



Ayudantía 2

Negación y modelación en Clingo

Por Blanca Romero y Willy Pugh

25 de agosto 2023



Contenidos de hoy

1. Reglas básicas
2. Variables
3. Negación
4. Restricciones de cardinalidad
5. Modelación



Reglas Básicas

$p :- q$



Reglas básicas

Una **regla** es de la siguiente forma:

$$Head \leftarrow Body$$

Si *Body* se encuentra en el modelo, entonces *Head* también.



Reglas básicas - clingo

```
p :- q.  
% si q se encuentra en el modelo, p también  
  
p :- q, r.  
% si q y r están en el modelo, p también
```

← Estas son **reglas**



Reglas básicas - clingo

```
p.  
% p se encuentra en el modelo  
  
:- q.  
% q NO se encuentra en el modelo  
  
perro(pepe).  
% el término "pepe" cumple el predicado  
"perro"
```

← Esto es un **hecho**

← Esto es una **restricción** (regla sin *Head*)

← Esto es una **proposición**



Variables

$p(X)$



Variables

```
perro(pepe).
```

```
perro(frodo).
```

```
gato(pililo).
```

```
animal(pepe) :- perro(pepe).
```

```
animal(frodo) :- perro(frodo).
```

```
animal(pililo) :- gato(pililo).
```

Si “pepe” es un perro, es un animal.

Si “frodo” es un perro, es un animal.

Si “pililo” es un gato, es un animal.



Variables

```
perro(pepe).  
perro(frodo).  
gato(pililo).
```

```
animal(x) :- perro(pepe).  
animal(frodo) :- perro(frodo).  
animal(pililo) :- gato(pililo).
```

Si “pepe” es un perro, es un animal.
Si “frodo” es un perro, es un animal.
Si “pililo” es un gato, es un animal.



Variables

```
perro(pepe).  
perro(frodo).  
gato(pililo).
```

```
animal(X) :- perro(X).  
animal(X) :- gato(X).
```

Las variables nos sirven para **generalizar** las reglas:

Todo perro es un animal.

Todo gato es un animal.

* Las variables se escriben con **mayúscula**, mientras los términos se escriben con **minúscula**



Variables - Ejemplo

```
profesor(jorge).  
profesor(hans).  
  
catedra(X) :- profesor(X).  
catedra(X,Y) :- profesor(X), profesor(Y).  
  
#show catedra/1.  
#show catedra/2.
```

¿Podemos mejorarlo?
¡Vamos a la consola!



Negación

not $p(X)$



Negación

Definición

La palabra **not** indica la **ausencia** de un átomo en un modelo.

En la lógica formal, por ejemplo, **not p.** equivale a escribir **$\neg p$**



Negación

$\text{not } p.$

p **no** pertenece al modelo

* en este caso es equivalente a $\text{:- } p.$

$p \text{ :- not } q.$

p pertenece al modelo sólo si q **no** pertenece a él



Negación

Algunos casos útiles

```
cansado(X) :- not dormir(X), not comer(X).
```

Definiciones que naturalmente traen una negación

```
catedra(X,Y) :- profesor(X), profesor(Y), X != Y, not catedra(Y,X).
```

Evitar redundancias.

¡Ojo! No confundir con reglas de **cabeza vacía**.



Negación - Ejemplo

```
persona(willy).  
persona(blanca).  
persona(amanda).  
  
int_lactosa(amanda).  
% amanda es intolerante a la lactosa :(  
  
come_helado(A) :- persona(A), not int_lactosa(A).
```

Una persona come helado si
no es intolerante a la lactosa



Restricciones de cardinalidad

$n \{p; q; r\} m$



Restricciones de cardinalidad

```
{a; b; c}.
```

Este programa nos entrega todas las combinaciones posibles, de cualquier tamaño. Es decir, nos entrega los siguientes 8 modelos:

Modelo 1: { }

Modelo 2: {a}

Modelo 3: {b}

Modelo 4: {c}

Modelo 5: {a b}

Modelo 6: {a c}

Modelo 7: {b c}

Modelo 8: {a b c}



Restricciones de cardinalidad

1 {a; b; c} 2.

Este programa nos entrega todas las combinaciones posibles, de tamaño entre 1 y 2. Es decir, nos entrega los siguientes 6 modelos:

Modelo 1: {a}

Modelo 2: {b}

Modelo 3: {c}

Modelo 4: {a b}

Modelo 5: {a c}

Modelo 6: {b c}



Restricciones de cardinalidad

* La proposición *come_helado*(*A*, *B*) indica que la persona *A* come helado de sabor *B*.

```
persona(juan).  
persona(pedro).  
  
1 {  
  come_helado(A, chocolate);  
  come_helado(A, vainilla);  
  come_helado(A, frutilla)  
} 2 :- persona(A).
```

← Esta restricción indica que cada persona come entre 1 y 2 sabores (entre chocolate, vainilla y frutilla)



Restricciones de cardinalidad

```
persona(juan).  
persona(pedro).  
  
helado(chocolate).  
helado(vainilla).  
helado(frutilla).  
  
1 {come_helado(A, B): helado(B)} 2 :- persona(A).
```

← Podemos **generalizar** para todos los sabores de helado con esta notación.



Restricciones de cardinalidad - Ejemplo

```
#const n = 10.  
tiempo(1..n). % tiempo(1), ... , tiempo(10)  
  
persona(willy).  
persona(blanca).  
persona(amanda).  
  
{ vota(P, T) : tiempo(T) } 1 :- persona(P).  
  
:- vota(P1, T), vota(P2, T), P1 != P2.
```

* $vota(P, T)$ indica que la persona P vota en el tiempo T

¿Qué hacen estas reglas?



Restricciones de cardinalidad - Ejemplo

```
#const n = 10.  
tiempo(1..n). % tiempo(1), ... , tiempo(10)  
  
persona(willy).  
persona(blanca).  
persona(amanda).  
  
{ vota(P, T) : tiempo(T) } 1 :- persona(P).  
  
:- vota(P1, T), vota(P2, T), P1 != P2.
```

* $vota(P, T)$ indica que la persona P vota en el tiempo T

- ← Todas las personas pueden votar máximo una vez
- ← No pueden votar dos personas al mismo tiempo



Modelación



Modelación

Algunos tips

- Como es un lenguaje declarativo, pensar en el problema **resuelto**, no el cómo resolverlo.
- Probar que las reglas funcionan **individualmente** sirve para entender qué funciona y qué no.
- Soltar la mano, especialmente en pasar predicados lógicos a Clingo.
- ¡Ejercitar! Hay muchos ejemplos, básicos, medios y avanzados.



Modelación - Campeonato

Se tienen 8 jugadores de tenis que quieren jugar un campeonato de dobles. Modela un programa que indique los posibles equipos.



Modelación - Plantaciones

Haz un programa que indique quién debe plantar cada planta (y cuándo):

- Se tienen 3 trabajadores que trabajan plantando distintas plantas (Juan1, Juan2 y Juan3)



Modelación - Plantaciones

Haz un programa que indique quién debe plantar cada planta (y cuándo):

- Se tienen 3 trabajadores que trabajan plantando distintas plantas (Juan1, Juan2 y Juan3)
- Se deben plantar 7 plantas en un rango de 3 días. (sauce, eucalipto, almendro, naranjo, manzano, ciruelo y cactus)



Modelación - Plantaciones

Haz un programa que indique quién debe plantar cada planta (y cuándo):

- Se tienen 3 trabajadores que trabajan plantando distintas plantas (Juan1, Juan2 y Juan3)
- Se deben plantar 7 plantas en un rango de 3 días. (sauce, eucalipto, almendro, naranjo, manzano, ciruelo y cactus)
- Cada trabajador puede plantar como máximo una planta al día.



Modelación - Plantaciones

Haz un programa que indique quién debe plantar cada planta (y cuándo):

- Se tienen 3 trabajadores que trabajan plantando distintas plantas (Juan1, Juan2 y Juan3)
- Se deben plantar 7 plantas en un rango de 3 días. (sauce, eucalipto, almendro, naranjo, manzano, ciruelo y cactus)
- Cada trabajador puede plantar como máximo una planta al día.
- Algunos trabajadores tienen alergias, por lo que no pueden plantar ciertas plantas:
 - Juan1 es alérgico al almendro, naranjo y sauce.
 - Juan2 es alérgico al sauce, cactus y eucalipto.
 - Juan3 es alérgico al eucalipto.



Modelación - Plantaciones

Haz un programa que indique quién debe plantar cada planta (y cuándo):

- Se tienen 3 trabajadores que trabajan plantando distintas plantas (Juan1, Juan2 y Juan3)
- Se deben plantar 7 plantas en un rango de 3 días. (sauce, eucalipto, almendro, naranjo, manzano, ciruelo y cactus)
- Cada trabajador puede plantar como máximo una planta al día.
- Algunos trabajadores tienen alergias, por lo que no pueden plantar ciertas plantas:
 - Juan1 es alérgico al almendro, naranjo y sauce.
 - Juan2 es alérgico al sauce, cactus y eucalipto.
 - Juan3 es alérgico al eucalipto.
- Un Juan2 se enferma al día 2 y durante ese día no puede trabajar.



Modelación - Plantaciones

Haz un programa que indique quién debe plantar cada planta (y cuándo):

- Se tienen 3 trabajadores que trabajan plantando distintas plantas (Juan1, Juan2 y Juan3)
- Se deben plantar 7 plantas en un rango de 3 días. (sauce, eucalipto, almendro, naranjo, manzano, ciruelo y cactus)
- Cada trabajador puede plantar como máximo una planta al día.
- Algunos trabajadores tienen alergias, por lo que no pueden plantar ciertas plantas:
 - Juan1 es alérgico al almendro, naranjo y sauce.
 - Juan2 es alérgico al sauce, cactus y eucalipto.
 - Juan3 es alérgico al eucalipto.
- Un Juan2 se enferma al día 2 y durante ese día no puede trabajar.
- Si Juan3 planta algo un día, tiene que descansar al día siguiente (no puede plantar algo dos días consecutivos)

¡Vamos paso por paso!



Ejercicio Propuesto

Organizar asientos

- Hay n sillas alrededor de una mesa **circular**, y queremos elegir una silla para cada invitado, que van del 1 al n , cumpliendo **2 condiciones**.
- Algunas personas se caen bien y quieren sentarse juntas, representadas por pares que serán entregados de la forma `like(i,j ; k,l ; m,n ; ...)`.
- Además, hay personas que no se caen muy bien :(, por lo que no las debes sentar juntas, que será indicado por `dislike(o,p ; q,r ; s,t ; ...)`.



Ayudantía 2

Negación y modelación en Clingo

Por Blanca Romero y Willy Pugh

25 de agosto 2023