New Virus expert system in Prolog
Muhammad Ali

This report details the development of a simple expert system to detect the presence of a new virus which is transmitted through proximity to infected person. The code will be analysed and explained and overall performance discussed in detail. Furthermore, the usability aspect of expert system along with possible improvements will also be discussed to provide a comprehensive critique.

The Assignment instructions consist of a diverse range of symptoms (common, less common and serious), the incubation period is defined as the time between exposure to the virus and onset of symptoms, people with different range and severity of symptoms are advised either to stay at home or seek medical advice (call not in-person contact), both people over 70 and those with pre-existing health conditions are at higher risk and males are more susceptible to develop severe symptoms as comparted to females.

(Brammer, 2005) states that a simple Prolog program consists of facts and rules; facts are predicates with a head only, void of a body and variables whilst rules have both a head and a body. The predicate *consultdoctor* (figure 1) has 4 arguments which represent serious symptoms of the new virus.

```
% Defining the consultdoctor predicate and its arguments which represent serious symptoms.

consultdoctor(chestpain).
consultdoctor(chestpressure).
consultdoctor(loss_of_speech_or_movement).
consultdoctor(breathingproblem).

% Common symptoms predicate and its arguments.

infectionpresent(fever).
infectionpresent(dry_cough).
infectionpresent(tiredness).


% Less common symptoms predicated and its arguments.

infectionpresent(conjunctivitis).
infectionpresent(sorethroat).
infectionpresent(diarrhoea).
infectionpresent(pains).
infectionpresent(headache).
infectionpresent(anosmia).
infectionpresent(runningnose).


% High chance of severe infection predicates and its arguments.

seriousexistingcondition(diabetes).
seriousexistingcondition(hypertension).
seriousexistingcondition(cardiovasculardisease).
seriousexistingcondition(chronicrespiratorydisease).
seriousexistingcondition(cancer).
```

Figure 1: Facts of the new virus expert system.

Figure 2 shows the rules of the new expert system with the input variables defined in the head and predicates in the body.

```
message_for_age_gdr_prec(Age, Gender, Precondition):-
    Age >=70, write('High chance of severe infection due to old age'), nl;
    Gender = 'male', write('High chance of severe infection as males are more vunerable.'), nl;
    seriousexistingcondition(Precondition), write('High chance of severe infection.'), nl.

message_for_infection(IncubationPeriod, Symptom):-
    IncubationPeriod > 14, write('Low chances of Virus'), nl;
    infectionpresent(Symptom), not(consultdoctor(Symptom)), write('This is a symptom of new virus. Stay at home and monitor your symptoms.'), nl;
    consultdoctor(Symptom), write('See a medical professional.'), nl.
```

Figure 2: Rules of the new expert system.

Figure 3 shows the dynamic inputs of the system collected when user runs the function.

```
:-dynamic(p_data/2).

age_input(Y):-
    (   p_data(age,Y),! );
    (   write("What is the age of the patient? (0 <= x <= 100)"),nl,
        read(Y),nl,
        assert(p_data(age,Y))
    ).

gender_input(X):-
    (   p_data(gender,X),! );
    (   write("What is the gender of the patient? (male/female)"),nl,
        read(X),nl,
        assert(p_data(gender,X))
    ).

incubationperiod_input(D):-
    (   p_data(incubationperiod,D),! );
    (   write("What is the incubation period (time between exposure to virus and onset of symptoms)? (0 <= x <= 100)"),nl,
        read(D),nl,
        assert(p_data(incubationperiod,D))
    ).

symptoms_input(Z):-
    (   p_data(symptoms,Z),! );
    (   write("Please input your symptom."),nl,
        read(Z),nl,
        assert(p_data(symptoms,Z))
    ).

preexistingcondition_input(C):-
    (   p_data(preexistingcondition,C),! );
    (   write("Please input any of your pre-exisitng health condition. (Write no if none)"),nl,
        read(C),nl,
        assert(p_data(preexistingcondition,C))
    ).
```

Figure 3: Dynamic inputs of the new expert system.

Figure 4 shows the interface of the expert system once the program file [aliprolog.pl] is run at https://swish.swi-prolog.org with the call function command **checkAll(M).** User are asked the following questions: “What is the gender of the patient?”, “What is the age of the patient?”, “What is the incubation period?”, “Please input your symptom” and “Please input any of your pre-existing health condition”.

Usability is improved by keeping the interface simple and input questions (facts from figure 1) both concise and well defined. However, the advice given, *stay at home* or *see a medical professional*, carries the risk of generalising the varied health problems of the users. An extension of this project can look at adding more health conditions and segmenting the age by introducing more brackets instead of a generic over and under 70 years rule. Providing more specific advice can allow users to self-evaluate and seek timely medical advice.

The expert system serves the purpose of being an initial diagnostic tool consisting of just 5 easy questions for people who are oblivious of the new virus and run the risk of discarding symptoms. Asymptomatic carriers of the virus are not modelled due to both lack of information and understanding of modes of transmission

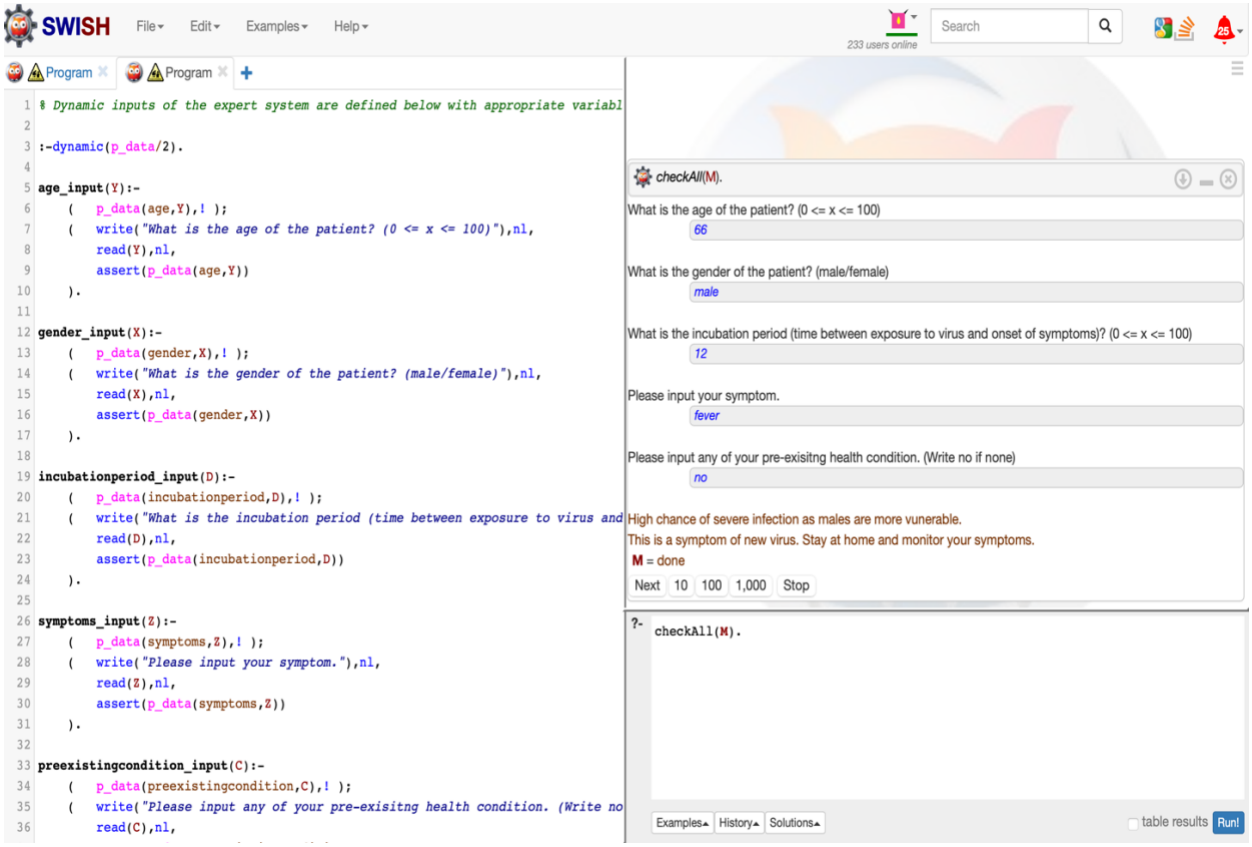

Figure 4: Interface of the expert system.

(Singla, 2013) proposes a lung disease detection expert system which follows similar logic to this report: starting with defining simple facts and input variables followed by creating rules to manipulate and present diagnosis. (Singla, 2013) agrees with the author of this report in acknowledging the incompletes of the knowledge base (facts) and existence of multiple and critical health conditions not covered in these systems.

References:

Brammer, M., 2005. *Logic Programming with Prolog.* Portmouth: Springer.

Singla, J., 2013. The Diagnosis of Some Lung Diseases in a Prolog Expert System. *International Journal of Computer Applications,* 78(15), p. 0975 – 8887.