

AI ASSIGNMENT – 4

1. W.A.P.P to find factorial of a number.

Source Code:

```
factorial(0, 1).
factorial(N, F) :-
    N > 0,
    Prev is N -1,
    factorial(Prev, R),
    F is R * N.
```



The screenshot shows a terminal window titled 'sakshi@sakshi: ~/Desktop/AI/ass04'. The user has entered the command 'swipl' to start the SWI-Prolog interpreter. The interpreter displays a welcome message and instructions. The user then enters several Prolog queries: '?- consult('q1.pl').', '?- factorial(4,X).', '?- factorial(6,X).', and '?- factorial(1,X)'. The interpreter responds with 'true.' for the first query and the calculated factorial values for the others: 'X = 24 .', 'X = 720 .', and 'X = 1 .'. The prompt '?- ' is shown at the bottom, indicating the interpreter is ready for the next query.

```
sakshi@sakshi: ~/Desktop/AI/ass04$ swipl
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).

?- consult('q1.pl').
true.

?- factorial(4,X).
X = 24 .

?- factorial(6,X).
X = 720 .

?- factorial(1,X).
X = 1 .

?- 
```

2. W.A.P.P to print Fibonacci series.

The Fibonacci sequence $f(1)$, $f(2)$, $f(3)$...is: 1, 1, 2, 3, 5, 8, 13, 21, 34, 55....

Example:

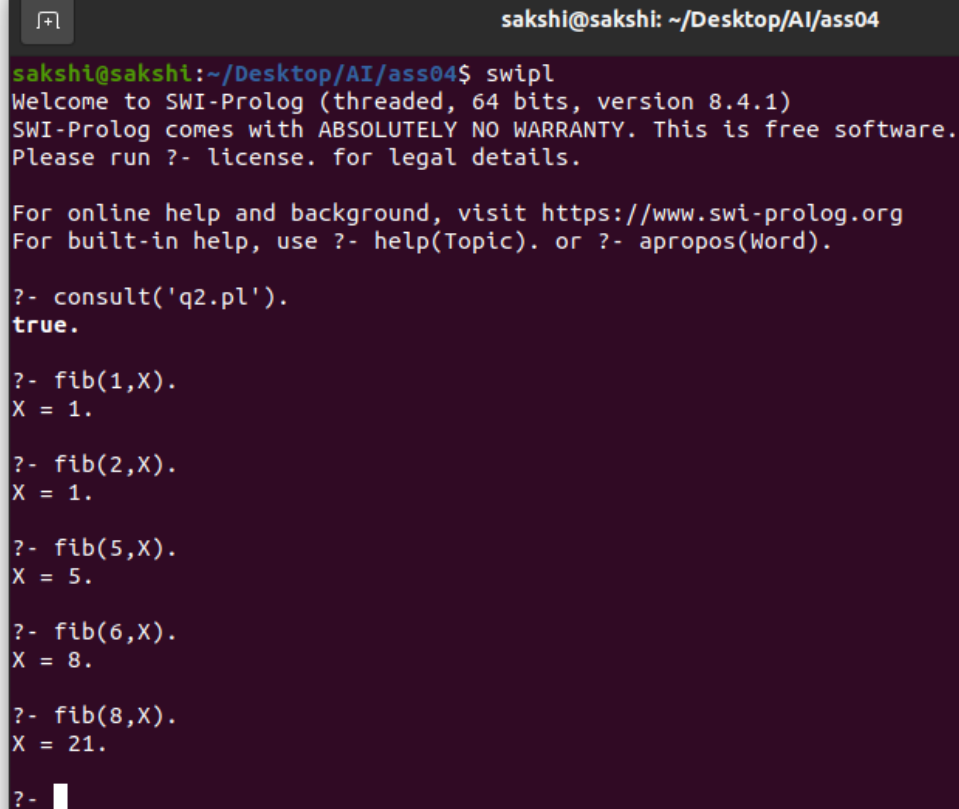
?- fib (6, R).

R = 8

Source Code:

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

```
fib(N,Ans) :-  
    N1 is N-1,  
    N2 is N-2,  
    fib(N1,Ans1),  
    fib(N2,Ans2),  
    Ans is Ans1+Ans2.
```



The screenshot shows a terminal window titled 'sakshi@sakshi: ~/Desktop/AI/ass04'. The user has entered the command 'swipl' to start the SWI-Prolog interpreter. The interpreter displays a welcome message and instructions. The user then enters several queries to test the Fibonacci program: 'consult('q2.pl').', 'fib(1,X).', 'fib(2,X).', 'fib(5,X).', 'fib(6,X).', and 'fib(8,X)'. The interpreter returns the results: 'true.', 'X = 1.', 'X = 1.', 'X = 5.', 'X = 8.', and 'X = 21.' respectively. The prompt '?- ' is visible at the bottom.

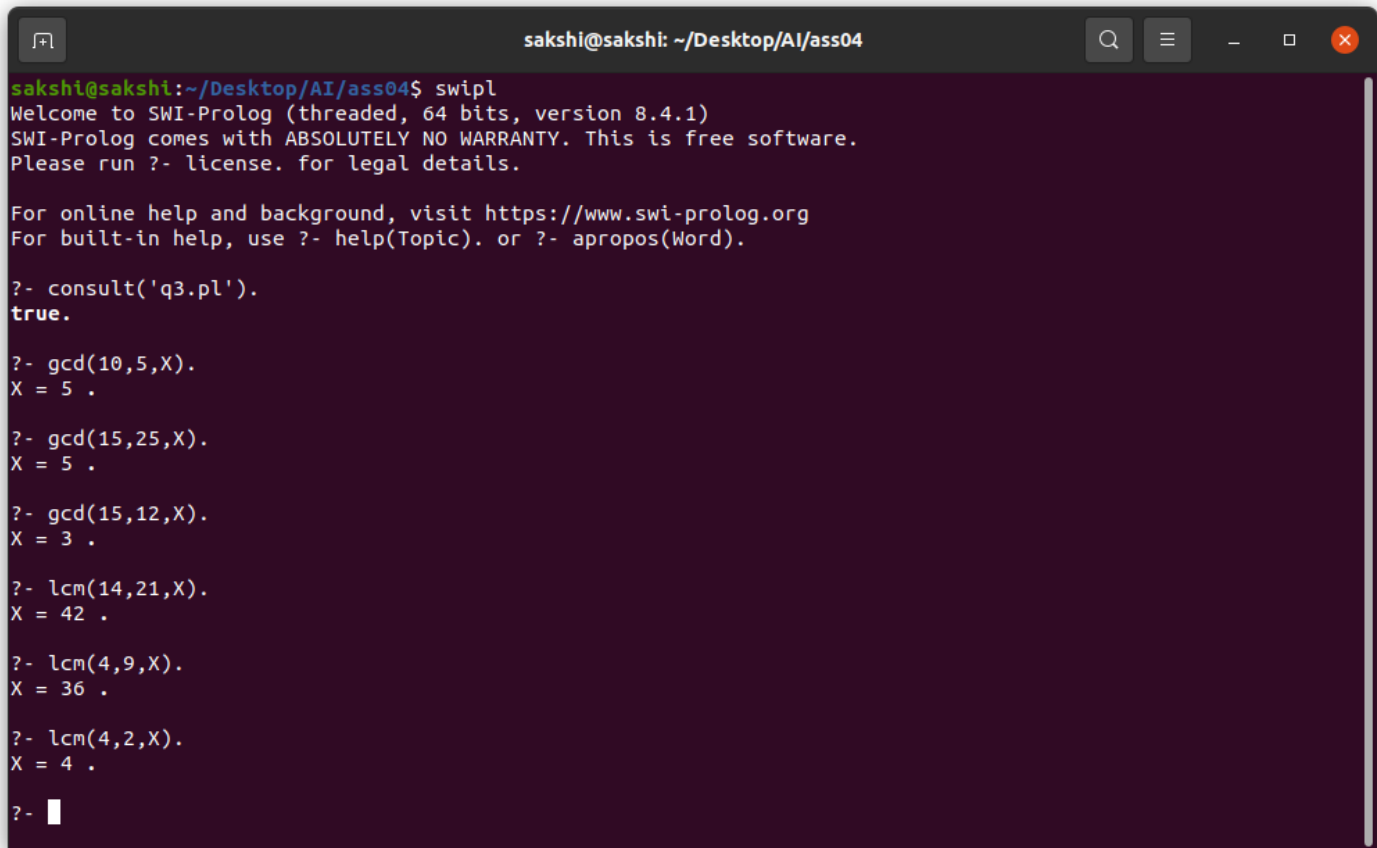
```
sakshi@sakshi:~/Desktop/AI/ass04$ swipl  
Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.1)  
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.  
Please run ?- license. for legal details.  
  
For online help and background, visit https://www.swi-prolog.org  
For built-in help, use ?- help(Topic). or ?- apropos(Word).  
  
?- consult('q2.pl').  
true.  
  
?- fib(1,X).  
X = 1.  
  
?- fib(2,X).  
X = 1.  
  
?- fib(5,X).  
X = 5.  
  
?- fib(6,X).  
X = 8.  
  
?- fib(8,X).  
X = 21.  
  
?-
```

3. W.A.P.P to finding the greatest common divider (GCD) and the least common multiple (LCM) of two integers.

Source Code:

```
% Greatest Common Divisor  
gcd(X, Y, G) :- X = Y, G = X.  
  
gcd(X, Y, G) :-  
    X < Y,  
    Y1 is Y - X,  
    gcd(X, Y1, G).  
  
gcd(X, Y, G) :- X > Y, gcd(Y, X, G).
```

```
% Least Common Multiple
lcm(X,Y,LCM) :- gcd(X,Y,GCD), LCM is X*Y//GCD.
```



```
sakshi@sakshi: ~/Desktop/AI/ass04$ swipl
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).

?- consult('q3.pl').
true.

?- gcd(10,5,X).
X = 5 .

?- gcd(15,25,X).
X = 5 .

?- gcd(15,12,X).
X = 3 .

?- lcm(14,21,X).
X = 42 .

?- lcm(4,9,X).
X = 36 .

?- lcm(4,2,X).
X = 4 .

?- 
```

4. W.A.P.P.

- A. To find length of the list.
- B. To find first and last element of the list.
- C. To find the nth element of the list.
- D. To increment each number in the list.
- E. To reverse the list.
- F. To verify if a list has an even number of elements.
- G. To count vowels in the list
- H. To remove duplicates from the list

Source Code:

```
% Length of the list
list_length([],0).
list_length([_|Tail],N) :- list_length(Tail,N1),N is N1+1.
```

```

% First and Last Element
firstElement([A|_],A).

lastElement([A],A).
lastElement([_|B],Last) :- lastElement(B,Last).

% Nth element of the list
nth_element([A|_],1,A).
nth_element([_|B],N,X) :-
    N1 is N-1,
    nth_element(B,N1,X).

% Increment each number
increment([],[]).
increment([A|B],[A1|B1]) :- A1 is A+1,increment(B,B1).

% Reverse List
list_concat([],L,L).
list_concat([X1|L1],L2,[X1|L3]) :- list_concat(L1,L2,L3).

list_rev([],[]).
list_rev([Head|Tail],Reversed) :-
    list_rev(Tail, RevTail),
    list_concat(RevTail, [Head],Reversed).

% Even number of elements
even_elements([]).
even_elements([_|B]) :- odd_elements(B).

odd_elements([]).
odd_elements([_|B]) :- even_elements(B).

% Count vowels
vowel(X):- member(X,[a,e,i,o,u]).

count_vowel([], 0).
count_vowel([X|T], N):-
    vowel(X),
    count_vowel(T,N1),
    N is N1+1.
count_vowel([_|T], N):-
    count_vowel(T,N).

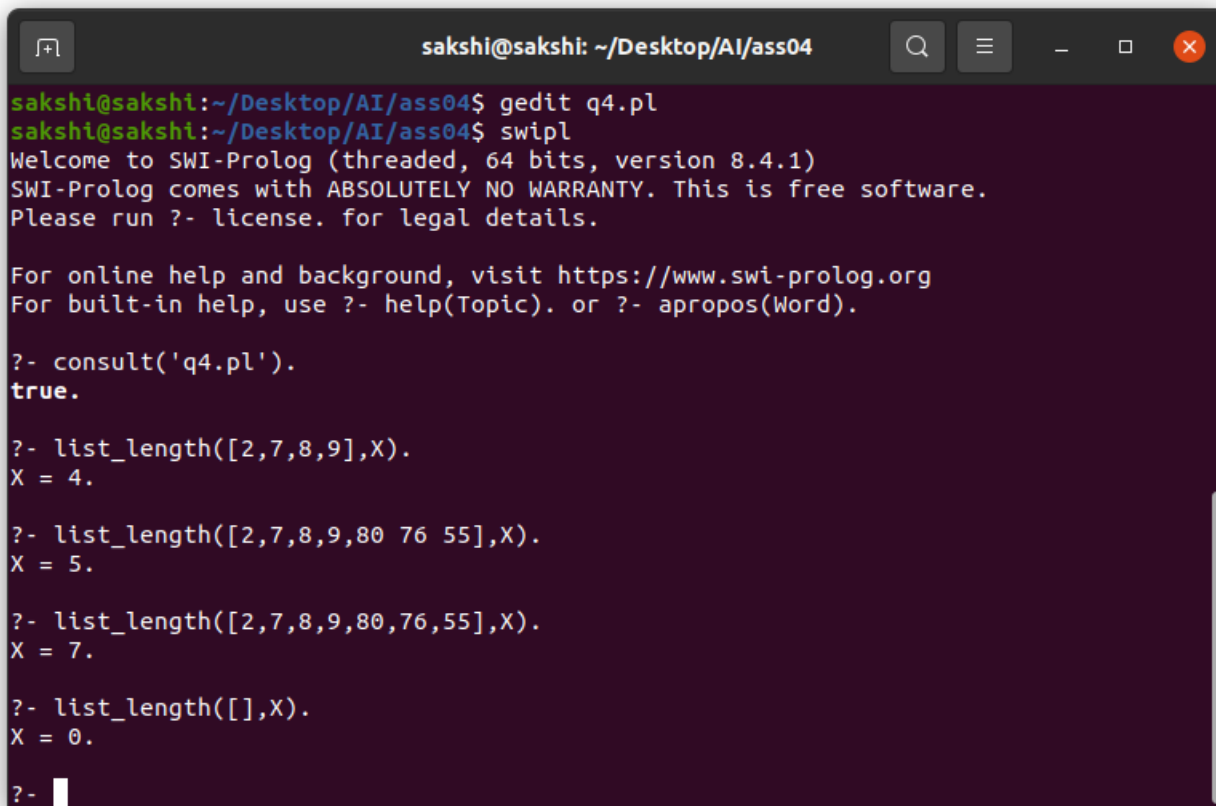
```

```
% Remove duplicates
remove_duplicates([],[]).

remove_duplicates([H | T], List) :-
    member(H, T),
    remove_duplicates(T, List).

remove_duplicates([H | T], [H|T1]) :-
    \+member(H, T),
    remove_duplicates(T, T1).
```

Length of the list:

A terminal window titled 'sakshi@sakshi: ~/Desktop/AI/ass04' with standard window controls. The terminal shows a user running 'gedit q4.pl' and then 'swipl'. The SWI-Prolog welcome message is displayed. The user enters several Prolog queries to test the 'list_length' predicate. The first query 'consult('q4.pl').' returns 'true.'. Subsequent queries 'list_length([2,7,8,9],X).', 'list_length([2,7,8,9,80 76 55],X).', and 'list_length([2,7,8,9,80,76,55],X).' return 'X = 4.', 'X = 5.', and 'X = 7.' respectively. The final query 'list_length([],X).' returns 'X = 0.'. The prompt '?- ' is visible at the bottom.

```
sakshi@sakshi:~/Desktop/AI/ass04$ gedit q4.pl
sakshi@sakshi:~/Desktop/AI/ass04$ swipl
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).

?- consult('q4.pl').
true.

?- list_length([2,7,8,9],X).
X = 4.

?- list_length([2,7,8,9,80 76 55],X).
X = 5.

?- list_length([2,7,8,9,80,76,55],X).
X = 7.

?- list_length([],X).
X = 0.

?- 
```

First and last element of the list:

```
sakshi@sakshi: ~/Desktop/AI/ass04
sakshi@sakshi:~/Desktop/AI/ass04$ swipl
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).

?- consult('q4.pl').
true.

?- firstElement([a],X).
X = a.

?- firstElement([4,3,1,5,6,9],X).
X = 4.

?- lastElement([a,b,c,d],X).
X = d.

?- lastElement([6],X).
X = 6.

?- 
```

Nth element of the list:

```
sakshi@sakshi: ~/Desktop/AI/ass04
sakshi@sakshi:~/Desktop/AI/ass04$ swipl
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).

?- consult('q4.pl').
true.

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

?- nth_element([a,b,9,d,8],1,X).
X = a.

?- nth_element([a,b,9,d,8],5,X).
X = 8.

?- nth_element([a,b,9,d,8],6,X).
false.

?- 
```

Increment each number in the list:

```
sakshi@sakshi: ~/Desktop/AI/ass04
sakshi@sakshi:~/Desktop/AI/ass04$ swipl
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).

?- consult('q4.pl').
true.

?- increment([1,3,4,5],X).
X = [2, 4, 5, 6].

?- increment([0,0,0,0],X).
X = [1, 1, 1, 1].

?- increment([45,12,90,100,81,2,53,0,7],X).
X = [46, 13, 91, 101, 82, 3, 54, 1, 8].

?- increment([],X).
X = [].

?- 
```

Reverse the list:

```
sakshi@sakshi: ~/Desktop/AI/ass04
sakshi@sakshi:~/Desktop/AI/ass04$ swipl
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).

?- consult('q4.pl').
true.

?- list_concat([a,b],[c,d],Y).
Y = [a, b, c, d].

?- list_rev([a,b,c],X).
X = [c, b, a].

?- list_rev([34,h,a,b,21,90,1,b],X).
X = [b, 1, 90, 21, b, a, h, 34].

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

?- list_rev([],X).
X = [].

?- list_rev([1,2,3,4,5],X).
X = [5, 4, 3, 2, 1].

?- 
```

Verify if a list has an even number of elements:

```
sakshi@sakshi: ~/Desktop/AI/ass04
sakshi@sakshi:~/Desktop/AI/ass04$ swipl
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).

?- consult('q4.pl').
true.

?- even_elements([a,b,c,d]).
true .

?- even_elements([1,2,3,4,5]).
false.

?- even_elements([]).
true.

?- even_elements([6]).
false.

?- 
```

Count vowels in the list:

```
sakshi@sakshi: ~/Desktop/AI/ass04
sakshi@sakshi:~/Desktop/AI/ass04$ swipl
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).

?- consult('q4.pl').
true.

?- count_vowel([a,e,i,o,e],X).
X = 5 .

?- count_vowel([a,b,c,d,e],X).
X = 2 .

?- count_vowel([h,k,i,l,o,o,p,a,a,z,e],X).
X = 6 .

?- count_vowel([],X).
X = 0.

?- 
```

Remove duplicates from the list:


```
sakshi@sakshi: ~/Desktop/AI/ass04
sakshi@sakshi:~/Desktop/AI/ass04$ swipl
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).

?- consult('q4.pl').
true.

?- remove_duplicates([a,a,b,c,b,d],L).
L = [a, c, b, d] .

?- remove_duplicates([1,1,1,1,1],L).
L = [1] .

?- remove_duplicates([],L).
L = [].

?- remove_duplicates([x,1,1,5,x,5,x,1,3,5,z],L).
L = [x, 1, 3, 5, z] .

?- []
```