AIML Experiment - 7

Name : Aniket Narbariya
UID : 2019130043
Branch : TE COMPS

Aim : To design and implement an expert system, incorporating the match algorithm and the rule language.
1. It should provide a fact base updating function.
2. It should provide a function that checks the rules' LHS and returns which rules were matched.
3. It should support firing RHS according to matches.

Theory :
An expert system is a computer program that is designed to solve complex problems and to provide decision making ability like a human expert. It performs this by extracting knowledge from its knowledge base using the reasoning and inference rules according to the user queries.

The expert system is a part of AI, and the first ES was developed in the year 1970, which was the first successful approach of artificial intelligence. It solves the most complex issue as an expert by extracting the knowledge stored in its knowledge base. The system helps in decision making for complex problems using both facts and heuristics like a human expert. It is called so because it contains the expert knowledge of a specific domain and can solve any complex problem of that particular domain. These systems are designed for a specific domain, such as medicine, science, etc.

The performance of an expert system is based on the expert's knowledge stored in its knowledge base. The more knowledge stored in the KB, the more that system improves its performance. One of the common examples of an ES is a suggestion of spelling errors while typing in the Google search box.

Below are some popular examples of the Expert System:
o DENDRAL: It was an artificial intelligence project that was made as a chemical analysis expert system. It was used in organic chemistry to detect unknown organic molecules with the help of their mass spectra and knowledge base of chemistry.

o MYCIN: It was one of the earliest backward chaining expert systems that was designed to find the bacteria causing infections like bacteraemia and meningitis. It was also used for the recommendation of antibiotics and the diagnosis of blood clotting diseases.

o PXDES: It is an expert system that is used to determine the type and level of lung cancer. To determine the disease, it takes a picture from the upper body, which looks like the shadow. This shadow identifies the type and degree of harm. o CaDeT: The CaDet expert system is a diagnostic support system that can detect cancer at early stages.

Characteristics of Expert System

o High Performance: The expert system provides high performance for solving any type of complex problem of a specific domain with high efficiency and accuracy.

o Understandable: It responds in a way that can be easily understandable by the user. It can take input in human language and provide the output in the same way.

o Reliable: It is much reliable for generating an efficient and accurate output.

o Highly responsive: ES provides the result for any complex query within a very short period of time.

Problem Statement:

Read the below passage carefully and answer the questions:

Five cities all got more rain than usual this year. The five cities are: Last Stand, Mile City, New Town, Olliopolis, and Polberg. The cities are located in five different areas of the country: the mountains, the forest, the coast, the desert, and in a valley. The rainfall amounts were: 12 inches, 27 inches, 32 inches, 44 inches, and 65 inches.

* The city in the desert got the least rain; the city in the forest got the most rain.

* New Town is in the mountains.

* Last Stand got more rain than Olliopolis.

* Mile City got more rain than Polberg, but less rain than New Town.

* Olliopolis got 44 inches of rain.

* The city in the mountains got 32 inches of rain; the city on the coast got 27 inches of rain.

Code :

```
city(C) :-
length(C, 5),

% city names
member(h('Last Stand', _, _), C),
member(h('Mile City', _, _), C),
member(h('New Town', _, _), C),
member(h('Olliopolis', _, _), C),
member(h('Polberg', _, _), C),

% city areas
member(h(_, mountains, _), C),
member(h(_, forest, _), C),
member(h(_, coast, _), C),
member(h(_, desert, _), C),
member(h(_, valley, _), C),

% rainfall amounts
member(h(_, _, 12), C),
member(h(_, _, 27), C),
member(h(_, _, 32), C),
member(h(_, _, 44), C),
member(h(_, _, 65), C),

% Hints
% The city in the desert got the least rain;
```

```
% the city in the forest got the most rain.
member(h(_, desert, 12), C),
member(h(_, forest, 65), C),

% New Town is in the mountains.
member(h('New Town', mountains, _), C),

% Last Stand got more rain than Olliopolis.
member(h('Last Stand', _, A), C),
member(h('Olliopolis', _, B), C),
A > B,

% Mile City got more rain than Polberg, but less rain than New Town.
member(h('Mile City', _, D), C),
member(h('Polberg', _, E), C),
D > E,
member(h('New Town', _, F), C),
F > D,

% Olliopolis got 44 inches of rain.
member(h('Olliopolis', _, 44), C),

% The city in the mountains got 32 inches of rain; the
% city on the coast got 27 inches of rain.
member(h(_, mountains, 32), C),
member(h(_, coast, 27), C).

query_rain_amount(City_Name, Rainfall_Amount) :-
city(C),
member(h(City_Name, _, Rainfall_Amount), C),
write(City_Name), write(" received "),
write(Rainfall_Amount), write(" inches of rain."),
nl.


query_city_region(City_Name, Region) :-
city(C),
member(h(City_Name, Region, _), C),
write(City_Name), write(" is located in the "),
write(Region), nl.
```

Results:

Conclusion:

I learned to use prolog for solving logical problems.
Here,an expert system emulates the decision-making ability of the human expert.It helps us to get answer for tedious logical operations.