AI LAB-4

Objectives: To learn arithmetic operations and recursion in Prolog.

Description:

THE EQUAL OPERATOR:-

Five basic rule govern the equal sign in turbo prolog,

1. If one operand is a free variable and the other is an object ,The variable will become bound to the object value, and the goal will succeed . For example ,if X= 4 +3,X will be bound to 7.

2. If one operand is a bound variable and the other is an object, the goal will succeed only if the variable is already bound to the value of the object. For example ,if X is already bound to 7 and X = 5 +2 is assessed , the program will be display
   True
   Because 5 + 2 does equal the bound X variable.

3. If one operand is a free variable and the other is bound variable, the free variable will be bound to same value as the bound variable , and the goal will succeed. For example , If Y has been bound to 7 ,X=Y will bind X to 7.

4. If both variables are bound variables, the goal will succeed only if both are bound to the same value. For example, if X is bound to 7 and so is Y,Then X=Y will be true; otherwise ,it will be false.

5. If both operands are objects , the goal will succeed only if they are the same.For example apple = apple will succeed.

For example,

```
predicates
    go
clauses
    go :-
    X= sqrt (2),
    X= 1.414.
```

O/P:

```
Goal:go
No.
```

```
predicates
    go
clauses
    go :-
    X= 4 + 3,
    write(X), nl,
    X=4 – 3,
    write (x),nl.
```

O/P:

```
Goal:go
7
No.
```

COMPARISION OPERATORS :

Symbol	Meaning
<	Less than
>	Greater than
=	Equal to
<=	Less than or equal to
>=	Greater than or equal to
< >	Not equal to

For example,

```
predicate
go
clauses
go : -
X = 2,
X < = 4 +3,nl.
```

O/P :

```
Goal : go
       True
Goal :
```

Turbo prolog mathematical predicate function (prefix notation) :-

Predicate	Function
abs(X)	Absolute value of X
cos(X)	Cosine of X (radians)
sin(X)	Sine of X (radians)
tan(X)	tangent of X (radians)
arctan(X)	Arctangent of X
exp(X)	e raised to the power of X
ln(X)	Log of base e
log(X)	Log of base10
sqrt(X)	square root of X
random	Binds X to a random real number 0 <= X <1
round(X)	Round X

Turbo prolog logical predicate function (prefix notation) :-

Predicate	Function
Btand (X,Y.Z)	Bitwise AND (Z is result)
Bitor(X,Y,Z)	bitwise OR
Bitnot(X,Z) `	bitwise NOT
Bitxor(X,Y,Z)	Bitwise exclusive OR
Bitleft(X,N,Y)	Bitwise shift N bits left
Bitright(X,N,Y)	Bitwise shift N bits right

USING ARITHMETHIC PREDICATE :-

Be sure all conditions are tested when creating predicate for numerical processing. For example, recall that the plus predicate.

Plus (X,Y,Sum) :-
Sum = X +Y.
Only works if X& Y are bound and the sum is free.

Simple recursive program:

```
predicates
        count(integer)
clauses
        count(9).
        count(N):-
                clearwindow,
                write(" ",N),
                NN=N+1,
                count(NN),
                write("You made it here. Layer ",NN),nl.
```

O/P:

Goal: ??? Check, it out???

Exercises

1. Write a prolog program to find roots (real roots only) of quadratic equation.
2. Write a prolog program to implement a logon routine. This routine must asks username and password and verify with pair of username and password available (i.e. stored as clauses) as facts. On successful match system display "welcome message" and on an unsuccessful attempt user is allowed 3 times to reenter valid credentials. If user enters incorrect credential continuously 3 times then system exits with "unsuccessful attempt message".
3. Write a prolog program to find factorial of a given number.
4. Write a prolog program to find sum of first n number.
5. Write a prolog program to print n^{th} term of Fibonacci series.
6. Write a prolog program to print Fibonacci series up-to n^{th} term.