

**MSc Computing Science 2012-2013**  
**Mock Prolog Laboratory Test**

**Imperial College London**

**Monday 10 December 2012, 17.00-18.00**

- Log into the Lexis exam system using your DoC login as both your login and as your password (**do not use your usual password**).
- You are required to:  
Submit an extension of the program in the file **mockAnswers.pl** according to the specifications, overleaf. You will find file **mockAnswers.pl** in your Lexis home directory (**/exam**). If you are missing this file please alert one of the invigilators.
- **Save your work regularly.**
- The system will log you out automatically once the exam has finished. **It is therefore important that you save your work and quit your editor when you are told to stop writing.** No further action needs to be taken to submit your files – the final state of your Lexis home directory (**/exam**) will be your submission.
- No communication with any other student or with any other computer is permitted.
- Do not use any Prolog list processing or aggregation built-in functions, except, if required, *member*, *append*, *length*, *setoff*, *findall*.

1. Some self-explanatory facts are given in the file **mockAnswers.pl**. Add the following definitions.

i) **ecb(M)**

to mean M is a mother who is entitled to child benefit. A mother is entitled to child benefit if she has a child who is 14 years old or younger, or if she is not employed and has a child who is older than 14 but not older than 16.

Example: the query `ecb(X)`

Should succeed with `X = mary; X = rita ; X=tina; X = jane.`

There may be repeated answers. That is acceptable and correct.

ii) **mother\_of\_the\_youngest(M)**

to mean M is the mother of the youngest child. If there is more than one child with the same youngest age, then M should be the mother whose name comes alphabetically before the others.

Example: the query `mother_of_the_youngest(M)`

Should succeed with `M=rita.`

iii) **mothers\_of\_young(LM)**

to mean LM is the list of all mothers who have a child of 10 or younger. LM should be sorted and have no repetition. If there are no mothers with children of 10 or younger then the query `mothers_of_young(LM)` should fail.

Example: the query `mothers_of_young(LM)`

Should return `LM = [mary, rita, tina].`

2. Write programs for the following relations and add them to your file.

i) **merge(L1, L2, L)**

You can assume that in any call to *merge* the first two arguments (corresponding to L1 and L2) are sorted lists of the same type. L should then be an ordered list resulting from merging the elements of L1 and L2.

Example: the query `merge([1,3,5,5], [2,3,4,6], L)`

Should return `L=[1,2,3,3,4,5,5,6].`

ii) **findElement(N, L, E)**

to mean E is the Nth element in list L. You can assume that in any call to *findElement* N and L are given. If N is less than 1 or is greater than the length of L then the query should fail.

Example: the query `findElement (3, [1,3,5,5], E)`

Should return `E=5.`