**Universität Ulm**
**Fakultät für Informatik**
Prof. Dr. Thom Frühwirth

**Constraint-Programmierung**, WS 02/03
**Übungsblatt 5**

Abgabe bis Mi, 4.12.02, 10:00

## Constraint Handling Rules (CHR)

### Aufgabe 5-1

Vergleichen Sie die in den Teilaufgaben gegebenen Programme, die aus jeweils einer der gegebenen Regeln bestehen, indem Sie die gegebenen Ziele auswerten. Überprüfen Sie Ihre Antworten am Rechner und machen Sie sich klar, warum man unterschiedliche Ergebnisse erhält.

Zweiköpfige Simplifikationsregeln:

```
c1 @ c(X), c(X) <=> q(X,X).
c2 @ c(X), c(Y) <=> r(X,Y).
```

Ziele: `c(X), c(X)`, sowie `c(X), c(Y)` und `c(X), c(Y), X=Y`

Weitere Variationen:

```
q1 @ p(X,Z), q(Z,Y) <=> q(X,Y).
q2 @ q(Z,Y), p(X,Z) <=> q(X,Y).
q3 @ p(X,Z), q(Z,Y) ==> q(X,Y).
q4 @ q(Z,Y), p(X,Z) ==> q(X,Y).
q5 @ p(X,Z) \ q(Z,Y) <=> q(X,Y).
q6 @ q(Z,Y) \ p(X,Z) <=> q(X,Y).
```

Ziele: `p(A,B), q(B,C)` und `p(A,B), q(B,C), p(D,A)`

Bei dieser Aufgabe ist nichts abzugeben.

### Aufgabe 5-2

Implementieren Sie in CHR die Constraints `less/2` (für $<$) und `leq/2` (für $\leq$) mit ihren Beziehungen untereinander als CHR-Programm. Sie können sich an dem in der Vorlesung vorgestellten CHR-Programm für das Constraint $\leq$ orientieren.
Als Anfrage fassen Sie ihren Nachnamen als Folge von einbuchstabigen Variablen auf, zwischen denen jeweils ein `leq`-Constraint eingefügt wird. Was ist das Ergebnis?
Beispiel: Der Name *Fruehwirth* ergibt die Anfrage
`F leq R, R leq U, U leq E, E leq H, H leq W, W leq I, I leq R, R leq T, T leq H`
und die Antwort `F leq E, H=E, I=E, R=E, T=E, U=E, W=E`.
Geben Sie noch mindestens drei weitere Anfragen an.

Ergänzen Sie das Programm um die "constrain-and-generate" Version des Sortier-Beispiels von Übungsblatt 3, wobei Sie das clpq-Constraint `=<` durch das CHR-Constraint `leq` ersetzen. Vergleichen Sie das alte und neue Programm anhand von mindestens zwei Beispielen.

Geben Sie das lauffähige Programm zusammen mit den verlangten Beispielen ab.