

TE Comps-A
Manasvi Patil – 9219

Experiment No: 6

Title: Prolog Programming Set 1

Objective: To get acquainted with logical programming

Implement

1. “Hello, World!” program

Code:

```
1 :- initialization(main).  
2 main :- write('Hello World!').
```

Output:

```
compiling /home/cg/root/64299359880f2/main.pg for byte code...  
/home/cg/root/64299359880f2/main.pg compiled, 1 lines read - 311 bytes written, 2 ms  
Hello World!| ?-
```

2. Program to check if an element is a member of a list

Code:

```
1 list_member(X, [X | _]) :- !.      % If the head of the list is X  
2  
3 list_member(X, [_ | Tail]) :-      % else recur for the rest of the list  
4     list_member(X, Tail).
```

Output:

```
compiling /home/cg/root/64299f82a908a/main.pg for byte code...  
/home/cg/root/64299f82a908a/main.pg compiled, 3 lines read - 478 bytes written, 3 ms  
| ?- list_member(c, [a, b, c, d]).  
yes  
| ?- list_member(f, [a, b, c, d]).  
list_member(f, [a, b, c, d]).  
  
no  
| ?-
```

3. Program to append two lists

Code:

```
1 % If L1 is empty, resultant list will be equal to L2 (base case)
2 append_list([], L2, L2).
3
4 append_list([X | L1], L2, [X | L3]) :-
5     append_list(L1, L2, L3).
```

Output:

```
compiling /home/cg/root/64299f82a908a/main.pg for byte code...
/home/cg/root/64299f82a908a/main.pg compiled, 4 lines read - 511 bytes written, 2 ms
| ?- append_list([1, b, c], [5, q], Ans).
append_list([1, b, c], [5, q], Ans).
Ans = [1,b,c,5,q]

yes
| ?- |
```

4. Program to reverse a list

Code:

```
1 reverse([], Y, R) :-
2     R = Y.
3 reverse([H|T], Y, R) :-
4     reverse(T, [H|Y], R).
5
```

Output:

```
compiling /home/cg/root/642a910f6812b/main.pg for byte code...
/home/cg/root/642a910f6812b/main.pg compiled, 3 lines read - 531 bytes written, 4 ms
| ?- reverse([10,8,4,3,2], R).
reverse([10,8,4,3,2], R).
R = [2,3,4,8,10]

yes
| ?- |
```

5. Program to find the length of a list

Code:

```
1 % length of empty list is 0 (base case)
2 list_length([], 0).
3 list_length(_ | L, N) :-
4     list_length(L, N1),
5     N is N1 + 1.
6 % If length of L is N1, then length of [_ | L] will be N1 + 1
```

Output:

```
compiling /home/cg/root/642a750d116ee/main.pg for byte code...
/home/cg/root/642a750d116ee/main.pg compiled, 5 lines read - 647 bytes written, 3 ms
| ?- list_length([1, 2, 3], N).
list_length([1, 2, 3], N).
N = 3

yes
| ?-
```

6. Program to find the maximum of two numbers

Code:

```
1 find_max(X, Y, X) :- X >= Y, !.
2 find_max(X, Y, Y) :- X < Y.
3
4 find_min(X, Y, X) :- X <= Y, !.
5 find_min(X, Y, Y) :- X > Y.
```

Output:

```
compiling /home/cg/root/642a750d116ee/main.pg for byte code...
/home/cg/root/642a750d116ee/main.pg compiled, 4 lines read - 859 bytes written, 3 ms
| ?- find_max(100,200,Max).
find_max(100,200,Max).
Max = 200

yes
| ?- |
```

7. Program to find the factorial of a number

Code:

```
1 % Write a Prolog program to calculate the factorial of a given number.
2
3
4 fact(0,1).
5 fact(N,F):-
6 (
7
8 % The below is for +ve factorial.
9 N>0 ->
10 (
11 N1 is N-1,
12 fact(N1,F1),
13 F is N*F1
14 )
15 ;
16
17
18 % The below is for -ve factorial.
19 N<0 ->
20 (
21 N1 is N+1,
22 fact(N1,F1),
23 F is N*F1
24 )
25 ).
```

Output:

```
compiling /home/cg/root/642a750d116ee/main.pg for byte code...
/home/cg/root/642a750d116ee/main.pg compiled, 24 lines read - 1810 bytes written, 3 ms
| ?- fact(5, R).
fact(5, R).
R = 120 ? |
```

8. Program to find the nth Fibonacci number

Code:

```
1 fib(1,0).
2 fib(2,1).
3 fib(N,X):- N1 is N-1,N2 is N-2,fib(N1,X1),fib(N2,X2),X is X1+X2,!.
```

Output:

```
compiling /home/cg/root/642a750d116ee/main.pg for byte code...
/home/cg/root/642a750d116ee/main.pg compiled, 2 lines read - 1100 bytes written, 4 ms
| ?- fib(5,X).
fib(5,X).
X = 3

yes
| ?-
```

9. Program to find the sum of a list of numbers

Code:

```
1 list_sum([],0).
2 list_sum([Head|Tail], TotalSum):-
3 list_sum(Tail, Sum1),
4 TotalSum is Head+Sum1.
```

Output:

```
GNU Prolog 1.5.0 (64 bits)
Compiled Feb 22 2023, 13:01:45 with gcc
Copyright (C) 1999-2023 Daniel Diaz

compiling /home/cg/root/642a750d116ee/main.pg for byte code...
/home/cg/root/642a750d116ee/main.pg compiled, 3 lines read - 640 bytes written, 3 ms
| ?- list_sum([5,2,0,4], Sum).
list_sum([5,2,0,4], Sum).
Sum = 11

yes
| ?-
```

10. Program to find the smallest element in a list.

Code:

```
1 min_of_two(X,Y,X) :- X =< Y, !.  
2 min_of_two(X,Y,Y) :- X > Y.  
3 list_min_elem([X],X).  
4 list_min_elem([X,Y|Rest],Min) :-  
5 list_min_elem([Y|Rest],MinRest),  
6 min_of_two(X,MinRest,Min).  
7
```

Output:

```
compiling /home/cg/root/642a910f6812b/main.pg for byte code...  
/home/cg/root/642a910f6812b/main.pg compiled, 5 lines read - 1171 bytes written, 4 ms  
| ?- list_min_elem([10,1,2,5,3],Min).  
list_min_elem([10,1,2,5,3],Min).  
Min = 1 ? |
```