Departamento de Cs. e Ingeniería de la Computación
Universidad Nacional del Sur

Lógica para Ciencias de la Computación

Trabajo Práctico "F"

Semántica de Punto Fijo

Primer Cuatrimestre de 2009

> **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.          q :- s.
     r.                  s.
     ```
   - $P_3 =$
     ```
     p :- q, r.          p :- a, b, c.
     a :- p.             b :- a, d.
     c.                  d :- r.
     r.                  a :- r, c.
     ```
   - $P_4 =$
     ```
     d :- a, b.          d :- b, c.
     d :- c, a.          b :- p.
     q.                  a.
     ```
   - $P_5 =$
     ```
     p :- a, b, c.       p :- q, c.
     r :- a, c.          r :- p.
     a :- q, r.          a :- d, s.
     b :- s, t.          s :- c, d.
     t :- b, d.          t :- a, d.
     d :- c.             c.
     ```

   calcular los siguientes conjuntos:

   a) $T_P(\emptyset)$
   b) $T_P(\{q, r, a\})$
   c) $T_P(T_P(\{c, p\}))$
   d) $T_P \uparrow 3$
   e) 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).      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).      cte(b).      cte(c).
   par(0).   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 nmero natural par en notacin $s^n(0)$, 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).      cte(b).      cte(c).
   lista([ ]). 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 intencin, el programador escribe el siguiente programa lgico 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 intencin del programador). Tenindolo 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 arbol 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.

8. Supongamos que se desea implementar un predicado cte/1, que determine si su argumento es una de las constantes del conjunto $\{a, b, c\}$, y un predicado intercalar/3, tal que dadas como argumento dos listas de constantes (a, b y c) de igual longitud, retorne la lista que resulta de intercalarlas. Formalmente, dadas las listas [X_1, ..., X_n] e [Y_1, ..., Y_n], intercalar/3 retornar la lista [X_1, Y_1, ..., X_n, Y_n], donde X_i e $Y_i \in \{a, b, c\}$, $1 \leq i \leq n$.

   a) En base a estos requerimientos, un programador escribe el siguiente programa lgico definido:

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

   intercalar([ ], [ ], [ ]).
   intercalar([X | Xs], [Y|Ys], [X, Y|Zs]):- cte(X), cte(X), intercalar(Xs, Ys, Zs).
   ```

   Determinar formalmente si la implementacin propuesta es correcta, comparando su significado pretendido (requerimientos) con el significado efectivo del programa anterior.

   b) En base a los mismos requerimientos, un programador escribe el siguiente programa lgico definido:

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

   intercalar([ ], [ ], [ ]).
   intercalar([X | Xs], [Y|Ys], [X, X|Zs]):- cte(X), cte(Y), intercalar(Xs, Ys, Zs).
   ```

   Determinar formalmente si la implementacin propuesta es correcta, comparando su significado pretendido (requerimientos) con el significado efectivo del programa anterior.