

# Prova 1 - IA (15/10/2013)

## Questão 1 (20 pontos)

Descreva uma semelhança e uma diferença semântica entre um código imperativista e um código da programação em Lógica. Se facilitar o entendimento, exemplifique com fragmentos de códigos tanto a semelhança como a diferença que você focalizar.

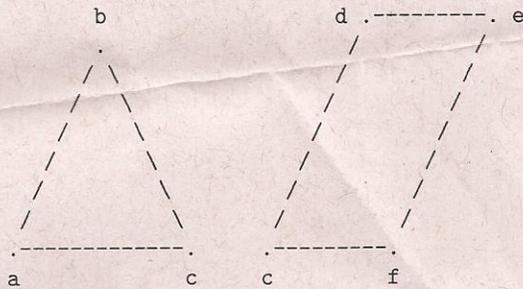
## Questão 2 (20 pontos)

Construir um predicado recursivo em Prolog, denominado `penultimo_elemento`, o qual é uma relação binária capaz de determinar o penúltimo elemento de uma lista de qualquer tamanho. O comportamento do predicado é expresso abaixo.

```
?- penultimo_elemento([y,b,1,k,p,55,7,a,3], E).  
E = a ?  
yes
```

## Questão 3 (30 pontos)

Suponha a categoria das redes de computadores representadas sob a forma de um grafo não direcionado com apenas uma componente conexa. Assuma que essa única componente conexa, por sua vez, é sempre criada a partir de 2 (dois) subgrafos que possuem um único vértice comum, ou seja, ele é uma "articulação" de acordo com a Teoria dos Grafos. Veja o exemplo abaixo, onde o vértice "c" seria a articulação na composição da rede com dois subgrafos.



Construa o predicado `articulacao`, com aridade 3 (três), o qual relaciona 2 (dois) subgrafos da mesma rede com o único vértice comum entre ambos os subgrafos. Veja um exemplo de sua ativação:

```
?- articulacao([[a,b],[b,c],[a,c]],  
               [[c,d],[d,e],[e,f],[c,f]],  
               Articulacao).
```

```
Articulacao = c ?  
yes
```

Para facilitar a definição do predicado `articulacao`, assumo a existência dos predicados `uniao_multipla` e `pertence_a`. Não é necessário construir nenhum deles. O predicado `uniao_multipla` é binário e relaciona um conjunto de conjuntos com o conjunto dos vértices de todos os conjuntos. Veja um exemplo de sua ativação:

```
?- uniao_multipla([[u,v],[v,t],[u,t]], U).  
U = [u,t,v] ?  
yes
```

O predicado `pertence_a` já é bem conhecido. Ele é binário e relaciona um elemento com um conjunto a que esse elemento pertence. Veja exemplos de sua ativação:

```
?- pertence_a(a, [x,y,z]).  
no
```

```
?- pertence_a(E, [x,y,z]).  
X = x ?;  
X = y ?;  
X = z ?;  
no
```

## Questão 4 (30 pontos)

Um "Quadrado Mágico" de 9 (nove) elementos (3 linhas por 3 colunas) pode ser definido como a aplicação dos algarismos de 1 até 9, sem repetição, organizados de forma bi-dimensional, tal que a soma dos elementos de quaisquer das linhas, colunas ou diagonais do quadrado resulte sempre no valor 15 (quinze). Abaixo, é apresentada uma das possíveis combinações de "Quadrado Mágico".

```
2 7 6  
9 5 1  
4 3 8
```

São dados os fatos abaixo, os quais fazem parte da base axiomática do Prolog.

```
val(1).  
val(2).  
val(3).  
val(4).  
val(5).  
val(6).  
val(7).  
val(8).  
val(9).
```

Construir um predicado em Prolog, denominado `um_quadrado_magico`, o qual expressa uma relação de aridade 0 (zero) e é definido com base no predicado "val" (além de outros predicados listados abaixo). Seu comportamento é o seguinte:

```
?- um_quadrado_magico.  
2 7 6  
9 5 1  
4 3 8  
yes
```

Para facilitar a definição do predicado `um_quadrado_magico`, assumo a existência dos seguintes predicados (não é necessário construir nenhum deles):

```
?- todos_sao_diferentes([2,7,6,9,5,1,4,3,8]).  
yes  
?- soma_das_linhas([2,7,6,9,5,1,4,3,8], S).  
S = 15 ?  
yes  
?- soma_das_colunas([2,7,6,9,5,1,4,3,8], S).  
S = 15 ?  
yes  
?- soma_das_diagonais([2,7,6,9,5,1,4,3,8], S).  
S = 15 ?  
yes  
?- matriz_impressa([2,7,6,9,5,1,4,3,8]).  
2 7 6  
9 5 1  
4 3 8  
yes
```