Intelligent Systems: Reasoning and Recognition

James L. Crowley

a) Define an abstract class for person with slots NAME (capitalized), father, mother, brother and sister. The slots for brother and sister must be multi-slots so that they can contain a list.

```
(defclass PERSON (is-a USER)
     (role abstract)
     (slot NAME (create-accessor read-write))
     (slot father (create-accessor read-write) )
     (slot mother (create-accessor read-write) )
     (multislot brothers (create-accessor read-write))
     (multislot sisters (create-accessor read-write))
)
```

Define a concrete class for MAN as a subclass of person, with the slots "wife" and "gender" having fixed values of "male". Assure that that object of class MAN can activate rules.

Define a concrete class for WOMAN as a subclass of person, with the slots "husband" and "gender" having fixed values of "female". Assure that that object of class MAN can activate rules.

b) Rules to build the family structure

Create a rule ask-wife to ask for the name of the wife for a man if the wife is nil. Create a rule ask-husband to ask for the name of the husband for a woman if the husband is nil. Write the rule create-husband and create-wife that will create a MAN or WOMAN for the husband or wife if they do not currently exist.

c) Write the rules ask-father and ask-mother to create the father and mother for each person if these are nil. Do not create the father or mother if the user names them as unknown.

d) define a message handler named get-father-name to get the NAME of the father of a PERSON. If the father is unknown, then print a message stating that the father is unknown.

e) Define a message handler named get-paternal-grandfather to get a ptr to the father of the father of a PERSON. If a father is unknown, then print a message stating that the father is unknown.

f) Define a message handler that return the names of the paternal grandfather for a person