



दिल्ली विश्वविद्यालय

University of Delhi

Name : Naman Jain

College roll no. : 19/1425

Exam roll no. : 19020570024

Course : B.SC.(H) CS

Semester : 6th

Artificial Intelligence Practical File

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

Solution :

Code -

```
sum(X,Y,Z):- Z is X+Y.
```

Output :

```
% c:/Users/jnama/Documents/Prolog/Q1_Sum.pl compiled 0.00 sec, 1 clauses
?- sum(3,5,S).
S = 8.
```

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

Solution :

Code -

```
max(X,Y,M):-X>Y, M is X.
max(X,Y,M):-Y>=X, M is Y.
```

Output :

```
% c:/Users/jnama/Documents/Prolog/Q2_Max.pl compiled 0.00 sec, 2 clauses
?- max(3,8,M).
M = 8.
```

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

Solution :

Code -

```
fact(0,1).
```

`fact(N,X):-N1 is N-1,fact(N1,Y),X is Y*N,!,`

Output :

```
% c:/Users/jnama/Documents/Prolog/Q3_Factorial.pl compiled 0.00 sec, 2 clauses
?- fact(5,F).
F = 120.
```

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

Solution :

Code -

```
fab(1,0).
fab(2,1).
fab(N,X):-N1 is N-1, N2 is N-2,fab(N1,X1),fab(N2,X2), X is X1+X2,!,
```

Output :

```
% c:/Users/jnama/Documents/Prolog/Q4_Fabonacci.pl compiled 0.02 sec, 3 clauses
?- fab(6,T).
T = 5.
```

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

Solution :

Code -

```
gcd(0,A,A):-!.
gcd(A,0,A):-!.
gcd(A,B,R):- B1 is mod(A,B),gcd(B,B1,R).
```

Output :

```
% c:/Users/jnama/Documents/Prolog/Q5_GCD.pl compiled 0.03 sec, 3 clauses
?- gcd(12,18,R).
R = 6.
```

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

Solution :

Code -

```
pow(X,0):-!.
pow(Num,Pow, Ans):- Ans is Num^Pow.
```

Output :

```
% c:/Users/jnama/Documents/Prolog/Q6_Power.pl compiled 0.02 sec, 2 clauses
?- pow(2,5,Ans).
Ans = 32.
```

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

Solution :

Code -

```
multi(X,0).
multi(N1,N2,R):-R is N1*N2.
```

Output :

```
% c:/Users/jnama/Documents/Prolog/Q7_Multi.pl compiled 0.02 sec, 2 clauses
?- multi(3,5,R).
R = 15.
```

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

Solution :

Code -

```
member(X,[X|Tail]).  
member(X,[Head|Tail]):-member(X,Tail).
```

Output :

```
% c:/Users/jnama/Documents/Prolog/Q8_Member.pl compiled 0.00 sec, 0 clauses  
?- member(4,[1,3,5,4,6,8]).  
true .
```

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

Solution :

Code -

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

Output :

```
% c:/Users/jnama/Documents/Prolog/Q9_Concat.pl compiled 0.00 sec, 2 clauses  
?- conc([1,2],[3,4,5],L3).  
L3 = [1, 2, 3, 4, 5].  
-
```

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

Solution :

Code -

```
conc([],L2,L2).  
conc([H|L1],L2,[H|L3]):-conc(L1,L2,L3).reverse([],[]).
```

```
reverse([H|Tail],R):-reverse(Tail,RevTail),conc(RevTail,[H],R).
```

Output :

```
% c:/Users/jnama/Documents/Prolog/Q10_Reverse.pl compiled 0.00 sec, 2 clauses
?- reverse([1,2,3,4,5],R).
R = [5, 4, 3, 2, 1].
```

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

Solution :

Code -

```
conc([],L2,L2).
conc([H|L1],L2,[H|L3]):-conc(L1,L2,L3).
palindrome([]).
palindrome([_]).
palindrome(L):-conc([H|T],[H],L),palindrome(T).
```

Output :

```
% c:/Users/jnama/Documents/Prolog/Q11_Palindrom.pl compiled 0.02 sec, 5 clauses
?- palindrome([1,2,3,2,1]).
true .
```

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

Solution :

Code -

```
sum([],0).
sum([H|T],S):-sum(T,ST), S is H+ST.
```

Output :

```
?-
% c:/Users/jnama/Documents/Prolog/Q12_SumList.pl compiled 0.00 sec, 2 clauses
?- sumlist([1,2,3,4,5],S).
S = 15.
```

Question 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.

Solution :

Code -

```
evenlength([]).
evenlength([_|T]):-oddlength(T).
oddlength([]).
oddlength([_|T]):-evenlength(T).
```

Output :

```
?-
% c:/Users/jnama/Documents/Prolog/Q13_OddEven.pl compiled 0.00 sec, 4 clauses
?- evenlength([1,2,3,4]).
true .

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

Question 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 `N`th element of `L`.

Solution :

Code -

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

Output :

```
?-
% c:/Users/jnama/Documents/Prolog/Q14_nthElement.pl compiled 0.00 sec, 2 clauses
?- nth_element(2,[1,3,7,2,5],X).
X = 3 .
```

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

Solution :

Code -

```
max(X,Y,Z):-X>Y,Z is X.
max(X,Y,Z):- Y>=X , Z is Y.
max_list([H|T],R):-max_list(T,R1),max(H,R1,R).
```

Output :

```
% c:/Users/jnama/Documents/Prolog/Q15_MaxList.pl compiled 0.00 sec, 2 clauses
?- max_list([1,3,5,2,7,4],M).
M = 7 .
```

Question 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.

Solution :

Code -

```
insertn(I,1,List,[I|List]).
insertn(I,Pos,[H|List],[H|R]):-Pos1 is Pos-1, insertn(I,Pos1,List,R).
```

Output :

```
?-
% c:/Users/jnama/Documents/Prolog/Q16_Insert_nth.pl compiled 0.00 sec, 2 clauses
?- insertn(3,4,[1,2,3,5,8,7],R).
R = [1, 2, 3, 3, 5, 8, 7] .
```


Question 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.

Solution :

Code -

```
removen(1,[_|List],List).
```

```
removen(Pos,[H|List],[H|Result]):-Pos1 is Pos-1, removen(Pos1,List,Result).
```

Output :

```
?-  
% c:/Users/jnama/Documents/Prolog/Q17_Remove_nth.pl compiled 0.00 sec, 2 clauses  
  
?- removen(3,[1,4,6,8,3,5],R).  
R = [1, 4, 8, 3, 5] ,  
-
```

Question 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.

Solution :

Code -

```
merge(X,[],X).
```

```
merge([],Y,Y).
```

```
merge([X|X1],[Y|Y1],[X|Z]):-X<Y,!,merge(X1,[Y|Y1],Z).
```

```
merge([X|X1],[Y|Y1],[X,Y|Z]):-X=Y,!,merge(X1,Y1,Z).
```

```
merge([X|X1],[Y|Y1],[Y|Z]):-X>Y,!,merge([X|X1],Y1,Z).
```

Output :

```
% c:/Users/jnama/Documents/Prolog/Q18_MergeList.pl compiled 0.00 sec, 0 clauses  
?- merge([1,2,3],[4,5,6],L3).  
L3 = [1, 2, 3, 4, 5, 6].
```

----- THE END -----