

Twan Tran @029136612
Pornthep Bootchot @027817775
Gavin Lampton @026357147

Lab 5 – Prolog Programming

1. Answer:

- a. tell me which of the individuals you know about is a woman.

—?- woman(X).

X = mia ;

X = jody ;

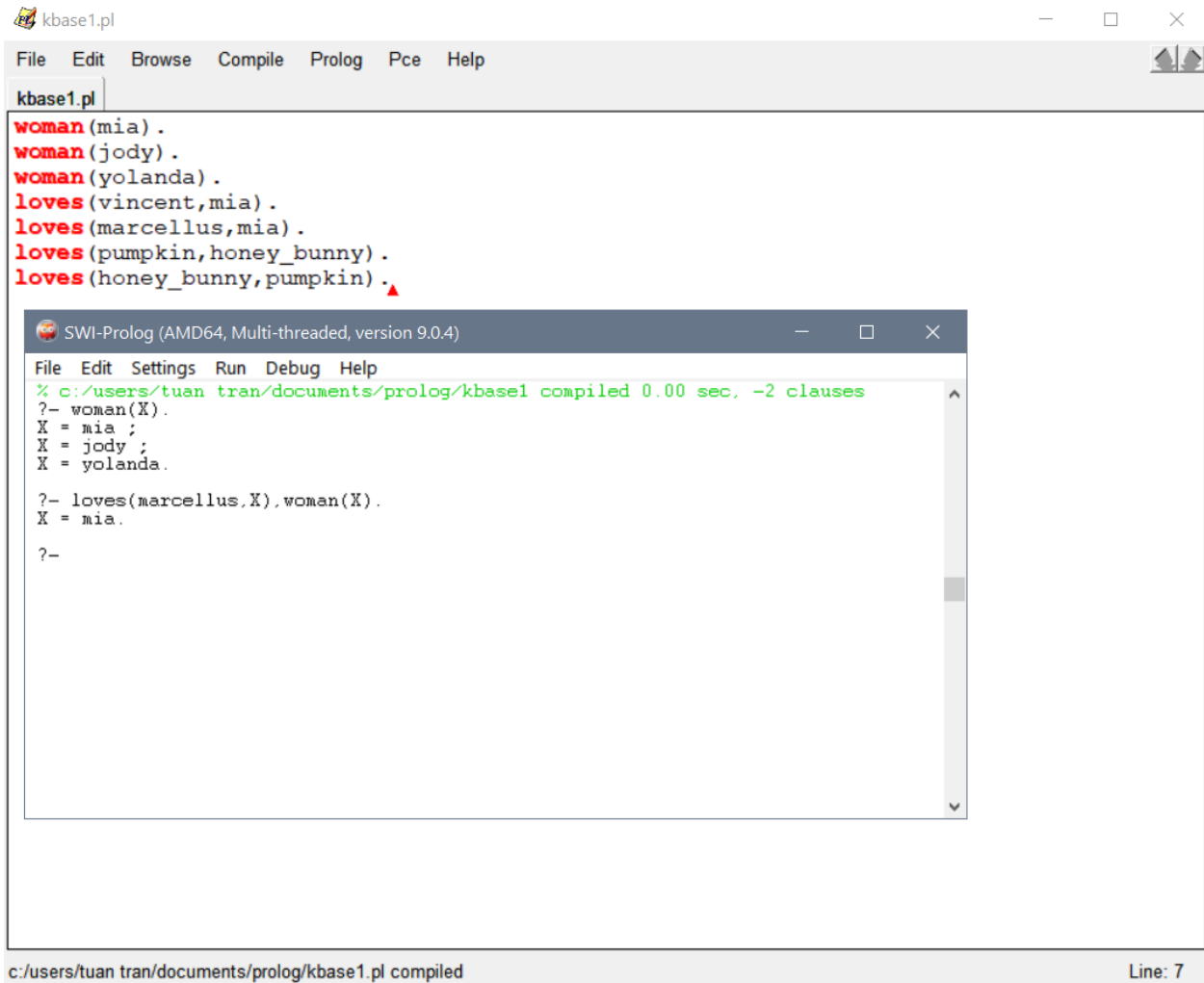
X = Yolanda .

- b. Is there any individual X such that Marcellus loves X and X is a woman?

?- loves(marcellus, X), woman(X).

X = mia.

Output:



```
kbbase1.pl
File Edit Browse Compile Prolog Pce Help
kbbase1.pl
woman(mia) .
woman(jody) .
woman(yolanda) .
loves(vincent,mia) .
loves(marcellus,mia) .
loves(pumpkin,honey_bunny) .
loves(honey_bunny,pumpkin) .

SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)
File Edit Settings Run Debug Help
% c:/users/tuan tran/documents/prolog/kbase1 compiled 0.00 sec, -2 clauses
?- woman(X).
X = mia ;
X = jody ;
X = yolanda.

?- loves(marcellus,X),woman(X).
X = mia.

?-
```

c:/users/tuan tran/documents/prolog/kbase1.pl compiled Line: 7

2. Answer:

- a. Create a rule:

It says that an individual X will be jealous of an individual Y if there is some individual Z that X loves, and Y loves that same individual Z too.

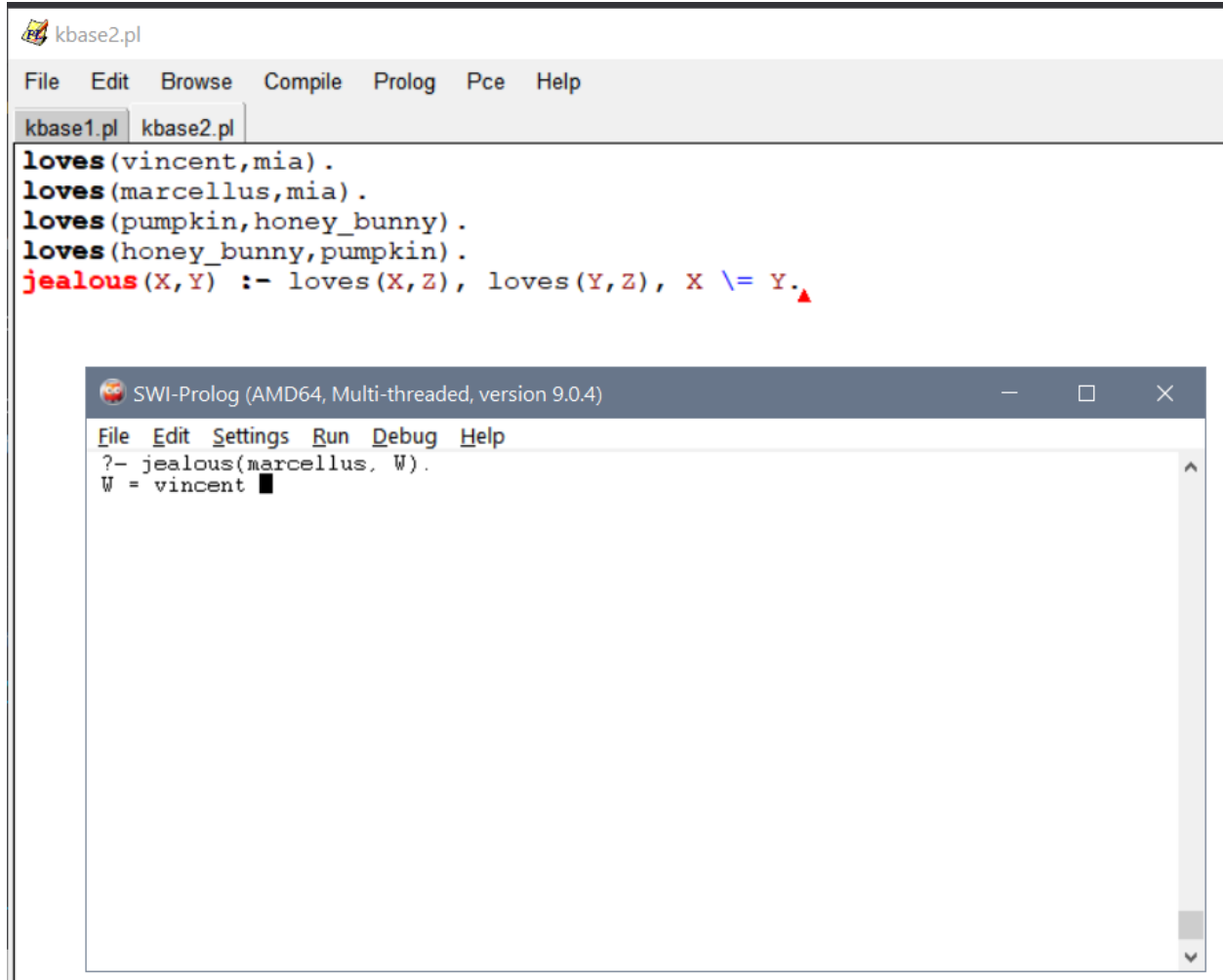
Rule: `jealous(X,Y) :- loves(X,Z), loves(Y,Z), X \= Y.`

- b. Create the following query:

Can you find an individual W such that Marcellus is jealous of W?

`?- jealous(marcellus, W).`

Output:



The screenshot shows a Prolog environment with a menu bar (File, Edit, Browse, Compile, Prolog, Pce, Help) and a tab for `kbase2.pl`. The code in the editor is:

```
loves(vincent,mia).
loves(marcellus,mia).
loves(pumpkin,honey_bunny).
loves(honey_bunny,pumpkin).
jealous(X,Y) :- loves(X,Z), loves(Y,Z), X \= Y.
```

Below the editor is a window titled "SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)" with a menu bar (File, Edit, Settings, Run, Debug, Help). The query `?- jealous(marcellus, W).` has been executed, and the result `W = vincent` is displayed.

3. Answer:

`zeros([], 0).`

`zeros([0 | T], Z) :- zeros(T, Z1), Z is Z1 + 1, !.`

`zeros([_ | T], Z) :- zeros(T, Z).`

Query:

`?- zeros([1,0,0,5],X).`

`?- zeros([],X).`

`?- zeros([0,1,2,3],X).`

Output:

The screenshot shows a Prolog IDE window titled 'kbase3.pl'. The menu bar includes 'File', 'Edit', 'Browse', 'Compile', 'Prolog', 'Pce', and 'Help'. Below the menu bar are three tabs: 'kbase1.pl', 'kbase2.pl', and 'kbase3.pl'. The main editor area contains the following Prolog code:

```
zeros([], 0).
zeros([0 | T], Z) :- zeros(T, Z1), Z is Z1 + 1, !.
zeros([_ | T], Z) :- zeros(T, Z).
```

Below the editor is a console window titled 'SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)'. It has a menu bar with 'File', 'Edit', 'Settings', 'Run', 'Debug', and 'Help'. The console displays the results of the queries:

```
?- zeros([1,0,0,5],X).
X = 2.

?- zeros([],X).
X = 0.

?- zeros([0,1,2,3],X).
X = 1.

?-
```

4. Answer:

```
intersect([], _, []).
```

```
intersect([X|R], Y, [X|Z]) :- member(X, Y), !, intersect(R, Y, Z).
```

```
intersect(_|R, Y, Z) :- intersect(R, Y, Z).
```

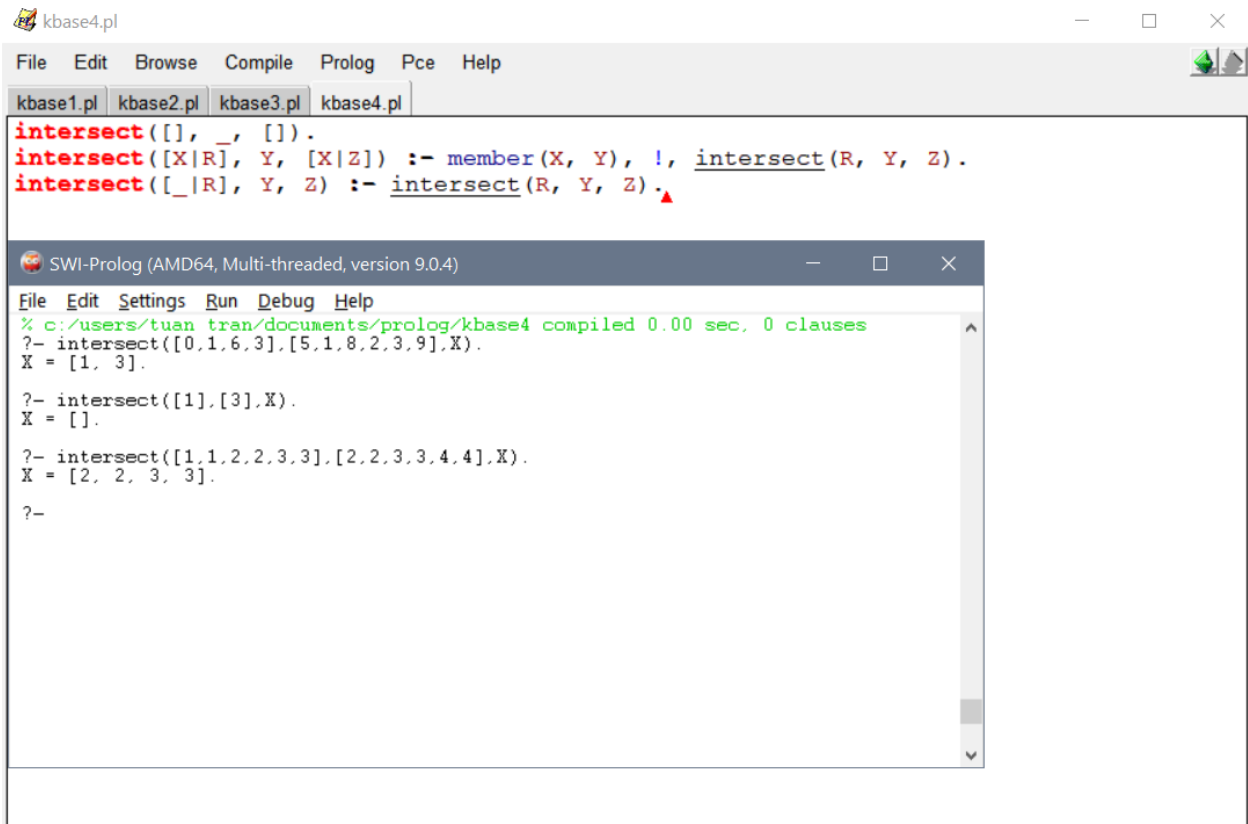
Query:

```
?- intersect([0,1,6,3],[5,1,8,2,3,9],X).
```

```
?- intersect([1],[3],X).
```

```
?- intersect([1,1,2,2,3,3],[2,2,3,3,4,4],X).
```

Output:



The screenshot shows a Prolog IDE window titled 'kbase4.pl'. The menu bar includes File, Edit, Browse, Compile, Prolog, Pce, and Help. The editor displays the following Prolog code:

```
intersect([], _, []).
intersect([X|R], Y, [X|Z]) :- member(X, Y), !, intersect(R, Y, Z).
intersect(_|R, Y, Z) :- intersect(R, Y, Z).
```

Below the editor is a SWI-Prolog console window (version 9.0.4) showing the execution of the queries:

```
% c:/users/tuan tran/documents/prolog/kbase4 compiled 0.00 sec, 0 clauses
?- intersect([0,1,6,3],[5,1,8,2,3,9],X).
X = [1, 3].

?- intersect([1],[3],X).
X = [].

?- intersect([1,1,2,2,3,3],[2,2,3,3,4,4],X).
X = [2, 2, 3, 3].

?-
```