Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación

Guía — Answer Set Programming

IIC2613 - Inteligencia Artificial

Primer Semestre, 2019

1. Escriba todos los conjuntos respuesta de:

$$\begin{aligned}\Pi = \{&p(X) \leftarrow \mathit{not}\ q(X),\\ &q(X) \leftarrow \mathit{not}\ p(X),\\ &r(X) \leftarrow p(X)\},\end{aligned}$$

   de $\Pi' = \Pi \cup \{s(a)\}$ y de $\Pi'' = \Pi \cup \{s(f(a))\}$. (Para responder, use sólo la definición de conjunto respuesta.)

2. Considere un programa Π cuyo único modelo contiene los átomos en $\{\mathit{bolita}(x) : x \in \{1, \ldots, n\}\}$ más exactamente un átomo de la forma $\{\mathit{color}(x, c)\}$, para cada $x \in \{1, \ldots, n\}$, donde $c \in \{1, \ldots, m\}$. Además, para cada $c \in \{1, \ldots, m\}$, existe un $x \in \{1, \ldots, n\}$ tal que el modelo contiene $\mathit{color}(x, c)$.

   a) Dé una expresión en función de n y m para el número de modelos que tiene Π al agregarle la sentencia $4\{\mathit{enUrna}(x) : \mathit{bolita}(x)\}5$.

   b) Diga qué regla(s)/restriccion(es) puede agregar a Π para que el número de modelos resultantes corresponda al número de formas que es posible llenar una bolsa con exactamente 4 bolitas de 2 colores distintos. ¿Cuántos modelos tiene el programa resultante si $m = 1$?

3. Demuestre que los programas que solo tienen reglas de la forma $q \leftarrow p_1, \ldots, p_n$, con $n \geq 0$, (es decir sin negación ni restricciones) siempre tienen exactamente un modelo.

4. Considere el programa de los regadores visto en clases (regadores.lp), pero ahora piense que la regla que describe cuándo una planta está regada se escribe así:

$$\mathit{regada}(P) \leftarrow \mathit{encendido}(R), \mathit{planta}(P), \mathit{regador}(R), \mathit{riega}(R, P), \mathit{not}\, \mathit{defectuoso}(R)$$

   Diga cómo resolver usando ASP el problema de descubrir cuál o cuáles son los regadores defectuosos dados: (1) una lista de regadores encendidos (2) una lista de regadores no encendidos (3) una lista de plantas que están regadas (4) una lista de plantas que no están regadas. Para llevarlo al contexto del ejemplo de clases, su programa debiera servir para encontrar que si r4 está encendido y el oregano no está regado, entonces r4 está defectuoso.

5. Diga cómo modificar el programa `grafo.lp`, visto en clases, para que tenga un modelo si y solo si es posible agregar hasta una arista adicional al grafo dado por el usuario, para que X sea alcanzable desde Y, donde X e Y son dados.

6. Suponga que, como en los ejemplos de clases, se representa un grafo dirigido con los predicados $\mathit{nodo}/1$ y $\mathit{arco}/2$, tal que $\mathit{nodo}(a)$ representa que a es un nodo y $\mathit{arco}(u, v)$ representa que hay un arco dirigido entre u y v. Sea Π un programa que describe a un grafo.

a) Agregue reglas a Π para que los modelos de Π contengan átomos de la forma $camino(u, v, k)$ cuando hay un camino de k aristas entre u y v. ¿Qué ocurre si su grafo tiene un ciclo?

b) Agregue reglas a Π para que los modelos de Π contenga el átomo *ciclo* si y sólo si el grafo tiene un ciclo. ¿Cómo cambia esta definición cuando el grafo es no dirigido y se usa el predicado $unidos(x, y)$, definido en términos de *arco*, para expresar que hay una arista no dirigida entre x y y?

c) Agregue reglas a Π para que los modelos de Π contengan el átomo $superalcanzable(u)$ si es posible construir un camino desde todo nodo del grafo v, con $v \neq u$, hasta u.

d) Un subconjunto de nodos fuertemente conexos en un grafo dirigido es un subconjunto V' de nodos tal que para todo $u, v \in V'$, v es alcanzable desde u. Describa como extender el programa para que cada modelo en cuentre un subconjunto de nodos fuertemente conexo y que sea tal que el programa no tenga modelos si no existen estos subconjuntos.

7. Un *cubrimiento de vértices* en un grafo no dirigido $G = (V, E)$ es un conjunto $V' \subseteq V$ tal que para todo $\{u, v\} \in E$ se tiene que $u \in V'$ o $v \in V'$ (o ambos). (Más detalles de este problema: google *vertex cover*.) Suponga que ha escrito un programa en donde $unidos(u, v)$ representa que $\{u, v\} \in E$ y $nodo(v)$ representa que $v \in V$.

   a) Un ingeniero de conocimiento propone que agregando esta regla al programa:

   $$\{enCubrimiento(X); enCubrimiento(Y)\}\, 2 \leftarrow unidos(X, Y)$$

   el conjunto respuesta del programa será tal que el cubrimiento está dado por los nodos que pertenecen a la extensión del predicado *enCubrimiento* y que, adicionalmente, para cada cubrimiento existente, habrá un modelo que lo describe.

   Muestre que la afirmación no es correcta.

   b) Proponga una forma alternativa (y correcta) para encontrar un cubrimiento de vértices que consista en *agregar* una restricción (y tal vez otro predicado) al programa del ingeniero.

   c) Proponga una forma alternativa (y correcta) para encontrar un cubrimiento de vértices que consista en *modificar* la sentencia dada por el ingeniero.

8. Escriba un programa que resuelva el puzle (SEND+MORE=MONEY) descrito con más detalle acá: `https://en.wikipedia.org/wiki/Verbal_arithmetic`.