AI PRACTICAL FILE

Department of Computer Science
Aryabhatta College University of Delhi
Session: 2023-2024

Submitted to :

Ms. Neha Kumari

Submitted by :

Harsh Dev Shukla
College Rollno : CSC/21/60
University Rollno : 21059570012

Q11. Write a program in PROLOG to implement palindrome (L) which checks whether a list L is a palindrome or not.

```
check([H|T],[X|Y]):-H=:=X,check(T,Y).
check([_H|_T],[_X|_Y]):-write(" given list is Not a Plaindrome").
check([],[]):-write(" given list is Palindrome").
palindrome([H|T]):- reverse([H|T],[X|Y]),check([H|T],[X|Y]).
```

q11 - Notepad

File Edit Format View Help

```
check([H|T],[X|Y]):-H=:=X,check(T,Y).
check([_H|_T],[_X|_Y]):-write(" given list is Not a Plaindrome").
check([],[]):-write(" given list is Palindrome").
palindrome([H|T]):- reverse([H|T],[X|Y]),check([H|T],[X|Y]).
```

Output

```
?-
% f:/Himanshu_AI_practical/q11.pl compiled 0.00 sec, 4 clauses
?- palindrome([1,2,3,2,1]).
 given list is Palindrome
true
```

Q12. Write a Prolog program to implement sumlist(L, S) so that S is the sum of a given list L.

```
sumlist([], 0).
sumlist([Head|Tail], S) :-
 sumlist(Tail, TailSum),
 S is Head + TailSum.
main :-
 write('Enter a list: '),
 read(L),
 sumlist(L, S),
 write('Sum of the given list is: '),
 write(S).
```

File Edit Format View Help

```
sumlist([], 0).
sumlist([Head|Tail], S) :-
 sumlist(Tail, TailSum),
 S is Head + TailSum.
main :-
 write('Enter a list: '),
 read(L),
 sumlist(L, S),
 write('Sum of the given list is: '),
 write(S).
```

Output

```
true
% f:/Himanshu_AI_practical/q12.pl compiled 0.00 sec, 3 clauses

Unknown action: m (h for help)
Action? .

?- main.
Enter a list: [1,2,34,3,5].
Sum of the given list is: 45
true.

?-
```

Q13. Write a Prolog program to implement two predicates evenlength(List) and oddlength(List) so that they are true if their argument is a list of even or odd length respectively

```
evenlength([]).
evenlength([_, _|Tail]) :-
 evenlength(Tail).
oddlength([_]).
oddlength([_, _|Tail]) :-
 oddlength(Tail).
main :-
 write('Enter a list: '),
 read(List),
 ( evenlength(List)
 -> write('List has even length.')
 ; write('  ')
 ), nl,
 ( oddlength(List)
 -> write('List has odd length.')
 ; write('  ')
 ).
```

File Edit Format View Help

```
evenlength([]).
evenlength([_, _|Tail]) :-
 evenlength(Tail).
oddlength([_]).
oddlength([_, _|Tail]) :-
 oddlength(Tail).
main :-
 write('Enter a list: '),
 read(List),
 ( evenlength(List)
 -> write('List has even length.')
 ; write('  ')
 ), nl,
 ( oddlength(List)
 -> write('List has odd length.')
 ; write('  ')
 ).
```

Output

```
% f:/Himanshu_AI_practical/q13.pl compiled 0.00 sec, 5 clauses
?- main.
Enter a list: [1,2,3,4,5].

List has odd length.
true.

?-
```

Q14. Write a Prolog program to implement nth_element (N, L, X) where N is the desired position, L is a list and X represents the Nth element of L.

```
nth_element(1, [X|_], X).
nth_element(N, [_|Tail], X) :-
 N > 1,
 N1 is N - 1,
 nth_element(N1, Tail, X).
main :-
 write('Enter a list: '),
 read(List),
 write('Enter the position: '),
 read(Position),
 nth_element(Position, List, Element),
 format('The element at position ~w is: ~w', [Position, Element]).
```

q14 - Notepad

File Edit Format View Help

```
nth_element(1, [X|_], X).
nth_element(N, [_|Tail], X) :-
 N > 1,
 N1 is N - 1,
 nth_element(N1, Tail, X).
main :-
 write('Enter a list: '),
 read(List),
 write('Enter the position: '),
 read(Position),
 nth_element(Position, List, Element),
 format('The element at position ~w is: ~w', [Position, Element]).
```

Output

```
% f:/Himanshu_AI_practical/q14.pl compiled 0.00 sec, 3 clauses
?- main.
Enter a list: [1,3,2,4,5,8].
Enter the position: |: 1
|: .
The element at position 1 is: 1
true .

?-
```