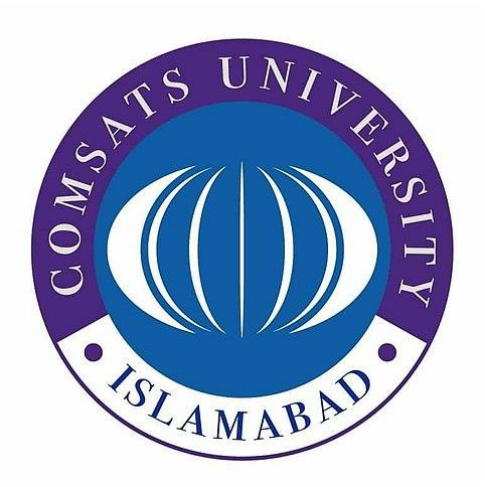


Lab 13

Artificial Intelligence

BY

Muhammad Naqeeb

FA20-BCS-012

TO

Sir Mr. Qazi Zia

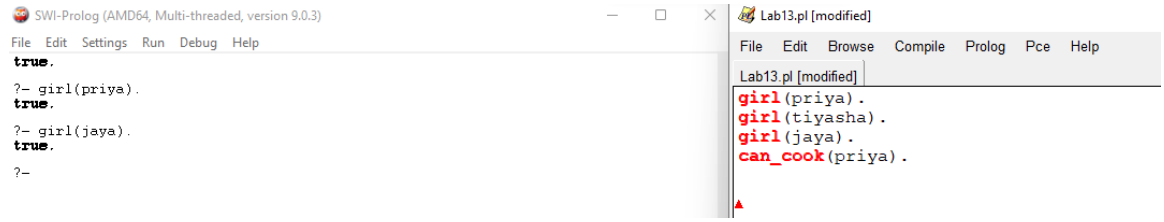
DEPARTMENT OF COMPUTER SCIENCE

COMSATS INSTITUTE OF INFORMATION AND TECHNOLOGY ATTOCK CAMPUS.

Lab 13 Complex Knowledge-bases

Knowledge Base 1

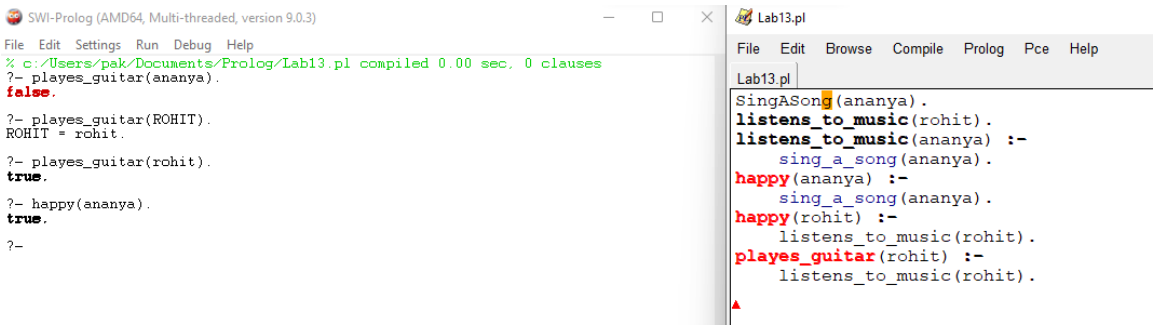
```
girl(priya).
girl(tiyasha).
girl(jaya).
can_cook(priya).
```



Knowledge Base 2

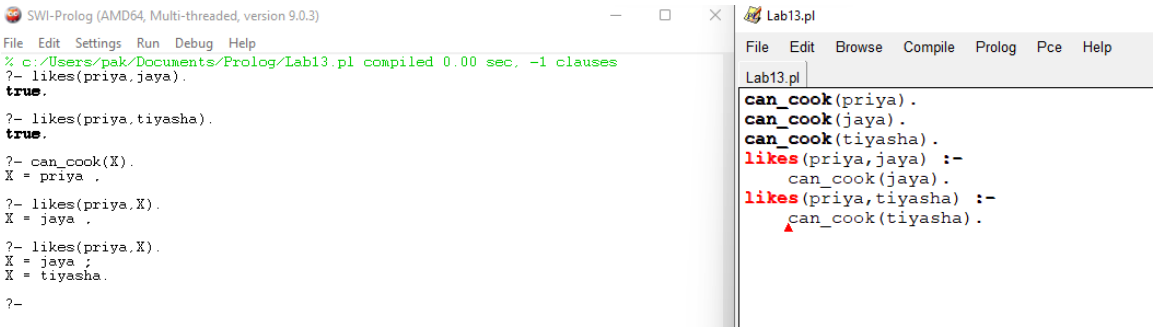
Let us see another knowledge base, where we have some rules. Rules contain some information that are conditionally true about the domain of interest. Suppose our knowledge base is as follows

```
SingASong(ananya).
listens_to_music(rohit).
listens_to_music(ananya) :-
    sing_a_song(ananya).
happy(ananya) :-
    sing_a_song(ananya).
happy(rohit) :-
    listens_to_music(rohit).
playes_guitar(rohit) :-
    listens_to_music(rohit).
```



Knowledge Base 3

The facts and rules of Knowledge Base 3 are as follows

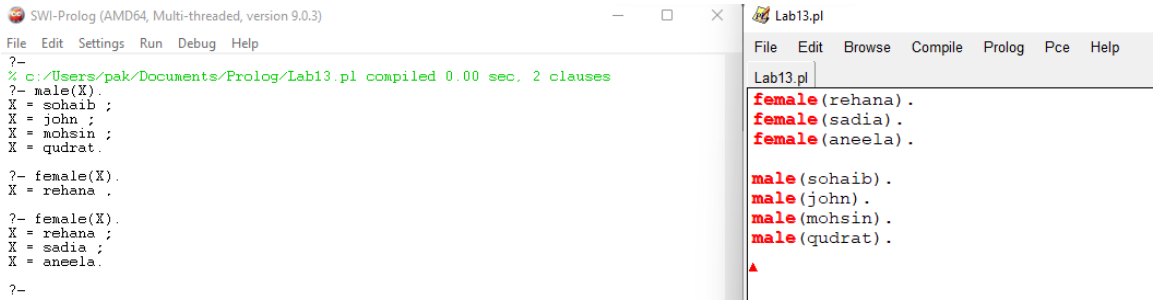


Activity 1:

Using variables and listing functors in prolog

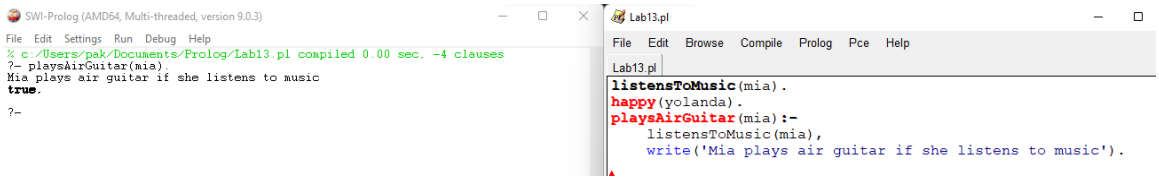
Solution:

A functor is simply the function name and the number of arguments of that functor is called arity. For instance, woman(mia) is a complex term with functor woman and arity 1, while loves(vincent, mia) is a complex term with functor loves and arity 2.



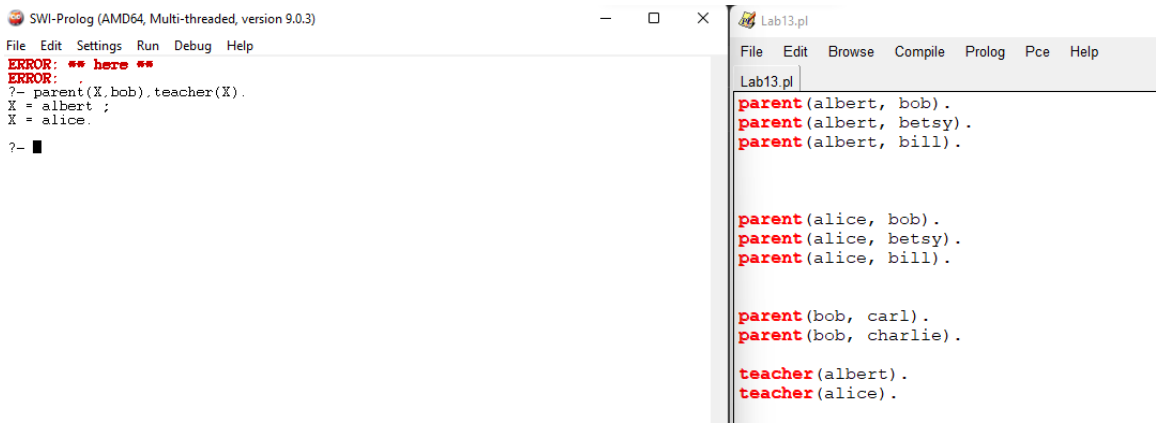
Activity 2:

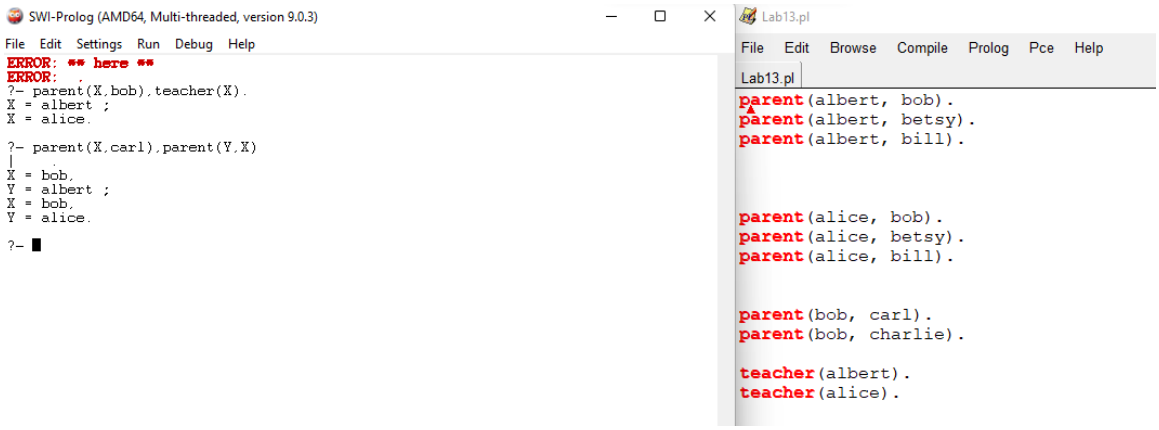
How to print text in prolog?



Activity 3:

How to ask complex queries?



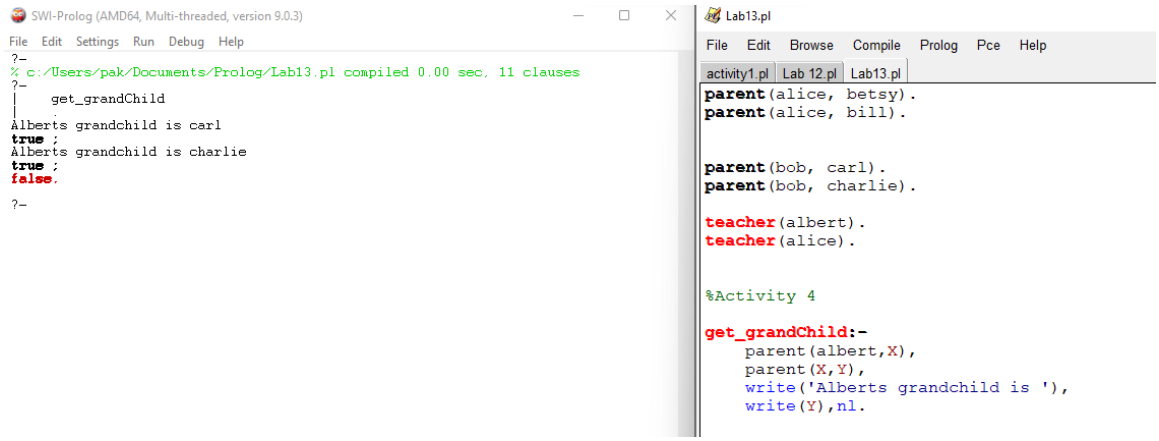



```
parent(albert, bob).
parent(albert, betsy).
parent(albert, bill).
```

```
parent(alice, bob).
parent(alice, betsy).
parent(alice, bill).


parent(bob, carl).
parent(bob, charlie).

teacher(albert).
teacher(alice).
```

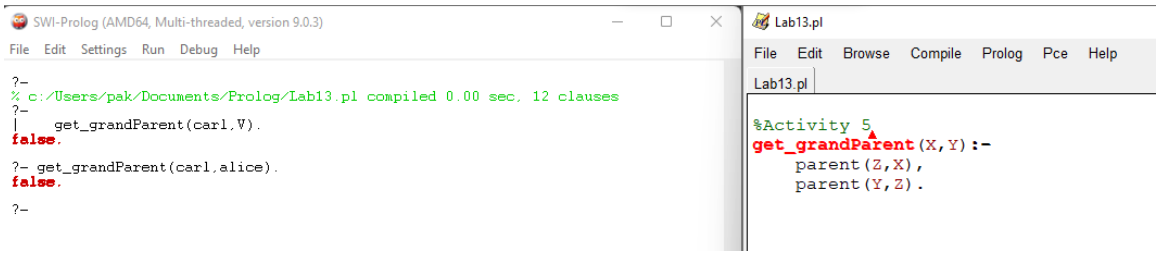
Activity 4:

Write a query in a similar fashion to determine grandchildren of Albert in above knowledgebase.



Activity 5:

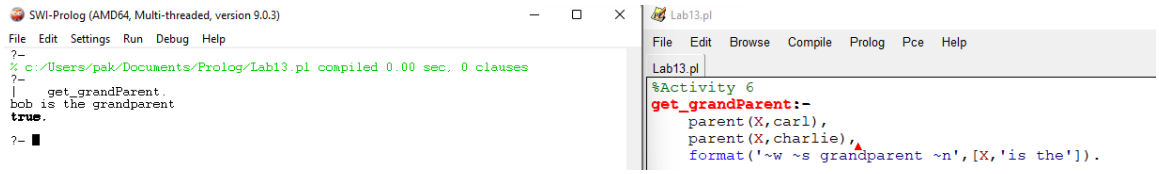
We can also define variables within a consequent of a predicate which helps us to find grandparent of any X.




```
get_grandParent(X,Y):-
    parent(Z,X),
    parent(Y,Z).
```

Activity 6:

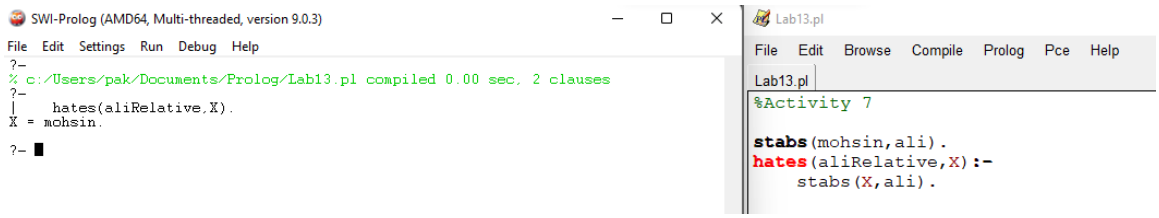
How to use format command to print inside a Knowledge Base?




```
getGrandParent:-
    parent(X,carl),
    parent(X,charlie),
    format('~w ~s grandparent ~n',[X,'is the']).
```

Activity 7:

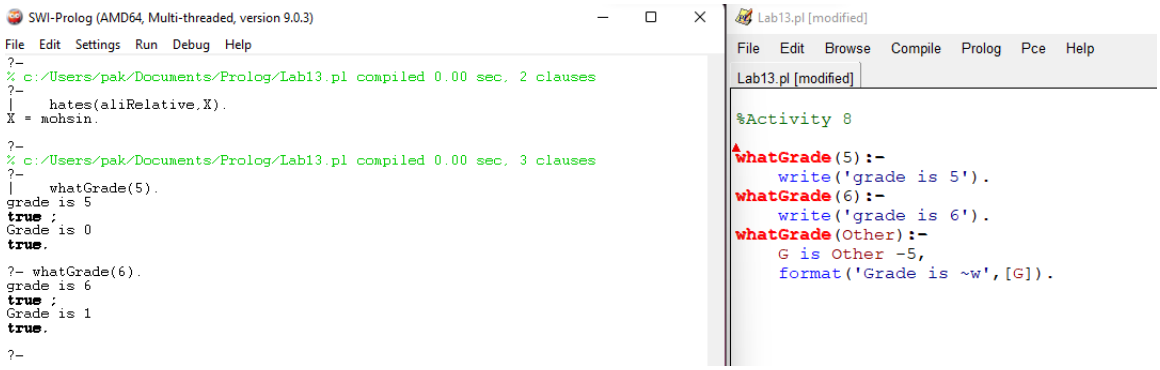
Deriving facts using variables?




```
stabs(mohsin,ali).
hates(aliRelative,X):-
    stabs(X,ali).
```

Activity 8:

Using a single predicate with constants and variables as arguments?




```
whatGrade(5):-
    write('grade is 5').
whatGrade(6):-
    write('grade is 6').
whatGrade(Other):-
    G is Other -5,
    format('Grade is ~w',[G]).
```

Lab Task 1:

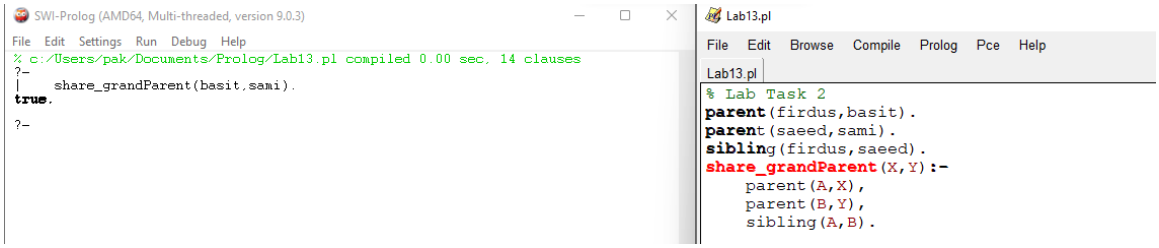
Consider the knowledgebase in activity 3. Define a brotherhood relationship and write a query that gives every uncle of a person X in the knowledgebase. An uncle is defined as brother of one's parent.

```
% Lab Task 1
parent(albert, bob).
parent(albert, betsy).
parent(albert, bill).

parent(alice, bob).
parent(alice, betsy).
parent(alice, bill).

parent(bob, carl).
parent(bob, charlie).


brother(X,Y):-
```

```
    male(X),
    male(Y),
    parent(Z,X),
    parent(Z,Y).


uncle_of(X,Y):-
    male(X),
    parent(P,Y),
    sibling(X,P).
```

Lab Task 2

Consider the knowledge base of activity 4. Write a query to find all pairs that share grand-parenthood relationship.




```
% Lab Task 2
parent(firdus,basit).
parent(saeed,sami).
sibling(firdus,saeed).
share_grandParent(X,Y):-
    parent(A,X),
    parent(B,Y),
    sibling(A,B).
```