos a su antojo. Después de unos días Botas se dio cuenta que había creado demasiados robots. Al ver esto, decidió que estos robots compitan. Botas catalogó a los robots en dos tipos (tipo A y tipo B). La idea de este concurso es que los robots estén emparejados (es decir dos robots se toman de los brazos). Las reglas de emparejamiento son las siguientes:

- Un emparejamiento debe ser de dos robots del mismo tipo
- Dentro de un emparejamiento válido pueden existir varios emparejamientos que además deben ser válidos (esto es posible por la capacidad que tienen los robots de aumentar la longitud de sus brazos)
- Dos emparejamientos no pueden entrelazarse
- Todo robot debe tener una pareja

Después que los robots buscan la mejor forma de emparejarse, estos llegan a un estado final y Botas debe verificar si este estado cumple con las reglas de emparejamiento. Como Botas sigue en segundo semestre, aún no conoce algunos trucos de programación, es por eso que debes ayudar a Botas creando un programa que dado el estado final de los robots, ayude a Botas a ver si dicho estado es válido o no. Algunos ejemplos de estados válidos e inválidos son:



Input

La primera línea contiene un número T ($1 \leq T \leq 50$) que es la cantidad de casos. Cada caso comienza con un número N ($1 \leq N \leq 100$) que es la cantidad de robots, luego viene una cadena S de tamaño N que representa el estado final de los robots, cada carácter representa a un robot, los robots de tipo A están representados por los caracteres '[', '[' y los robots del tipo B están representados por ']', ']'.

Ouput

Por cada caso, se debe imprimir “V” si el estado es válido y “NV” si no lo es.

Ejemplos de entrada	Ejemplos de salida
4	V
6	NV
{[{}]}	V
5	NV
[[{}]	
8	
[{}] [] {}	
4	
{[]}	

Problema E

El niño mimado

Botas tiene un montón de juguetes de diferentes tamaños. Cuando se acerca la hora de acostarse, es muy cuidadoso: los coloca en fila, y se pone a recogerlos en orden, metiéndolos uno a uno en varios baúles, que tiene también colocados numerados junto a la pared. Cuando el siguiente juguete de la fila no entra en el baúl que está llenando (aunque aún tenga mucho espacio), lo cierra hasta el día siguiente, y mete el juguete que tiene en la mano en un baúl vacío. Nunca abre un baúl que ya ha cerrado, nunca parte un juguete para meter un trozo en un baúl y otro en otro, nunca desordena los juguetes de la fila, y siempre llena los baúles en orden. Sus padres, que lo creen un poco raro, no quieren quitarle esa extraña costumbre, porque les parece chistosa. Además, ¡es una suerte que sea él mismo quién recoge su habitación! Sin embargo, los baúles están un poco viejos, tienen esquinas y pesan mucho, por lo que temen que se pueda hacer daño al cerrarlos o si se golpea con ellos. Han decidido comprarle varias cajas nuevas, del mínimo tamaño posible para que todos sus juguetes entren con la curiosa forma que tiene de guardarlos. ¿Puedes ayudarles?. Por ejemplo, si los juguetes de Botas tienen los tamaños 1, 2, 3, 4 y 5 (en ese orden) y sus padres le van a comprar 3 cajas, el tamaño mínimo de las cajas será de 6: en la primera caja el niño meterá los juguetes de tamaños 1, 2, 3 (6 en total), en la segunda meterá el de tamaño 4 y en la última el de tamaño 5. Fíjate que si comprarán las cajas de tamaño 5, en la primera irían los juguetes de tamaños 1 y 2, en la segunda el de tamaño 3, en la tercera el de tamaño 4, y Botas no tendría caja para su último juguete. Si los juguetes tienen tamaño 4, 78 y 9, y deciden comprarle solamente dos cajas, el tamaño mínimo tendrá que ser de 82, para que Botas meta los dos primeros juguetes en la primera caja, y el último en la segunda, que se quedará prácticamente vacía. Un tamaño de 81 sería insuficiente: Botas necesitaría otra caja más.

Input

La entrada consiste en dos números N ($1 \leq N \leq 1000$) y M ($1 \leq M \leq 1000000000$) donde N es el número de juguetes que posee Botas, y M es el número de cajas que le comprarán a Botas. A continuación sigue una línea con N números enteros, que son los tamaños de los juguetes de Botas.

Output

Se debe imprimir el tamaño mínimo que deben tener las cajas de Botas.

Ejemplos de entrada	Ejemplos de salida
5 3 1 2 3 4 5 3 2 4 78 9	6 82

Problema F

Foquitos

Ya se acerca la Navidad y Botas quiere armar su arbolito de Navidad con algunos adornos y foquitos pero se percata que algunos foquitos no prenden y decide contar aquellos foquitos que no prenden.

El sabe que los foquitos estan conectados como un arbol (no simple binario) y que la raiz siempre será el foquito 1 con la unica variación de que los ultimos foquitos estan conectados al enchufe que les provee de electricidad(Vea la imagen).

Lo que significa que si un foquito no prende es por que esta quemado o por que uno de sus hijos esta quemado no dejando pasar electricidad.

Asi Botas solo marca los foquitos quemados olvidandose de contar los foquitos que no prenden.

Botas ya se encuentra cansado y te pide ayuda para que encuentres cuantos foquitos no prenderán.

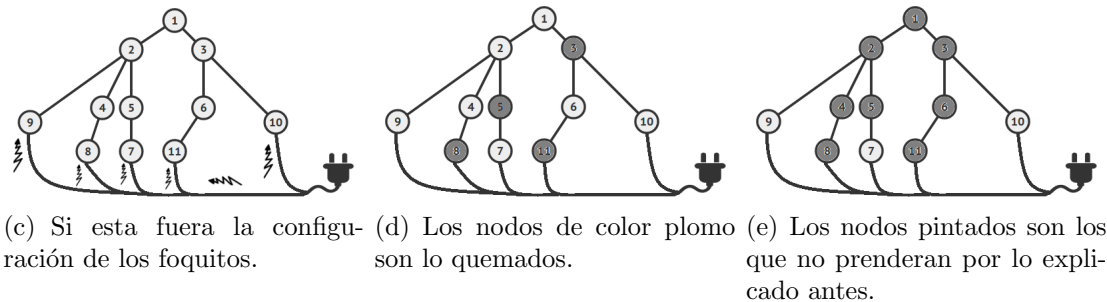


Figura 1: Ejemplo del primer caso de entrada el resultado es 8

Input

La primera linea contiene un número entero T ($1 \leq T \leq 1000$) que representa el número de casos de prueba, para cada caso la entrada consta de dos números N ($1 \leq N \leq 10^5$) Q ($1 \leq Q \leq N$) que representa el número de foquitos y la cantidad de foquitos quemados. Luego siguen $N - 1$ lineas con dos números u, v ($1 \leq u < v \leq N$) denotando que existe una conexión de u a v .

Output

Para cada caso de prueba mostrar la cantidad de foquitos que no prenderan.

Ejemplos de entrada	Ejemplos de salida
2	8
11 4	2
1 2	
2 9	
2 4	
2 5	
3 6	
3 10	
4 8	
5 7	
6 11	
8 5 11 3	
2 1	
1 2	
2	

Problema G

Girl

Nuestro querido amigo Botas está enamorado, y para poder conocer a la chica que le gusta necesita sentarse cerca a ella en clase. Los asientos en el aula están acomodados uno alado de otro en forma paralela, Botas tiene mas posibilidades de conocer a la chica si se sienta justo alado de ella, si se sienta diagonalmente a ella, también la podría conocer pero tiene menos posibilidades de hacerlo. Botas acaba de llegar a su curso y la chica no está, debemos ayudarle a determinar si existe un lugar libre donde si él se sentara tendría la oportunidad de conocerla, tomando en cuenta que ella podría sentarse en cualquier lugar que está libre.

Input

La entrada comienza con T casos de prueba ($1 \leq T \leq 100$). Cada caso de prueba comienza con dos enteros m y n ($1 \leq n, m \leq 1000$) que representan la cantidad de filas y columnas en el curso. La entrada consiste de un conjunto de caracteres que representan los asientos del curso. Un punto (.) representa un asiento libre y un simbolo de numeral (#) representa un asiento ocupado por cualquier otro compañero.

Output

Existen tres posibles respuestas que tú puedes darle a Botas. Si existe un lugar libre donde si Botas se sienta, si o si estará alado de la chica que le gusta, deberás imprimir "Genial!" sin comillas. Si el mejor lugar donde sentarse solo consigue que si o si Botas esté sentado diagonalmente a su chica, deberás imprimir "hmmm..." sin comillas. Si no ocurre ninguno de los anteriores casos imprimir "No va dar Botas" sin comillas.

Ejemplos de entrada	Ejemplos de salida
4	Genial!
2 2	hmmm...
.#	No va dar Botas
..	hmmm...
3 3	
.#.	
##	
.#.	
3 4	
....	
.###	
####	
3 3	
...	
...	
...	

Problema H

Homero

Este problema esta basado en un capitulo de los Simpsons. Homero esta jugando baseball, para el equipo de la planta nuclear en la que trabaja, su coach es el señor Burns, el cual lo llama para batear. Como muchos deben saber, se utilizan muchas señas para comunicarse y asi no revelar el secreto a los rivales. El señor Burns le dice a Homero: Simpson, estas son las senales: Si me toco la gorra de esta forma, esa señal es falsa, pero se hace real si me sacudo las manos, Si quiero un toque, me toco la hebilla del cinturon, no una ni dos sino tres veces. En ese momento, Homero se cansa de escuchar las indicaciones del señor Burns y piensa: No se de que me esta hablando, porque no me deja batear y ya. El problema es sencillo, se te pide contar cuantas instrucciones da el señor Burns antes de que Homero se canse de escucharlo y diga porque, (en minusculas).

Input

La primera linea tiene el numero de casos ($1 \leq T \leq 100$). Cada caso comienza con un numero ($1 \leq N \leq 100$) que indica las palabras en el dialogo. Las siguientes N lineas, tienen una palabra en cada una (no hay la necesidad de leer la linea, solo sera una cadena sin espacios). Se garantiza que existira un solo "porque" en cada caso de entrada. Cada palabra tendra como maximo 20 caracteres.

Output

Por cada caso de entrada, imprimir en una linea el numero de instrucciones que escucha Homero.

Ejemplos de entrada	Ejemplos de salida
3	3
4	1
instruccion1	0
instruccion2	
instruccion3	
porque	
3	
instruccion1	
porque	
instruccion2	
2	
porque	
blahblahblah	

Problema I

I Love You XOR

Botas tiene un arreglo. El quiere encontrar el valor obtenido haciendo XOR a los subarreglos contiguos, seguidos por el resultado de hacer XOR a los valores obtenidos. Puedes ayudarlo en esta tarea? Nota: $[1,2,3]$ es un subarreglo continuo de $[1,2,3,4]$, mientras que $[1,2,4]$ no lo es.

Input

La primera línea contiene el numero T ($1 \leq T \leq 5$), de los casos de prueba. La primera línea de cada caso de prueba contiene un entero N ($2 \leq N \leq 10^5$), numero de los elementos del arreglo. La segunda línea de cada prueba contiene N enteros, que son los elementos del arreglo ($0 \leq \text{numerosdelarreglo} \leq 10^8$).

Output

Imprime en una línea separada la respuesta correspondiente a cada caso de prueba.

Ejemplos de entrada	Ejemplos de salida
2	2
3	0
1 2 3	
4	
8 7 6 5	

Explicación

$$XOR = \oplus$$

$$1 \oplus 2 \oplus 3 \oplus (1 \oplus 2) \oplus (2 \oplus 3) \oplus (1 \oplus 2 \oplus 3) = 2$$

$$8 \oplus 7 \oplus 6 \oplus 5 \oplus (8 \oplus 7) \oplus (7 \oplus 6) \oplus (6 \oplus 5) \oplus (8 \oplus 7 \oplus 6) \oplus (7 \oplus 6 \oplus 5) \oplus (8 \oplus 7 \oplus 6 \oplus 5) = 0$$

Problema J

Jackiado

Le jackiaron su contraseña de Facebook a Botas, y publicaron en Infoamigos que es gay. Ahora debe crear una nueva contraseña y esta vez no debe ser tan obvia. Se dice que una contraseña es segura cuando tiene almenos una letra mayúscula, un número y un carácter especial. Ayuda a verificar si la nueva contraseña de Botas es segura o no.

Input

La entrada comienza con T casos de prueba ($1 \leq T \leq 100000$). Cada caso de prueba es una contraseña con maximo 20 caracteres ($1 \leq \text{contraseña} \leq 20$), y que solo consta de minúsculas, mayúsculas, números y los caracteres especiales arroba (@), punto (.), barra baja (_), menor (<), mayor (>) y guión (-).

Output

Si la contraseña es segura, imprimir "Dale no te jackiaran esta vez." sin comillas. Caso contrario imprimir "No va dar Botas.".

Ejemplos de entrada	Ejemplos de salida
5	No va dar Botas.
Botas123	No va dar Botas.
75803727	Dale no te jackiaran esta vez.
B0t45_Ru135	No va dar Botas.
31-07-1996	Dale no te jackiaran esta vez.
D0R4_14_Exp10r4D0r4_I_<3_U	

Problema K

Kuadragonia

Kuadraconia es una ciudad, como podran imaginar es cuadrada, que consiste de N filas y M columnas de casas de colores, cada casa esta representada por un caracter, la inicial del color de la casa. kuadraconia tiene otra particularidad, la primera fila esta al lado de la ultima fila, asi mismo la primera columna esta al lado de la ultima columna. Botas esta realizando una aplicacion movil para mapear y mostrar kuadraconia teniendo como punto de referencia en la parte superior izquierda a la casa del usuario. Por ejemplo kuadraconia sin la aplicacion es representado asi:

```
ABRCB
RCARA
RAMNB
```

Si la casa del usuario se encontrara en la fila 3 y columna 4, entonces al usar la aplicacion, kuadraconia deberia ser mostrada asi:

```
NBRAM
CBABR
RARCA
```

Botas es muy flojo, todavia no ha avanzado en esta parte del proyecto, por ello pide tu ayuda.

Input

La primera linea tiene el numero de casos ($1 \leq T \leq 100$). Cada caso de entrada, comienza con 2 numeros enteros ($1 \leq N \leq 100$) y ($1 \leq M \leq 100$). Las siguientes N lineas, cada una consiste de una cadena que consiste de M caracteres en mayusculas. La siguiente linea tiene dos numeros ($1 \leq F \leq N$) la fila y ($1 \leq C \leq M$) la columna, que representa la localizacion de la casa del usuario que usara la aplicacion.

Output

Por cada caso de entrada, imprimir N lineas con M caracteres, el resultado de que el usuario que esta en la (F, C) use la aplicacion.

Ejemplos de entrada	Ejemplos de salida
<pre>1 3 5 ABRCB RCARA RAMNB 3 4</pre>	<pre>NBRAM CBABR RARCA</pre>

Problema L

Loco por el uno

Botas es un muchacho a quien le gusta mucho los números en especial el numero uno (por que el sueña ser algún día el número uno en programación). Botas elije un número entero positivo y en el puede realizar una de las tres siguientes operaciones:

- Restarle 1. ($n = n - 1$)
- Si es divisible entre 2, dividirlo entre 2. ($n = n/2$)
- Si es divisible entre 3, dividirlo entre 3. ($n = n/3$)

Ahora la pregunta es, dado un entero n , encontrar el mínimo número de operaciones para llegar al número 1.

Input

La primera línea contiene un numero T ($1 \leq T \leq 10^5$), que representa el número de casos de prueba. Cada una de las T siguientes líneas contiene un entero n ($1 \leq n \leq 1000$) el numero elegido por Botas;

Output

Imprimir en una línea el mínimo número de operaciones requeridas para llegar a 1.

Ejemplos de entrada	Ejemplos de salida
2	4
10	3
9	

Problema M

Modulo

Botas esta tratando de aprender aritmetica modular. En este momento se encuentra leyendo el libro *Introduction to algorithms* y justo se encuentra con la parte que dice:

Dados A , B y M numero enteros naturales, siempre se cumple las siguientes propiedades:

$$\blacksquare (A + B) \% M = ((A \% M) + (B \% M)) \% M$$

$$\blacksquare (A * B) \% M = ((A \% M) * (B \% M)) \% M$$

donde el simbolo $\%$ significa la operacion *modulo*.

Ahora Botas esta tratando de resolver un problema que encontro en un juez online. El problema le pide: Dado dos numeros A y M ($1 \leq A < 10^{10^6}$, $1 \leq M \leq 1000$), calcular: $A \% M$ Ayuda a botas a resolver este problema.

Nota: A podria ser un numero muy grande (Un numero con 1000000 digitos).

Input

La primera línea contiene un numero T ($1 \leq T \leq 10$), que representa el número de casos de prueba. Cada una de las T siguientes líneas contiene dos enteros A y M .

Output

T líneas, cada línea con el resultado de $A \% M$ en el mismo orden de la entrada

Ejemplos de entrada	Ejemplos de salida
4	83
651838061 99	32
836075 51	735
284235 810	17
17 533	