

UFU/FACOM

Disciplina: Programação Lógica

Período: 2018/1

Ref: Segunda Lista de Exercícios

EXERCÍCIOS

1. Implementar o n-ésimo termo de uma PA (Progressão Aritmética) considerando como dados de entrada a posição N, o primeiro elemento da PA e a razão. Considere duas versões recursivas: recursiva de cauda e recursiva que não seja de cauda.
2. Implementar a soma dos N primeiros termos de uma PA considerando como dados de entrada a posição N, o primeiro elemento da PA e a razão. Considere duas versões recursivas: recursiva de cauda e recursiva que não seja de cauda.
3. Quais as respostas Prolog para as seguintes consultas?
 1. `?-[X]=[a].`
 2. `?-X=[].`
 3. `?-[X]=[].`
 4. `?-X=[a].`
 5. `?-[X,Y]=[a].`
 6. `?-[X,Y]=[a,b].`
 7. `?-[X,Y]=[a,b,c].`
 8. `?-[X,Y,Z]=[a,b,c].`
 9. `?-[X,Y,X]=[a,b,c].`
 10. `?-[X,Y,X]=[a,b,a].`
 11. `?-[X|Y]=[a].`
 12. `?-[X|Y]=[].`
 13. `?-[X|Y]=[a,b,c].`
 14. `?-[X,Y|Z]=[a,b,c].`
 15. `?-[X,Y|Z]=[a,b].`
 16. `?-[X,Y|Z]=[a].`
 17. `?-[X,Y|Z]=[a,b,c,d].`
 18. `?-[X,Y|Z]=[a,b,a].`
 19. `?-[X,Y|Z]=[[a],b,a].`
4. N-ésimo elemento de uma lista: **n_esimo(N, Elemento, Lista)**.
5. Número de elementos de uma lista: **no_elementos (Lista, N)**.
6. Retirar a primeira ocorrência de um elemento de uma lista:
7. **tire_elemento (Elemento, Lista, Lista_nova)**.
8. Retirar todas as ocorrências de um elemento de uma lista:
9. **retirar_ocor (Elemento, Lista, Lista_nova)**.
10. Retirar elementos repetidos de uma lista: **retire_repet (Lista, Lista_nova)**.
11. Concatenar duas listas quaisquer: **concatenar (Lista1, Lista2, Lista_concat)**.
12. Encontrar o maior elemento de uma lista numérica: **maior (Lista, Elemento)**.
13. Encontrar o menor elemento de uma lista numérica: **menor (Lista, Elemento)**.
14. Pegar elementos de uma lista dada a lista de suas posições: **pegar (Lista_posições, Lista, Lista_resultante)**
15. Inserir elemento na primeira posição de uma lista: **inserir_cabeça (Elemento, Lista, Lista_resultante)**.

16. Inserir elemento numa posição N da lista: **inserir_N (Elemento, N, Lista, Lista_resultante)**
17. Inverter uma lista: **inverter (Lista, Lista_resultante)**
18. Substituir um elemento de uma lista por um outro elemento: **substitui (X, Y, Lista, Lista_resultante)**
19. Duplicar elementos de uma lista: **duplicar_todos (Lista, Lista_resultante)**
20. Duplicar um elemento de uma lista: **duplicar_um (Elemento, Lista, Lista_resultante)**
21. Verificar se a intersecção entre dois conjuntos não é vazia: **nao_vazia (Lista1, Lista2).**
22. Fazer a união de dois conjuntos: **uniao (Lista1, Lista2, Lista_uniao).**
23. Verificar se dois conjuntos são disjuntos: **disjuntos (Lista1, Lista2).**
24. Verificar se dois conjuntos são iguais: **iguais (Lista1, Lista2).**