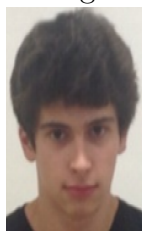


Exercício Prático Nº 2 - Grupo 3 - Sistemas de Representação de Conhecimento e Raciocínio

André Rodrigues Freitas



A74619

João Tiago Pereira Dias



A72095

Joel Tomás Morais



A70841

Sofia Manuela Gomes de Carvalho



A76658

9 de Abril de 2017
Universidade do Minho



Resumo

Neste exercício prático, utilizamos a linguagem de programação em lógica estendida PROLOG para desenvolver um sistema de representação de conhecimento e raciocínio que permite caracterizar o universo de discurso na área da prestação de cuidados de saúde. Ao longo deste relatório são abordados os vários aspectos a ter em conta para o exercício, bem como para a sua resolução.

Conteúdo

1	Introdução	2
2	Preliminares	3
3	Descrição do Trabalho e Análise de Resultados	4
3.1	Conhecimento Positivo	6
3.2	Conhecimento Negativo	7
3.3	Conhecimento Imperfeito através de valores nulos	9
3.3.1	Incerto	9
3.3.2	Impreciso	11
3.3.3	Interdito	13
4	Conclusões e Sugestões	15
5	Bibliografia	16
6	Anexos	17
6.1	Exemplos de execução da aplicação	17

1 Introdução

No segundo exercício prático, recorreremos ao PROLOG para desenvolver um sistema de representação de conhecimento e raciocínio para caracterizar o universo de discurso na área da prestação de cuidados de saúde. Para tal, respeitamos o panorama sugerido no enunciado que é constituído por utentes, cuidados prestados e atos médicos e implementamos várias funcionalidades, funcionalidades essas, na nossa opinião, úteis para a realização e controlo de serviços médicos, e que serão visitadas detalhadamente no decorrer deste mesmo documento. Essas funcionalidades são subjacentes à programação em lógica estendida e à representação de conhecimento imperfeito, recorrendo-se para isso à temática dos valores nulos.

2 Preliminares

Este trabalho prático é um prolongamento do primeiro exercício entregue e remonta igualmente a um pedido do Serviço Nacional de Saúde (SNS) para armazenar e tratar dados relativos a vários aspetos integrantes do SNS através da implementação de várias funcionalidades cruciais que nos foram apresentadas.

Alguns exemplos dessas funcionalidades são registar e remover utentes, mas também representar casos problemáticos em que, por exemplo, algum tipo de anomalia nos dados possa afetar a capacidade de inserção de novo conhecimento.

3 Descrição do Trabalho e Análise de Resultados

No desenvolvimento do código para este segundo exercício prático foram feitas alterações ao código desenvolvido para o primeiro exercício.

Em relação ao primeiro exercício, apenas se mantiveram os predicados inserir, testar, retirar, comprimento, encontrar, evolucao e involucao por serem os necessários para utilizar e testar as temáticas dos conhecimentos negativo e imperfeito. Para além disso, mantiveram-se também todos os invariantes estruturais e referenciais, visto serem cruciais para respeitar os pressupostos do enunciado quando se utilizam os predicados evolucao e involucao.

Como o tema principal deste exercício é a programação em lógica estendida e o conhecimento imperfeito, foi inserido conhecimento na zona do desconhecido e informação incompleta, o que permite representar, além do conhecimento positivo representado até então, conhecimento negativo e imperfeito.

A primeira extensão feita ao código do primeiro exercício prático foi a definição da quantidade de argumentos que define o novo predicado nao, utilizando para isso o *dynamic*. Este predicado tem associado a si um único parâmetro, como é possível ver pela definição apresentada na imagem abaixo.

```
:- dynamic nao/1.
```

Figura 1: Definição inicial

Foram acrescentados dois novos predicados que permitem definir o sistema de inferência, quer para uma única pergunta (demo) quer para várias perguntas (demoLista).

O predicado demo permite obter a resposta a uma questão que seja colocada. A resposta devolvida é "verdadeiro" se existir uma prova de que essa questão é verdadeira, é "falso" se existir uma prova de que essa questão é falsa ou o demo retorna "desconhecido" se não existir uma prova de que a questão não é verdadeira nem falsa.

```
% Extensao do meta-predicado demo: Questao,Resposta -> {V,F}

demo( Questao,verdadeiro ) :-
    Questao.

demo( Questao, falso ) :-
    -Questao.

demo( Questao,desconhecido ) :-
    nao( Questao ),
    nao( -Questao ).
```

Figura 2: Predicado demo

O predicado `demoLista` em tudo se assemelha ao predicado `demo`, exceto pelo facto de que este predicado, apresentado na figura abaixo, retorna uma lista com a resposta a cada pergunta colocada. Este foi o sistema de inferência final, quando foi ponderado recorrer a um sistema que utilizaria a conjunção e a disjunção das respostas dadas às várias perguntas, pois era do interesse da nossa Base de conhecimento saber a resposta específica a cada questão quando dávamos várias questões de uma vez como *input*, algo que não era possível com um sistema de inferência que recorria à conjunção e à disjunção, pois dava apenas uma resposta para todas as questões.

```
% Extensao do meta-predicado demoLista: LQuestao,LResposta -> {V,F}

demoLista( [],[] ).

demoLista( [Questao|L],[Resposta|S] ) :- demo( Questao,Resposta ), demoLista(L,S).
```

Figura 3: Predicado `demoLista`

O predicado `nao`, que é ilustrado na figura abaixo, foi acrescentado para poder representar a negação forte de conhecimento.

```
% Extensao do meta-predicado nao: Questao -> {V,F}

nao( Questao ) :-
    Questao, !, fail.

nao( Questao ).
```

Figura 4: Predicado `nao` (a negação forte)

3.1 Conhecimento Positivo

Neste segundo exercício prático, sentimos a necessidade de aumentar o número de factos na nossa Base de conhecimento, mantendo também os factos já usados no primeiro exercício. Para tal, foram acrescentados os factos que se apresentam nas seguintes figuras.

```
% Extensão do predicado utente: IdUt, Nome, Idade, Morada -> {V,F,D}

utente(8, andre, 21, selva).

utente(2, joao, 21, braga).

utente(15, joel, 21, braganca).

utente(9, sofia, 20, praia).

utente(4, ana, 25, porto).

utente(40,miguel,21,braga).
```

Figura 5: Factos - utente

```
% Extensão do predicado cuidados: IdServ, Descrição, Instituição, Cidade -> {V,F,D}

cuidados(1, cardiologia, hospitalBraga, braga).

cuidados(2, oftalmologia, hospitalBraganca, braganca).

cuidados(3, pediatria, hospitalTrofa, braga).

cuidados(4, neurologia, hospitalGuimaraes, guimaraes).

cuidados(6, dermatologia, hospitalTrofa, braga).

cuidados(7, dentista, denteCintilante, guimaraes).
```

Figura 6: Factos - cuidados


```
% Extensão do predicado atoMedico: Data, IdUt, IdServ, Custo -> {V,F,D}

atoMedico(01-01-2001, 9, 1, 30).

atoMedico(14-10-2009, 2, 2, 18).

atoMedico(21-01-2014, 15, 4, 23).

atoMedico(30-07-2010, 8, 3, 50).

atoMedico(30-08-2015, 8, 1, 40).

atoMedico(20-11-2016, 9, 2, 20).

atoMedico(27-02-2018, 9, 6, 20).

atoMedico(01-07-2017, 9, 7, 130).
```

Figura 7: Factos - atoMedico

3.2 Conhecimento Negativo

Alterações ao primeiro exercício prático denotam-se primeiramente pelo facto de o conhecimento representado não ser sempre verdadeiro ou falso, recorrendo à negação forte. Para atingir tal propósito, como agora é utilizada uma Base de Conhecimento em vez de uma Base de Dados, temos casos em que o conhecimento é falso, algo que é permitido através da introdução da negação forte.

Neste Pressuposto do Mundo Aberto nada é assumido falso se não estiver declarado como sendo mentira, ou seja, para que o conhecimento seja falso este tem de ser inserido e declarado como falso. São então este tipo de motivos que nos fazem representar explicitamente informação falsa.

Este tipo de conhecimento pode ser inserido por um vasto leque de motivos e será relativo aos factos previamente inseridos.

Um exemplo dessa negação terá como possível origem o falecimento de um dos utentes, algo que é possível visualizar na figura apresentada abaixo.

```
-utente(20,amelia,87,faro).
```

Figura 8: Negação de Utente

Outra hipótese de conhecimento negativo advém de sabendo que um utente decidiu emigrar por motivos de força maior, deixará de estar registado nos cuidados médicos portugueses. Esta situação representada através de conhecimento negativo encontra-se na figura abaixo.

```
-utente(34,alberto,46,porto).
```

Figura 9: Negação de Utente

Um caso recente foi um erro nas máquinas do hospital de Guimarães que

necessitavam de manutenção pois estavam mal tratadas. Como tal, precisavam de ser arrançadas e, assim sendo, afastadas daquele estabelecimento de saúde. Esta reparação das máquinas levou a que os tratamentos que delas necessitavam já não pudessem ser realizados naquelas instalações. A inserção deste conhecimento negativo é verificado na figura apresentada abaixo.

-cuidados(6,urologia,hospitalGuimaraes,guimaraes).

Figura 10: Negação de Cuidados

Devido a um incêndio, o serviço de Urologia, situado na cidade de Guimarães, no Hospital de Guimarães, encontra-se encerrado. Para representar este facto, o conhecimento negativo foi codificado conforme consta na figura abaixo.

-cuidados(5, radiologia, hospitalGuimaraes, guimaraes).

Figura 11: Negação de Cuidados

Por motivos de força maior, já não se poderão realizar os serviços apresentados na figura abaixo, visto que o incêndio referido em cima flagrou a zona onde estes eram prestados e a reparação dos danos causados pelo incêndio acarreta um custo que o hospital não consegue suportar.

-atoMedico(09-09-2017, 9, 5, 25).

-atoMedico(21-09-2017, 4, 5, 30).

Figura 12: Negação de Ato Medico

3.3 Conhecimento Imperfeito através de valores nulos

Existem três tipos de valores nulos que são utilizados para a representação de conhecimento imperfeito, sendo eles:

- Incerto;
- Impreciso;
- Interdito.

Para a demonstração dos nossos exemplos relativos a estas três formas de representação simbólica de informação incerta, imprecisa e interdita, decidimos recriar algumas histórias meramente hipotéticas, de modo a que se perceba melhor o nosso intuito na realização destes tipos de conhecimento imperfeito.

Antes disso, convém referir que existiu a necessidade de formalizar o pressuposto do mundo fechado para os predicados `utente`, `cuidados` e `atoMedico` de modo a poder utilizar valores nulos, sendo possível encontrar isso na imagem abaixo.

```
-utente(IDU,N,ID,M) :- nao(utente(IDU,N,ID,M)), nao(excecao(utente(IDU,N,ID,M))).  
-cuidados(IDS,DES,I,C) :- nao(cuidados(IDS,DES,I,C)), nao(excecao(cuidados(IDS,DES,I,C))).  
-atoMedico(D,IDU,IDS,CUS) :- nao(atoMedico(D,IDU,IDS,CUS)), nao(excecao(atoMedico(D,IDU,IDS,CUS)))
```

Figura 13: Conhecimento imperfeito

3.3.1 Incerto

Recorremos à utilização de valores incertos para a representação de conhecimento, conhecimento este que desconhecemos por completo.

Para proceder à representação de conhecimento nulo incerto, criamos um novo facto em que o(s) parâmetro(s) desconhecido(s) são representados com uma variável que simboliza o desconhecido e de seguida recorremos a uma exceção para indicar a exceção ao parâmetro desconhecido.

- Para a nossa primeira representação deste tipo de conhecimento decidimos representar algo simples, dizendo que apenas sabemos que o utente foi o 10º a ser registado, chama-se Abel e que tem 23 anos de idade mas que, devido a um incêndio, foi perdida parte da sua informação, não sabendo a morada do utente. A representação deste conhecimento está ilustrada na figura abaixo.

```
utente(10,abel,23,xpto023).  
excecao(utente(Id,N,I,M)):-  
    utente(Id,N,I,xpto023).
```

Figura 14: Tipo incerto nº1

- Nesse mesmo incêndio foi recolhida informação relativa ao 20º utente a ser registado, sendo essa informação a seguinte: é um utente de Lisboa, chamado Juliano e que no dia 21 de Janeiro de 2014 pagou um certo valor por uma consulta de Neurologia (Id=10) na cidade de Lisboa. Todos os outros dados são desconhecidos. Este conhecimento incerto apresenta-se na imagem abaixo.

```
utente(20,juliano,xpto123,lisboa).  
excecao(utente(Id,N,I,M)):-  
    utente(Id,N,xpto123,M).  
  
cuidados(10,neurologia,xpto312,lisboa).  
excecao(cuidados(Id,D,I,C)):-  
    cuidados(Id,D,xpto312,C).  
  
atoMedico(21-01-2014,20,10,xpto456).  
excecao(atoMedico(D,IdU,IdS,C)):-  
    atoMedico(D,IdU,IdS,xpto456).
```

Figura 15: Tipo incerto nº2

- Outro caso de perda de informação foi verificado nesse mesmo incêndio, sendo esta perda relativa à paciente número 30 a ser registada. Em relação a esta, apenas se sabe que o seu nome é Susana e que foi a uma consulta de cardiologia (Id=11).

```

utente(30,susana,xpto001,xpto002).

excecao(utente(IdU,N,I,M)):-
    utente(IdU,N,xpto001,xpto002).

cuidados(11,cardiologia,xpto003,xpto004).

excecao(cuidados(IdS,D,I,C)):-
    cuidados(IdS,D,xpto003,xpto004).

atoMedico(xpto2000,30,11,xpto2001).

excecao(atoMedico(D,IdU,IdS,C)):-
    atoMedico(xpto2000,IdU,IdS,xpto2001).

```

Figura 16: Tipo incerto nº3

3.3.2 Impreciso

Este tipo de conhecimento refere-se ao conhecimento que se desconhece mas que, porém, se encontra restringido a um conjunto de hipóteses. Para tal, são utilizados valores incertos para a representação de conhecimento.

Para representar este conhecimento, declaramos exceções para os casos em que se está em dúvida sobre qual a informação verdadeiramente correta.

- Um acontecimento em que nos deparámos com conhecimento impreciso foi quando houve uma falha no sistema e não foi possível apurar se o 2º utente a ser registado, que se chama Miguel, tem 21 anos e reside em Braga, foi a uma consulta de cardiologia (Id=1) ou de pediatria (Id=3) no dia 07-06-2005, sabendo apenas que foi na cidade de Braga e que pagou 40 euros. Para lidar com esta situação, foi construído o código que se apresenta na figura seguinte.

```

excecao(atoMedico(07-06-2005,40,1,40)).

excecao(atoMedico(07-06-2005,40,3,40)).

```

Figura 17: Tipo impreciso nº1

- Um caso recente foi quando, devido a um aumento nos preços de cuidados médicos, a variar entre 5 a 10 euros, o utente João (nº 2), com 21 anos de idade e residente em Braga, não sabe quanto é que será o preço final de uma consulta de oftalmologia (Id=2), a ser realizada no dia 14 de Outubro de 2017, na cidade de Bragança, no Hospital de Bragança, sabendo apenas que inicialmente tinha um custo associado de 18 euros.

```
excecao(atoMedico(14-10-2017,2,2,Custo)):- Custo>=23, Custo<28.
```

Figura 18: Tipo impreciso nº2

- Outro exemplo de conhecimento impreciso acontece quando não se sabe, por exemplo, qual dos quatro utentes que deu entrada no Hospital de Bragança, em Bragança, foi à consulta de Oftalmologia (Id=2). O conhecimento certo que se tem é que o utente número 8 chama-se André, tem 21 anos de idade e veio da Selva, o utente número 2 chama-se João, tem 21 anos de idade e veio de Braga, o utente número 15 chama-se Joel, tem 21 anos de idade e veio de Bragança e, por último, o utente número 9 chama-se Sofia, tem 20 anos de idade e veio da Praia. Para representar todas estas imprecisões, é necessário criar exceções, para todos estes utentes, do ato médico que tem em comum apenas a sua data.

```
excecao(atoMedico(06-04-2017,2,2,50)).
```

```
excecao(atoMedico(06-04-2017,8,2,50)).
```

```
excecao(atoMedico(06-04-2017,9,2,50)).
```

```
excecao(atoMedico(06-04-2017,15,2,50)).
```

Figura 19: Tipo impreciso nº3

- Uma paciente chamada Ana (IdUt=4) marcou uma consulta de Dermatologia, mas não se lembra do dia para que marcou a consulta, estando indecisa entre os dias 05-04-2017, 05-05-2017 e 15-06-2017.

```
excecao(atoMedico(05-04-2017,4,6,55)).
excecao(atoMedico(05-05-2017,4,6,55)).
excecao(atoMedico(15-06-2017,4,6,55)).
```

Figura 20: Tipo impreciso nº4

3.3.3 Interdito

Para representar conhecimento que é desconhecido e que nunca será possível vir a conhecer recorre-se à utilização de valores incertos.

Este conhecimento fica declarado usando uma variável para indicar a incerteza do valor de algum campo e declarando uma exceção para esse predicado. Além disso, é preciso declarar como nula a variável usada para indicar a incerteza e é necessário definir um invariante para garantir que sempre que se procura um predicado com aquela informação incerta não é obtido nenhum resultado.

Um caso onde nos deparámos com este tipo de conhecimento foi quando, após um roubo no edifício principal do SNS, informações acerca do valor pago por um certo utente para um ato médico realizado no dia 4 de abril de 2017 foram perdidas. Como não havia uma cópia de segurança dos atos médicos mais recentes, é impossível saber qual o custo deste ato médico. Para lidar com esta perda, foi inserido na Base de conhecimento o código que se apresenta na figura abaixo.

```
atoMedico(04-04-2017, 2, 1, xpto700).

excecao(atoMedico(D,IDU,IDS,C)) :- atoMedico(D,IDU,IDS,xpto700).

nulo(xpto700).

+atoMedico(D,IDU,IDS,C) :: (encontrar(Cs,(atoMedico(04-04-2017, 2, 1, Cs),nao(nulo(Cs))),S), comprimento(S,N), N==0).
```

Figura 21: Tipo interdito nº1

Num certo dia, o sistema informático de registo de atos médicos não se encontrava operacional, o que impediu o médico de registar o ato médico de um dado utente. Dias depois, quando o sistema voltou a estar operacional, o médico já não se recordava da data na qual não pôde registar o ato médico no sistema e visto que só foi possível voltar a usar o sistema alguns dias depois do dia do ato médico, o médico não conseguirá lembrar-se nunca do dia. A representação desta situação de conhecimento interdito foi introduzida, estando ilustrada na figura abaixo.

```

atoMedico(xpto800, 4, 7, 30).
execcao(atoMedico(D,IDU,IDS,C)) :- atoMedico(xpto800,IDU,IDS,C).
nulo(xpto800).
+atoMedico(D,IDU,IDS,C) :: (encontrar(Ds,(atoMedico(Ds, 4, 7, 30),nao(nulo(Ds))),S), comprimento(S,N), N==0).

```

Figura 22: Tipo interdito nº2

Outra situação recente que surgiu foi o caso do João que atingiu a idade adulta mas frequentava a pediatria porque gostava mais do médico e tinha uma estatura baixa, logo passava despercebido. Certo dia alguém lhe perguntou a idade e pediram-lhe a identificação. Após verem a identificação, perceberam que tinha mais de 18 anos e, por isso, proibiram-no de ir à pediatria.

```

atoMedico(xpto900,2,3,xpto999).
execcao(atoMedico(D,IDU,IDS,C)) :- atoMedico(xpto900,IDU,IDS,xpto999).
nulo(xpto900).
nulo(xpto999).
+atoMedico(D,IDU,IDS,C) :: (encontrar((Ds,Cs),(atoMedico(Ds, 2, 3, Cs),nao(nulo(Ds)),nao(nulo(Cs))),S), comprimento(S,N), N==0).

```

Figura 23: Tipo interdito nº3

4 Conclusões e Sugestões

Fazendo uma avaliação global da realização deste projeto, não nos depa-ramos com grandes dificuldades, criando assim um projeto, na nossa opinião, fluído e bem pensado, passivo de uma boa compreensão por parte de elementos externos à realização do mesmo e da própria linguagem utilizada (**PROLOG**).

Em primeiro lugar, o maior e talvez mais relevante obstáculo deste projeto seria uma imaginação um pouco fértil por parte dos elementos do grupo para a criação de casos que, de forma concisa e estruturada, pudessem representar o conhecimento imperfeito, que é o pilar estrutural do exercício em mãos.

Porém, superando este percalço, todo o trabalho segue uma linha de raciocínio bastante semelhante à seguida nas aulas sendo que assim apenas existe a necessidade de importar essas capacidades aprendidas e utilizá-las de forma útil na resolução deste projeto.

Em suma, fundamentamos as nossas escolhas, o que é notório na própria estrutura do trabalho, e estamos satisfeitos com as conclusões a que chegamos e com o plano de ação que elaboramos sendo que respondemos de forma simples e eficaz aos requisitos propostos. Foram ainda realizados testes para garantir o funcionamento correto dos invariantes que mantêm os dados consistentes nunca obtendo resultados não aceitáveis e redundantes.

5 Bibliografia

- Textos pedagógicos disponibilizados na página da Unidade Curricular;
- “PROLOG: Programming for Artificial Intelligence”, Ivan Bratko;
- “A Inteligência Artificial em 25 Lições”, Hélder Coelho.

6 Anexos

6.1 Exemplos de execução da aplicação

Nesta secção, é possível encontrar exemplos da utilização dos predicados definidos para a implementação das funcionalidades pretendidas. Mostra-se o funcionamento dos predicados `demo` e `demoLista`, de forma a que se perceba o objetivo de cada um deles e as situações em que melhor se enquadram e deve-se destacar ainda a execução dos predicados `evolucao` e `involucao` por utilizarem os vários invariantes definidos inicialmente, mas também novos invariantes que foram sendo inseridos conforme necessário para o conhecimento interdito, sendo que tentamos mostrar ao máximo que estes predicados respeitam os pressupostos desses invariantes.

```
| ?- listing(utente).
utente(8, andre, 21, selva).
utente(2, joao, 21, braga).
utente(15, joel, 21, braganca).
utente(9, sofia, 20, praia).
utente(4, ana, 25, porto).
utente(10, abel, 23, xpto023).
utente(20, juliano, xpto123, lisboa).
utente(30, susana, xpto001, xpto002).
utente(40, miguel, 21, braga).

yes
| ?- evolucao(utente(9,manuela,21,porto)).
no
| ?- listing(utente).
utente(8, andre, 21, selva).
utente(2, joao, 21, braga).
utente(15, joel, 21, braganca).
utente(9, sofia, 20, praia).
utente(4, ana, 25, porto).
utente(10, abel, 23, xpto023).
utente(20, juliano, xpto123, lisboa).
utente(30, susana, xpto001, xpto002).
utente(40, miguel, 21, braga).

yes
| ?- evolucao(utente(11,manuela,21,porto)).
yes
| ?- listing(utente).
utente(8, andre, 21, selva).
utente(2, joao, 21, braga).
utente(15, joel, 21, braganca).
utente(9, sofia, 20, praia).
utente(4, ana, 25, porto).
utente(10, abel, 23, xpto023).
utente(20, juliano, xpto123, lisboa).
utente(30, susana, xpto001, xpto002).
utente(40, miguel, 21, braga).
utente(11, manuela, 21, porto).

yes
```

Figura 24: Exemplo de registar utentes

```

| ?- listing(cuidados).
cuidados(1, cardiologia, hospitalBraga, braga).
cuidados(2, oftalmologia, hospitalBraganca, braganca).
cuidados(3, pediatria, hospitalTrofa, braga).
cuidados(4, neurologia, hospitalGuimaraes, guimaraes).
cuidados(6, dermatologia, hospitalTrofa, braga).
cuidados(7, dentista, denteCintilante, guimaraes).
cuidados(10, neurologia, xpto312, lisboa).
cuidados(11, cardiologia, xpto003, xpto004).

yes
| ?- evolucao(cuidados(1,ortodontia,hospitalTrofa,braga)).
no
| ?- listing(cuidados).
cuidados(1, cardiologia, hospitalBraga, braga).
cuidados(2, oftalmologia, hospitalBraganca, braganca).
cuidados(3, pediatria, hospitalTrofa, braga).
cuidados(4, neurologia, hospitalGuimaraes, guimaraes).
cuidados(6, dermatologia, hospitalTrofa, braga).
cuidados(7, dentista, denteCintilante, guimaraes).
cuidados(10, neurologia, xpto312, lisboa).
cuidados(11, cardiologia, xpto003, xpto004).

yes
| ?- evolucao(cuidados(12,ortodontia,hospitalTrofa,braga)).
yes
| ?- listing(cuidados).
cuidados(1, cardiologia, hospitalBraga, braga).
cuidados(2, oftalmologia, hospitalBraganca, braganca).
cuidados(3, pediatria, hospitalTrofa, braga).
cuidados(4, neurologia, hospitalGuimaraes, guimaraes).
cuidados(6, dermatologia, hospitalTrofa, braga).
cuidados(7, dentista, denteCintilante, guimaraes).
cuidados(10, neurologia, xpto312, lisboa).
cuidados(11, cardiologia, xpto003, xpto004).
cuidados(12, ortodontia, hospitalTrofa, braga).

yes _

```

Figura 25: Exemplo de registrar cuidados

```

| ?- listing(utente).
utente(8, andre, 21, selva).
utente(2, joao, 21, braga).
utente(15, joel, 21, braganca).
utente(9, sofia, 20, praia).
utente(4, ana, 25, porto).
utente(10, abel, 23, xpto023).
utente(20, juliano, xpto123, lisboa).
utente(30, susana, xpto001, xpto002).
utente(40, miguel, 21, braga).

yes
| ?- listing(cuidados).
cuidados(1, cardiologia, hospitalBraga, braga).
cuidados(2, oftalmologia, hospitalBraganca, braganca).
cuidados(3, pediatria, hospitalTrofa, braga).
cuidados(4, neurologia, hospitalGuimaraes, guimaraes).
cuidados(6, dermatologia, hospitalTrofa, braga).
cuidados(7, dentista, denteCintilante, guimaraes).
cuidados(10, neurologia, xpto312, lisboa).
cuidados(11, cardiologia, xpto003, xpto004).

yes
| ?- listing(atoMedico).
atoMedico(1-1-2001, 9, 1, 30).
atoMedico(14-10-2009, 2, 2, 18).
atoMedico(21-1-2014, 15, 4, 23).
atoMedico(30-7-2010, 8, 3, 50).
atoMedico(30-8-2015, 8, 1, 40).
atoMedico(20-11-2016, 9, 2, 20).
atoMedico(27-2-2018, 9, 6, 20).
atoMedico(1-7-2017, 9, 7, 130).
atoMedico(21-1-2014, 20, 10, xpto456).
atoMedico(xpto2000, 30, 11, xpto2001).
atoMedico(4-4-2017, 2, 1, xpto700).
atoMedico(xpto800, 4, 7, 30).
atoMedico(xpto900, 2, 3, xpto999).

yes
| ?- evolucao(atoMedico(20-01-2019,9,11,23)).
yes
| ?- listing(atoMedico).
atoMedico(1-1-2001, 9, 1, 30).
atoMedico(14-10-2009, 2, 2, 18).
atoMedico(21-1-2014, 15, 4, 23).
atoMedico(30-7-2010, 8, 3, 50).
atoMedico(30-8-2015, 8, 1, 40).
atoMedico(20-11-2016, 9, 2, 20).
atoMedico(27-2-2018, 9, 6, 20).
atoMedico(1-7-2017, 9, 7, 130).
atoMedico(21-1-2014, 20, 10, xpto456).
atoMedico(xpto2000, 30, 11, xpto2001).
atoMedico(4-4-2017, 2, 1, xpto700).
atoMedico(xpto800, 4, 7, 30).
atoMedico(xpto900, 2, 3, xpto999).
atoMedico(20-1-2019, 9, 11, 23).

yes _

```

```

| ?- evolucao(atoMedico(10-02-2018,100,7,34)).
no
| ?- listing(atoMedico).
atoMedico(1-1-2001, 9, 1, 30).
atoMedico(14-10-2009, 2, 2, 18).
atoMedico(21-1-2014, 15, 4, 23).
atoMedico(30-7-2010, 8, 3, 50).
atoMedico(30-8-2015, 8, 1, 40).
atoMedico(20-11-2016, 9, 2, 20).
atoMedico(27-2-2018, 9, 6, 20).
atoMedico(1-7-2017, 9, 7, 130).
atoMedico(21-1-2014, 20, 10, xpto456).
atoMedico(xpto2000, 30, 11, xpto2001).
atoMedico(4-4-2017, 2, 1, xpto700).
atoMedico(xpto800, 4, 7, 30).
atoMedico(xpto900, 2, 3, xpto999).
atoMedico(20-1-2019, 9, 11, 23).

yes
| ?- evolucao(atoMedico(10-03-2020,8,1,45)).
no
| ?- listing(atoMedico).
atoMedico(1-1-2001, 9, 1, 30).
atoMedico(14-10-2009, 2, 2, 18).
atoMedico(21-1-2014, 15, 4, 23).
atoMedico(30-7-2010, 8, 3, 50).
atoMedico(30-8-2015, 8, 1, 40).
atoMedico(20-11-2016, 9, 2, 20).
atoMedico(27-2-2018, 9, 6, 20).
atoMedico(1-7-2017, 9, 7, 130).
atoMedico(21-1-2014, 20, 10, xpto456).
atoMedico(xpto2000, 30, 11, xpto2001).
atoMedico(4-4-2017, 2, 1, xpto700).
atoMedico(xpto800, 4, 7, 30).
atoMedico(xpto900, 2, 3, xpto999).
atoMedico(20-1-2019, 9, 11, 23).

yes
| ?- evolucao(atoMedico(20-01-2020,8,100,58)).
no
| ?- listing(atoMedico).
atoMedico(1-1-2001, 9, 1, 30).
atoMedico(14-10-2009, 2, 2, 18).
atoMedico(21-1-2014, 15, 4, 23).
atoMedico(30-7-2010, 8, 3, 50).
atoMedico(30-8-2015, 8, 1, 40).
atoMedico(20-11-2016, 9, 2, 20).
atoMedico(27-2-2018, 9, 6, 20).
atoMedico(1-7-2017, 9, 7, 130).
atoMedico(21-1-2014, 20, 10, xpto456).
atoMedico(xpto2000, 30, 11, xpto2001).
atoMedico(4-4-2017, 2, 1, xpto700).
atoMedico(xpto800, 4, 7, 30).
atoMedico(xpto900, 2, 3, xpto999).
atoMedico(20-1-2019, 9, 11, 23).

yes _

```

Figura 26: Exemplo de registrar atos médicos

```

| ?- listing(atoMedico).
atoMedico(1-1-2001, 9, 1, 30).
atoMedico(14-10-2009, 2, 2, 18).
atoMedico(21-1-2014, 15, 4, 23).
atoMedico(30-7-2010, 8, 3, 50).
atoMedico(30-8-2015, 8, 1, 40).
atoMedico(20-11-2016, 9, 2, 20).
atoMedico(27-2-2018, 9, 6, 20).
atoMedico(1-7-2017, 9, 7, 130).
atoMedico(21-1-2014, 20, 10, xpto456).
atoMedico(xpto2000, 30, 11, xpto2001).
atoMedico(4-4-2017, 2, 1, xpto700).
atoMedico(xpto800, 4, 7, 30).
atoMedico(xpto900, 2, 3, xpto999).

yes
| ?- listing(utente).
utente(8, andre, 21, selva).
utente(2, joao, 21, braga).
utente(15, joel, 21, braganca).
utente(9, sofia, 20, praia).
utente(4, ana, 25, porto).
utente(10, abel, 23, xpto023).
utente(20, juliano, xpto123, lisboa).
utente(30, susana, xpto001, xpto002).
utente(40, miguel, 21, braga).

yes
| ?- involucao(utente(15,joel,21,braganca)).
no
| ?- listing(utente).
utente(8, andre, 21, selva).
utente(2, joao, 21, braga).
utente(9, sofia, 20, praia).
utente(4, ana, 25, porto).
utente(10, abel, 23, xpto023).
utente(20, juliano, xpto123, lisboa).
utente(30, susana, xpto001, xpto002).
utente(40, miguel, 21, braga).
utente(15, joel, 21, braganca).

yes
| ?- evolucao(utente(11,manuel,43,faro)).
yes
| ?- listing(utente).
utente(8, andre, 21, selva).
utente(2, joao, 21, braga).
utente(9, sofia, 20, praia).
utente(4, ana, 25, porto).
utente(10, abel, 23, xpto023).
utente(20, juliano, xpto123, lisboa).
utente(30, susana, xpto001, xpto002).
utente(40, miguel, 21, braga).
utente(15, joel, 21, braganca).
utente(11, manuel, 43, faro).

yes
| ?- involucao(utente(11,manuel,43,faro)).
yes
| ?- listing(utente).
utente(8, andre, 21, selva).
utente(2, joao, 21, braga).
utente(9, sofia, 20, praia).
utente(4, ana, 25, porto).
utente(10, abel, 23, xpto023).
utente(20, juliano, xpto123, lisboa).
utente(30, susana, xpto001, xpto002).
utente(40, miguel, 21, braga).
utente(15, joel, 21, braganca).

yes
_

```

Figura 27: Exemplo de remover utentes

```

| ?- listing(atoMedico).
atoMedico(1-1-2001, 9, 1, 30).
atoMedico(14-10-2009, 2, 2, 18).
atoMedico(21-1-2014, 15, 4, 23).
atoMedico(30-7-2010, 8, 3, 50).
atoMedico(30-8-2015, 8, 1, 40).
atoMedico(20-11-2016, 9, 2, 20).
atoMedico(27-2-2018, 9, 6, 20).
atoMedico(1-7-2017, 9, 7, 130).
atoMedico(21-1-2014, 20, 10, xpto456).
atoMedico(xpto2000, 30, 11, xpto2001).
atoMedico(4-4-2017, 2, 1, xpto700).
atoMedico(xpto800, 4, 7, 30).
atoMedico(xpto900, 2, 3, xpto999).

yes
| ?- listing(cuidados).
cuidados(1, cardiologia, hospitalBraga, braga).
cuidados(2, oftalmologia, hospitalBraganca, braganca).
cuidados(3, pediatria, hospitalTrofa, braga).
cuidados(4, neurologia, hospitalGuimaraes, guimaraes).
cuidados(6, dermatologia, hospitalTrofa, braga).
cuidados(7, dentista, denteCintilante, guimaraes).
cuidados(10, neurologia, xpto312, lisboa).
cuidados(11, cardiologia, xpto003, xpto004).

yes
| ?- involucao(cuidados(1,cardiologia,hospitalBraga,braga)).
no
| ?- listing(cuidados).
cuidados(2, oftalmologia, hospitalBraganca, braganca).
cuidados(3, pediatria, hospitalTrofa, braga).
cuidados(4, neurologia, hospitalGuimaraes, guimaraes).
cuidados(6, dermatologia, hospitalTrofa, braga).
cuidados(7, dentista, denteCintilante, guimaraes).
cuidados(10, neurologia, xpto312, lisboa).
cuidados(11, cardiologia, xpto003, xpto004).
cuidados(1, cardiologia, hospitalBraga, braga).

yes
| ?- evolucao(cuidados(12,ortodontia,hospitalTrofa,braga)).
yes
| ?- listing(cuidados).
cuidados(2, oftalmologia, hospitalBraganca, braganca).
cuidados(3, pediatria, hospitalTrofa, braga).
cuidados(4, neurologia, hospitalGuimaraes, guimaraes).
cuidados(6, dermatologia, hospitalTrofa, braga).
cuidados(7, dentista, denteCintilante, guimaraes).
cuidados(10, neurologia, xpto312, lisboa).
cuidados(11, cardiologia, xpto003, xpto004).
cuidados(1, cardiologia, hospitalBraga, braga).
cuidados(12, ortodontia, hospitalTrofa, braga).

yes
| ?- involucao(cuidados(12,ortodontia,hospitalTrofa,braga)).
yes
| ?- listing(cuidados).
cuidados(2, oftalmologia, hospitalBraganca, braganca).
cuidados(3, pediatria, hospitalTrofa, braga).
cuidados(4, neurologia, hospitalGuimaraes, guimaraes).
cuidados(6, dermatologia, hospitalTrofa, braga).
cuidados(7, dentista, denteCintilante, guimaraes).
cuidados(10, neurologia, xpto312, lisboa).
cuidados(11, cardiologia, xpto003, xpto004).
cuidados(1, cardiologia, hospitalBraga, braga).

yes _

```

Figura 28: Exemplo de remover cuidados


```

| ?- listing(atoMedico).
atoMedico(1-1-2001, 9, 1, 30).
atoMedico(14-10-2009, 2, 2, 18).
atoMedico(21-1-2014, 15, 4, 23).
atoMedico(30-7-2010, 8, 3, 50).
atoMedico(30-8-2015, 8, 1, 40).
atoMedico(20-11-2016, 9, 2, 20).
atoMedico(27-2-2018, 9, 6, 20).
atoMedico(1-7-2017, 9, 7, 130).
atoMedico(21-1-2014, 20, 10, xpto456).
atoMedico(xpto2000, 30, 11, xpto2001).
atoMedico(4-4-2017, 2, 1, xpto700).
atoMedico(xpto800, 4, 7, 30).
atoMedico(xpto900, 2, 3, xpto999).

yes
| ?- involucao(atoMedico(1-1-2001,9,1,30)).
yes
| ?- listing(atoMedico).
atoMedico(14-10-2009, 2, 2, 18).
atoMedico(21-1-2014, 15, 4, 23).
atoMedico(30-7-2010, 8, 3, 50).
atoMedico(30-8-2015, 8, 1, 40).
atoMedico(20-11-2016, 9, 2, 20).
atoMedico(27-2-2018, 9, 6, 20).
atoMedico(1-7-2017, 9, 7, 130).
atoMedico(21-1-2014, 20, 10, xpto456).
atoMedico(xpto2000, 30, 11, xpto2001).
atoMedico(4-4-2017, 2, 1, xpto700).
atoMedico(xpto800, 4, 7, 30).
atoMedico(xpto900, 2, 3, xpto999).

yes _

```

Figura 29: Exemplo de remover atos médicos

```

| ?- listing(atoMedico).
atoMedico(1-1-2001, 9, 1, 30).
atoMedico(14-10-2009, 2, 2, 18).
atoMedico(21-1-2014, 15, 4, 23).
atoMedico(30-7-2010, 8, 3, 50).
atoMedico(30-8-2015, 8, 1, 40).
atoMedico(20-11-2016, 9, 2, 20).
atoMedico(27-2-2018, 9, 6, 20).
atoMedico(1-7-2017, 9, 7, 130).
atoMedico(21-1-2014, 20, 10, xpto456).
atoMedico(xpto2000, 30, 11, xpto2001).
atoMedico(4-4-2017, 2, 1, xpto700).
atoMedico(xpto800, 4, 7, 30).
atoMedico(xpto900, 2, 3, xpto999).

yes
| ?- involucao(atoMedico(4-4-2017,2,1,xpto700)).
yes
| ?- listing(atoMedico).
atoMedico(1-1-2001, 9, 1, 30).
atoMedico(14-10-2009, 2, 2, 18).
atoMedico(21-1-2014, 15, 4, 23).
atoMedico(30-7-2010, 8, 3, 50).
atoMedico(30-8-2015, 8, 1, 40).
atoMedico(20-11-2016, 9, 2, 20).
atoMedico(27-2-2018, 9, 6, 20).
atoMedico(1-7-2017, 9, 7, 130).
atoMedico(21-1-2014, 20, 10, xpto456).
atoMedico(xpto2000, 30, 11, xpto2001).
atoMedico(xpto800, 4, 7, 30).
atoMedico(xpto900, 2, 3, xpto999).

yes
| ?- evolucao(atoMedico(4-4-2017,2,1,80)).
no
| ?- listing(atoMedico).
atoMedico(1-1-2001, 9, 1, 30).
atoMedico(14-10-2009, 2, 2, 18).
atoMedico(21-1-2014, 15, 4, 23).
atoMedico(30-7-2010, 8, 3, 50).
atoMedico(30-8-2015, 8, 1, 40).
atoMedico(20-11-2016, 9, 2, 20).
atoMedico(27-2-2018, 9, 6, 20).
atoMedico(1-7-2017, 9, 7, 130).
atoMedico(21-1-2014, 20, 10, xpto456).
atoMedico(xpto2000, 30, 11, xpto2001).
atoMedico(xpto800, 4, 7, 30).
atoMedico(xpto900, 2, 3, xpto999).

yes _

```

Figura 30: Exemplo de utilização do invariante para o 1º exemplo de conhecimento interdito

```

| ?- listing(atoMedico).
atoMedico(1-1-2001, 9, 1, 30).
atoMedico(14-10-2009, 2, 2, 18).
atoMedico(21-1-2014, 15, 4, 23).
atoMedico(30-7-2010, 8, 3, 50).
atoMedico(30-8-2015, 8, 1, 40).
atoMedico(20-11-2016, 9, 2, 20).
atoMedico(27-2-2018, 9, 6, 20).
atoMedico(1-7-2017, 9, 7, 130).
atoMedico(21-1-2014, 20, 10, xpto456).
atoMedico(xpto2000, 30, 11, xpto2001).
atoMedico(4-4-2017, 2, 1, xpto700).
atoMedico(xpto800, 4, 7, 30).
atoMedico(xpto900, 2, 3, xpto999).

yes
| ?- involucao(atoMedico(xpto800,4,7,30)).
yes
| ?- listing(atoMedico).
atoMedico(1-1-2001, 9, 1, 30).
atoMedico(14-10-2009, 2, 2, 18).
atoMedico(21-1-2014, 15, 4, 23).
atoMedico(30-7-2010, 8, 3, 50).
atoMedico(30-8-2015, 8, 1, 40).
atoMedico(20-11-2016, 9, 2, 20).
atoMedico(27-2-2018, 9, 6, 20).
atoMedico(1-7-2017, 9, 7, 130).
atoMedico(21-1-2014, 20, 10, xpto456).
atoMedico(xpto2000, 30, 11, xpto2001).
atoMedico(4-4-2017, 2, 1, xpto700).
atoMedico(xpto900, 2, 3, xpto999).

yes
| ?- evolucao(atoMedico(3-4-2017,4,7,30)).
no
| ?- listing(atoMedico).
atoMedico(1-1-2001, 9, 1, 30).
atoMedico(14-10-2009, 2, 2, 18).
atoMedico(21-1-2014, 15, 4, 23).
atoMedico(30-7-2010, 8, 3, 50).
atoMedico(30-8-2015, 8, 1, 40).
atoMedico(20-11-2016, 9, 2, 20).
atoMedico(27-2-2018, 9, 6, 20).
atoMedico(1-7-2017, 9, 7, 130).
atoMedico(21-1-2014, 20, 10, xpto456).
atoMedico(xpto2000, 30, 11, xpto2001).
atoMedico(4-4-2017, 2, 1, xpto700).
atoMedico(xpto900, 2, 3, xpto999).

yes _

```

Figura 31: Exemplo de utilização do invariante para o 2º exemplo de conhecimento interdito

```

| ?- listing(atoMedico).
atoMedico(1-1-2001, 9, 1, 30).
atoMedico(14-10-2009, 2, 2, 18).
atoMedico(21-1-2014, 15, 4, 23).
atoMedico(30-7-2010, 8, 3, 50).
atoMedico(30-8-2015, 8, 1, 40).
atoMedico(20-11-2016, 9, 2, 20).
atoMedico(27-2-2018, 9, 6, 20).
atoMedico(1-7-2017, 9, 7, 130).
atoMedico(21-1-2014, 20, 10, xpto456).
atoMedico(xpto2000, 30, 11, xpto2001).
atoMedico(4-4-2017, 2, 1, xpto700).
atoMedico(xpto800, 4, 7, 30).
atoMedico(xpto900, 2, 3, xpto999).

yes
| ?- involucao(atoMedico(xpto900,2,3,xpto999)).
yes
| ?- listing(atoMedico).
atoMedico(1-1-2001, 9, 1, 30).
atoMedico(14-10-2009, 2, 2, 18).
atoMedico(21-1-2014, 15, 4, 23).
atoMedico(30-7-2010, 8, 3, 50).
atoMedico(30-8-2015, 8, 1, 40).
atoMedico(20-11-2016, 9, 2, 20).
atoMedico(27-2-2018, 9, 6, 20).
atoMedico(1-7-2017, 9, 7, 130).
atoMedico(21-1-2014, 20, 10, xpto456).
atoMedico(xpto2000, 30, 11, xpto2001).
atoMedico(4-4-2017, 2, 1, xpto700).
atoMedico(xpto800, 4, 7, 30).

yes
| ?- evolucao(atoMedico(5-5-2017,2,3,60)).
no
| ?- listing(atoMedico).
atoMedico(1-1-2001, 9, 1, 30).
atoMedico(14-10-2009, 2, 2, 18).
atoMedico(21-1-2014, 15, 4, 23).
atoMedico(30-7-2010, 8, 3, 50).
atoMedico(30-8-2015, 8, 1, 40).
atoMedico(20-11-2016, 9, 2, 20).
atoMedico(27-2-2018, 9, 6, 20).
atoMedico(1-7-2017, 9, 7, 130).
atoMedico(21-1-2014, 20, 10, xpto456).
atoMedico(xpto2000, 30, 11, xpto2001).
atoMedico(4-4-2017, 2, 1, xpto700).
atoMedico(xpto800, 4, 7, 30).

yes _

```

Figura 32: Exemplo de utilização do invariante para o 3º exemplo de conhecimento interdito

```

| ?- listing(utente).
utente(8, andre, 21, selva).
utente(2, joao, 21, braga).
utente(15, joel, 21, braganca).
utente(9, sofia, 20, praia).
utente(4, ana, 25, porto).
utente(10, abel, 23, xpto023).
utente(20, juliano, xpto123, lisboa).
utente(30, susana, xpto001, xpto002).
utente(40, miguel, 21, braga).

yes
| ?- demo(utente(8, andre, 21, selva), Resp).
Resp = verdadeiro ?
yes
| ?- demo(utente(9, andre, 21, selva), Resp).
Resp = falso ?
yes
_

```

Figura 33: Exemplo de utilização do interpretador demo para conhecimento perfeito

```

| ?- demo(utente(10, abel, 23, braga), Resp).
Resp = desconhecido ?
yes
_

```

Figura 34: Exemplo de utilização do interpretador demo para um exemplo de conhecimento incerto

```

| ?- demo(atoMedico(14-10-2017, 2, 2, 25), Resp).
Resp = desconhecido ?
yes
| ?- demo(atoMedico(14-10-2017, 2, 2, 20), Resp).
Resp = falso ?
yes
_

```

Figura 35: Exemplo de utilização do interpretador demo para um exemplo de conhecimento impreciso

```

| ?- demo(atoMedico(01-04-2017, 4, 7, 30), Resp).
Resp = desconhecido ?
yes
_

```

Figura 36: Exemplo de utilização do interpretador demo para um exemplo de conhecimento interdito

```

| ?- listing(utente).
utente(8, andre, 21, selva).
utente(2, joao, 21, braga).
utente(15, joel, 21, braganca).
utente(9, sofia, 20, praia).
utente(4, ana, 25, porto).
utente(10, abel, 23, xpto023).
utente(20, juliano, xpto123, lisboa).
utente(30, susana, xpto001, xpto002).
utente(40, miguel, 21, braga).

yes
| ?- demoLista([utente(8, andre, 21, selva), utente(9, andre, 21, selva)], Resp).
Resp = [verdadeiro, falso] ?
yes
_

```

Figura 37: Exemplo de utilização do interpretador demoLista para conhecimento perfeito já usado no exemplo do demo

```
| ?- demoLista([utente(10,abel,23,braga),atoMedico(14-10-2017,2.2.25),atoMedico(14-10-2017,2.2.20),atoMedico(01-04-2017,4.7.30)]).Resp).
Resp = [desconhecido,desconhecido,falso,desconhecido] ?
yes _
```

Figura 38: Exemplo de utilização do interpretador demoLista para todas as questões sobre conhecimento imperfeito usadas nos exemplos do demo