

Problema A

La formula Luhn

Tú estas desarrollando una aplicación para transacciones online y quieres aceptar muchas tarjetas de crédito. Los números de estas tarjetas, sin embargo, son usualmente largas, tanto que es fácil cometer errores cuando lo ingresas. Tú quieres crear un programa que verifique los números ingresados por el usuario.

Tú sabes que la Formula Luhn aplica para todos los números de crédito aceptables.

La formula Luhn funciona de la siguiente manera.

Primero, separa los dígitos individualmente de las tarjetas de crédito. Por ejemplo:

21378 se vuelve

2 1 3 7 8

Sí la cantidad de dígitos es par, multiplica por 2 cada digito que se encuentra en una posición impar. De otra manera multiplica por 2 cada digito que se encuentra en una posición par. Posiciones están indexadas desde el 1, de tal manera que el primer digito está en la posición 1. El número del ejemplo anterior contiene una cantidad de dígitos impar, entonces nosotros multiplicamos por 2 cada dígito que está en una posición par.

2 1 3 7 8 se vuelve

2 2 3 14 8

Nótese que la posición par se refiere al número original, de tal manera que ellos no cambian inclusive cuando un número de 2 dígitos aparece.

Finalmente, tomamos la suma de todos los dígitos (para la suma del número de 2 dígitos inserta ambos dígitos separadamente en la suma):

$$2+2+3+1+4+8 = 20$$

Si la suma es un múltiplo de 10, el número es valido, en caso contrario, es inválido.

Input

La primera línea del INPUT comienza con un número M ($1 \leq M \leq 100$), M líneas le siguen, cada línea contiene un string P que contiene entre 0 y 50 (inclusive) caracteres representando el número de tarjeta de crédito, cada carácter será un dígito ('0'-'9').

Output

Ud. Deberá formatear su salida de la siguiente forma "Caso #N: V" (Comillas para claridad) donde N es el número de caso y V es "VALID" en caso sea válido o "INVALID" (Comillas para claridad) en caso sea un número de tarjeta no válido.

Input de ejemplo	Output del Input de ejemplo
5	Caso #1: VALID
21378	Caso #2: INVALID
31378	Caso #3: VALID
11111101	Caso #4: VALID
50005	Caso #5: INVALID
542987223412	

Problema B

Número de Caja

Tú quieres pegar el número de caja en la tapa. La única tienda en tu calle vende conjuntos de dígitos de plástico, cada conjunto contiene exactamente 10 dígitos (0-9). Responde el número de conjuntos requeridos para escribir el número de la caja. Nótese que el 6 puede ser usado como 9 y viceversa.

Input

La primera línea contiene un entero N del numero de casos a analizar, N líneas siguen donde cada línea es un entero P ($1 \leq P \leq 1,000,000$) que representa el numero que se desea pegar en la caja.

Output

Ud. Deberá formatear su salida de la siguiente forma "Caso #N: X" (Comillas para claridad) donde N es el número de caso y X es el número de sets necesarios para formar el número P.

Input de ejemplo	Output del Input de ejemplo
4 122 9999 12635 888888	Caso #1: 2 Caso #2: 2 Caso #3: 1 Caso #4: 6

* (1er Caso) Dos conjuntos se requieren porque cada conjunto contiene solo un dígito '2'.

* (2do Caso) Cada conjunto contiene un dígito '6' y un dígito '9'. '6' podría ser usado como '9' y así 2 sets serian suficientes.

Problema C

Tablero de Ajedrez

Un tablero de ajedrez de 8x8 es usualmente marcado de la siguiente manera: filas son marcadas por dígitos, 1 hasta 8, y columnas son marcadas por letras, 'a' hasta 'h'. Una celda es descrita por su marca de columna y su marca de fila, como "e4", tal como se

a8	b8	c8	d8	e8	f8	g8	h8
a7	b7	c7	d7	e7	f7	g7	h7
a6	b6	c6	d6	e6	f6	g6	h6
a5	b5	c5	d5	e5	f5	g5	h5
a4	b4	c4	d4	e4	f4	g4	h4
a3	b3	c3	d3	e3	f3	g3	h3
a2	b2	c2	d2	e2	f2	g2	h2
a1	b1	c1	d1	e1	f1	g1	h1

Trabajando en un programa de ajedrez, necesitas convertir estas descripciones en tus números internos de celda. Internamente, las celdas están numeradas fila por fila desde el 1 hasta el 64 en tu programa, es decir celda "a1" tiene el número 1, celda "b1" tiene el número 2, celda "c1" tiene número 3,..., celda "h8" tiene número 64.

57	58	59	60	61	62	63	64
49	50	51	52	53	54	55	56
41	42	43	44	45	46	47	48
33	34	35	36	37	38	39	40
25	26	27	28	29	30	31	32
17	18	19	20	21	22	23	24
9	10	11	12	13	14	15	16
1	2	3	4	5	6	7	8

Dado una cadena describiendo la marca de la celda o el número de la celda, se le pide cambiar la notación (es decir si te dan la marca necesitaras retornar el número, y viceversa).

Input

La primera línea contiene un entero N del numero de casos a analizar, N líneas siguen donde cada línea es un string **CELL** que contendrá o una marca o un número de celda.

- Si **CELL** contiene una marca de celda, contendrá exactamente 2 caracteres: una letra en minúscula entre 'a' y 'h', inclusive, seguido.

- Si **CELL** contiene un número de celda, tendrá un entero entre 1 y 64, inclusive, sin ceros a la izquierda.

Output

Ud. Deberá formatear su salida de la siguiente forma "Caso #N: X"
(Comillas para claridad) donde N es el número de caso y X es la respuesta.

Input de ejemplo	Output del Input de ejemplo
6	Caso #1: a1
1	Caso #2: b1
2	Caso #3: b4
26	Caso #4: 3
c1	Caso #5: 29
e4	Caso #6: 64
h8	

Problema D

k-Enfermo

Un string es llamado k-Enfermo si contiene exactamente **k** veces la palabra “dota” en él. Determinar **k**.

Input

La primera línea contiene un entero N del numero de casos a analizar, N líneas siguen donde cada línea representa a un string que contendrá entre 0 y 100 caracteres.

Output

"Caso #N: X" (Comillas para claridad) donde N es el número de caso y X es la respuesta.

Input de ejemplo	Output del Input de ejemplo
6	Caso #1: 1
dota	Caso #2: 0
DOTA	Caso #3: 0
DotA	Caso #4: 1
chino sucks playing dota	Caso #5: 4
dotaxdotaxdotaxdotax	Caso #6: 0
d o t a	

Problema E

Sucios Hackers

Tú eres el jefe de SuciosHackers S.A. Una compañía que crean pequeñas piezas de código malicioso, los cuales los adolescentes pueden usar para molestar a sus amigos. La compañía acaba de terminar su primer producto y es tiempo de venderlo. Tú quieres ganar tanto dinero como puedas y considerar la publicidad de tal manera que incremente las ventas. Contrataste un analista para predecir la ganancia con y sin publicidad. Tú ahora quieres tomar una decisión tanto como publicitar o no, dado las ganancias esperadas.

Input

La primera línea contiene un entero **N** del número de casos a analizar, **N** líneas siguen donde cada línea contiene 3 enteros **R**, **E** y **C** donde; **R** es la ganancia esperada si no publicitas, **E** es la ganancia esperada si publicitas, y **C** es el costo por la publicidad.

$$0 \leq N \leq 100$$

$$-10^6 \leq R, E \leq 10^6$$

$$0 \leq C \leq 10^6$$

Output

"Caso #N: X" (Comillas para claridad) donde N es el número de caso y X es o bien "publicitar", "no publicitar" o "no importa" teniendo en cuenta cual es lo más beneficioso o si no hace ninguna diferencia.

Input de ejemplo	Output del Input de ejemplo
3 0 100 70 100 130 30 -100 -70 40	Caso #1: publicitar Caso #2: no importa Caso #3: no publicitar