

Problema 1

Suponga que se dispone de un agente que se mueve en una cuadrícula bidimensional compuesta de celdas cuadradas, las cuales pueden o no estar llenas de piedras. El agente se puede mover al norte, sur, este u oeste, una posición. Una de las celdas vacías contiene una caja la cual puede ser movida a una celda adyacente si el agente se para en la celda vecina a la misma, y luego se mueve en la dirección de la caja.

La caja no puede moverse de ninguna otra manera, lo cual implica que si la caja se mueve hacia una esquina, la misma no podrá ser retirada de la misma. Una de las celdas vacías se marca como la celda objetivo a la cual se desea llevar la caja.

P	P	P	P	P	P	P	P	P	P	P
P	X	P	P							P
P		P		P			P	P	P	P
P					C					P
P		P	P	P	P	P	P			P
P						A				P
P	P	P	P	P	P	P	P	P	P	P

Considere que el agente necesita el doble de energía al empujar la caja que cuando realiza otro tipo de acción.

P: hay piedras en esa celda

C: celda donde se encuentra la caja

A: posición inicial del agente

X: celda objetivo

Se pide plantear formalmente este problema

Problema 2

Se tiene tres monedas sobre una mesa en posición cara-sello-cara. Considere una manera de jugar que consiste en dar vuelta un par de monedas cada vez. Se desea que las tres monedas queden en la configuración sello-sello-sello en exactamente tres jugadas. Se pide plantear formalmente el problema

Problema 3

Se tiene un puzzle formado por tres fichas negras, tres fichas blancas y un espacio en blanco, colocados inicialmente como en la siguiente figura:

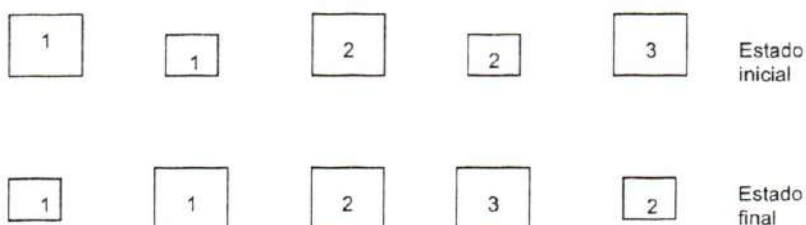
N N N B B B

En este juego una ficha puede moverse a una posición adyacente vacía. En este caso, el costo es 1. Además una ficha puede saltar sobre una o dos fichas hasta alcanzar el espacio vacío. En este caso el costo es igual al número de fichas saltadas. El objetivo consiste en conseguir que todas las fichas blancas estén a la izquierda de las negras. La posición del espacio en blanco no tiene importancia.

Para este problema se pide plantearlo formalmente

Problema 4

Este problema consiste en modificar la posición inicial de un grupo de cajas hasta llegar a otra posición previamente fijada. El grupo de cajas está formado por tres cajas grandes y dos pequeñas, todas enumeradas y situadas formando una fila. Los estados inicial y final son los indicados en las siguientes figuras:



Se pide planearlo formalmente

1) Se representará las posiciones de la región mediante pares (fila, columna) siendo (1,1) la esquina superior izquierda y (7,11) la esquina inferior derecha.

Estados:

Los representaremos estados como una terna:

(posagente, poscaja, pospiedras) donde
posagente = (filaA, colA) representa la posición del agente

poscaja = (filaC, colC) representa la posición de la caja.

pospiedras = a un conjunto de posiciones (fila, columna) de las piedras

Estado Inicial:

((6,7), (4,6), { (1,1), (1,2), (1,3), (1,4), (1,5), (1,6),
(1,7), (1,8), (1,9), (1,10), (1,11), (2,1),
(2,3), (2,4), (2,11), (3,1), (3,3), (3,5),
(3,8), (3,9), (3,10), (3,11), ..., (7,11) })

Estado Final:

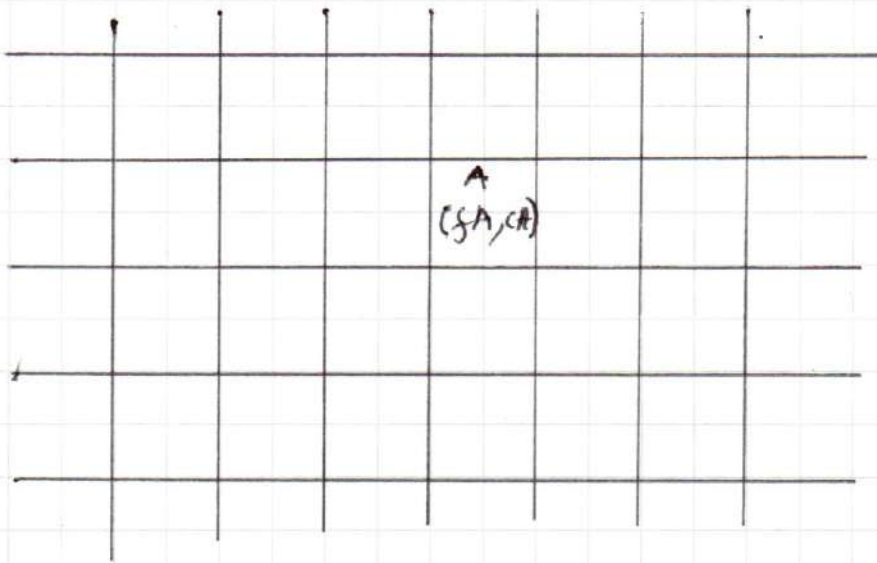
(—, (2,2), { % })
— = cualquier estado

Operadores:

Como los bordes de la cuadrícula están cerrados por piedras, no es necesario verificar los límites de la misma en los operadores.

Debemos distinguir en términos generales los dos operadores cuyos efectos es mover el agente y mover la caja:

- Movimientos del Agente:



01) Mover izquierda:

Si $(filaA, colA - 1) \notin \text{pospiedras}$ \wedge

$(filaA, colA - 1) \neq \text{poscaja} \Rightarrow$

$((filaA, colA - 1), \text{poscaja}, \text{pospiedra})$

02) Mover Derecha

Si $(filaA, colA+1) \notin Pospiedras \wedge$
 $(filaA, colA+1) \neq poscaja \Rightarrow$

$((filaA, colA+1), poscaja, pospiedras)$

03) Mover Arriba

Si $(filaA-1, colA) \notin Pospiedras \wedge$

$(filaA-1, colA) \neq poscaja \Rightarrow$

$((filaA-1, colA), poscaja, pospiedras)$

04) Mover Abajo

Si $(filaA+1, colA) \notin Pospiedras \wedge$

$(filaA+1, colA) \neq poscaja \Rightarrow$

$((filaA+1, colA), poscaja, pospiedras)$

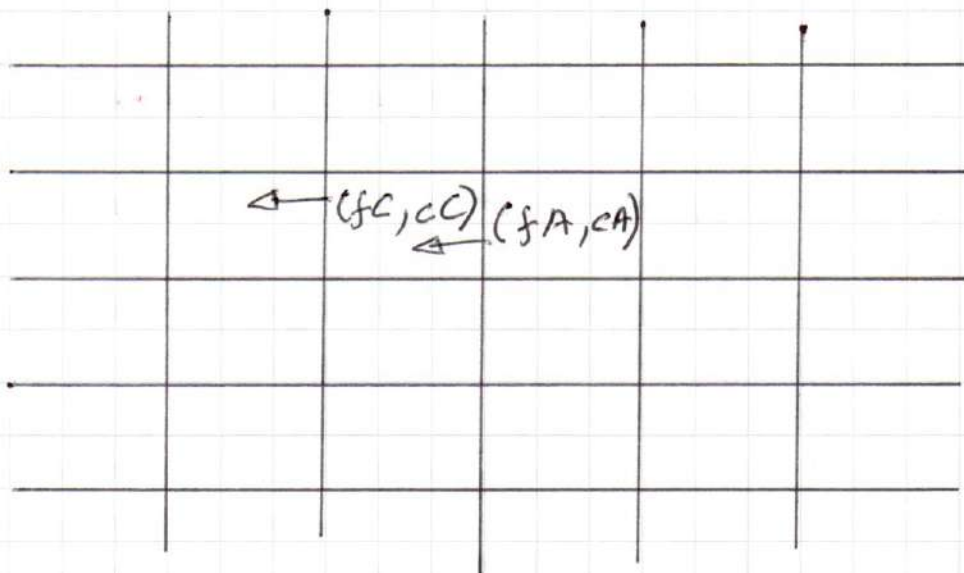
- Movimientos de la caja:

05) Empujar Izquierda:

Si $(fila A, col A - 1) = poscaja \wedge$

$(fila C, col C - 1) \notin pospiedras \Rightarrow$

$((fila A, col A - 1), (fila C, fila C - 1), pospiedras)$



06) Empujar Derecha

Si $(fila A, col A + 1) = poscaja \wedge$

$(fila C, col C + 1) \notin pospiedras \Rightarrow$

$((fila A, col A + 1), (fila C, col C + 1), pospiedras)$

07) Empujar Arriba

Si $(fila A - 1, col A) = poscaja \wedge$

$(fila C - 1, col C) \notin pospiedras \Rightarrow$

$((fila A - 1, col A), (fila C - 1, col C), pospiedras)$

08) Empujar Abajo

Si $(filaA+1, colA) = poscaja \wedge$

$(filaC+1, colC) \neq pospietra \Rightarrow$

$((filaA+1, colA), (filaC+1, colC), pospietra)$

Costo de Ruta:

Considerando la energía adicional que el agente necesita para empujar la caja, se proponen los siguientes costos de aplicación de los operadores:

01 a 04 : costo de 1

05 a 08 : costo de 2

$$c(op) = \begin{cases} 1 & \text{si } op = 01-04 \\ 2 & \text{si } op = 05-08 \end{cases}$$

Prueba de meta:

Comparar el estado actual con el estado final, es decir con

$(-, (2,2), pospietra)$

o bien si

$poscaja = (2,2)$

Problema de las fichas blancas y negras.

Los estados los podemos definir por una 7-upla, una componente por cada ficha, incluyendo el espacio ~~en blanco~~ vacío. Las componentes tomarán los valores: N si la ficha es negra, V ~~si está~~ para representar el espacio ~~en~~ vacío, y B para indicar las fichas blancas.

$$EE = \{ (c_1, c_2, c_3, c_4, c_5, c_6, c_7) / c_i \in \{N, V, B\}, i=1,7 \}$$

$$EI : (N, N, N, V, B, B, B)$$

$$EF : (B, B, B, N, N, N, V)$$

Operadores:

O1: mover-adyacente(i, j) : mueve una ficha i a una posición j

Precondiciones:

- i) Si $|i - j| = 1$, es decir las ~~ficha~~ posiciones i y j son adyacentes
- ii) $c_j = V$, es decir ~~en~~ la posición j está vacía.
- iii) $1 \leq i \leq 7$ y $1 \leq j \leq 7$

Post-condiciones:

- i) ~~$c_j = c_i$~~ , es decir, ~~en la posición j ponemos~~ el espacio vacío que estaba, es decir, en la posición j ~~coloca~~ ~~se queda~~ queda la ficha que estaba en la posición i
- ii) $c_i = V$, es decir la posición i queda vacía.

OP2:

(4)

$\text{mover-salto1}(i, j)$: mueve una ficha desde la posición i a la posición j (V) saltando otra ficha intermedia

Precondiciones:

- i) $|i - j| = 2$, es decir, la ficha de la posición i está separada por una ficha del espacio en ~~blanco~~ vacío j
- ii) $c_j = V$, es decir la posición j está en vacía.
- iii) $1 \leq i \leq 7, 1 \leq j \leq 7$

Postcondiciones:

- i) $c_j = c_i$, es decir, en la posición j queda la ficha de la posición i
- ii) $c_i = V$ en la posición i queda el espacio vacío.

OP3:

$\text{mover-salto2}(i, j)$: mueve una ficha desde la posición i a la posición j (V) saltando dos fichas intermedias.

PRACTICA 3 INTELIGENCIA ARTIFICIAL

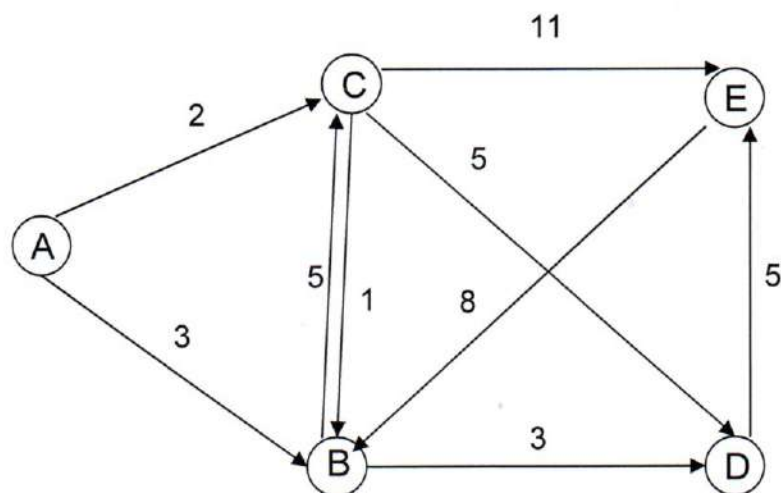
1.- Supongamos que tenemos d heurísticas distintas, $\{h_i\}_{i=1}^d$ todas admisibles. Para las siguientes heurísticas se pide determinar, fundadamente, si las siguientes heurísticas son admisibles

a) $\max \{h_i(n) / i=1, 2, \dots, d\}$

b) $\frac{1}{d} \sum_{i=1}^d h_i(n)$

c) $\sum_{i=1}^d h_i(n)$

2.- Consideremos el problema de búsqueda con estados A, B, C, D y E. En el siguiente grafo se indica el costo de pasar de un estado a otro, siendo A el estado inicial y E el estado final.



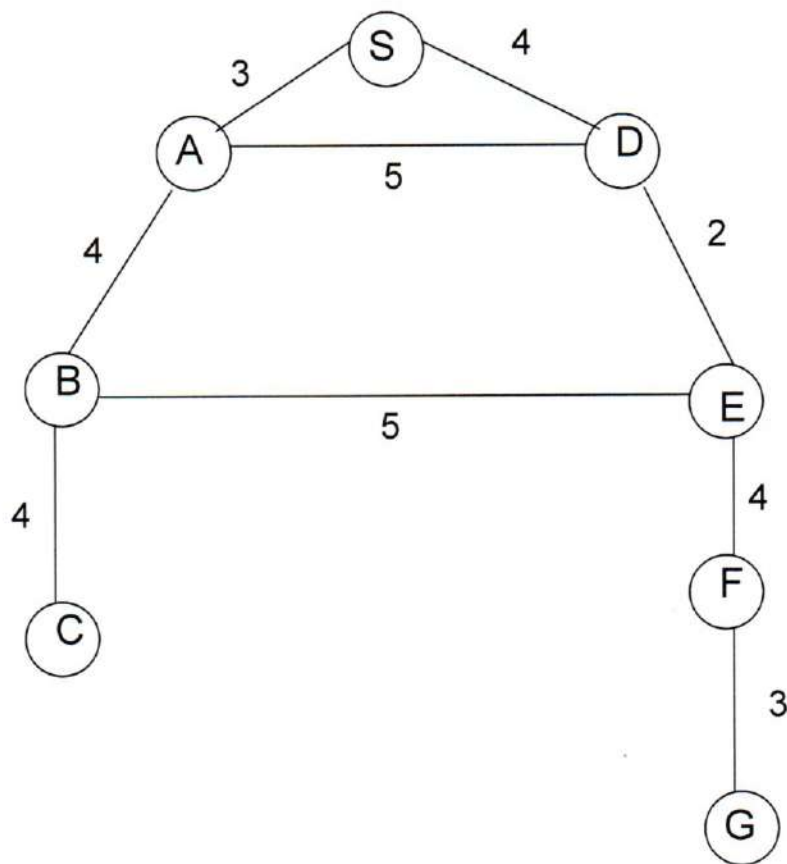
Considere las heurísticas h_1 y h_2 que se indican

nodo	$h_1(\text{nodo})$	$h_2(\text{nodo})$
A	8	7
B	6	8
C	6	5
D	4	7
E	0	0

Se pide determinar, fundadamente, si el algoritmo A* es óptimo con las heurísticas indicadas anteriormente.

3.- Para el siguiente grafo, donde el estado inicial es S y los estados metas son C y G, se pide determinar si A* garantiza encontrar una solución óptima. En caso de no ser posible introduzca algunas modificaciones de manera que A* garantice encontrar la solución óptima. Considere la siguiente función heurística

n	S	D	A	E	B	F	G	C
$h(n)$	13	8	10	6	5	3	0	0



A - Supongamos que $h^*(n)$ es el verdadero costo desde el estado n a la meta, para cada n . Entonces,

$$h_i(n) \leq h^*(n) \quad \forall n \in E$$

$$a) \max_i h_i(n) / i=1, 2, \dots, n \leq h^*(n) \quad \forall n$$

$$h'(n) = \max_i h_i(n) / i=1, 2, \dots, n \leq h^*(n) \quad \forall n$$

sigue

$$h'(n) \leq h^*(n) \quad \forall n$$

Por lo tanto $h'(n)$ definida como el máximo de las $h_i(n)$ es admisible (5)

$$b) \sum_{i=1}^d h_i(n) \leq \sum_{i=1}^d h^*(n)$$

$$\sum_{i=1}^d h_i(n) \leq d h^*(n)$$

$$\frac{1}{d} \sum_{i=1}^d h_i(n) \leq \frac{1}{d} d h^*(n) = h^*(n)$$

$$\Rightarrow \frac{1}{d} \sum_{i=1}^d h_i(n) \text{ es admisible} \quad (5)$$

$$c) \quad \sum_{i=1}^d h_i(n) \leq \sum_{i=1}^d h_i^*(n) = d h^*(n) \Rightarrow$$

$$\sum_{i=1}^d h_i(n) \text{ no es admisible.} \quad (5)$$

1) A^* es óptimo si h es admisible

h es admisible si

$$h(n) \leq h^*(n) \quad \forall n$$

donde h^* es la verdadera distancia desde el estado n al nodo meta mas cercano.

Debemos encontrar h^* y compararlo con h_1 y h_2

Costo del nodo $n \in E$ (todos los casos)	$h^*(n)$	$h_1(n)$	$h_2(n)$
A(13, 12, 11, 19, 11)	11	8	7
B(12, 8, 15)	8	6	8
C(11, 10, 9)	9	6	5
D(5)	5	4	7
E	0	0	0

A-C-E : 13

B-C-E : 12

A-C-D-E : 12

B-D-E : 8

A-C-B-D-E : 11

B-C-D-E : 15

A-B-C-E : 19

A-B-D-E : 11

C-E : 11

C-D-E : 10

C-B-D-E : 9

Como $h_1(n) \leq h^*(n) \quad \forall n$, A^* es óptimo con h_1 .

h_2 no es admisible, por lo tanto

3))

S-A-D-E-F-G	$3+5+2+4+3 = 17$
S-A-B-E-F-G	$3+4+5+4+3 = 19$
S-D-E-F-G	$4+2+4+3 = 13$
S-D-A-B-E-F-G	$4+5+4+4+3 = 20$
S-A-B-C	$3+4+4 = 11 *$
S-A-D-E-B-C	$3+5+2+5+4 = 19$
S-D-A-B-C	$4+5+4+4 = 17$
S-D-E-B-C	$4+2+5+4 = 15$

A-D-E-F-G	$5+2+4+3 = 14$
A-B-E-F-G	$4+5+4+3 = 16$
A-D-E-B-C	$5+2+5+4 = 16$
A-B-C	$4+4 = 8 *$

D-E-F-G	$2+4+3 = 9 *$
D-A-B-E-F-G	$5+4+5+4+3 = 21$
D-E-B-C	$2+5+4 = 11$
D-A-B-C	$5+4+4 = 13$

B-E-F-G	$5+4+3 = 12$
B-A-D-E-F-G	$4+5+2+4+3 = 18$
B-C	$4 = 4 *$

E-B-C	$5+4 = 9$
E-F-G	$4+3 = 7 *$

F-G	$3 = 3 *$
F-E-B-C	$4+5+4 = 13$

n	S	D	A	E	B	F	G	C
$h^*(n)$	11	9	8	7	4	3	0	0
$l(n)$	13	8	10	6	5	3	0	0
	10		7		4			

15

5

PRACTICA 3
IA

1.- Considere el siguiente conocimiento:

Asterix es un galo
Los romanos que son amigos de algún galo odian a César
Asterix ayudó a Marco
Marco es amigo de quién le ayuda
Quien odia a algún romano lucha contra él
Marco es un romano

Se pide representar el conocimiento anterior en términos de la lógica de predicados de primer orden

2.- Represente el siguiente conocimiento en términos de la lógica de predicado de primer orden

- a) Hay personas que ni son libres ni sienten ningún deseo de serlo
- b) El alma enamorada es alma blanda, mansa, humilde y paciente
- c) Las sustancias radioactivas tienen corta vida. Ningún isótopo de uranio que sea radiactivo tiene corta vida. Todos los isótopos de uranio son radiactivos, entonces todos los isótopos de uranio tienen valor medicinal

3.- Represente el siguiente conocimiento mediante la lógica de predicados de primer orden.

Se sabe que hay algunos hombres lobos que son vampiros. Los vampiros atacan a todo hombre lobo desprevenido. De manera general, ningún hombre lobo atacaría a uno de su especie. No todas las criaturas son hombres lobos desprevenidos.

4.- Considere el siguiente conocimiento

Antonio, Miguel y Juan son estudiantes de informática.
A todos los estudiantes de informática o les gusta el fútbol o el ajedrez o ambas cosas.
A nadie que le guste el fútbol le gusta la lluvia.
A todos a quienes le gusta el ajedrez les gusta la nieve.
A Miguel le desagrada todo lo que le gusta a Antonio y le gusta todo lo que desagrada a Antonio.
A Antonio le gusta la lluvia y la nieve.

Se pide construir la base de conocimientos en términos de la lógica de predicados

Práctica 3

//

Problema 1:

1) galo(asterix)

2) $\forall x, \exists y \text{ romano}(x) \wedge \text{amigo}(x, y) \wedge \text{galo}(y) \rightarrow \text{odia}(x, \text{cesar})$

3) ayuda(asterix, marco)

4) $\forall x \text{ ayuda}(x, \text{marco}) \rightarrow \text{amigo}(x, \text{marco})$

5) $\forall x, \exists y \text{ odia}(x, y) \wedge \text{romano}(y) \rightarrow \text{lucha}(x, y)$

6) romano(marco)

Semántica:

$\text{amigo}(x, y)$: x es amigo de y

$\text{ayuda}(x, y)$: x ayuda a y

$\text{galo}(x)$: x es galo

$\text{odia}(x, y)$: x odia a y

$\text{romano}(x)$: x es romano

$\text{lucha}(x, y)$: x lucha con y

Dominio 4 personas 4

Problema 2:

2/

- a) $H(x)$: x es hombre
 $L(x)$: x es libre
 $D(x)$: x desea ser libre

$$\exists x (H(x) \wedge \neg L(x) \wedge \neg D(x))$$

- b) $E(x)$: x es alma enamorada
 $B(x)$: x es alma blanda
 $M(x)$: x es alma mansa
 $H(x)$: x es alma humilde
 $P(x)$: x es alma paciente

$$\forall x (E(x) \rightarrow B(x) \wedge M(x) \wedge H(x) \wedge P(x))$$

- c) $S(x)$: x es sustancia radioactiva
 $M(x)$: x tiene valor medicinal
 $C(x)$: x tiene corta vida
 $U(x)$: x es isótopo de uranio

$$\forall x S(x) \rightarrow C(x)$$

$$\forall x U(x) \wedge S(x) \rightarrow \neg C(x)$$

$$\forall x U(x) \wedge S(x) \rightarrow U(x) \wedge M(x)$$

Problema 3:

3/

Dominio {criaturas de la noche}

$HL(x)$: x es hombre lobo

$V(x)$: x es vampiro

$Ataca(x, y)$: x ataca a y

$Desp(x)$: desprevenido x

$\exists x \quad HL(x) \wedge V(x)$

$\forall x \forall y :$

$V(x) \wedge HL(y) \wedge Desp(y) \rightarrow ataca(x, y)$

$\forall x \forall y \quad HL(x) \wedge HL(y) \rightarrow \neg ataca(x, y)$

$\forall x \quad \neg (HL(x) \wedge Desp(x))$

Problema 4:

$est(x)$: x es estudiante de informática

$gusta(x, y)$: x le gusta y

1) $est(antonio) \wedge est(miguel) \wedge est(juan)$

2) $\forall x \quad est(x) \wedge [gusta(x, futbol) \vee gusta(x, ajedrez) \vee gusta(x, lluvia) \vee gusta(x, nieve)]$

3) $\forall x \quad gusta(x, futbol) \rightarrow \neg gusta(x, lluvia)$

4) $\forall x \quad gusta(x, ajedrez) \rightarrow gusta(x, nieve)$

5) $\forall x \quad gusta(antonio, x) \rightarrow \neg gusta(miguel, x)$

6) $\forall x \quad \neg gusta(antonio, lluvia) \rightarrow gusta(antonio, nieve)$

7) $gusta(antonio, lluvia) \wedge gusta(antonio, nieve)$