

Problema A

Fuuusion

Fuuusion es el proceso de unir dos **KIes** lo cual resulta en la suma de los dos. Trunks y Goten se enfrentarán Chino y Danny, tú recibirás el **KI** de los cuatro en el mismo orden, y tu debes determinar cuanto es el **KI** resultante del ganador (**Gotenks** o **Channy**) de la pelea (ganará la fusión que tiene el **KI** más elevado).

Gotenks es la Fuuusion de Trunks y Goten. **Channy** es la Fuuusion de Chino y Danny.

Input

La primera línea del INPUT comienza con un número M indicando el número de casos ($1 \leq M \leq 200$), M líneas le siguen.

- Cada línea tendrá 4 enteros separados por un espacio que son los 4 **KIes** de Trunks, Goten, Chino y Danny respectivamente.
 - o Cada **KI** estará entre 0 y 100, inclusive.

Output

Deberá formatear su salida de la forma "Caso #N: R".

(Comillas para claridad) donde N es el número de caso y R es el KI mas elevado de la Fuuusion ganadora.

Input	Output
3 3 3 5 5 1 1 1 1 3 15 1 1	Caso #1: 10 Caso #2: 2 Caso #3: 18

Problema B

ViejitosEnCola

A Xhalla le encomendaron una tarea en el banco, en una ventanilla de atención al cliente. Él está parado junto con la primera persona en la cola, él conoce las edades de todas las personas, y él está encargado de llegar a la persona de mayor edad y hacer que lo atiendan de inmediato.

Dado una lista con las edades de todas las personas en la cola de espera, retorne cuantas personas Xhalla tiene que pasar para llegar a la persona de mayor edad. Si existen más de 1 persona con la mayor edad, él escoge al primero que encuentra. Por ejemplo:

```
15 25 80 30 80
X
```

A Xhalla le tomará 2 pasos en llegar al primer anciano (de 80).

Input

La primera línea del INPUT comienza con un número M indicando el número de casos ($1 \leq M \leq 100$), M líneas le siguen.

- Cada línea tendrá una lista (entre 1 y 100 elementos, inclusive) de enteros separados por un espacio.
 - o Cada elemento de la lista será un entero entre 1 y 100;
- La primera persona de la cola es el primer elemento de la lista.

Output

Deberá formatear su salida de la forma "Caso #N: R".

(Comillas para claridad) donde N es el número de caso y R es el resultado.

Input	Output
2 15 25 80 30 80 1 1 1 1 1	Caso #1: 2 Caso #2: 0

Problema C

BinarioMasUno

El sistema de numeración binario representa los valores numéricos usando dos símbolos: 0 y 1. Contar en binario es similar a contar en cualquier otro sistema de numeración. Si tú quieres incrementar un número en 1, intenta incrementar el último dígito de su representación binaria. Si no se puede. Asigna 0 al último dígito, e intenta incrementar el dígito previo, y así sucesivamente hasta que logres incrementar un dígito.

Por ejemplo, la secuencia decimal:

1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, ...

Convertido a binario:

1, 10, 11, 100, 101, 110, 111, 1000, 1001, 1010, 1011, ...

Dado un String que contiene la representación binaria de un entero positivo **X**. retorna un String conteniendo la representación binaria de **(X+1)**. El resultado no debe tener ceros adelante.

Input

La primera línea del INPUT comienza con un número M indicando el número de casos ($1 \leq M \leq 50$), M líneas le siguen.

- Cada línea tendrá entre 1 y 30 caracteres, inclusive.
- Cada carácter de la cadena será 1 o 0.
- El primer carácter de la cadena será 1.

Output

Deberá formatear su salida de la forma "Caso #N: R".

(Comillas para claridad) donde N es el número de caso y R es el resultado, Asegúrese de agregar las llaves que encierran a la colección.

Input

```
5
10011
10000
1111
1
10101010101010101010101010101010
```

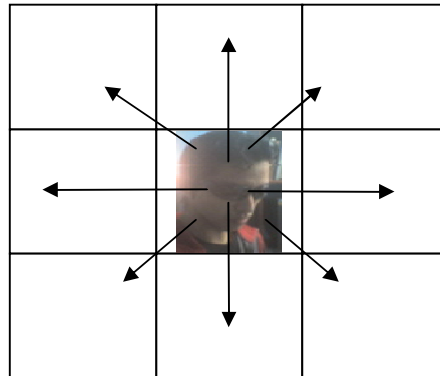
Output

```
Caso #1: 10100
Caso #2: 10001
Caso #3: 10000
Caso #4: 10
Caso #5: 101010101010101010101010101011
```

Problema D

ElComeTiburones

Xhalla, más conocido como el come tiburones, le encanta comer tiburones, bueno la verdad primero juega yu-gi-ho con su victima y después se lo come. Xhalla y tiburonsin están en una piscina dividida por cuadros (todos del mismo tamaño). Algunos cuadros están ocupados y Xhalla no puede nadar por ahí, Xhalla quiere llegar al tiburón lo más pronto posible, Xhalla puede nadar a las ocho direcciones a su alrededor:



El tiburonsin no se mueve. El trasladarse de un cuadro a otro cuadro adyacente le demora 1 segundo a Xhalla, Tu tarea es devolver la cantidad mínima de segundos que le tomaría a Xhalla atrapar a tiburonsin, si Xhalla no puede llegar a tiburonsin responda "FML" (Sin comillas).

Input

La primera línea del INPUT comienza con un número M indicando el número de casos ($1 \leq M \leq 50$), M configuraciones de laberintos le siguen. La configuración es la siguiente

- La primera línea de cada caso esta compuesta por dos enteros "**H W**" que están entre 2 y 10, inclusive que representa el tamaño de la piscina, H es el Nro. de filas y W el Nro. de columnas. H líneas siguen:
 - o Solo existirán los caracteres 'X' '@' '#' y '.'.
 - X es la ubicación inicial de Xhalla.
 - @ es la ubicación de **tiburonsin** y es donde Xhalla debe llegar.

- # significa que el cuadro está ocupado (obstáculo) y que Xhalla no puede viajar por ahí.
- Xhalla no puede salir de la piscina.
- Y '.' Significa que es un espacio vacío.

Output

Deberá formatear su salida de la forma "Caso #N: R".

(Comillas para claridad) donde N es el número de caso y R es el resultado.

Input

```
5
3 3
X..
. @.
...
3 3
X#.
#@.
...
3 3
X#@
.#.
.#.
6 5
X#...
.##.
.##.
.##.
...#@
.##.
8 7
X#.....
.#..#..
..#..#.
.....#
#.#..#.
..#.#..
#....@#
.#.#...
```

Output

```
Caso #1: 1
Caso #2: 1
Caso #3: FML
Caso #4: 12
Caso #5: 7
```