

Pinturas

Quatro crianças, o Aníbal, a Bela, a Carla e o Daniel, pintaram desenhos para oferecer à professora na sua primeira aula do jardim infantil. Cada criança pintou um e um só desenho, utilizando um único instrumento diferente. Os desenhos foram apresentados sequencialmente na aula, de acordo com uma determinada ordem. O objectivo deste grupo consiste em determinar qual o instrumento que cada criança utilizou e em que lugar cada criança mostrou o seu trabalho, a partir das pistas seguintes:

1. O Daniel mostrou o seu desenho em quarto lugar.
2. O Aníbal não mostrou o seu desenho em segundo lugar.
3. A Carla desenhou com o marcador.
4. O desenho mostrado em terceiro lugar foi pintado pela Bela ou pela Carla.
5. Não pode ocorrer simultaneamente que o Aníbal utilize a esferográfica e a Bela utilize o pincel.
6. A criança que utilizou o lápis mostrou o seu trabalho imediatamente após aquele pintado com o pincel.

Recorrendo à programação por conjuntos de resposta, irá produzir neste grupo parte de um programa para o sistema Smodels que lhe permitirá resolver este enigma. Considere o seguinte programa, capturando alguma da informação enunciada:

```
crianca(anibal). instr(marcador). ordem(1..4).
crianca(bela). instr(lapis).
crianca(carla). instr(pincel).
crianca(daniel). instr(esferografica).

desenho(C,I,O) :- crianca(C), instr(I), ordem(O), not n_desenho(C,I,O).
n_desenho(C,I,O) :- crianca(C), instr(I), ordem(O), not desenho(C,I,O).
```

a) Explique sucintamente o objectivo da seguinte regra e restrições:

```
aux(C) :- crianca(C), desenho(C,I,O), instr(I), ordem(O).
:- crianca(C), not aux(C).

:- crianca(C), desenho(C,I1,O1), desenho(C,I2,O2),
   instr(I1), instr(I2), ordem(O1), ordem(O2), neq(I1,I2).
```

b) Indique se o seguinte código Smodels modela correctamente a informação expressa na primeira pista do enunciado, corrigindo-o caso entenda necessário:

```
ic1 :- not desenho(daniel,I,4), instr(I).
:- not ic1.
```

c) Adicione ao seu programa uma ou mais restrições para capturar a informação expressa nas pistas 2 e 3 do enunciado.

d) Apresente uma ou mais restrições que capturem a informação descrita nos pontos 4 e 5 do enunciado. Repare que para o ponto 5 é aceitável obter conjuntos de resposta em que o Aníbal utiliza a esferográfica ou a Bela utiliza o pincel; contudo devem ser excluídos os conjuntos de resposta nos quais se tem simultaneamente o Aníbal a utilizar a esferográfica e a Bela a utilizar o pincel.

e) Recorrendo a uma ou mais restrições, implemente a última pista do enunciado.

f) Refaça o gerador de hipóteses recorrendo à notação de conjuntos.

Torneio de Futebol

Efectuou-se um torneio de futebol envolvendo os seguintes clubes de futebol: Aguiense, Belavista, Cruzense, e Dragoense. O torneio foi efectuado a uma só volta, tendo cada equipa jogado apenas uma vez com cada um dos três adversários. Foram realizados os seguintes seis jogos de futebol:

Aguiense-Dragoense	Cruzense-Aguiense
Belavista-Cruzense	Belavista-Dragoense
Dragoense-Cruzense	Aguiense-Belavista

Sabe-se ainda que:

- i1. O Dragoense teve resultados iguais nos jogos que efectuou com o Aguiense e o Cruzense.
- i2. O Cruzense ganhou ao vencedor do jogo Aguiense-Dragoense.
- i3. O Belavista não ganhou qualquer jogo.
- i4. O Aguiense perdeu com o vencedor do jogo Dragoense-Cruzense.
- i5. O Aguiense empatou com o derrotado do jogo Belavista-Dragoense
- i6. O Cruzense nunca empatou um jogo fora.

Dados os seus conhecimentos de programação por conjuntos de resposta, os organizadores do torneio pediram-lhe ajuda no desenvolvimento de um programa, para o sistema Smodels, que lhes permita saber o resultado de cada jogo. Considere o seguinte programa, representando alguma da informação enunciada:

```
equipa(a) .      jogo(a,d) .      res(1) .
equipa(b) .      jogo(b,c) .      res(x) .
equipa(c) .      jogo(d,c) .      res(2) .
equipa(d) .      jogo(c,a) .
                  jogo(b,d) .
                  jogo(a,b) .

resultado(C,V,R) :- jogo(C,V), res(R), not neg_resultado(C,V,R) .
neg_resultado(C,V,R) :- jogo(C,V), res(R), not resultado(C,V,R) .
```

O predicado **equipa/1** lista as quatro equipas em competição, em que **a** significa Aguiense, **b** designa o clube Belavista, etc. O predicado **jogo/2** descreve os seis jogos realizados, em que o primeiro argumento é a equipa da casa e o segundo argumento a equipa visitante. O predicado **res/1** explicita os três resultados possíveis para um jogo de futebol: **1** – vitória da equipa da casa; **x** – empate; e **2** – vitória da equipa visitante (a que joga fora).

4a) Indique qual a função das duas regras que constam do programa apresentado acima. Quantos conjuntos de resposta existem para esse programa (factos mais regras) ?

4b) Adicione restrições ao programa apresentado, **que garantam** que para cada jogo exista um e um só resultado. Reimplemente o predicado **resultado/2** de uma maneira mais simples.

4c) Compare o efeito das seguintes restrições, indicando os conjuntos de resposta eliminados por cada uma delas.

- a) `:- resultado(a,d,x) .`
- b) `:- not resultado(a,d,1), not resultado(a,d,2) .`

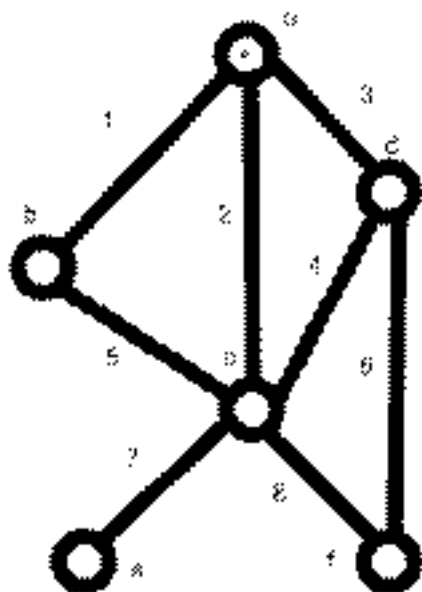
4d) Apresente uma ou mais restrições que capturem a informação descrita na afirmação **i1** do enunciado (repare que num dos jogos o Dragoense joga em casa e noutra é visitante).

4e) Para representar as afirmações **i2** a **i5**, sugere-se a construção do predicado **vencedor/3**, em que uma instância **vencedor(C,V,X)** pertence a um conjunto de resposta quando **X** é o vencedor do jogo **(C,V)**. Defina o predicado **vencedor/3** e, estando definido o predicado **vencedor/3**, implemente o predicado **derrotado/3**.

4f) Apresente restrições que capturem a informação descrita nas afirmações **i2** a **i6**, devendo utilizar os predicados **perdeu/2**, **ganhou/2** e **empatou/2** e quaisquer dos definidos anteriormente, de maneira a que o (único) conjunto de resposta do programa obtido corresponda à (única) solução do problema.

Policciamento

Grupo 1 (Pesquisa)



A última onda de atentados terroristas tem preocupado a opinião pública mundial. Assim, o governo decidiu adoptar a máxima de “um polícia em cada rua” para sossegar os portugueses. Considere o grafo não orientado apresentado na figura, em que os vértices estão identificados por letras e os arcos por números. Cada um dos vértices indica um local onde pode ser colocado um polícia e os arcos uma rua a ser policiada. Dada a contenção orçamental, o objectivo consiste em colocar o menor número de polícias de maneira que toda a rua fique vigiada pelo menos por um polícia numa das suas extremidades.

Por exemplo, uma solução óptima recorre apenas a 3 polícias, colocados nos vértices **a**, **c** e **d**. Repare que um polícia em **a** cobre as ruas 1, 2, e 3; um polícia em **c**, vigia as ruas 2, 4, 5, 7 e 8; enquanto que o polícia em **d**, trata das ruas 3, 4 e 6. Portanto, estamos na presença de uma solução pois todas as ruas se encontram vigiadas pelo menos por um polícia, e algumas delas até por dois agentes da autoridade. Ao adicionarmos um ou mais vértices ao conjunto anterior, continuaremos a obter uma solução mas que obviamente já não será óptima.

Modele este problema para resolução recorrendo a Programação por Conjuntos de Resposta, não se preocupando com a questão da minimalidade.

PintaGrelha

O puzzle “Pinta Grelha” é jogado num quadrado n por n . Para cada linha e cada coluna é dado o número total de casas pintadas. O objectivo do puzzle consiste em descobrir qual a figura pintada na grelha. Considere-se o seguinte puzzle (à esquerda) e uma solução possível (à direita).

	1	4	6	3	3	3
2						
3						
2						
5						
5						
3						

Problema

	1	4	6	3	3	3
2						
3						
2						
5						
5						
3						

Uma solução

Recorrendo à programação por conjuntos de resposta, irá produzir neste grupo um programa para o sistema Smodels que lhe permitirá resolver este tipo de puzzles. Considere o seguinte programa, capturando alguma da informação enunciada:

linha(1..6).	totalLinha(1,2).	totalColuna(1,1).
coluna(1..6).	totalLinha(2,3).	totalColuna(2,4).
casa(L,C) :- linha(L), coluna(C).	totalLinha(3,2).	totalColuna(3,6).
canto(1,1).	totalLinha(4,5).	totalColuna(4,3).
canto(1,6).	totalLinha(5,5).	totalColuna(5,3).
canto(6,1).	totalLinha(6,3).	totalColuna(6,3).
canto(6,6).		

O predicado **totalLinha(Lin,Total)** especifica o número **Total** de casas pintadas na linha **Lin** (numeradas de cima para baixo). Similarmente, o predicado **totalColuna(Col,Total)** especifica o número **Total** de casas pintadas na coluna **Col** (numeradas da esquerda para a direita).

III.1) Recorrendo à programação por conjuntos de resposta gere todas as alternativas correspondendo a que cada casa possa estar pintada ou não. Use o predicado **pintada(Lin,Col)** para indicar esse facto num conjunto de resposta.

III.2) Para cada linha e para cada coluna, o número de casas pintadas deve coincidir com os totais dados no problema.

- Apresente uma solução que garanta esta regra do puzzle, para a linha 1 e para a coluna 1.
- Apresente uma solução geral, válida para todas as linhas e colunas, recorrendo aos factos de **totalLinha/2** e **totalColuna/2**. Note que os limites usados na notação de conjuntos do Smodels podem ser variáveis.

III.3) Indique como representaria cada uma das restrições seguintes em programação por conjuntos de resposta, sem recorrer à sintaxe de conjuntos do sistema SMOELS.

- A casa (1,2) e a casa (2,1) estão pintadas.
- Nenhum dos cantos está pintado (recorra ao predicado **canto/2** para implementar esta restrição).

III.4) Considere os predicados **l/4** e **a/4** definidos por intermédio das regras apresentadas abaixo. Indique o seu significado, assim como o objectivo da restrição de integridade incluída no código.

Nota: A função **abs** do Smodels determina o valor absoluto de um número.

<pre> l(L1,C1,L2,C2) :- casa(L1,C1), casa(L2,C2), pintada(L1,C1), pintada(L2,C2), a(L1, C1, L2, C2). </pre>	<pre> l(L1,C1,L3,C3) :- casa(L1,C1), casa(L2,C2), casa(L3,C3), l(L1,C1,L2,C2), l(L2,C2,L3,C3). </pre>	<pre> a(L,C1,L,C2) :- casa(L,C1), casa(L,C2), eq(abs(C1-C2),1). </pre>	<pre> a(L1,C,L2,C) :- casa(L1,C), casa(L2,C), eq(abs(L1-L2),1). </pre>
<pre> :- casa(L1,C1), casa(L2,C2), pintada(L1,C1), pintada(L2,C2), not l(L1,C1,L2,C2), L1 != L2. :- casa(L1,C1), casa(L2,C2), pintada(L1,C1), pintada(L2,C2), not l(L1,C1,L2,C2), C1 != C2. </pre>			

4. Batalha Naval

O puzzle da Batalha Naval consiste numa grelha quadrada n por n , totais em linhas e em colunas, e uma frota contendo um certo número de navios de comprimentos diversos (a largura dos navios é 1). A grelha inicialmente encontra-se vazia. O objectivo do puzzle é colocar o menor número de navios na grelha obedecendo às seguintes restrições:

1. Não existem dois navios que se tocam (mesmo diagonalmente);
2. O número de casas ocupadas na coluna (linha) i é igual ao total indicado para a coluna (linha) i .

Abaixo pode encontrar uma instância de um puzzle com uma grelha de dimensão 5 com uma frota de 2 navios de comprimento 1, 1 navio de comprimento 2, 1 navio de comprimento 3 e 1 navio de comprimento 4 e nenhum navio de comprimento 5. A solução com 4 navios para o problema encontra-se à direita da figura.

Grelha

					1
					3
					0
					1
					2
2	1	1	2	1	

Frota

2 x					
1 x					
1 x					
1 x					
0 x					

Exemplo de Problema

					1
					3
					0
					1
					2
2	1	1	2	1	

Solução

Matematicamente, a instância pode ser representada por um quádruplo $(5, [1, 3, 0, 1, 2], [2, 1, 1, 2, 1], [2, 1, 1, 1, 0])$ em que a primeira componente denota a dimensão da grelha, a segunda os totais em linhas, a terceira componente os totais em colunas e a última componente a frota. Repare que a soma S dos totais em colunas é sempre igual à soma dos totais em linha, sendo $S = 7$ para o puzzle dado.

Modele o problema em Programação por Conjuntos de Resposta. Utilize quando possível a notação de conjuntos.