

PROLOG - Lista de Exercícios 2

- 1) Dado o seguinte programa:
s(1,um).
s(f(1),dois).
s(f(f(1)),três).
s(f(f(f(X))),N):- s(X,N).
Como Prolog responderá às seguintes questões: (mostre o diagrama de derivação)
 - a) ?- s(f(1),X).
 - b) ?- s(f(f(1)),dois).
 - c) ?- s(f(f(f(f(f(f(1))))))),X).
 - d) ?- s(X,três).
- 2) O casamento a seguir sucede ou falha? Explique o resultado.
?- X = f(X).
- 3) Escreva um objetivo usando *conc* para apagar os três últimos elementos de uma lista L produzindo outra lista L1.
- 4) Escreva um objetivo usando *conc* para apagar os primeiros e os últimos 3 elementos de uma lista L produzindo L2.
- 5) Mostre o diagrama de derivação correspondente à seguinte questão:
?- conc([a], [1, 2, 3], L).
- 6) Defina a relação *último(Item,Lista)* que será verdadeira quando Item for o último elemento de Lista. (a) Usando *conc*. (b) Sem usar *conc*.
- 7) Defina os predicados *tamanhopar(Lista)* e *tamanhoimpar(Lista)* que serão verdadeiros quando Lista for par ou ímpar, respectivamente.
- 8) Defina a relação *invertida(Lista, ListaInvertida)*. Ex:
?- invertida([a, b, c],[c, b, a]).
Yes
- 9) Defina o predicado *palindroma(Lista)*. Uma lista é palíndroma se ela se “lê” da mesma forma do início para o final ou vice-versa. Exemplo: [r,e,l,e,r].
- 10) Defina a relação *roda(Lista1, Lista2)*, tal que Lista2 é idêntica a Lista1, rodada de uma posição. Por exemplo,
?- roda([1,2,3,4,5], L1), roda(L1,L2).
L1 = [2,3,4,5,1]
L2 = [3,4,5,1,2]
- 11) Defina a relação *traduz(L1,L2)* para traduzir uma lista de números entre 0 e 9 para uma lista com as correspondentes palavras. Por exemplo,
?-traduz([3,2,1],[três,dois,um]).
Yes
use a seguinte relação auxiliar:
significa(0,zero). significa(1,um),
....
- 12) Defina a relação *subconjunto(Conj,Subcon)*. Exemplo:
?- subconjunto([a,b,c], S).
S = [a,b,c];
S = [b,c];
S = [c];
S = [];
S = [a];
...
- 13) Defina a relação *max(X, Y, Max)* que será verdadeira quando Max for o maior dos dois números X e Y.
- 14) Defina o predicado *maxlista(ListaNum, Max)* que será verdadeiro quando Max for o maior número da lista de números *ListaNúm*.
- 15) Defina o predicado *somalista(ListaNum, Soma)* que será verdadeiro quando Max for a soma dos números na lista de números *ListaNúm*.
- 16) Defina o predicado *ordenada(Lista)* que será verdadeiro quando Lista for uma lista ordenada de números.
- 17) Defina o predicado *ordena(Lista, ListaOrdenada)*. Exemplo:
?- ordena([2,1,3], L).
L = [1, 2, 3]