

Trabalho 4: prolog

Prof. Chauã Queirolo

Exercício 1

Construir um predicado em Prolog, denominado “elem_repetidos”, o qual relaciona uma lista de itens com uma sub-lista (possivelmente vazia) de todos os itens da lista original que aparecem repetidos em qualquer numero de vezes. Seu comportamento é o expresso abaixo:

```
?- elem_repetidos([a,b], Z).  
Z = [] ?  
yes
```

```
?- elem_repetidos([a,a,b], Z).  
Z = [a] ?  
yes
```

```
?- elem_repetidos([a, b, c, c, d, e, f, g, h, c, e, a], Z).  
Z = [a, c, e] ?  
yes
```

Exercício 2

Construir um predicado em Prolog, denominado “intercalada”, o qual relaciona duas listas de itens (L1 e L2 – possivelmente vazias) com uma terceira lista (L3), onde esta última contém a intercalação dos elementos das duas outras listas (L1 e L2). Caso L1 e L2 sejam de tamanhos diferentes, completar L3 com os itens não intercalados da lista de maior número de itens. Seu comportamento é o expresso a seguir:

```
?- intercalada([], [], Z).  
Z = [] ?  
yes
```

```
?- intercalada([a], [], Z).
```

```
Z = [a] ?
```

```
yes
```

```
?- intercalada([], [1,2], Z).
```

```
Z = [1, 2] ?
```

```
yes
```

```
?- intercalada([], [1,2,3,4], Z).
```

```
Z = [1, 2, 3, 4] ?
```

```
yes
```

```
?- intercalada([a,b], [1,2,3,4], Z).
```

```
Z = [a, 1, b, 2, 3, 4] ?
```

```
yes
```

```
?- intercalada([a,b,c,d], [1,2,3,4], Z).
```

```
Z = [a, 1, b, 2, c, 3, d, 4] ?
```

```
yes
```

Exercício 3

Construir um predicado em Prolog, denominado “insercao_ord”, o qual relaciona um item (I), uma lista ordenada (L1 – possivelmente vazia) de itens da mesma natureza de I, e uma lista ordenada (L2 – não vazia). A lista L2 contém todos os itens de L1 e ainda o item I em uma posição que mantenha L2 de forma ordenada. Seu comportamento é o expresso abaixo:

```
?- insercao_ord( 6, [], Z).
```

```
Z = [6] ?
```

```
yes
```

```
?- insercao_ord( 6, [5], Z).
```

```
Z = [5, 6] ?
```

```
yes
```

```
?- insercao_ord( 6, [3, 4, 5, 8, 9, 10], Z).
```

```
Z = [3, 4, 5, 6, 8, 9, 10] ?
```

```
yes
```

Exercício 4

Construir um predicado em Prolog, denominado “ordenada”, o qual relaciona duas listas (L1 e L2 – possivelmente vazias) de itens numéricos, onde L2 tem exatamente os mesmos itens de L1, porém de maneira ordenada. Seu comportamento é o expresso abaixo:

```
?- ordenada([ ], Z).
```

```
Z = [ ] ?
```

```
yes
```

```
?- ordenada([11 , 10, 9], Z).
```

```
Z = [9, 10, 11] ?
```

```
yes
```

```
?- ordenada([11, 10, 9, 5, 7, 2, 4, 8, 9, 2, 3, 45, 7], Z).
```

```
Z = [2, 2, 3, 4, 5, 7, 7, 8, 9, 9, 10, 11, 45] ?
```

```
yes
```

```
?- ordenada([ 5 ], [ 4 ]).
```

```
no
```

```
?- ordenada([ 5, 6, 7 ], [ 4, 5, 6 ]).
```

```
no
```

Informações Gerais

- **Data de entrega:** 25/11
- **Número de integrantes:** individual

Os algoritmos devem ser implementados utilizando a linguagem de programação Prolog. O código deverá estar disponível no repositório em:

- **inteligencia-artificial/trabalho04/trabalho.pl**

Entrega

- Os trabalhos devem ser entregues via **github**¹.
- Criar um repositório chamado **inteligencia-artificial**.
- Adicionar o usuário do professor (**chaua**) aos colaboradores do projeto.
- Criar o arquivo README.md com o curso, disciplina, nome do aluno.
- Submeter no **AVA** os arquivos fonte e a url do repositório.

Referências

Exercícios adaptados da apostila de taplog do prof. Alexandre Direne ².

¹<https://github.com/>

²<http://www.inf.ufpr.br/alex/tap/taplog.pdf>