T.V.Tharusha Induwara Vithanage

CST/19/029 – Intelligent Systems – Prolog Individual Assignment

Computer Science & Technology

Department Of Computer Science & Informatics

Faculty Of Applied Siences

Uva Wellassa University Of Sri Lanka

**Prolog Code**

% - T.V.Tharusha Induwara Vithanage - Individual Assignment -

:- use\_module(library(jpl)).

start :-sleep(0.4),

write('-----------------------------------------------------------------'),nl,

sleep(0.4),

write('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*'),nl,

sleep(0.2),

write("###################||| WEATHER PREDICTION EXPERT SYSTEM |||#########################"),nl,

sleep(0.4),

write('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*'),nl,

sleep(0.4),

write('-----------------------------------------------------------------'),nl,nl,nl,

interface2.

symptom(Weather,windspeed) :- verify(Weather," Is it a windspeed (y/n) ?").

symptom(Weather, humidy) :- verify(Weather," Is it a humidy (y/n) ?").

symptom(Weather,atmosphericpressure) :- verify(Weather," Is it a atmosphericpressure (y/n) ?").

symptom(Weather, temperature) :- verify(Weather," Is it a temperature (y/n) ?").

symptom(Weather,timerangeofpressure) :- verify(Weather," Is it timerangeofpressure (y/n) ?").

symptom(Weather, cloudyness) :- verify(Weather," Is it cloudyness (y/n) ?").

hypothesis(Weather,light\_rainy\_day) :-

symptom(Weather,humidy),

symptom(Weather,windspeed),

symptom(Weather,timerangeofpressure).

hypothesis(Weather,heavy\_rainy\_day) :-

symptom(Weather,windspeed),

symptom(Weather,humidy),

symptom(Weather,cloudyness),

symptom(Weather,atmosphericpressure).

hypothesis(Weather,lightthunderwith\_heavyrainy\_day) :-

symptom(Weather,windspeed),

symptom(Weather,humidy),

symptom(Weather,atmosphericpressure),

symptom(Weather,timerangeofpressure).

hypothesis(Weather,sunny\_day) :-

symptom(Weather,temperature),

symptom(Weather,cloudyness),

symptom(Weather,windspeed).

hypothesis(Weather,cloudy\_day) :-

symptom(Weather,temperature),

symptom(Weather,windspeed),

symptom(Weather,humidy).

hypothesis(\_,"Misunderstoodable. But I'm Sorry, We can't tell weather predictions under the features.").

response(Reply) :-

read(Reply),

write(Reply),nl.

ask(Weather,Question) :-

write(Weather),write(', do you'),write(Question),

/\*read(N),

( (N == yes ; N == y)

->

assert(yes(Question)) ;

assert(no(Question)), fail),\*/

interface(', do you',Weather,Question),

write('Loading.'),nl,

sleep(1),

write('Loading..'),nl,

sleep(1),

write('Loading...'),nl,

sleep(1),

nl.

:- dynamic yes/1,no/1.

verify(P,S) :-

(yes(S)

->

true ;

(no(S)

->

fail ;

ask(P,S))).

undo :- retract(yes(\_)),fail.

undo :- retract(no(\_)),fail.

undo.

pt(Weather):-

hypothesis(Weather,Feature),

interface3(Weather,', you probably have ',Feature,'.'),

write(Weather),write(', you probably have '),write(Feature),write('.'),undo,end.

end :-

nl,nl,nl,

sleep(0.7),

write('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*'),nl,

sleep(0.4),

write("################||| THANK YOU FOR GETTING FROM OUR SERVICE!!! |||#####################"),nl,

sleep(0.4),

write('\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*'),nl.

interface(X,Y,Z) :-

atom\_concat(Y,X, FAtom),

atom\_concat(FAtom,Z,FinalAtom),

jpl\_new('javax.swing.JFrame', ['Expert System'], F),

jpl\_new('javax.swing.JLabel',['--- WEATHER PREDICTION EXPERT SYSTEM ---'],LBL),

jpl\_new('javax.swing.JPanel',[],Pan),

jpl\_call(Pan,add,[LBL],\_),

jpl\_call(F,add,[Pan],\_),

jpl\_call(F, setLocation, [400,300], \_),

jpl\_call(F, setSize, [400,300], \_),

jpl\_call(F, setVisible, [@(true)], \_),

jpl\_call(F, toFront, [], \_),

jpl\_call('javax.swing.JOptionPane', showInputDialog, [F,FinalAtom], N),

jpl\_call(F, dispose, [], \_),

write(N),nl,

( (N == yes ; N == y)

->

assert(yes(Z)) ;

assert(no(Z)), fail).

interface2 :-

jpl\_new('javax.swing.JFrame', ['Expert System'], F),

jpl\_new('javax.swing.JLabel',['--- WEATHER PREDICTION EXPERT SYSTEM ---'],LBL),

jpl\_new('javax.swing.JPanel',[],Pan),

jpl\_call(Pan,add,[LBL],\_),

jpl\_call(F,add,[Pan],\_),

jpl\_call(F, setLocation, [400,300], \_),

jpl\_call(F, setSize, [400,300], \_),

jpl\_call(F, setVisible, [@(true)], \_),

jpl\_call(F, toFront, [], \_),

jpl\_call('javax.swing.JOptionPane', showInputDialog, [F,'Hi.We are Weather Prediction Expert System.Can you tell me your name please'], N),

jpl\_call(F, dispose, [], \_),

/\*write(N),nl,\*/

( N == @(null)

-> write('you cancelled'),interface3('you cancelled. ','Thank you ','for getting ','from our service.'),end,fail

; write("Hi.We are Weather Prediction Expert System.Can you tell me your name please : "),write(N),nl,pt(N)

).

interface3(P,W1,D,W2) :-

atom\_concat(P,W1, A),

atom\_concat(A,D,B),

atom\_concat(B,W2,W3),

jpl\_new('javax.swing.JFrame', ['Expert System'], F),

jpl\_new('javax.swing.JLabel',['--- WEATHER PREDICTION EXPERT SYSTEM ---'],LBL),

jpl\_new('javax.swing.JPanel',[],Pan),

jpl\_call(Pan,add,[LBL],\_),

jpl\_call(F,add,[Pan],\_),

jpl\_call(F, setLocation, [400,300], \_),

jpl\_call(F, setSize, [400,300], \_),

jpl\_call(F, setVisible, [@(true)], \_),

jpl\_call(F, toFront, [], \_),

jpl\_call('javax.swing.JOptionPane', showMessageDialog, [F,W3], N),

jpl\_call(F, dispose, [], \_),

/\*write(N),nl,\*/

( N == @(void)

-> write('')

; write("")

).

help :- write("To start the expert system please type 'start.' and press Enter key").









