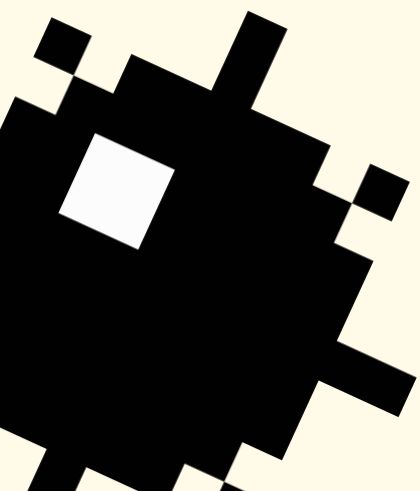


INTELIGENCIA ARTIFICIAL I - 2023

ANTEPROYECTO INTEGRADOR

Hecho por Agustín Yornet y Gabriel Magione



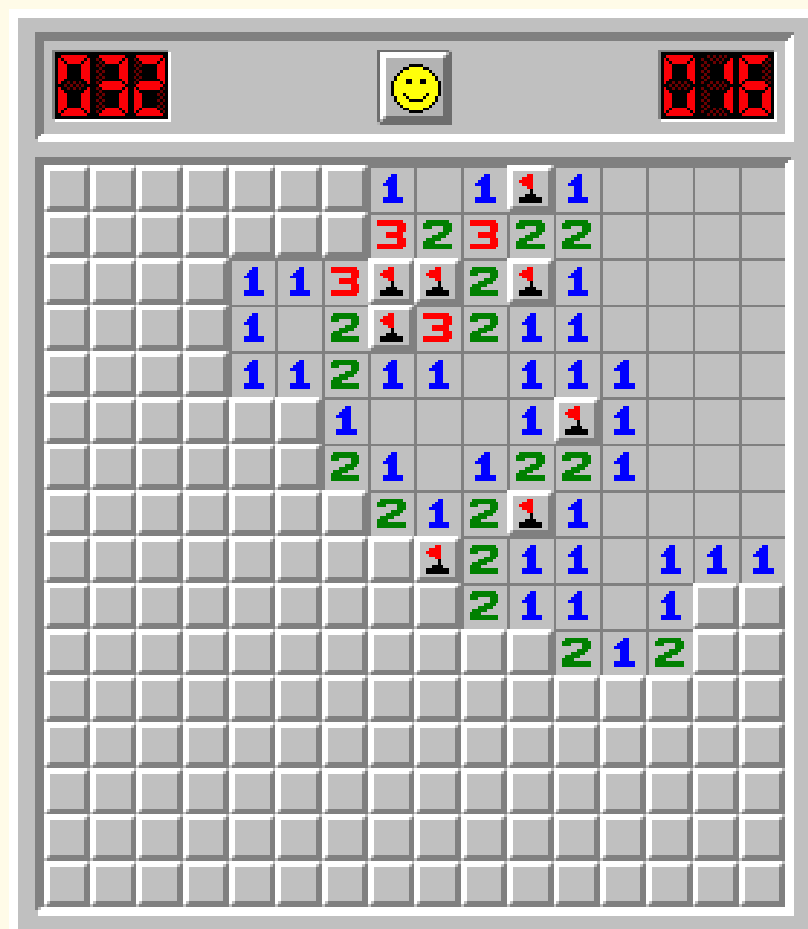
INTRODUCCIÓN

“Buscaminas” es un videojuego para un jugador creado por Curt Johnson y Robert Donner en 1989.

El objetivo del juego es despejar un campo de minas sin detonar ninguna. Algunas casillas tienen un número que indica la cantidad de minas que son adyacentes a la casilla despejada. Así, si una casilla tiene el número 3, significa que de las ocho casillas que hay alrededor, 3 contienen minas y 5 no contienen ninguna. Si se descubre una casilla sin número significa que ninguna de las casillas vecinas tiene mina y estas se descubren automáticamente.

Si se descubre una casilla con una mina se pierde la partida.

Se puede poner una marca en las casillas para indicar que las mismas contienen o pueden contener minas.



PROPUESTA DE PROYECTO

Hemos observado que el problema de “resolver una partida de Buscaminas” puede ser tratado como un Problema de Satisfacción de Restricciones (Constraint Satisfaction Problem).

Un CSP es un problema matemático y computacional que implica encontrar una solución que cumpla con una serie de restricciones dadas. En el caso del Buscaminas, las restricciones son las pistas numéricas que indican cuántas minas están presentes en las casillas adyacentes a una celda en particular. Resolver el Buscaminas implica asignar un valor (mina o seguro) a cada variable (celda) de manera que se cumplan todas las restricciones dadas y se descubran todas las casillas seguras sin detonar ninguna mina.

Dominios de variables:



Restricciones:



Buscamos resolver juegos de Buscaminas por medio de técnicas como Backtracking, Forward checking, Propagación de Restricciones (AC-3) para observar y analizar cuál es la mejor estrategia para resolverlos.

Consideramos que este problema es un desafío porque, si bien todas las partidas son solubles, no en todos los casos el jugador puede determinar una solución; por ende las partidas se pueden clasificar en **guess** y **no-guess**. Nuestra tarea va a consistir en maximizar tanto como sea posible el desempeño de nuestro agente frente a agentes estocásticos y otros algoritmos.

