

Tema 2. Agentes basados en Objetivos

CUESTIONES

1. ¿Quién es el Agente Inteligente: un humano, una máquina, el programador, ...?. Describe sus características principales.
2. En este tema se han tratado los Agentes Inteligentes que resuelven problemas de búsqueda en entornos (selecciona las opciones correctas):
 - Estáticos
 - Dinámicos
 - Observables
 - Parcialmente Observables
 - Deterministas
 - No Deterministas
 - Continuos
 - Discretos
3. Busca de entre las propiedades anteriores las que corresponden a cada una de las siguientes definiciones:
 - La formalización del problema y la búsqueda de la solución se realiza sin tener en cuenta los cambios que puedan ocurrir a posteriori.
 - Se conoce el estado inicial.
 - Es posible enumerar todas las líneas de acción posibles.
 - El siguiente estado está totalmente determinado por el estado actual y la acción posible a tomar.
4. ¿Por qué es necesario realizar una abstracción del problema cuando éste se está formalizando?
5. Razona por qué en el problema de las ciudades rumanas, si las acciones fueran viajar en Avión en vez de por carretera entre ciudades que tuvieran aeropuerto, sería insuficiente la información con la que contamos actualmente en el problema. ¿Qué información sería irrelevante? ¿Podría resolverse cualquier caso que se planteara de ir de una ciudad a otra? ¿Qué restricciones habría que establecer?
6. Indica qué elementos son necesarios definir para la formalización de un problema mediante búsqueda en un espacio de estados.
7. En el tema 3 de Russel & Norvig: IA un enfoque moderno, (*Tema 3: Resolver problemas mediante búsqueda*) entre los componentes de un problema bien definido se describe la función Sucesor. ¿A qué elementos correspondería esta función sucesor según lo que se ha estudiado en nuestras implementaciones de clase?
8. Se pretende que un agente inteligente resuelva el siguiente problema del 8 puzle. Indica si las siguientes tareas las realiza el humano que programa al agente inteligente, o si las realiza directamente el agente inteligente.

1	2	3
0	4	5
6	7	8

Inicial

1	2	3
8	0	4
7	6	5

Final

Tarea	Agente Inteligente	Humano
Dar la solución al problema		
Establecer el estado inicial		
Establecer el estado final		
Comprobar si se ha llegado a un estado final		
Comprobar si un movimiento es válido		
Programar la función esVálido		
Generar el próximo movimiento válido		

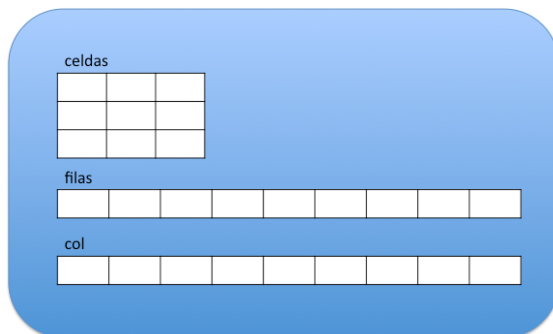
TÉCNICAS DE BÚSQUEDA NO INFORMADA

9. Dadas la formalización del problema de las Jarras y del problema del 8-puzle. Estudia detenidamente cada algoritmo para identificar todas las similitudes y diferencias que veas entre ambos problemas.

En concreto indica si las siguientes afirmaciones son verdaderas o falsas y justifica por qué debe ser así.

¿Verdadero o Falso?	V	F
El número de operadores, viene dado siempre con la misma constante: NUM_OPERADORES, pero su valor es diferente, dependiendo del problema.		
Sólo en el problema de las jarras existe un tipo de datos que se llama tEstado.		
El prototipo de la función aplicaOperador es la misma en los dos problemas, sin embargo los prototios de la función esValido cambia de un problema a otro.		
El conjunto de instrucciones de la función esValido es diferente en cada problema.		
El conjunto de instrucciones de la función testObjetivo es el mismo en cada problema.		

10. En el problema del 8-puzle la estructura de una variable de tipo tEstado podría representarse gráficamente así:



```

const
  N Cel=3
tipo
  matriz[N Cel,N Cel] de entero: mat
  vector[N Cel*N Cel] de entero: vect
registro: tEstado
          mat: celdas
          vect: filas, col
fin_registro

```

Dados los siguientes puzles inicial y final, y suponiendo tres variables de tipo tEstado: actual y sucesor. Indica el contenido de cada variable tras realizar las siguientes asignaciones:

<table><tr><td>1</td><td>2</td><td>3</td></tr><tr><td>0</td><td>4</td><td>5</td></tr><tr><td>6</td><td>7</td><td>8</td></tr></table> <p>Inicial</p>	1	2	3	0	4	5	6	7	8	<pre>op←ARRIBA actual ← crearEstado(puzle_inicial) si esValido(op, actual) entonces sucesor ← aplicaOperador(op, actual) fin_si</pre>
1	2	3								
0	4	5								
6	7	8								

¿Qué hace la función crearEstado? ¿Qué se almacena en los vectores filas y col?

TÉCNICAS DE BÚSQUEDA NO INFORMADA




PROBLEMAS DE AUTOEVALUACIÓN

1. EL PROBLEMA DE LAS 8 REINAS

En el tema 3 de Russel & Norvig: IA un enfoque moderno, (*Tema 3: Resolver problemas mediante búsqueda*) se describe una formalización del problema de las 8-reinas. Modifica esta formalización para que se adapte a la estructura vista en clase (funciones esValido, aplicaOperador, testObjetivo, etc.)

2. EL ROBOT Y LOS LADRONES

Un dispositivo automático está almacenando sacos de dinero en una caja fuerte. Los sacos están en fila y conectados por un hilo de alarma, de forma que sólo pueden retirarse sacos de los extremos de la fila sin disparar la alarma. Cada saco tiene impresa la cantidad en millones de dólares.

		2\$	5\$	7\$	1\$	8\$	
---	---	-----	-----	-----	-----	-----	---

Los ladrones pueden retirar y transportar los sacos rápidamente a la furgoneta de uno en uno, pero cada vez que cogen un saco el dispositivo se apodera del que ha quedado en el extremo derecho y lo pone fuera del alcance de los ladrones.



Los ladrones quieren coger la máxima cantidad de dinero posible sin accionar la alarma. Por suerte cuentan con un estudiante de Inteligencia Artificial.

Realiza la formalización del problema (en pseudocódigo).

Describe el estado inicial, el estado final, la función testObjetivo y el conjunto de operadores para este problema (funciones esValido y aplicaOperador).

IMPLEMENTACIONES

Implementa en C las formalizaciones realizadas hasta ahora: el problema de las Jarras, el problema de las 8-Reinas y el Robot y los Ladrones. En la función main de cada programa escribe las instrucciones necesarias para comprobar que todas las funciones implementadas hasta ahora funcionan.

En el próximo tema estudiaremos cómo programar al Agente Inteligente para que proporcione la solución a estos problemas que le planteamos.