TERM Work 1

Problem statement : Study PROLOG standards and Syntaxes

THEORY

→ What is prolog?
Prolog is a programming language that is based on logic programming. This name prolog stands for programming in logic

→ History of prolog
Prolog was developed in 1970s by a group of researchers in marsille France and since then is has became a popular tool for implementing artificial intelligence applications and for teaching logic based programming

* Syntax
The system of prolog is based on a set of logical rules that differ relations between objects and it is designed to be very readable & easy to understand.
Basic elements of prolog
i] Atoms : These are identifiers that begin with a lower-case letters
Ex: 'hello' 'name' and 'a'

ii] Variables: These are the identifiers that begin with uppercase letter in prolog
Eg: 'X' 'Y' and 'animal'

iii) Predicates :- These are logical statements that express relationships between objects. They are represented as a series of classes that specify conditions.
Eg:- 'Parent(X,Y)' : 'Parent(John, mary)' specifies John is mary's parent.

Lists :- These are argument sequences of prolog terms enclosed in square brackets and separated by commas.
Ex:- '[1,2,3]' is a list containing 1, 2 & 3

Arithmetic Expressions : These involve arithmetic operations such as addition, substitution, multiplication & division
Ex:- '3+4' these elements evaluates to 7

Comments : These are used to add explantory text to prolog code. They bring with the Percent sign (%) and continue to end of the line

Building Block of prolog

(1) Facts : A fact is a basic unit of knowledge that expresses the relationship between objects. Facts are used to divisible discrete properties or attributes of objects using predicate.

Rules
Rule is an orth implicit relationship between objects so facts are conditionally true. So where one associates condition is true. the predicate is also true.

Queries
Queries are some questions on the relationship between objects and object properties

Knowledge Base
Knowledge base is a collection of facts of facts and rules
Ex: KB1
Fact
sing_a_song(anmal)
listens_to_music(sachi)

- Rules
listens_to_music(anmal) :- sing_a_song(sachi)
happy(anmal) :- sing_a_song(anmal)
happy(sachi) :- listens_to_music(sachi)
players guitar(sachi) :- listens_to_music(sachi)

Query
1?- happy(sachi).
yes
1?- sing_a_song(sachi).
no
1? - plays - guitar(sachi)
no

Eg 1: food(dosa)
food(idli)
food(pizza)
lunch(idli)
dinner(pizza)
meal(x) :- food(x)

?- food(pizza)
true
?- meal(x), lunch(x).
x = idli
?- dinner(pizza)
true
? meal(what).
what = dosa;
what = idli;
what = pizza;

Eg 2: Studies(kirti, ai)
Studies(rohit, ai)
Studies(geeta, ds)
Studies(john, nt)
Studies(sgs, ai)
teaches(jk, nt)
teaches(jk, rpa)
Professor(x, y) :- teaches(x, c), studies(y, c)

Queries

?- studies(kirti, what)
what = ai;
?- professor(sgs, students)
students = rohit;
students = kirti;
?- studies(who, ai)
who = kirti;
who = rohit;
?- teaches(who, which)
who = sgs
which = ai;
who = jk;
which = nt;
who = rjr
which = rpa;
who = spd;
which = ds;

Eg: f(1, one)
f(s(1), two)
f(s(s(1)), three)
f(s(s(s(X))), N) :- f(X, N)

Queries
?- f(s(1), A).
A = two
?- f(s(s(1)), two)
false
?- f(s(s(s(s(s(s(s(1))))))), C).
C = one
?- f(0, three)
D = s(s(1))

Eg 4: big(bear)
big(elephant).
small(cat).
brown(bear).
black(cat).
grey(elephant).
dark(Z) :- black(Z).
dark(Z) :- brown(Z)

Queries
?- dark(X), big(X).
X = bear
?- big(X), dark(X).
X = bear.

09/10

24/04/23

Conclusion