1. c ( [ a , [ a ] , b ] ,X ).

X = 2.

1. Step-by-Step:

|  |  |  |
| --- | --- | --- |
| Subgoal List | Matched Clause | Substitutions |
| c([a, [a], b], X) | 1 | {a/H, [[a], b]/T} |
| **c(a, N1)**, c([[a], b], M1), X is N1 + M1 | 2 | {1/N1} |
| **c(a, N2)**, c([], M2), X1 is N2 + M2 | 2 | {1/N2} |
| c(a, N2), c([], M2), X1 is N2 + M2 | 3 | {0/M2} |
| c(a, N2), c([], M2), X1 is N2 + M2 | 1 | {1/X1} |
| **c([a], N3)**, c(b, M3), X2 is N3 + M3 | 1 | {1/N3} |
| **c(b, N4)**, c([], M4), X3 is N4 + M4 | 3 | {0/N4} |
| c(b, N4), c([], M4), X3 is N4 + M4 | 3 | {0/M4} |
| c(b, N4), c([], M4), X3 is N4 + M4 | 1 | {1/X3} |
| c([a], N3), c(b, M3), X2 is N3 + M3 | 1 | {0/M3} |
| c([a], N3), c(b, M3), X2 is N3 + M3 | 1 | {1/X2} |
| c(a, N1), c([[a], b], M1), X is N1 + M1 | 2 | {1/M1} |
| c(a, N1), c([[a], b], M1), X is N1 + M1 | 2 | {2/X} |

1. Removing the cut term makes no difference in this case is because of what the cut term does; it always succeeds and prevents any backtracking. But in this case this is the bottom of the ‘recursion’ so there will be no backtracking for this clause hence the answer does not change.
2. If the cut from first line is removed then there will be multiple results:

c ( [ a , [ a ] , b ] ,X ).

X = 2

X = 2

X = 1

X = 1

X = 1

X = 0

This happens because even after c(H, N), c(T, M), X is N+M succeed in finding bindings due to backtracking. After the first bindings have been processed, because queries ‘c([b],M).’, ‘c([[a],b],M).’ can be satisfied by ‘c(\_,0)’ clause multiple values for X are returned.