**Sumario**

[Trasfondo 1](#__RefHeading___Toc106_2468051652)

[Descripción 1](#__RefHeading___Toc280_2468051652)

[Escenario 1](#__RefHeading___Toc282_2468051652)

[Objetivos 3](#__RefHeading___Toc302_2468051652)

[Requerimientos 3](#__RefHeading___Toc304_2468051652)

[Evaluación 3](#__RefHeading___Toc307_2468051652)

# Trasfondo

Las actividades de búsqueda y rescate de víctimas en catástrofes a gran escala son problemas sociales de gran relevancia y, desde un punto de vista científico, plantean numerosos problemas técnicos diferentes en los campos de la inteligencia artificial, la robótica y los sistemas multiagente. Por esta razón, es importante el desarrollo de herramientas que permitan explorar estrategias adecuadas para este tipo de actividades.

# Descripción

El reto de este bloque será implementar una simulación basada en agentes del juego de mesa “Flash Point: Fire Rescue” de Indie Boards & Cards[[1]](#footnote-2). Por esta razón, las reglas de esta simulación están basadas en las reglas del juego de mesa.

## Escenario

El espacio de la simulación es un espacio bidimensional no toroidal de 6 x 8 celdas cuya descripción se recibe a través de un archivo de texto.

El archivo empieza con 6 líneas de 8 grupos de 4 dígitos. Cada grupo de 4 dígitos representa una celda del espacio; cada dígito representa las paredes de una celda (arriba, izquierda, abajo, derecha). El número 1 representa que si tiene pared, el 0 que no tiene. Por ejemplo, 1010 indica que la celda sólo tiene pared arriba y abajo.

A continuación, los marcadores de puntos de interés, 3 líneas de tres elementos. El primer elemento, un número entero, representa el renglón. El segundo elemento, un número entero, representa la columna. Y el tercer elemento, un carácter, indicando si es una víctima (v) o una falsa alarma (f).

Enseguida, los marcadores de fuego, 10 líneas de 2 elementos. El primer elemento, un número entero, representa el renglón. El segundo elemento, un número entero, representa la columna.

Posteriormente, los marcadores de puerta, 8 líneas con 4 números enteros, r1, c1, r2, c2, indicando las dos celdas que conecta una puerta en particular.

Por último, los puntos de entrada, 4 líneas de 2 números enteros, indicando el renglón y la columna de los puntos de entrada.

Figura 1: Ejemplo de configuración inicial

El ejemplo que se muestra a continuación muestra la configuración inicial de la Figura 1.

arriba, izquierda, abajo, derecha)

1100 1000 1001 1100 1001 1100 1000 1001

0100 0000 0011 0100 0011 0110 0010 0011

0100 0001 1100 1000 1000 1001 1100 1001

0100 0011 0110 0010 0010 0011 0110 0011

1100 1000 1000 1000 1001 1100 1001 1101

0110 0010 0010 0010 0011 0110 0011 0111

2 4 v

5 1 f

5 8 v

2 2

2 3

3 2

3 3

3 4

3 5

4 4

5 6

5 7

6 6

1 3 1 4

2 5 2 6

2 8 3 8

3 2 3 3

4 4 5 4

4 6 4 7

8 5 8 6

8 7 8 8

1 6

3 1

4 8

6 3

# Objetivos

Desarrolla un modelo multiagente con una visualización en 3D que permita rescatar a las victimas atrapadas dentro de edificios en llamas antes de que el fuego quede fuera de control o que el edificio se desplome.

# Requerimientos

* Como indicamos anteriormente, esta simulación se basará en las reglas del juego antes mencionado. En este misma carpeta, encontrarás las reglas oficiales del juego. Es importante que tu implementación respete estas reglas. Específicamente, usaremos las reglas llamadas “Family Game Setup”.
* La simulación inicia 6 agentes bombero, un total de 10 marcadores de posibles victimas, un total 5 marcadores de falsa alarma y un total 24 contadores de daño. Recuerda descontar de este total, los marcadores de interés (víctimas o falsas alarmas que recibes en el archivo de entrada).
* Puede haber más de un agente por celda.
* Un agente sólo puede cargar una víctima.
* La simulación termina cuando se gane o pierda el escenario. El escenario se gana cuando los agentes han rescatado 7 víctimas. El escenario se pierde cuando se han perdido 4 víctimas o cuando el edificio colapsa.
* Todo lo que suceda en esta simulación se debe visualizar en Unity con una vista superior (2D) utilizando modelos en 3D. Solo se muestra una parte de escenario y debe ser posible desplazarse la cámara por todo el escenario.

# Evaluación

* Algoritmo, 25%.
* Rendimiento, 25%.
* Cliente-Servidor. 15%.
* Visualización Gráfica 3D, 15%.
* Narrativa, 10%.
* Innovación, 5%.
* Presentación final, 5%.

1. https://indieboardsandcards.com/our-games/flash-point-fire-rescue/ [↑](#footnote-ref-2)