



MEMORIA PRÁCTICA 1

ALGORITMO A*

INGENIERÍA DEL CONOCIMIENTO

CURSO 2023-2024

Carlos Canero Mérida
Clara Rodríguez Prieto

ÍNDICE

1. LENGUAJE Y ENTORNO SELECCIONADO	1
2. PROCEDIMIENTO DE IMPLEMENTACIÓN	1
3. MANUAL DE USUARIO	2

1. LENGUAJE Y ENTORNO SELECCIONADO

Para realizar la práctica, hemos utilizado Java como lenguaje ya que es en el que mejor nos desenvolvemos ambos estudiantes. Como entorno hemos utilizado Eclipse porque ya lo hemos utilizado en otras asignaturas, junto a GitHub Desktop para poder trabajar en conjunto.

2. PROCEDIMIENTO DE IMPLEMENTACIÓN

Para implementar la práctica, hemos creado un proyecto basado en el modelo Vista-Controlador y dividido en dos capas: Negocio y Presentación.

En la capa de Negocio, tenemos las clases con la lógica del proyecto. En esta capa tenemos cinco clases y una enumeración:

- `Nodo.java`: define el objeto nodo. Equivale a lo que sería una casilla dentro de la cuadrícula o tablero.
- `TiposNodos.java`: es una enumeración con los diferentes tipos de nodos o casillas que puede haber. Estos tipos son:
 - `VACIO`: es el tipo por defecto de un nodo ya que no tiene ninguna propiedad especial.
 - `SALIDA`: indica la casilla de inicio del algoritmo.
 - `META`: indica la casilla final del algoritmo.
 - `PROHIBIDO`: casillas que no se pueden incluir como solución.
 - `PELIGROSO`: casillas que se pueden incluir en la solución pero con un coste.
 - `WAYPOINT`: casilla que se debe incluir como parte de la solución.
 - `SOLUCION`: casilla perteneciente a la solución.

- `Cuadrícula.java`: define el objeto cuadrícula. Servirá como tablero para representar el algoritmo.
- `AEstrella.java`: contiene las funciones necesarias para ejecutar el algoritmo, tanto con *waypoints* como sin ellos.
- `Controlador.java`: es la clase que enlaza la capa de Negocio con la capa de Presentación. Contiene las funciones necesarias para poder mostrar y poder ejecutar el algoritmo.
- `Main.java`: inicializa el proyecto.

Por otro lado, tenemos la capa Presentación, donde se encuentran las clases para generar las vistas del proyecto. Para ello, hemos utilizado la biblioteca `JSwing` y sus componentes. Tenemos un total de cuatro clases:

- `DimensionesCuadrículaView.java`: crea un diálogo inicial para elegir el tamaño de la cuadrícula deseada.
- `PrincipalView.java`: panel principal donde se crean la cuadrícula y los controles. Este panel no se muestra hasta que no se ha cerrado el diálogo de las dimensiones de la cuadrícula.
- `CuadrículaView.java`: panel donde se pinta una cuadrícula de las dimensiones elegidas en el diálogo inicial formada por.
- `ConfiguraciónView.java`: panel formado por una serie de botones para indicar el tipo de un nodo, iniciar la simulación y reiniciar el tablero.

3. MANUAL DE USUARIO

Para poder ejecutar la práctica, es necesario tener el entorno de desarrollo de Eclipse. Para ello, deberemos tener en nuestro *workspace* importado el proyecto. Si no lo tuviéramos importado, tendríamos que ir a `File > Open Projects from File System...` para abrir la ventana que se

muestra en la Imagen 1. En esa ventana, seleccionaremos **Directory**, y se nos abrirá nuestro explorador de archivos, donde elegiremos nuestro proyecto. Una vez elegido, le daremos a **Finish** y se nos habrá importado a nuestro *workspace*, y aparecerá en el panel **Package Explorer** (panel lateral izquierdo), como se muestra en la Imagen 2.

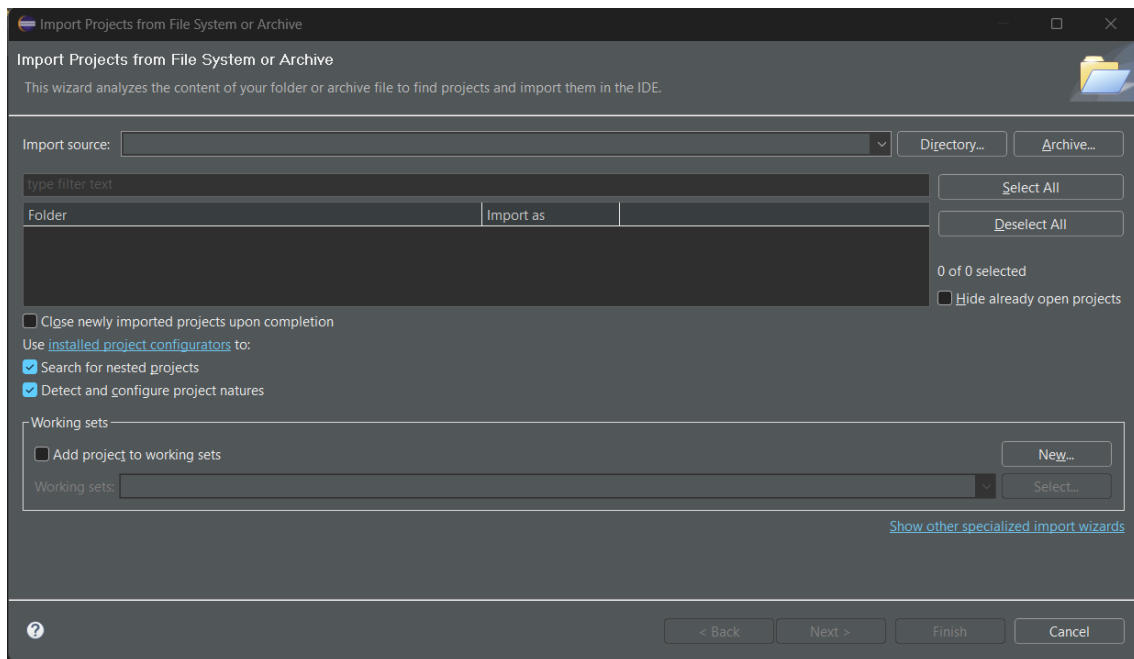


Imagen 1. Ventana para importar un proyecto en Eclipse

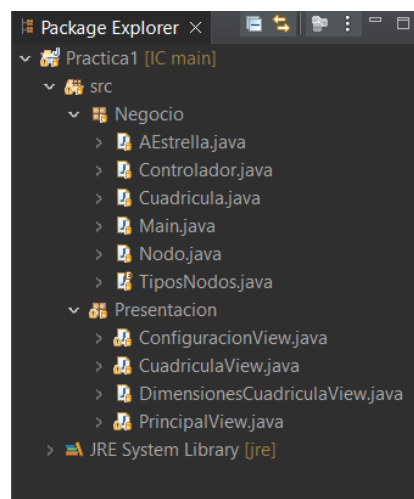


Imagen 2. Proyecto importado en Eclipse

Una vez tenemos nuestro proyecto importado, vamos a abrir nuestra clase `Main.java` y vamos a ejecutarlo en el botón de **Run Main** que se

encuentra en la barra superior de herramientas, como se indica en la Imagen 3.

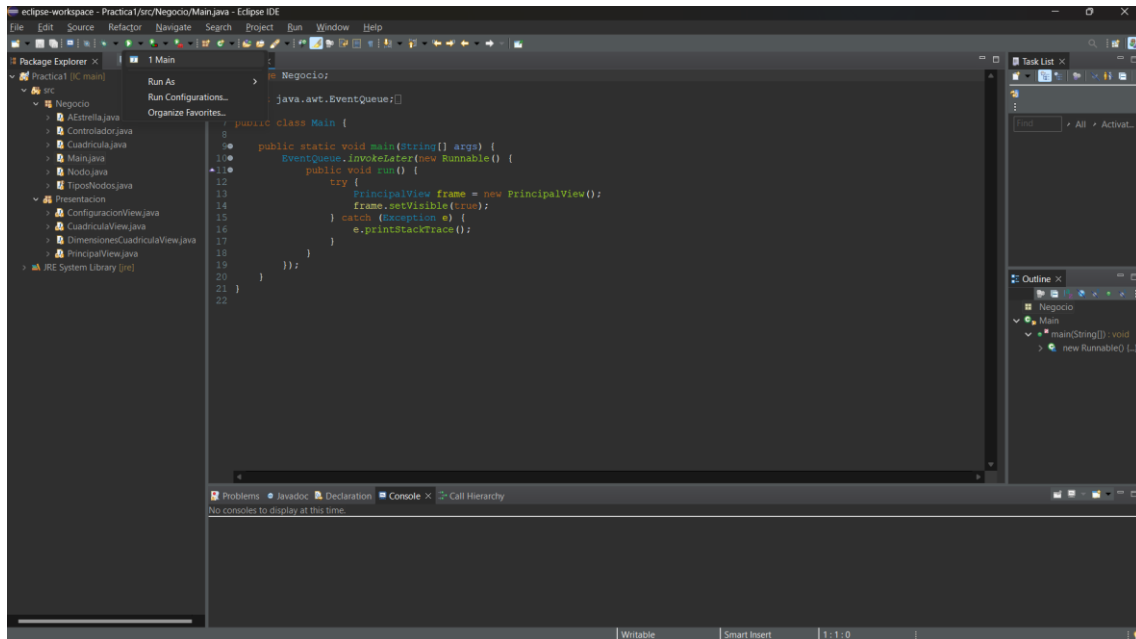


Imagen 3. Run Main

Cuando ejecutemos el Main, nos aparecerá un diálogo como el de la Imagen 4, con dos *spinners* uno para seleccionar el número de filas y otro para seleccionar el número de columnas que queremos que tenga nuestra cuadrícula. Una vez seleccionados, le daremos a “ACEPTAR”.

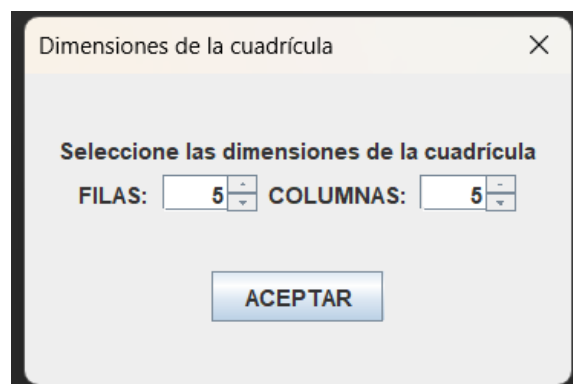


Imagen 4. Diálogo inicial de selección de dimensiones de la cuadrícula

Una vez le hayamos dado a “ACEPTAR” se cargará la pantalla principal donde aparecerán una sección con los diferentes botones y otra sección con la cuadrícula de las dimensiones seleccionadas en el diálogo inicial. En la

Imagen 5, se muestra un ejemplo de pantalla principal con una cuadrícula de 5x5.

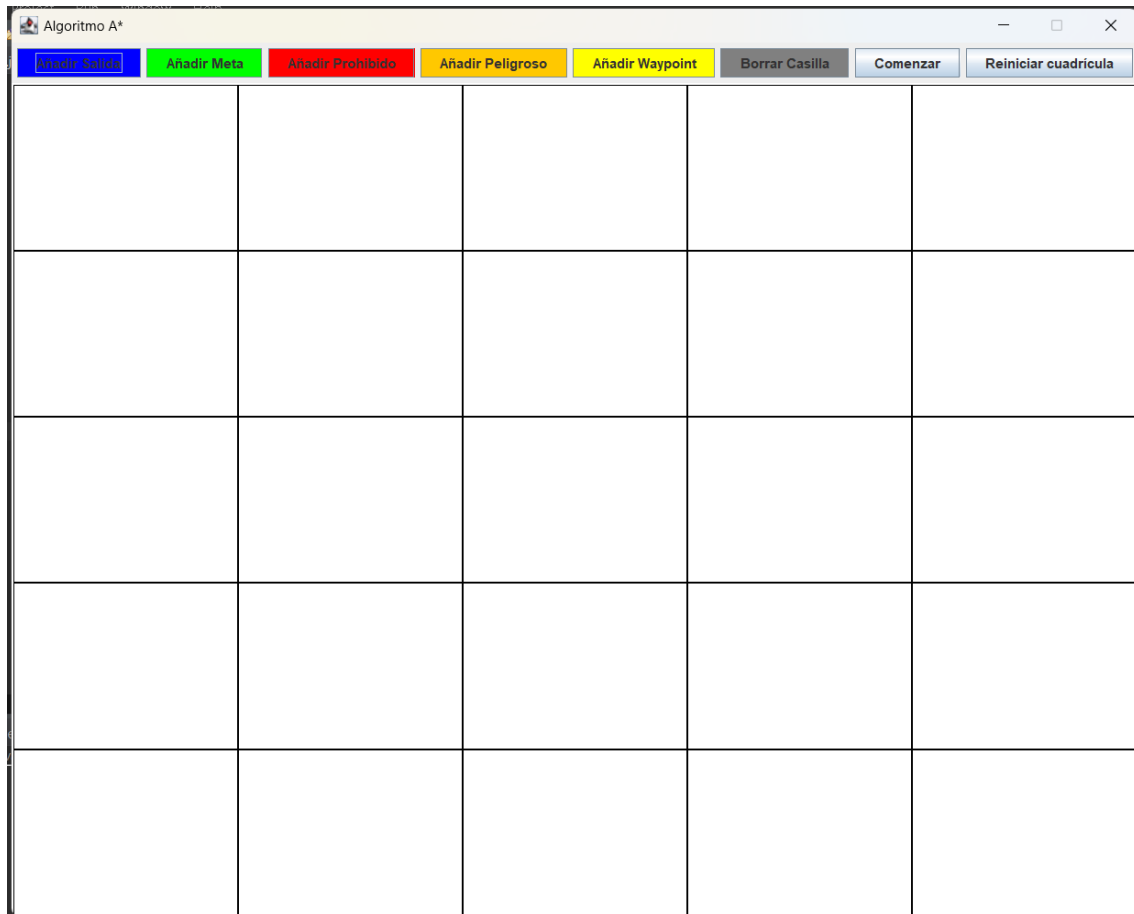


Imagen 5. Pantalla Principal con una cuadrícula de 5x5

Como podemos observar en la parte superior de nuestra pantalla, tenemos un total de ocho botones. Los seis primeros sirven para modificar el tipo de nodo de una de las casillas. El botón de “Borrar Casilla” pondrá el tipo de nodo a VACIO. El botón de “Comenzar” iniciará el algoritmo y el botón de “Reiniciar Cuadrícula” pondrá y pintará todos los nodos de tipo VACIO.

Para poder iniciar el algoritmo, es necesario que haya colocados tanto una SALIDA como una META, si no, mostrará un mensaje de error diciendo que falta alguna de estas dos casillas, como se muestra en la Imagen 6.

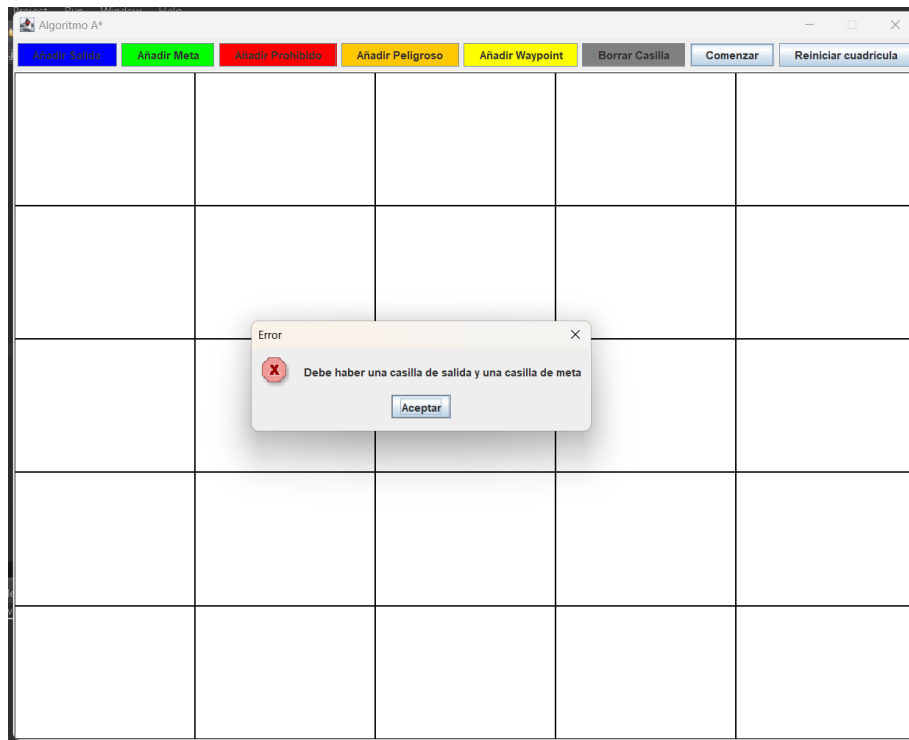


Imagen 6. Ejemplo de mensaje de error si faltan alguna de las casillas de SALIDA o META

Una vez colocadas las casillas de SALIDA y META, podremos darle al botón de “Comenzar”. Al pulsarle, se pintará el recorrido que según el algoritmo se debe seguir, como se muestra en la Imagen 7.

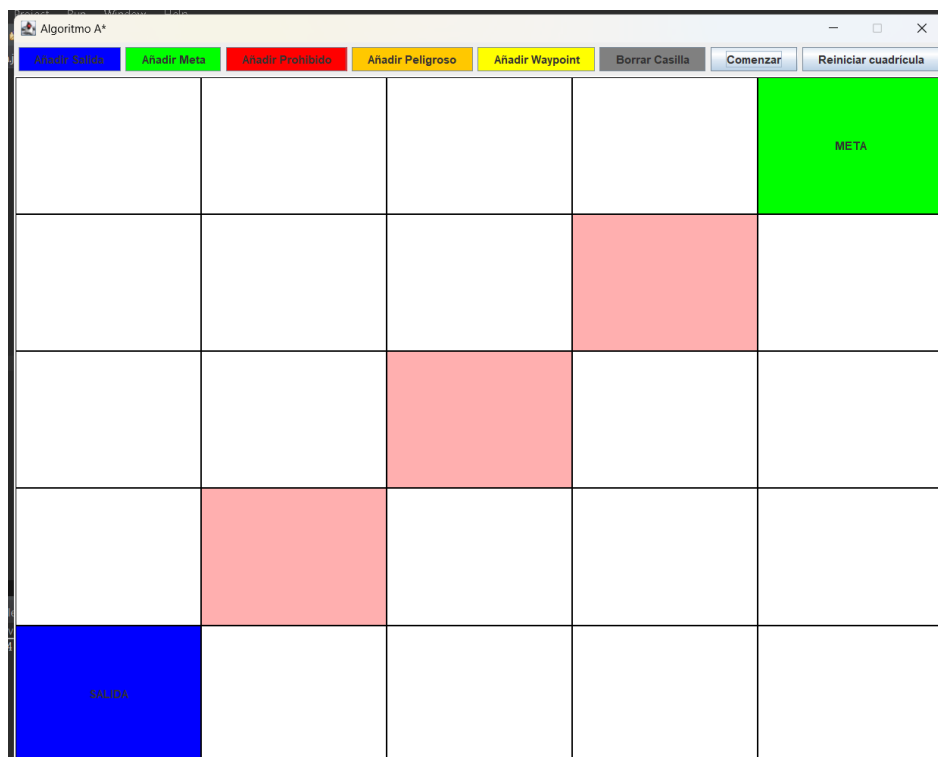


Imagen 7. Ejemplo de ejecución del algoritmo

También podemos ejecutar el algoritmo utilizando otro tipo de casillas, como PROHIBIDAS, PELIGROSAS O WAYPOINTS. Para ello, simplemente hay que pulsar en el botón del tipo de nodo que se quiere implementar y a continuación en la casilla donde se quiere implementar dicho tipo. En las Imágenes 8 y 9, podemos ver un ejemplo y la resolución del algoritmo utilizando este tipo de nodos.

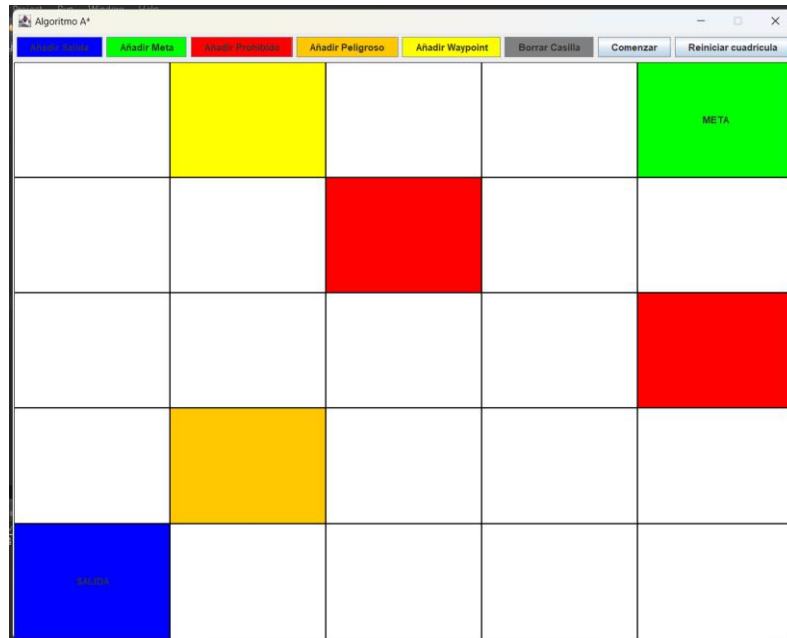


Imagen 8. Ejemplo de uso de los tipos de nodos PROHIBIDO, PELIGROSO y WAYPOINT

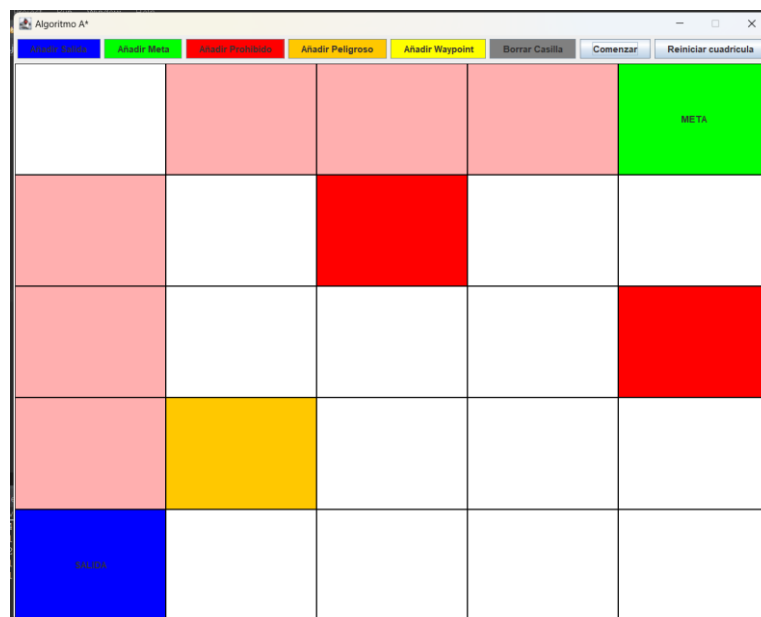


Imagen 9. Ejemplo de resolución del algoritmo con casillas con tipos de nodo PROHIBIDO, PELIGROSO y WAYPOINT

Si la META fuera inalcanzable, aparecería un mensaje informativo indicándonos que es inalcanzable, tal y como se muestra en la Imagen 10.

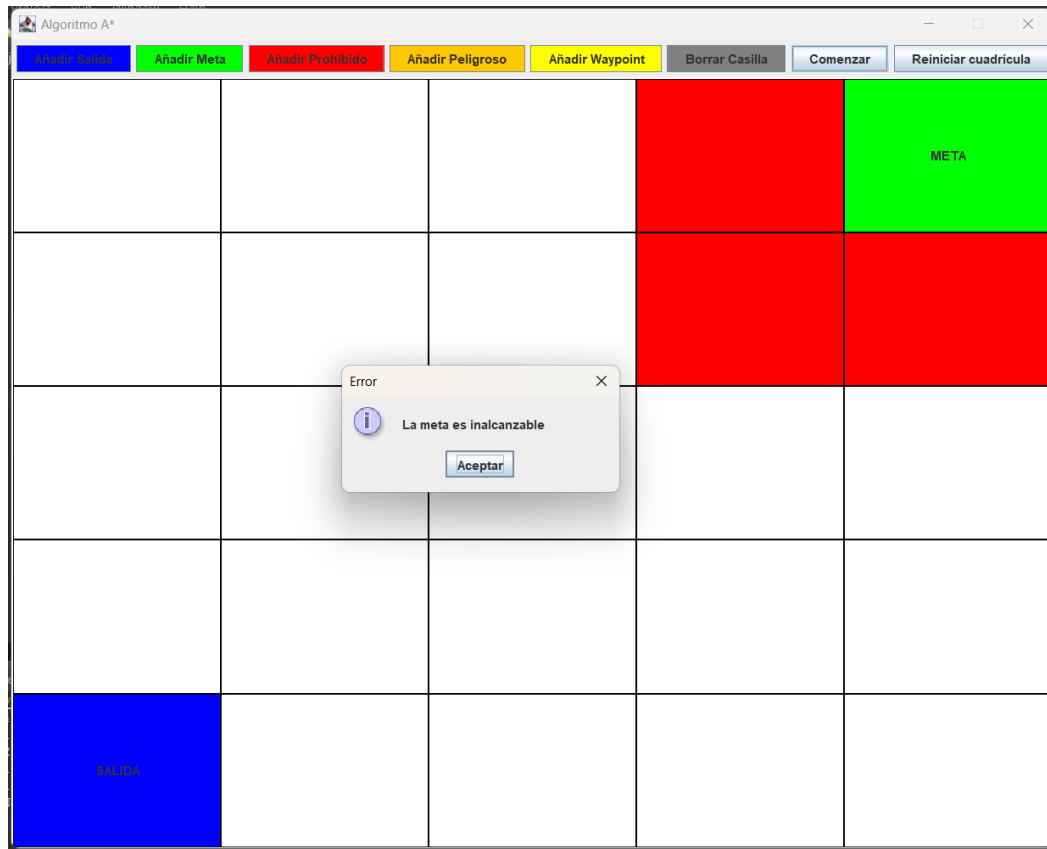


Imagen 10. Ejemplo de META inalcanzable