

### Evaluación (junio 2016)

**Duración:** 3 horas

**Ejercicio 1** [evaluación individual del laboratorio]

- a) Indique cómo modeló el tablero en su implementación.
- b) Indique qué función heurística utilizó para evaluar los tableros en el algoritmo minimax.

**Ejercicio 2** [15 puntos]

a) Dé una interpretación que sea modelo y una que no lo sea para los siguientes programas lógicos:

i.  $p(f(X), Y) \leftarrow p(X, Y).$   
 $p(X, X).$   
 $q \leftarrow s.$

ii.  $p(X, s(X, Xs)).$   
 $p(X, s(Y, Ys)) \leftarrow p(X, Y).$

b) Para el programa i, considere el objetivo  $\leftarrow p(X, Y)$  y la sustitución  $\{X|f(4), Y|4\}$ . ¿Es una respuesta correcta? ¿Es una respuesta computada?

**Ejercicio 3** [23 puntos]

Sea el siguiente programa lógico P:

$p(X, Y, Z) :-$   
 $q(X, Y, Z).$   
 $p(X, Y, Z) :-$   
 $X1 \text{ is } X + 1,$   
 $p(X1, Y, Z).$

$q(X, X, [X]) :-$   
 $X > 0.$

$q(X, Y, [X|Z]) :-$   
 $X1 \text{ is } X - 1,$   
 $q(X1, Y, Z).$

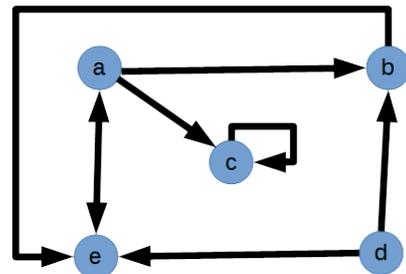
- a) Construya el árbol SLD correspondiente al objetivo  $\leftarrow p(2, A, B)$  asumiendo que el intérprete selecciona el átomo de más a la izquierda como regla de computación.
- b) Indique qué respuestas dará el intérprete de Prolog para el objetivo anterior.
- c) Indique qué respuestas dará el intérprete de Prolog si se agrega un cut en la primer regla del predicado q/3 de la siguiente manera:

$q(X, X, [X]) :-$   
 $X > 0, !.$

**Ejercicio 4** [35 puntos]

Considere un grafo dirigido representado mediante el predicado arista. Por ejemplo, si se tiene el siguiente programa:

$arista(a, b).$   
 $arista(a, c).$   
 $arista(a, e).$   
 $arista(b, e).$   
 $arista(c, c).$   
 $arista(d, b).$   
 $arista(d, e).$   
 $arista(e, a).$



se estaría representando el grafo de la figura de la derecha.

Implementar los siguientes predicados, implementando los predicados auxiliares que se usen (excepto los de segundo orden):

**a)** `matriz_adyacencia(+LV, ?M) <-` M es la matriz de adyacencia del grafo representado en el programa mediante el predicado `arista`, para la lista de vértices LV. Cada celda (I, J) tiene un 1 si hay una arista entre el vértice en la posición I de la lista LV y el de la posición J de LV. En caso contrario tiene un 0. La matriz se representa como una lista de listas.

**b)** `traza(+M, ?T) <-` T es la cantidad de lazos del grafo (vértices que están relacionados con ellos mismos), que en este caso coincide con el valor de la traza de la matriz de adyacencia M (la traza de una matriz es la suma de los elementos de la diagonal). Este predicado no utiliza al predicado `arista` como auxiliar.

**c)** `camino_simple(+V, +U, ?C) <-` C es un camino simple (que no repite vértices) del vértice V al vértice U del grafo representado en el programa mediante el predicado `arista`. C es representado como una lista de vértices, que comienza con V y termina con U. Nota: siempre existe el camino trivial [V] desde un vértice a sí mismo, independientemente de que el vértice tenga un lazo o no.

**d)** `clausura_transitiva(+V, -LV) <-` LV es la lista de vértices a los cuales se llega haciendo recorridas sobre el grafo representado en el programa mediante el predicado `arista`, desde el vértice V. La lista incluye a V y puede tener elementos repetidos.

### Ejercicio 5 [12 puntos]

En Programación Lógica Inductiva (algoritmo FOIL) se genera una o más cláusulas para un predicado objetivo a partir de ejemplos positivos y negativos y cláusulas opcionales de background.

**a)** Explique brevemente qué significa que una cláusula es **consistente** respecto al conjunto de ejemplos y al conocimiento background.

**b)** Explique brevemente qué significa que una cláusula es **completa** respecto al conjunto de ejemplos y al conocimiento background.

**c)** Considere el siguiente caso de ILP, con el predicado `camino(X,Y)` como predicado objetivo a aprender e hipótesis de mundo cerrado para la negación.

#### Background

```
arista(a,b), arista(c,d), arista(a,c), arista(b,e)
```

#### Ejemplos

```
camino(a,b)
camino(c,d)
camino(a,c)
camino(b,e)
camino(a,e)
camino(a,d)
```

Para cada una de las siguientes cláusulas indique si es consistente y/o completa. Justifique brevemente:

i. `camino(X,Y) :- arista(X,Y).`

ii. `camino(X,Y).`

iii. `camino(X,Y) :- arista(X,Z), arista(Z,Y).`