Memorama

|  |  |  |  |
| --- | --- | --- | --- |
| Puntos |  | Límite de memoria | 32MB |
| Límite de tiempo (caso) | 1s | Límite de tiempo (total) | 60s |

**Historia**

Karel encontró su memorama de animales bebés regado por todo su cuarto y para jugar con sus amigos necesita saber si está completo.

Las cartas del memorama están representadas por montones de zumbadores: la cantidad de zumbadores en cada montón representa la carta, es decir, dos montones de (por ejemplo) 3 zumbadores, hacen una pareja.

Karel quiere saber si el memorama está completo.

**Problema**

Escribe un programa que ayude a Karel a saber si su memorama está completo. El memorama está completo si todas las cartas tienen pareja. Si el memorama está completo, Karel debe apagarse orientado al norte. Si el memorama NO ESTÁ completo, Karel debe apagarse orientado al sur.

**Consideraciones**

•Karel no tiene zumbadores en la mochila.

•Karel inicia en la casilla (1,1) con orientación desconocida.

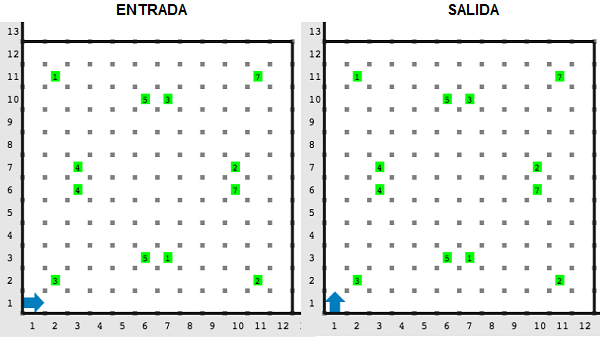
•Nunca hay más de 2 cartas con el mismo número de zumbadores.

•La cantidad de zumbadores en la carta con número más alto es menor a las dimensiones del mundo.

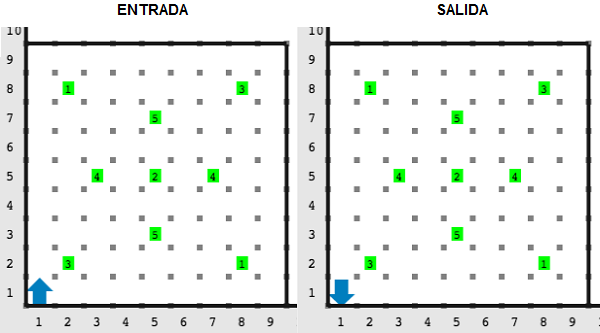
•El memorama está incompleto si hay al menos una carta sin pareja.

•Para obtener los puntos del problema Karel puede apagarse en cualquier casilla del mundo, pero debe quedar viendo al norte si el memorama está completo o viendo al sur si está incompleto. No importan los zumbadores que queden en el mundo.

**Ejemplos**



En este caso, el memorama está completo. Karel debe apagarse orientado al norte.



En este caso, el memorama está incompleto porque solamente hay una carta con el número 2. Karel debe apagarse orientado al sur.

# Karel y Warel

|  |  |  |  |
| --- | --- | --- | --- |
| Puntos |  | Límite de memoria | 32MB |
| Límite de tiempo (caso) | 1s | Límite de tiempo (total) | 30s |

**Historia**

Karel y Warel, enemigos eternos e igual de poderosos, han decidido ponerle fin a su épica pelea.

Karel y Warel mantendrán un duelo en la primera fila del mundo. Por suerte, Karel se ha enterado de la estrategia que usará Warel durante el duelo y ahora quiere saber si podrá derrotarlo.

El resultado de la lucha entre Karel y Warel depende de la energía que cada uno tenga, sin embargo, la energía de Warel disminuye conforme se mueve en el mundo. El duelo inicia en la casilla (1, 1) del mundo y se recorre una casilla a la derecha cada turno. Cada que Warel se mueve una casilla a la derecha, su energía disminuye en 1. Al llegar a la nueva casilla, puede haber un módulo de recarga que estará representado por un montón de zumbadores, en caso de que lo haya, Warel lo usa y su energía aumenta en una cantidad igual al número de zumbadores del montón. Si en algún momento la energía de Warel disminuye a 0 y no hay ningún módulo de recarga, Warel no puede continuar el duelo.

Karel logró encontrar un mapa que dice dónde habrá módulos de recarga y cuánta energía tendrán. Usando este mapa, Karel desea saber si Warel tendrá energía suficiente para recorrer todo el mundo y si la tiene, saber cuál será la cantidad máxima de energía que tendrá durante la duración del duelo.

Warel inicia el duelo con energía 0.

**Problema**

Escribe un programa que dado el mapa de los módulos de recarga determine si Warel tendrá suficiente energía para recorrer el mundo y si la tiene cuál será la máxima cantidad de energía que llegará a tener.

Tu programa deberá dejar en la casilla (1, 1) una cantidad de zumbadores igual a: + Cero: Si Warel no tendrá energía suficiente para recorrer el mundo, es decir, si en algún momento su energía llegará a cero sin que tenga un módulo de recarga. + La cantidad máxima de energía que Warel llegué a tener en el caso de que Warel nunca llegué a quedarse sin energía.

**Consideraciones**

* Karel inicia en la posición (1,1) con orientación desconocida.
* Para obtener los puntos de este problema no importa la posición ni orientación final de Karel, solo los zumbadores que queden en la casilla (1,1).
* En este problema habrá diferentes subtareas que varían dependiendo de la cantidad de zumbadores que Karel lleva en la mochila y dimensiones del mundo.

**Subtareas**

**Subtarea 1 (17 puntos)**

* Karel inicia con infinitos zumbadores en la mochila.
* El mundo de Karel tiene 100 casillas de altura.
* La primera fila tiene 100 casillas de largo.

**Subtarea 2 (21 puntos)**

* Karel inicia con infinitos zumbadores en la mochila.
* El mundo de Karel tiene 2 casillas de altura.
* La primera fila tiene una longitud variable.

**Subtarea 3 (23 puntos)**

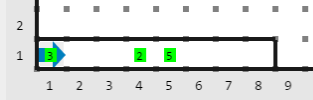
* Karel inicia con infinitos zumbadores en la mochila.
* El mundo de Karel tiene 1 casilla de altura.
* La primera fila tiene una longitud variable.

**Subtarea 4 (39 puntos)**

* Karel inicia con 0 zumbadores en la mochila.
* El mundo de Karel tiene 1 casilla de altura.
* La primera fila tiene una longitud variable.

**Ejemplo 1**

**Entrada**



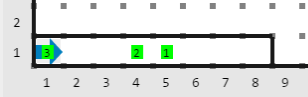
**Salida**



En este ejemplo, la energía de Warel nunca llega a cero, y la cantidad máxima de energía que llegó a tener fue de 6 en la casilla 5. Por lo tanto Karel debe dejar 6 zumbadores en la casilla (1,1).

**Ejemplo 2**

**Entrada**



**Salida**



En este ejemplo, la energía de Warel llega a cero antes de que pueda llegar al final del mundo, por lo que tu programa debe dejar cero zumbadores en la casilla (1, 1).

Nuevo Veracruz

|  |  |  |  |
| --- | --- | --- | --- |
| Puntos |  | Límite de memoria | 32MB |
| Límite de tiempo (caso) | 1s | Límite de tiempo (total) | 60s |

**Historia**

Recientemente, en una audaz estrategia de expansión, el Grupo Karso ha decidido construir un nuevo mega centro comercial en la ciudad de Veracruz.

Sin embargo, Veracruz es una ciudad que ya cuenta con muchos edificios construidos, y para hacer que el mega centro comercial sea lo más mega posible, se le ha encomendado a Karel Mosby, arquitecto, la importante tarea de encontrar el área de superficie cuadrada más grande posible donde el nuevo centro comercial pueda ser construido.

Para realizar esta tarea, Karel cuenta con un mapa de Veracruz donde cada edificio se representa por una pared horizontal (**No existe ninguna pared vertical dentro del mundo**).

**Problema**

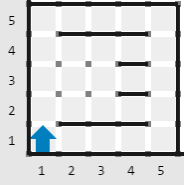
Escribe un programa que encuentre, en el mapa, el área cuadrada más grande que no tenga ningún edificio construido. Tu programa debe dejar en la casilla (1, 1) un montón de zumbadores igual al área de dicho cuadrado.

**Consideraciones**

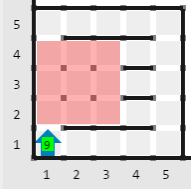
* Karel inicia en la posición (1, 1) viendo al norte.
* Karel inicia con infinitos zumbadores en la mochila.
* El mundo de Karel puede ser rectángular, mide como máximo 50 x 50 y está delimitado por paredes.
* No existen paredes verticales adentro del mundo que representa Veracruz.
* El área que buscas debe ser cuadrada.
* Para obtener los puntos de este problema no importan la posición ni orientación final de Karel, solo los zumbadores en la casilla (1, 1).

**Ejemplo**

**Entrada**



**Salida**



*En el ejemplo de salida, y solo para fines informativos, el área cuadrada más grande (que consta de 9 casillas) se encuentra resaltada en color rojo.*

La muerte de Karel

|  |  |  |  |
| --- | --- | --- | --- |
| Puntos |  | Límite de memoria | 32MB |
| Límite de tiempo (caso) | 15s | Límite de tiempo (total) | 60s |

**Historia**

Finalmente, Karel será castigado por sus crímenes contra la informática. Después de 13 años de hacer sufrir a los competidores de la OMI, el Comité de la Olimpiada Mexicana de Informática ha sentenciado a Karel a la muerte.

Sin embargo, antes de recibir su merecido, Karel tiene que esperar en una fila compuesta de varios prisioneros esperando su destino. Todos los prisioneros tienen alturas enteras y positivas.

La fila de los prisioneros está representada con montones de zumbadores en la primera fila; cada montón representa un prisionero y la cantidad de zumbadores en el montón representa su altura.

Todos los prisioneros están viendo a la izquierda. Cada prisionero puede ver sólo hasta el siguiente prisionero más alto hacia la izquierda o hasta la pared, ya que cualquiera de estas dos cosas le obstruyen su visión.

La fila de prisioneros está delimitada por paredes que hacen un mundo de una fila de alto y *N* columnas de ancho. En este mundo Karel se enfrenta con su último problema de la OMI:

**Problema**

Escribe un programa que, en cada posición de la primera fila del mundo, deje un montón de zumbadores igual a la altura del prisionero que obstruye la visión del prisionero que originalmente está en esa posición. En caso de que la visión del prisionero sea obstruida por la pared, deberás dejar 0 zumbadores en esa posición.

**Consideraciones**

* Karel inicia en la posición (1,1) y su orientación depende de la subtarea que se está ejecutando.
* Karel tiene infinitos zumbadores en su mochila.
* Cada posición en la fila tiene más de 0 y menos de 100 zumbadores.
* Para obtener puntos en este problema, Karel deberá dejar en cada posición de la primera fila un montón de zumbadores igual a la altura del prisionero que obstruye la visión del prisionero originalmente en esa posición. En caso de que el prisionero en esa posición alcance a ver hasta la pared, deberá dejar 0 zumbadores.
* No importan la orientación ni la posición final de Karel.
* No importa la cantidad final de zumbadores en la mochila de Karel.
* **En este problema la cantidad máxima de instrucciones avanza/move que puede ejecutar Karel, así como las propiedades del mundo, cambian dependiendo de la subtarea.**

**Subtareas**

**Subtarea 1 [18 puntos]**

* Karel empieza orientado al norte.
* Si hay *N* columnas en el mundo, Karel podrá ejecutar avanza/move *N*×(*N*+1)  veces.

**Subtarea 2 [24 puntos]**

* Karel empieza orientado al sur.
* Todas las casillas tienen menos de 10 zumbadores.
* Si hay *N*  columnas en el mundo, Karel podrá ejecutar avanza/move 20*N* veces.

**Subtarea 3 [58 puntos]**

* Karel empieza con una orientación desconocida.
* Si hay *N*  columnas en el mundo, Karel podrá ejecutar avanza/move *N*  veces.

**Ejemplo**

**Entrada**



**Salida**

