MA410 Prolog Practical 3 - Family Tree and NLP

1 Outputting to File

1.1 Short Example

(a) In prolog, use the command:

```
?- tell('*out.txt').
```

to create new output file `out.txt` where * stands for file path.

(b) Enter the command:

```
?- write('My results'), nl.
```

This will write the text 'My results' to the file followed by a new line, respectively.

(c) Now type in `?- told.`

(d) Edit the file `out.txt` and see how the content of the file has changed.

(e) Repeat step (b) without a file open for output.

When re-opening `out.txt` (if you do not want to lose results already recorded) use `append` rather than `tell`.

2 A.I. based family questions

2.1 First Example

Download and open the file `lab3qs.pl`.

(a) Create a predicate `uncle_list`.

(b) Alter code to allow the user to ask the question:
`?- Who are the uncles of X.`

(c) Open the file '`out.txt`' for output.

(d) Enter the command:
`?- write('The uncles of dawn are:'), nl.`

(e) Find the uncles of `dawn` and write to file by:
`?- Who are the uncles of dawn, write(Who), nl.`

(f) Now add to file sisters of someone in your family.

(g) Close the file for output by `?- told.`

2.2 More Natural Output

(a) Download file `ltos.pl` & include in `lab3qs.pl`.

(b) Create a predicate `brother_list`.

(c) Enter the code
```
who are the brothers of X:-
brother_list(X,Who), write_list(Who), !.
```

(d) Test the code with various inputs.

(e) Do the same for `sisters` and `uncles`.

(f) In prolog, try the following:

```
?- length([a,b], N).
?- length([c,d,e], N).
?- length([e,f], N), N > 1.
?- length([e,f], N), N < 2.
```

(g) Change the definition in (c) so the output is

`X has no brothers.`	if X has no brother.
`The brother of X is ...`	if X has only 1 brother.
`The brothers of X are ...`	if X has > 1 brother.

Note: you can break up output by entering multiple `write` commands separated by commas, e.g.

`write('it is '), write('sunny').`

will write '`it is sunny`' to the screen.

2.3 More Generalised Approach

(a) Restart Prolog and download file `lab3qsg.pl`.

(b) Check out how the univ '`=..`' operator works, see:
`http://www.cse.unsw.edu.au/~billw/prologdict.html`

Try, for example:
```
?- T =..[book, on, table].
?- T =..[brother, X, daniel], T.
```

(c) Define the set of parents as follows:
`parents(X, L):- setof(Y, parent(Y,X), L).`
(Note we define predicate as `parents` and not `parent_list`)

(d) Do the same for `grandparents`, `brothers`, `sisters`, `siblings` and `firstcousins`.

(e) Add the following line of code:
```
who are the X of Y :- T =..[X,Y,Who], T,
                      write_list(Who), !.
```

(f) Test out the code for various inputs, e.g.
`?- who are the grandparents of charlie.`

Task: Send e-mail with heading "NLP Family Lab" with a copy of `lab3qsg.pl` and a sample input and output from both the made-up family tree and from your own family tree.

2.4 Other Questions

(a) Allow the user to ask the following questions:

```
?- how many brothers has X.
?- has X any sisters.
?- are X and Y blood_related.
```