

Programación con restricciones - Práctica 13.

Hasta el 26 de mayo: 10 puntos. Hasta el 27 de mayo: 8 puntos.

En este mini-tutorial, práctica, vamos a echar un vistazo a la librería de restricciones de dominios finitos incorporada a casi todos los sistemas del lenguaje Prolog. En particular, nos vamos a fijar en SWI prolog. La especificación de esta librería se puede encontrar en:

<http://www.swi-prolog.org/man/clpfd.html>

Preparación:

Para entrar en SWI Prolog, desde un terminal Linux:

```
swipl
```

Nos aparecerá el prompt ?-

Nos podemos asegurar de que estamos en hlocal tecleando:

```
?- pwd.
```

Nota: En Prolog todas las instrucciones deben acabar en '.'

La práctica

1) [2 puntos] Escribimos en el shell de Prolog:

```
?- A*A#= 625.
```

Contestar la pregunta 1.

2) [4 puntos] Ahora vamos escribir en un editor (por ejemplo gedit) el programa pract.pl. La primera línea debe ser

```
:- use_module(library(clpfd)).
```

que indica que vamos a usar la librería de dominios finitos

Probar el siguiente ejemplo (en el editor, a continuación del `use_module(library(clpfd)).`)

```
g(X,Y) :- f(X), X#>Y.  
f(X) :- X in 1..3.  
f(X) :- X in 5..7 .  
f(X) :- X in 8..12 .
```

Tras grabar, vamos al terminal y dentro de SWI cargamos el programa:

```
consult('pr.pl').
```

Ahora podemos lanzar objetivos como

```
g(X,Y).
```

nos mostrará una primera respuesta

```
X in 1..3,
```

```
Y#=<X+ -1,
```

Y in inf..2

obtenida con la primera regla de f. Si tecleamos ';' el sistema reevaluará, intentado ahora la segunda regla de f. Es lo que se conoce como backtracking.

Contestar a las preguntas 2,3

3) [2 puntos] Se puede definir el dominio de una lista de variables con ins. Probar el objetivo [X,Y,Z] ins 1..3, X#Y.

Sin embargo, la respuesta es en forma de restricciones, y a veces nos interesan las soluciones individuales. Para ello podemos usar el etiquetado:

L = [X,Y,Z], L ins 1..3, X#Y, label(L).

Ahora añadimos al programa .pl el predicado:

```
trio([X,Y,Z],Pto) :- [X,Y,Z] ins 0..Pto,  
                    Y#=Z-1,  
                    Z#>=Pto-1,  
                    all_different([X,Y,Z]), label([X,Y,Z]).
```

desde el shell reconsultamos (consult(pract).) y pasamos a contestar a las preguntas 4

4) [2 puntos] Copiar en la pregunta 5 un predicado para obtener cuadrados de tamaño 3x3 (las filas y columnas deben sumar 15, y los valores estar entre 1 y 9, siendo todos diferentes). El predicado comienza de la forma:

```
magico([A,B,C,D,E,F,G,H,I] ) :- ..... .
```

Un ejemplo de salida esperada:

```
?- magico([A,B,C,D,E,F,G,H,I]).
```

A = 1,

B = 5,

C = 9,

D = 6,

E = 7,

F = 2,

G = 8,

H = 3,

I = 4

Entrega: subir el fichero preguntas.txt con las respuestas. No olvidar los nombres de los integrantes del grupo.