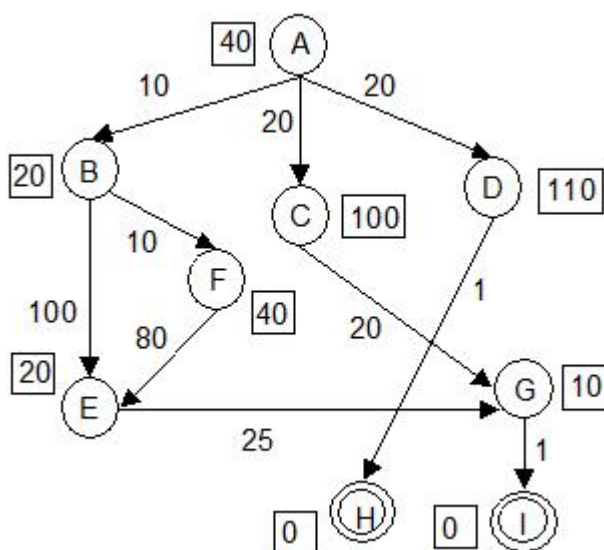


Universidad Nacional de Educación a Distancia
Ingeniería Técnica en Informática de Sistemas
Introducción a la Inteligencia Artificial (2º curso)

Soluciones exámenes junio 2010

Semana 1. Ejercicio 1. (Valoración: 4 puntos)

En el grafo de la figura, A es el nodo inicial y H e I los nodos meta. Cada arco lleva asociado su coste y en cada nodo aparece la estimación de la menor distancia desde ese nodo a la meta.



(a) (máx. 3 puntos) Aplicar los siguientes algoritmos a este grafo: búsqueda en amplitud, búsqueda primero el mejor y A*. Describir cada uno de los pasos seguidos indicando el contenido de las listas ABIERTA y CERRADA y dibujando en cada etapa del algoritmo el subgrafo parcial creado. Devolver igualmente el camino solución.


(b) (máx. 0.5 puntos) ¿Es monótona la heurística empleada? Razonar la respuesta e indicar qué consecuencias tiene en el proceso de búsqueda cuando se utiliza el algoritmo A*.

(c) (máx. 0.5 puntos) ¿Cuál es el objetivo de una función heurística aplicada a la búsqueda en el espacio de estados?

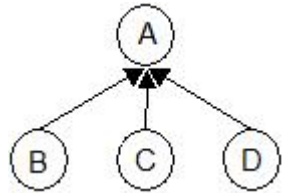
Solución por Elena Gaudioso Vázquez:

(a.1) Búsqueda en amplitud

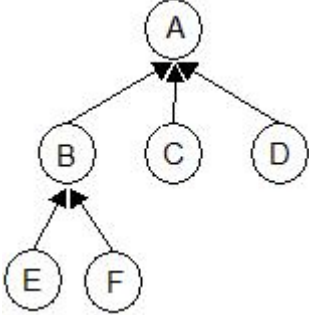
Paso 1: Situación inicial

	COLA={A}
---	----------

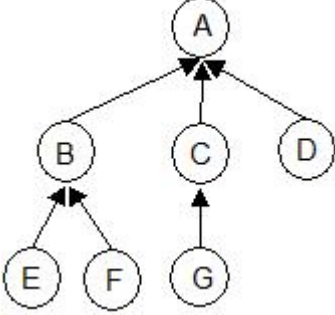
Paso 2: Se expande A

	COLA={B,C,D}
---	--------------

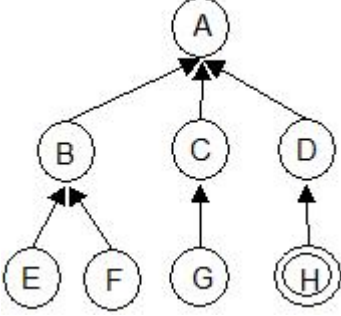
Paso 3: Se expande B

	COLA={C,D,E,F}
---	----------------

Paso 4: Se expande C

	COLA={D,E,F,G}
---	----------------


Paso 5: Se expande D

	COLA={E,F,G,H}
---	----------------

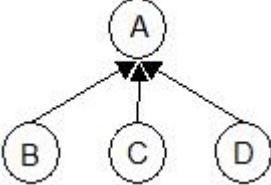
CAMINO SOLUCIÓN: A → D → H

(a.2) Búsqueda primero el mejor

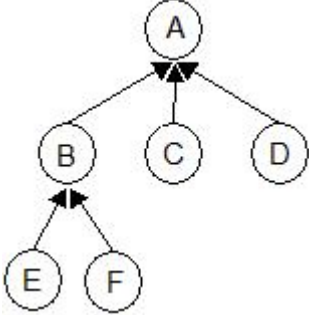
Paso 1: Situación inicial

	ABIERTA={A(40)} CERRADA={}
---	-------------------------------

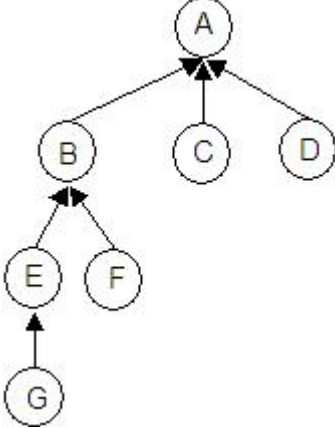
Paso 2: Se expande A

	ABIERTA={B(20),C(100),D(110)} CERRADA={A}
---	--

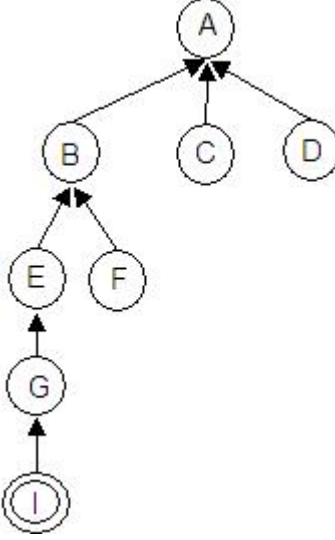
Paso 3: Se expande B

	ABIERTA={E(20),F(40),C(100),D(110)} CERRADA={A,B}
---	--

Paso 4: Se expande E

	ABIERTA={G(10),F(40),C(100),D(110)} CERRADA={A,B,E}
--	--


Paso 5: Se expande G

	ABIERTA={F(40),C(100),D(110)} CERRADA={A,B,E,G}
---	--

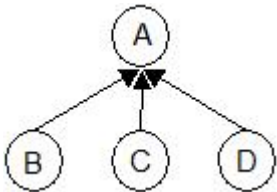
CAMINO SOLUCIÓN: A → B → E → G → I

(a.3) Búsqueda A*

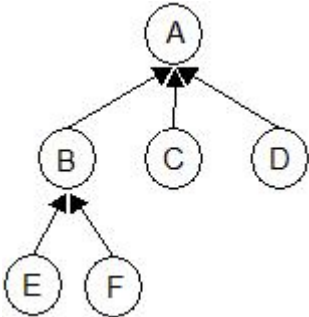
Paso 1: Situación inicial

	ABIERTA={A(40)} CERRADA={}
---	-------------------------------

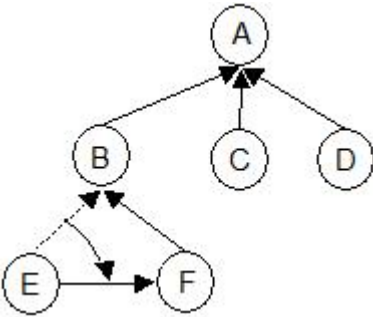
Paso 2: Se expande A

	ABIERTA={ <u>B(30)</u> ,C(120),D(130)} CERRADA={A}
---	---

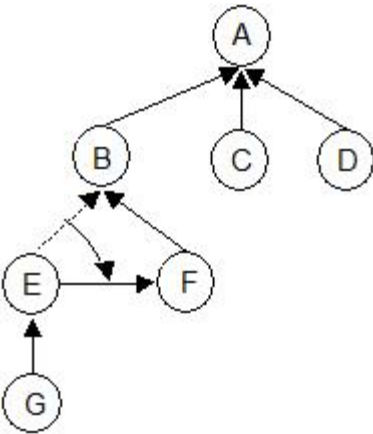
Paso 3: Se expande B

	ABIERTA={ <u>F(60)</u> ,C(120),E(130),D(130)} CERRADA={A,B}
---	--

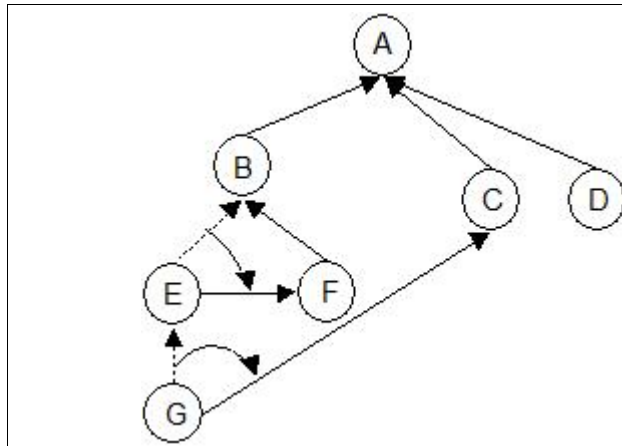
Paso 4: Se expande F

	ABIERTA={ <u>E(120)</u> ,C(120),D(130)} CERRADA={A,B,F}
--	--

Paso 5: Se expande E

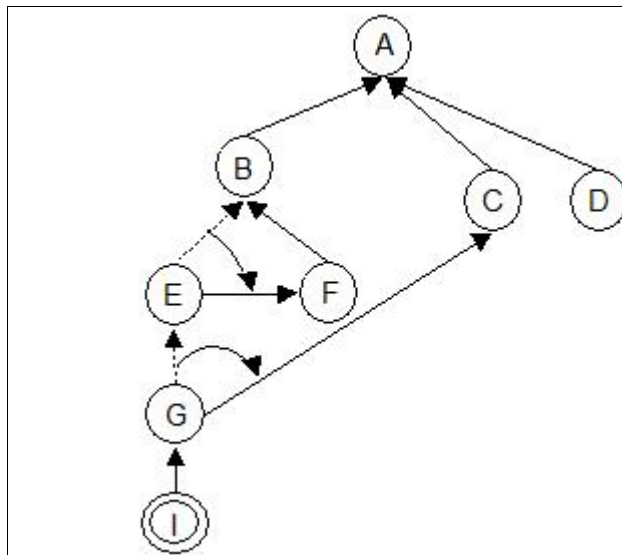
	ABIERTA={ <u>C(120)</u> ,D(130),G(135)} CERRADA={A,B,F,E}
---	--

Paso 6: Se expande C



ABIERTA={G(50),D(130)}
CERRADA={A,B,F,E,C}

Paso 7: Se expande G



ABIERTA={I(0),D(130)}
CERRADA={A,B,F,E,C,G}

Paso 8: Se expande I

El proceso finaliza cuando se expande el nodo I que es un nodo meta

CAMINO SOLUCIÓN: $A \rightarrow C \rightarrow G \rightarrow I$

(b)

La función de evaluación heurística no es monótona puesto que no se cumple que:

$$\square (n,n') \quad h(n) \leq \text{coste}(n,n') + h(n')$$

donde n' es un nodo sucesor de n y $\text{coste}(n,n')$ representa el coste del enlace entre n y n' .

Por ejemplo: $h(A)=40 > (10+h(B))=30$

Como consecuencia, ha sido necesario revisar la existencia de reorientaciones de enlaces en la ejecución del algoritmo A^* .

(c)

Estimar la adecuación de un nodo para ser expandido

Semana 1. Ejercicio 2. (Valoración: 3 puntos)

Responder a las siguientes cuestiones:

- (a) (máx. 1 punto) ¿Cuáles son las diferencias existentes entre la lógica de predicados y la lógica modal? Dé ejemplos prácticos que ilustren su explicación.
- (b) (máx. 1 punto) ¿Cómo podría utilizarse la lógica modal para abordar el tratamiento del razonamiento temporal?
- (c) (máx. 1 punto) Dar una prueba formal del siguiente razonamiento: “*No es posible que Daniel salga a la calle y no vaya Silvia*” “*Por tanto, necesariamente, si Daniel va a la calle, Silvia también*”.

Solución por Elena Gaudioso Vázquez:

(a) La lógica modal constituye una extensión de la lógica de predicados con la capacidad de evaluar argumentos que implican necesidad o posibilidad. Para ello se introducen dos nuevos operadores: $\Box A$ “Es necesario que A” y $\Diamond A$ “Es posible que A”. Necesidad y posibilidad son dos modos de verdad. “Es necesario que A” es más fuerte que “A es cierto” y significa que “A tiene que ser cierto”. Por otro lado “Es posible que A” es más débil que “A es cierto” y significa A podría ser cierto. Para el tratamiento de los conceptos de necesidad y posibilidad, la lógica modal se apoya en la idea de mundo. Un enunciado necesario sería aquél que es cierto en todos los mundos posibles. Un enunciado verdadero es aquél que es cierto en el mundo real. Finalmente, un enunciado posible es aquél que es cierto en algún posible mundo. Un enunciado posible puede o no ser cierto en el mundo real.

- Ejemplos de razonamiento que se pueden realizar con lógica de predicados (apartado 5.3 del libro base de teoría) y lógica modal (apartado 5.5.1 del libro base de teoría)

(b) Consultar problema 3.14 del libro de problemas

(c) Consultar problema 3.13 del libro de problemas

Semana 1. Ejercicio 3. (Valoración: 3 puntos)

¿Son ciertas las siguientes afirmaciones? Razone brevemente su respuesta e ilústrela con ejemplos:

- (a) (**máx. 1 punto**) No existen diferencias entre el método MINIMAX y el método de poda alfa-beta en cuanto al método de búsqueda que emplean.
- (b) (**máx. 1 punto**) Sólo se puede realizar razonamiento de tipo no monótono si se utiliza la lógica no monótona como método de representación de conocimiento e inferencia.
- (c) (**máx. 1 punto**) La complejidad en espacio de la búsqueda en amplitud es mayor que en el caso de la búsqueda en profundidad.

Solución por Elena Gaudioso Vázquez:

(a) Esta afirmación es VERDADERA. El método MINIMAX hace un recorrido exhaustivo del árbol mediante una búsqueda en profundidad. El método de poda alfa-beta, a la vez que realiza una exploración en profundidad del árbol de búsqueda, tiene la posibilidad de realizar podas en el mismo. Para ver dos ejemplos prácticos se puede consultar el problema 1 del examen resuelto de Junio 2007 (2ª semana).

(b) Esta afirmación es FALSA. Este tipo de razonamiento puede implementarse en varios métodos de representación e inferencia:

-- Lógica:

- a) En la “Lógica no monótona” se introduce el operador modal “M”. Ejemplo en la página 227 del libro base de teoría.
- b) En la “Lógica por defecto” se introducen las “reglas por defecto”. Ejemplo en la página 228 del libro base de teoría.
- c) En Prolog, mediante el predicado “retract”. Es similar al caso en que en el formalismo de reglas se introduce la acción “retractar”.

--Reglas:

- Mediante la introducción de la acción “retractar” y la consideración del axioma del mundo cerrado. Ejemplo en el problema 4.4 del libro base de problemas.
- Mediante la definición del tipo de dependencia, reversible o irreversible, en una regla. Ejemplo en la página 243 del libro base de teoría.

--Redes:

- En las redes de clasificación, la llegada de nueva información invalida resultados inferidos en el proceso de herencia por defecto. Es similar a lo que ocurre en el caso de los marcos.

--Marcos: Ejemplo en el problema 3.25 del libro base de problemas.

--Guiones: Ocurre algo similar al caso de los marcos. En este método, la información por defecto se obtendría a partir de la descripción de las escenas hecha en el guión.

(c) Esta afirmación es VERDADERA. En la búsqueda en amplitud la complejidad espacial es de $O(n^p)$ ya que antes de abandonar la generación de todos los sucesores de un nivel se deben almacenar todos los nodos de dicho nivel. Por el contrario en la búsqueda en profundidad, a lo largo del grafo de búsqueda, sólo es necesario guardar constancia del camino construido hasta el momento. Luego la complejidad para un camino de longitud p será dicho valor multiplicado por el factor de ramificación (r), esto es $O(pr)$ ya que se guarda todo el camino hasta la raíz y al expandir cada nodo se generan los r sucesores.

Semana 2. Ejercicio 1. (Valoración: 2.5 puntos)

Explique detalladamente qué diferencias existen entre el formalismo de *reglas* y el de *lógica de predicados*.

Solución por Severino Fernández Galán:

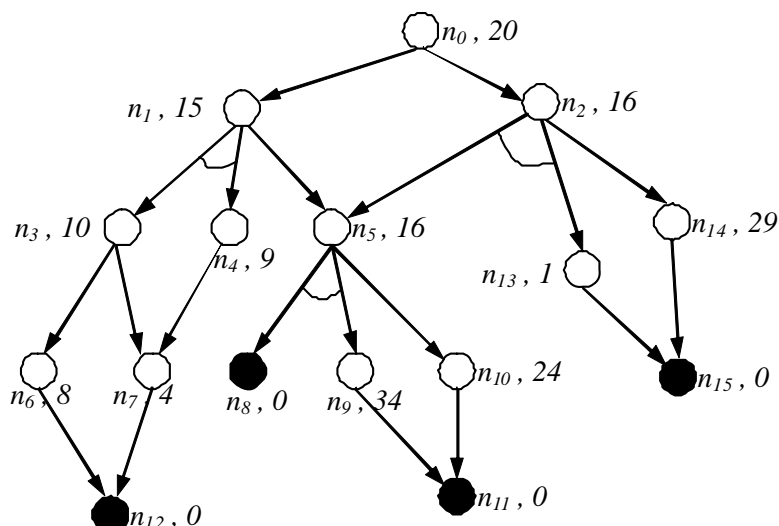
La contestación a esta pregunta debería corresponderse con la explicación dada en la sección 6.7.2 del libro base de teoría. Los puntos principales a tratar son los siguientes:

- Las reglas pueden considerarse como una versión reducida de la lógica de predicados.
- Las reglas reducen la expresividad y la capacidad de inferencia de la lógica de predicados con el fin de lograr mayor eficiencia:
 - No se usa el cuantificador existencial.
 - No se aplica el modus tollens.
 - El encadenamiento de reglas no extrae todas las conclusiones que dicta la lógica, aunque las conclusiones que extrae las extrae de forma mucho más eficiente.
- Una de las limitaciones de la lógica clásica es que no permite tratar la incertidumbre. Mediante reglas sí existen mecanismos para tratar la incertidumbre (factores de certeza de MYCIN o reglas difusas, por ejemplo).

Semana 2. Ejercicio 2. (Valoración: 4 puntos)

Considere el grafo Y/O (también conocido como A/O) de la figura. Describa paso a paso el desarrollo de la exploración de dicho grafo mediante el algoritmo YO* (también conocido como AO*). Para ello suponga que el coste de cada arco es 1 y que se tienen los siguientes valores para la función heurística “ h ” de estimación del coste del subgrafo solución óptimo desde cada nodo:

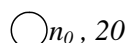
$h(n_0) = 20$	$h(n_3) = 10$	$h(n_6) = 8$	$h(n_9) = 34$	$h(n_{12}) = 0$	$h(n_{15}) = 0$
$h(n_1) = 15$	$h(n_4) = 9$	$h(n_7) = 4$	$h(n_{10}) = 24$	$h(n_{13}) = 1$	
$h(n_2) = 16$	$h(n_5) = 16$	$h(n_8) = 0$	$h(n_{11}) = 0$	$h(n_{14}) = 29$	



Considere que n_0 es el nodo inicial y los nodos terminales o meta son n_8 , n_{11} , n_{12} y n_{15} .

Solución por Severino Fernández Galán:

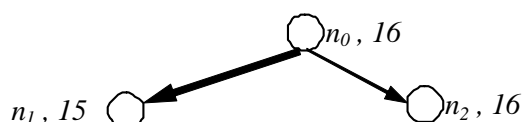
a) Inicialmente:



Seguidamente se describe cada ciclo del algoritmo, en cada uno de los cuales primeramente se expande un nodo hoja cualquiera del subgrafo solución parcial que cuelga de n_0 y a continuación, si es necesario, se actualizan hacia arriba los costes de los subgrafos solución parciales que cuelgan de los nodos antepasados del nodo expandido. Para ello utilizaremos un conjunto denominado S tal que si sacamos un nodo de S y hay que actualizarlo, entonces introducimos en S los padres del nodo actualizado. De S se irán sacando aquellos nodos que no tengan descendientes en S . Inicialmente, el nodo hoja expandido es el único elemento contenido en S .

b) Ciclo 1:

- > Se expande n_0 .
- > La evolución del conjunto S es la siguiente:
 - 1) $S = \{n_0\}$
 - 2) Se saca n_0 de S . Su nuevo coste es $15+1 = 16$.
 - 3) $S = \{\}$
- > La situación final es la siguiente:

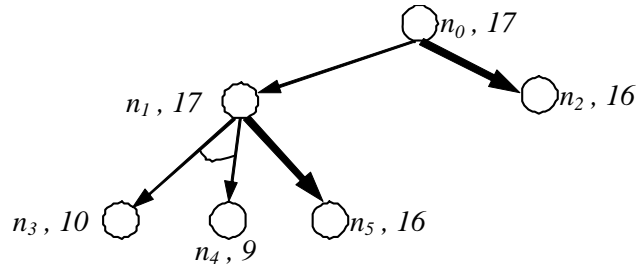


c) Ciclo 2:

- > Se expande n_1 .
- > La evolución del conjunto S es la siguiente:

- 1) $S = \{n_1\}$
- 2) Se saca n_1 de S . Su nuevo coste es 17. Al haber cambiado este coste respecto al antiguo, se introducen los padres de n_1 en S .
- 3) $S = \{n_0\}$
- 4) Se saca n_0 de S . Su nuevo coste es 17 tras la redirección del mejor subárbol que cuelga de él.
- 5) $S = \{\}$

> La situación final es la siguiente:



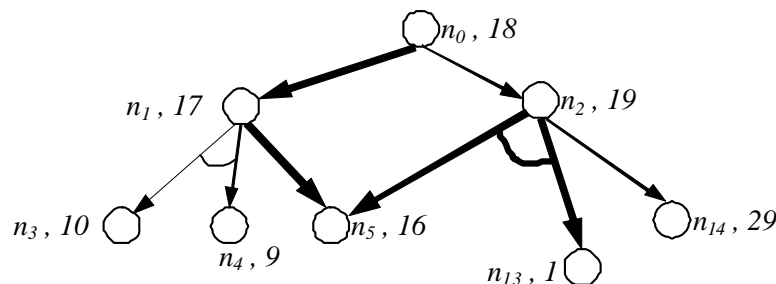
d) Ciclo 3:

> Se expande n_2 .

> La evolución del conjunto S es la siguiente:

- 1) $S = \{n_2\}$
- 2) Se saca n_2 de S . Su nuevo coste es 19. Al haber cambiado este coste respecto al antiguo, se introducen los padres de n_2 en S .
- 3) $S = \{n_0\}$
- 4) Se saca n_0 de S . Su nuevo coste es 18 tras una nueva redirección del mejor subárbol que cuelga de él.
- 5) $S = \{\}$

> La situación final es la siguiente:



e) Ciclo 4:

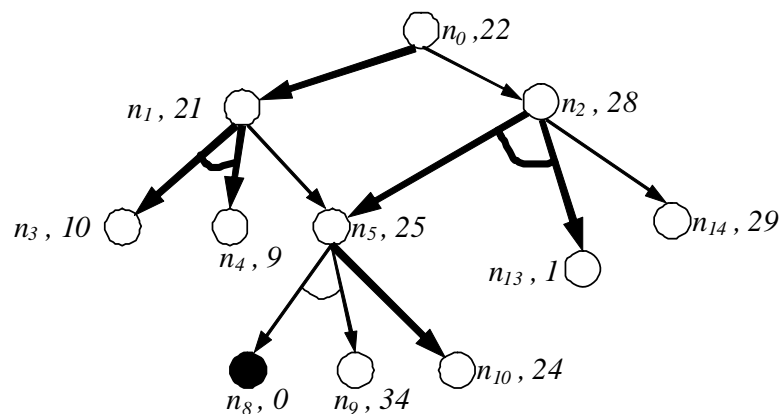
> Se expande n_5 .

> La evolución del conjunto S es la siguiente:

- 1) $S = \{n_5\}$
- 2) Se saca n_5 de S . Su nuevo coste es 25. Al haber cambiado este coste respecto al antiguo, se introducen los padres de n_5 en S .
- 3) $S = \{n_1, n_2\}$
- 4) Se saca n_1 de S , aunque también se podría haber sacado n_2 , ya que ni n_2 es descendiente de n_1 ni viceversa. El nuevo coste de n_1 es 21, ya que ha de redirigirse el enlace que apunta al mejor subgrafo que cuelga de n_1 . Por tanto, introducimos su padre, n_0 , en S .
- 5) $S = \{n_0, n_2\}$
- 6) No podemos sacar n_0 de S , ya que n_2 es descendiente suyo y ya está en S . Por tanto, sacamos n_2 de S . El nuevo coste de n_2 es 28.
- 7) Se saca n_0 de S . Su nuevo coste es 22.

8) $S = \{\}$

> La situación final es la siguiente:



f) Ciclo 5:

> Se expande n_3 , aunque también se podría haber expandido n_4 .

> La evolución del conjunto S es la siguiente:

1) $S = \{n_3\}$

2) Se saca n_3 de S . Su nuevo coste es 5. Al haber cambiado este coste respecto al antiguo, se introducen los padres de n_3 en S .

3) $S = \{n_1\}$

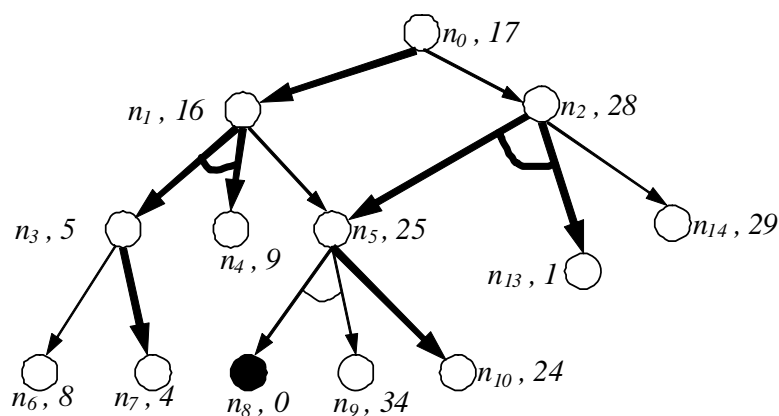
4) Se saca n_1 de S . Su nuevo coste es 16. Al haber cambiado este coste respecto al antiguo, se introducen los padres de n_1 en S .

5) $S = \{n_0\}$

6) Se saca n_0 de S . Su nuevo coste es 17.

7) $S = \{\}$

> La situación final es la siguiente:



g) Ciclo 6:

> Se expande n_4 , aunque también se podría haber expandido n_7 .

> La evolución del conjunto S es la siguiente:

1) $S = \{n_4\}$

2) Se saca n_4 de S . Su nuevo coste es 5. Al haber cambiado este coste respecto al antiguo, se introducen los padres de n_4 en S .

3) $S = \{n_1\}$

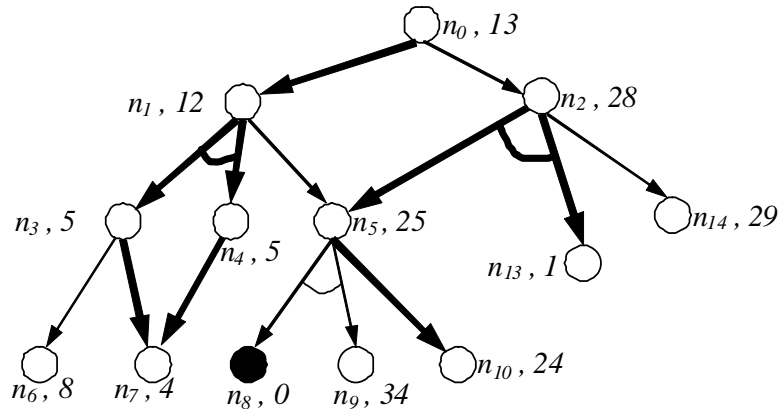
4) Se saca n_1 de S . Su nuevo coste es 12. Al haber cambiado este coste respecto al antiguo, se introducen los padres de n_1 en S .

5) $S = \{n_0\}$

6) Se saca n_0 de S . Su nuevo coste es 13.

7) $S = \{\}$

> La situación final es la siguiente:



h) Ciclo 7:

> Se expande n_7 .

> La evolución del conjunto S es la siguiente:

1) $S = \{n_7\}$

2) Se saca n_7 de S . Su nuevo coste es 1, quedando resuelto. Al haber cambiado este coste respecto al antiguo, se introducen los padres de n_7 en S .

3) $S = \{n_3, n_4\}$

4) Se saca n_3 de S , aunque también se podría haber sacado n_4 ya que ninguno de los dos nodos tiene descendientes en S . Su nuevo coste es 2, quedando resuelto. Al haber cambiado este coste respecto al antiguo, se introducen los padres de n_3 en S .

5) $S = \{n_1, n_4\}$

6) Se saca n_4 de S , ya que n_1 no se puede sacar debido al hecho de que n_4 es descendiente de n_1 y está contenido en S . El nuevo coste de n_4 es 2, quedando resuelto.

7) $S = \{n_1\}$

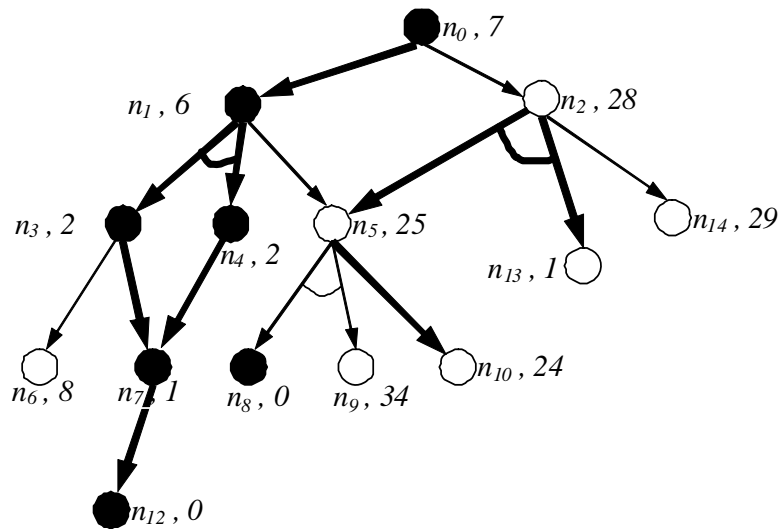
8) Se saca n_1 de S . Su nuevo coste es 6, quedando resuelto. Al haber cambiado este coste respecto al antiguo, se introducen los padres de n_1 en S .

9) $S = \{n_0\}$

10) Se saca n_0 de S . Su nuevo coste es 7, quedando resuelto.

11) $S = \{\}$

> La situación final es la siguiente:



Al final de este ciclo hemos encontrado un subgrafo solución para el nodo n_0 , por lo que el algoritmo finaliza. Obsérvese que el subgrafo solución hallado tiene coste 7 a pesar de estar formado por 6 arcos. Ello es debido a que el arco (n_7, n_{12}) se cuenta dos veces, una para hallar el coste de n_3 y otra para hallar el coste de n_4 , a la hora de calcular el coste del subgrafo solución.

Semana 2. Ejercicio 3. (Valoración: 3.5 puntos)

Describe en qué aspectos se asemejan y en qué aspectos difieren la propuesta de marcos de Minsky y el formalismo de marcos actual. Incluye ejemplos sencillos que ilustren la descripción.

Solución por Severino Fernández Galán:

La contestación a esta pregunta debe basarse en las explicaciones que figuran en las secciones 8.1.1 y 8.1.4 del texto base de teoría. Deben tocarse los siguientes puntos, ofreciendo los ejemplos correspondientes donde sea preciso:

- La propuesta de Minsky tuvo su origen en la década de los 70. En ella se inspiraron las propuestas de marcos que aparecen en las herramientas actuales.
- ¿Qué representa cada marco en cada propuesta?
- ¿Qué representa cada enlace entre dos marcos en cada propuesta?
- ¿Cómo se realiza la inferencia en cada propuesta?
- Mientras que la propuesta de Minsky ha quedado en cierto modo obsoleta, la propuesta actual de marcos ha influido en paradigmas como la programación orientada a objetos.
- La propuesta de Minsky es apropiada para tareas de reconocimiento (por ejemplo, en visión artificial), mientras que la propuesta actual se suele utilizar para la representación de conocimiento en sistemas expertos.