



## LÓGICA PARA CIENCIAS DE LA COMPUTACIÓN

Trabajo Práctico “F”  
**Semántica de Punto Fijo**  
Primer Cuatrimestre de 2008

**Observación importante:** Las consultas relacionadas con los temas desarrollados en los trabajos prácticos serán únicamente respondidas durante las propias clases prácticas a lo largo del cursado.

### Ejercicios

1. Para cada uno de los siguientes programas lógicos definidos,

- |           |  |   |
|-----------|--|---|
| ▪ $P_1 =$ | $p :- q, r.$   | $q.$  |
| ▪ $P_2 =$ | $p :- q, r.$<br>$r.$   | $q :- s.$<br>$s.$   |
| ▪ $P_3 =$ | $p :- q, r.$<br>$a :- p.$<br>$c.$<br>$r.$  | $p :- a, b, c.$<br>$b :- a, d.$<br>$d :- r.$<br>$a :- r, c.$                      |
| ▪ $P_4 =$ | $d :- a, b.$<br>$d :- c, a.$<br>$q.$   | $d :- b, c.$<br>$b :- p.$<br>$a.$   |
| ▪ $P_5 =$ | $p :- a, b, c.$<br>$r :- a, c.$<br>$a :- q, r.$<br>$b :- s, t.$<br>$t :- b, d.$<br>$d :- c.$ | $p :- q, c.$<br>$r :- p.$<br>$a :- d, s.$<br>$s :- c, d.$<br>$t :- a, d.$<br>$c.$ |

calcular los siguientes conjuntos:

- $T_P(\emptyset)$
- $T_P(\{q, r, a\})$
- $T_P(T_P(\{c, p\}))$
- $T_P \uparrow 3$
- $M_P$  en base a  $T_P$

NOTA: Para cada caso, emplear el operador  $T_P$  que corresponda (*i.e.*, para un cierto  $P_i$ , se debe usar  $T_{P_i}$ ).

2. Encontrar dos puntos fijos distintos para el operador  $T_P$  correspondiente al siguiente programa lógico definido:

$p :- q.$

$q :- p.$

3. Sea  $P$  el siguiente programa lógico definido:

$p(X) :- q(s(X)), r(X).$

$q(0). \quad q(s(0)).$

$r(0).$

En este contexto, determinar el mínimo modelo de Herbrand para  $P$  mediante la obtención del menor punto fijo del operador  $T_P$ , es decir, apelando al teorema que establece la equivalencia entre la semántica declarativa y la de punto fijo.

4. Sea  $P_6$  el siguiente programa lógico definido:

$cte(a). \quad cte(b). \quad cte(c).$

$par(0). \quad par(s(s(X))) :- par(X).$

Obtener el conjunto de todas las posibles consultas que tiene sentido formular en el contexto del programa  $P_6$  (*i.e.*,  $\mathbb{B}_{P_6}$ ). Asumiendo que el predicado  $cte/1$  tiene por objeto determinar si su argumento es una de las constantes contempladas, y que el predicado  $par/1$  tiene por objeto determinar si su argumento es un número natural par en notación  $s^n(0)$ , identificar qué subconjunto de  $\mathbb{B}_{P_6}$  tiene sentido distinguir como el significado pretendido para  $P'$ .

5. Sea  $P_7$  el siguiente programa lógico definido:

$cte(a). \quad cte(b). \quad cte(c).$

$lista([]). \quad lista([X|Xs]) :- cte(X), lista(Xs).$

Obtener el conjunto de todas las posibles consultas que tiene sentido formular en el contexto del programa  $P_7$  (*i.e.*,  $\mathbb{B}_{P_7}$ ). Asumiendo que el predicado  $lista/1$  tiene por objeto verificar si su argumento es una lista de constantes, y que el predicado  $cte/1$  determinar si su argumento es una de las constantes contempladas, identificar qué subconjunto de  $\mathbb{B}_{P_7}$  tiene sentido distinguir como el significado pretendido para  $P_7$ .

6. Supongamos que se desean implementar los predicados  $lista/1$ , para verificar si su argumento es una lista, y  $suma/3$ , para obtener la suma de dos números en notación  $s^n(0)$ . Con esta intención, el programador escribe el siguiente programa lógico definido:

$lista([]).$

$lista([X|Xs]) :- lista(Xs).$

$suma(0, X, X).$

$suma(s(X), Y, s(Z)) :- suma(X, Y, Z).$

Indicar cuál es el subconjunto de su base de Herbrand que captura el significado pretendido para estos predicados (la intención del programador). Teniéndolo en cuenta, ¿se puede decir que la implementación propuesta es sensata? ¿Es completa? ¿Es correcta? Justificar de forma apropiada.

7. Suponiendo que se desea implementar los predicados `binario/1`, para verificar si su argumento es un árbol binario construido a partir del functor `tree` y de la constante `'#'` para denotar al árbol vacío, con sus nodos etiquetados con constantes, y `cte/1`, que determina si su argumento es una de las constantes contempladas, empleando el siguiente programa lógico definido,

```
cte(a).      cte(b).      cte(c).
```

```
binario(#).
```

```
binario(tree(Nodo, HI, HD)) :- binario(HI), binario(HD).
```

determinar si la implementación propuesta es correcta, comparando su significado pretendido con el significado efectivo del programa anterior.