

Hay que resolver en prolog los problemas listados a continuación.

## Problema(s) A:

Primer problema, el de “Hacer Aguas”: disponemos de un grifo de agua, un cubo de 5 litros y otro de 8 litros. Se puede verter el contenido de un cubo en otro, llenar un cubo, o vaciar un cubo del todo, y queremos saber la secuencia mínima de operaciones para obtener exactamente 4 litros de agua en el cubo de 8 litros.

El segundo problema: dado un natural  $n > 0$ , una posición inicial  $(Fila_I, Columna_I)$ , una posición final  $(Fila_F, Columna_F)$ , y un número de pasos  $P$ , encontrar un camino de  $(Fila_I, Columna_I)$  a  $(Fila_F, Columna_F)$ , en un tablero de ajedrez de  $n \times n$  en exactamente  $P$  pasos de caballo. El programa ha de fallar si para la  $n$  en cuestión no existe tal camino.

Usa (es obligatorio) el siguiente esquema prolog para resolver los dos problemas:

```
camino( E,E, C,C ).
camino( EstadoActual, EstadoFinal, CaminoHastaAhora, CaminoTotal ):-
    unPaso( EstadoActual, EstSiguiente ),
    \+member(EstSiguiente,CaminoHastaAhora),
    camino( EstSiguiente, EstadoFinal, [EstSiguiente|CaminoHastaAhora], CaminoTotal ).

solucionOptima:-
    nat(N),                               % Buscamos solución de "coste" 0; si no, de 1, etc.
    camino([0,0],[0,4],[[0,0]],C), % En "hacer aguas": -un estado es [cubo5,cubo8], y
    length(C,N),                          % -el coste es la longitud de C.
    write(C).
```

**Problema B:** Se trata de resolver el puzzle de <http://www.learn4good.com/games/puzzle/brainteasers.htm>. Formato: para resolver el level 3 de esta web, la llamada sería `solve([[4,1],[4,2],[5,2],[5,3]])` y una respuesta correcta sería escribir:

```
[5,3] jumps over [4,2]
[3,1] jumps over [4,1]
[5,1] jumps over [5,2]
```

**Problema C:** After consulting a dietician, we have a list of  $N$  nutrients that one must eat at least once a week. We also know, for each one of the  $P$  products of a grocery store, which of those nutrients it contains. Write a predicate `shopping(K,L)` that, for a given  $K$ , lists in  $L$  a list of at most  $K$  products so that all  $N$  nutrients are included in at least one of the products. Assume the input data are given as:

```
numNutrients(8).
product(milk,[2,4,6]).
product(meat,[1,8]).
...
```

**Problema D:** Suppose we describe in Prolog a roadmap between cities as in the following example:

```
cities([1,2,3,4]).
road(1,2, 10). % road between cities 1 and 2 of 10km
road(1,4, 20).
road(2,3, 25).
road(3,4, 12).
```

We want to find a set of main roads that connect all cities spending at most a certain number of kilometers. Program in Prolog a new predicate `mainroads(K,M)` which means that, for the given  $K$ ,  $M$  is a subset of the roads that connects all cities forming a tree and total length of the roads in  $M$  is at most  $K$  (note that being a tree means that there are no cycles).