ARTIFICIAL INTELLIGENCE

PRACTICAL FILE

Submitted By:

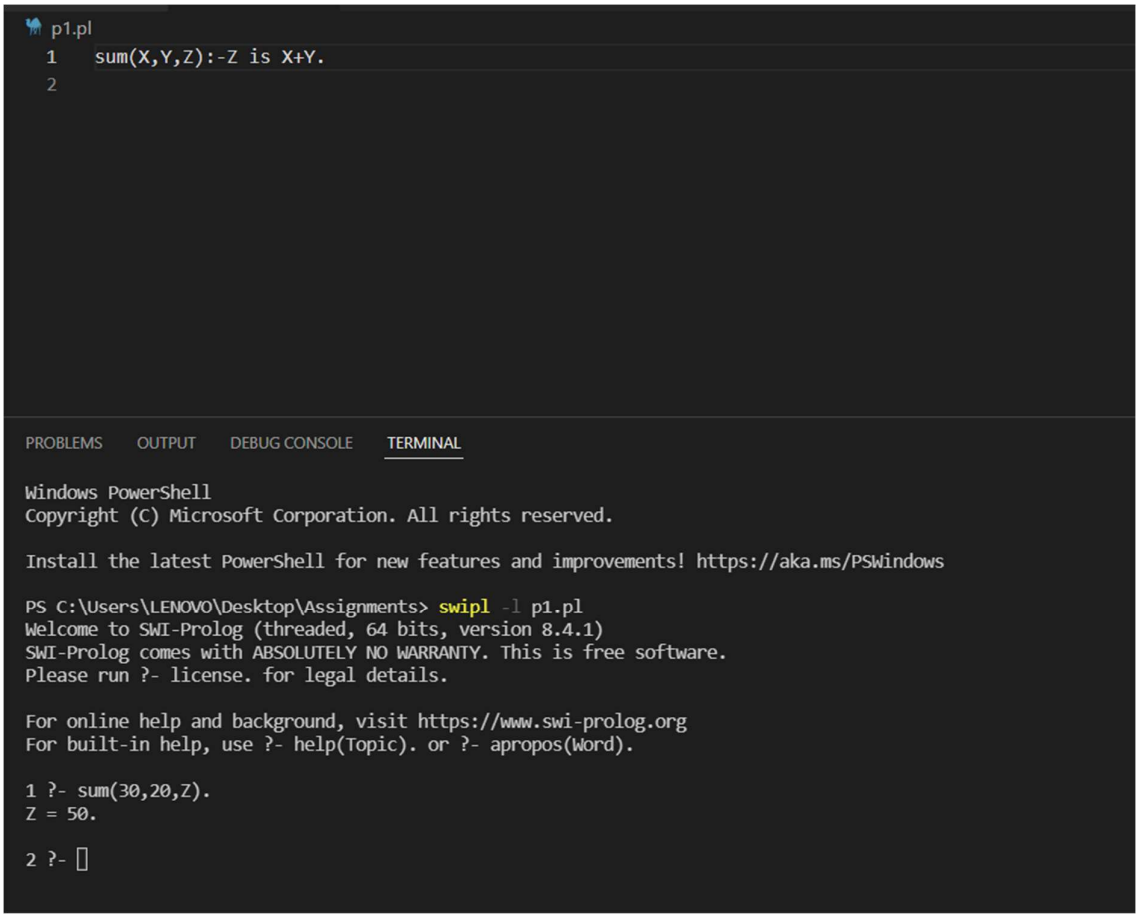
Name: Lakshay Sharma

College Roll No: 19/78013

Course: BSc(H) Computer Science

Sem: VI

Q1. Write a prolog program to calculate the sum of two numbers



2. Write a Prolog program to implement max(X, Y, M) so that M is the maximum of two numbers X and Y.

Q3. Write a program in PROLOG to implement factorial (N, F) where F represents the factorial of a number N.

p3.pl

```
1   factorial(0,1).
2   factorial(N,M):-
3   N>0,
4   N1 is N-1,
5   factorial(N1,M1),
6   M is N*M1.
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.1)
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).

1 ?- factorial(3,6).
true
Unknown action: f (h for help)
Action? .

2 ?- factorial(3,6).
true .

2 ?- factorial(3,4).
false.

3 ?- factorial(5,120).
true
```

Q4. Write a program in PROLOG to implement generate_fib(N,T) where T represents the Nth term of the fibonacci series.

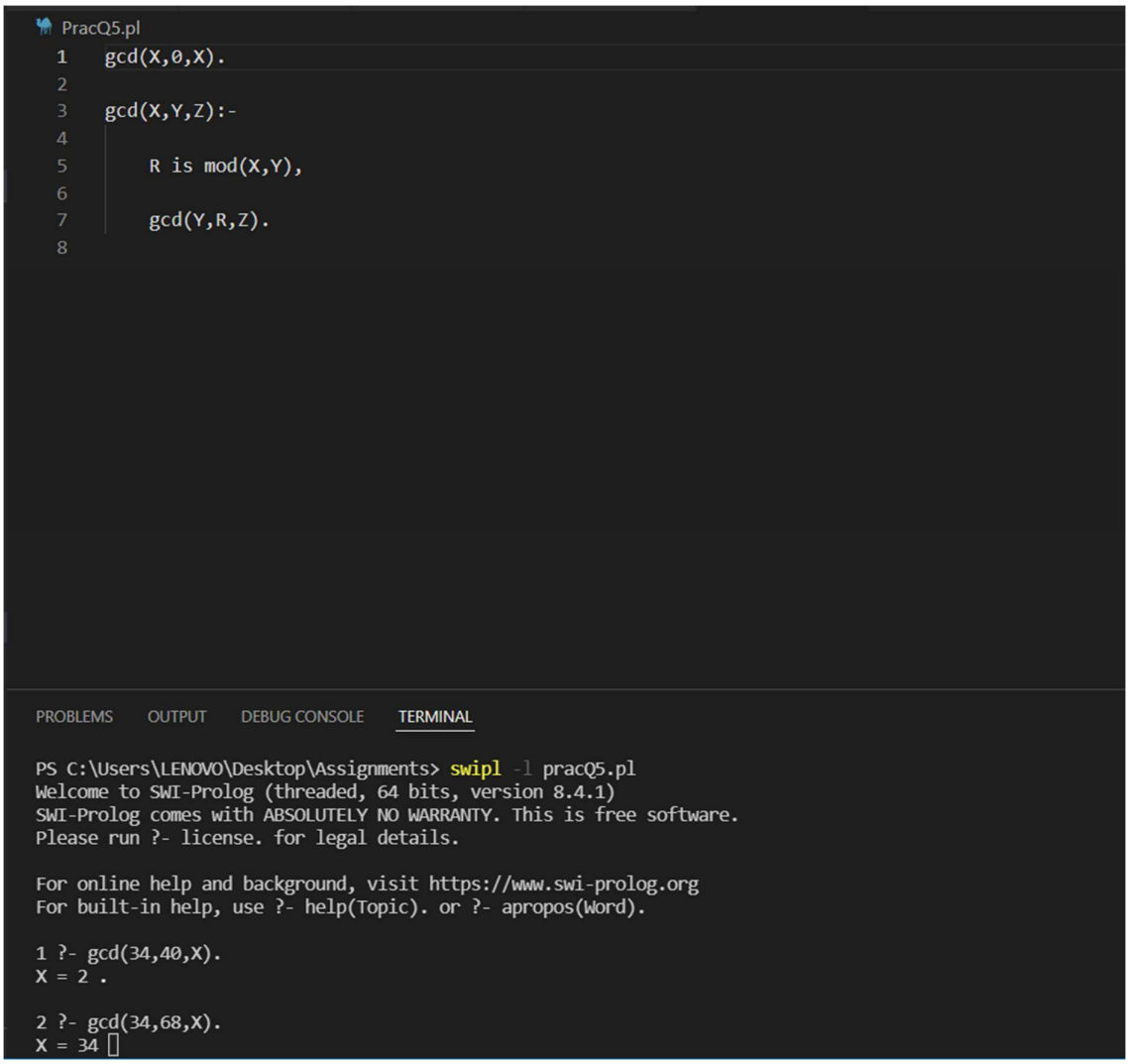
pracQ4.pl

```
fib(0, 1) :- !.

fib(1, 1) :- !.

fib(N, F) :-

        N > 1,

        N1 is N-1,

        N2 is N-2,

        fib(N1, F1),

        fib(N2, F2),

        F is F1+F2.

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\LENOVO\Desktop\Assignments> swipl -l pracQ4.pl
Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.1)
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).

1 ?- fib(10,R).
R = 89.
```

Q5. Write a Prolog program to implement GCD of two numbers.



PracQ5.pl

```
1   gcd(X,0,X).
2
3   gcd(X,Y,Z):-
4
5       R is mod(X,Y),
6
7       gcd(Y,R,Z).
8
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
PS C:\Users\LENOVO\Desktop\Assignments> swipl -l pracQ5.pl
Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.1)
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).

1 ?- gcd(34,40,X).
X = 2 .

2 ?- gcd(34,68,X).
X = 34
```

Q6. Write a Prolog program to implement power (Num,Pow, Ans) : where Num is raised to the power Pow to get Ans.

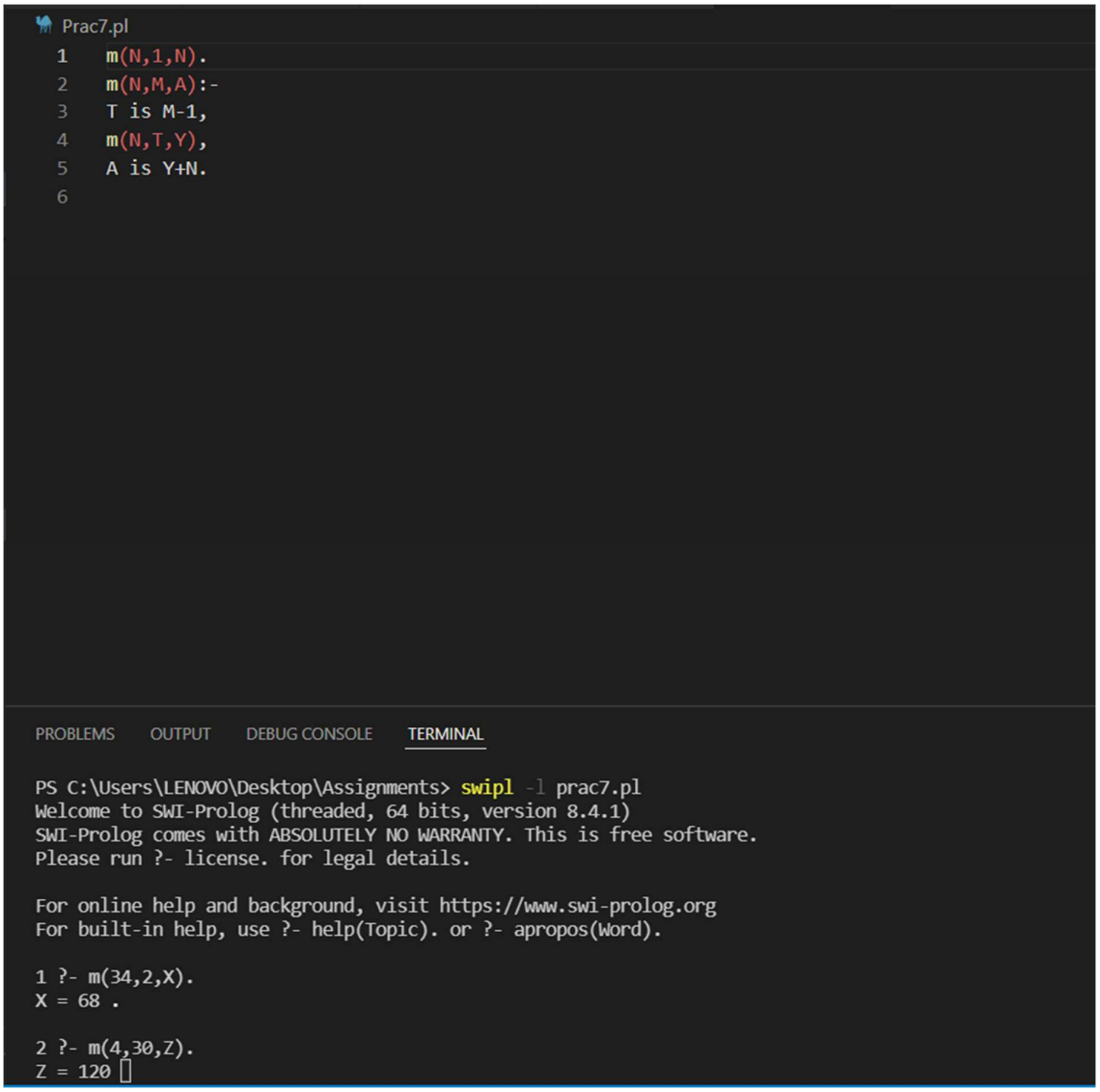
PracQ6.pl

```
power(0,P,0):- P>0.
power(X,0,1):- X>0.
power(X,P,A):- X>0,P>0,P1 is P-1,
power(X,P1,Ans),
A is Ans*X.
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
X = 34 .

3 ?- halt.
PS C:\Users\LENOVO\Desktop\Assignments> swipl -l pracQ6.pl
Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.1)
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).

1 ?- power(2,5,X).
X = 32
```

7. Prolog program to implement multi (N1, N2, R) : where N1 and N2 denotes the numbers to be multiplied and R represents the result.



Q8. Write a Prolog program to implement memb(X, L): to check whether X is a member
of L or not.

PracQ8.pl

```
1   memb(X, [X|Tail]).
2   memb(X, [Head|Tail]):- memb(X, Tail).
3
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
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).

1 ?- memb(3,[1,2,3]).
true .

2 ?- memb(34,[1,2,3,4]).
false.
```

Q9. Write a Prolog program to implement conc(L1, L2, L3) where L2 is the list to be appended with L1 to get the resulted list L3.

PracQ9.pl

```
1   conc([],L1,L1).
2   conc([X|T],L2,[X|T1]):- conc(T,L2,T1).
3
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
3 ?- halt.
PS C:\Users\LENOVO\Desktop\Assignments> swipl -l pracQ9.pl
Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.1)
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).

1 ?- conc([1,2,3],[3,2,1,4],X).
X = [1, 2, 3, 3, 2, 1, 4].
```

Q10. Write a Prolog program to implement reverse (L, R) where List L is original and List R is reversed list.

PracQ10.pl

```
1   reverse_list(Inputlist,Outputlist):-
2   reverse(Inputlist,[],Outputlist).
3   reverse([],Outputlist,Outputlist).
4   reverse([Head|Tail],List1,List2):-
5   reverse(Tail,[Head|List1],List2).
6
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
2 ?- halt.
PS C:\Users\LENOVO\Desktop\Assignments> swipl -l pracQ10.pl
Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.1)
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).

1 ?- reverse_list([2,3,4,5],C).
C = [5, 4, 3, 2].
```

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

PracQ11.pl

```
palind([]):- write('palindrome').
palind([_]):- write('palindrome').
palind(L) :-
 append([H|T], [H], L),
 palind(T)
 ;
 write('Not a palindrome').

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
1 ?- palind('l','a','k','s','h','a','y').
ERROR: Unknown procedure: palind/7
ERROR:     However, there are definitions for:
ERROR:         palind/1
false.

2 ?- palind(['l','a','k','s','h','a','y']).
Not a palindrome
true.

3 ?- palind(['y','a','h','s','h','a','y']).
palindrome
true
```

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

PracQ12.pl

```
1   sumlist([],0).
2   sumlist([H|T],S):- sumlist(T,S1),S is H+S1.
3
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
4 ?- halt.
PS C:\Users\LENOVO\Desktop\Assignments> swipl -l pracQ12.pl
Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.1)
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).

1 ?- sumlist([1,2,3,4],X).
X = 10.
```

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.

PracQ13.pl

```
1   evenlength([]).
2   evenlength([_|T]):- oddlength(T).
3   oddlength([_]).
4   oddlength([_|T]):- evenlength(T).
5
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
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).

1 ?- evenlength([1,2,3,4]).
true .

2 ?- evenlength([1,2,3]).
false.

3 ?- oddlength([1,2,3]).
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.

Prac14.pl

```
1   nth_element(1,[H|T],H).
2   nth_element(N,[H|T],X):- N1 is N-1,nth_element(N1,T,X).
3
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
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).

1 ?- nth_element([1,2,4,51,23],4).
ERROR: Unknown procedure: nth_element/2
ERROR:     However, there are definitions for:
ERROR:         nth_element/3
false.

2 ?- nth_element(4,[1,2,4,51,23],X).
X = 51
```

Q15. Write a Prolog program to implement maxlist(L, M) so that M is the maximum number in the list.

PracQ15.pl

```
1   maxlist([H],H).
2   maxlist([H|T],R):-
3    maxlist(T,M1),
4    H>=M1,
5    R is H.
6   maxlist([H|T],R):-
7    maxlist(T,M1),
8    H<M1,
9    R is M1.
10
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
X = 51 .

3 ?- halt.
PS C:\Users\LENOVO\Desktop\Assignments> swipl -l pracQ15.pl
Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.1)
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).

1 ?- maxlist([23,12,33,44],X).
X = 44
```

Q16. 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.

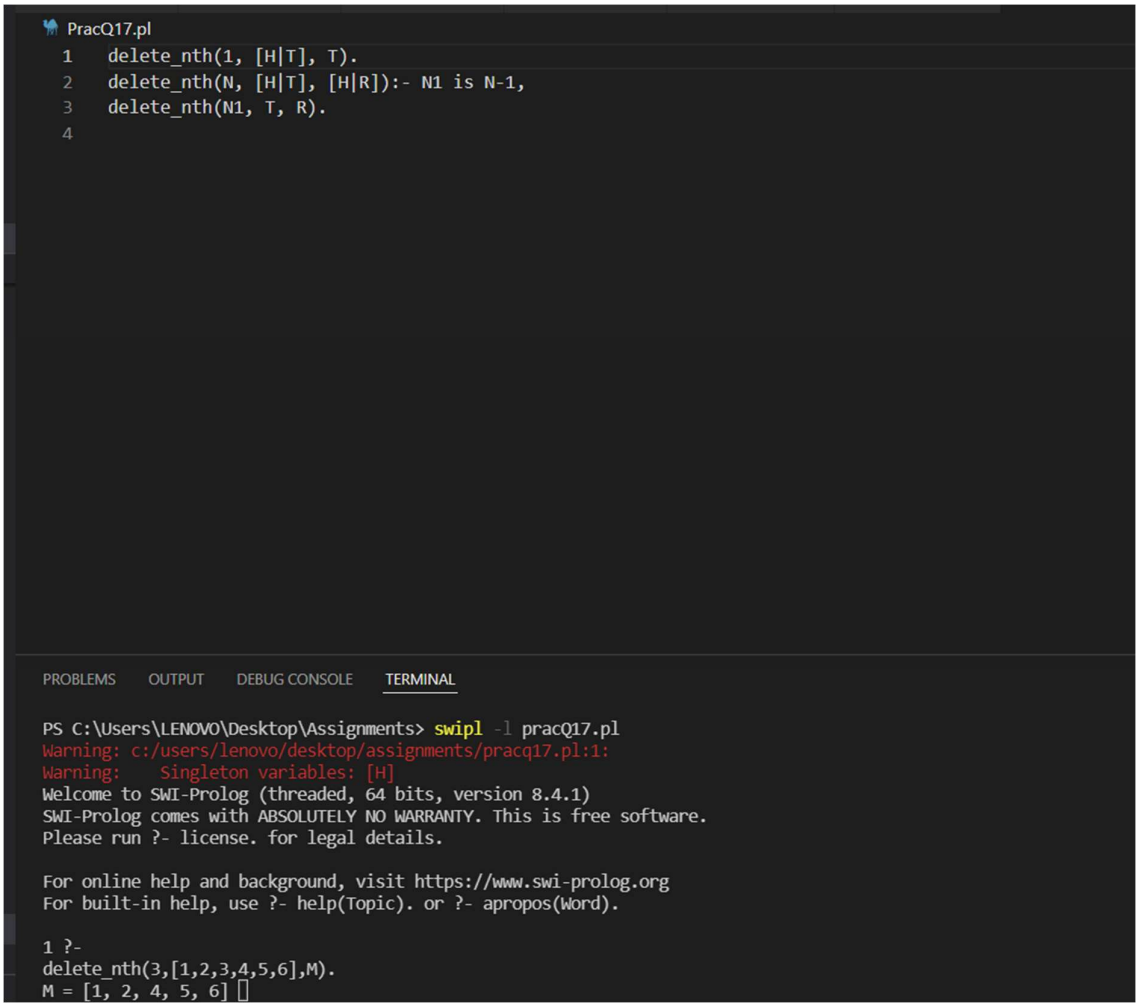
PracQ16.pl

```
1   hmem(X,[X|_]).
2
3   mem(X,[_|T]):- mem(X,T).
4
5   insert(L,[_X|_Y],[L|_]).
6
7   insert(L,P,[X|Y],[X|M]):-
8
9    P>1,
10
11   P1 is P-1,
12
13   insert(L,P1,Y,M).
14
15  insert(L,1,[X|Y],M):- append([L],[X|Y],M).
16
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
PS C:\Users\LENOVO\Desktop\Assignments> swipl -l pracQ16.pl
Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.1)
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).

1 ?- insert(23,4,[a,b,44,55,df]).
false.

2 ?- insert(23,4,[a,b,44,55,df],L).
L = [a, b, 44, 23, 55, df]
```

Q17. 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.



Q18. 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.

Prac18.pl

```
1   merge([],[],[]).
2   merge([],L2,L2).
3   merge(L1,[],L1).
4   merge([H1|T1],[H2|T2],[H1|T3]):- H1=<H2,
5   merge(T1, [H2|T2], T3).
6   merge([H1|T1],[H2|T2],[H2|T3]):- merge([H1|T1], T2, T3).
7
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
ERROR: script_file `'pracQ18.pl'' does not exist
PS C:\Users\LENOVO\Desktop\Assignments> swipl -l prac18.pl
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).

1 ?- merge([1,2,3],[2,3,4],X).
X = [1, 2, 2, 3, 3, 4]
```