1999 Prolog Question 1. WFC

p5q7
WFC
p12q8

The following story is from N. Wirth's textbook *Algorithms+data structures=programs* (1976):

> I married a widow (call her W) who has a grown-up daughter (D). My father (F), who visited us quite often, fell in love with my step-daughter and married her. Hence my father became my son-in-law and my step-daughter became my mother. Some months later, my wife gave birth to a son (S1), who became the brother-in-law of my father, as well as my uncle. The wife of my father — that is, my step-daughter — also had a son S2.

Using Prolog, create a list of facts that represents the situation in the above story [5 marks]. Add rules defining the family relationships (such as father-in-law) described in the story [5 marks]. Show how a Prolog system would use your program to prove the goal 'I am my own grandfather' [10 marks].

Solution Guidelines

First define a fact for each relationship in the story. Use constants for each person, including the ones unknown in the story: wh for widow's husband and m for my mother. To make the rules easier, we express symmetry in the facts:

```
male(i). male(f). male(s1). male(s2). male(wh).
female(m). female(w). female(d).

childOf(i, f, m). childOf(i, m, f).
married(i, w). married(w, i).
childOf(d, w, wh). childOf(d, wh, w).
married(f, d). married(d,f).
childOf(s1, i, w). childOf (s1, w, i).
childOf(s2, d, f). childOf (s2, f, d).
```

Next define some derived relationships. The final part of the question depends on fathers as including stepfathers.

```
sonInLawOf(X, Y) :- childOf(D, Y,_), female(D), married(D, X).
stepDaughterOf(X, Y) :- married(Y, M), childOf(X, M, _), not(childOf(X, Y, _)), female(X).
stepSonOf(X, Y) :- married(Y, M), childOf(X, M, _), not(childOf(X, Y, _)), male(X).
fatherOf(X, Y) :- childOf(Y, X, _), male(X).
fatherOf(X, Y) :- stepSonOf(Y, X).
grandfatherOf(X, Y) :- fatherOf(X, F), fatherOf(F, Y).
etc.
```

Finally, the goal is proved (my father is my father by blood as well as my son by marriage) as follows: grandFatherOf(i, i) because fatherOf(f, i) and fatherOf(i, f). fatherOf(f,i) is the blood relation <- childOf(i,f,m). fatherOf(i, f) <- stepSonOf(f,i) <- etc.