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

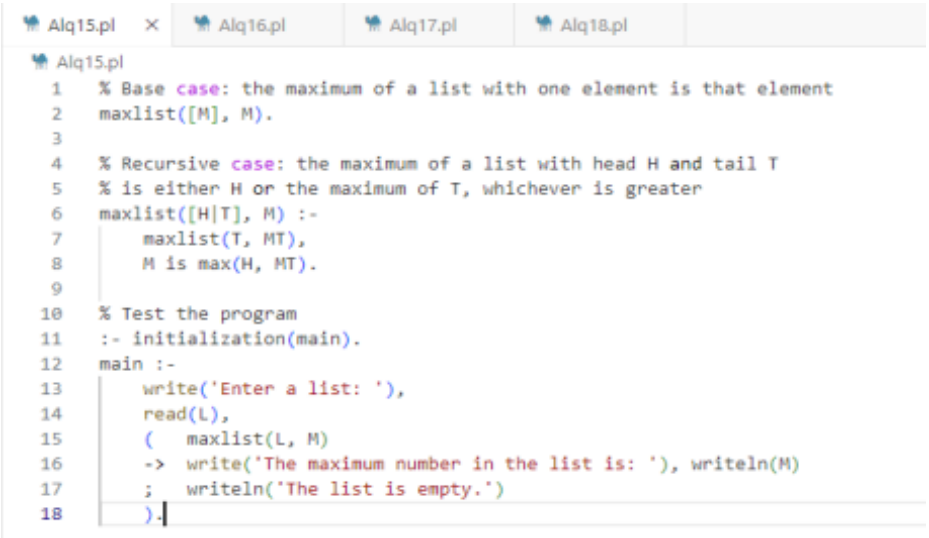
Practical file for Core Paper XIII: Artificial Intelligence

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

15.Write a Prolog program to implement maxlist(L, M) so that M is the maximum number in the list.

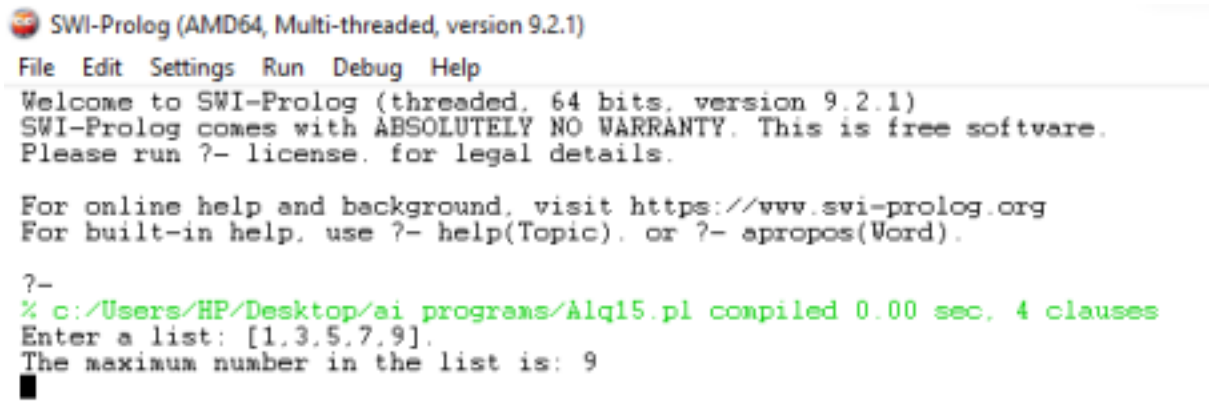
Editor Code

```
% Base case: the maximum of a list with one element is that element
maxlist([M], M).

% Recursive case: the maximum of a list with head
H and tail T % is either H or the maximum of T,
whichever is greater
maxlist([H|T], M) :-
    maxlist(T, MT),
    M is max(H, MT).

% Test the program
:- initialization(main).
main :-
    write('Enter a list: '),
    read(L),
    ( maxlist(L, M)
    -> write('The maximum number in the list is: '), writeln(M)
    ; writeln('The list is empty.')
    ).
```



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

Output



16. Write a prolog program to implement insert_nth (I, N, L, R) that inserts an item I into Nth position of list L to generate a list R.

Editor Code

```
% Base case: Inserting at the beginning of the list
insert_nth(I, 1, L, [I|L]).

% Recursive case: Inserting at a position other than
the beginning insert_nth(I, N, [H|T], [H|R]) :-
    N > 1, % Ensure we're not trying to insert at a position
    less than 1 N1 is N - 1, % Decrement the position
    insert_nth(I, N1, T, R). % Recursive call with the decremented
position and  the tail of the list

% Predicate to read list from user
read_list(L) :-
    write('Enter the list (in square brackets): '),
    read(L).

% Predicate to read position from user
read_position(N) :-
    write('Enter the position: '),
```

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

```
    read(N).

% Main predicate to run the program
:- initialization(main).
main :-
    read_list(L),
    read_position(N),
    write('Enter the item to insert: '),
    read(I),
    insert_nth(I, N, L, R),
    write('The resulting list is: '),
    write(R), nl.
```

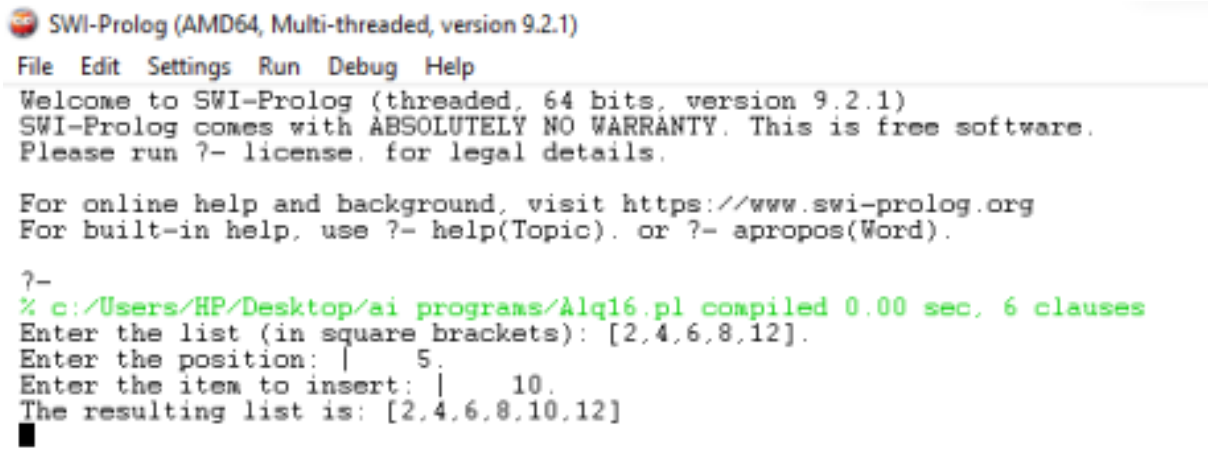
Alq15.pl | Alq16.pl ● | Alq17.pl | Alq18.pl

Alq16.pl

```
1  % Base case: Inserting at the beginning of the list
2  insert_nth(I, 1, L, [I|L]).
3
4  % Recursive case: Inserting at a position other than the beginning
5  insert_nth(I, N, [H|T], [H|R]) :-
6      N > 1, % Ensure we're not trying to insert at a position less than 1
7      N1 is N - 1, % Decrement the position
8      insert_nth(I, N1, T, R). % Recursive call with the decremented position and the tail of the list
9
10 % Predicate to read list from user
11 read_list(L) :-
12     write('Enter the list (in square brackets): '),
13     read(L).
14
15 % Predicate to read position from user
16 read_position(N) :-
17     write('Enter the position: '),
18     read(N).
19
20 % Main predicate to run the program
21 :- initialization(main).
22 main :-
23     read_list(L),
24     read_position(N),
25     write('Enter the item to insert: '),
26     read(I),
27     insert_nth(I, N, L, R),
28     write('The resulting list is: '),
29     write(R), nl.
```

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

Output



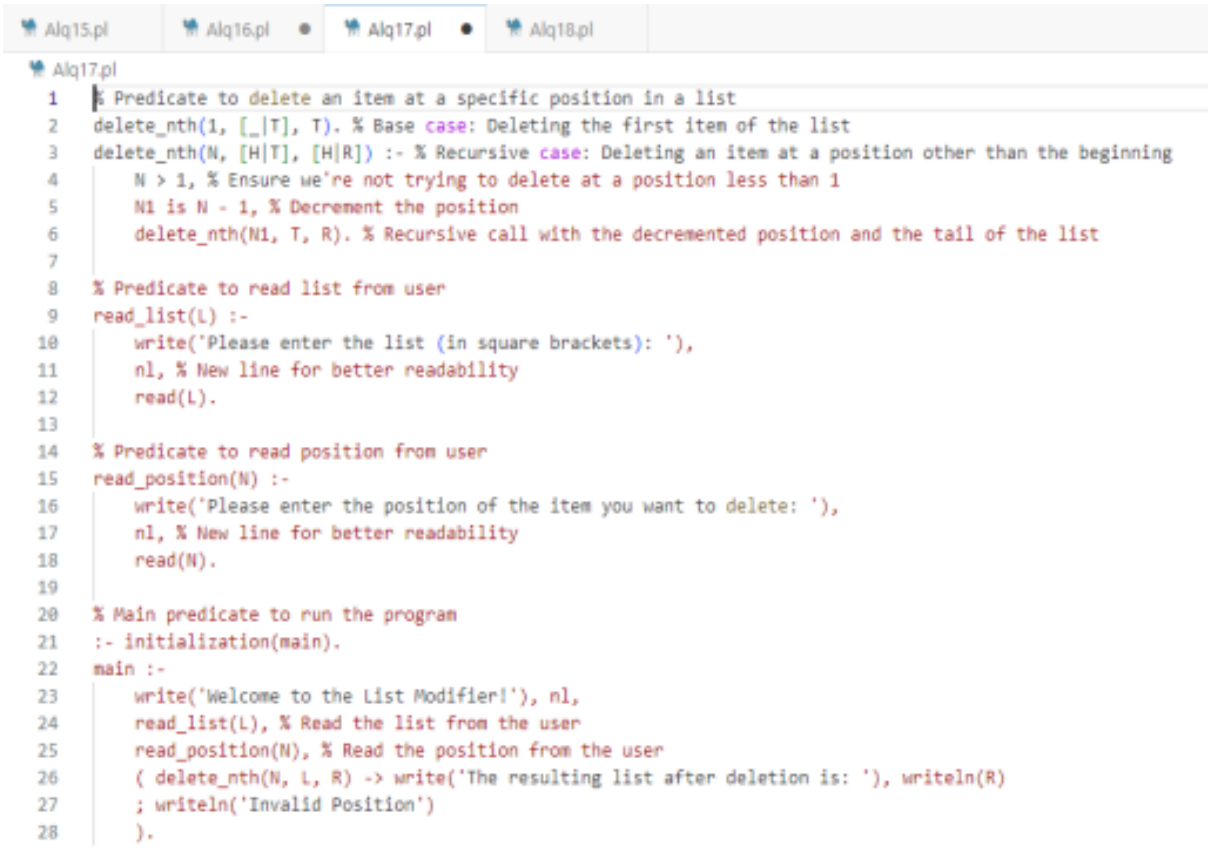
17. Write a Prolog program to implement delete_nth (N, L, R) that removes the element on Nth position from a list L to generate a list R.

Editor Code

```
% Predicate to delete an item at a specific position in a list
delete_nth(1, [_|T], T). % Base case: Deleting the first item of the
list delete_nth(N, [H|T], [H|R]) :- % Recursive case: Deleting an
item at a position  other than the beginning
    N > 1, % Ensure we're not trying to delete at a position
    less than 1 N1 is N - 1, % Decrement the position
    delete_nth(N1, T, R). % Recursive call with the decremented
position and the  tail of the list

% Predicate to read list from user
read_list(L) :-
    write('Please enter the list (in square brackets): '),
    nl, % New line for better readability
    read(L).

% Predicate to read position from user
read_position(N) :-
    write('Please enter the position of the item you want to
    delete: '), nl, % New line for better readability
    read(N).
```

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

```
% Main predicate to run the program
:- initialization(main).
main :-
    write('Welcome to the List Modifier!'), nl,
    read_list(L), % Read the list from the user
    read_position(N), % Read the position from the user
    ( delete_nth(N, L, R) -> write('The resulting list after
deletion is: '),  writeln(R)
    ; writeln('Invalid Position')
    ).
```



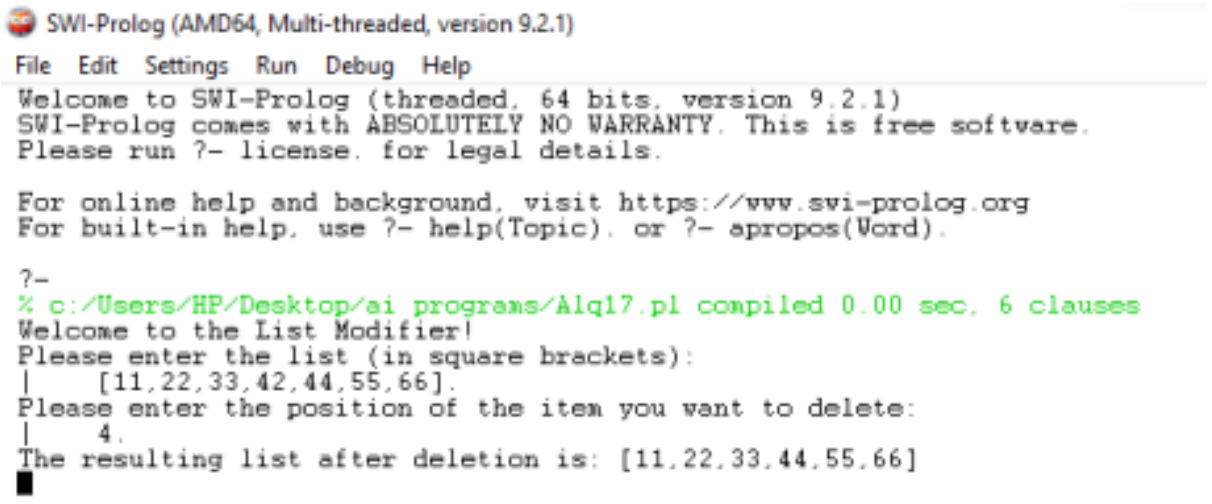
Alq15.pl | Alq16.pl | Alq17.pl | Alq18.pl

Alq17.pl

```
1  % Predicate to delete an item at a specific position in a list
2  delete_nth(1, [_|T], T). % Base case: Deleting the first item of the list
3  delete_nth(N, [H|T], [H|R]) :- % Recursive case: Deleting an item at a position other than the beginning
4      N > 1, % Ensure we're not trying to delete at a position less than 1
5      N1 is N - 1, % Decrement the position
6      delete_nth(N1, T, R). % Recursive call with the decremented position and the tail of the list
7
8  % Predicate to read list from user
9  read_list(L) :-
10     write('Please enter the list (in square brackets): '),
11     nl, % New line for better readability
12     read(L).
13
14 % Predicate to read position from user
15 read_position(N) :-
16     write('Please enter the position of the item you want to delete: '),
17     nl, % New line for better readability
18     read(N).
19
20 % Main predicate to run the program
21 :- initialization(main).
22 main :-
23     write('Welcome to the List Modifier!'), nl,
24     read_list(L), % Read the list from the user
25     read_position(N), % Read the position from the user
26     ( delete_nth(N, L, R) -> write('The resulting list after deletion is: '), writeln(R)
27     ; writeln('Invalid Position')
28     ).
```

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

Output



18. Write a program in PROLOG to implement merge (L1, L2, L3) where L1 is first ordered list and L2 is second ordered list and L3 represents the merged list.

Editor Code

```
merge([], L, L).
merge(L, [], L).
merge([H1|T1], [H2|T2], [H1|T]) :- H1 =< H2, merge(T1,
[H2|T2], T). merge([H1|T1], [H2|T2], [H2|T]) :- H1 >
H2, merge([H1|T1], T2, T).

main :-
    write('Enter first list: '),
    read(L1),
    write('Enter second list: '),
    read(L2),
    merge(L1, L2, L3),
    write('Merged list: '),
    write(L3).

:-initialization(main).
```

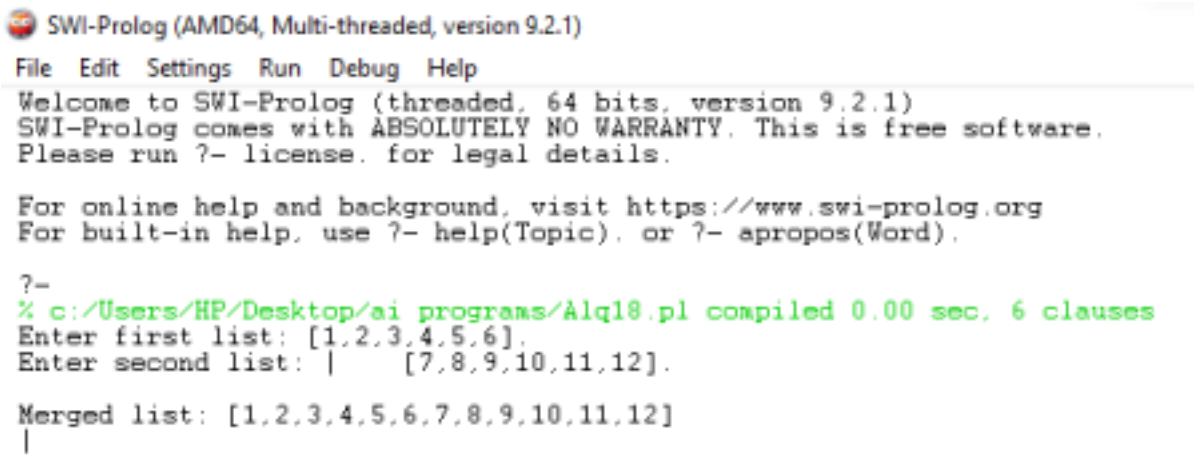
Name: Khushi Chhatwani
College Roll no. : CSC/21/55
University Roll no. : 21059570021

Alq15.pl | Alq16.pl | Alq17.pl | Alq18.pl

Alq18.pl

```
1   merge([], L, L).
2   merge(L, [], L).
3   merge([H1|T1], [H2|T2], [H1|T]) :- H1 =< H2, merge(T1, [H2|T2], T).
4   merge([H1|T1], [H2|T2], [H2|T]) :- H1 > H2, merge([H1|T1], T2, T).
5
6   main :-
7       write('Enter first list: '),
8       read(L1),
9       write('Enter second list: '),
10      read(L2),
11      merge(L1, L2, L3),
12      write('Merged list: '),
13      write(L3).
14
15  :-initialization(main).
```

Output



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