LABORATORIO ATESTADO DE PERROS

AUTOR: LAUTARO LASORSA

Solución Esperada

La solución de este problema consiste en modelarlo como un grafo, donde cada baldosa (celda de la matriz que describe el laboratorio) es un nodo. Una forma fácil de asignar los números de los nodos es que la baldosa ubicada en la fila **i** y la columna **j** tiene **id = i*M + j**.

En este modelo cada baldosa está conectada a las 4 adyacentes, es decir, a las que puede alcanzar moviéndose 1 baldosa hacia arriba, abajo, izquierda o derecha. A su vez, solo vamos a tener en cuenta las baldosas libres, es decir que el grafo que modela el laboratorio no contendrá los nodos correspondientes a baldosas bloqueadas.

Resumen:
Baldosas => Nodos
Que una baldosa esté al lado de otra => Aristas

Además, se considerará que todas las baldosas entrada que estén libres están conectadas entre si.

Ahora podemos observar que la pregunta de cuantas baldosas son accesibles es equivalente a preguntar el tamaño de la componente conexa que contiene a las entradas libres (si no hay ninguna entrada libre porque todas fueron ocupadas por perros, entonces está componente no existe y se considera de tamaño 0).

Por lo tanto, podemos reformular el enunciado como :

“Dado un grafo, y una serie de nodos especiales (las entradas E) que están todos conectados entre sí, del que se van eliminando nodos de uno a uno, decir después de que cada nodo es eliminado cuál es el tamaño de la componente conexa que contiene a los nodos especiales”

Lo importante en este caso es que es offline, es decir que desde el principio sabemos en qué baldosas se ubicaran los perros (cuáles nodos serán eliminados). Por eso mismo, podemos pensar el problema en sentido inverso :

LABORATORIO ATESTADO DE PERROS

AUTOR: LAUTARO LASORSA

"Tengo un grafo y voy agregando nodos, de los cuales algunos pueden ser especiales, y esos están conectados todos entre sí. Cada vez que voy a agregar un nodo, quiero antes saber el tamaño de la componente conexa que contiene a los nodos especiales"

Planteado este problema podemos utilizar el algoritmo de Union Find (concreamente, la implementación de la solución oficial utiliza la versión que Wikipedia llama "Bosque de Conjuntos Disjuntos", pero ambas formas son validas y efectivas para este problema https://es.wikipedia.org/wiki/Estructura_de_datos_para_conjuntos_disjuntos), donde al agregar un nodo a su vez agregamos todas las aristas que lo unen a otros nodos validos. Al agregar una arista utilizamos la función Unir sobre los 2 nodos en sus extremos.

Notar que también podemos utilizar este mismo algoritmo para saber cuales son las componentes conexas antes de agregar cualquier nodo. (viendo los nodos que están en el grafo uno por uno y agregando sus aristas).

Así, inicialmente tendremos todos los perros en sus lugares (lo que nos da el **P-esimo** escenario, última posición del vector que se debe retornar) e iremos quitando los perros en el orden inverso al que entraron (es decir, recorremos el vector al revez), y cada vez que hacemos eso estaremos agregando un nodo libre (**posiblemente una entrada**) y utilizaremos el algoritmo de Union Find.

Soluciones Parciales

Para la subtarea **P=1**, la solución consiste en poner al perro en la baldosa (bloqueandola) y luego recorrer el grafo con cualquier algoritmo (DFS o BFS) que permita encontrar las componentes conexas. Luego de eso miramos la componente que contiene a las entradas, si es que existe, y sabemos su valor. También puede verse como la cantidad de nodos alcanzables desde los nodos E.

La subtarea donde **NxMxP<=1.000.000** utiliza exactamente la misma idea, solo que debe volverse a recorrer todo el grafo después de ubicar a cada perro (sacar un nodo).

LABORATORIO ATESTADO DE PERROS

AUTOR: LAUTARO LASORSA

Estas 2 soluciones son **O(NxMxP)**, ya que recorren el grafo de tamaño **NxM** por cada uno de los **P** perros.

En la subtarea **N=1** es fundamental observar que el laboratorio descripto resulta ser una fila. De esa fila, hay un conjunto contiguo de baldosas que son las entradas, es decir describen un intervalo de baldosas en la fila. Por lo mismo, para cada baldosa es posible decir si pertenece a ese intervalo, si está antes o si está después.

Así, cuando bloqueamos una baldosa que antes era accesible sucede qué :

1. Si la baldosa está antes del intervalo de entrada, o es el extremo menor del mismo, deja inaccesibles todas las baldosas anteriores.
2. Si la baldosa está después del intervalo de entrada, o es el extremo mayor del mismo, deja inaccesibles todas las baldosas posteriores.
3. Si la baldosa está contenida en el intervalo de entradas y no es ninguno de sus extremos, no afecta a ninguna otra baldosa.

Es importante notar que en los supuestos 1 y 2, cuando estamos marcando todas las anteriores/posteriores como inaccesibles y encontramos una que ya está marcada, debemos cortar ahí y no seguir recorriendo la fila.

Como cada baldosa puede ser marcada como inaccesible solo una vez, esta solución es **O(M)**, lo que entra perfectamente en tiempo.