

T-79.5102
Laskennallisen logiikan erikoiskurssi
Laskuharjoitus 10
Ratkaisut

Syksy 2005

1. %% Graafi esitetään faktoja vertex(_) ja egde(,,) käyttäen,
%% k on halutun peiton maksimikoko.

% Solmupeittoon valitaan solmujen osajoukko.

{ in_cover(V) }k :- vertex(V).

% Peittoon saa kuulua korkeintaan k solmua.

:- k+1{ in_cover(V): vertex(V) }.

% Tarkastetaan, etteivät molemmat kaaren päätepisteet kuulu

% peittoon.

:- edge(U,V), not in_cover(U), not in_cover(V).

2. %% Graafi esitetään faktoja vertex(_) ja egde(,,) käyttäen.

% Valitaan ytimeen solmujen osajoukko.

{ in_kernel(V) } :- vertex(V).

% Tarkastetaan, että jokaiselle solmulle V, joka ei kuulu

% ytimeen löytyy kaari, jonka loppupiste V on ja alkupiste

% kuuluu ytimeen.

:- { in_kernel(U): edge(U,V) } 0, not in_kernel(V), vertex(V).

% Tarkastetaan, että kaarien molemmat päätepisteet eivät kuulu

% ytimeen.

:- edge(U,V), in_kernel(U), in_kernel(V).

3. Esitetään yleisempi ratkaisu, joka ratkaisee $n:n$ kuningattaren ongelman halutulla kuningattarien lukumäärällä n .

```
% q(X,Y): kuningatar on ruudussa (X,Y)
% negq(X,Y): kuningatar ei ole ruudussa (X,Y)
% hasq(X): sarakkeessa X on kuningatar
% d(1..queens): laudan dimensio faktoina

% Asetetaan kuningattaria ruutuihin.
q(X,Y) :- d(X), d(Y), not negq(X,Y).
negq(X,Y) :- d(X), d(Y), not q(X,Y).

% Tarkastetaan, etteivät kuningattaret uhkaa toisiaan.
:- d(X), d(Y), d(X1), q(X,Y), q(X1,Y), X1 != X.
:- d(X), d(Y), d(Y1), q(X,Y), q(X,Y1), Y1 != Y.
:- d(X), d(Y), d(X1), d(Y1), q(X,Y), q(X1,Y1),
    X != X1, Y != Y1, abs(X - X1) == abs(Y - Y1).

% Tarkastetaan, että jokaisessa sarakkeessa on kuningatar.
:- d(X), not hasq(X).
hasq(X) :- d(X), d(Y), q(X,Y).

d(1..queens).

% Kahdeksan kuningattaren ongelman ratkaiseminen:
% lparse -d none -c queens=8 queens.lp | smodels 1

Vaihtoehtoinen koodaus:

% q(X,Y): kuningatar on ruudussa (X,Y)
% d(1..queens): laudan dimensio faktoina

% Asetataan täsmälleen yksi kuningatar kuhunkin
% sarakkeeseen.

1{ q(X,Y):d(Y) }1 :- d(X).

% Tarkastetaan, etteivät kuningattaret uhkaa toisiaan.

:- d(X), d(Y), d(X1), q(X,Y), q(X1,Y), X1 != X.
```

```
:- d(X), d(Y), d(X1), d(Y1), q(X,Y), q(X1,Y1),
   X != X1, Y != Y1, abs(X - X1) == abs(Y - Y1).

d(1..queens).

% Kahdeksan kuningattaren ongelman ratkaiseminen:
% lparse -d none -c queens=8 queens.lp | smodels 1
```