

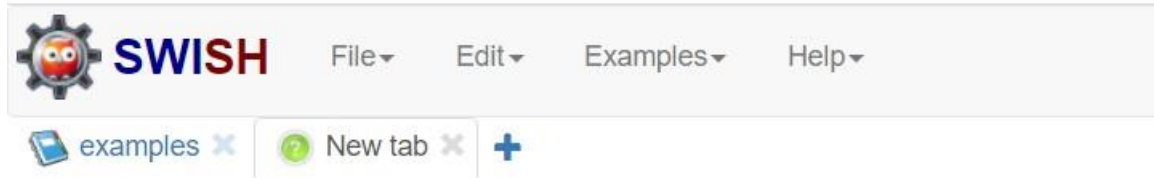
Cognitive Science Knowledge Representation & Organisation

Erasmus Programme
Professor Markos Dendrinou

PROLOG

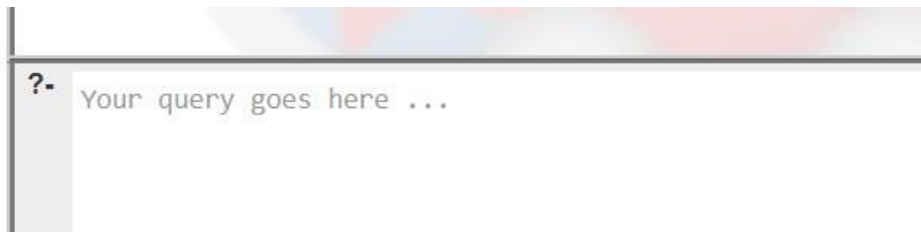
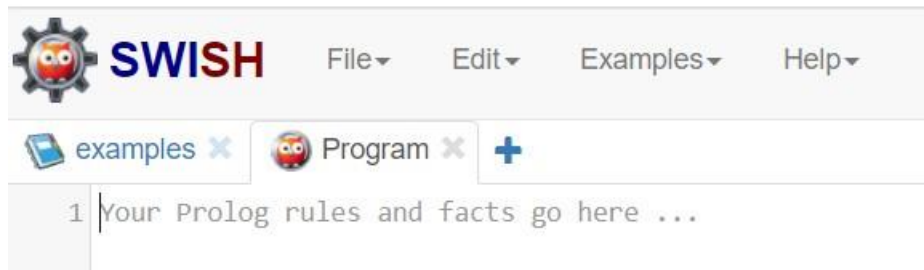
- Free online command line environment of PROLOG for automatic reasoning.
- Declaration area for facts and rules
- Query area for user queries
- Facts like `male(george) / child(nikos, mary)`
- Rules like `parent(X,Y) := child(Y,X)`
- Write in web : [swi prolog online...](#)

Starting Swi Prolog



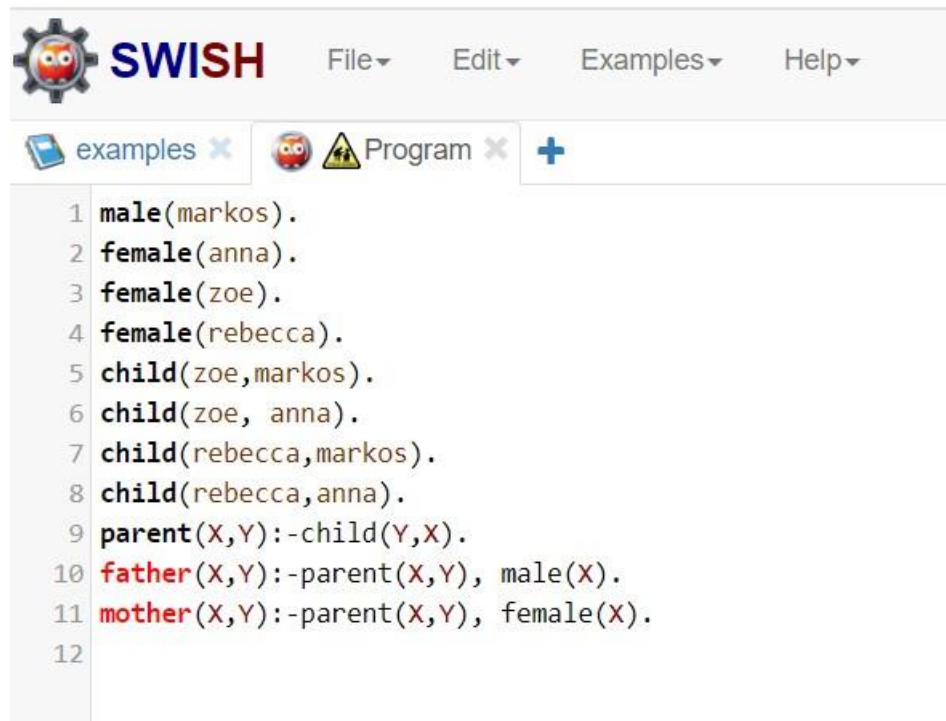
Create a **Program** **Notebook** here

based on **Empty** **Student** **CLP** **s(CASP)** profile



Declaration area (facts and rules)

- An example

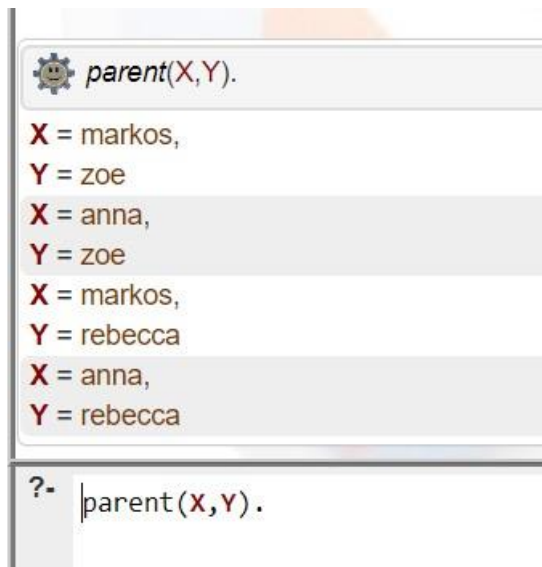


The screenshot shows the SWISH Prolog editor interface. The title bar includes the SWISH logo and menu items: File, Edit, Examples, and Help. Below the title bar, there are two tabs: 'examples' and 'Program'. The 'Program' tab is active, displaying a list of Prolog facts and rules. The code is as follows:

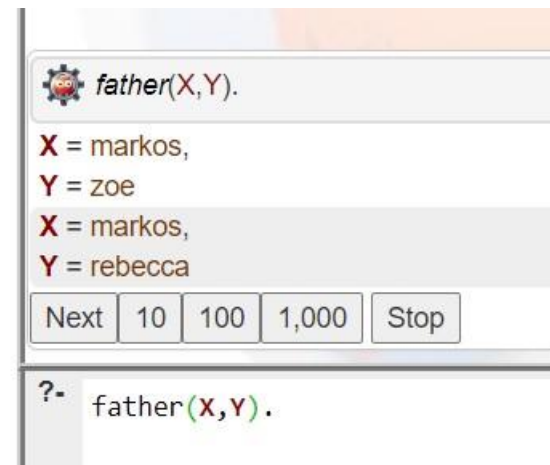
```
1 male(markos).
2 female(anna).
3 female(zoe).
4 female(rebecca).
5 child(zoe,markos).
6 child(zoe,anna).
7 child(rebecca,markos).
8 child(rebecca,anna).
9 parent(X,Y):-child(Y,X).
10 father(X,Y):-parent(X,Y), male(X).
11 mother(X,Y):-parent(X,Y), female(X).
12
```

Query area

- An example



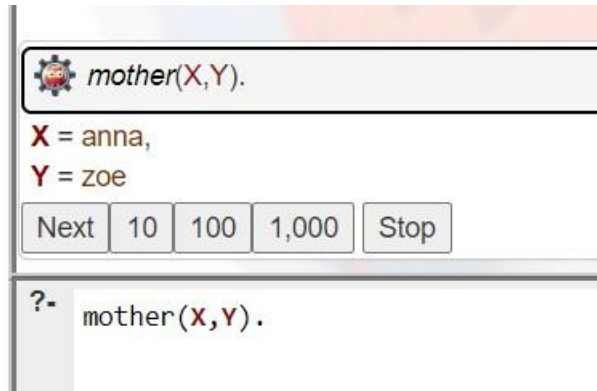
A screenshot of a query interface. At the top, there is a gear icon followed by the query `parent(X,Y).`. Below this, there are four rows of results, each with a light gray background. The first row shows `X = markos,` and `Y = zoe`. The second row shows `X = anna,` and `Y = zoe`. The third row shows `X = markos,` and `Y = rebecca`. The fourth row shows `X = anna,` and `Y = rebecca`. At the bottom of the interface, there is a prompt `?- parent(X,Y).` with a cursor.




A screenshot of a query interface. At the top, there is a gear icon followed by the query `father(X,Y).`. Below this, there are two rows of results, each with a light gray background. The first row shows `X = markos,` and `Y = zoe`. The second row shows `X = markos,` and `Y = rebecca`. Below the results, there is a control bar with buttons for `Next`, `10`, `100`, `1,000`, and `Stop`. At the bottom of the interface, there is a prompt `?- father(X,Y).` with the variables `X` and `Y` highlighted in green.

Query area (cont)

- An example

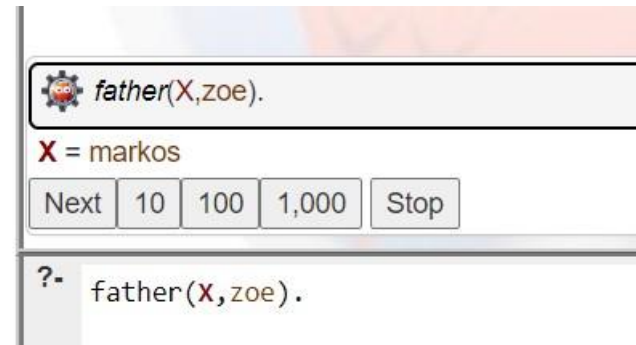



 *mother*(X,Y).

X = anna,
Y = zoe

Next 10 100 1,000 Stop

?- *mother*(X,Y).

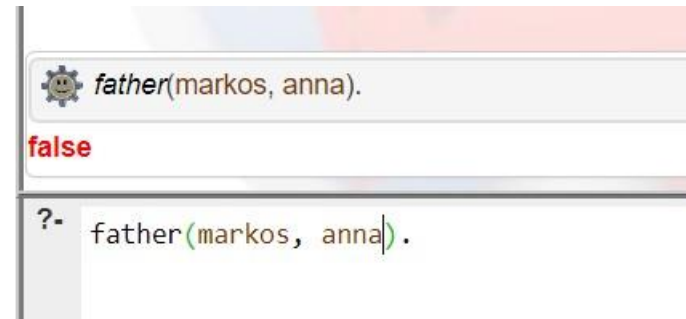



 *father*(X,zoe).

X = markos

Next 10 100 1,000 Stop

?- *father*(X,zoe).



 *father*(markos, anna).

false

?- *father*(markos, anna).

Exercise 5

- Make your family tree
- Your father, mother, brothers and sisters
- Write down in the declaration area the following facts: male or female for all of them, all the child pairs
- Write down next, after the facts, the rules: parent, father and mother
- Make in the query area the following queries:
- `parent(X,Y)` and take all the results through CTRL+ENTER and then the NEXT button.
- `father(X,Y)` and take all the results through CTRL+ENTER and then the NEXT button.
- `mother(X,Y)` and take all the results through CTRL+ENTER and then the NEXT button.

Exercise 5 (cont)

- A query with one variable and one constant
- A query with all constants to take true or false
- Make your family tree
- Take a snapshot of the declaration area and a snapshot for every query and its results.
- Put all the snapshots in a word file and send it as your 5th exercise by email