

Question 1

Prolog Code:

sibling(X, Y):-
 father(M, X),
 father(M, Y).

sister(X, Y):-
 female(X),
 father(M, X),
 father(M, Y).

grandson(X, Y):-
 male(X),
 father(M, X),
 father(Y, M).

grandson(X, Y):-
 male(X),
 mother(M, X),
 mother(Y, M).

descendant(X, Y):-
 father(Y, X).

descendant(X, Y):-
 mother(Y, X).

descendant(X, Y):-
 father(N, X),
 descendant(N, Y).

descendant(X, Y):-
 mother(N, X),
 descendant(N, Y).

male("Sasank").
male("Manoj").
male("Siva Sarma").
male("Ramana").
male("Siva Ram").
female("Naga Lakshmi").
female("Gayatri").

```
father("Siva Sarma", "Sasank").
father("Siva Sarma", "Manoj").
father("Ramana", "Naga Lakshmi").
father("Ramana", "Siva Ram").
mother("Naga Lakshmi", "Sasank").
mother("Naga Lakshmi", "Manoj").
mother("Gayatri", "Naga Lakshmi").
mother("Gayatri", "Siva Ram").
```

Output:





 <code>sibling("Manoj", "Sasank").</code>	  
true	1
 <code>sibling("Siva Ram", "Sasank").</code>	  
false	
 <code>sister("Naga Lakshmi", "Siva Ram").</code>	  
true	1
 <code>sister("Naga Lakshmi", "Ramana").</code>	  
false	
 <code>grandson("Sasank", "Gayatri")</code>	  
true	1
 <code>grandson("Siva Sarma", "Gayatri")</code>	  
false	
 <code>descendant("Sasank", "Ramana")</code>	  
true	1
 <code>descendant("Siva Sarma", "Ramana")</code>	  
false	

Question 2





Prolog Code:

```
remove(E,L,R) :-  
    select(E,L,R).
```





Output:

 `remove(a,[b,a,d,a],R)`   





`R = [b, d, a]`
`R = [b, a, d]`

 `remove(E,[b,a,d,a],R).`   

`E = b,`
`R = [a, d, a]`
`E = a,`
`R = [b, d, a]`
`E = d,`
`R = [b, a, a]`
`E = a,`
`R = [b, a, d]`

 `remove(E,L,[b,a,d]).`   

`L = [E, b, a, d]`
`L = [b, E, a, d]`
`L = [b, a, E, d]`
`L = [b, a, d, E]`
false

 `remove(p(X),[a,p(a),p(p(a)),p(p(p(a)))],R).`   

`R = [a, p(p(a)), p(p(p(a)))],`
`X = a`
`R = [a, p(a), p(p(p(a)))],`
`X = p(a)`
`R = [a, p(a), p(p(a))],`
`X = p(p(a))`

Question 3





Prolog Code:

```
subsequence([X|L1],[X|L2]) :-  
    subsequence(L1,L2).
```

```
subsequence([X|L1],[_|L2]) :-  
    subsequence([X|L1],L2).
```





```
subsequence([],_).
```

Output:

 `subsequence([a,d],[b,a,d,a]).`   

`true` 1





`false`

 `subsequence([b,a],[b,a,d,a]).`   

`true` 1

`true` 2

`false`

 `subsequence([X,Y],[b,a,d,a]).`   

`X = b,`

`Y = a`

`X = b,`

`Y = d`

`X = b,`

`Y = a`

`X = a,`

`Y = d`

`X = Y, Y = a`

`X = d,`

`Y = a`

`false`

```
subsequence(S,[b,a,d,a]).  
  
S = [b, a, d, a]  
S = [b, a, d]  
S = [b, a, a]  
S = [b, a]  
S = [b, d, a]  
S = [b, d]  
S = [b, a]  
S = [b]  
S = [a, d, a]  
S = [a, d]  
S = [a, a]  
S = [a]  
S = [d, a]  
S = [d]  
S = [a]  
S = []
```

Question 4

Prolog Code:

```
union_lists(L1,L2):-  
    union(L1, L2, U),  
    write('The Union of 2 lists is:'),  
    write(U).
```

Output:

```
union_lists([a,a],[b,a,d,a])  
The Union of 2 lists is: [b, a, d, a]  
true  
  
union_lists([a,b,c,d],[b,a,d,a])  
The Union of 2 lists is: [c, b, a, d, a]  
true
```

Question 5

Prolog Code:

```
find_min(L) :-  
    min_list(L, X),  
    write('Minimum Value is '),  
    write(X).
```

Output:



The image shows two screenshots of a Prolog interpreter window. Each screenshot has a title bar with a gear icon, the text `find_min([4,1,2,3])`, and standard window controls (down arrow, minus, close). The first screenshot shows the output: `Minimum Value is 1` and `true`, with a green `1` in the bottom right corner. The second screenshot shows the output: `Minimum Value is 2` and `true`, with a green `1` in the bottom right corner.

Explanation:

This code finds the minimum value in a given list and prints it.