Logic Programming – CS307

Dr. Ahmed Sallam

Final Exam – 50 Points

1st June 2016 – 120 Mins.

This exam in 2 pages, make sure to answer all the questions

1. What does Prolog return for the following queries? [10 Points]

- **a.** `?- [1,2,3]=[A,C].` false
- **b.** `?- X = 3+3.` X=3+3
- **c.** `?- [_|_] = [a,b].` True
- **d.** `?- write('2+3').` 2+3
- **e.** `?- (a+b) =.. L.` L=[+,a,b]
- **f.** `?- 1>5 , !.` False
- **g.** `?- atomic(7).` True
- **h.** `?- [_|Z] = [a, b, c].`Z=[b,c]
- **i.** `?- 0<3 ; fail.` True
- **j.** `?- functor([2,4], F, N).`F='.'N=2

2. Read the following program then answer the questions [10 Points]

```
1.  anonymous(X,0,X):- !.
2.  anonymous(X,Y,D):- R is X mod Y, anonymous(Y,R,D).
```

- **a. What does prolog return for the query: ?-** `anonymous(30,6, D).` D=6
- **b. What does prolog return for the query: ?-** `anonymous(15,4, D).` D=1
- **c. If we removed (!), is there any difference ? Explain.** Yes, because the program will backtrack to line 2 and divide by zero (error)
- **d. If we rewrite line 2 as following :**

  `anonymous(X,Y,D):- anonymous(Y,R,D), R is X mod Y.`

  What does prolog return for the query: ?- `anonymous(30,6, D).` error, R is not evaluated yet
- **e. Suggest a better name instead of "**`anonymous`**".** GCD (Greater Common Divisor)

3. Use the following Prolog program to answer the questions [10 Points]

```
my_reverse([],[]).
my_reverse([H|T],L):- my_reverse(T,R),append(R,[H],L).
append([],L,L).
append([H|X],Y,[H|L]):-append(X,Y,L).
```

- **a. The above program uses two predicates (** `my_reverse`, `append` **) to reverse a given list. Rewrite the above program to do the same job with only one predicate (** `my_reverse` **).**

  my_reverse([],Acc,Acc).

  my_reverse([X|T],Acc,Rev):-my_reverse(T,[X|Acc],Rev).
- **b. Rewrite the above program such that** `my_reverse` **has no arguments at all, and the user should provide the required parameters at run time.**

  Add the following line before the above program

  my_reverse:-write('Enter List:'),nl,read(X), my_reverse(X,Y),nl,write(Y).

4. Write a Prolog program [7 Points]

Write a Prolog program to detect whether a given number is Perfect or not. A perfect number is a positive integer that is equal to the sum of its divisors; however, the number itself is not included in the sum. For example: a perfect number 6=1+2+3 **while a non-perfect number** 5≠1+5**.**

```
perfect(X):-perfect(X,1,0).
perfect(X,X,X):-!.
 perfect(X,Y,S):-Y<X, Z is X mod Y , Z=0, NewS is S+Y,NewY is Y+1, perfect(X,NewY,NewS).
 perfect(X,Y,S):-Y<X, Z is X mod Y , Z\=0, NewY is Y+1, perfect(X,NewY,S).
```

5. Write a Prolog program [7 Points]

Write a Prolog program to detect whether a given list is a set or not. A list represents a set if no element repeats, a simple run of the program looks as following:

```
?- set([a,b,c,c,d]).
false
?- set([a,b,c]).
true
?- set([a,[b,a],c]).
true
?- set([]).
True
```

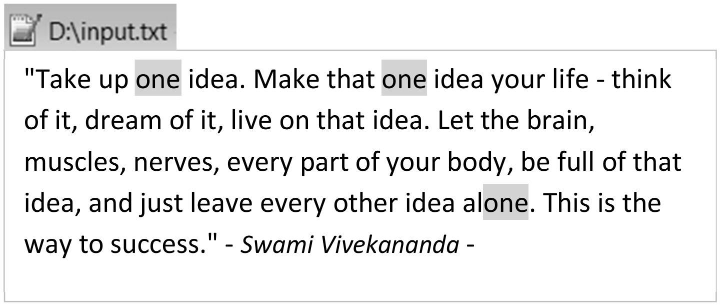
```
set([]).
set([H|T]):- \+member(H,T),set(T).
```

6. Write a Prolog program [6 Points]

Write the predicate `find(File, Atom)` **to return true if** `Atom` **was found in stored** `File`**. Suppose that lowercase letters are only used, a simple run of the program looks as following:**

```
?- find('d:\\input.txt','one').
true
```

Note: `'d:\\input.txt'` *is shown here* →



"Take up one idea. Make that one idea your life - think of it, dream of it, live on that idea. Let the brain, muscles, nerves, every part of your body, be full of that idea, and just leave every other idea alone. This is the way to success." - *Swami Vivekananda* -

```
find(F,Atom):- open(F,read,X),
          current_input(Stream),
          set_input(X),
          atom_codes(Atom,L),
          read_file(L,L),
          close(X),
          set_input(Stream).


read_file([],_).
read_file([H|L],X):-get_char(C),C\='end_of_file',
            atom_codes(C,CL),comp(H,CL),
            read_file(L,X).
read_file([H|_],X):-get_char(C),C\='end_of_file',
            atom_codes(C,CL),\+comp(H,CL),
            read_file(X,X).
comp(H,[H|_]).
```