ARTIFICIAL INTELLIGENCE PRACTICAL FILE

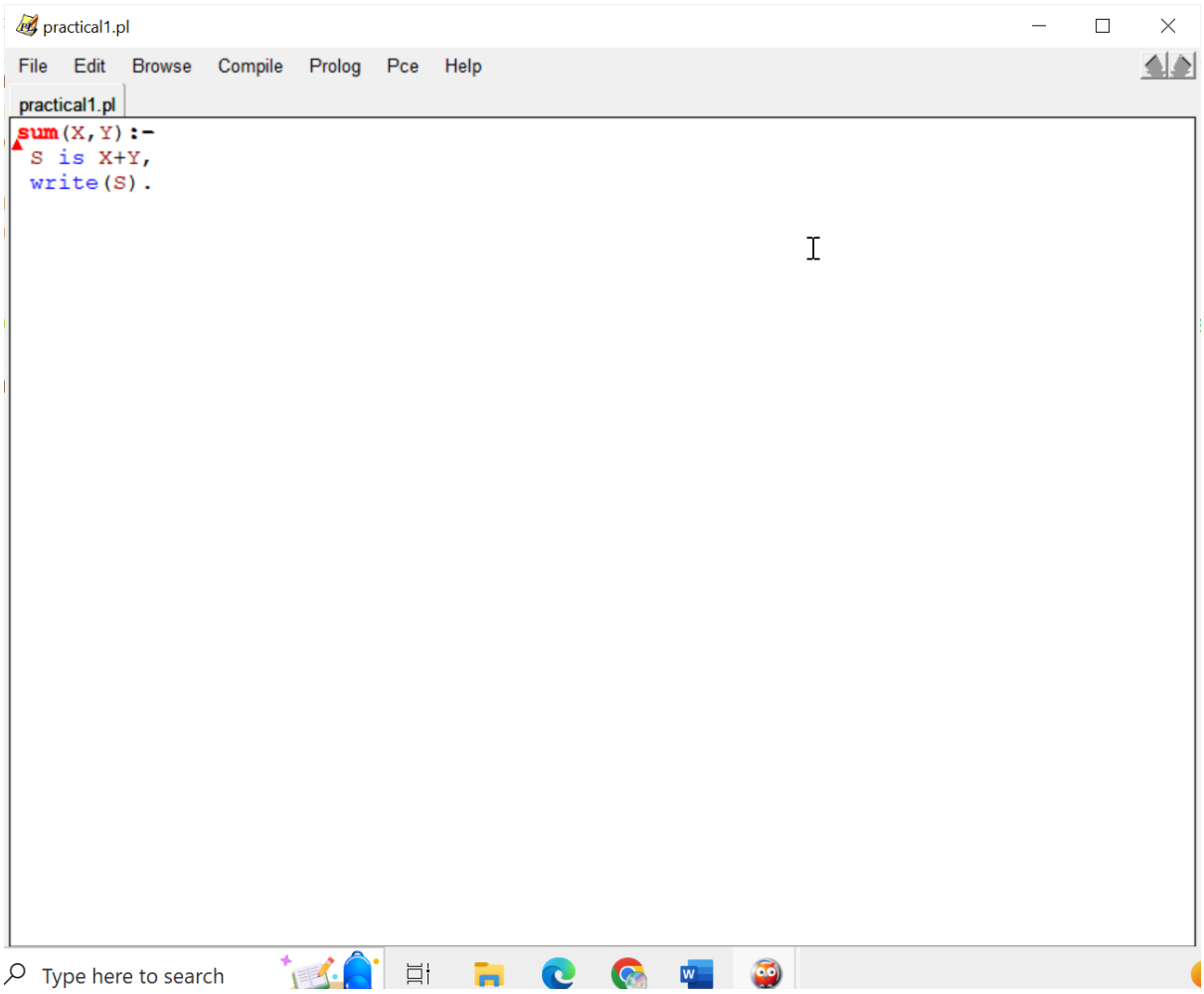
AADITYA KEDIYAL

BSc(Hons) Computer Science

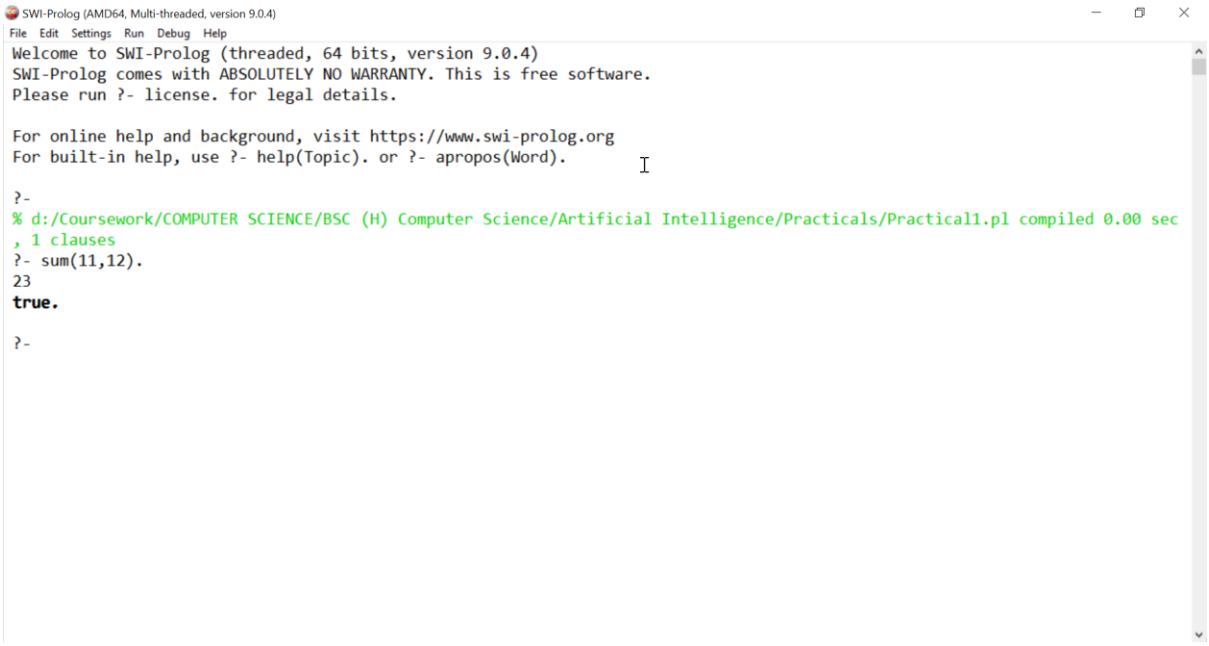
20201401

Ramanujan College

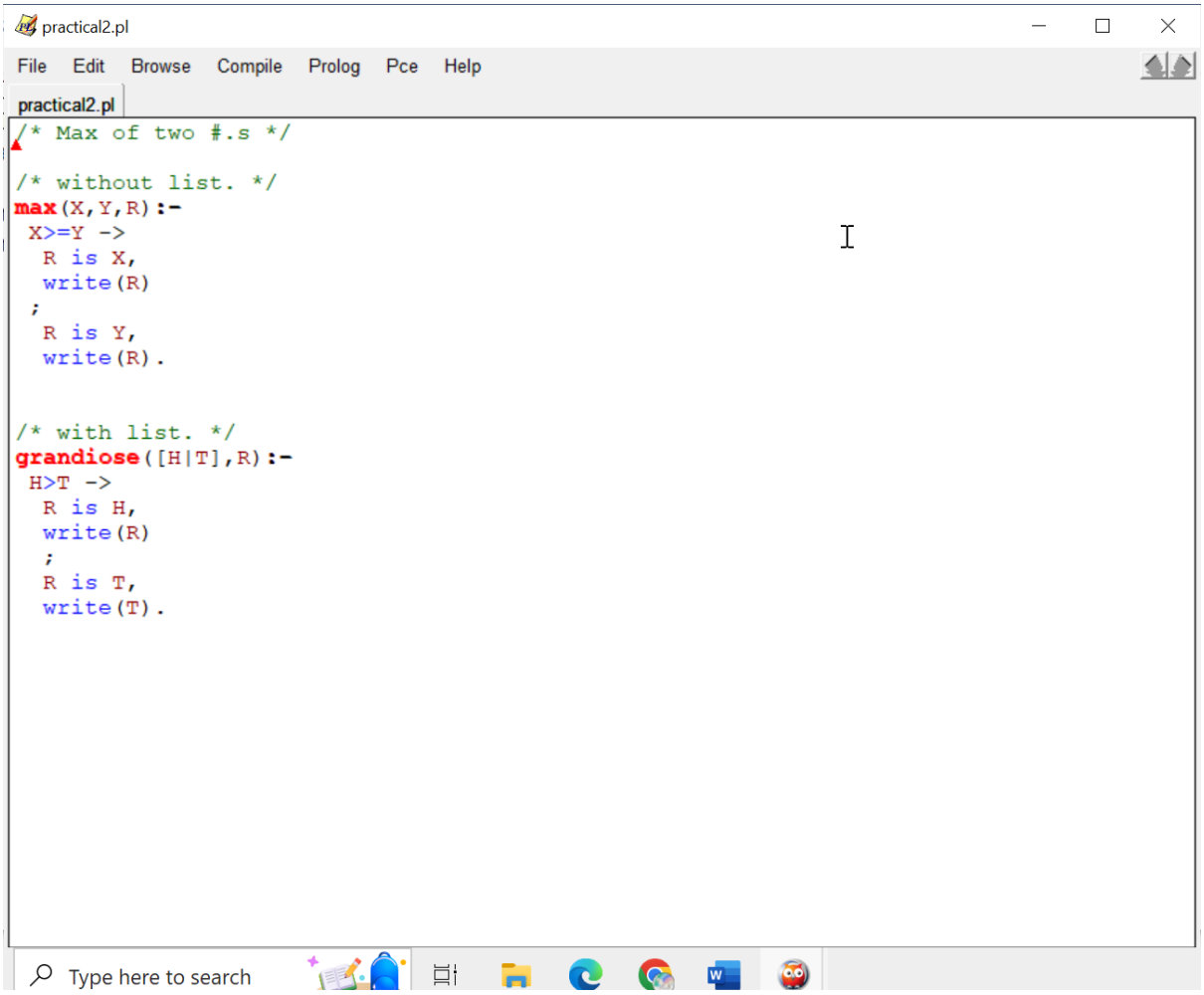
1. Write a prolog program to calculate the sum of two numbers



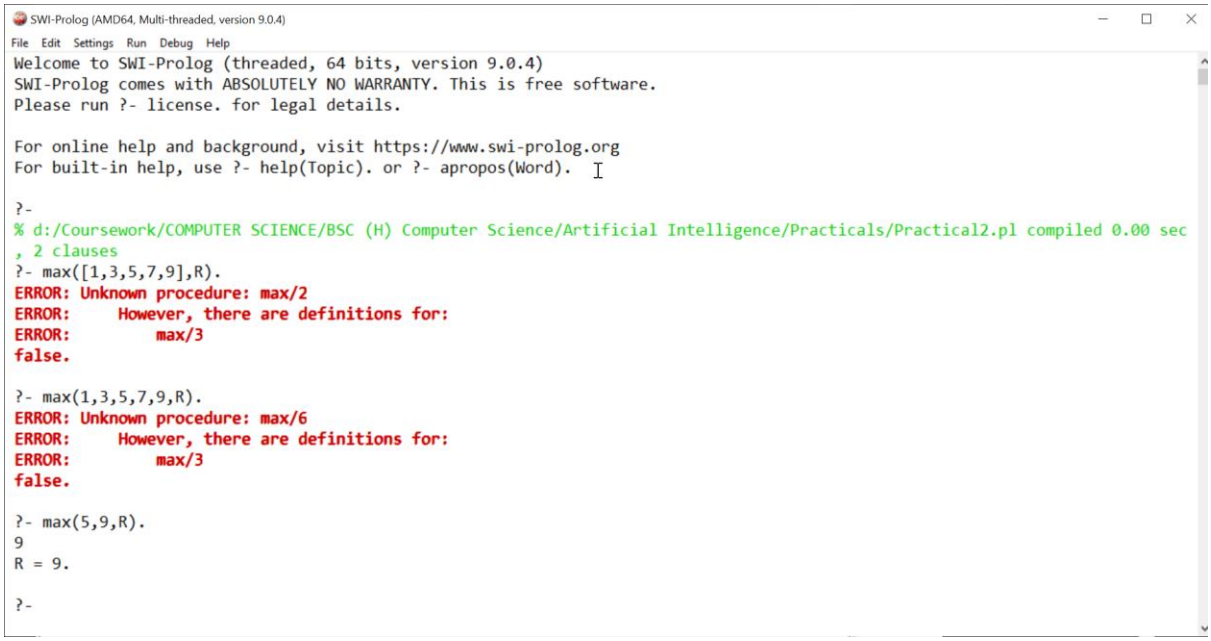
OUTPUT



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



OUTPUT



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

practical3.pl

File Edit Browse Compile Prolog Pce Help

practical3.pl

```
fact(0,1).
fact(N,F):-
(

 % The below is for +ve factorial.
 N>0 ->
 (
  N1 is N-1,
  fact(N1,F1),
  F is N*F1
 )
 ;

 % The below is for -ve factorial.
 N<0 ->
 (
  N1 is N+1,
  fact(N1,F1),
  F is N*F1
 )
).
```

Type here to search

OUTPUT

SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)

File Edit Settings Run Debug Help

```
Welcome to SWI-Prolog (threaded, 64 bits, version 9.0.4)
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).

?-
% d:/Coursework/COMPUTER SCIENCE/BSC (H) Computer Science/Artificial Intelligence/Practicals/Practical3.pl compiled 0.00 sec
, 2 clauses
?- fact(5,F).
F = 120
```

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

practical4.pl

File Edit Browse Compile Prolog Pce Help

practical4.pl

```
generate_fib(0,1).
generate_fib(1,1).
generate_fib(N,T):- N1 is N-1,
generate_fib(N1,T1),
N2 is N-2,
generate_fib(N2,T2),
T is T1+T2.
```

Type here to search

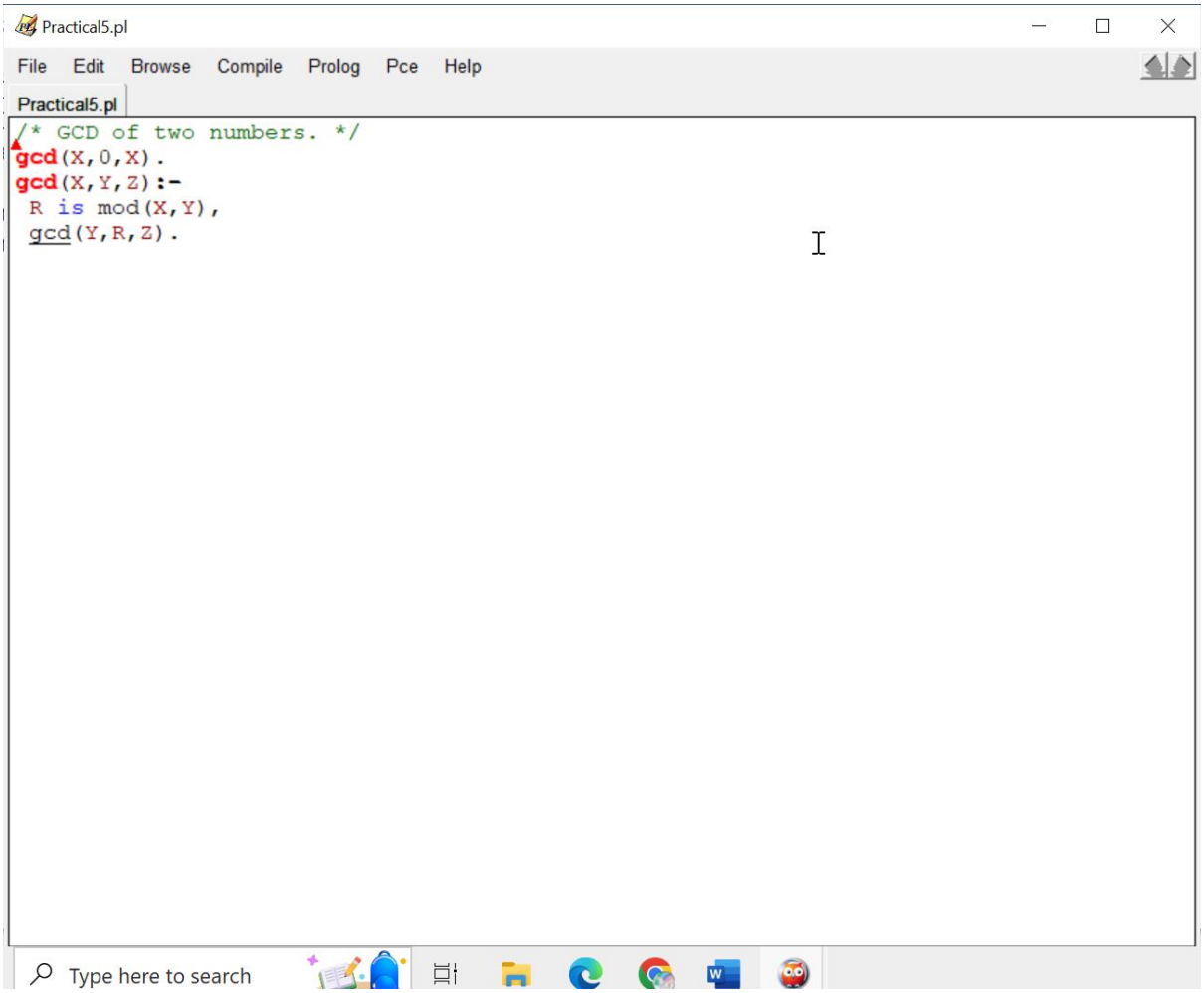
OUTPUT

SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)

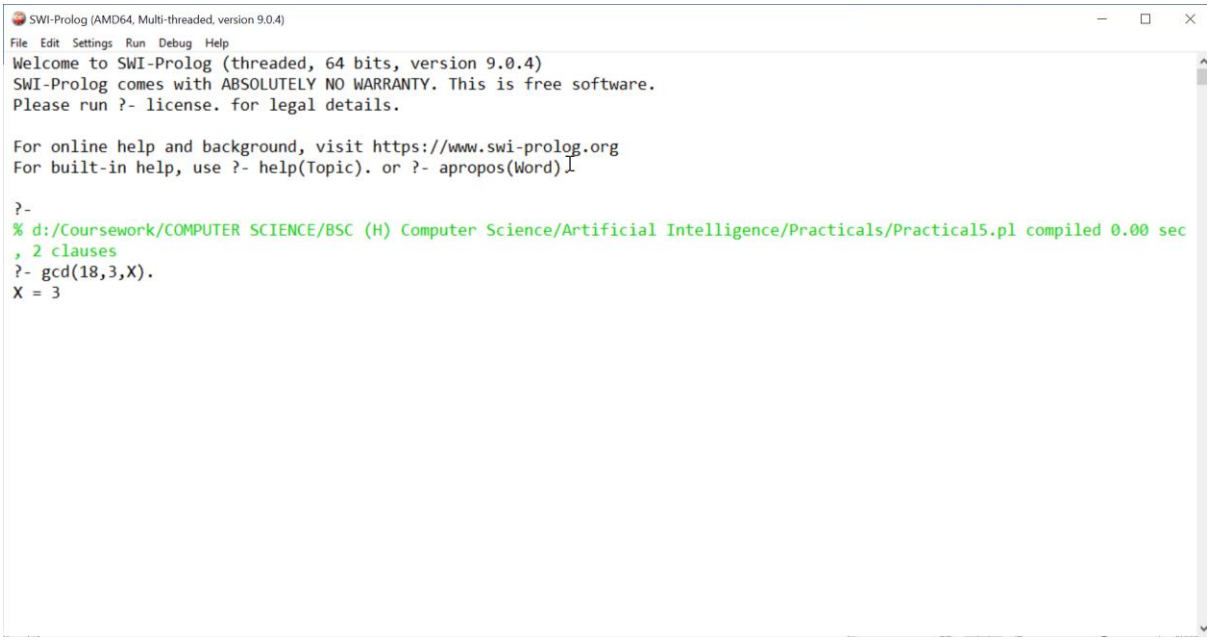
File Edit Settings Run Debug Help

```
Welcome to SWI-Prolog (threaded, 64 bits, version 9.0.4)
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).

?-
% d:/Coursework/COMPUTER SCIENCE/BSC (H) Computer Science/Artificial Intelligence/Practicals/Practical4.pl compiled 0.00 sec
, 3 clauses
?- generate_fib(5,T).
T = 8
```

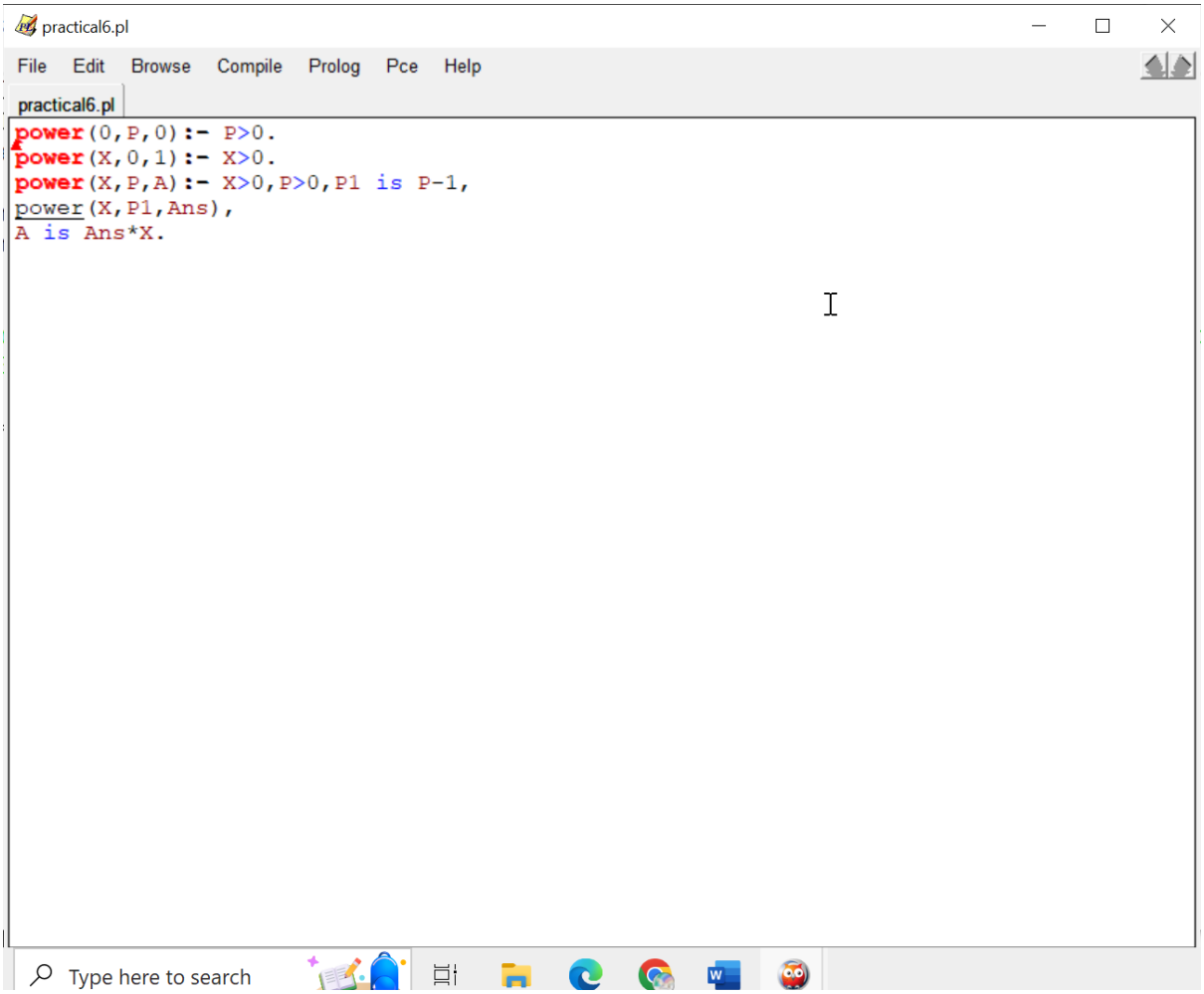
5. Write a Prolog program to implement GCD of two numbers.



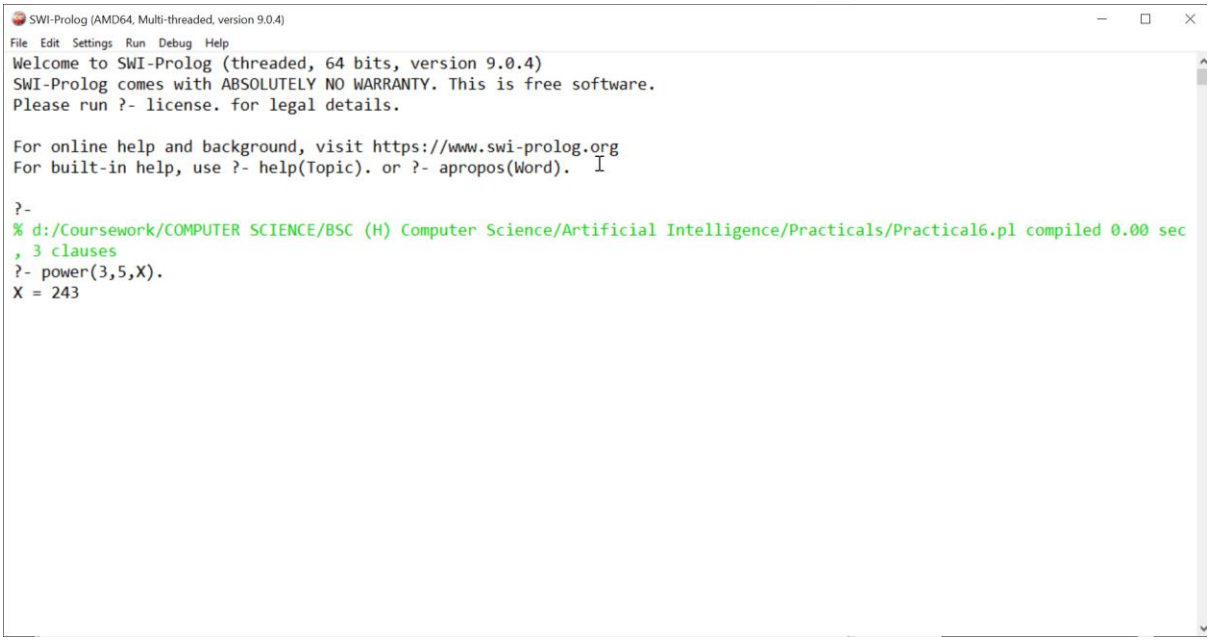
OUTPUT



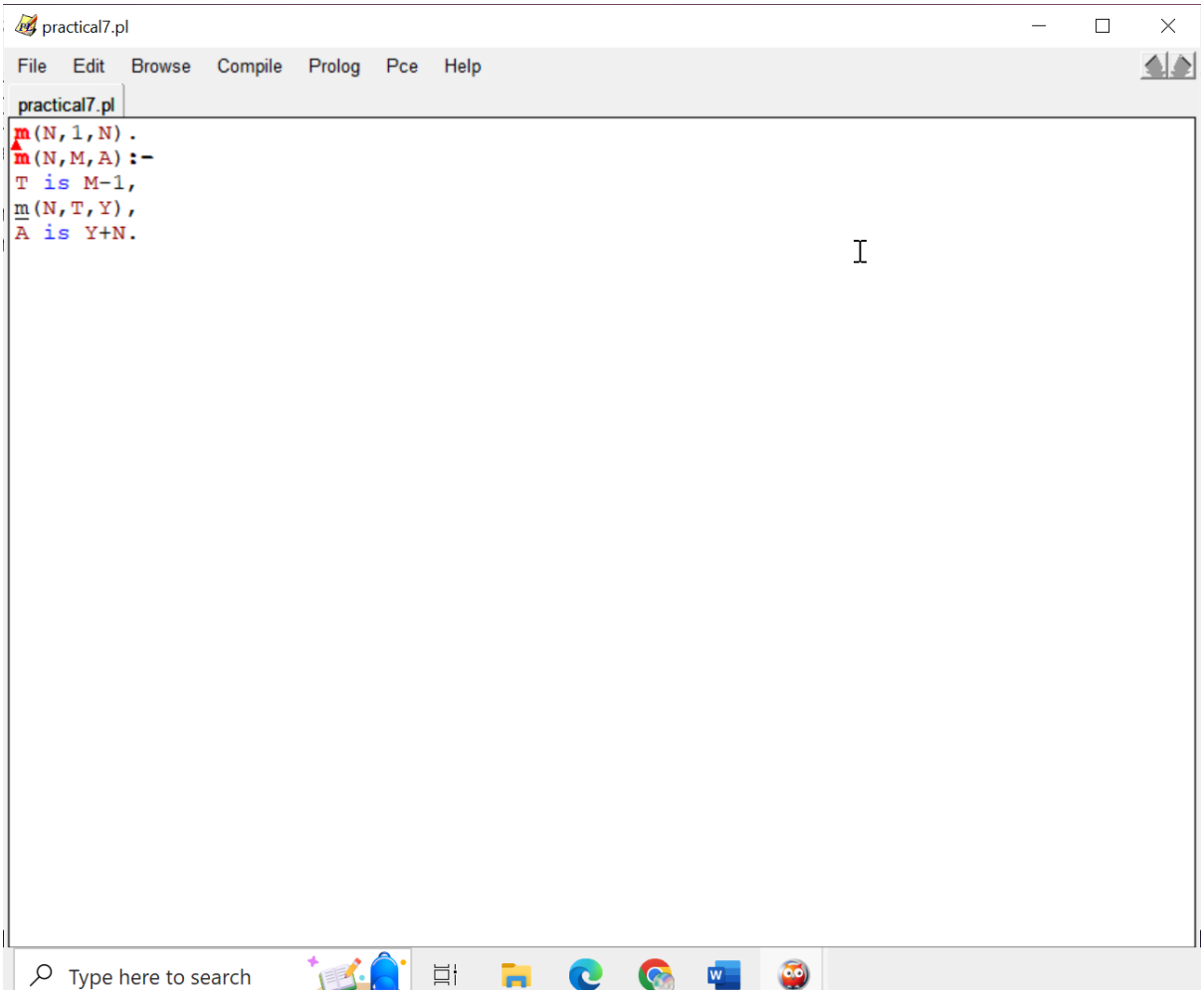
6. Write a Prolog program to implement power (Num,Pow, Ans) : where Num is raised to the power Pow to get Ans.



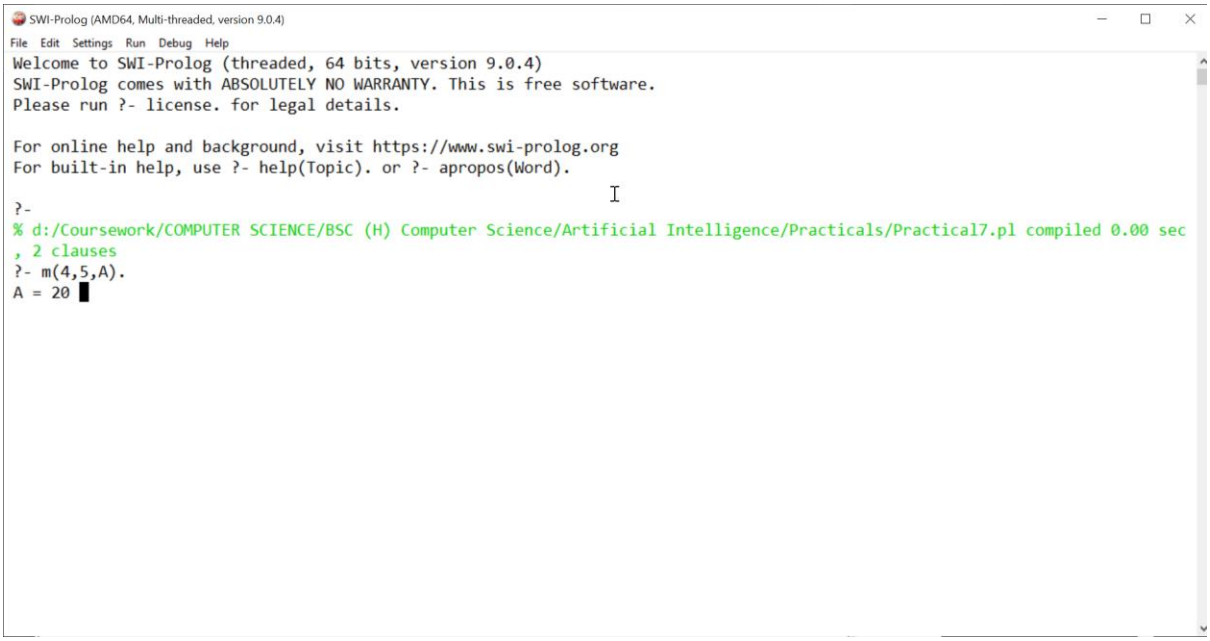
OUTPUT



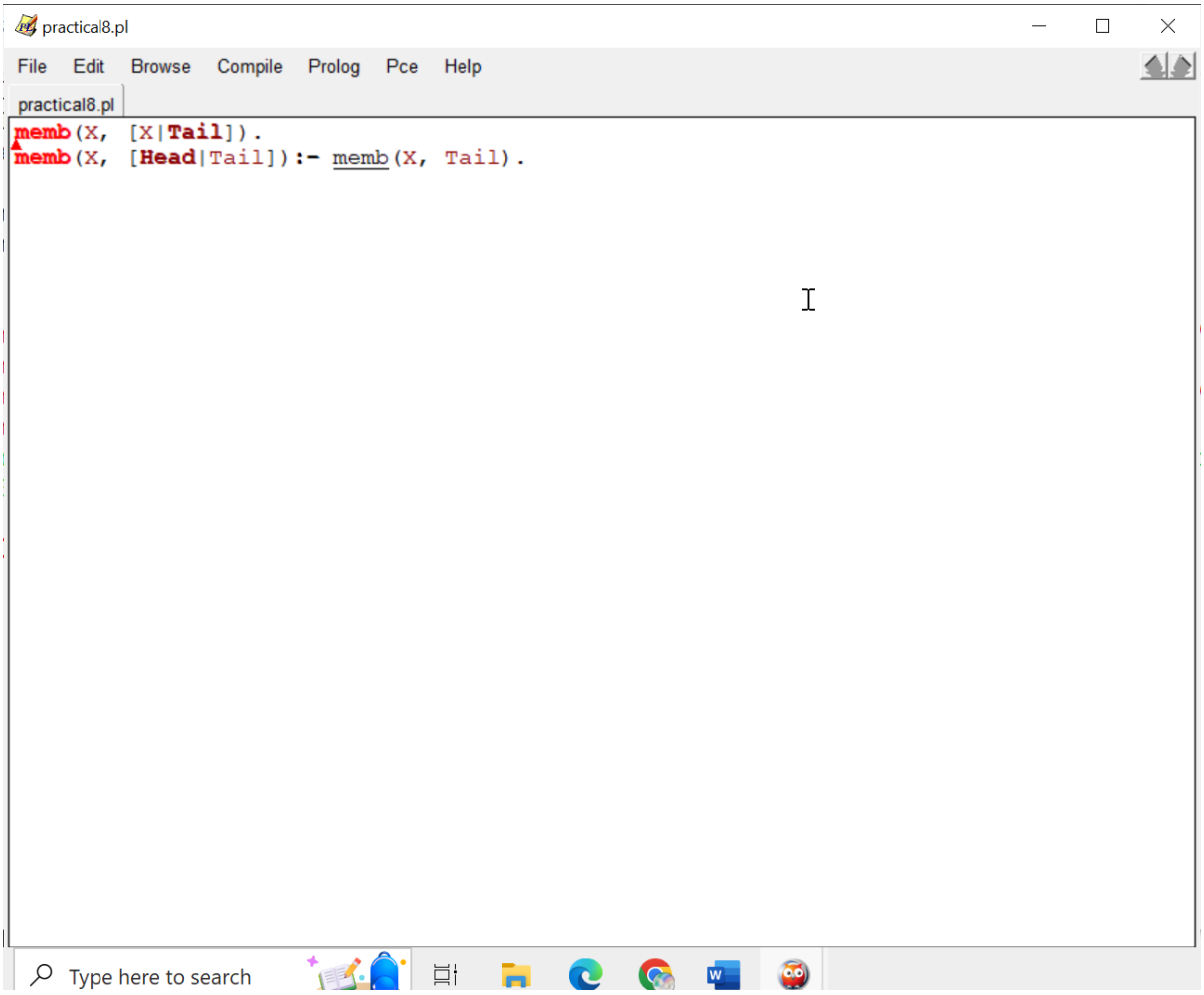
7. Prolog program to implement multi (N1, N2, R) : where N1 and N2 denotes the numbers to be multiplied and R represents the result.



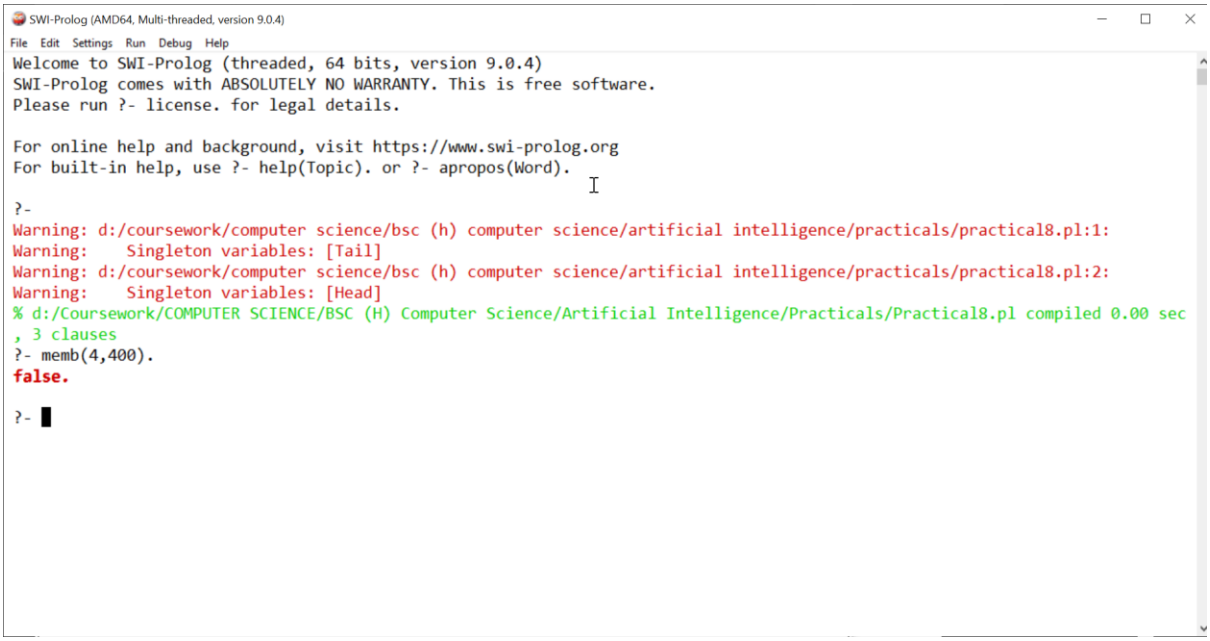
OUTPUT



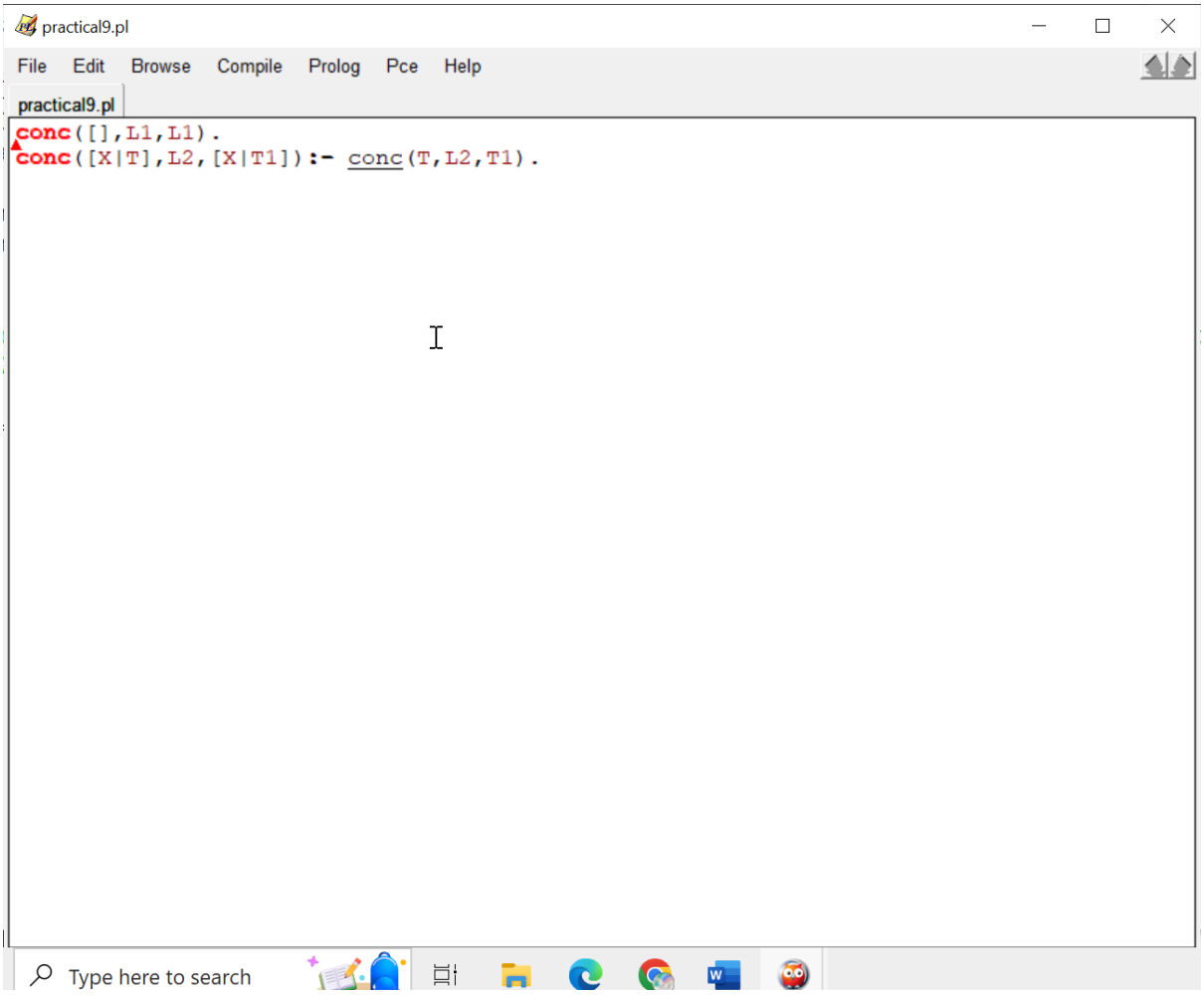
8. Write a Prolog program to implement memb(X, L): to check whether X is a member of L or not



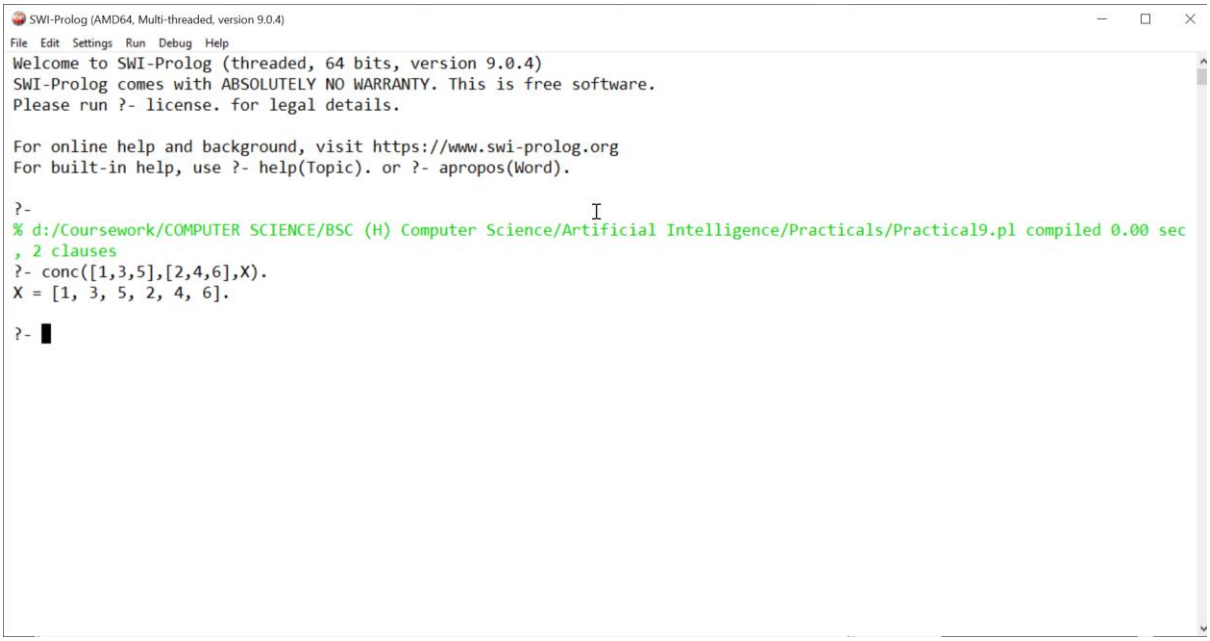
OUTPUT



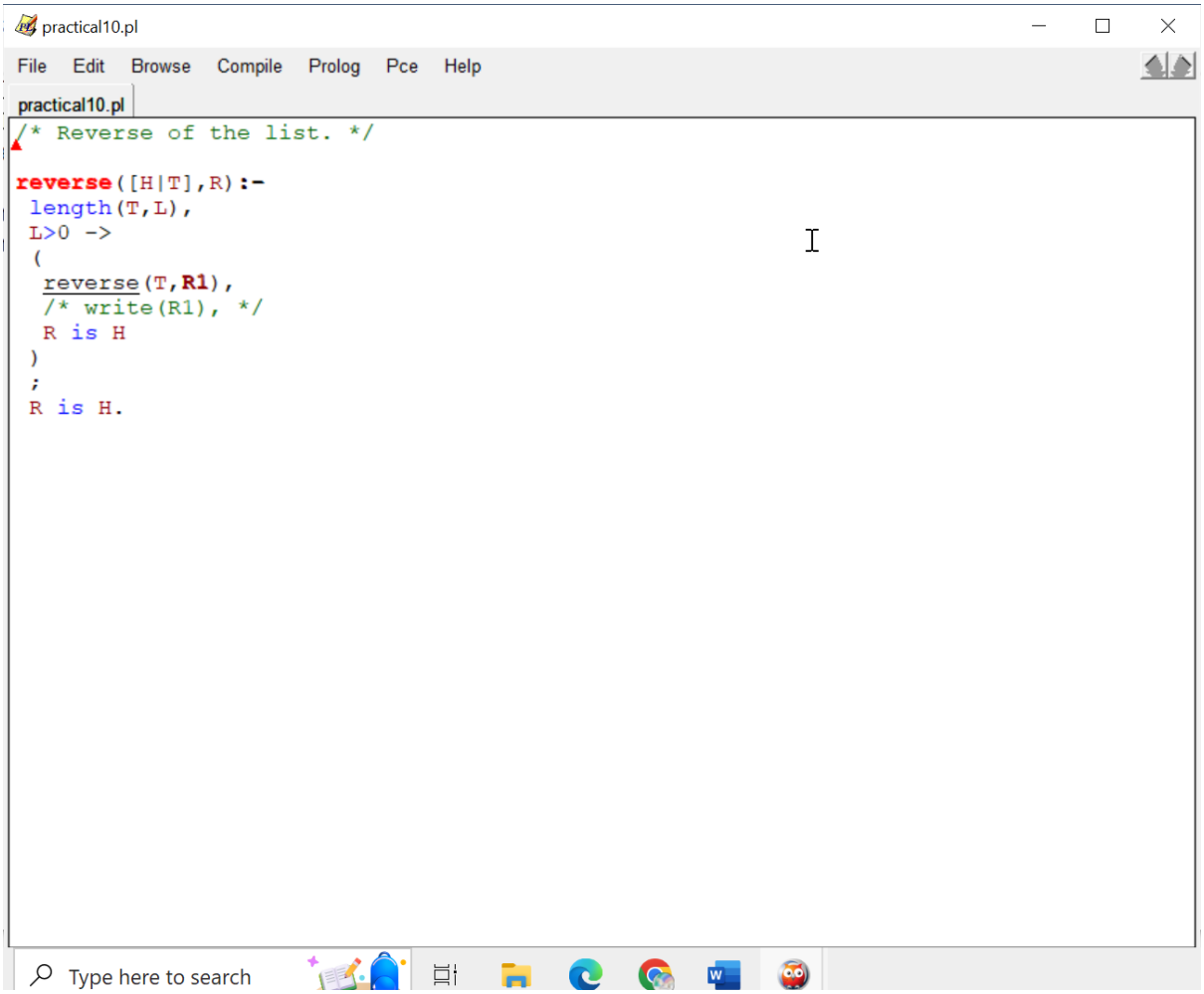
9. Write a Prolog program to implement conc (L1, L2, L3) where L2 is the list to be appended with L1 to get the resulted list L3



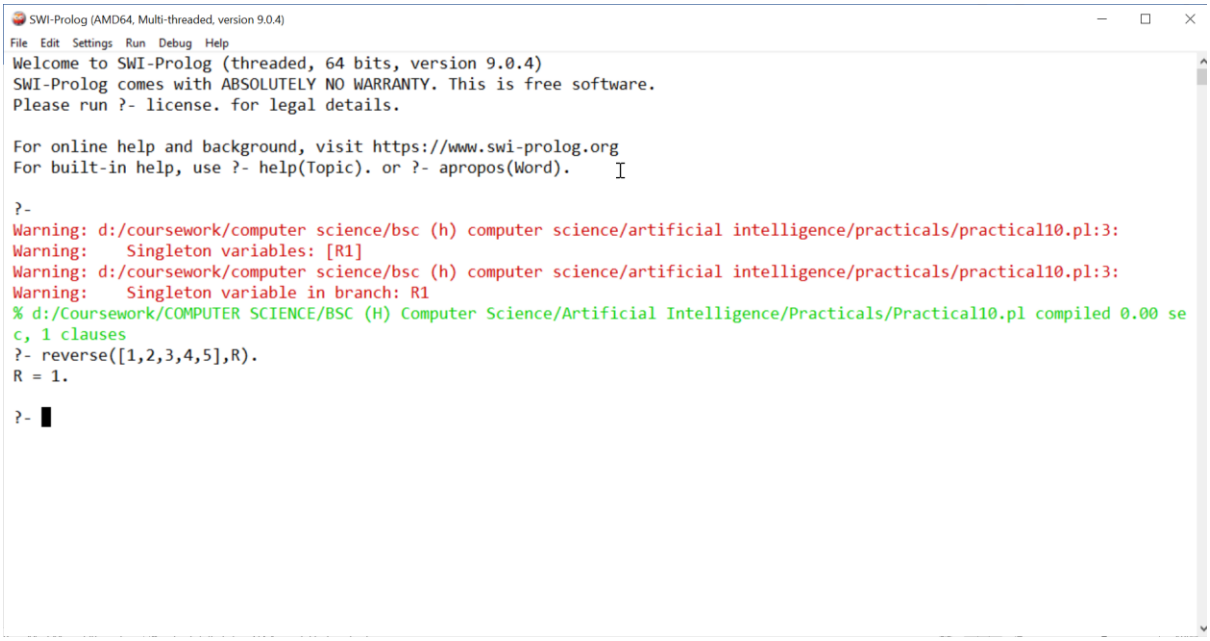
OUTPUT



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



OUTPUT



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

Practical11.pl

File Edit Browse Compile Prolog Pce Help

Practical11.pl

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

Type here to search

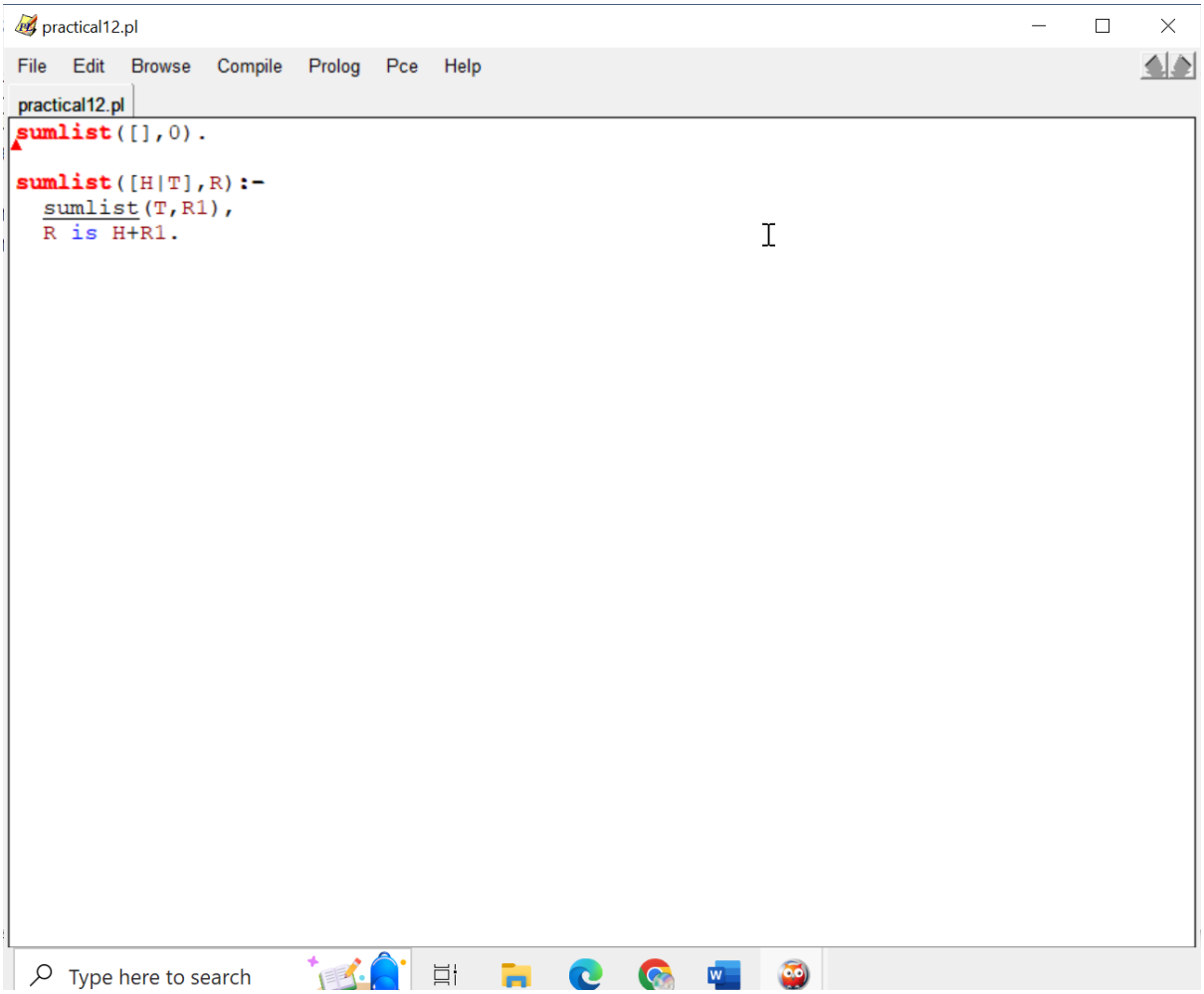
OUTPUT

SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)

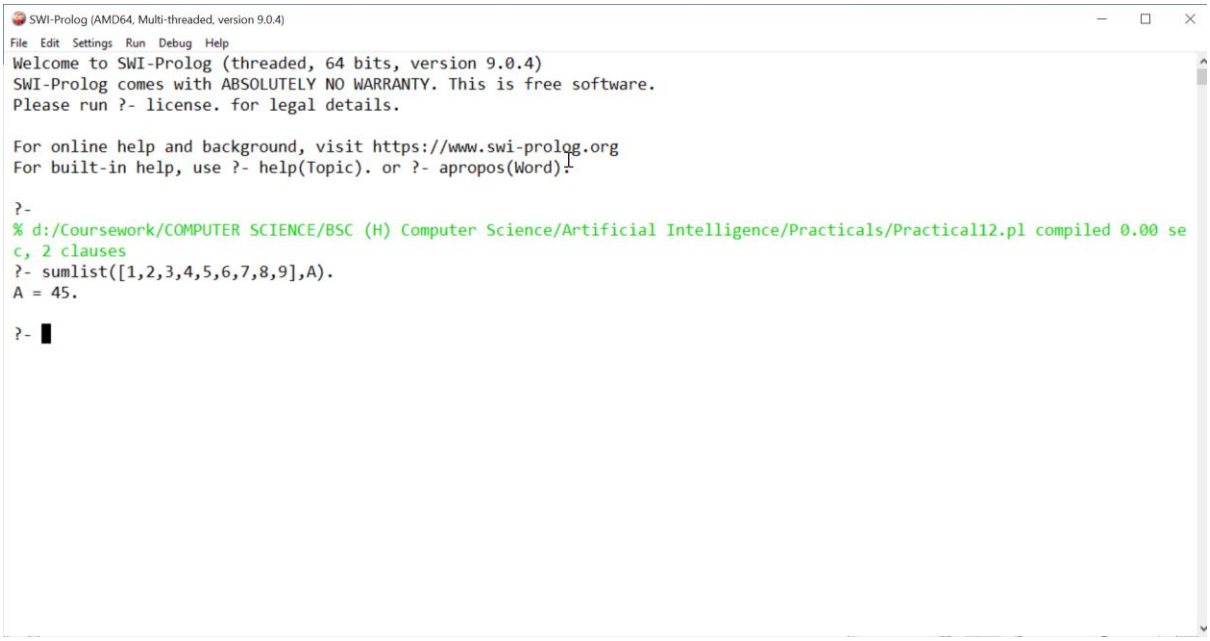
File Edit Settings Run Debug Help

```
Welcome to SWI-Prolog (threaded, 64 bits, version 9.0.4)
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).

?-
% d:/Coursework/COMPUTER SCIENCE/BSC (H) Computer Science/Artificial Intelligence/Practicals/Practical11.pl compiled 0.00 se
c, 3 clauses
?- palind([n,i,t,i,n]).
palindrome
true
```

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



OUTPUT



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.

practical13.pl

File Edit Browse Compile Prolog Pce Help

practical13.pl

```
evenlength:-
 write('true --> even').
oddlength:-
 write('true --> odd').

oddeven([H|T]):-
 length(T,L),
 L>=0 ->
 (
  L1 is L+1,
  L2 is mod(L1,2),
  L2=:=0 ->
   evenlength
  ;
   oddlength
 ).
```

Type here to search

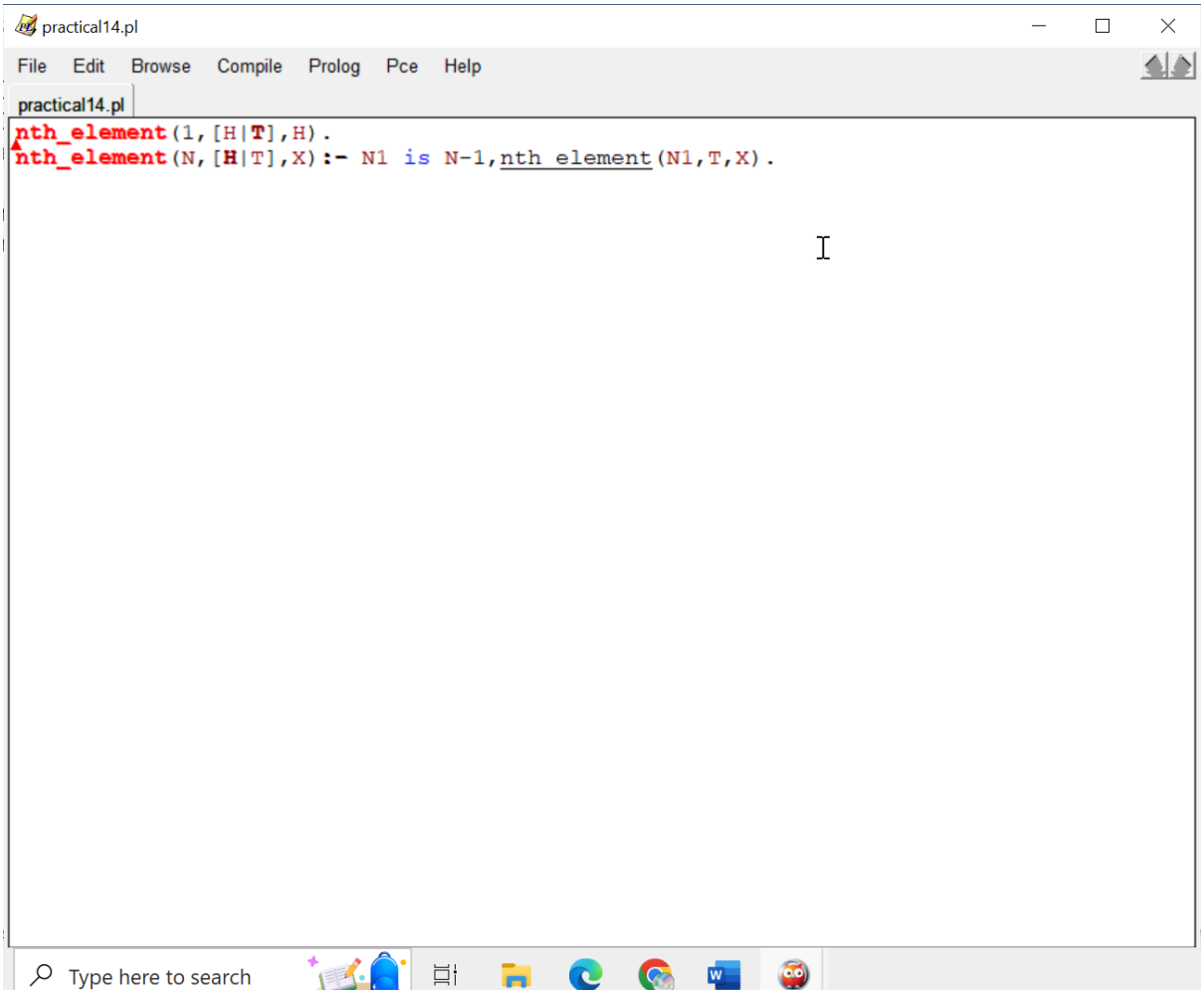
OUTPUT

SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)

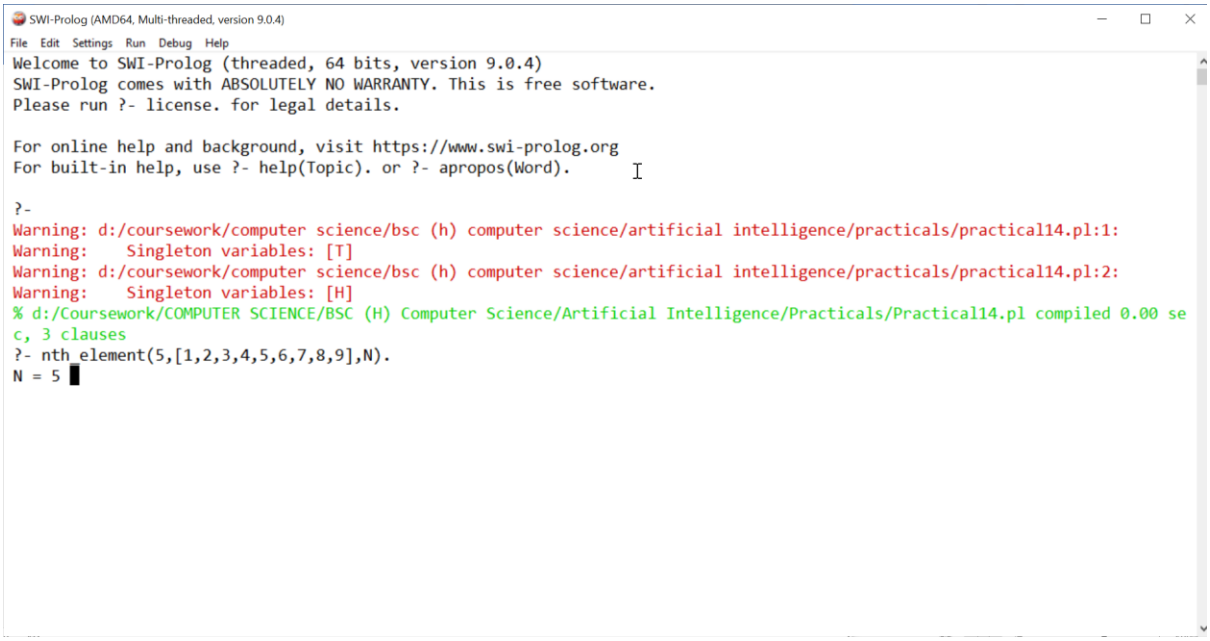
File Edit Settings Run Debug Help

```
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).

?-
Warning: d:/coursework/computer science/bsc (h) computer science/artificial intelligence/practicals/practical13.pl:6:
Warning:    Singleton variables: [H]
% d:/Coursework/COMPUTER SCIENCE/BSC (H) Computer Science/Artificial Intelligence/Practicals/Practical13.pl compiled 0.02 se
c, 4 clauses
?- oddeven([1,2,3,4,5,6,7],A).
ERROR: Unknown procedure: oddeven/2
ERROR:     However, there are definitions for:
ERROR:         oddeven/1
false.

?- oddeven(1,2,3,4,5,6,7).
ERROR: Unknown procedure: oddeven/7
ERROR:     However, there are definitions for:
ERROR:         oddeven/1
false.

?- oddeven([1,2,3,4,5,6,7]).
true --> odd
true.

?-
```

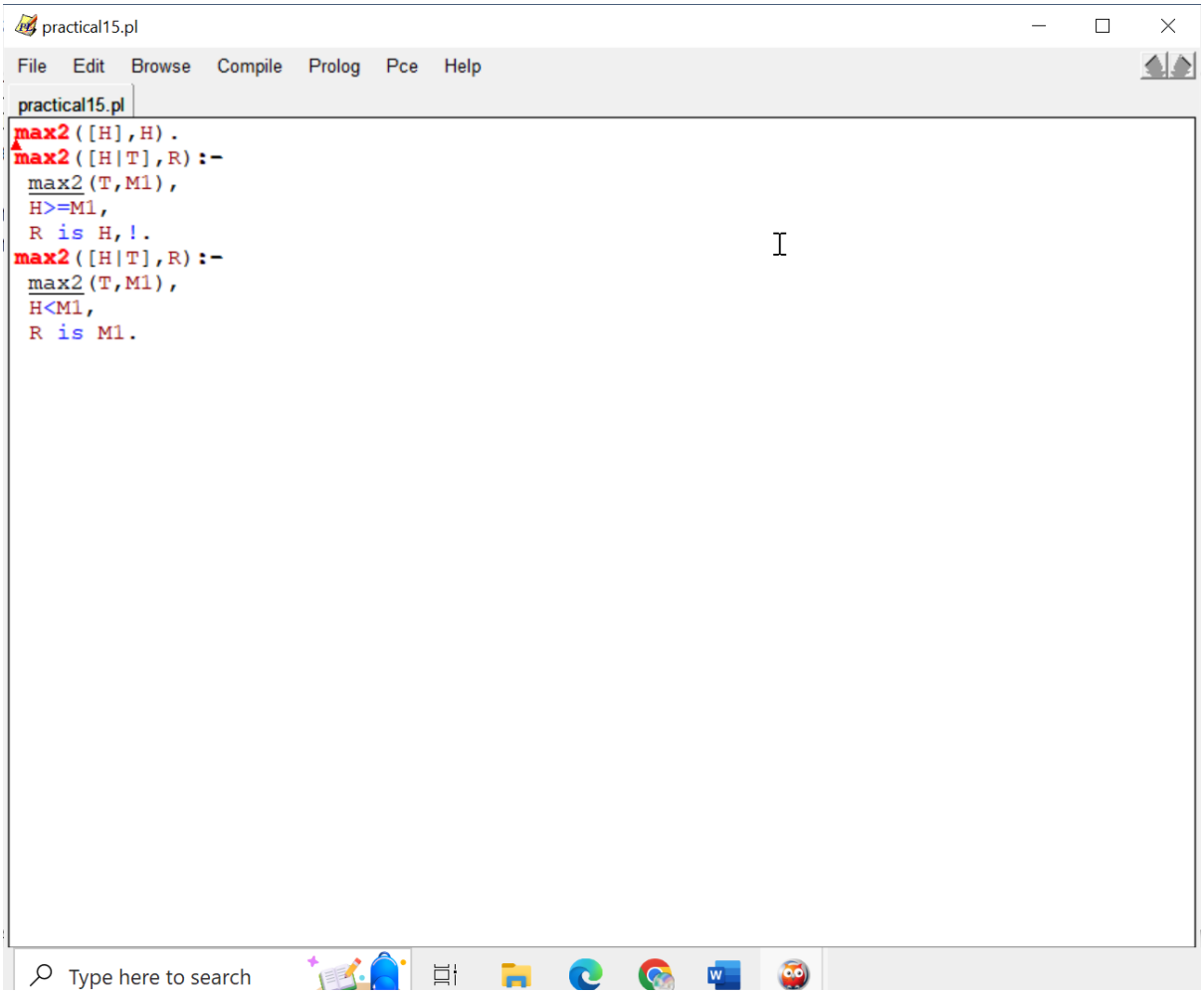
14. Write a Prolog program to implement nth_element (N, L, X) where N is the desired position, L is a list and X represents the Nth element of L.



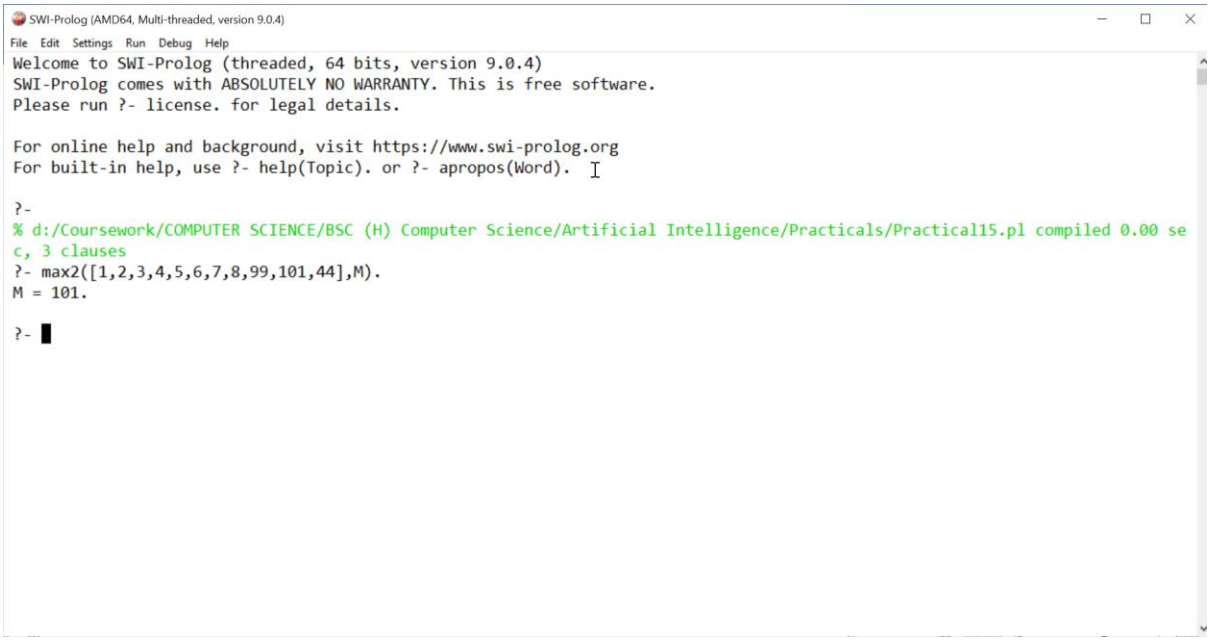
OUTPUT



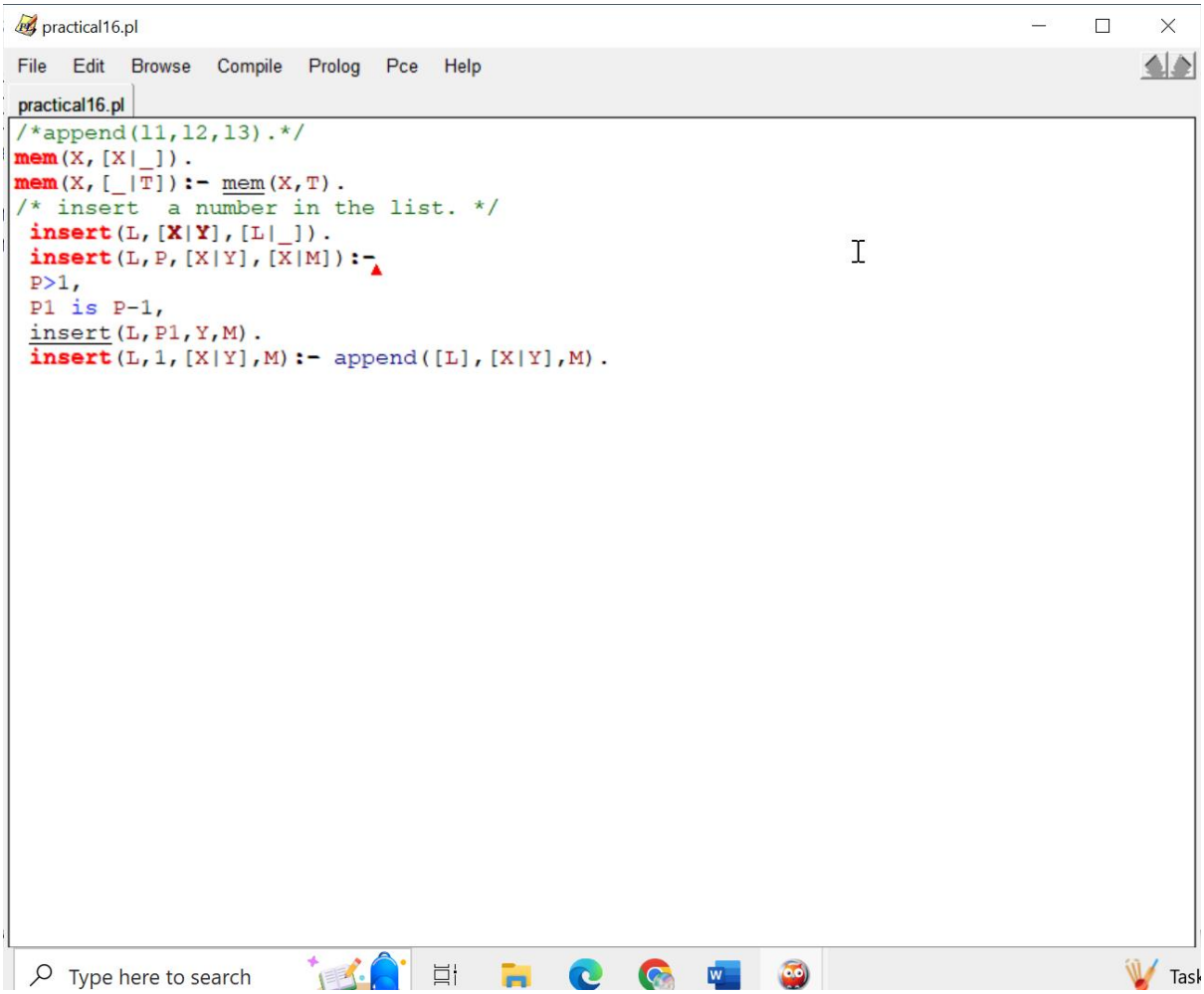
15. Write a Prolog program to implement maxlist(L, M) so that M is the maximum number in the list



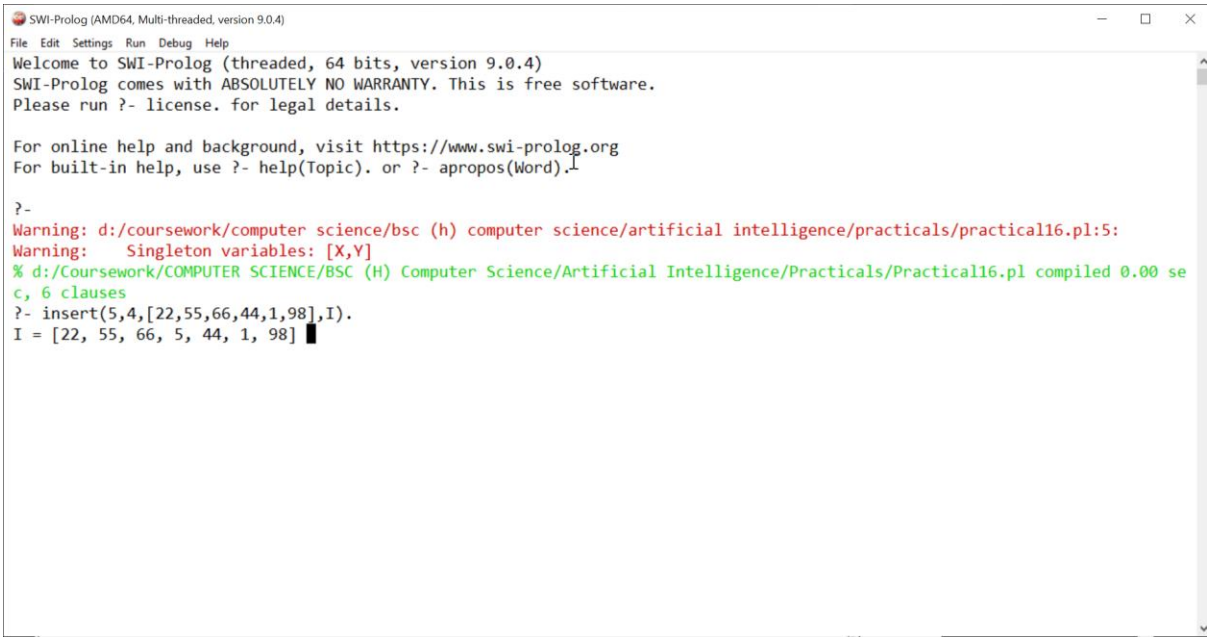
OUTPUT



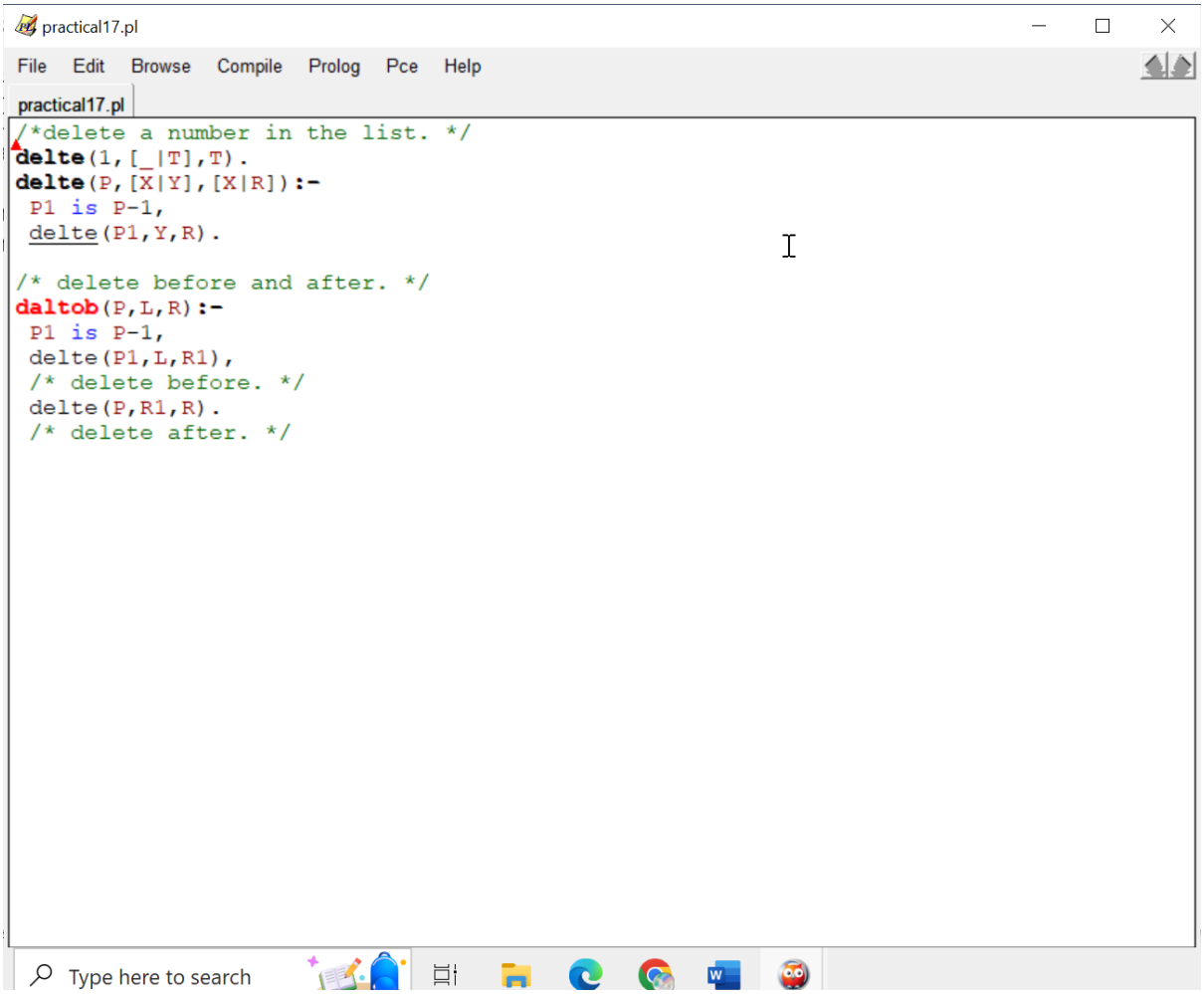
16. Write a prolog program to implement insert_nth (I, N, L, R) that inserts an item I into Nth position of list L to generate a list R



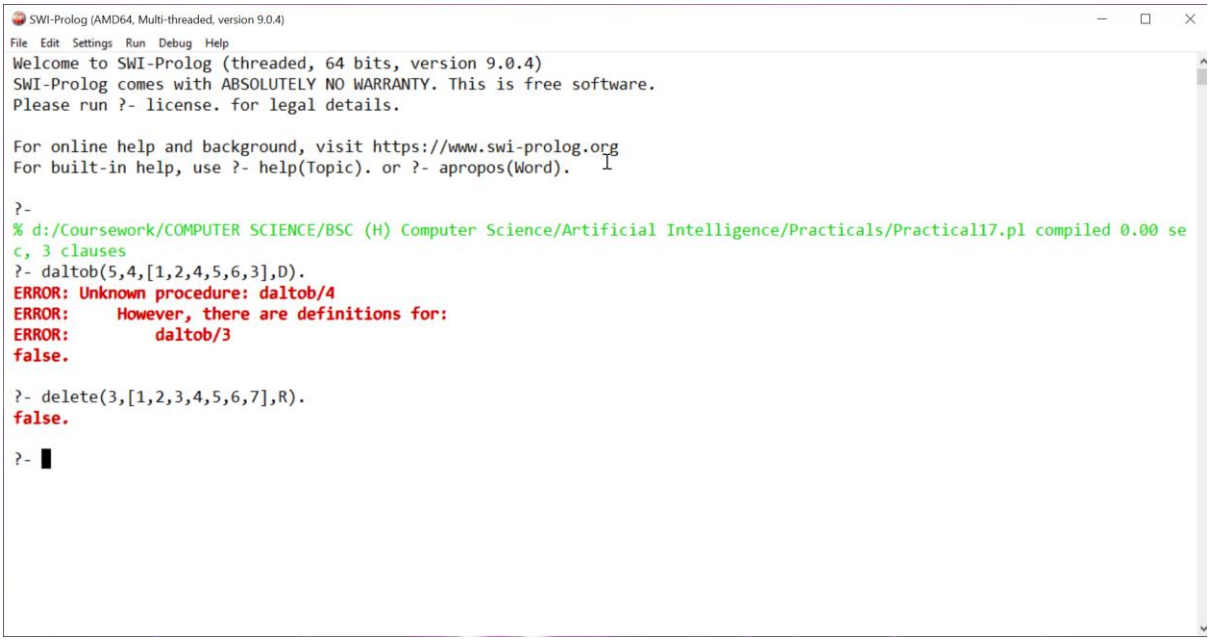
OUTPUT



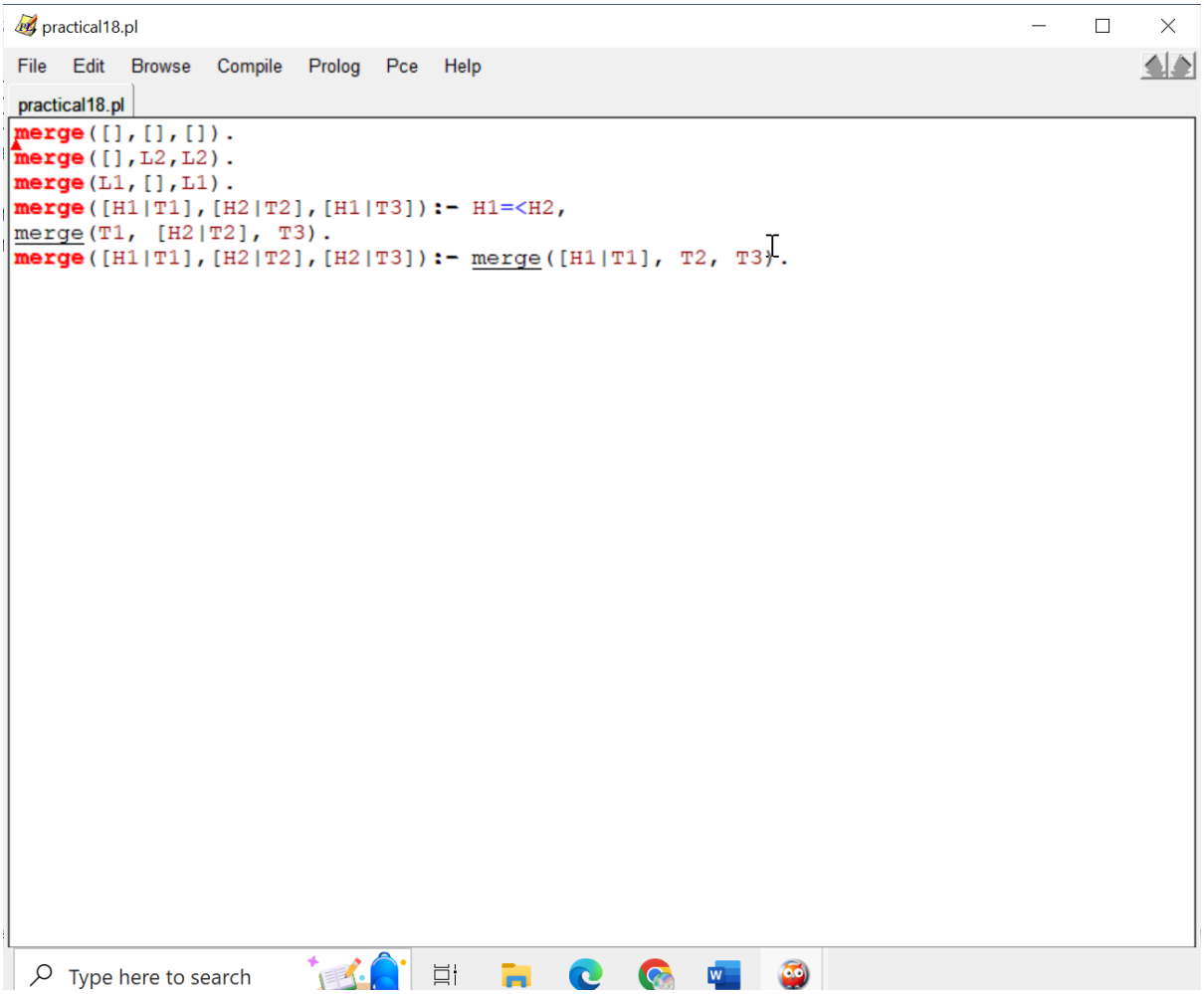
17. write a Prolog program to implement delete_nth (N, L, R) that removes the element on Nth position from a list L to generate a list R.



OUTPUT



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



OUTPUT