

Aula 08 - PROLOG - Listas



**Disciplina: MATA56 - Paradigmas de Linguagens de
Programação**

Prof^o Claudio Junior Nascimento da Silva
claudiojns@ufba.br

Agenda

- Lista:
 - Conceitos;
 - Exemplos.
- Exercícios:
 - Resolução de exercícios em sala de aula;
 - Envio para e-mail (claudiojr.professor@gmail.com) até 30/05/2023;

Listas

- Estrutura de dados que representa uma coleção de objetos homogêneos;
- Notação:
 - “[” - início da lista;
 - “]” - fim da lista.
- Uma lista é:
 - Ou uma lista vazia, escrita como [];
 - Ou uma estrutura com dois componentes: cabeça e cauda;
 - Listas são processadas dividindo-as em cabeça e caudas (exceto a lista vazia).

Listas

- Elementos de uma lista são lidos da esquerda para a direita:
 - letras([a, b, c, d]).
 - Primeiro elemento ou cabeça: a;
 - Cauda (sub-lista): [b, c, d];
- Como distinguir de onde se encontra a cabeça da cauda da lista?
 - Pipe ou |;

Listas

Lista	Cabeça	Cauda
[a, b, c]	a	[b, c]
[]	Não tem	Não tem
[[o, gato], sentou]	[o, gato]	[sentou]
[o, [gato, sentou]	o	[[gato, sentou]]
[o, [gato, sentou], ali]	o	[[gato, sentou], ali]
[X + Y, x + y]	X + Y	[x + y]
[a]	a	[]

Listas

```
1 matriculado(pedro,estatistica).
2 matriculado(pedro,economia).
3 matriculado(pedro,programacao).
4 matriculado(pedro,"calculo I").
5 matriculado(pedro,"banco de dados").
6 matriculado(maria,estatistica).
7 matriculado(maria,"calculo I").
8 matriculado(maria,"banco de dados").
9 matriculado(carlos,estatistica).
10 matriculado(carlos,"calculo I").
11 matriculado(carlos,"banco de dados").
```

?- matriculado(_,Y).

 matriculado(_,Y).

Y = estatistica

Y = economia

Y = programacao

Y = "calculo I"

Y = "banco de dados"

Y = estatistica

Y = "calculo I"

Y = "banco de dados"

Y = estatistica

Y = "calculo I"

Y = "banco de dados"

Listas

```
1 matriculado(pedro,estatistica).
2 matriculado(pedro,economia).
3 matriculado(pedro,programacao).
4 matriculado(pedro,"calculo I").
5 matriculado(pedro,"banco de dados").
6 matriculado(maria,estatistica).
7 matriculado(maria,"calculo I").
8 matriculado(maria,"banco de dados").
9 matriculado(carlos,estatistica).
10 matriculado(carlos,"calculo I").
11 matriculado(carlos,"banco de dados").
12
13 % regra:
14 listar_disciplinas :-
15     findall(Disciplina,matriculado(_,Disciplina),ListaDisciplina),
16     write(ListaDisciplina).
```

?- listar_disciplinas.

Y = economia
Y = programacao
Y = "calculo I"
Y = "banco de dados"
Y = estatistica
Y = "calculo I"
Y = "banco de dados"
Y = estatistica
Y = "calculo I"
Y = "banco de dados"

listar_disciplinas.

[estatistica, economia, programacao, calculo I, banco de dados,
estatistica, calculo I, banco de dados, estatistica, calculo I, banco
de dados]
true

Listas

```
1 matriculado(pedro,estatistica).
2 matriculado(pedro,economia).
3 matriculado(pedro,programacao).
4 matriculado(pedro,"calculo I").
5 matriculado(pedro,"banco de dados").
6 matriculado(maria,estatistica).
7 matriculado(maria,"calculo I").
8 matriculado(maria,"banco de dados").
9 matriculado(carlos,estatistica).
10 matriculado(carlos,"calculo I").
11 matriculado(carlos,"banco de dados").
12
13 % regra:
14 listar_disciplinas :-
15     findall(Disciplina,matriculado(_,Disciplina),ListaDisciplina),
16     write(ListaDisciplina).
17
18 listar_disciplinas_sr :-
19     findall(Disciplina,matriculado(_,Disciplina),ListaDisciplina),
20     list_to_set(ListaDisciplina, ListaSemDuplicidade),
21     write(ListaSemDuplicidade).
```

?- listar_disciplinas_sr.

Y = estatistica
Y = "calculo I"
Y = "banco de dados"
Y = estatistica
Y = "calculo I"
Y = "banco de dados"

listar_disciplinas.

[estatistica, economia, programacao, calculo I, banco de dados, estatistica, calculo I, banco de dados, estatistica, calculo I, banco de dados]

true

listar_disciplinas_sr.

[estatistica, economia, programacao, calculo I, banco de dados]

Listas

```
17
18 listar_disciplinas_sr:-
19 findall(Disciplina,matriculado(_,Disciplina),ListaDisciplina),
20 list_to_set(ListaDisciplina, ListaSemDuplicidade),
21 write(ListaSemDuplicidade),
22
23 length(ListaSemDuplicidade, Comprimento), ←
24 write('Comprimento.....: '),write(Comprimento),writeln(' elementos'),
25
26 nth0(0, ListaSemDuplicidade, Primeiro), ←
27 nth0(3, ListaSemDuplicidade, Enesimo),
28 last(ListaSemDuplicidade, Ultimo),
29 write('Primeiro elemento: '),writeln(Primeiro),
30 write('Enésimo elemento.: '),writeln(Enesimo),
31 write('Ultimo elemento..: '),writeln(Ultimo),
32
33 [Cabeça | Cauda] = ListaSemDuplicidade, ←
34 write('Cabeça.....: '), writeln(Cabeça),
35 write('Cauda.....: '), writeln(Cauda),
36
37 reverse(ListaSemDuplicidade, ListaInvertida), ←
38 write('Lista invertida..: '), write(ListaInvertida).
```

listar_disciplinas_sr.

[estatistica, economia, programacao, calculo I, banco de dados]

Comprimento.....: 5 elementos

Primeiro elemento: estatistica

Enésimo elemento.: calculo I

Ultimo elemento..: banco de dados

Cabeça.....: estatistica

Cauda.....:

[economia, programacao, calculo I, banco de dados]

Lista invertida..:

[banco de dados, calculo I, programacao, economia, estatistica]

true

?- listar_disciplinas_sr.

Examples▲

History▲

Solutions▲

☐ table results

Run!

Listas

```
18 listar_disciplinas_sr :-
19   findall(Disciplina, matriculado(_, Disciplina), ListaDisciplina),
20   list_to_set(ListaDisciplina, ListaSemDuplicidade),
21   writeln(ListaSemDuplicidade),
22   length(ListaSemDuplicidade, Comprimento),
23   write('Comprimento.....: '), write(Comprimento), writeln(' elementos'),
24
25   nth0(0, ListaSemDuplicidade, Primeiro),
26   nth0(3, ListaSemDuplicidade, Enesimo),
27   last(ListaSemDuplicidade, Ultimo),
28   write('Primeiro elemento: '), writeln(Primeiro),
29   write('Enésimo elemento.: '), writeln(Enesimo),
30   write('Último elemento..: '), writeln(Ultimo),
31
32   [Cabeça | Cauda] = ListaSemDuplicidade,
33   write('Cabeça.....: '), writeln(Cabeça),
34   write('Cauda.....: '), writeln(Cauda),
35
36   reverse(ListaSemDuplicidade, ListaInvertida),
37   write('Lista invertida..: '), writeln(ListaInvertida),
38
39   Elemento = "calculo II",
40   (member(Elemento, ListaSemDuplicidade) ->
41     write('O elemento '), write(Elemento), write(' está na Lista.') ;
42     write('O elemento '), write(Elemento), write(' não está na Lista.') ),
43   writeln('Percorrer a Lista: '),
44   percorre_lista(ListaSemDuplicidade) .
45
46 percorre_lista([]).
47 percorre_lista([CabeçaLista | CaudaLista]) :-
48   write('Elemento: '), writeln(CabeçaLista),
49   percorre_lista(CaudaLista).
```

listar_disciplinas_sr.

[estatistica, economia, programacao, calculo I, banco de dados]

Comprimento.....: 5 elementos

Primeiro elemento: estatistica

Enésimo elemento.: calculo I

Último elemento.: banco de dados

Cabeça.....: estatistica

Cauda.....: [economia, programacao, calculo I, banco de dados]

Lista invertida..: [banco de dados, calculo I, programacao, economia, estatistica]

O elemento calculo II não está na Lista. Percorrer a Lista:

Elemento: estatistica

Elemento: economia

Elemento: programacao

Elemento: calculo I

Elemento: banco de dados

true

?- listar_disciplinas_sr.

Examples▲

History▲

Solutions▲

Listas

Lista - Disciplinas ✕ +

```
[Cabeca | Cauda] = ListaSemDuplicidade,
write('Cabeça.....: '), writeln(Cabeca),
write('Cauda.....: '), writeln(Cauda),

reverse(ListaSemDuplicidade, ListaInvertida),
write('Lista invertida.: '), writeln(ListaInvertida),

Elemento = "calculo II",
(member(Elemento, ListaSemDuplicidade) ->
  write('O elemento '),write(Elemento),writeln(' está na Lista.') ;
  write('O elemento '),write(Elemento),writeln(' não está na Lista.').
writeln('Percorrer a Lista: '),
percorre_lista(ListaSemDuplicidade),

ListacomNovoElemento=["calculo II" | ListaSemDuplicidade],
delete(ListacomNovoElemento, "banco de dados", ListasemElemento),

writeln('Lista Original: '),
writeln(ListaDisciplina),
writeln('Lista sem duplicidade: '),
writeln(ListaSemDuplicidade),
writeln('Nova lista (+): '),
writeln(ListacomNovoElemento),
writeln('Nova lista (-): '),
writeln(ListasemElemento).
```

```
[estatistica, economia, programacao, calculo I, banco de dados]
Comprimento.....: 5 elementos
Primeiro elemento: estatistica
Enésimo elemento.: calculo I
Último elemento.: banco de dados
Cabeça.....: estatistica
Cauda.....: [economia, programacao, calculo I, banco de dados]
Lista invertida.: [banco de dados, calculo I, programacao, economia, estatistica]
O elemento calculo II não está na Lista.
Percorrer a Lista:
Elemento: estatistica
Elemento: economia
Elemento: programacao
Elemento: calculo I
Elemento: banco de dados
Lista Original:
[estatistica, economia, programacao, calculo I, banco de dados, estatistica, calculo I, banco de dados,
estatistica, calculo I, banco de dados]
Lista sem duplicidade:
[estatistica, economia, programacao, calculo I, banco de dados]
Nova lista (+):
[calculo II, estatistica, economia, programacao, calculo I, banco de dados]
Nova lista (-):
[calculo II, estatistica, economia, programacao, calculo I]
true
```

?- listar_disciplinas_sr.

Listas

- Ordenar Listas:
 - `sort(Lista, ListaOrdenada)`.
 - Remove elementos duplicados da lista.
 - `msort(Lista, ListaOrdenada)`.
 - Mantém elementos duplicados.