Submitted To : Ma'am Neha

Name: Khushi Chhatwani

Course: B.Sc (hons.) Computer Science, III Year, VI Semester

College Roll no. : CSC/21/55

University Roll no. : 21059570021

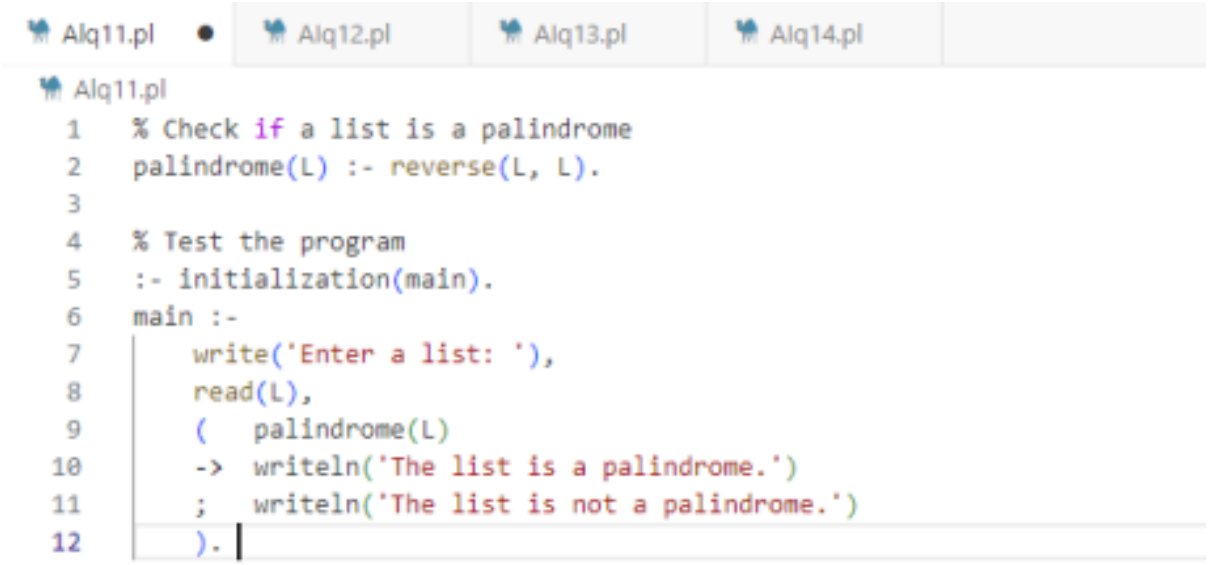
Practical file for Core Paper XIII: Artificial Intelligence

Name: Khushi Chhatwani
College Roll no. : CSC/21/55
University Roll no. : 21059570021

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

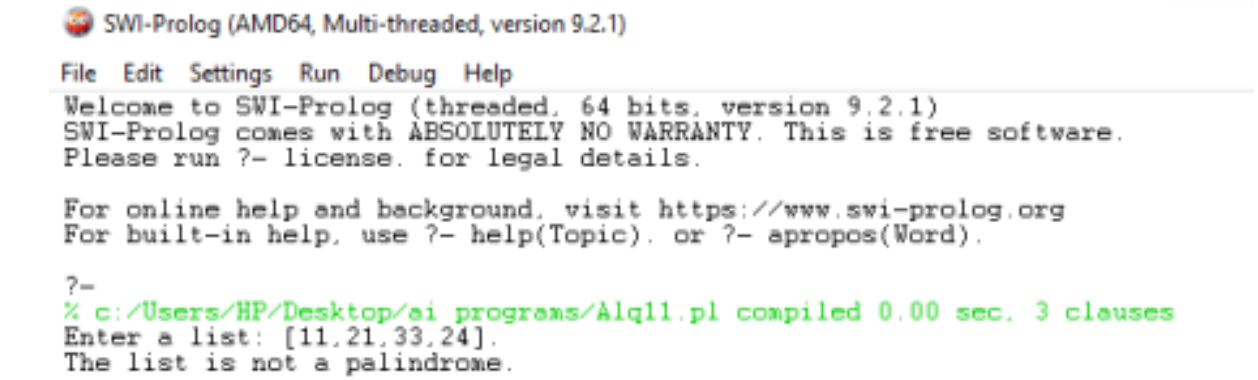
Editor Code

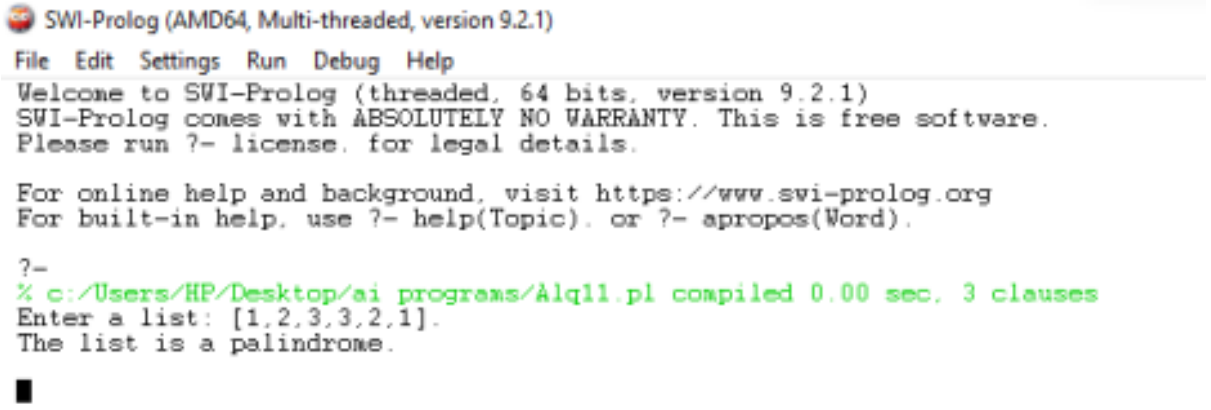
```
% Check if a list is a palindrome
palindrome(L) :- reverse(L, L).

% Test the program
:- initialization(main).
main :-
    write('Enter a list: '),
    read(L),
    ( palindrome(L)
    -> writeln('The list is a palindrome.')
    ; writeln('The list is not a palindrome.')
    ).
```



Name: Khushi Chhatwani
College Roll no. : CSC/21/55
University Roll no. : 21059570021

Output





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

Editor Code

```
% Base case: the sum of an empty list is 0
sumlist([], 0).

% Recursive case: the sum of a list with head H and tail T
% is H plus the sum of T
sumlist([H|T], S) :-
    sumlist(T, ST),
    S is H + ST.
```

Name: Khushi Chhatwani
College Roll no. : CSC/21/55
University Roll no. : 21059570021

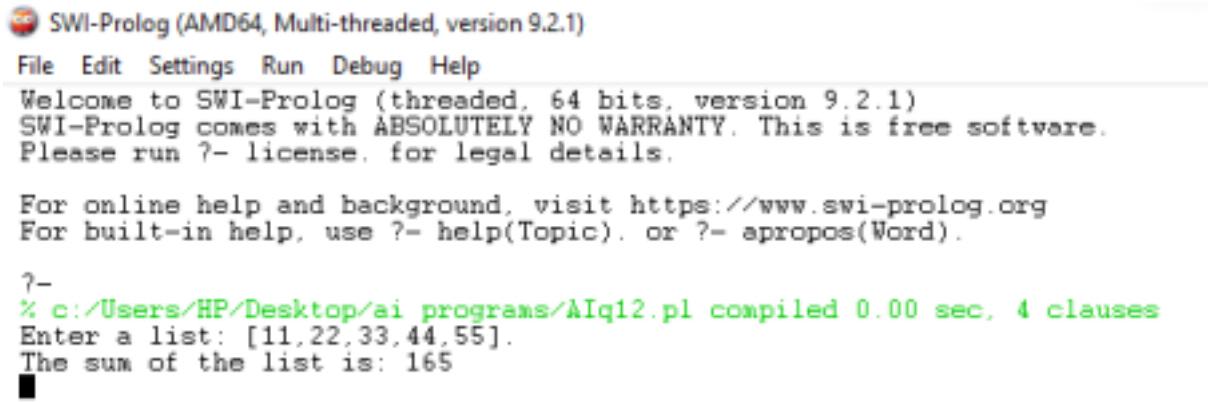
```
% Test the program
 :- initialization(main).
main :-
    write('Enter a list: '),
    read(L),
    sumlist(L, S),
    write('The sum of the list is: '), writeln(S).
```

Alq11.pl ● Alq12.pl ● Alq13.pl Alq14.pl

Alq12.pl

```
1   % Base case: the sum of an empty list is 0
2   sumlist([], 0).
3
4   % Recursive case: the sum of a list with head H and tail T
5   % is H plus the sum of T
6   sumlist([H|T], S) :-
7       sumlist(T, ST),
8       S is H + ST.
9
10  % Test the program
11  :- initialization(main).
12  main :-
13      write('Enter a list: '),
14      read(L),
15      sumlist(L, S),
16      write('The sum of the list is: '), writeln(S).
```

Output



Name: Khushi Chhatwani
College Roll no. : CSC/21/55
University Roll no. : 21059570021

13. 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.

Editor Code

```
% Base case: an empty list has even length
evenlength([]).

% Recursive case: if the tail of a list has odd length, then the list
has even  length
evenlength([_|T]) :- oddlength(T).

% Base case: a list with one element has odd length
oddlength([_]).

% Recursive case: if the tail of a list has even length, then the list
has odd  length
oddlength([_|T]) :- evenlength(T).

% Test the program
:- initialization(main).
main :-
    write('Enter a list: '),
    read(L),
    ( evenlength(L)
    -> writeln('The list has even length.')
    ; writeln('The list has odd length.')
    ).
```

Name: Khushi Chhatwani
College Roll no. : CSC/21/55
University Roll no. : 21059570021

PRACTICAL FILE - Core Paper XIII: Artificial Intelligence

Alq11.pl ● Alq12.pl ● Alq14.pl Alq13.pl ×

Alq13.pl

```
1   % Base case: an empty list has even length
2   evenlength([]).
3
4   % Recursive case: if the tail of a list has odd length, then the list has even length
5   evenlength([_|T]) :- oddlength(T).
6
7   % Base case: a list with one element has odd length
8   oddlength([_]).
9
10  % Recursive case: if the tail of a list has even length, then the list has odd length
11  oddlength([_|T]) :- evenlength(T).
12
13  % Test the program
14  :- initialization(main).
15  main :-
16      write('Enter a list: '),
17      read(L),
18      (   evenlength(L)
19      -> writeln('The list has even length.')
20      ;   writeln('The list has odd length.')
21      ).
```

Output

SWI-Prolog (AMD64, Multi-threaded, version 9.2.1)

File Edit Settings Run Debug Help

```
Welcome to SWI-Prolog (threaded, 64 bits, version 9.2.1)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

For online help and background, visit https://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?-
% c:/Users/HP/Desktop/ai programs/Alq13.pl compiled 0.00 sec, 6 clauses
Enter a list: [11,12,13,14,15,16].
The list has even length.
```

SWI-Prolog (AMD64, Multi-threaded, version 9.2.1)

File Edit Settings Run Debug Help

```
Welcome to SWI-Prolog (threaded, 64 bits, version 9.2.1)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

For online help and background, visit https://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?-
% c:/Users/HP/Desktop/ai programs/Alq13.pl compiled 0.00 sec, 6 clauses
Enter a list: [1,2,3,4,5].
The list has odd length.
```

Name: Khushi Chhatwani
College Roll no. : CSC/21/55
University Roll no. : 21059570021

14. 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.

Editor Code

```
% Base case: the first element of a list is at position 1
nth_element(1, [X|_], X).

% Recursive case: the Nth element of a list with head H and
tail T % is the (N-1)th element of T
nth_element(N, [_|T], X) :-
    N > 1,
    N1 is N - 1,
    nth_element(N1, T, X).

% Test the program
:- initialization(main).
main :-
    write('Enter a list: '),
    read(L),
    write('Enter a position: '),
    read(N),
    ( nth_element(N, L, X)
    -> write('The element at position '), write(N), write(' is: '),
    writeln(X) ; writeln('The position is out of range.')
    ).
```

Name: Khushi Chhatwani
College Roll no. : CSC/21/55
University Roll no. : 21059570021

Alq11.pl ● Alq12.pl ● Alq13.pl Alq14.pl ×

Alq14.pl

```
% Base case: the first element of a list is at position 1
nth_element(1, [X|_], X).

% Recursive case: the Nth element of a list with head H and tail T
% is the (N-1)th element of T
nth_element(N, [_|T], X) :-
    N > 1,
    N1 is N - 1,
    nth_element(N1, T, X).

% Test the program
:- initialization(main).
main :-
    write('Enter a list: '),
    read(L),
    write('Enter a position: '),
    read(N),
    (   nth_element(N, L, X)
    ->  write('The element at position '), write(N), write(' is: '), writeln(X)
    ;   writeln('The position is out of range.')
    ).
```

Output

SWI-Prolog (AMD64, Multi-threaded, version 9.2.1)

File Edit Settings Run Debug Help

```
Welcome to SWI-Prolog (threaded, 64 bits, version 9.2.1)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

For online help and background, visit https://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?-
% c:/Users/HP/Desktop/ai programs/AIq13.pl compiled 0.00 sec, 4 clauses
Enter a list: [10,11,13,12,16,15].
Enter a position: |    4.
The element at position 4 is: 12
```

Name: Khushi Chhatwani
College Roll no. : CSC/21/55
University Roll no. : 21059570021