

Práctico 3: Árbol de derivación SLD, Corte y Negación

Profesora: Dra. Yisel Garí

Recordar:

- Árbol de derivación SLD
- Funcionamiento del operador de corte (!)
- Operador de Negación (not)
- Operador de chequeo de unificación (=)

1. A partir del programa familia:

```
(P1) padre(luis,alicia).  
(P2) padre(luis,josé).  
(P3) padre(jose,ana).  
(M1) madre(alicia,dario).  
(A1) abuelo(X,Y) :- padre(X,Z),madre(Z,Y).  
(A2) abuelo(X,Y) :- padre(X,Z),padre(Z,Y).
```

Haga una deducción de los siguientes objetivos a través de la derivación-SLD, de tal modo que obtenga una refutación-SLD y en otro caso una derivación-SLD fracaso. En el caso de triunfos (refutaciones-SLD) determine los valores de salida de las variables en los objetivos (la Sustitución Respuesta Computada -SRC-).

- a) :- abuelo(X,Y).
- b) :- padre(X,Z), abuelo(Y,Z).

2. Considere los programas concatena/3 y pertenece/2 constituidos por las siguientes cláusulas:

```
(C1) concatena([],X,X).  
(C2) concatena([X|Y], Z, [X|W]) :- concatena(Y,Z,W).  
(P1) pertenece(X, [X|_]).  
(P2) pertenece(X, [_|Y]) :- pertenece(X, Y).
```

Haga una derivación-SLD de los siguientes objetivos (tipo refutación o de tipo fracaso). En el caso de triunfos, determinando los valores de salida de las variables en los objetivos (SRC).

- a) :- concatena ([1,2], [3],X).
- b) :- concatena ([X|[2]], [4], [1|W]), pertenece(2, W).
- c) :- concatena ([1], Z, [2|Y]).

3. Dada las relaciones `f(X,Y)` y `f_1(X,Y)`

```
f(X,0) :- X < 3.  
f(X,2) :- 3 =< X, X < 6.  
f(X,4) :- 6 =< X.
```

```
f_1(X,0) :- X < 3, !.
f_1(X,2) :- 3 <= X, X < 6, !.
f_1(X,4) :- 6 <= X.
```

Construir los árboles de derivación SLD correspondientes a los siguientes objetivos:

- a) :- f(7,Y), 2 < Y.
- b) :- f_1(1,Y), 2 < Y.
- c) :- f_1(7,Y).

4. La relación pertenece está definida por:

```
pertenece(X,[X|_]) :- !.
pertenece(X,[_|L]) :- pertenece(X,L).
```

Analice qué ocurre en los árboles SLD de las siguientes consultas:

- a) :- pertenece(X,[a,b,c]), X=a.
- b) :- pertenece(X,[a,b,c]), X=b.
- c) :- X=b, pertenece(X,[a,b,c]).

5. Definir la relación **diferencia**(+C1,+C2,-C3) que se verifique si C3 es la diferencia de los conjuntos C1 y C2. Definir una versión con negación (*diferencia*₁) y otra con corte (*diferencia*₂).
6. Definir la relación **sumaPares**(+L,-N) que se verifica si N es la suma de todos los números pares de la lista de números L. Definir una versión con negación (*sumaPares*₁) y otra con corte (*sumaPares*₂).