COMP4418, 2019 – Exercises

---

1 Answer Set Programming

1.1 Modelling

Let $S = \{s_1, \ldots, s_n\}$ be a set of sets. A *set cover* of S is a set $C \subseteq S$ such that $\bigcup_{s \in S} s = \bigcup_{s \in C} s$. A *k-set cover* is a set cover of size k, that is, $|C| = k$.

For instance, for an input $S = \{\{1,2\},\{2,3\},\{4,5\},\{1,2,3\}\}$, there is a 2-set cover $C = \{\{1,2,3\},\{4,5\}\}$ since $\bigcup_{s \in S} s = \{1,2\} \cup \{2,3\} \cup \{4,5\} \cup \{1,2,3\} = \{1,2,3\} \cup \{4,5\} = \bigcup_{s \in C} s$.

Write an ASP program that decides the k-Set-Cover problem:

> Input: a set of sets and a natural number $k \geq 0$.
>
> Problem: decide if there is a k-set cover.

Assume the input parameter $S = \{s_1, \ldots, s_n\}$ is encoded by a binary predicate `s` in the way that $x \in s_i$ iff `s(`i`,`x`)`. The input parameter k is given as constant symbol `k`. Use a unary predicate `c` to represent the output C in the way that $s_i \in C$ iff `c(`i`)`.

```
% Instance encoding of the above example:
% s(1, (1;2)). is a shorthand for s(1,1). s(1,2).
s(1, (1;2)).
s(2, (2;3)).
s(3, (4;5)).
s(4, (1;2;3)).

% Helper predicates.
universe(X) :- s(S,X).
covered(X) :- c(S), s(S,X).

% Generate candidate of cardinality k.
k { c(S) : s(S,X) } k.

% Test that the candidate covers the whole universe.
:- universe(X), not covered(X).

#show c/1.
```

1.2 Semantics

Consider the following program P.

$$a.$$
$$c \leftarrow \text{not}\, b, \text{not}\, d.$$
$$d \leftarrow a, \text{not}\, c.$$

Determine the stable models of S.

S	Reduct P^S	Stable model?
$\{a, b, c, d\}$	$a.$	✗
$\{a, b, c\}$	$a.$	✗
$\{a, b, d\}$	$a.\quad d \leftarrow a.$	✗
$\{a, c, d\}$	$a.$	✗
$\{b, c, d\}$	$a.$	✗
$\{a, b\}$	$a.\quad d \leftarrow a.$	✗
$\{a, c\}$	$a.\quad c.$	✓
$\{a, d\}$	$a.\quad d \leftarrow a.$	✓
$\{b, c\}$	$a.$	✗
$\{b, d\}$	$a.\quad d \leftarrow a.$	✗
$\{c, d\}$	$a.$	✗
$\{a\}$	$a.\quad c.\quad d \leftarrow a.$	✗
$\{b\}$	$a.\quad d \leftarrow a.$	✗
$\{c\}$	$a.\quad c.$	✗
$\{d\}$	$a.\quad d \leftarrow a.$	✗
$\{\}$	$a.\quad c.\quad d \leftarrow a.$	✗