CS395T, Spring 2011

The Language of CLINGO: Cardinality Expressions

For any formula $F(x)$ and any positive integer n, by

$$n\ \{x\ :\ F(x)\}$$

we denote the formula

$$\exists x_1 \cdots x_n \left(\bigwedge_{1 \leq i \leq n} F(x_i) \wedge \bigwedge_{1 \leq i < j \leq n} x_i \neq x_j \right),$$

which expresses that there exist at least n values of x such that $F(x)$. By

$$\{x\ :\ F(x)\}\ n,$$

where n is a nonnegative integer, we denote the formula

$$\neg(n{+}1\ \{x\ :\ F(x)\})$$

("there are at most n values of x such that $F(x)$"). A conjunction of the form

$$(m\ \{x\ :\ F(x)\})\ \wedge\ (\{x\ :\ F(x)\}\ n)$$

("the number of values of x such that $F(x)$ is between m and n") can be written as

$$m\ \{x\ :\ F(x)\}\ n.$$

These abbreviations can be used in CLINGO programs, for example:

```
p(a;b;c).
{q(X)} :- p(X).
:- not 2 { <X> : q(X) } 2.
```

The stable models of this program represent the 2-element subsets of $\{a, b, c\}$.

A pair of rules of the form

$$\{F(x)\} \leftarrow G(x),$$
$$\bot \leftarrow \neg(m\ \{F(x) \wedge G(x)\}\ n)$$

can be written as

$$m\ \{F(x) : G(x)\}\ n,$$

and similarly when only one of the boundaries m, n is present. Using this abbreviation we can rewrite the CLINGO program above as

```
p(a;b;c).
2 { <X> : q(X) : p(X)} 2.
```

Problem 32^e. Write (and test!) a CLINGO program for generating cliques of cardinality $\geq n$.

Problem 33^e. A set of vertices in a graph is *independent* if no two of its elements are adjacent. Write a CLINGO program for generating independent sets of cardinality $\geq n$.

Here is a CLINGO program for grap coloring:

```
% File color
1 { <C> : color(X,C) : C=1..n } 1 :- vertex(X).
:- edge(X,Y), color(X,C), color(Y,C).
#hide. #show color(X,C).
```

The command line

```
% clingo -c n=4 color graph
```

instructs clingo to determine whether the graph in file `graph` is 4-colorable.

Problem 34^e. File `g_10_25` describes a graph with 10 vertices and 25 edges. Find a 5-coloring of this graph that uses each color exactly twice.