

---

**Harrichu y el laberinto****P15601\_es**

---

Harrichu, nieto del famoso arqueólogo Indiana Jones, es ya un hombre mayor. Siente que hoy en día, a su edad, no tiene sentido pasarse el día entero recorriendo laberintos arriba y abajo, y más teniendo un sofá de piel tan cómodo como el que tiene en un su despacho y una botella de chivas a medio terminar.

Después de consultarlo con su amigo Jorge Minish, reputado experto en informática, ha decidido que el próximo laberinto lo recorrerá un becario, mientras Harrichu le irá diciendo por SMS que camino debe seguir. Para guiar a su ayudante, nuestro no-tan-intrépido arqueólogo cuenta con un plano del laberinto.

–De hecho –le dice Jorge mientras comentan la idea en el lujoso despacho de Harrichu– ni siquiera es necesario que te molestes a guiar al becario. Es muy sencillo hacer un programa que resuelva el laberinto.

–¿De veras? –responde Harrichu pensativo. Por una lado le da pereza escribir tantos SMS, pero por otro lado le preocupa pensar lo fácil que puede resultar ser substituído por una máquina– Mmmmm. En realidad, no necesito algo tan complejo como lo que propones... Pero, la verdad, me sería útil un programa que me dijera de antemano si un laberinto tiene o no solución, así sólo gastaría mi valioso tiempo en aquellos laberintos provechosos. ¿Puedes ayudar a Harrichu escribiendo tal programa?

**Entrada**

La entrada consiste en una línea con un número  $n$  de laberintos, entre 1 y 100, seguida de los  $n$  laberintos. Un laberinto viene dado por dos números  $f$  y  $c$  en una línea, separados por un espacio, que indican el número de filas y de columnas que tiene el laberinto. Se cumple  $4 \leq f, c \leq 30$ . A continuación vienen  $f$  líneas de exactamente  $c$  caracteres cada una, describiendo las  $f$  filas del laberinto.

En la descripción de un laberinto (ved los ejemplos de entrada propuestos), un carácter '.' indica una casilla del laberinto por la que el becario puede pasar, mientras que un carácter '#' o una letra mayúscula indica una pared o una trampa, recuadros por los que el becario no puede pasar. En cada laberinto aparece una única vez el carácter 'b' (indicando la posición inicial del becario) y el carácter 'g' (la posición a la que debe llegar el becario). Se te garantiza que los extremos del laberinto (o sea, las casillas que pertenecen a la primera o última fila o columna) están cubiertos por caracteres '#'.  
El becario puede moverse por el laberinto mediante pasos horizontales o verticales, pero no diagonales.

Vuestro programa deberá resolver una entrada como la propuesta en 1 segundo de tiempo.

**Salida**

La salida consiste en  $n$  líneas. Para cada laberinto, debes escribir "Good" si es resoluble (es decir, si el becario puede llegar de 'b' a 'g' mediante pasos horizontales o verticales avanzando únicamente por casillas '.') y "Bad" si no lo es.

### Ejemplo de entrada 1

```
3
6 4
####
#g.#
#H.#
#.#
#b##
####
9 8
#####
#.g#...#
#.#....#
#...Y#.#
####...#
#.#...#
#...b...#
#.#...#
#####
7 7
#####
##S...#
#...#.#
#.bQB.#
####.#
#.g..M#
#####
```

### Ejemplo de salida 1

```
Good
Good
Bad
```

### Información del problema

Autoría: Omer Giménez

Generación: 2026-01-25T10:12:23.629Z

© *Jutge.org*, 2006–2026.

<https://jutge.org>