ARTIFICIAL INTELLIGENCE PRACTICAL FILE

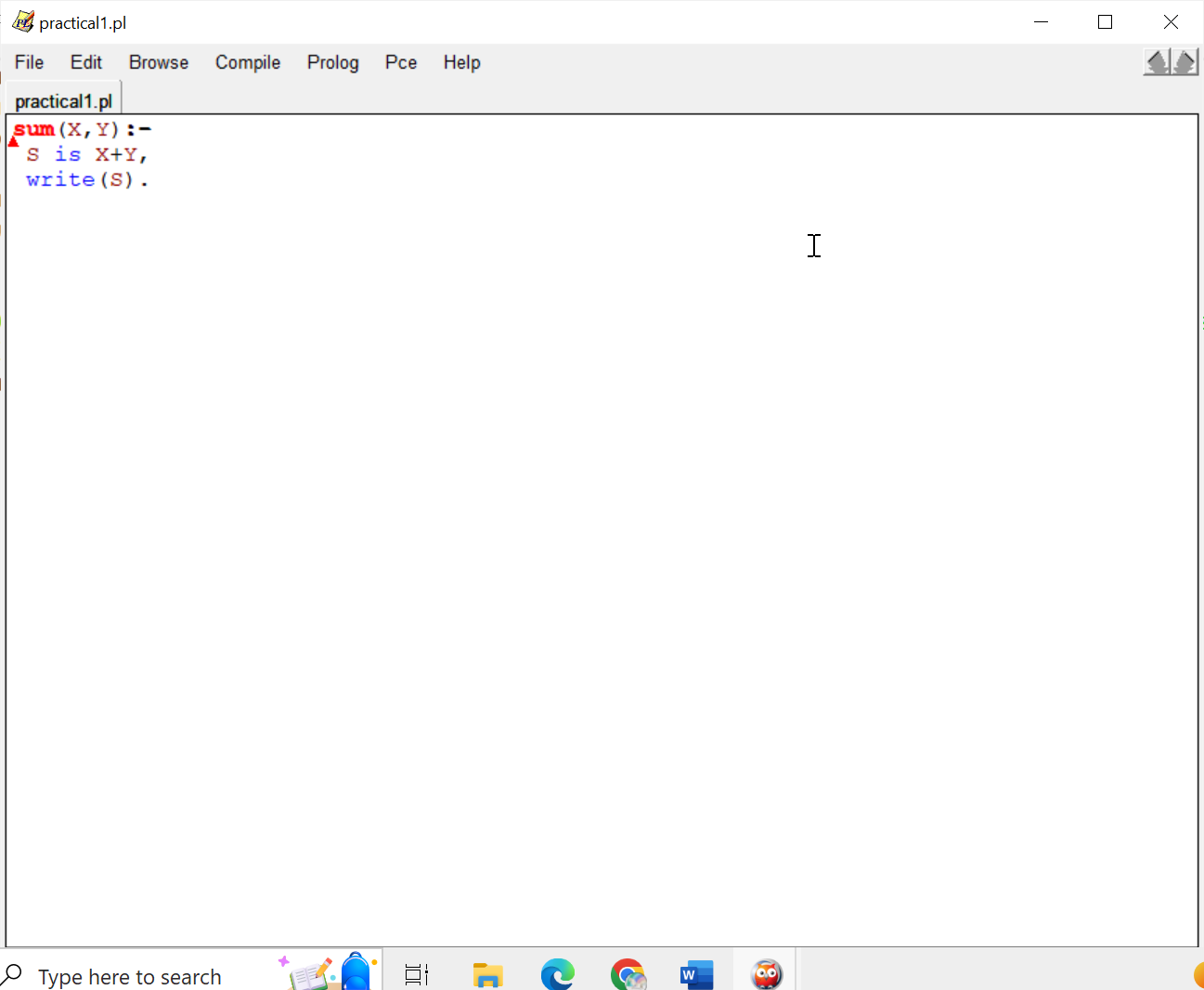
AADITYA KEDIYAL

BSc(Hons) Computer Science

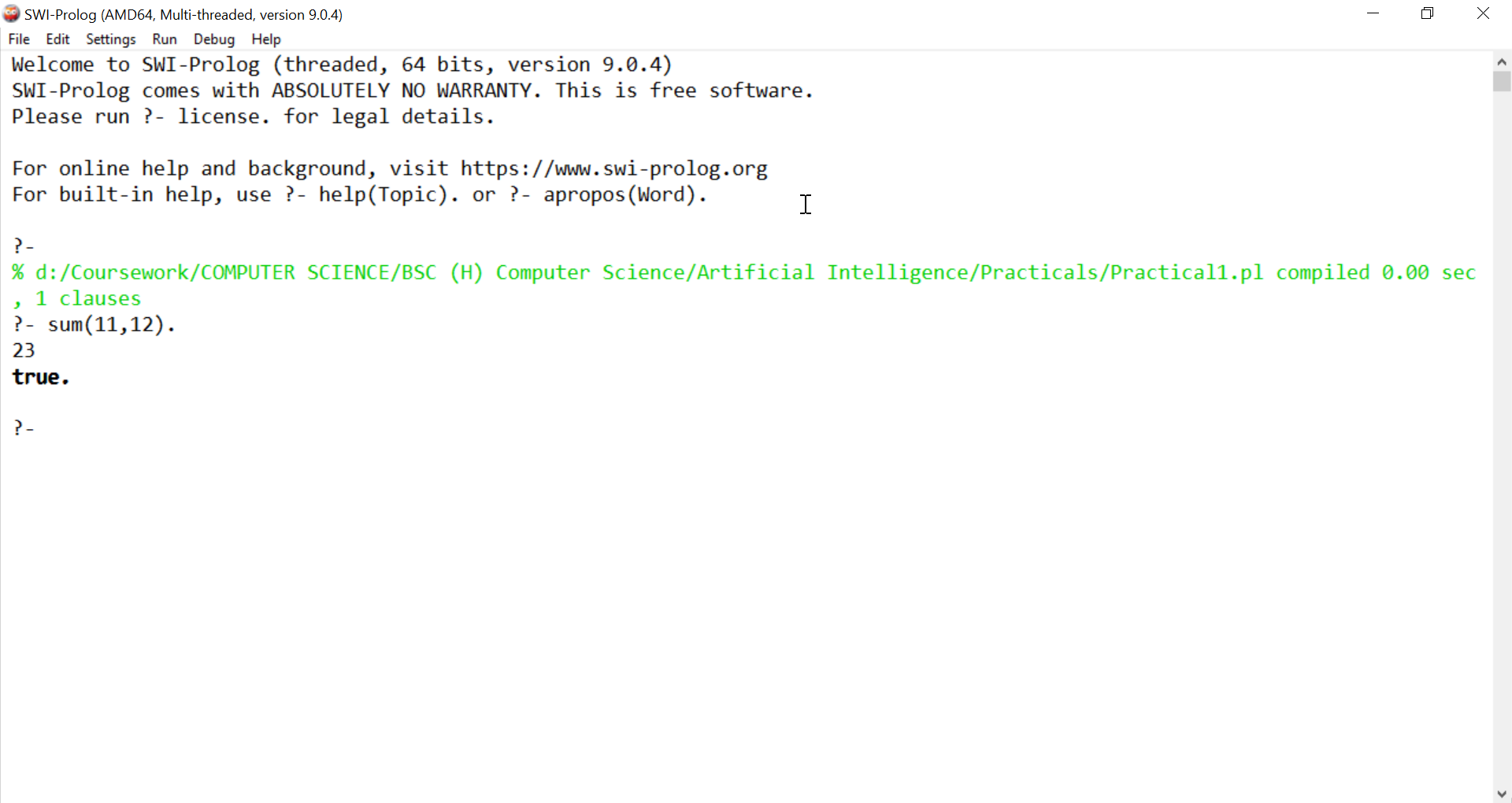
20201401

Ramanujan College

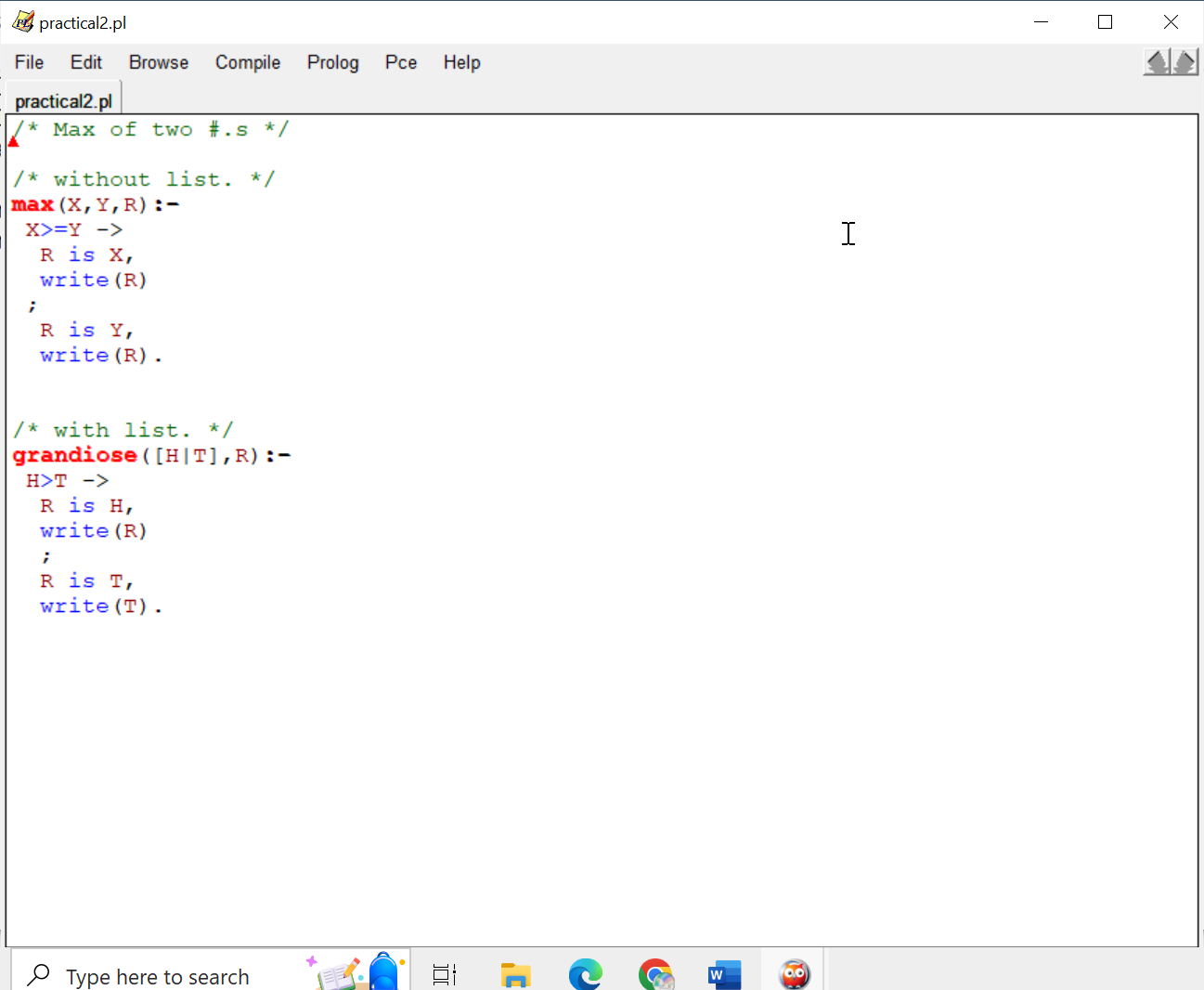
1. Write a prolog program to calculate the sum of two numbers



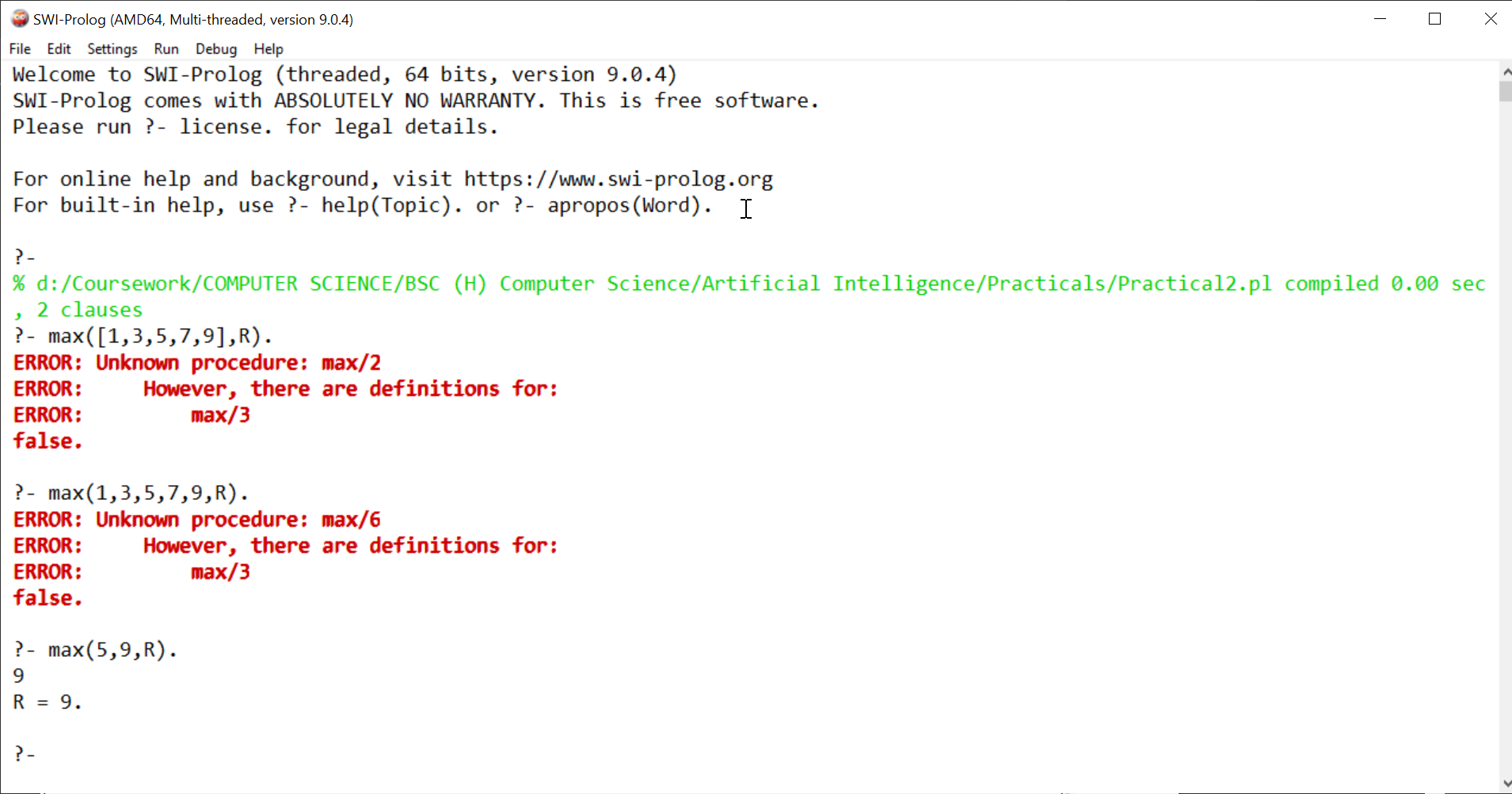
OUTPUT



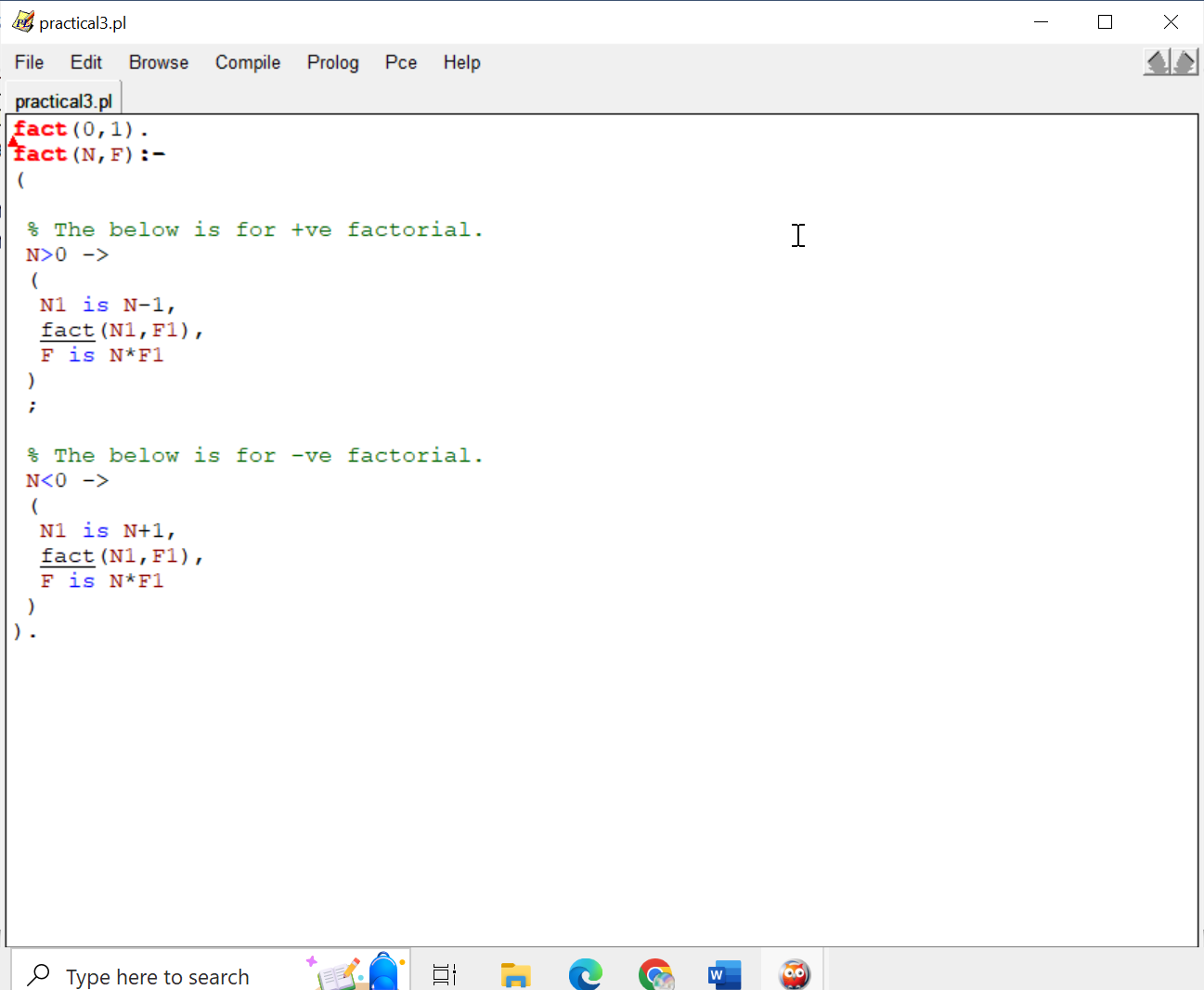
1. Write a Prolog program to implement max(X, Y, M) so that M is the maximum of two numbers X and Y.



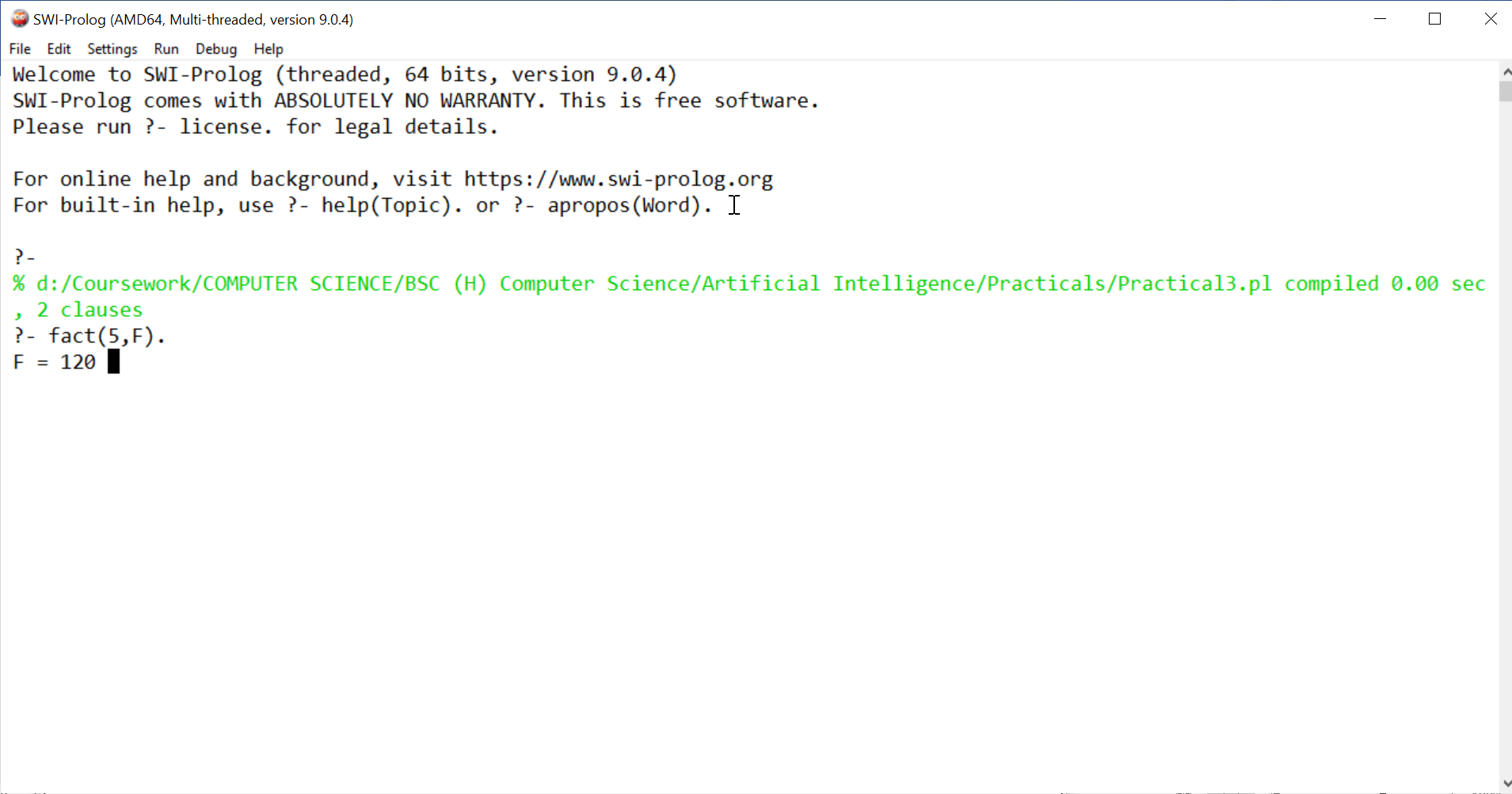
OUTPUT



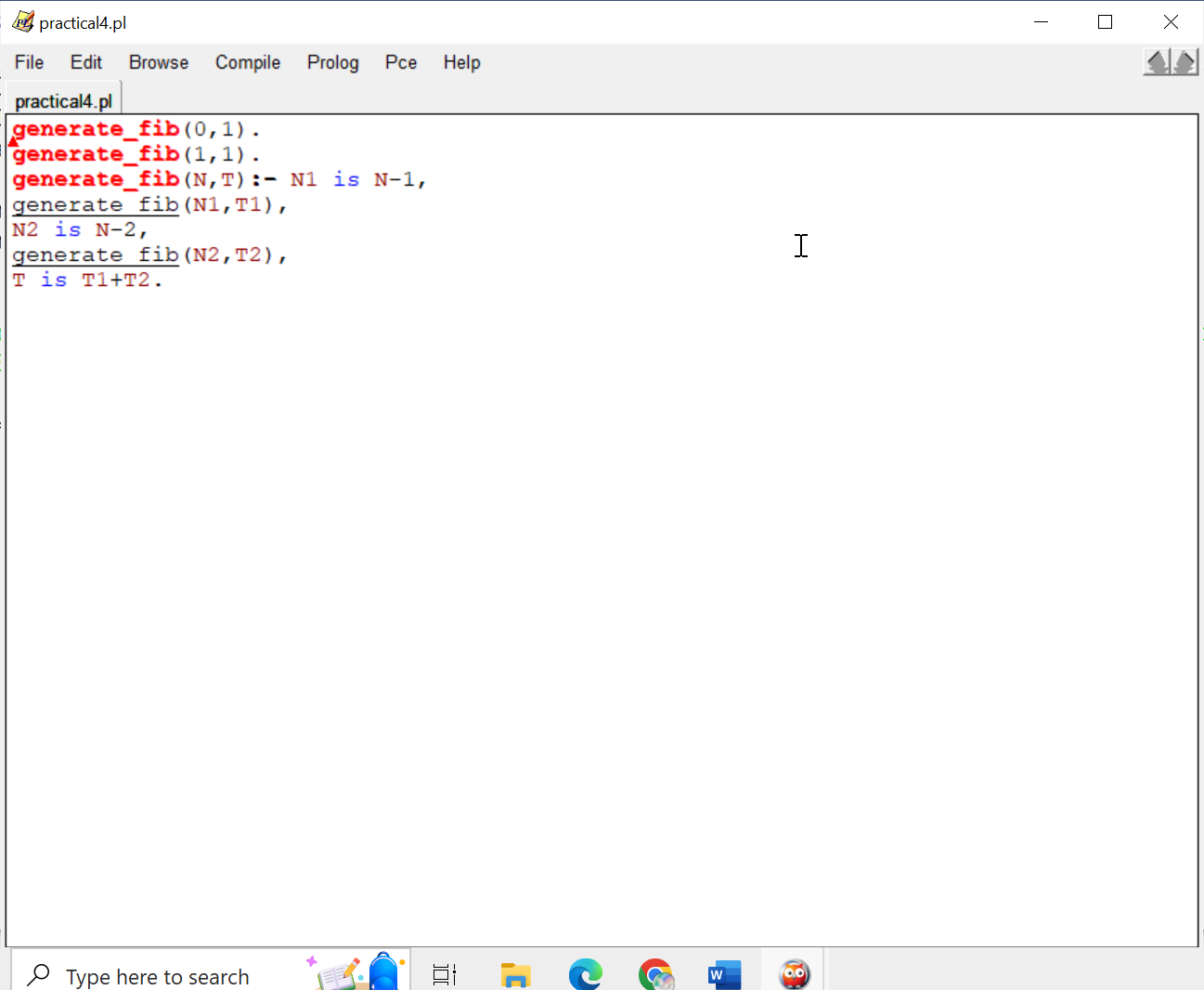
1. Write a program in PROLOG to implement factorial (N, F) where F represents the factorial of a number N.



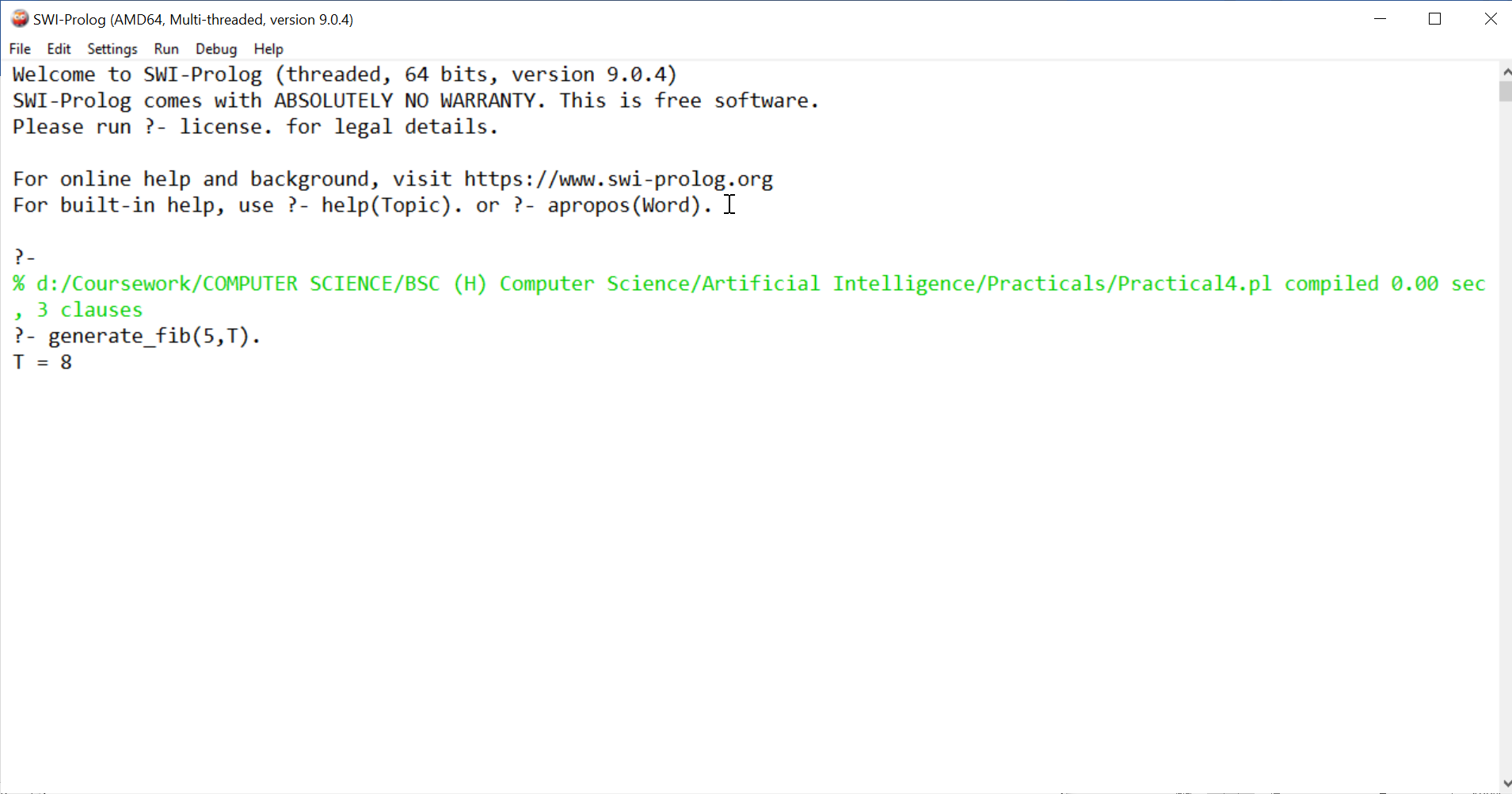
OUTPUT



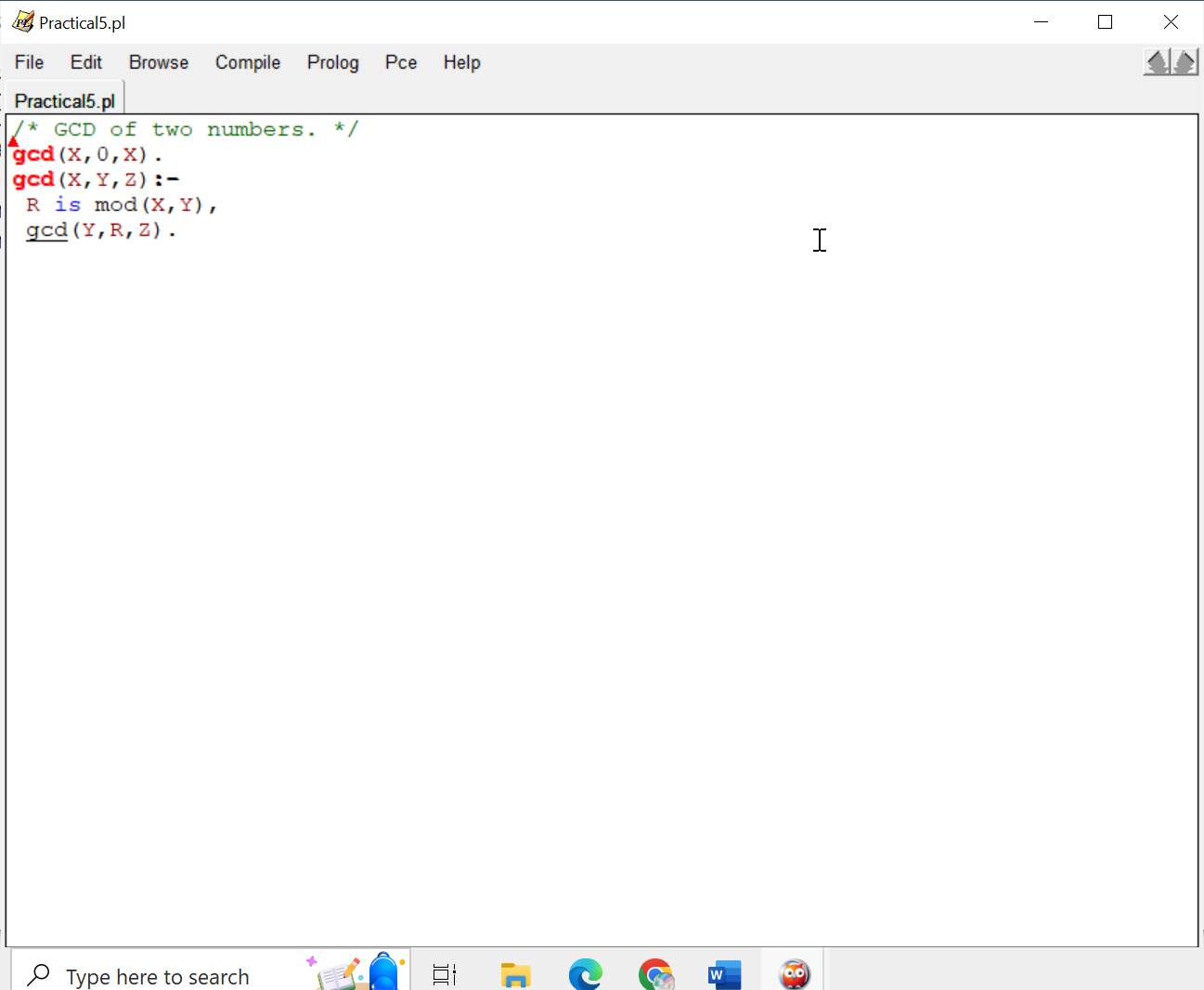
1. Write a program in PROLOG to implement generate\_fib(N,T) where T represents the Nth term of the fibonacci series



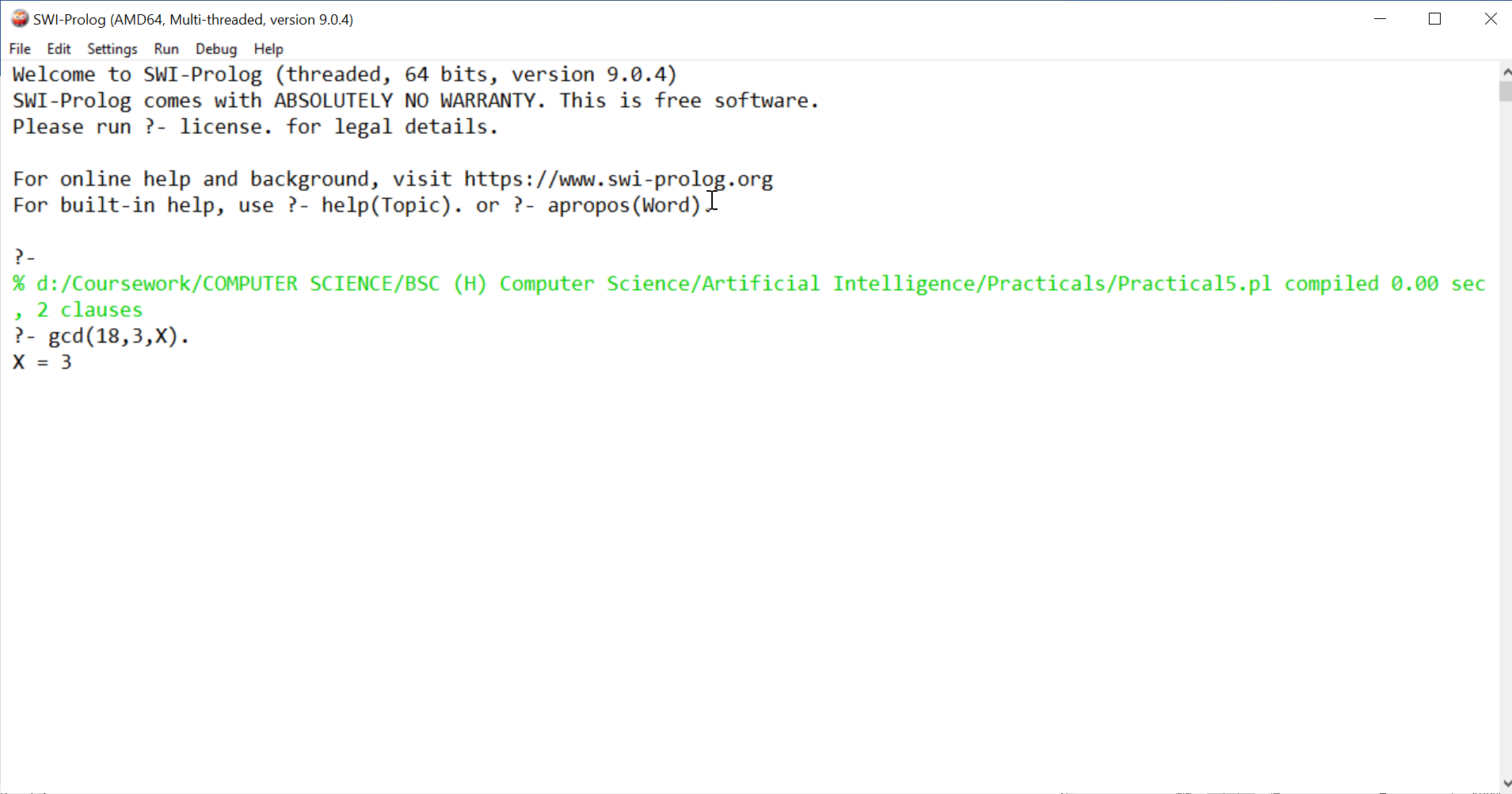
OUTPUT



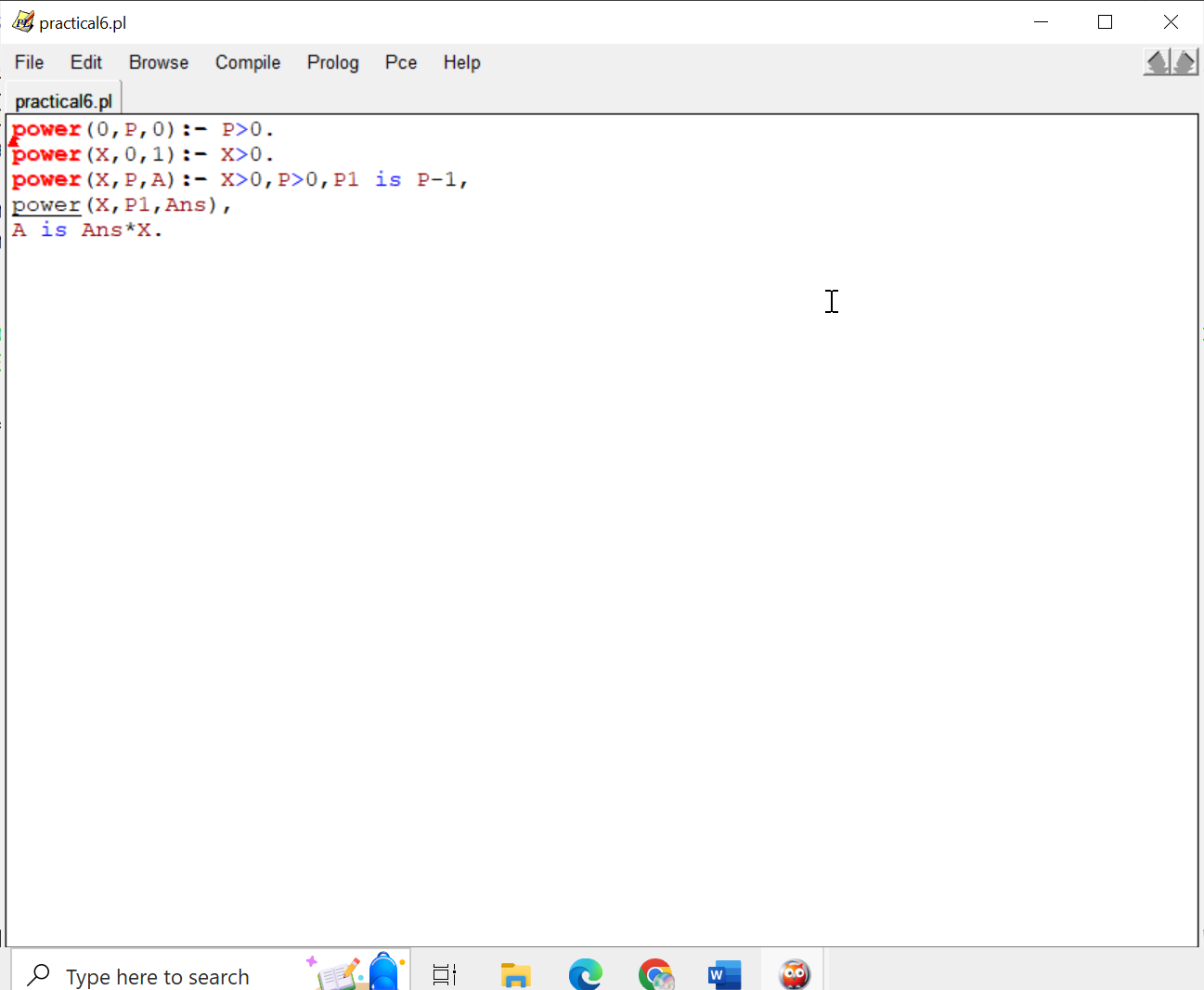
1. Write a Prolog program to implement GCD of two numbers.



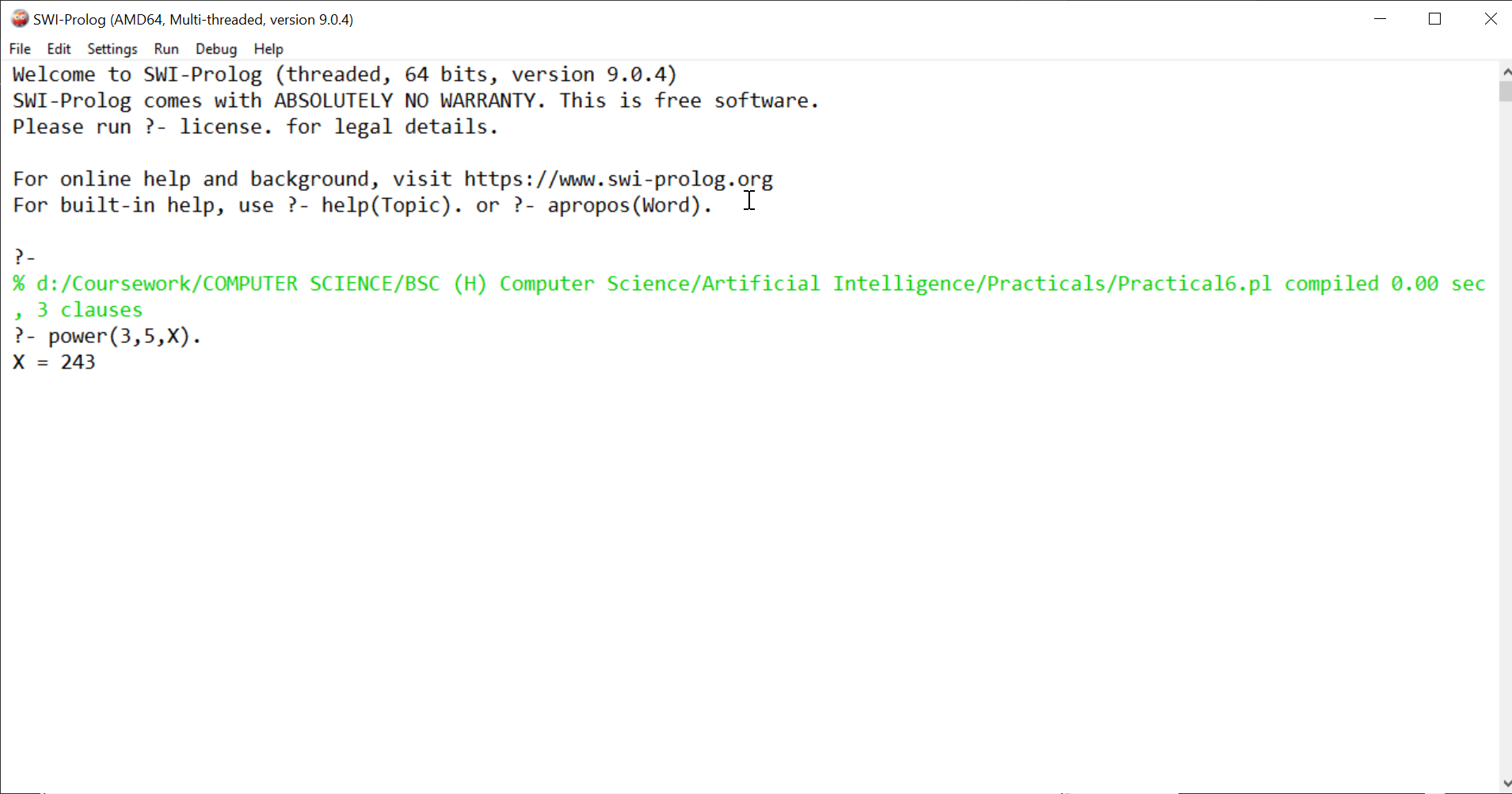
OUTPUT



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



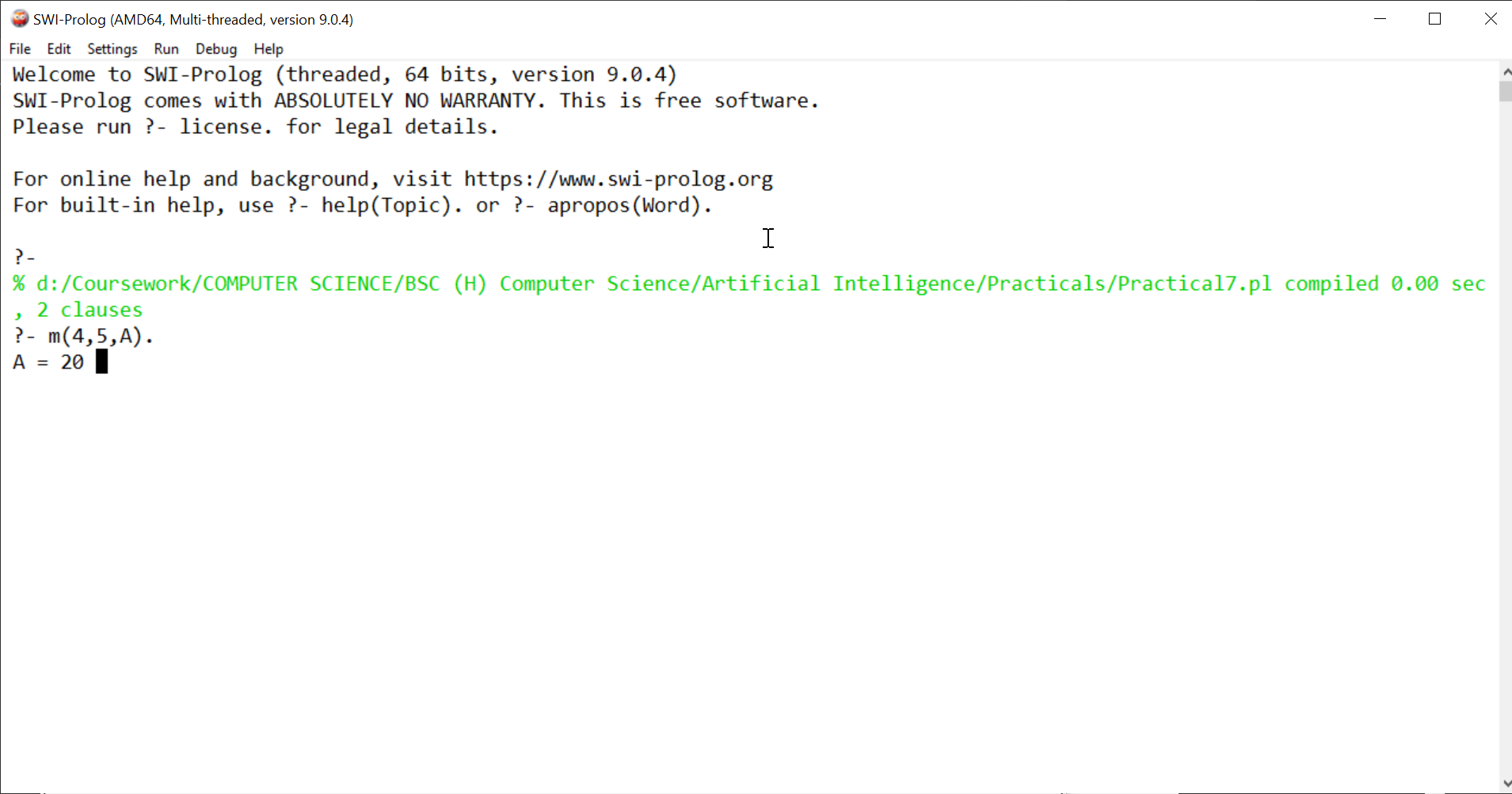
OUTPUT



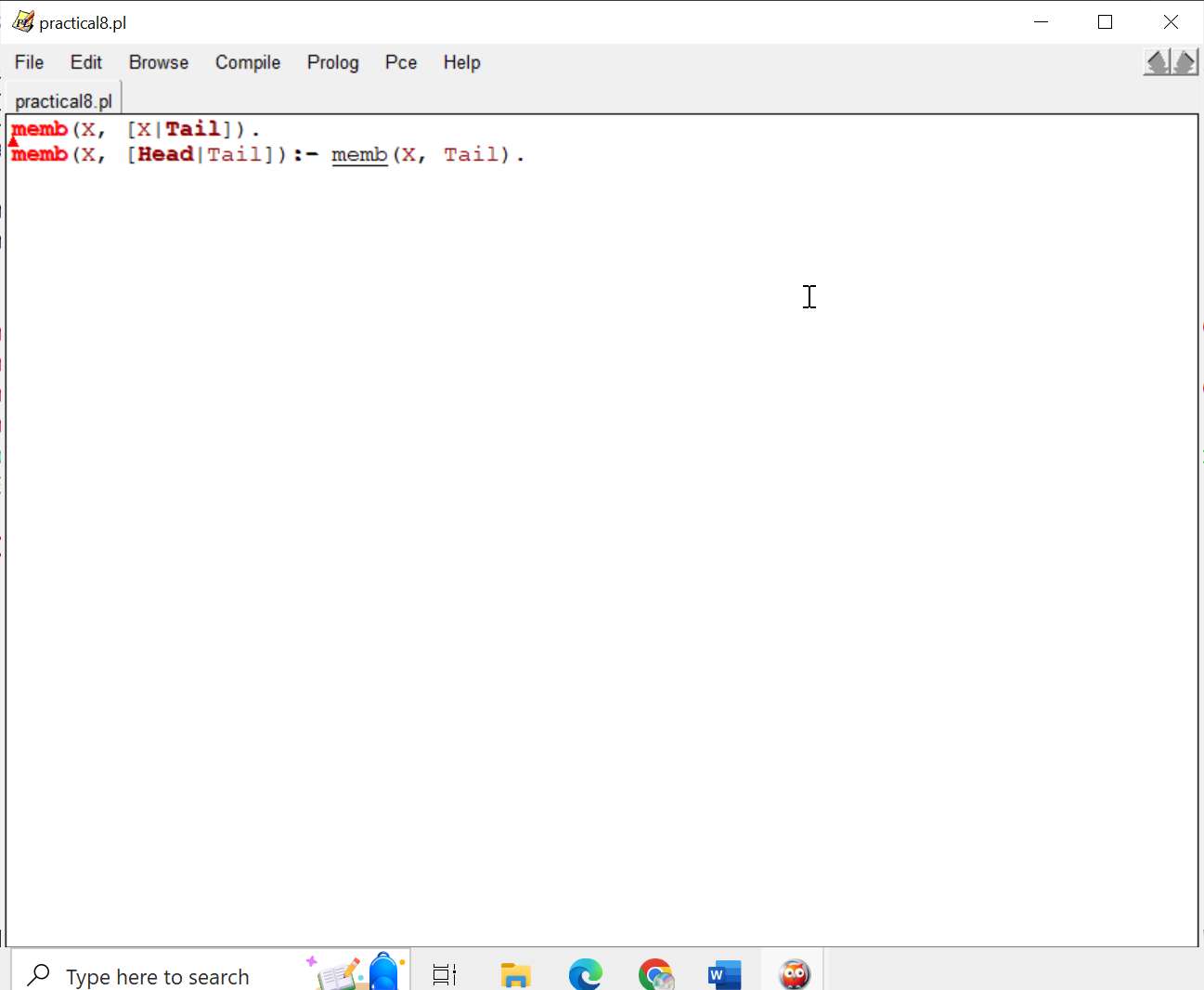
1. Prolog program to implement multi (N1, N2, R) : where N1 and N2 denotes the numbers to be multiplied and R represents the result.



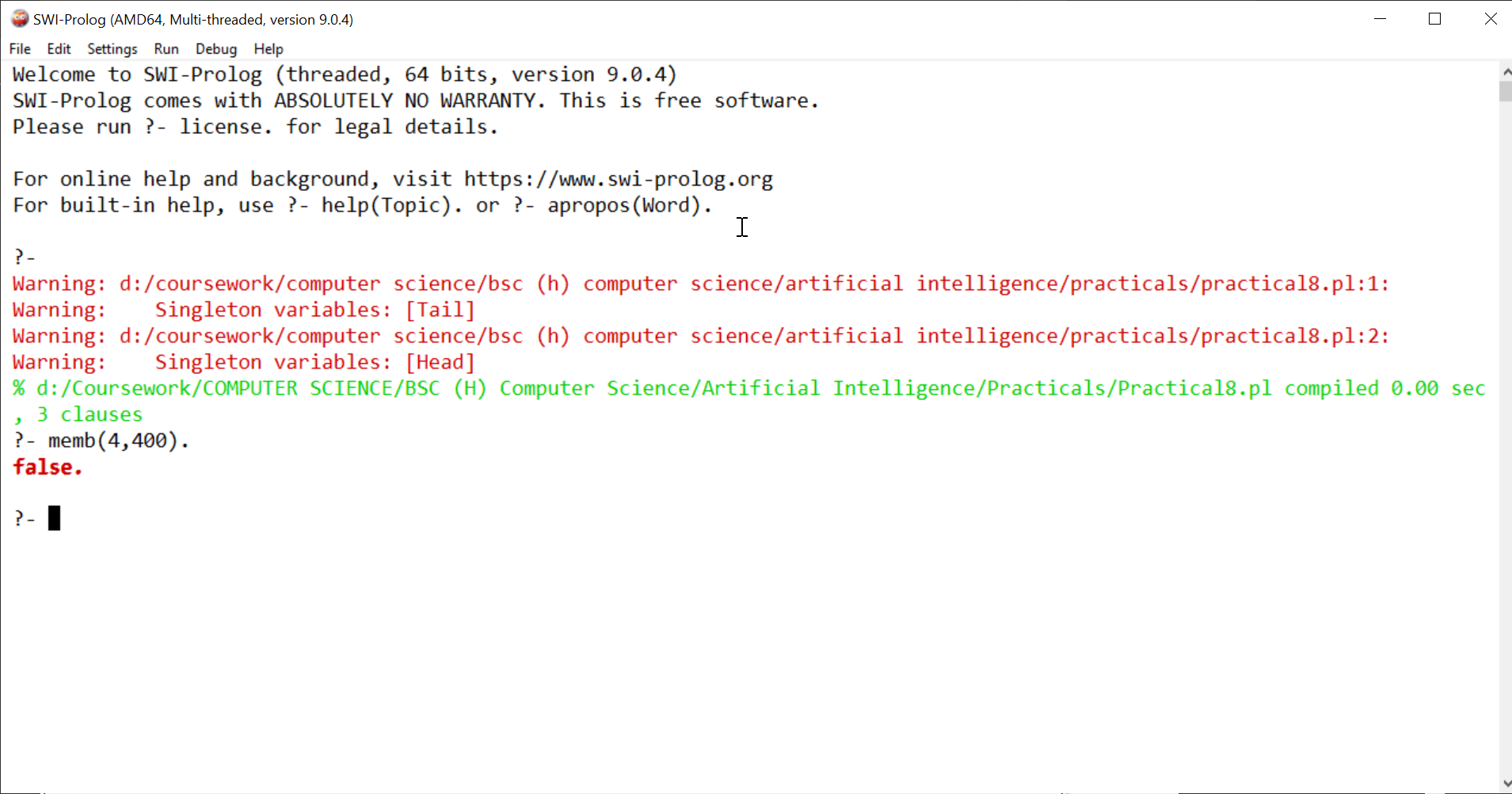
OUTPUT



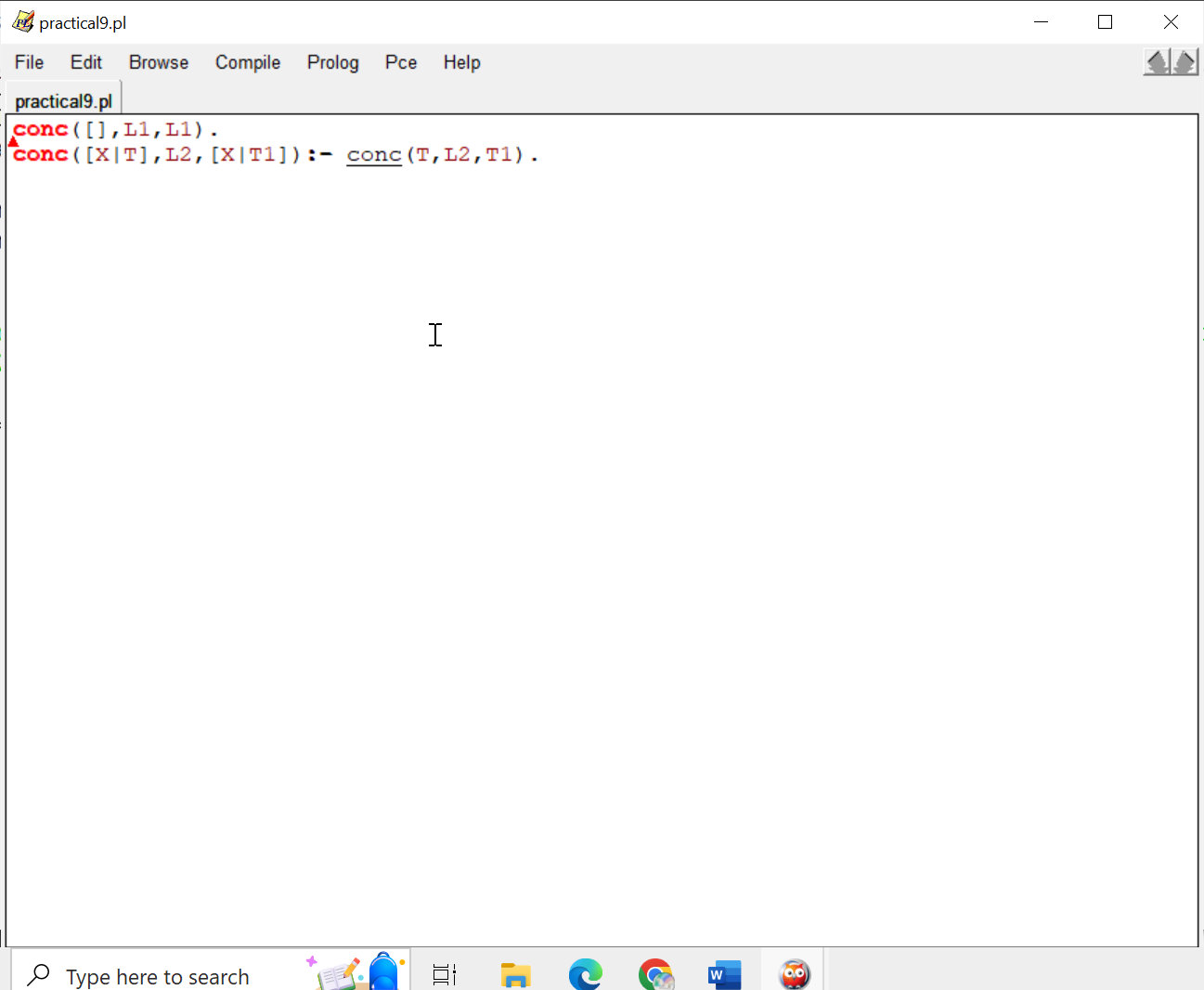
1. Write a Prolog program to implement memb(X, L): to check whether X is a member of L or not



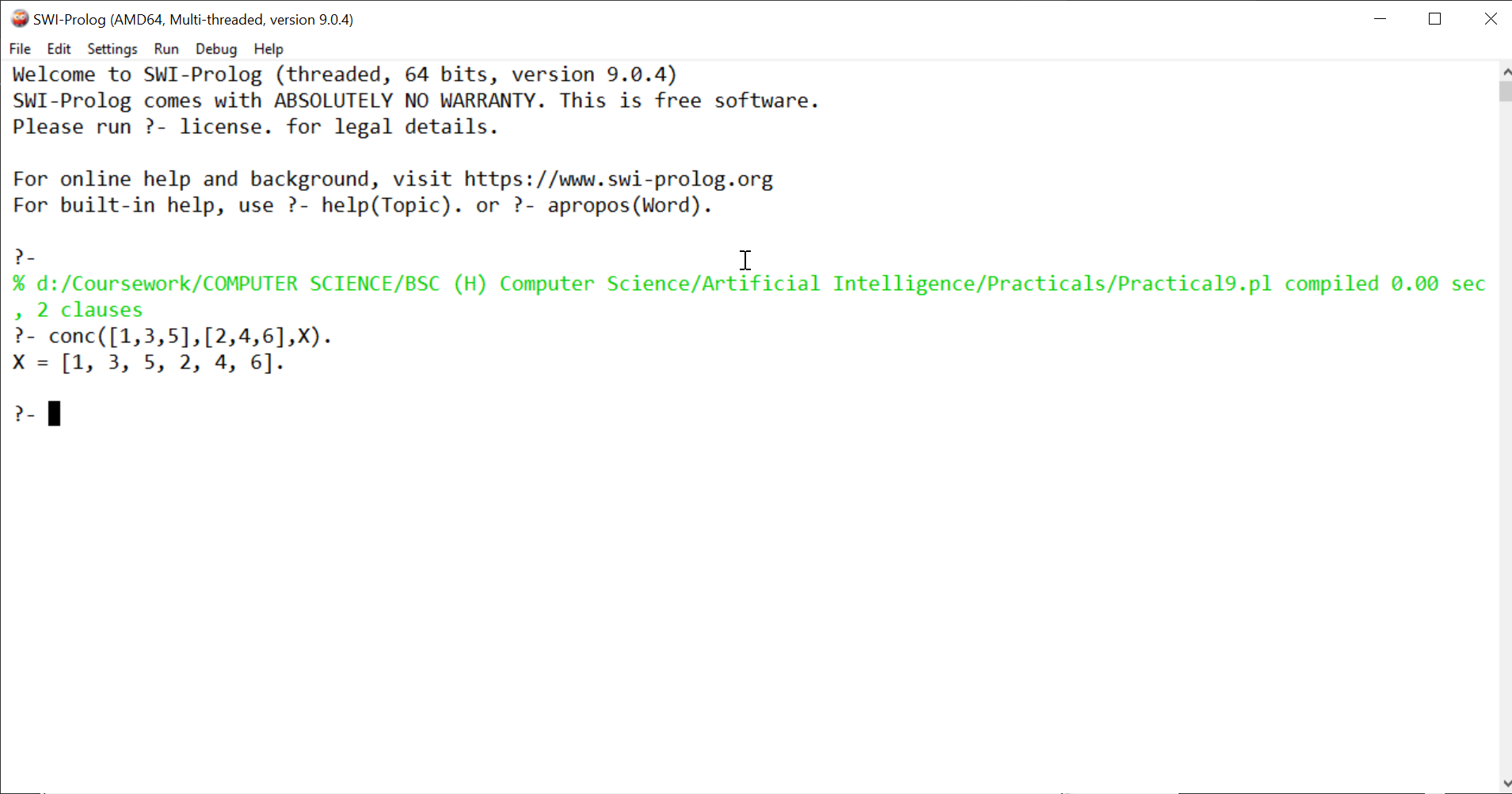
OUTPUT



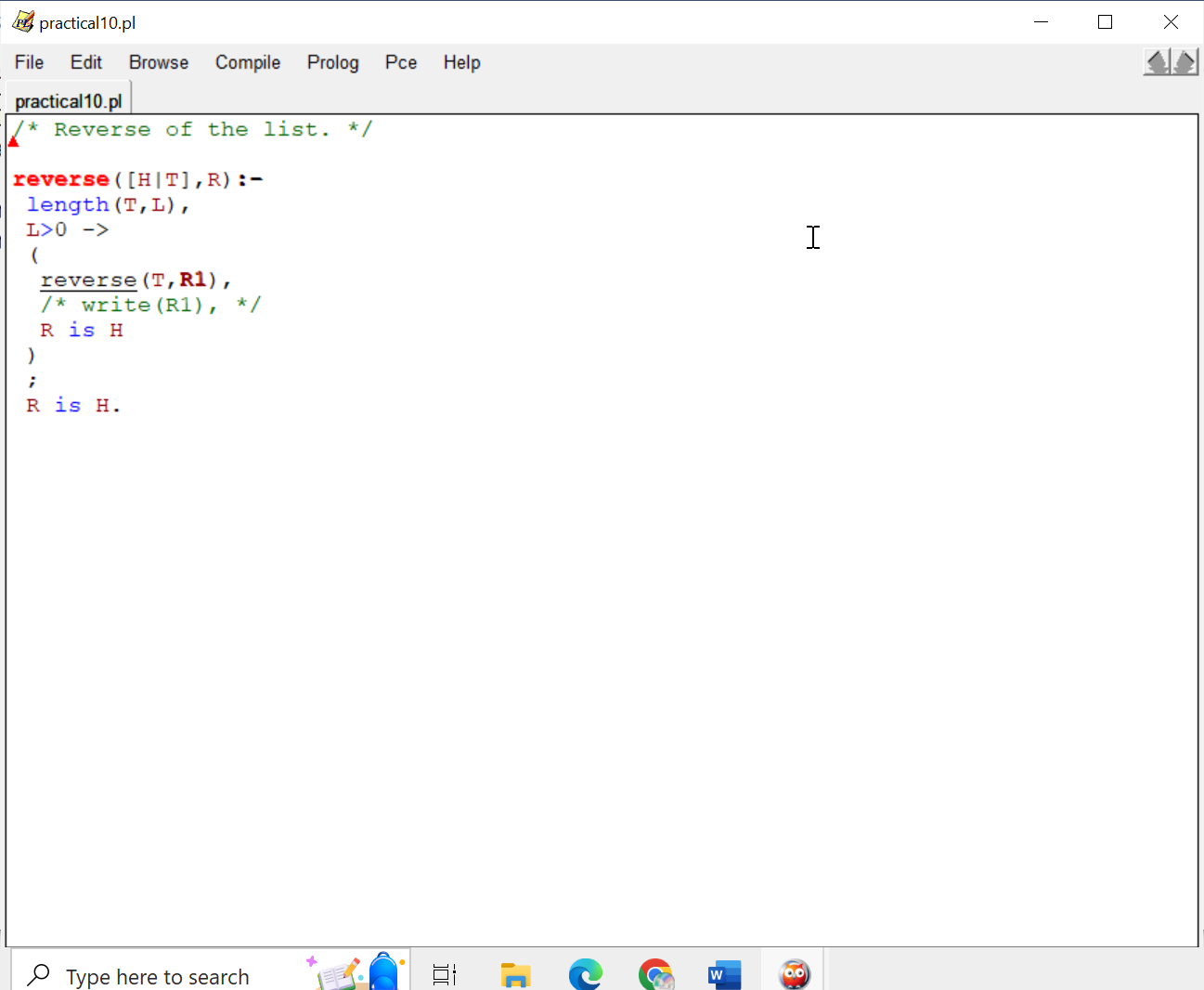
1. Write a Prolog program to implement conc (L1, L2, L3) where L2 is the list to be appended with L1 to get the resulted list L3



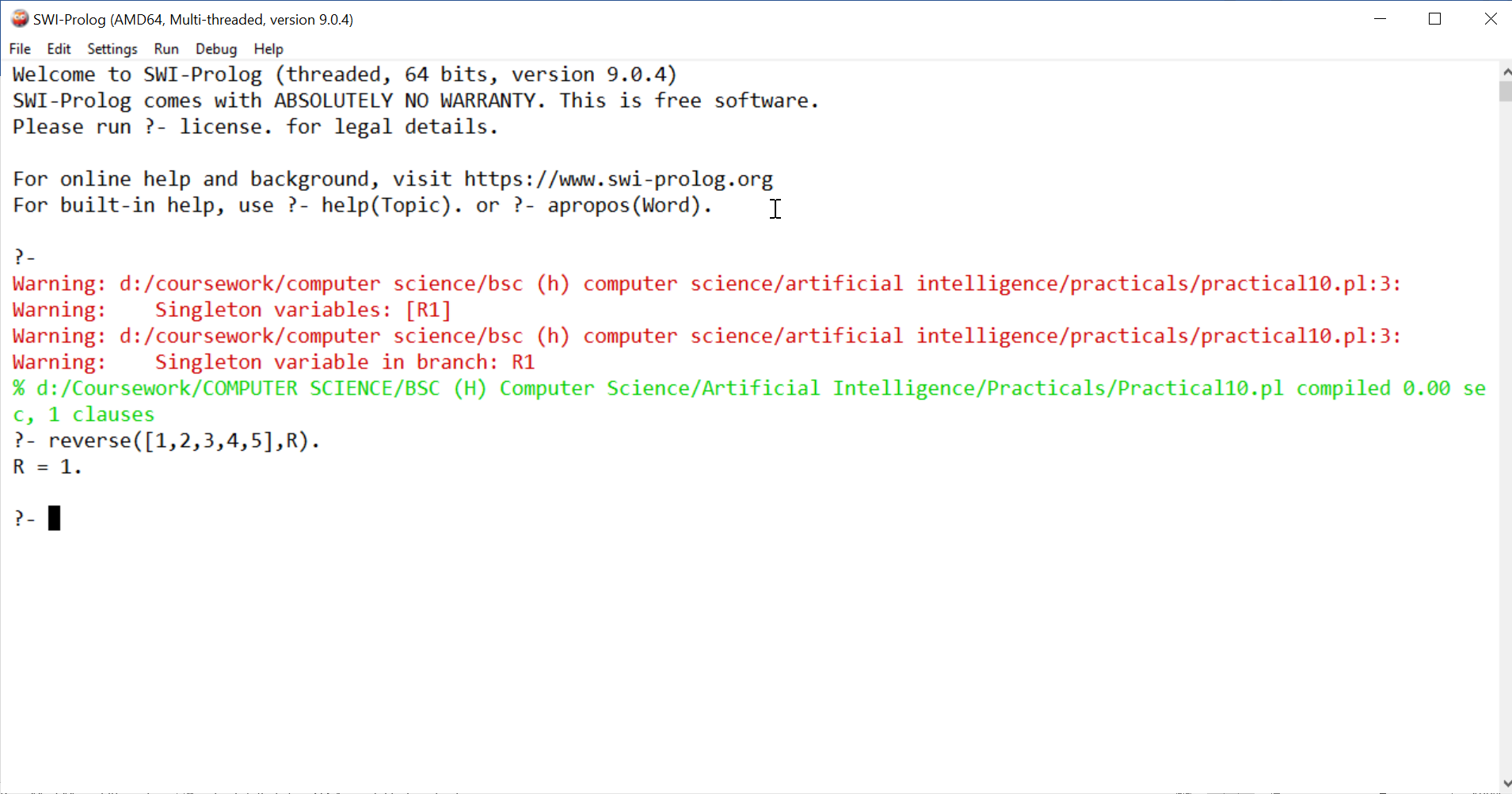
OUTPUT



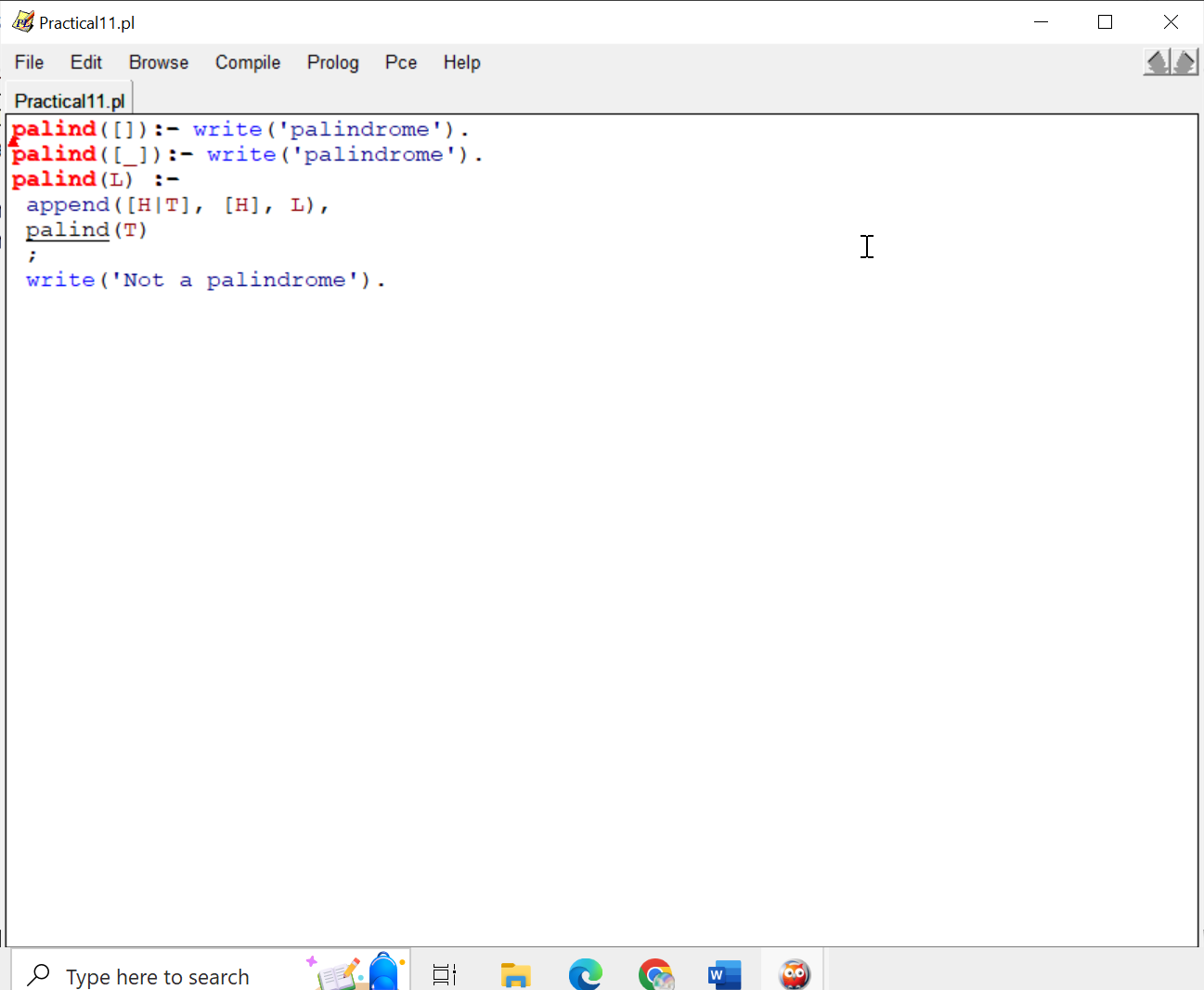
1. Write a Prolog program to implement reverse (L, R) where List L is original and List R is reversed list.



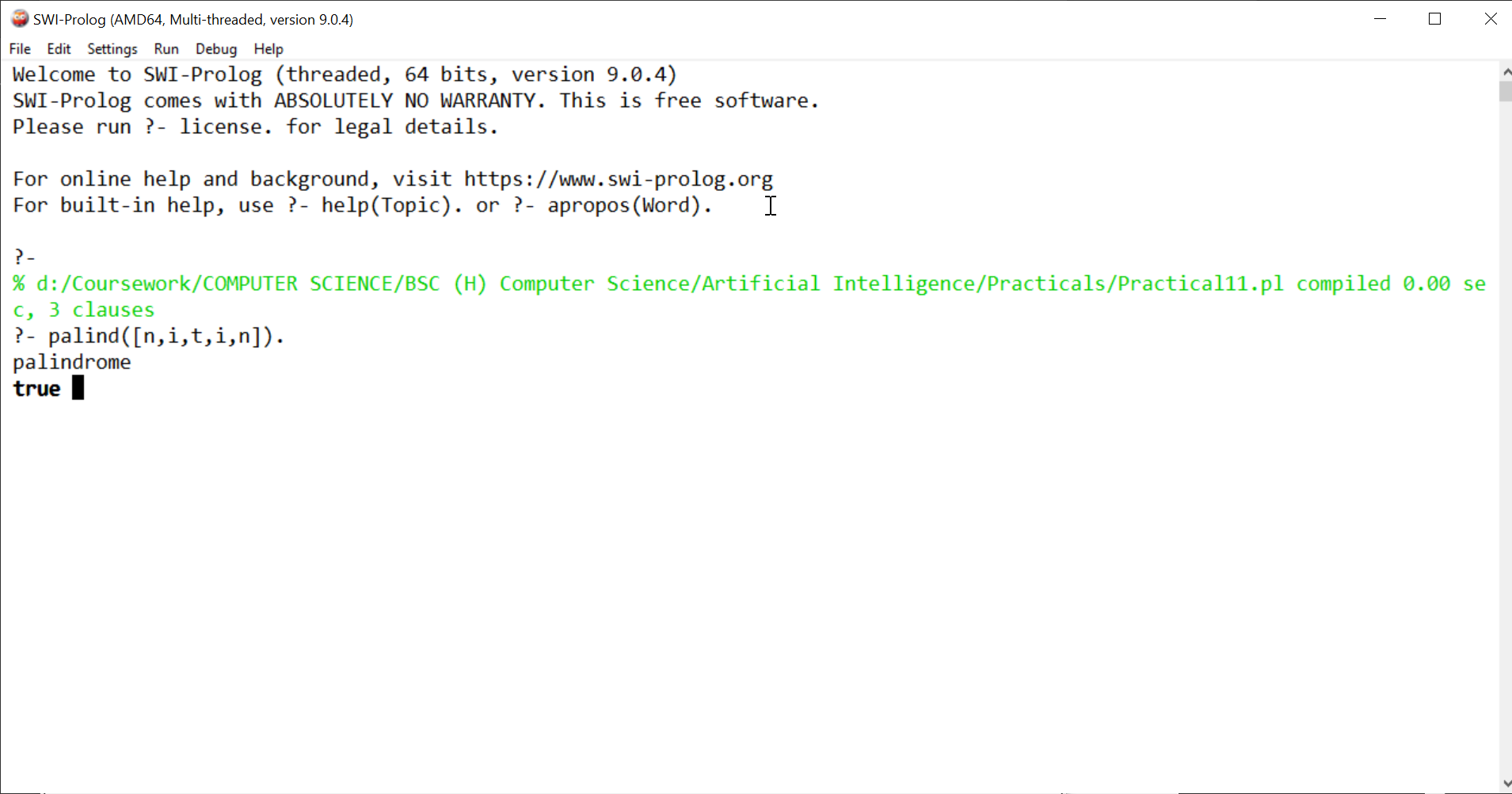
OUTPUT



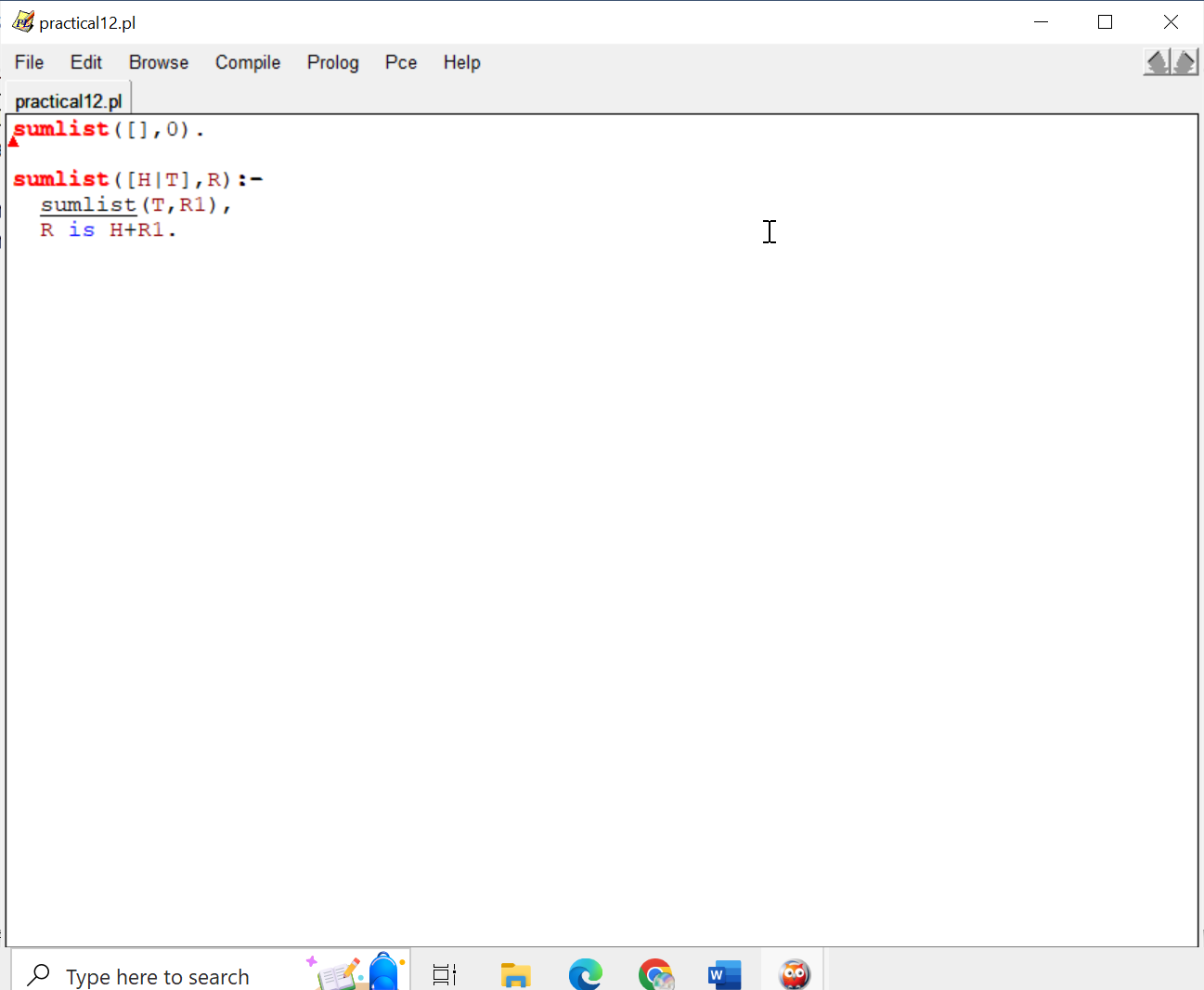
1. Write a program in PROLOG to implement palindrome (L) which checks whether a list L is a palindrome or not



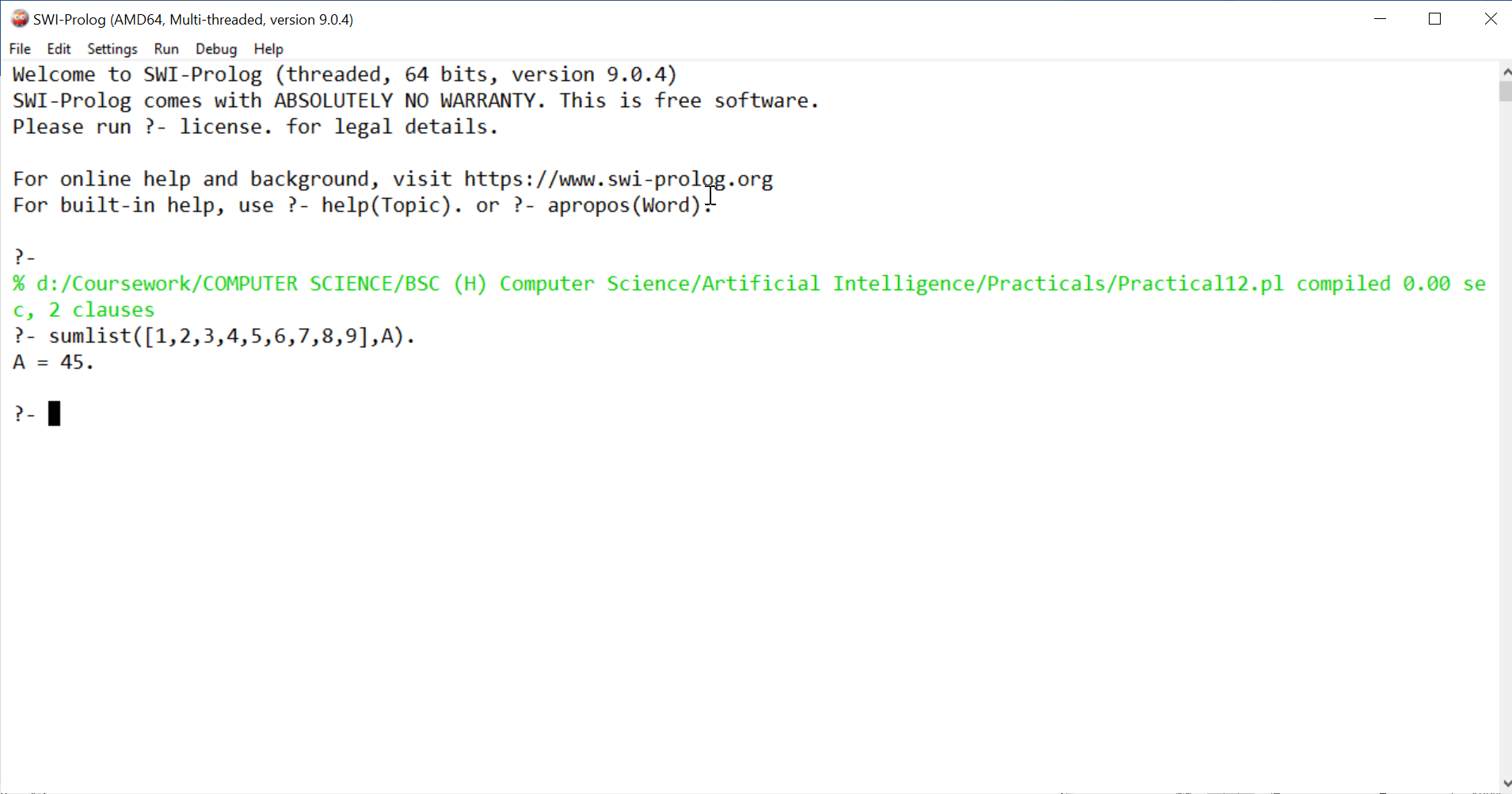
OUTPUT



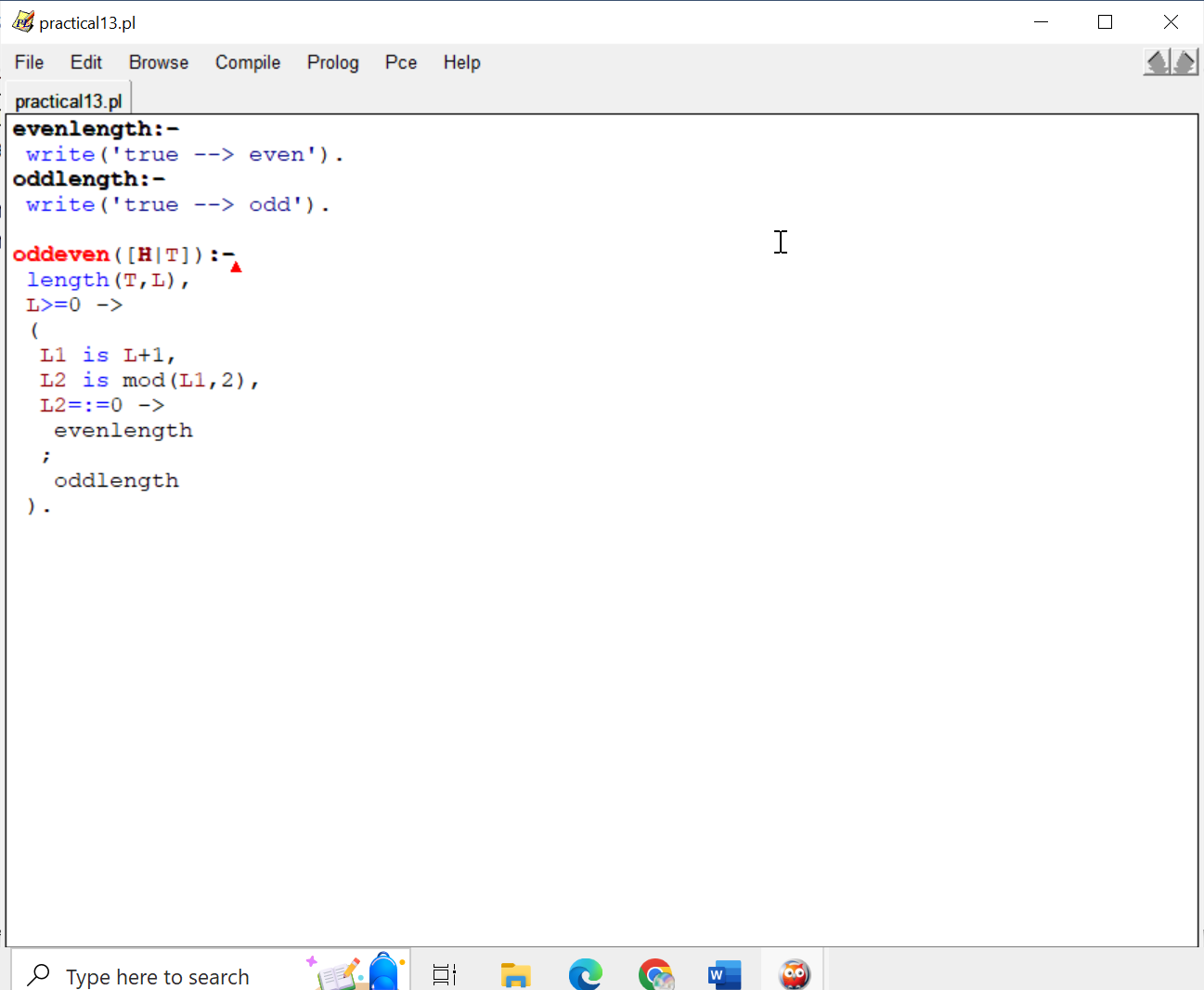
1. Write a Prolog program to implement sumlist(L, S) so that S is the sum of a given list L.



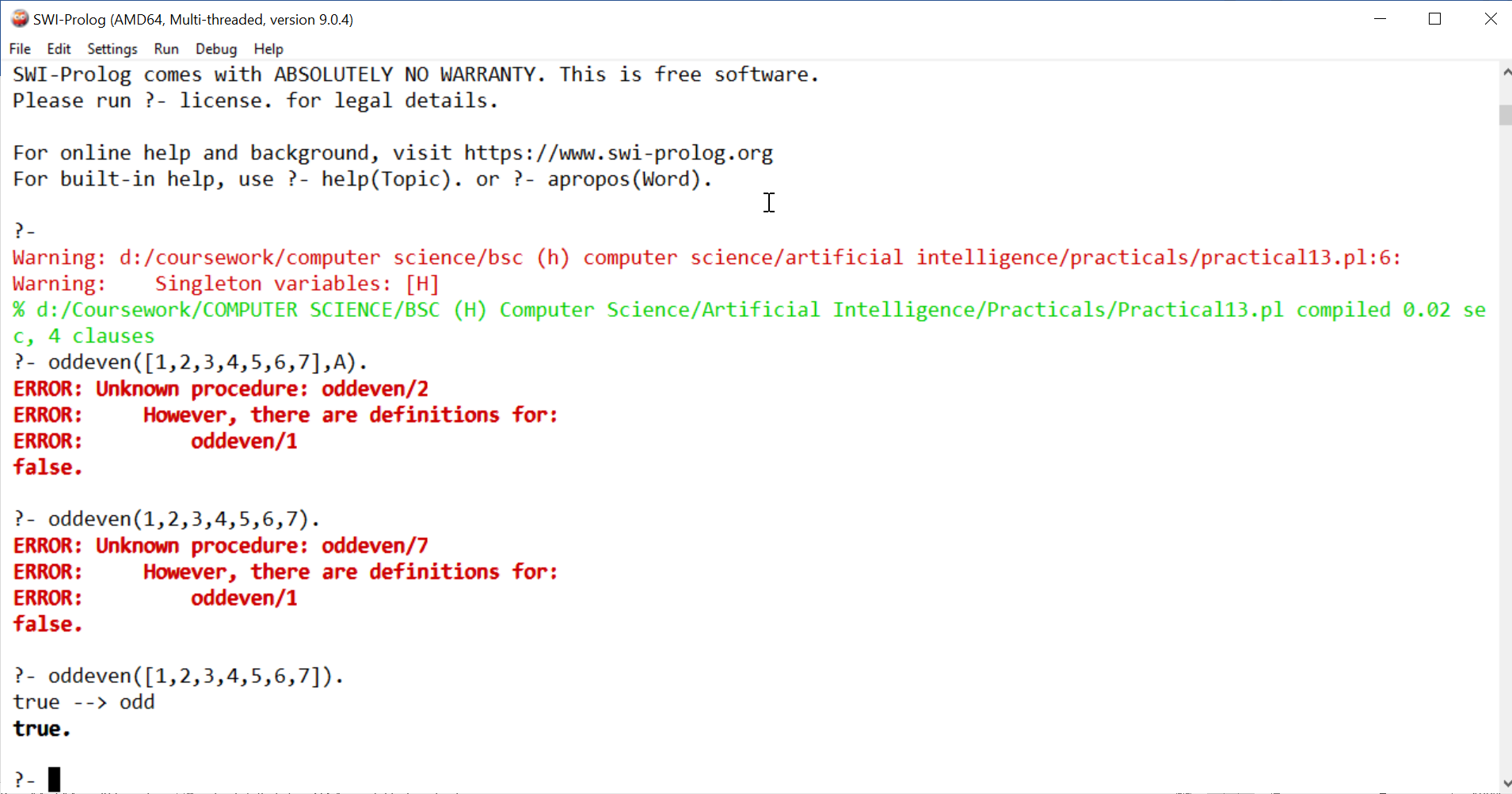
OUTPUT



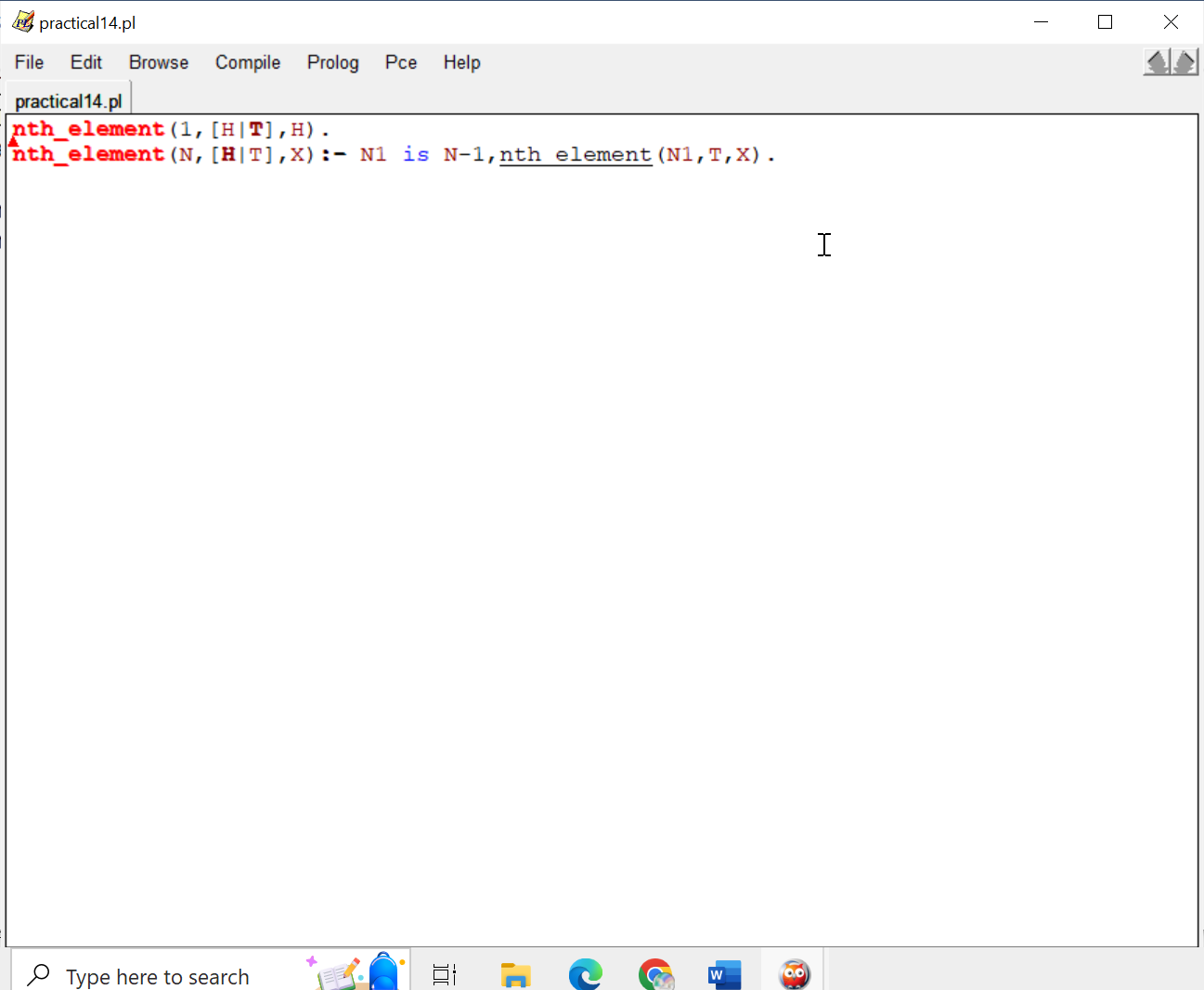
1. Write a Prolog program to implement two predicates evenlength(List) and oddlength(List) so that they are true if their argument is a list of even or odd length respectively.



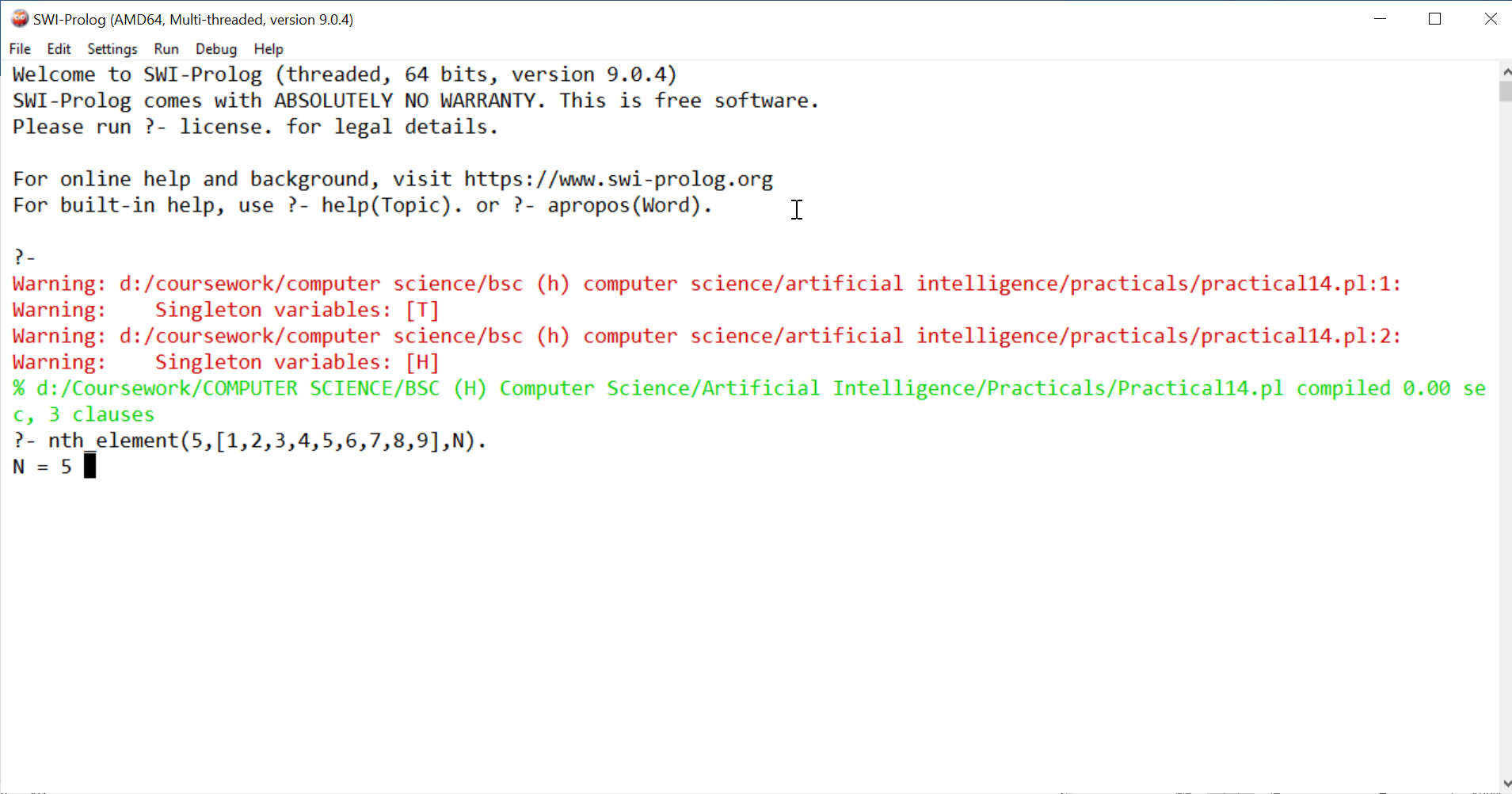
OUTPUT



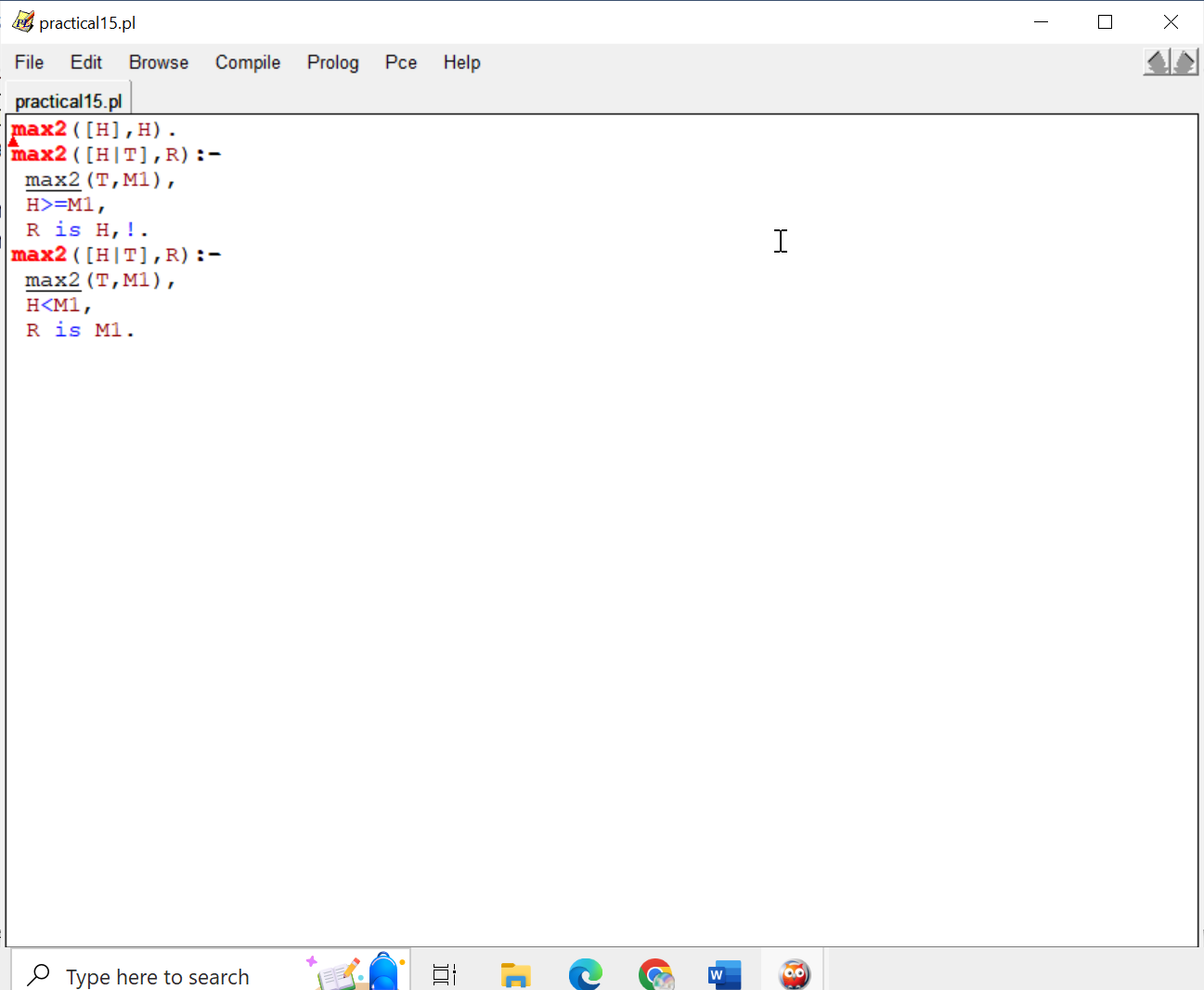
1. Write a Prolog program to implement nth\_element (N, L, X) where N is the desired position, L is a list and X represents the Nth element of L.



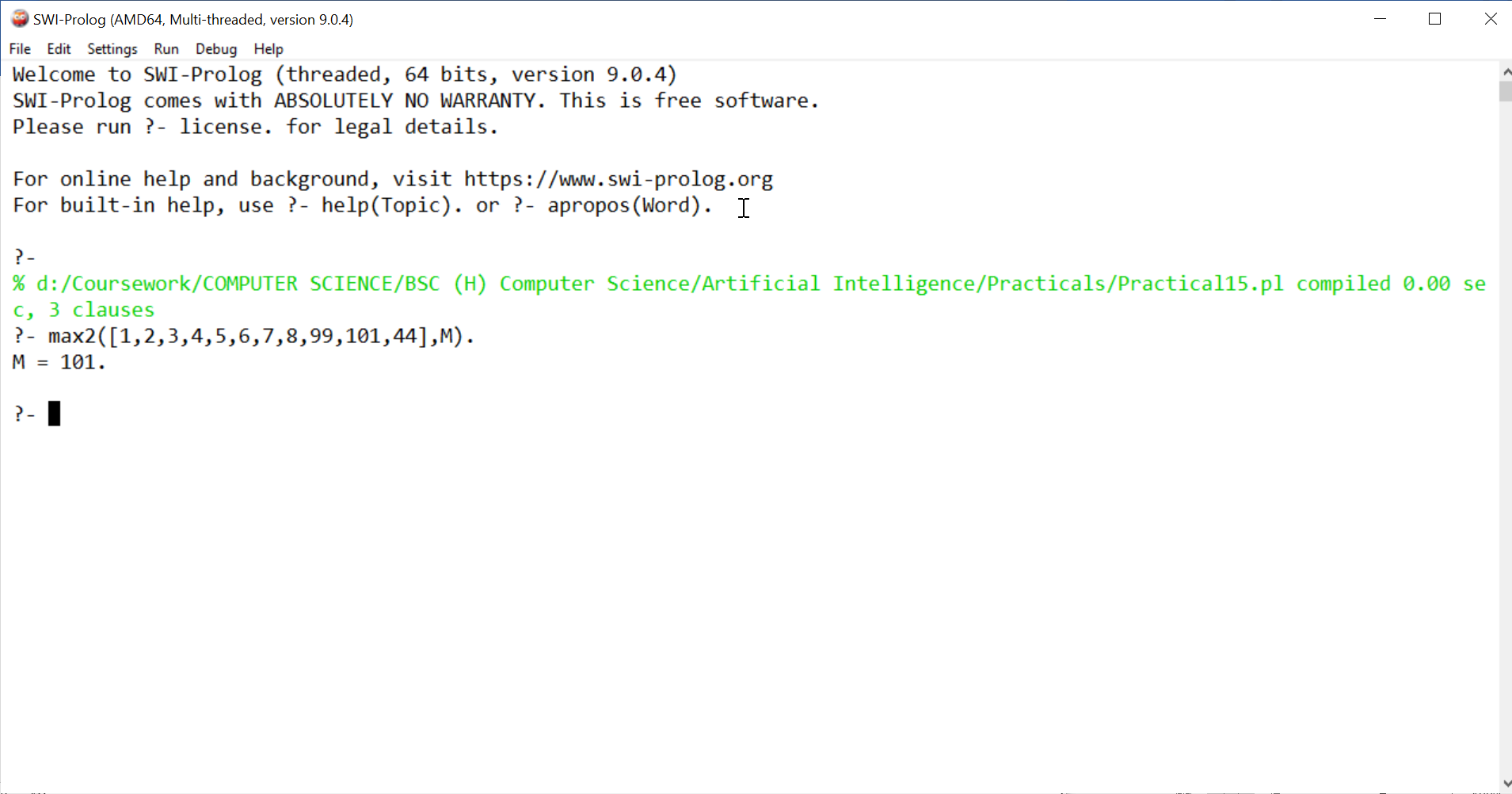
OUTPUT



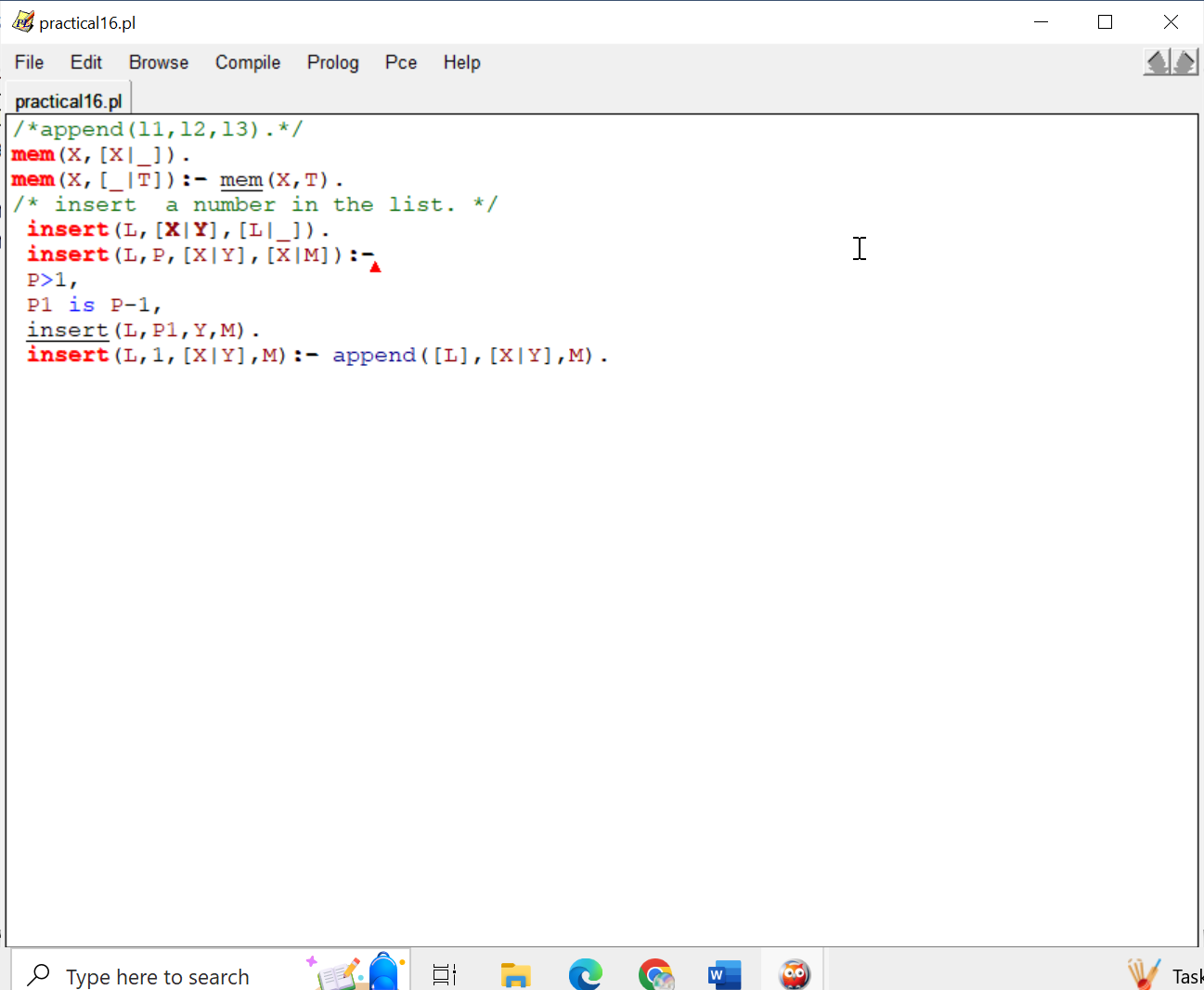
1. Write a Prolog program to implement maxlist(L, M) so that M is the maximum number in the list



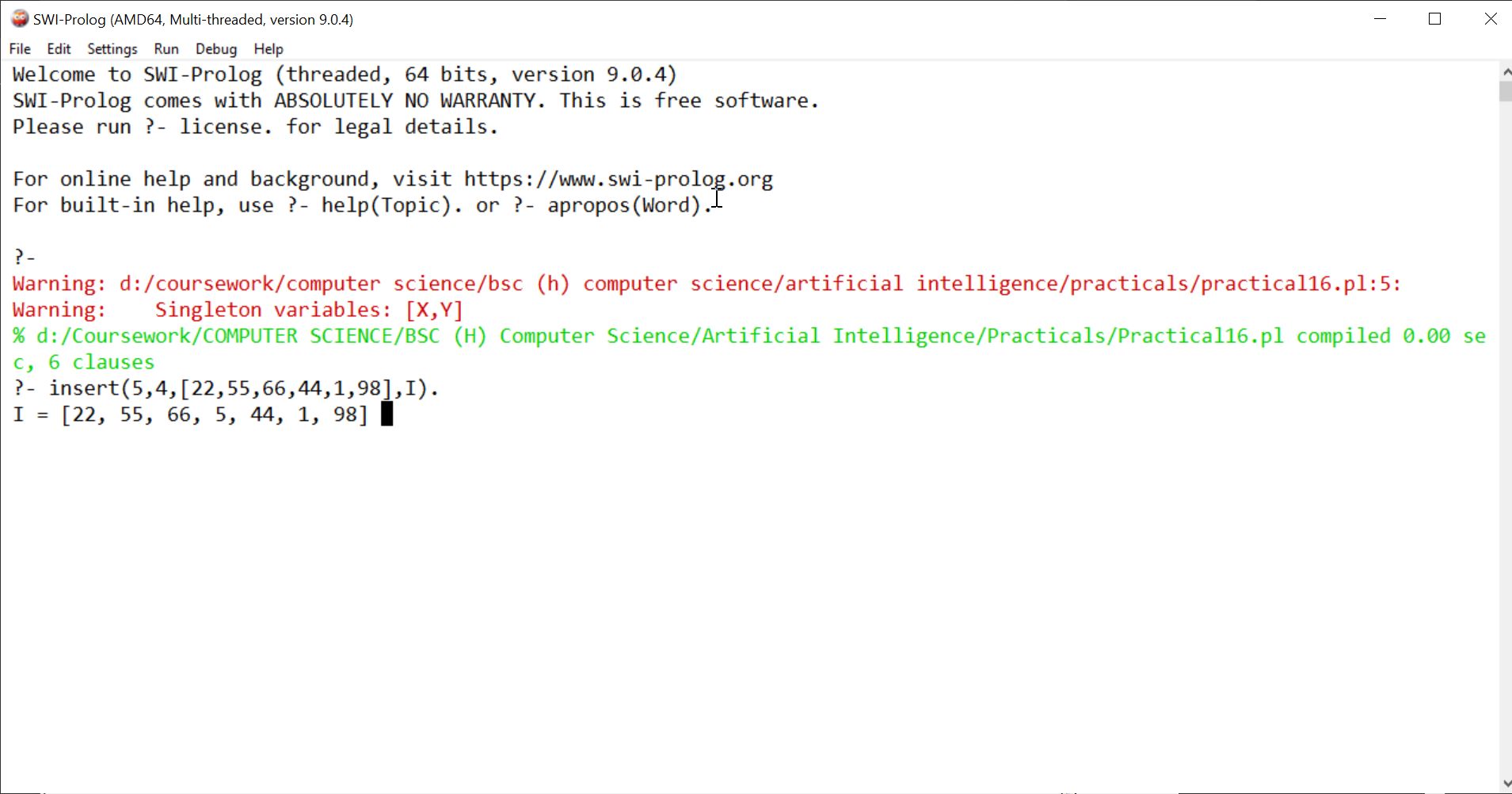
OUTPUT



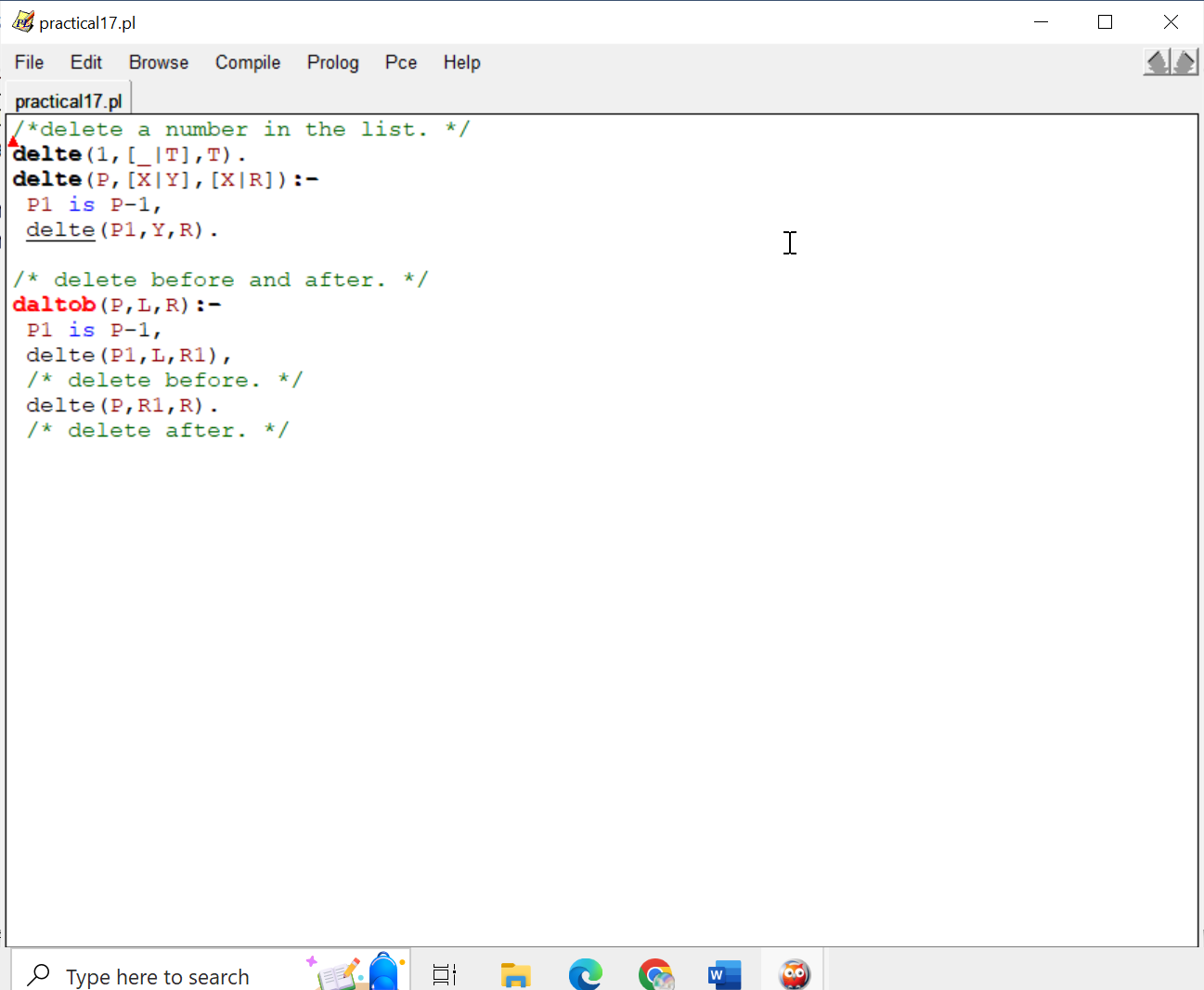
1. Write a prolog program to implement insert\_nth (I, N, L, R) that inserts an item I into Nth position of list L to generate a list R



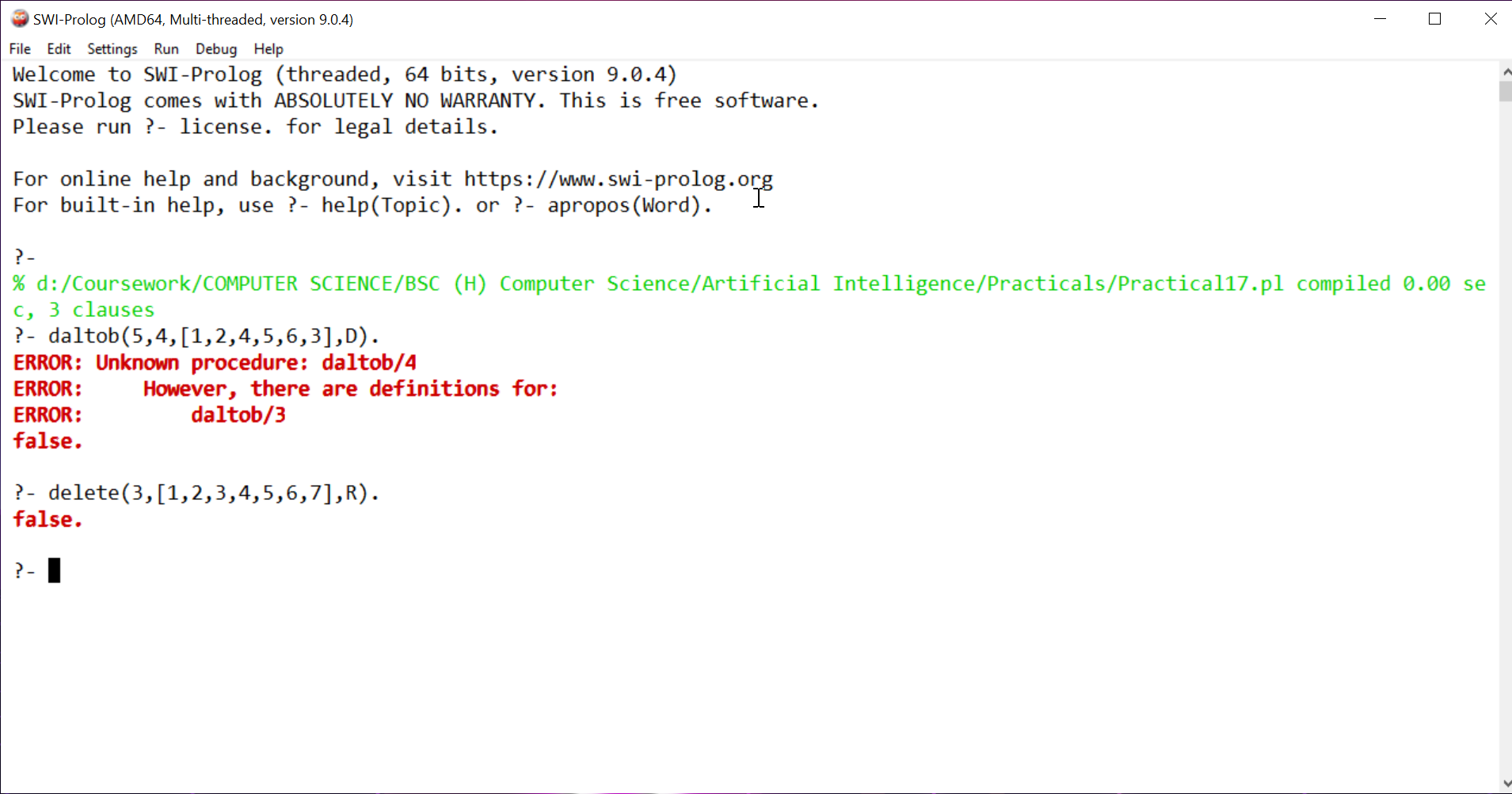
OUTPUT



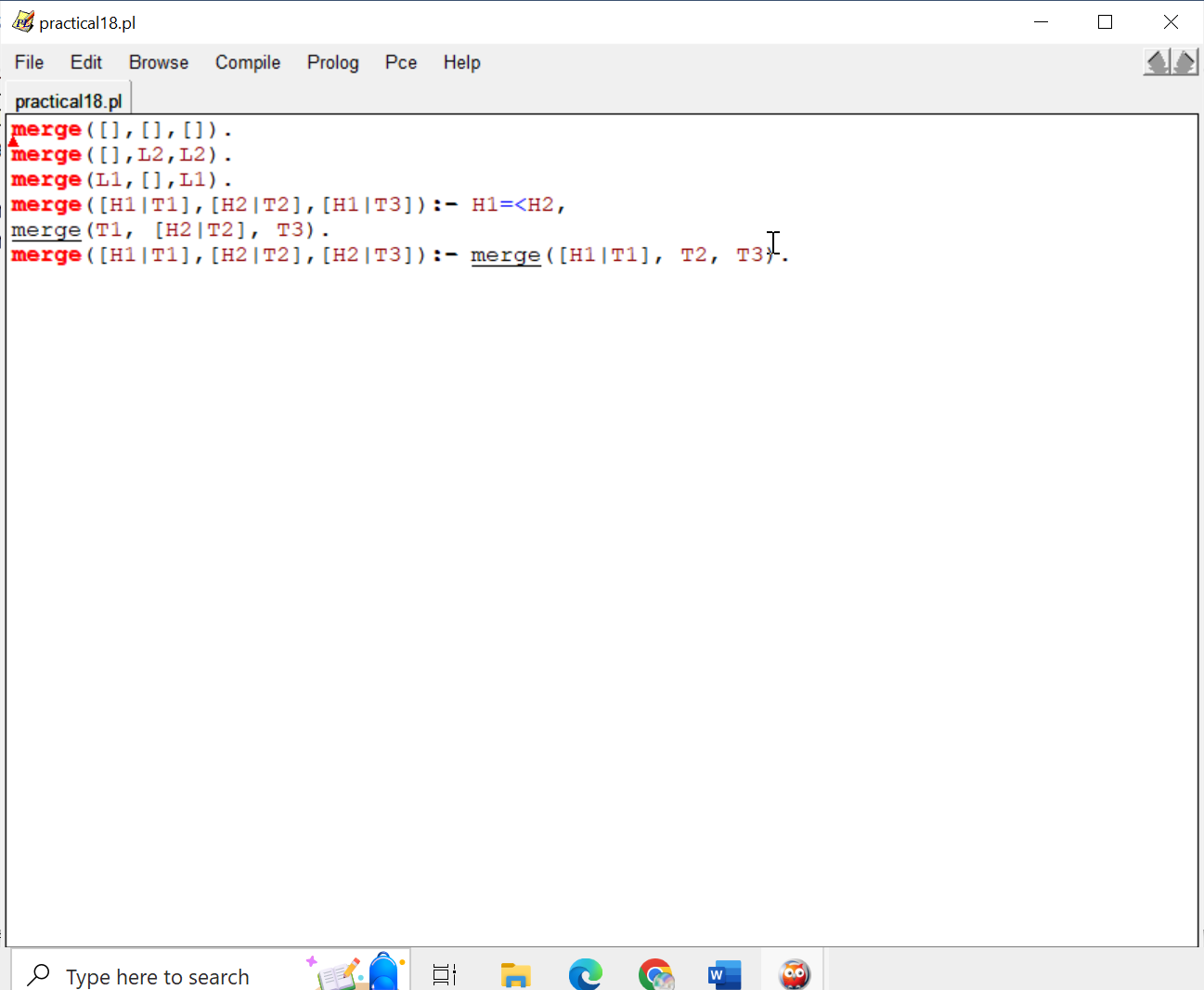
1. write a Prolog program to implement delete\_nth (N, L, R) that removes the element on Nth position from a list L to generate a list R.



OUTPUT



1. 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

