

Inteligencia Artificial

Evaluación Continua, Grupo 81

28/02/14

Nombre y Apellidos.....

Ejercicio 1

Se pretende diseñar un sistema de control de una "aspiradora robot" que se desplaza por una habitación y simular su funcionamiento e interacción con el entorno empleando reglas de producción. Se supondrá que el entorno en el que actuará el robot es una habitación cuadrada dividida en cuadrículas. Esto es, tendrá $N \times N$ cuadrículas o casillas. Cada una de esas casillas se le asociará un único contenido. Estos contenidos pueden ser:

- Un obstáculo: una casilla donde el robot no puede penetrar.
- Una zona sucia: una casilla que el robot debe limpiar.
- Una papelera: una casilla donde el robot debe depositar la suciedad recogida en alguna zona sucia.

Supondremos que si no se especifica nada sobre una casilla, esta no contendrá ningún objeto, ni ningún obstáculo, y estará limpia. El robot puede realizar las siguientes acciones:

- **Moverse 1 casilla arriba, abajo, izquierda, derecha:** El robot que se encuentra en una determinada casilla se mueve a una casilla adyacente siempre que no sea un obstáculo.
- **Aspirar:** El robot con su depósito de suciedad vacío se encuentra en una casilla sucia y la aspira. El depósito de suciedad del robot pasará a estar lleno y la casilla dejará de estar sucia.
- **Vaciar depósito:** Si el robot tiene el depósito de suciedad lleno y se encuentra en una casilla con una papelera, vacía su depósito.

Diseñar un sistema de producción que determine la secuencia de acciones del brazo de robot para llegar a una configuración de bloques determinada partiendo de una configuración inicial. Para ello habrá que especificar:

1. Los predicados con los cuales el robot puede realizar las diferentes operaciones.
2. Las reglas que describen las operaciones descritas en el enunciado (moverse y aspirar). Para ello, habrá que emplear los predicados definidos en el punto anterior.

Ejercicio 2

En un sistema de producción tenemos las siguientes reglas:

R1(X): SI A(X) Y B(X) ENTONCES E(X)

R2(X,Y): SI C(X) Y B(Y) Y A(Y) ENTONCES D(X)

R3(X,Y): SI A(X) Y E(Y) ENTONCES F(Y)

R4(X): SI D(X) Y F(X) ENTONCES G(X)

La base de hechos contiene los siguiente hechos al inicio: A(alberto), B(alberto), C(juan)

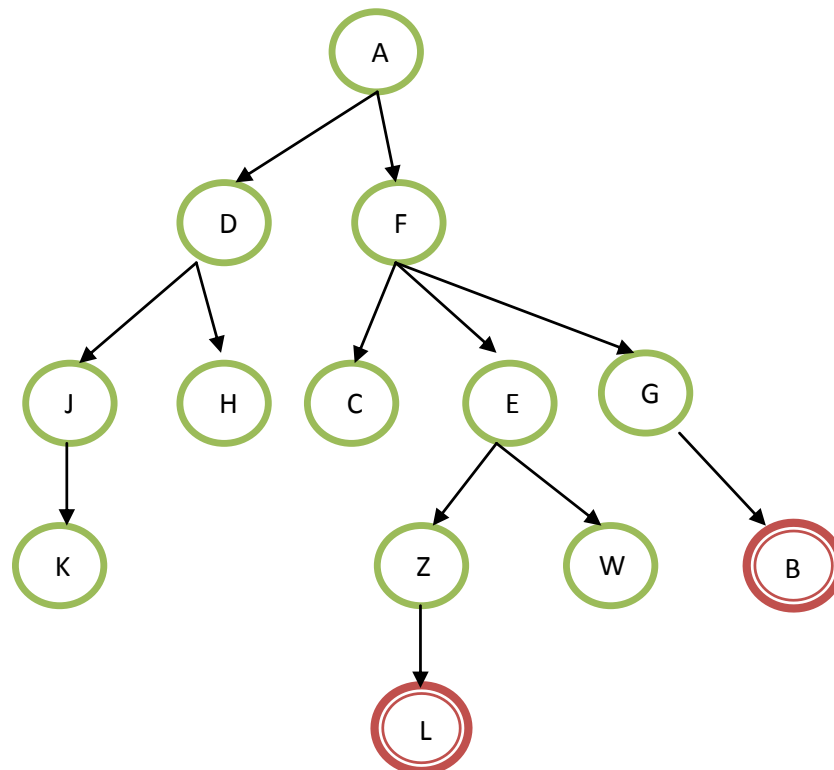
Si el sistema opera con una estrategia de resolución de conflictos en PROFUNDIDAD, mostrar la secuencia de reglas ejecutadas y datos en la base de hechos en una tabla.

Ejercicio 3

Dado el árbol de la figura 1 donde B y L son los 2 únicos nodos meta y A es el nodo inicial, indica en qué orden se visitarían los nodos, indicando en una tabla el contenido de la lista abierta, los nodos expandidos y los nodos generados en cada momento, para los siguientes algoritmos:

1. Amplitud
2. Profundidad

El orden de generación de los sucesores es alfabético.



Ejercicio 4

En la orilla izquierda de un río existen 5 misioneros, 2 caníbales de la tribu A y 2 caníbales de la tribu B. Existe una barca que permite cruzar el río y que admite sólo

dos personas, pudiendo hacer el trayecto de orilla a orilla. Como restricciones se tienen:

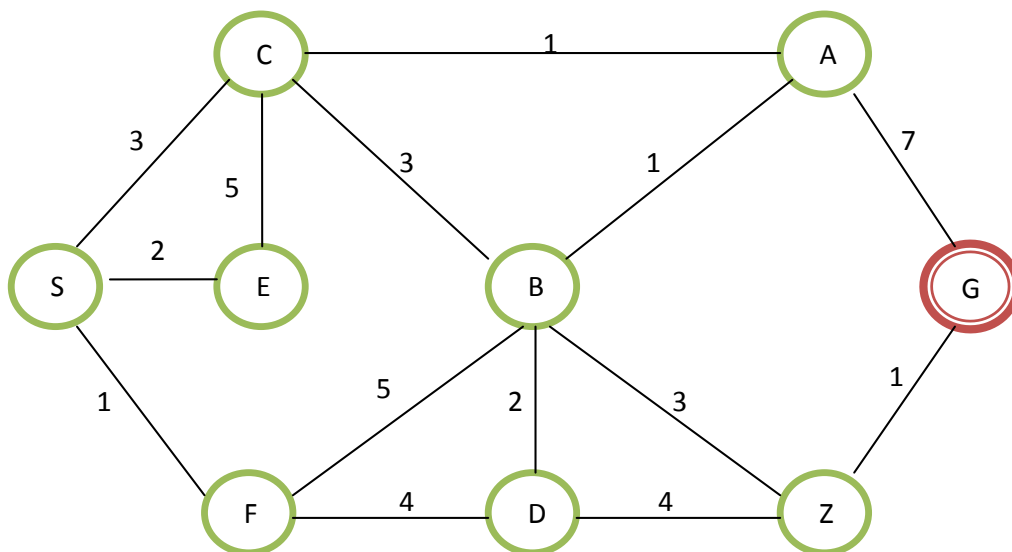
- Los misioneros no pueden verse superados en número en ninguna orilla.
- En el caso en que en alguna orilla no haya misioneros, nunca debe superar el número de caníbales de un tipo al del otro tipo, ya que en caso contrario unos se comerían a los otros.
- La barca puede ser conducida tanto por misioneros como por caníbales y se requiere al menos un individuo para manejarla.

El objetivo del problema es pasar los caníbales y misioneros de la orilla izquierda a la orilla derecha.

1. ¿Cómo puede representarse cada estado? Explicar la decisión tomada.
2. ¿Cuáles son los operadores para la representación escogida?
3. Generar el árbol de búsqueda con los 3 primeros movimientos utilizando una estrategia en PROFUNDIDAD partiendo de un estado inicial en el que todos los individuos, misioneros y caníbales tanto de una tribu como de la otra, están en la orilla izquierda del río. No es necesario dibujar los estados que no cumplen con las restricciones del enunciado.

Ejercicio 5

Recorrer el grafo de la siguiente figura utilizando el algoritmo de búsqueda heurística A* teniendo en cuenta los costes indicados en cada arco.



Generar el árbol de búsqueda indicando para cada nodo el valor $g(n)$, $h(n)$ y $f(n)$. Tomar como nodo inicial el nodo S y como único estado meta el nodo G. Cada nodo del grafo tiene el valor heurístico descrito en la siguiente tabla.

n	S	A	B	C	D	E	F	G	Z
$h(n)$	8	2	3	5	4	6	6	0	1

1.) 1. $\text{robot}(x, y, D)$ x : pos x y : pos y D : Capacidad

$\text{maxC}(M)$ M : Capacidad máxima robot

$\text{celda}(x, y, O)$ x : pos x y : pos y

O : situación celda \in {sucio, obstaculo}

2. 3º Moverse 1A_{rr}: $\text{robot}(x, y, D), \text{celda}(x, y+1, C) \rightarrow \text{robot}(x, y+1, D)$
 $C \neq \text{"block"}$

Moverse 1A_b: $\text{robot}(x, y, D), \text{celda}(x, y-1, C) \rightarrow \text{robot}(x, y-1, D)$
 $C \neq \text{"block"}$

Moverse 1I_{zq}: $\text{robot}(x, y, D), \text{celda}(x-1, y, C) \rightarrow \text{robot}(x-1, y, D)$
 $C \neq \text{"block"}$

Moverse 1A_{ch}: $\text{robot}(x, y, D), \text{celda}(x+1, y, C) \rightarrow \text{robot}(x+1, y, D)$
 $C \neq \text{"block"}$

1º Aspirar: $\text{robot}(x, y, D), \text{celda}(x, y, \text{"sucio"}) \rightarrow \text{robot}(x, y, D+1)$
 $D < M \quad \text{maxC}(M)$
 $\text{celda}(x, y, \text{"limpio"})$

2º Vaciar: $\text{robot}(x, y, D), \text{celda}(x, y, \text{"basura"}) \rightarrow \text{robot}(x, y, 0)$

2.) Profundidad. LIFO

$$MT_0 = \{A(\text{alberto}), B(\text{alberto}), C(\text{juan})\}$$

$$CC_0 = \{R1(x=\text{alberto}), R2(x=\text{juan}, y=\text{alberto})\} \quad \text{Ejemplamos } R2(x=\text{juan}, y=\text{alberto})$$

$$MT_1 = \{A(\text{alberto}), B(\text{alberto}), C(\text{juan}), D(\text{juan})\}$$

$$CC_1 = \{R1(x=\text{alberto})\} \quad \text{Ejemplamos } R1(x=\text{alberto})$$

$$MT_2 = \{A(\text{alberto}), B(\text{alberto}), C(\text{juan}), D(\text{juan}), E(\text{alberto})\}$$

$$CC_2 = \{R3(x=\text{alberto}, y=\text{alberto})\}$$

$$MT_3 = MT_2 \cup \{F(\text{alberto})\}$$

3.)

1. Amplitud: FIFO, el orden de entrada es el de salida.

Lista Abierta	Nodos expandidos	Nodos generados
<u>A</u>	A	D, F
D, F	A, D	J, H
<u>F</u> , J, H	A, D, F	C, E, G
J, H, C, E, G	A, D, F, J	k
H, C, E, G, k	A, D, F, J, H	
C, E, G, k	A, D, F, J, H, C	
E, G, k	A, D, F, J, H, C, E	z, w
<u>G</u> , k, z, w	A, D, F, J, H, C, E, G	B
k, z, w, <u>B</u>	<u>A</u> , <u>D</u> , <u>F</u> , <u>J</u> , <u>H</u> , <u>C</u> , <u>E</u> , <u>G</u> , <u>B</u>	Se ha encontrado B que es final.

2. Profundidad: LIFO

Lista abierta

A

D, F

D, C, E, G

D, C, E, B

Nodos expandidos

A

A, F

A, F, G

A, F, G, B

Nodos generados

D, F

C, E, G

B

Fin encontrado en final

5.)

A*

