OMI 2022 - Solución para *El jardín de las fuentes*

CASO 1: $N, M = 3$.

Debido a que las casillas en la orilla del jardín siempre están ocupadas, un cuadro de 3×3 sólo tiene una posición libre y por lo tanto debes llenara con una fuente.

CASO 2: $N, M = 100$, no hay cuadros libres adyacentes en el jardín.

Si no hay cuadros libres adyacentes en el jardín entonces todas las posiciones libres están rodeadas por casillas ocupadas, de modo que debes llenar todas con una fuente.

CASO 3: $N = 3$, $M = 500$

El jardín tiene únicamente una fila de casillas libres adyacentes.

Intuitivamente puedes llegar a la idea de que para llenar la fila completa el arreglo óptimo es `F.F.F.F.F`... hasta llenar la fila completa. Esta configuración utiliza $\lceil \frac{largo+1}{2} \rceil$ donde $largo$ es el número de casillas libres adyacentes en la fila.

Es sencillo ver que la configuración anterior llena toda la fila de agua. ¿Se puede hacer lo mismo con menos fuentes?

Llamemos casillas con *fuente* aquellas dónde se coloca una fuente y casillas con *agua* aquellas que se llenan de agua debido a que dos de sus casillas adyacentes tienen fuente o agua.

Observa que en la configuración de *línea* cada casilla libre tiene exactamente 2 casillas libres adyacentes, a excepción de las dos casillas en los extremos de la línea. Por lo tanto, para que una casilla se llene de *agua* ambas casillas a sus lados deben tener *fuente* o *agua*.

Supón que hay dos casillas adyacentes A y B y se quiere buscar una configuración en que ambas terminen con *agua*. Para que A se llene de *agua*, B tiene que tener primero *agua* y para que B se llene de *agua*, A debe tener *agua* antes. De modo que es imposible encontrar una configuración que llene el jardín en la que se dejen dos casillas libres adyacentes.

Además, por tener sólo una casilla libre adyacente, las dos casillas de los extremos de la línea siempre deberán llevar *fuente*.

Conclusión, la configuración `F.F.F.F.F`... es óptima.

CASO 4: $N, M = 499$, los únicos cuadros ocupados son los del contorno.

Las casillas libres del jardín forman un cuadrado al centro del mismo.

Observa que sin importar la configuración de *fuentes* que pongas, el área del jardín que se llena de *agua* siempre tiene una forma rectángular. Esto puede demostrarse con un razonamiento inductivo de la siguiente manera:

- Hay una primera *fuente* que se coloca y llena de *agua* su propia casilla. Esta configuración en sí misma tiene una forma rectangular.
- Supón que se han colocado x *fuentes* y el área con *agua* es rectangular, los lugares posibles donde se puede colocar una nueva fuente son:
  - En una casilla adyacente al área con *agua*: En este caso, el área con *agua* crece el rectángulo en una fila o una columna completa y mantiene la forma rectángular.
  - En una de las 4 casillas en las *esquinas* del área rectangular: En este caso, el área con *agua* crece una fila y una columna, manteniendo la forma rectángular.
  - En una casilla adyacente a una casilla adyacente al área con *agua*: En este caso, el área con *agua* crece dos filas o dos columnas, manteniendo la forma rectángular.
  - En cualquier otra casilla que se ponga una fuente no *interactúa* con el área con *agua* y crea su propia área rectangular.

Del razonamiento anterior puedes ver además que en las filas y columnas frontera del área rectángular siempre habrá al menos una *fuente*.

Por lo tanto puedes deducir que para llenar un área cuadrada puedes proceder de dos formas: * Llena una fila y una columna en cuyo caso se requiere de una *fuente* por cada dos filas y una *fuente* por cada dos columnas. * Llena la diagonal del cuadrado con *fuentes*.

Aunque en el caso del problema ambos caminos resultan en el mismo número de fuentes, la opción de la diagonal es mejor en cuadrados cuyo lado sea par.

CASO 5: $N = 500$, $M = 250$, los únicos cuadros ocupados son los del contorno.

Usando las observaciones del caso anterior puedes llenar una fila y una columna o llenar la diagonal de un cuadrado de 250×250 y extenderlo con una *fuente* por cada dos filas.

CASO 6: $N, M = 500$, los cuadros libres forman áreas rectangulares.

Para cada área rectangular utiliza el algoritmo del caso 5.

CASO 7: $N, M = 500$, los cuadros libres forman una cruz en el centro.

Divide la cruz en áreas rectángulares y utiliza el algoritmo del caso 5.

CASO 8: $N, M = 500$, los cuadros libres forman una X en el centro.

En base a las observaciones del caso 4 llena una de las diagonales de la cruz y luego la otra diagonal evitando todas las casillas del cuadro central, el cual se habrá llenado con la primera diagonal.

CASOS 9 a 12: $N, M = 500$, la distribución de los cuadros libres es aleatoria.

Se podían tomar múltiples estrategías para resolver los casos aleatorios, algunas de las estrategias son:

Llenar aleatoriamente con *fuentes* los espacios libres del mapa.

Con esta idea se obtenía alrededor del 20% de los puntos de estos 4 casos.

Decidir las posiciones de las fuentes con una búsqueda en profundidad (DFS).

Se puede visualizar el mapa como un grafo en donde las casillas vacías son nodos y las adyacencias entre casillas vacías son aristas. Sobre este grafo se ejecuta el siguiente algoritmo:

- Se elige una casilla vacía que no tenga *fuente* ni *agua*.
- Con una búsqueda en profundidad (DFS) se recorren sus nodos adyacentes para ver cuáles de ellos deben tener fuente. En el árbol que se genera durante el recorrido DFS...
    - Las *hojas* llevan *fuente* y se marcan como que tienen *agua*.
    - En un nodo intermedio...
    - Si al menos dos de sus hijos tienen *agua* entonces el nodo se marca como que tiene *agua*.
    - Si únicamente uno de sus hijos tiene *agua*, el nodo se marca como *agua* y se avisa al padre que si él **debe** tener *agua* ya sea con *fuente* o con sus demás hijos.
    - Si ningún hijo tiene *agua* entonces el nodo debe llevar *fuente*.
    - Si algún hijo le indicó que **debía** tener *agua* y no la consigue a través de sus demás hijos y padre, entonces el nodo debe llevar *fuente*
    - La raíz debe llevar *fuente* si no tiene al menos dos hijos con *agua*.

La solución anterior obtiene alrededor del 75% de los puntos de los casos aleatorios. Su principal debilidad es que hay configuraciones que dejan espacios sin llenar. Usando el validador que se proveía en el texto del problema se pueden ubicar esos espacios sin llenar para los cuales basta poner una fuente. Si se hace un segundo proceso para llenar esos espacios, esta solución obtiene alrededor del 85% de los puntos de los casos aleatorios y alrededor del 88% de todos los puntos del problema si se utiliza para procesar todos los casos.

Decidir las posiciones de las fuentes con una búsqueda en amplitud (BFS).

De los primeros casos puedes observar que ir llenando áreas y creciéndolas usando las esquinas (para crecer una fila y una columna) o usando una casilla a dos filas o dos columnas de distancia parece ser una técnica que da buenos resultados. Esta técnica se puede generalizar para los casos aleatorios con el siguiente algoritmo:

- Elige una casilla vacía que no tenga *agua* y ponle una *fuente*.
- Agrega a una cola (queue) las siguientes casillas en caso de que estén libres:
    - Las casillas en sus esquinas. De las observaciones anteriores viste que era mejor crecer en diagonales que en filas o columnas.
    - Las casillas a dos filas o dos columnas de distancia.

- Procesa la cola hasta que se quede sin elementos de la siguiente manera:
  - Obtén el elemento al frente de la cola.
  - Si la casilla ya tiene *agua*, ignóralo.
  - Si la casilla aún no tiene *agua* ponle una *fuente* y simula el proceso de llenado de *agua* hasta que no existan más casillas por llenar.
  - Por cada casilla que hayas llenado (con *fuente* o *agua*) durante el proceso de simulación, agrega a la cola a sus casillas vecinas que aún estén libres.

Dependiendo de la forma en la que se elijan los puntos de inicio de esta búsqueda en profundidad esta solución dará entre el 94% y el 100% de los puntos del problema.