CECS 342 - Lab Assignment 6 - Prolog Programming

Due Date: Sunday, May 5

Team Members: Bryan Tineo & Maxwell Guillermo

Completion of Lab Assignment:

Both team members contributed equally and collaborated throughout the completion of the lab assignment.

Code:

Question 1 Code:

1. [5 points] Create the following knowledge base. Name the program kbase1.

```
woman(mia).
woman(jody).
woman(yolanda).

loves(vincent,mia).
loves(marcellus,mia).
loves(pumpkin,honey_bunny).
loves(honey_bunny,pumpkin).

Create the following queries:
   a. tell me which of the individuals you know about is a woman.
   b. Is there any individual X such that Marcellus
       loves X and X is a woman?
```

```
% Define the knowledge base with facts about women and relationships of love
woman(mia).
woman(jody).
woman(yolanda).
loves(vincent, mia).
loves(marcellus, mia).
loves(pumpkin, honey_bunny).
loves(honey_bunny, pumpkin).
```

Question 2 Code:

2. [5 points] Create the following knowledge base. Name the program kbase2.

```
loves(vincent,mia).
loves(marcellus,mia).
loves(pumpkin,honey_bunny).
loves(honey_bunny,pumpkin).
```

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.

b. Create the following query:

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

```
% Define the knowledge base with facts about women and relationships of love
loves(vincent,mia).
loves(marcellus,mia).
loves(pumpkin,honey_bunny).
loves(honey_bunny,pumpkin).
% Create a rule for 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.
jealous(X, Y) :- loves(X, Z), loves(Y, Z), X \= Y.
```

Question 3 Code:

3.[5 points] Write a Prolog relation that accepts a list of integers, and counts the number of zeros in the list, e.g.,

```
% Base case: An empty list has zero count of zeros.
zeros([], 0).
% Recursive case 1: The head of the list is 0.
% Increment the count and recurse on the tail.
zeros([0 | T], Z) :-
  zeros(T, Z1),
  Z is Z1 + 1.
% Recursive case 2: The head of the list is not 0.
% Do not increment the count and recurse on the tail.
zeros([H | T], Z) :-
  H \= 0,  % Make sure the head is not 0
  zeros(T, Z).
```

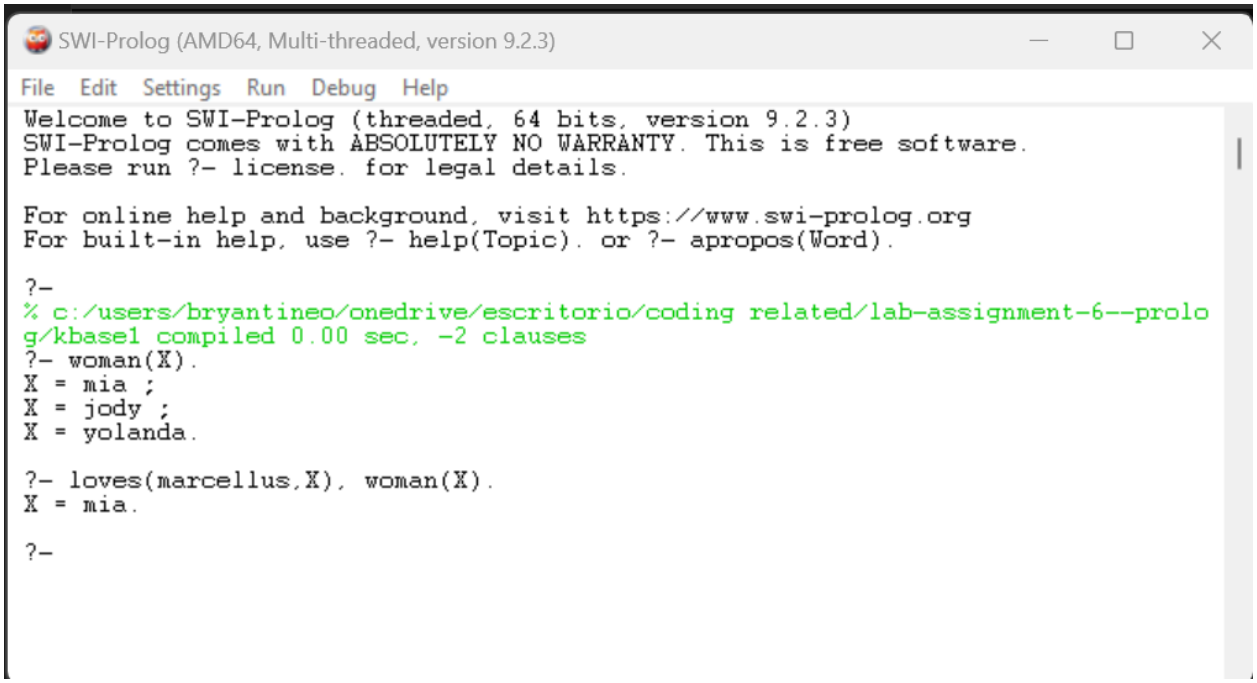
Question 4 Code:

4.[5 points] Write a Prolog relation "intersect(L1, L2, R)" that succeeds if R is the intersection of L1 and L2. (*Assume no duplicates*), e.g.,

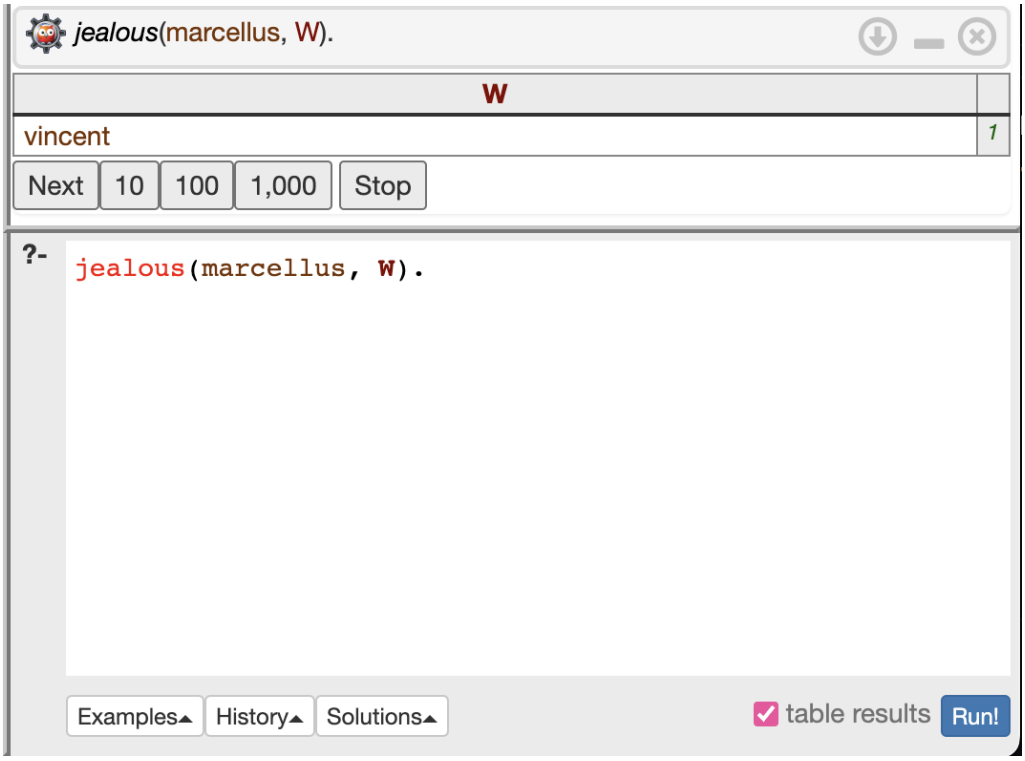
```
% Base case: If the first list is empty, the intersection is empty.
intersect([], _, []).
% If the head of the first list is a member of the second list,
% include it in the result list and continue with the tail of the first list.
intersect([X|R1], L2, [X|R3]) :- member(X, L2), !, intersect(R1, L2, R3).
% If the head of the first list is not a member of the second list,
% continue with the tail of the first list.
intersect([_|R1], L2, R3) :- intersect(R1, L2, R3).
```

Output:

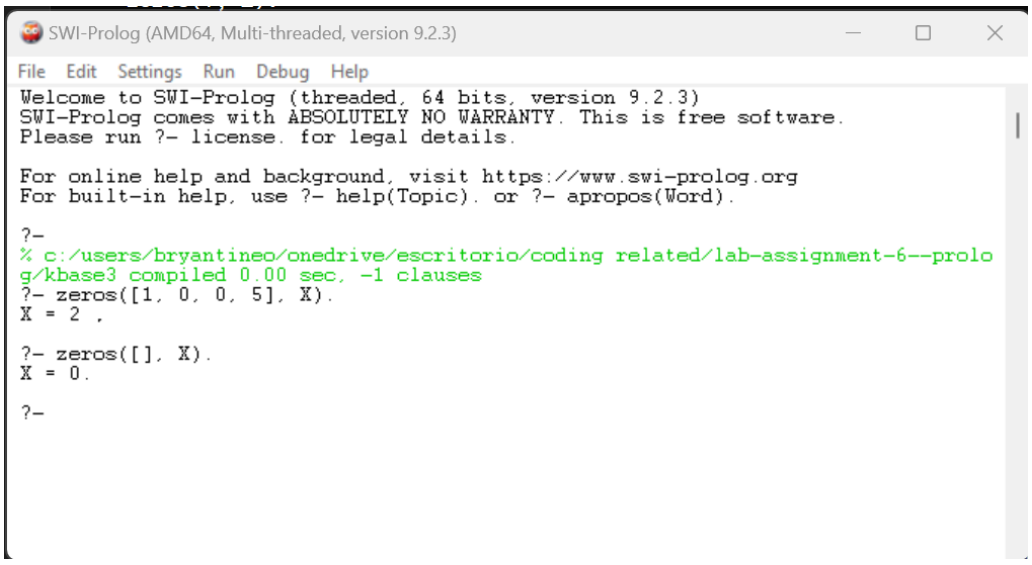
Question 1 Output:



Question 2 Output:



Question 3 Output:



Question 4 Output: