



Ayudantía Negación

Lunes 12 de abril

Introducción

”Una técnica que no hemos visto ni en clases ni ayudantías es la de usar una doble negación para simular una cuantificación universal. Para explicar a qué me refiero con esto, veamos el siguiente programa:”

```
util(bolita).  
util(lapiz).  
util(tijera).  
caja(c1).  
contiene(c1,bolita).  
contiene(c2,lapiz).
```

”Supón que quisieras escribir el predicado `caja_completa`, que es tal que `caja_completa(c)` aparece en el modelo cuando `c` es una caja que contiene todos los útiles. En clingo no existe sintaxis para expresar cuantificación universal. Sin embargo, usando la equivalencia lógica *”una caja contiene todos los útiles si no existe un útil que esta no contenga”*, es posible escribir el predicado. Conviene, entonces, escribir un predicado para la parte *”existe un útil que esta no contenga”* y luego definir el predicado pedido como su negación.”

```
no_contiene_algun_util(C) :- util(U), not contiene(C, U).  
caja_completa(C) :- caja(C), not no_contiene_algun_util(C).
```

Preguntas

1. Define el predicado **dag** tal que **dag** aparece en el modelo si y solo si el grafo es acíclico.
2. Define el predicado **superalcanzable** que es tal que **superalcanzable(A)** aparece en el modelo cuando desde todo nodo del grafo distinto de A, hay un camino hasta A.
3. Dado un grafo que contiene posiblemente un ciclo, ¿es posible eliminar hasta dos arcos y transformarlo en acíclico? Haz un programa que sirva para contestar una pregunta. Tu programa debe ser tal que en sus modelos aparezcan hasta dos átomos de la forma `eliminar(x,y)`, indicando cuáles son los arcos que se deben eliminar para que el grafo sea acíclico (`eliminar(x,y)` indica que el arco(x,y) puede ser eliminado). Si no es posible eliminar dos arcos para transformarlo en acíclico, tu programa no debe tener modelos.
4. Una componente fuertemente convexa (CFC) es un subconjunto maximal de nodos tal que existe un camino entre cada par de nodos. Define un programa tal que cada modelo de este es tal que `cfc(v)` aparece en el modelo para todo nodo `v` que aparece en una misma CFC. Tu programa debiera tener un modelo por cada CFC que el grafo posea.