

SRCR

TP1 & TP2

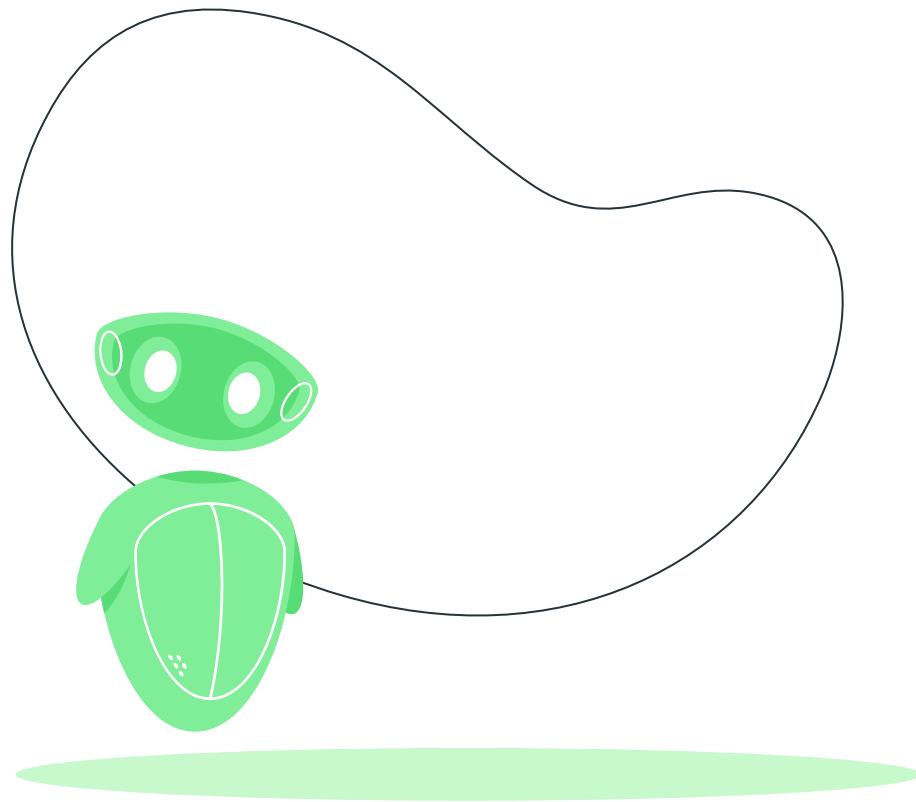
António Santos a83700

Joana A. Gomes a84912

Jorge Vieira a84240

José Magalhães a85852

Paulo R. Pereira a86475



Índice da Apresentação

1

TP1

- Decisões Iniciais
- Base de Conhecimento
- Invariantes
- Adicionar e Remover Conhecimento
- Identificações

2

TP2

- Base de Conhecimento (Conhecimento Perfeito e Imperfeito)
- Evolução do Conhecimento
- Involução do Conhecimento
- Sistema de Inferência



TP1

Programação em Lógica

Decisões Iniciais

No contexto do problema, foram criadas **3 fases de vacinação**:

- **Fase 1** (de 01-01-2020 a 31-03-2021)
 - Profissionais de Saúde;
 - Pessoas com idade igual ou superior a 80 anos;
 - Pessoas com idade igual ou superior a 50 anos e uma das seguintes doenças: {Insuficiência Cardíaca, Doença Coronária, Insuficiência Renal,DPOC}
- **Fase 2** (de 01-04-2020 a 31-08-2021)
 - Profissionais de Saúde
 - Pessoas com idade compreendida entre 50 e 64 anos e uma das seguintes doenças: {Diabetes, Neoplasia maligna ativa, Doença Renal Crónica, Insuficiência Hepática, Hipertensão arterial, Obesidade}
- **Fase 3** (a partir de 31-08-2021)
 - Restantes utentes.

Decisões Iniciais

Foram também definidos critérios que distinguem utentes que foram **vacinados** **indevidamente**.

A toma torna-se indevida se mais do que um destes **critérios** se verificar:

- Tomou numa fase que não lhe correspondia, ou seja, não encaixa nos critérios para a toma nessa fase (Ex: Um utente com menos de 50 anos tomou a vacina na Fase 2)
- Tomou duas vacinas distintas (Ex: Um utente tomou a vacina da Pfizer na primeira fase e da Moderna na segunda fase)
- Tomou mais que duas vacinas.

Base de Conhecimento



utente

#IdUtente, Número Segurança Social, Nome, Género, Data Nascimento, Email, Telefone, Morada, Profissão, [Doenças Crónicas], #IdCentroSaúde \leadsto {V,F}



centro

#IdCentroSaúde, Nome, Morada, Telefone, Email \leadsto {V,F}



staff

#IdStaff, IdCentroSaúde, Nome, email \leadsto {V,F}



vacinação

#staff, #utente, Data, Vacina, Toma \leadsto {V,F}



fase

#IdFase, DataInicio, DataFim \leadsto {V,F}

Dados (parcial)

```
utente('U0','Romulo Mota','22337840788','M','1936-04-22','RomuloMota@outlook.com','962935227','Rua Francisco Augusto Alvim 4700-004 Braga','Biologo',[],'C1').
utente('U1','Goncalo Melo','65105592957','M','1931-09-18','GoncaloMelo1991@hotmail.com','938823950','Travessa 1 de Maio 4700-008 Braga','Geriatra',[],'C2').
utente('U2','Bianca Moura','84874182284','F','1958-10-30','BiancaMoura244@gmail.com','921591637','Rua de Sao Rosendo ( Bispo de Dume ) Dume 4700-008 Braga','Personal trainer',[],'C3').
utente('U3','Vilar Nunes','05052988794','F','1976-02-17','VilarNunes1976@gmail.com','968452327','Rua de Remelhe Dume 4700-008 Braga','Vigilante',[],'C3').
utente('U4','Cinderela Nogueira','50775228700','F','1950-12-08','CinderelaNogueira@outlook.com','924781129','Rua Paulo Vi 4700-004 Braga','Auxiliar de limpeza',[],'C2').
utente('U5','Apolo Amaral','39446661300','M','1980-05-06','ApoloAmaral@hotmail.com','926158761','Avenida dos Descobrimentos Antas 4760-001 Vila Nova de Famalicao','Reporter',[],'Insuficiencia hepatica',[],'C3').
utente('U6','Zelia Abreu','94334540382','F','1995-06-19','ZeliaAbreu@hotmail.com','966727220','Rua do Brasil 4775-001 Cambeses','Eletricista',[],'C2').
utente('U7','Afonso Torres','82144830813','M','1992-12-24','AfonsoTorres@gmail.com','911799897','Rua Jose Albino Costa e Silva Azurem 4800-004 Guimaraes','Profissional de Saude',[],'C2').
utente('U8','Aldo Castro','89885876143','M','1980-04-16','AldoCastro388@hotmail.com','962967073','Rua Doutor Jose Regio Dume 4700-004 Braga','Programador',[],'C2').
utente('U9','Tiburcio Guerreiro','08519097436','M','1978-03-16','TiburcioGuerreiro342@outlook.com','936810784','Rua do Cortinhal 4775-001 Cambeses','Militar',[],'Obesidade',[],'C2').
```



(104 utentes)

```
staff('S0','Leonardina Anjos','12287559265','LeonardinaAnjos49@outlook.com','C2').
staff('S1','Fernando Guerreiro','28567061757','FernandoGuerreiro@outlook.com','C2').
staff('S2','Idelia Paiva','61816783608','IdeliaPaiva@hotmail.com','C1').
```

(19 membros do staff)



```
centro('C1','Centro de saude de Braga','Rua Paulo Vi 4700-004 Braga','219229219','suporte@debraga.com').
centro('C2','Centro de saude do Caranda','Rua de Goa Azurem 4800-004 Guimaraes','246512282','suporte@docaranda.com').
centro('C3','Centro de saude do Cavado','Rua de Sao Paulo 4700-004 Braga','233330348','suporte@docavado.com').
centro('C4','Centro de saude Dr.Paulo Novais','Rua Francisco Ribeiro de Castro Azurem 4800-004 Guimaraes','219282688','suporte@drpaulonovais.com').
```



(4 centros de saúde)

```
vacinacao('S10','U1','2021-06-08','Moderna',1).
vacinacao('S10','U1','2021-06-30','Moderna',2).
vacinacao('S15','U2','2021-07-30','Pfizer',1).
vacinacao('S15','U2','2021-08-19','Pfizer',2).
vacinacao('S5','U3','2021-07-07','Pfizer',1).
vacinacao('S5','U3','2021-07-25','Pfizer',2).
vacinacao('S0','U4','2021-04-03','Moderna',1).
vacinacao('S0','U4','2021-04-24','Moderna',2).
```

(209 registos de vacinação)



```
fase('F1','2020-12-01','2021-03-31').
fase('F2','2021-04-01','2021-08-31').
fase('F3','2021-09-01','9999-12-31').
```



(3 fases)

Invariantes



**Invariantes
de Utente**



**Invariantes
de Centro**



**Invariantes
de Staff**



**Invariantes
de Vacinação**



**Invariantes
de Fase**

Invariantes



Invariantes de Utente

- **Garantir que o ID de um utente é único**

```
+utente(Id,_,_,_,_,_,_,_,_,_) :: (solucoes(Id, utente(Id,_,_,_,_,_,_,_,_), R),  
| length(R, 1)).
```

- **Garantir que utentes com IDs diferentes têm diferente informação (número de segurança social, email, telefone):**

```
+utente(Id,N,NSS,G,DN,E,TLF,M,P,D,IDC) :: (solucoes((Id,NSS,E,TLF), utente(_,_,NSS,_,_,E,TLF,_,_,_,_), R),  
| length(R,1)).
```

Invariantes



Invariantes de Utente

- **Os utentes têm de ter um ID de centro existente**

```
+utente(_____,IdC) :: centro(IdC,_____).
```

- **Garantir que o género só pode ser M ou F**

```
+utente(_,_,_,G,_,_,_,_,_,_) :: generoValido(G).
```

- **Garantir que não se pode eliminar um utente se tiver vacinações marcadas**

```
-utente(Id,_,_,_,_,_,_,_,_,_) :: (findall(Id, vacinacao(_,Id,_,_,_), R),
length(R, 0)).
```

Invariantes



Invariantes de Centro

- **Garantir que o ID de cada centro é único**

```
+centro(Id,_,_,_,_) :: (solucoes(Id, centro(Id,_,_,_,_), R),  
| | | | | | | | | | length(R, 1)).
```

- **Garantir que não se pode eliminar um centro se este tiver uma vacinação associada**

```
-centro(Id,_,_,_,_) :: (findall(sId,staff(sId,_,_,_,Id),R),  
| | | | | | | | | | length(R, 0)).
```

- **Garantir que centros com IDs diferentes têm diferente informação (nome, morada, telefone, email)**

```
+centro(Id,N,_,T,E) :: (solucoes((Id,N,T,E), centro(_,N,_,T,E), R),  
| | | | | | | | | | length(R,1)).
```

Invariantes



Invariantes de Staff

- **Garantir que o ID de cada staff é único**

```
+staff(Id,_,_,_) :: (solucoes(Id, staff(Id,_,_,_), R),  
| | | | | | | | | | length(R, 1)).
```

- **Garantir que não se pode eliminar um staff se este tiver uma vacinação associada**

```
-staff(Id,_,_,_,_) :: (findall(Id, vaccinacao(Id,_,_,_,_), R),  
| | | | | | | | | | length(R, 0)).
```

- **Garantir que staff com IDs diferentes têm diferente informação (ID do centro de saúde, email)**

```
+staff(Id,_,SS,T,C) :: (solucoes((SS,T,C), staff(_,_,SS,T,C), R),  
| | | | | | | | | | length(R,1)).
```

Invariantes



Invariantes de Vacinação

- **Garantir que não há vacinações repetidas:**

```
+vacinacao(Id,B,C,D,E) :: (solucoes(Id, vaccinacao(Id,B,C,D,E), R),  
| | | | | | | | | | length(R, 1)).
```

- **Adicionar uma vacinação da toma 2 requer que a toma 1 já esteja na base de conhecimento**

```
+vacinacao(S,U,_,_,2) :: vaccinacao(S,U,_,_,1).
```

Invariantes



Invariantes de Fase

- Garantir que o ID de cada fase é único

```
+fase(Id,_,_) :: (solucoes(Id, fase(Id,_,_), R),  
| | | | | | | | | | length(R, 1)).
```

- Garantir que a data de início tem de ser antes da data de fim

```
+fase(_,DI,DF) :: comparaDatasStr(DI,DF).
```

```
% Ve se data 1 é antes de data 2 (1 = True , 0 = False)  
comparaDatas(A1,_,_,A2,_,_,0):- A1 > A2.  
comparaDatas(A1,M1,_,A2,M2,_,0):- M1 > M2, A1 >= A2.  
comparaDatas(A1,M1,D1,A2,M2,D2,0):- D1 > D2, A1 >= A2, M1 >= M2.  
comparaDatas(_,_,_,_,_,1).  
  
comparaDatasStr(X1,X2,R):- parseData(X1,A1,M1,D1), parseData(X2,A2,M2,D2), comparaDatas(A1,M1,D1,A2,M2,D2,R).
```

Adicionar Conhecimento

```
novosConhecimentos(T) :- solucoes(I, +T::I, L),  
                           insercao(T),  
                           testaPredicados(L).
```



Utente

```
novosUtentes(Id,N,NISS,G,Dn,E,Tlf,M,P,Dc,IdCs) :-  
    novosConhecimentos(utente(Id,N,NISS,G,Dn,E,Tlf,M,P,Dc,IdCs)).
```



Centro

```
novosCentros(Id,N,M,Tlf,E) :-  
    novosConhecimentos(centro(Id,N,M,Tlf,E)).
```



Staff

```
novosStaff(Id,NSS,IdCs,N,E) :-  
    novosConhecimentos(staff(Id,NSS,IdCs,N,E)).
```



Vacinação

```
novaVacinacao(IdS,IdU,D,V,T) :-  
    novosConhecimentos(vacinacao(IdS,IdU,D,V,T)).
```

Remover Conhecimento

```
removeConhecimento(T) :- solucoes(I, -T::I, L),  
                        remocao(T),  
                        testaPredicados(L).
```



Utente

```
removeUtente(Id) :- removeConhecimento(utente(Id,_,_,_,_,_,_,_,_)).
```



Centro

```
removeCentro(Id) :- removeConhecimento(centro(Id,_,_,_,_)).
```



Staff

```
removeStaff(Id) :- removeConhecimento(staff(Id,_,_,_,_)).
```



Vacinação

```
removeVacinao(IdU,T) :- removeConhecimento(vacinacao(_,IdU,_,_,T)).
```


Identificações

Predicados com o objetivo de identificação de entidades com características específicas que possam ser relevantes para a zona de estudo em questão.

- **Identificar utentes não vacinados (por fases)**
 - Totalmente
 - Parcialmente
- **Identificar utentes vacinados**
- **Vacinação Indevida**
 - Fase não correspondente
 - Toma de duas vacinas diferentes
 - Toma de mais do que duas vacinas

Identificações

- Identificar utentes não vacinados (por fases)

Foram usados alguns predicados auxiliares que selecionam o utentes com determinadas características que os distinguem no que toca à fase de vacinação atribuída, juntamente com os critérios que envolvem a idade.

```
profissao1Fase(IdU) :- utente(IdU,_,_,_,_,_,_,_,P,_),  
                        P = 'Profissional de Saude'.
```

```
doencas1Fase(IdU) :- utente(IdU,_,_,_,_,_,_,_,D,_),  
                        A = ['Insuficiencia renal','Doenca renal cronica','Doenca coronaria','DPOC'],  
                        temElementosComum(A,D).
```

```
doencas2Fase(IdU) :- utente(IdU,_,_,_,_,_,_,_,D,_),  
                        A = ['Diabetes','Neoplasia maligna ativa','Doenca renal cronica','Insuficiencia hepatica','Hipertensao arterial'],  
                        temElementosComum(A,D).
```

Identificações

- Identificar utentes não vacinados (por fases)

Foram usados alguns predicados auxiliares que selecionam o utentes com determinadas características que os distinguem no que toca à fase de vacinação atribuída, juntamente com os **critérios que envolvem a idade**.

```
calcularIdade(IdU,R) :- utente(IdU,_,_,DNasc,_,_,_,_,_),
    split_string(DNasc, "-", "", Sp),
    nth0(0,Sp,Ano), nth0(1,Sp,Mes), nth0(2,Sp,Dia),
    get_date_time_value(year, AnoN), get_date_time_value(month, MesN), get_date_time_value(day, DiaN),
    atom_number(Ano, Year), atom_number(Mes, Month), atom_number(Dia, Day),
    atom_string(AnoN, YearN), atom_number(YearN, YearNN),
    atom_string(MesN, MonthN), atom_number(MonthN, MonthNN),
    atom_string(DiaN, DayN), atom_number(DayN, DayNN),
    Age is (YearNN - Year),
    (Month > MonthNN -> AgeN is Age-1 ;
    Month < MonthNN -> AgeN is Age ;
    (Day > DayNN -> AgeN is Age-1 ;
    AgeN is Age)),
    R = AgeN.
```

Identificações

- **Identificar utentes não vacinados (por fases)**
 - **Totalmente**

Recorremos a um predicado auxiliar **naoVacinados** que identifica **utentes** candidatos à vacinação que **ainda não tomaram nenhuma dose da vacina**.

```
naoVacinados(R) :- listaIdUtentes(X),  
                    naoVacinadosAux(X,[],R).  
  
naoVacinadosAux([],Acc,Acc).  
naoVacinadosAux([H|T],Acc,R) :- (checkVacinado(H,1); checkVacinado(H,2)), naoVacinadosAux(T,Acc,R).  
naoVacinadosAux([H|T],Acc,R) :- naoVacinadosAux(T,[H|Acc],R).
```

Identificações

- **Identificar utentes não vacinados (por fases)**
 - **Totalmente**

De seguida foi criado o predicado **candidatosVacinacaoT** que identifica, dentro desses utentes que ainda não tomaram a vacina, os que correspondem a uma fase específica, baseando-se nos critérios definidos para cada uma.

```
candidatosVacinacaoT(1,R) :- % utentes que ainda nao tomaram nenhuma vacina
                             naoVacinados(S),
                             % utentes candidatos + de 80 anos
                             findall(X,(member(X,S),calcularIdade(X,C), C>=80), I),
                             % utentes candidatos que sao profissionais de saude
                             findall(X,(member(X,S),profissao1Fase(X)), Pr),
                             % utentes candidatos + de 50 anos e com doencas da fase 1
                             findall(X,(member(X,S),doencas1Fase(X),calcularIdade(X,C),C>=50), Do),
                             % juntar as listas
                             append(I,Pr,App), append(App,Do,App2),
                             % eliminar ids repetidos
                             repetidos(App2,R).
```

! O mesmo para a 2ª e 3ª fase e os respetivos critérios.

candidatosVacinacaoT(2,R)

candidatosVacinacaoT(3,R)

Identificações

- **Identificar utentes não vacinados (por fases)**

- **Parcialmente**

Identificar os **utentes que apenas tomaram a primeira dose da vacina**, ou seja, que ainda não estão totalmente vacinados, faltando a toma da segunda dose. O novo predicado auxiliar para ter este universo de pessoas é o **vacinadosSoUmaDose** abaixo.

```
vacinadosSoUmaDose(R) :- % encontrar os ids de utentes candidatos
                           solucoes(IdU,utente(IdU,_,_,_,_,_,_,_,_,_),V),
                           % encontrar o id dos que fizeram a toma 1
                           solucoes(IdU, vacinacao(_,IdU,_,_,1), T1),
                           % encontrar o id dos que fizeram a toma 2
                           solucoes(IdU, vacinacao(_,IdU,_,_,2), T2),
                           % ver os que pertencem a uma e nao ao outro
                           findall(X,(member(X,V),member(X,T1),\+member(X,T2)),R).
```

Identificações

- **Identificar utentes não vacinados (por fases)**

- **Parcialmente**

A lógica anterior repete-se, identificando os utentes que só tomaram uma dose em cada uma das fases.

```
candidatosVacinacaoP(1,R) :- % utentes que so tomaram a dose 1
    vacinadosSoUmaDose(S),
    % utentes candidatos + de 80 anos
    findall(X,(member(X,S),calcularIdade(X,C), C>=80), I),
    % utentes candidatos que sao profissionais de saude
    findall(X,(member(X,S),profissao1Fase(X), Pr),
    % utentes candidatos + de 50 anos e com doencas da fase 1
    findall(X,(member(X,S),doencas1Fase(X),calcularIdade(X,C),C>=50), Do),
    % juntar as listas
    append(I,Pr,App), append(App,Do,App2),
    % eliminar os que sao da fase 1
    repetidos(App2,R).
```

! O mesmo para a 2ª e 3ª fase e os respectivos critérios.

candidatosVacinacaoP(2,R)

candidatosVacinacaoP(3,R)

Identificações

- **Identificar utentes vacinados**

Identificar, de forma global (não referente a fases específicas), os **utentes que já tomaram uma ou duas doses da vacina**.

Para além do predicado **vacinadosSoUmaDose**, que já foi apresentado, foi também criado o predicado **vacinadosDuasDoses**, que identifica todos os utentes que tomaram tanto a primeira como a segunda dose.

```
vacinadosDuasDoses(R) :- solucoes(UId,vacinacao(A,UId,B,C,1),L1),  
                           solucoes(UId,vacinacao(A,UId,B,C,2),L2),  
                           intersecao(L1,L2,R).
```


Identificações

● Vacinação Indevida

- **Fase não correspondente**

vacinacaoFaseErrada, que recorre ao auxiliar **vacinacaoFaseErradaUtente**, que por sua vez recorre aos auxiliares **checkFaseUtente** e **pertenceFase**, que serão apresentados de seguida.

```
vacinacaoFaseErrada(R) :- listaIdUtentes(X),  
    |   |   |   |   |   |   |  
    vacinacaoFaseErradaAux(X,[],R).  
  
vacinacaoFaseErradaAux([],Acc,Acc).  
vacinacaoFaseErradaAux([H|T],Acc,R) :- (vacinacaoFaseErradaUtente(H,X),X = 1), vacinacaoFaseErradaAux(T,Acc,R).  
vacinacaoFaseErradaAux([H|T],Acc,R) :- vacinacaoFaseErradaAux(T,[H|Acc],R).  
  
%Ser vacinado numa fase que não lhe corresponde  
vacinacaoFaseErradaUtente(UId,X) :- (checkFaseUtente(UId,R1),  
    |   |   |   |   |   |   |  
    pertenceAFase(UId,R2)),  
    (R1 \= R2 -> X = 0; X = 1).
```

Identificações

- **Vacinação Indevida**

- **Fase não correspondente**

- **pertenceAFase**

Verifica a que fase de vacinação o utente pertence.

```
pertenceAFase(IDU,R) :- profissao(IDU,Prof),  
                        calcularIdade(IDU,Idade),  
                        (((doencas1Fase(IDU), Idade >= 50); Idade >= 80; Prof == 'Profissional de Saude') -> R is 1;  
                        ((doencas2Fase(IDU), Idade >= 50, Idade <= 64); Idade >= 65) -> R is 2;  
                        R is 3).
```

- **checkFaseUtente**

Compara com a data em que foi vacinado e verifica se é válido.

```
checkFase(D,R):- fase1DataI(DI1), comparaDatasStr(D,DI1,A1),  
                 fase1DataF(DF1), comparaDatasStr(D,DF1,A2),  
                 fase2DataI(DI2), comparaDatasStr(D,DI2,A3),  
                 fase2DataF(DF2), comparaDatasStr(D,DF2,A4),  
                 fase3DataI(DI3), comparaDatasStr(D,DI3,A5),  
                 fase3DataF(DF3), comparaDatasStr(D,DF3,A6),  
                 ((A1==0,A2==1) -> R is 1 ;  
                 (A3==0,A4==1) -> R is 2 ;  
                 (A5==0,A6==1) -> R is 3 ;  
                 R is 0).
```

%Verifica em fase é que um Utente foi vacinado

```
checkFaseUtente(UId,R):- vacinacao(_,UId,D,_,1), checkFase(D,R).
```

Identificações

- **Vacinação Indevida**

- **Toma de duas vacinas diferentes**

```
%Haver uma diferença de vacinas da toma 1 para a 2 (duas vacinas diferentes)
diferente(R):-
    solucoes(U_ID,vacinaDiferente(U_ID),R).

vacinaDiferente(U_ID):-
    vacinacao(_,U_ID,_,V1,1),
    vacinacao(_,U_ID,_,V2,2),
    V1 \= V2.
```

- **Toma de mais do que duas vacinas**

```
%Tomar mais do que duas doses
tomouMaisDe2Doses(R) :- listaIdUtentes(X),
    |
    |
    |
    |
    |
    |
    tomouMaisDe2DosesAux(X,[],R).

tomouMaisDe2DosesAux([],Acc,Acc).
tomouMaisDe2DosesAux([H|T],Acc,R) :- (quantasVacinas(H, N), N<3), tomouMaisDe2DosesAux(T,Acc,R).
tomouMaisDe2DosesAux([H|T],Acc,R) :- tomouMaisDe2DosesAux(T,[H|Acc],R).
```

TP2

Programação em lógica estendida
e Conhecimento imperfeito

PMF e PMA



utente



staff



vacinação



fase

Pressuposto do Mundo Fechado (PMF)

Qualquer utente, staff, vacinação ou fase que não esteja na base de conhecimento e que não contenha nenhuma exceção são considerados conhecimento falso.



centro

Pressuposto do Mundo Aberto (PMA)

Apenas os casos explicitamente negativos são considerados falsos. Isto faz sentido uma vez que nem todos os centros de saúde estão ligados à vacinação contra o COVID.

Conhecimento Perfeito

Conhecimento Perfeito Positivo

```
% #IdUte, Nome, Nr Ute, Genero, Data Nascimento, Email, Telefone, Morada, Profissao, [Doencas Cronicas], #IdCentroSaude
ute('U0','Romulo Mota','22337840788','M','1936-04-22','RomuloMota@outlook.com','962935227','Rua Francisco Augusto Alvim 4700-004 Braga','Biologo',[],'C1').
ute('U1','Goncalo Melo','65105592957','M','1931-09-18','GoncaloMelo1991@hotmail.com','938823950','Travessa 1 de Maio 4700-008 Braga','Geriatra',[],'C2').
ute('U5','Apolo Amaral','39446661300','M','1980-05-06','ApoloAmaral@hotmail.com','926158761','Avenida da Liberdade 4760-001 Vila Nova de Famalicao','Reporter',['Insuficiencia hepatica'],'C2').
ute('U7','Afonso Torres','82144830813','M','1992-12-24','AfonsoTorres@gmail.com','911799897','Rua Jose Albino Costa e Silva Azurem 4800-004 Guimaraes','Profissional de Saude',[],'C2').
ute('U8','Aldo Castro','89885876143','M','1980-04-16','AldoCastro388@hotmail.com','962967073','Rua Doutor Jose Regio Dume 4700-004 Braga','Programador',[],'C2').
ute('U9','Tiburcio Guerreiro','08519097436','M','1978-03-16','TiburcioGuerreiro342@outlook.com','936810784','Rua do Cortinhal 4775-001 Cambeses','Militar',['Obesidade'],'C2').
ute('U11','Florisbela Carvalho','38557360298','F','1969-02-01','FlorisbelaCarvalho1969@gmail.com','910843987','Rua Dona Teresa Aldao 4800-004 Guimaraes','Tecnico em edificacoes',[],'C4').
ute('U14','Tatiano Carvalho','39247286162','M','1960-12-30','TatianoCarvalho@outlook.com','921622230','Praceta do Viajante Azurem 4800-004 Guimaraes','Biologo',[],'C2').
ute('U15','Indro Leal','45346361911','M','1949-07-08','IndroLeal1949@gmail.com','929206797','Rua de Sao Lourenco da Ordem 4700-004 Braga','Webmaster',[],'C1').
ute('U17','Dinarda Matias','29224473940','F','1974-10-09','DinardaMatias1974@outlook.com','915914158','Largo do Rego Dume 4700-008 Braga','DevOps',[],'C4').
ute('U18','Marcilene Goncalves','77919813012','M','1989-12-04','MarcileneGoncalves305@hotmail.com','924527670','Rua do Carvalhal Dume 4700-005 Braga','Perito criminal',['Doenca coronaria'],'C1').
ute('U19','Milena Vicente','39280061650','F','2004-06-04','MilenaVicente@gmail.com','969590100','Travessa General Humberto Delgado 4760-001 Vila Nova de Famalicao','Jovem Aprendiz',[],'C2').
```



```
% #IdStaff, Nome, NSS, Email, IdCentroSaude
staff('S0','Leonardina Anjos','12287559265','LeonardinaAnjos49@outlook.com','C2').
staff('S1','Fernando Guerreiro','28567061757','FernandoGuerreiro@outlook.com','C2').
staff('S2','Idelia Paiva','61816783608','IdeliaPaiva@hotmail.com','C1').
staff('S3','Bianca Baptista','55616211371','BiancaBaptista@hotmail.com','C2').
staff('S7','Juliao Abreu','54101439180','JuliaoAbreu@gmail.com','C4').
staff('S10','Juliano Campos','62975266565','JulianoCampos1991@hotmail.com','C2').
```



```
% #IdFase, DataInicio, DataFim
fase('F1','2020-12-01','2021-03-31').
fase('F2','2021-04-01','2021-08-31').
fase('F3','2021-09-01','9999-12-31').
```



```
% #IdCentroSaude, Nome, Morada, Telefone, Email
centro('C1','Centro de saude de Braga','Rua Paulo Vi 4700-004 Braga','219229219','suporte@debraga.com').
centro('C2','Centro de saude do Caranda','Rua de Goa 4800-004 Guimaraes','246512282','suporte@docaranda.com').
centro('C4','Centro de saude Dr.Paulo Novais','Rua Francisco Ribeiro de Castro 4800-004 Guimaraes','219282688','suporte@drpaulonovais.com').
```



```
% #IdStaff, #IdUte, Data, Vacina, Tosa
vacinacao('S10','U1','2021-06-08','Moderna',1).
vacinacao('S10','U1','2021-06-30','Moderna',2).
vacinacao('S15','U2','2021-07-30','Pfizer',1).
vacinacao('S15','U2','2021-08-19','Pfizer',2).
vacinacao('S5','U3','2021-07-07','Pfizer',1).
vacinacao('S5','U3','2021-07-25','Pfizer',2).
vacinacao('S0','U4','2021-04-03','Moderna',1).
vacinacao('S0','U4','2021-04-24','Moderna',2).
vacinacao('S0','U7','2021-06-15','Astrazeneca',1).
vacinacao('S0','U7','2021-07-02','Astrazeneca',2).
vacinacao('S10','U8','2021-07-21','Sputnik V',1).
vacinacao('S10','U8','2021-08-08','Sputnik V',2).
vacinacao('S10','U9','2021-07-05','Pfizer',1).
vacinacao('S10','U9','2021-07-27','Pfizer',2).
vacinacao('S0','U10','2021-07-21','Sputnik V',1).
vacinacao('S0','U10','2021-08-13','Sputnik V',2).
vacinacao('S2','U12','2021-07-16','Moderna',1).
vacinacao('S2','U12','2021-08-05','Moderna',2).
vacinacao('S4','U13','2021-05-11','Astrazeneca',1).
vacinacao('S4','U13','2021-05-30','Astrazeneca',2).
vacinacao('S5','U14','2021-06-30','Sputnik V',1).
vacinacao('S5','U14','2021-07-23','Sputnik V',2).
vacinacao('S3','U17','2021-07-27','Pfizer',1).
```



Conhecimento Perfeito

- **Conhecimento Perfeito Negativo**

- Existem duas formas de ter conhecimento negativo: **negação forte** e **negação por falha**.
- Devido ao PMF, a forma como os predicados **utente**, **staff**, **vacinação** e **fase** estão implementados garante, implicitamente, conhecimento negativo, uma vez que qualquer predicado que não esteja na base de conhecimento com respeito aos mesmos, é conhecimento negativo.

Conhecimento Perfeito

● Conhecimento Perfeito Negativo

```
-utente('U16','Gino Campos','21515788464','M','2004-03-23','GinoCampos@gmail.com',920820869,'Lugar da Ordem 4700-008 Braga','Piloto de aviao',[],'C4').  
-utente('U21','Rodrigo Matos','90131842525','M','1996-06-04','RodrigoMatos@gmail.com',930344386,'Largo da Confeiteira Dume 4700-005 Braga','Cientista de dados',[],'C1').  
  
-staff('S4','Conceicao Lopes','21645651780','ConceicaoLopes2@outlook.com','C2').  
-staff('S8','Dominico Goncalves','65543192708','DominicoGoncalves@hotmail.com','C1').  
  
-centro('C3','Centro de saude do Cavado','Rua de Sao Paulo 4700-004 Braga','233330348','suporte@docavado.com').  
  
-vacinacao('S3','U5','2021-06-08','Sputnik V',1).  
-vacinacao('S7','U11','2021-06-19','Pfizer',1).
```



**Por exemplo, um
centro de saúde
que tenha fechado
(deixado de existir)**

Conhecimento Imperfeito

- **Conhecimento Imperfeito Incerto**

Conhecimento desconhecido dentro de um conjunto indeterminado ou **ilimitado** de hipóteses

- **Conhecimento Imperfeito Impreciso**

O conjunto de hipóteses para um campo do qual não se sabe o valor é um conjunto **limitado**

- **Conhecimento Imperfeito Interdito**

O conhecimento é desconhecido e não é permitido conhecê-lo

Conhecimento Imperfeito

- **Conhecimento Imperfeito Incerto**

A título de exemplo, uma utente em que o **email** é **desconhecido**:

```
utente('U13','Adelia Amaral','41527730111','F','1944-09-26',email_desconhecido,910419566,'Rua Doutor Joao Afonso Almeida Azurem 4800-004 Guimaraes','Bioquimico',[],'C2').  
excecao(utente(Id,N,Nu,G,DN,_,T,M,P,DC,IdCentro)) :- utente(Id,N,Nu,G,DN,email_desconhecido,T,M,P,DC,IdCentro).
```

- Questionar se o e-mail da utente Adelia Amaral é "adeliamaral@gmail.com" dá **desconhecido**.
- Mas perguntar se é do sexo masculino ('M') daria **falso**.

Consideramos também situações em que o número de **telefone** ou a **morada do utente** são **desconhecidos**, e também em que o **e-mail** de um membro do **staff** é **desconhecido**.

No que toca às outras entidades, o grupo concluiu que nenhum dos campos se encaixava em casos de conhecimento imperfeito. dado que no contexto do problema de vacinação saber com certeza esses parâmetros é ou essencial ou é muito relevante.

Conhecimento Imperfeito

- Conhecimento Imperfeito Impreciso

A título de exemplo, suponhamos que foi guardado o registo de duas moradas distintas de um **utente**, e é **desconhecido qual a sua morada** de residência atual, dentro desse **conjunto limitado de duas moradas**.

```
% Utente com ID U12 com duas moradas possíveis
```

```
excecao(utente('U12','Biana Andrade','03294496759','F','1982-05-02','BianaAndrade@hotmail.com',932477556,'Rua de Sao Martinho Dume 4700-008 Braga','Padeiro',[],'C1')).  
excecao(utente('U12','Biana Andrade','03294496759','F','1982-05-02','BianaAndrade@hotmail.com',932477556,'Rua do Monte 4700-009 Braga','Padeiro',[],'C1')).
```

Consideramos também situações em que, para um **utente**, existe um “*mix-up*” com mais do que um número de **telemóvel** ou **email**, e a situação em que o **gênero** é desconhecido, sendo conhecimento imperfeito pois apenas pode ser ‘M’ ou ‘F’.

Foi seguido o mesmo raciocínio para membros do *staff*.

Conhecimento Imperfeito

- Conhecimento Imperfeito Interdito

Para conhecimento perfeito interdito, consideramos a situação hipotética se um utente que não usa tecnologias e que, portanto, torna o seu campo de email impossível de conhecer.

```
% Utente com ID U20 cujo email é impossível de saber
utente('U20','Delmiro Lopes','36606379205','M','1956-05-23',email_impossivel,927819307,'Rua Monte de Baixo 4705-001 Arentim','Cerimonialista',[],'C1').
excecao(utente(Id,N,Nu,G,Dn_,T,M,P,D,Cs)) :- utente(Id,N,Nu,G,Dn,email_impossivel,T,M,P,D,Cs).
nulointerdito(email_impossivel).
+utente(Id,N,Nu,G,Dn,E,T,M,P,D,Cs) :: (solucoes((Id,N,Nu,G,Dn,E,T,M,P,D,Cs), (utente('U20','Delmiro Lopes','36606379205','M','1956-05-23',email_impossivel,927819307,'Rua Monte de Baixo 4705-001 Arentim','Cerimonialista',[],'C1'),
length(R,0)).
```

Foi seguido o mesmo raciocínio para um membro do *staff*.

```
% Membro do Staff com ID S5 cujo email é impossível de saber
staff('S5','Isandro Azevedo','41601362411','IsandroAzevedo107@gmail.com','C1').
excecao(utente(Id,N,Nss_,Cs)) :- utente(Id,N,Nss,email_impossivel,Cs).
nulointerdito(email_impossivel).
+staff('S5','Isandro Azevedo','41601362411','IsandroAzevedo107@gmail.com','C1') :: (solucoes((_,_,_,_), (staff('S5','Isandro Azevedo','41601362411','IsandroAzevedo107@gmail.com','C1'), nao_nulointerdito(email_impossivel))), R),
length(R,0)).
```

Invariantes

- Adicionar ou remover conhecimento exige procedimentos que controlem essas mesmas ações.
- Devido de no TP2 serem inseridos novos tipos de conhecimento, foi necessário **criar novos invariantes** e **adaptar alguns já criados**.

**Invariantes
Universais**



**Invariantes
de Utente**



**Invariantes
de Staff**



**Invariantes
de Fase**



**Invariantes
de Centro**



**Invariantes
de Vacinação**

Invariantes

- Invariantes Universais

Invariantes que dizem respeito a todos os predicados que possam ser inseridos na nossa base de conhecimento.

- Não existir conhecimento positivo a contradizer conhecimento negativo, e vice-versa.

```
+T :: nao(-T).  
+(-T) :: nao(T).
```

- Não existir conhecimento redundante.

```
% Garante que não há exceções repetidas  
+(excecao(T)) :: (solucoes(T, excecao(T), R),  
                 comprimento(R, 1)).
```

```
% Garante que não há conhecimento perfeito positivo repetido  
+T :: (solucoes(T, T, R),  
       comprimento(R, 1)).  
  
% Garante que não há conhecimento perfeito negativo repetido  
+(-T) :: (solucoes(T, -T, R),  
          comprimento(R, 1)).
```

Invariantes

- Como referido, foram criados invariantes relativos a todo o tipo de entidades.
- De seguida, a título de exemplo e de forma sugestiva, apresentamos alguns exemplos de **Invariantes de Utente** e **Invariantes de Vacinação**.

Invariantes



Invariantes de Utente

- No que diz respeito à **inserção de conhecimento**, há que garantir que...

... o ID de cada utente é único

```
+utente(Id,_,_,_,_,_) :: (solucoes(Id, utente(Id,_,_,_,_,_), R),  
comprimento(R, 1)).  
+(-utente(Id,_,_,_,_,_) :: (solucoes(Id, -utente(Id,_,_,_,_,_), R),  
comprimento(R, 1)).
```

... o género é 'M' ou 'F'

```
+utente(_,_,G,_,_,_) :: generoValido(G).  
+(-utente(_,_,G,_,_,_) :: genderValido(G)).
```

... utentes com IDs diferente têm diferente informação

```
+utente(Id,_,NSS,_,E,TLF,_) :: (solucoes((Id,NSS,E,TLF), utente(_,_,NSS,_,E,TLF,_,_), R),  
comprimento(R,1)).  
+(-utente(Id,_,NSS,_,E,TLF,_) :: (solucoes((Id,NSS,E,TLF), -utente(_,_,NSS,_,E,TLF,_,_), R),  
comprimento(R,1)).
```

... utentes têm um ID de um centro existente

```
+utente(_,_,_,_,_,_,IdC) :: centro(IdC,_,_,_,_,_).  
+(-utente(_,_,_,_,_,_,IdC) :: centro(IdC,_,_,_,_,_)).
```

- No que toca à **remoção de conhecimento**, há que garantir que **não se pode eliminar um utente que tenha vacinações marcadas**.

```
-utente(Id,_,_,_,_,_) :: (findall(Id, vacinacao(_,Id,_,_), R),  
comprimento(R, 0)).
```


Invariantes



Invariantes de Vacinação

- No que diz respeito à **inserção de conhecimento**, há que garantir que...

... adicionar uma vacinação da toma 2 requer que a toma 1 já esteja na base de conhecimento e que ambas as doses sejam do mesmo tipo

```
+vacinacao(_,U,_,V,2) :: vacinacao(_,U,_,V,1).
+~vacinacao(_,U,_,V,2) :: vacinacao(_,U,_,V,1).
```

... a vacinação é efetuada por um membro da staff que esteja na base de conhecimento

```
+vacinacao(S,_,_,_) :: staff(S,_,_,_).
+(-vacinacao(S,_,_,_)) :: staff(S,_,_,_).
```

... a vacinação é efetuada num utente que esteja na base de conhecimento

```
+vacinacao(,_U,_,_) :: utente(U,_,_,_,_,_,_,_).  
+(-vacinacao(,_U,_,_)) :: utente(U,_,_,_,_,_,_,_).
```

... a vacinação é efetuada por um membro da staff que trabalhe no centro onde o utente está inscrito

```
+vacinacao(S,U,_,__) :: (staff(S,_,__,C), utente(U,_,_,_,_,_,_,_,_,_)).  
+(-vacinacao(S,U,_,__) :: (staff(S,_,__,C), utente(U,_,_,_,_,_,_,_,_,_)).
```

Evolução do Conhecimento

Evolução do Conhecimento Perfeito

- Evolução de Conhecimento Perfeito Positivo
- Evolução de Conhecimento Perfeito Negativo

Evolução do Conhecimento Imperfeito

- Evolução de Conhecimento Imperfeito Incerto
- Evolução de Conhecimento Imperfeito Impreciso
- Evolução de Conhecimento Imperfeito Interdito

Evolução do Conhecimento

- **Evolução do Conhecimento Perfeito**

Para a evolução de conhecimento, abordando usamos um predicado *novoConhecimento* desenvolvido no TP1 e trocamos a sua designação para **evolucaoC**.

```
evolucaoC(T) :- solucoes(I, +T::I, L),  
               insercao(T),  
               testaPredicados(L).
```

```
% Procura todas as soluções de uma questão  
solucoes(T,Q,S) :- findall(T,Q,S).  
  
% Inserção de conhecimento  
insercao(Q) :- assert(Q).  
insercao(Q) :- retract(Q), !, fail.  
  
% Testa todos os predicados numa lista  
testaPredicados([]).  
testaPredicados([I|L]) :- I, testaPredicados(L).
```

(Funções auxiliares)

Para o **Conhecimento Perfeito Positivo** usamos este predicado acima, mas para o **Negativo** criamos uma especificação deste:

```
evolucaoC(T,conheNeg) :- solucoes(I, +(-T)::I, L),  
                        insercao(-T),  
                        testaPredicados(L).
```

Evolução do Conhecimento

● Evolução do Conhecimento Perfeito

Um exemplo de evolução de Conhecimento Perfeito Positivo:

```
?- listing(utente(_,_)).
:- dynamic utente/11.

utente('U0', 'Romulo Mota', '22337840788', 'M', '1936-04-22', 'RomuloMota@outlook.com', '962935227', 'Rua Francisco Augusto Alvim 4700-004 Braga', 'Biólogo', []).
utente('U1', 'Goncalo Melo', '65105592957', 'M', '1931-09-18', 'GoncaloMelo1991@hotmail.com', '938823950', 'Travessa 1 de Maio 4700-008 Braga', 'Geriatra', []).
utente('U5', 'Apolo Amaral', '39446661300', 'M', '1980-05-06', 'ApoloAmaral@hotmail.com', '926158761', 'Avenida da Liberdade 4760-001 Vila Nova de Famalicão', 'F'.
utente('U7', 'Afonso Torres', '82144830813', 'M', '1992-12-24', 'AfonsoTorres@gmail.com', '911799897', 'Rua Jose Albino Costa e Silva Azurem 4800-004 Guimaraes'.
utente('U8', 'Aldo Castro', '89885876143', 'M', '1980-04-16', 'AldoCastro388@hotmail.com', '962967073', 'Rua Doutor Jose Regio Dume 4700-004 Braga', 'Programador'.
utente('U9', 'Tiburcio Guerreiro', '08519097436', 'M', '1978-03-16', 'TiburcioGuerreiro342@outlook.com', '936810784', 'Rua do Cortinhal 4775-001 Cambeses', 'Mili'.
utente('U11', 'Florisbela Carvalho', '38557360298', 'F', '1969-02-01', 'FlorisbelaCarvalho1969@gmail.com', '910843987', 'Rua Dona Teresa Aldao 4800-004 Guimaraes'.
utente('U14', 'Tatiano Carvalho', '39247286162', 'M', '1960-12-30', 'TatianoCarvalho@outlook.com', '921622230', 'Praçeta do Viajante Azurem 4800-004 Guimaraes'.
utente('U15', 'Indro Leal', '45346361911', 'M', '1949-07-08', 'IndroLeal1949@gmail.com', '929206797', 'Rua de Sao Lourenço da Ordem 4700-004 Braga', 'Webmaster'.
utente('U17', 'Dinarda Matias', '29224473940', 'F', '1974-10-09', 'DinardaMatias1974@outlook.com', '915914158', 'Largo do Rego Dume 4700-008 Braga', 'DevOps', []).
utente('U18', 'Marcilene Gonçalves', '77919813012', 'M', '1989-12-04', 'MarcileneGoncalves305@hotmail.com', '924527670', 'Rua do Carvalhal Dume 4700-005 Braga'.
utente('U19', 'Milena Vicente', '39280061650', 'F', '2004-06-04', 'MilenaVicente@gmail.com', '969590100', 'Travessa General Humberto Delgado 4760-001 Vila Nova d'.
utente('U13', 'Adelia Amaral', '41527730111', 'F', '1944-09-26', 'email_desconhecido', '910419566', 'Rua Doutor Joao Afonso Almeida Azurem 4800-004 Guimaraes', 'Bi'.
utente('U4', 'Cinderela Nogueira', '50775228700', 'F', '1950-12-08', 'CinderelaNogueira@outlook.com', 'tlf_desconhecido', 'Rua Paulo Vi 4700-004 Braga', 'Auxilio'.
utente('U10', 'Jansenio Figueiredo', '02069412938', 'M', '1983-02-02', 'JansenioFigueiredo218@outlook.com', '914112112', 'acurada_desconhecida', 'Comissario de bord'.
utente('U20', 'Delairo Lopes', '36606379205', 'M', '1956-05-23', 'email_impossivel', '927819307', 'Rua Monte de Baixo 4705-001 Arentin', 'Cerimonialista', []). 'C1'.

true.

?- evolucaoC(utente('U2004', 'Julio Romao', '123212', 'M', '1919-07-22', 'julio@gmail.com', '92323430883', 'Terra', 'Gamer', [], C2)).
true.

?- listing(utente(_,_)).
:- dynamic utente/11.

utente('U0', 'Romulo Mota', '22337840788', 'M', '1936-04-22', 'RomuloMota@outlook.com', '962935227', 'Rua Francisco Augusto Alvim 4700-004 Braga', 'Biólogo', []).
utente('U1', 'Goncalo Melo', '65105592957', 'M', '1931-09-18', 'GoncaloMelo1991@hotmail.com', '938823950', 'Travessa 1 de Maio 4700-008 Braga', 'Geriatra', []).
utente('U5', 'Apolo Amaral', '39446661300', 'M', '1980-05-06', 'ApoloAmaral@hotmail.com', '926158761', 'Avenida da Liberdade 4760-001 Vila Nova de Famalicão', 'F'.
utente('U7', 'Afonso Torres', '82144830813', 'M', '1992-12-24', 'AfonsoTorres@gmail.com', '911799897', 'Rua Jose Albino Costa e Silva Azurem 4800-004 Guimaraes'.
utente('U8', 'Aldo Castro', '89885876143', 'M', '1980-04-16', 'AldoCastro388@hotmail.com', '962967073', 'Rua Doutor Jose Regio Dume 4700-004 Braga', 'Programador'.
utente('U9', 'Tiburcio Guerreiro', '08519097436', 'M', '1978-03-16', 'TiburcioGuerreiro342@outlook.com', '936810784', 'Rua do Cortinhal 4775-001 Cambeses', 'Mili'.
utente('U11', 'Florisbela Carvalho', '38557360298', 'F', '1969-02-01', 'FlorisbelaCarvalho1969@gmail.com', '910843987', 'Rua Dona Teresa Aldao 4800-004 Guimaraes'.
utente('U14', 'Tatiano Carvalho', '39247286162', 'M', '1960-12-30', 'TatianoCarvalho@outlook.com', '921622230', 'Praçeta do Viajante Azurem 4800-004 Guimaraes'.
utente('U15', 'Indro Leal', '45346361911', 'M', '1949-07-08', 'IndroLeal1949@gmail.com', '929206797', 'Rua de Sao Lourenço da Ordem 4700-004 Braga', 'Webmaster'.
utente('U17', 'Dinarda Matias', '29224473940', 'F', '1974-10-09', 'DinardaMatias1974@outlook.com', '915914158', 'Largo do Rego Dume 4700-008 Braga', 'DevOps', []).
utente('U18', 'Marcilene Gonçalves', '77919813012', 'M', '1989-12-04', 'MarcileneGoncalves305@hotmail.com', '924527670', 'Rua do Carvalhal Dume 4700-005 Braga'.
utente('U19', 'Milena Vicente', '39280061650', 'F', '2004-06-04', 'MilenaVicente@gmail.com', '969590100', 'Travessa General Humberto Delgado 4760-001 Vila Nova d'.
utente('U13', 'Adelia Amaral', '41527730111', 'F', '1944-09-26', 'email_desconhecido', '910419566', 'Rua Doutor Joao Afonso Almeida Azurem 4800-004 Guimaraes', 'Bi'.
utente('U4', 'Cinderela Nogueira', '50775228700', 'F', '1950-12-08', 'CinderelaNogueira@outlook.com', 'tlf_desconhecido', 'Rua Paulo Vi 4700-004 Braga', 'Auxilio'.
utente('U10', 'Jansenio Figueiredo', '02069412938', 'M', '1983-02-02', 'JansenioFigueiredo218@outlook.com', '914112112', 'acurada_desconhecida', 'Comissario de bord'.
utente('U20', 'Delairo Lopes', '36606379205', 'M', '1956-05-23', 'email_impossivel', '927819307', 'Rua Monte de Baixo 4705-001 Arentin', 'Cerimonialista', []). 'C1'.
utente('U2004', 'Julio Romao', '123212', 'M', '1919-07-22', 'julio@gmail.com', '92323430883', 'Terra', 'Gamer', [], _).

true.
```

Evolução do Conhecimento

- Evolução do Conhecimento Imperfeito
 - Conhecimento Imperfeito Incerto

Exemplo de evolução de conhecimento Imperfeito Incerto (morada desconhecida):

```
?- listing(utente(-----)).
:- dynamic utente/11.

utente('U0', 'Romulo Mota', '22337840788', 'M', '1936-04-22', 'RomuloMota@outlook.com', '962935227', 'Rua Francisco Augusto Alvina 4700-004 Braga', 'Biologo', [], 'C1').
utente('U1', 'Goncalo Melo', '65105592957', 'M', '1931-09-18', 'GoncaloMelo1991@hotmail.com', '938823950', 'Travessa 1 de Maio 4700-008 Braga', 'Geriatra', [], 'C2').
utente('U5', 'Apolo Aaaraal', '39446661300', 'M', '1980-05-06', 'ApoloAaaraal@hotmail.com', '926158761', 'Avenida da Liberdade 4760-001 Vila Nova de Fanelicaao', 'Reporter', ['Insuficiencia hepatica'], 'C2').
utente('U7', 'Afonso Torres', '92144930813', 'M', '1992-12-24', 'AfonsoTorres@gmail.com', '91179897', 'Rua Jose Albino Costa e Silva Azurem 4800-004 Guimaraes', 'Profissional de Saude', [], 'C2').
utente('U8', 'Aldo Castro', '98985876143', 'M', '1980-04-16', 'AldoCastro388@hotmail.com', '962967073', 'Rua Doutor Jose Regio Duse 4700-004 Braga', 'Programador', [], 'C2').
utente('U9', 'Tiburcio Guerreiro', '08519097436', 'M', '1978-03-16', 'TiburcioGuerreiro342@outlook.com', '936810784', 'Rua do Cortinhal 4775-001 Cambeses', 'Militar', ['Obesidade'], 'C2').
utente('U11', 'Florisbela Carvalho', '38557360298', 'F', '1969-02-01', 'FlorisbelaCarvalho1969@gmail.com', '910843987', 'Rua Dona Teresa Aldao 4800-004 Guimaraes', 'Tecnico em edificacoes', [], 'C4').
utente('U14', 'Tatiano Carvalho', '39242786162', 'M', '1960-12-30', 'TatianoCarvalho@outlook.com', '921622230', 'Praceta do Viajante Azurem 4800-004 Guimaraes', 'Biologo', [], 'C2').
utente('U15', 'Indro Leal', '45346361911', 'M', '1949-07-08', 'IndroLeal1949@gmail.com', '929206797', 'Rua de Sao Lourenco da Ordea 4700-004 Braga', 'Webmaster', [], 'C1').
utente('U17', 'Dinarda Matias', '29224473940', 'F', '1974-10-09', 'DinardaMatias1974@outlook.com', '915914158', 'Largo do Rego Duse 4700-008 Braga', 'DevOps', [], 'C4').
utente('U18', 'Marcilene Gonçalves', '77919813012', 'M', '1989-12-04', 'MarcileneGoncalves305@hotmail.com', '924527670', 'Rua do Carvalho Duse 4700-005 Braga', 'Perito criminal', ['Doenca coronaria'], 'C1').
utente('U19', 'Milena Vicente', '39280061650', 'F', '2004-06-04', 'MilenaVicente@gmail.com', '969590100', 'Travessa General Humberto Delgado 4760-001 Vila Nova de Fanelicaao', 'Jovem Aprendiz', [], 'C2').
utente('U13', 'Adelia Aaaraal', '41527730111', 'F', '1944-09-26', 'email_desconhecido', '910419566', 'Rua Doutor Joao Afonso Almeida Azurem 4800-004 Guimaraes', 'Bioquimico', [], 'C2').
utente('U4', 'Cinderela Nogueira', '60752287000', 'F', '1950-12-08', 'CinderelaNogueira@outlook.com', 'tlf_desconhecido', 'Rua Paulo Vi 4700-004 Braga', 'Auxiliar de limpeza', [], 'C2').
utente('U10', 'Jansenio Figueiredo', '02069412938', 'M', '1983-02-02', 'JansenioFigueiredo218@outlook.com', '914112112', 'morada_desconhecida', 'Comissario de bordo', [], 'C2').
utente('U20', 'Delairio Lopes', '36606379205', 'M', '1956-05-23', 'email_impossivel', '927819307', 'Rua Monte de Baixo 4705-001 Arentim', 'Cerimonialista', [], 'C1').

true.

?- evoluciono(utente('U2004', 'Julio Romao', '123212', 'M', '1919-07-22', 'julio@gmail.com', '92323430883', 'morada_desconhecida', 'Ganer', [], 'C2'), utente, conheImpInc, morada).
true.

?- listing(utente(-----)).
:- dynamic utente/11.

utente('U0', 'Romulo Mota', '22337840788', 'M', '1936-04-22', 'RomuloMota@outlook.com', '962935227', 'Rua Francisco Augusto Alvina 4700-004 Braga', 'Biologo', [], 'C1').
utente('U1', 'Goncalo Melo', '65105592957', 'M', '1931-09-18', 'GoncaloMelo1991@hotmail.com', '938823950', 'Travessa 1 de Maio 4700-008 Braga', 'Geriatra', [], 'C2').
utente('U5', 'Apolo Aaaraal', '39446661300', 'M', '1980-05-06', 'ApoloAaaraal@hotmail.com', '926158761', 'Avenida da Liberdade 4760-001 Vila Nova de Fanelicaao', 'Reporter', ['Insuficiencia hepatica'], 'C2').
utente('U7', 'Afonso Torres', '92144930813', 'M', '1992-12-24', 'AfonsoTorres@gmail.com', '91179897', 'Rua Jose Albino Costa e Silva Azurem 4800-004 Guimaraes', 'Profissional de Saude', [], 'C2').
utente('U8', 'Aldo Castro', '98985876143', 'M', '1980-04-16', 'AldoCastro388@hotmail.com', '962967073', 'Rua Doutor Jose Regio Duse 4700-004 Braga', 'Programador', [], 'C2').
utente('U9', 'Tiburcio Guerreiro', '08519097436', 'M', '1978-03-16', 'TiburcioGuerreiro342@outlook.com', '936810784', 'Rua do Cortinhal 4775-001 Cambeses', 'Militar', ['Obesidade'], 'C2').
utente('U11', 'Florisbela Carvalho', '38557360298', 'F', '1969-02-01', 'FlorisbelaCarvalho1969@gmail.com', '910843987', 'Rua Dona Teresa Aldao 4800-004 Guimaraes', 'Tecnico em edificacoes', [], 'C4').
utente('U14', 'Tatiano Carvalho', '39242786162', 'M', '1960-12-30', 'TatianoCarvalho@outlook.com', '921622230', 'Praceta do Viajante Azurem 4800-004 Guimaraes', 'Biologo', [], 'C2').
utente('U15', 'Indro Leal', '45346361911', 'M', '1949-07-08', 'IndroLeal1949@gmail.com', '929206797', 'Rua de Sao Lourenco da Ordea 4700-004 Braga', 'Webmaster', [], 'C1').
utente('U17', 'Dinarda Matias', '29224473940', 'F', '1974-10-09', 'DinardaMatias1974@outlook.com', '915914158', 'Largo do Rego Duse 4700-008 Braga', 'DevOps', [], 'C4').
utente('U18', 'Marcilene Gonçalves', '77919813012', 'M', '1989-12-04', 'MarcileneGoncalves305@hotmail.com', '924527670', 'Rua do Carvalho Duse 4700-005 Braga', 'Perito criminal', ['Doenca coronaria'], 'C1').
utente('U19', 'Milena Vicente', '39280061650', 'F', '2004-06-04', 'MilenaVicente@gmail.com', '969590100', 'Travessa General Humberto Delgado 4760-001 Vila Nova de Fanelicaao', 'Jovem Aprendiz', [], 'C2').
utente('U13', 'Adelia Aaaraal', '41527730111', 'F', '1944-09-26', 'email_desconhecido', '910419566', 'Rua Doutor Joao Afonso Almeida Azurem 4800-004 Guimaraes', 'Bioquimico', [], 'C2').
utente('U4', 'Cinderela Nogueira', '60752287000', 'F', '1950-12-08', 'CinderelaNogueira@outlook.com', 'tlf_desconhecido', 'Rua Paulo Vi 4700-004 Braga', 'Auxiliar de limpeza', [], 'C2').
utente('U10', 'Jansenio Figueiredo', '02069412938', 'M', '1983-02-02', 'JansenioFigueiredo218@outlook.com', '914112112', 'morada_desconhecida', 'Comissario de bordo', [], 'C2').
utente('U20', 'Delairio Lopes', '36606379205', 'M', '1956-05-23', 'email_impossivel', '927819307', 'Rua Monte de Baixo 4705-001 Arentim', 'Cerimonialista', [], 'C1').
utente('U2004', 'Julio Romao', '123212', 'M', '1919-07-22', 'julio@gmail.com', '92323430883', 'morada_desconhecida', 'Ganer', [], 'C2').

true.
```

Evolução do Conhecimento

- **Evolução do Conhecimento Imperfeito**
 - **Conhecimento Imperfeito Impreciso**

No caso do Conhecimento Imperfeito Impreciso são adicionadas **exceções** para que seja retornado **falso** caso seja uma opção que não as incluídas nas exceções e **desconhecido** caso esteja incluído numa delas.

À semelhança dos anteriores, damos como argumento o tipo de conhecimento (**conheImpImp**).

Este predicado **verifica se os invariantes que envolvem exceções continuam verdadeiros depois de ser inserido conhecimento** e, se continuam, então esse conhecimento mantém-se.

```
evolucaoC(T, conheImpImp) :- solucoes(I, +(excecao(T))::I, Lint),  
                             insercao(excecao(T)),  
                             testaPredicados(Lint).
```


Evolução do Conhecimento

- **Evolução do Conhecimento Imperfeito**

- **Conhecimento Imperfeito Impreciso**

Exemplo de evolução de Conhecimento Imperfeito Impreciso:

```
?- listing(excecao(_)).
excecao(utente('U12', 'Biana Andrade', '03294496759', 'F', '1982-05-02', 'BianaAndrade@hotmail.com', 932477556, 'Rua de Sao Martinho Dume 4700-008 Braga', 'Padeiro', [], 'C1')).
excecao(utente('U12', 'Biana Andrade', '03294496759', 'F', '1982-05-02', 'BianaAndrade@hotmail.com', 932477556, 'Rua do Monte 4700-009 Braga', 'Padeiro', [], 'C1')).
excecao(utente('U6', 'Zelia Abreu', '94334540382', 'F', '1995-06-19', 'ZeliaAbreu@hotmail.com', 966727220, 'Rua do Brasil 4775-001 Cambeses', 'Eletricista', [], 'C2')).
excecao(utente('U6', 'Zelia Abreu', '94334540382', 'F', '1995-06-19', 'ZeliaAbreu@hotmail.com', 916779510, 'Rua do Brasil 4775-001 Cambeses', 'Eletricista', [], 'C2')).
excecao(utente('U2', 'Bianca Moura', '84874182284', 'F', '1958-10-30', 'BiancaMoura244@gmail.com', 921591637, 'Rua de Sao Rosendo ( Bispo de Dume ) Dume 4700-008 Braga', 'Vigilante', [], 'C1')).
excecao(utente('U2', 'Bianca Moura', '84874182284', 'F', '1958-10-30', 'BiancaMoura299@gmail.com', 921591637, 'Rua de Sao Rosendo ( Bispo de Dume ) Dume 4700-008 Braga', 'Vigilante', [], 'C1')).
excecao(utente('U3', 'Vilar Nunes', '05052988794', 'M', '1976-02-17', 'VilarNunes1976@gmail.com', 968452327, 'Rua de Remelhe Dume 4700-008 Braga', 'Vigilante', [], 'C1')).
excecao(utente('U3', 'Vilar Nunes', '05052988794', 'F', '1976-02-17', 'VilarNunes1976@gmail.com', 968452327, 'Rua de Remelhe Dume 4700-008 Braga', 'Vigilante', [], 'C1')).
excecao(staff('S6', 'Almiro Soares', '65199252818', 'AlmiroSoares1991@gmail.com', 'C2')).
excecao(staff('S6', 'Almiro Soares', '65199252818', 'AlmiroSoares@gmail.com', 'C2')).

?- evolucaoC(utente('U2004', 'Julio Romao', '123212', 'M', '1919-07-22', 'julio@gmail.com', '92323430883', 'Terra', 'Gamer', [], C2), conheImpImp).
true

?- evolucaoC(utente('U2004', 'Julio Romao', '123212', 'M', '1919-07-22', 'juliogamesPT@gmail.com', '92323430883', 'Terra', 'Gamer', [], C2), conheImpImp).
true

?- listing(excecao(_)).
excecao(utente('U12', 'Biana Andrade', '03294496759', 'F', '1982-05-02', 'BianaAndrade@hotmail.com', 932477556, 'Rua de Sao Martinho Dume 4700-008 Braga', 'Padeiro', [], 'C1')).
excecao(utente('U12', 'Biana Andrade', '03294496759', 'F', '1982-05-02', 'BianaAndrade@hotmail.com', 932477556, 'Rua do Monte 4700-009 Braga', 'Padeiro', [], 'C1')).
excecao(utente('U6', 'Zelia Abreu', '94334540382', 'F', '1995-06-19', 'ZeliaAbreu@hotmail.com', 966727220, 'Rua do Brasil 4775-001 Cambeses', 'Eletricista', [], 'C2')).
excecao(utente('U6', 'Zelia Abreu', '94334540382', 'F', '1995-06-19', 'ZeliaAbreu@hotmail.com', 916779510, 'Rua do Brasil 4775-001 Cambeses', 'Eletricista', [], 'C2')).
excecao(utente('U2', 'Bianca Moura', '84874182284', 'F', '1958-10-30', 'BiancaMoura244@gmail.com', 921591637, 'Rua de Sao Rosendo ( Bispo de Dume ) Dume 4700-008 Braga', 'Vigilante', [], 'C1')).
excecao(utente('U2', 'Bianca Moura', '84874182284', 'F', '1958-10-30', 'BiancaMoura299@gmail.com', 921591637, 'Rua de Sao Rosendo ( Bispo de Dume ) Dume 4700-008 Braga', 'Vigilante', [], 'C1')).
excecao(utente('U3', 'Vilar Nunes', '05052988794', 'M', '1976-02-17', 'VilarNunes1976@gmail.com', 968452327, 'Rua de Remelhe Dume 4700-008 Braga', 'Vigilante', [], 'C1')).
excecao(utente('U3', 'Vilar Nunes', '05052988794', 'F', '1976-02-17', 'VilarNunes1976@gmail.com', 968452327, 'Rua de Remelhe Dume 4700-008 Braga', 'Vigilante', [], 'C1')).
excecao(staff('S6', 'Almiro Soares', '65199252818', 'AlmiroSoares1991@gmail.com', 'C2')).
excecao(staff('S6', 'Almiro Soares', '65199252818', 'AlmiroSoares@gmail.com', 'C2')).
excecao(utente('U2004', 'Julio Romao', '123212', 'M', '1919-07-22', 'julio@gmail.com', '92323430883', 'Terra', 'Gamer', [], _)).
excecao(utente('U2004', 'Julio Romao', '123212', 'M', '1919-07-22', 'juliogamesPT@gmail.com', '92323430883', 'Terra', 'Gamer', [], _)).
```

Evolução do Conhecimento

- **Evolução do Conhecimento Imperfeito**

- **Conhecimento Imperfeito Interdito**

Para este tipo de Conhecimento foi preciso criar mais um invariante, que não permite inserir um e-mail *a posteriori* para aqueles em que consideramos o e-mail interdito (de ser conhecido).

Eis um exemplo para o caso do utente:

(Novo invariante criado)

```
+utente(Id,N,Nu,G,DN,E,T,M,P,DC,IdCentro) :: (solucoes((Id,N,Nu,G,DN,E,T,M,P,DC,IdCentro), (utente(Id,N,Nu,G,DN,E,T,M,P,DC,IdCentro), nulointerdito(E)), R),  
comprimento(R,0)).
```

```
evolucaoC(utente(Id,N,Nu,G,DN,email_impossivel,T,M,P,DC,IdCentro), utente, conheImpInt, email) :-  
    evolucaoC(utente(Id,N,Nu,G,DN,email_impossivel,T,M,P,DC,IdCentro)),  
    insercao((excecao(utente(Id,N,Nu,G,DN,_,T,M,P,DC,IdCentro)) :-  
        | | | utente(Id,N,Nu,G,DN,email_impossivel,T,M,P,DC,IdCentro))),  
    insercao((nulointerdito(email_impossivel))).
```


Involução do Conhecimento

Involução do Conhecimento Perfeito

- Evolução de Conhecimento Perfeito Positivo
- Evolução de Conhecimento Perfeito Negativo

Involução do Conhecimento Imperfeito

- Evolução de Conhecimento Imperfeito Incerto
- Evolução de Conhecimento Imperfeito Impreciso
- Evolução de Conhecimento Imperfeito Interdito

Involução do Conhecimento

- **Involução do Conhecimento Perfeito**

À semelhança da evolução do conhecimento, foi criado um meta-predicado para o Conhecimento Perfeito Positivo, que foi depois especificado para cada tipo de conhecimento.

```
involucaoC(T) :- solucoes(I, -T::I, L),  
                remocao(T),  
                testaPredicados(L).
```

Dado o paralelismo com a evolução do conhecimento e a extensão da apresentação, apresentaremos apenas de seguida alguns exemplos de demonstração de Involução dos vários tipos de conhecimento, sendo que foram criados vários para todos os tipos.

Involução do Conhecimento

- Involução do Conhecimento Perfeito

Exemplo de involução de Conhecimento Perfeito Positivo.

```
?- listing(utente(_____,_____,_____)).
:- dynamic utente/11.

utente('U0', 'Romulo Mota', '22337840788', 'M', '1936-04-22', 'RomuloMota@outlook.com', '962935227', 'Rua Francisco Augusto Alvina 4700-004 Braga',
utente('U1', 'Goncalo Melo', '65105592957', 'M', '1931-09-18', 'GoncaloMelo1991@hotmail.com', '938823950', 'Travessa 1 de Maio 4700-008 Braga', 'Ge
utente('U5', 'Apolo Aamaral', '39446661300', 'M', '1980-05-06', 'ApoloAamaral@hotmail.com', '926158761', 'Avenida da Liberdade 4760-001 Vila Nova de
utente('U7', 'Afonso Torres', '82144830813', 'M', '1992-12-24', 'AfonsoTorres@gmail.com', '911799897', 'Rua Jose Albino Costa e Silva Azurem 48
utente('U8', 'Aldo Castro', '89885876143', 'M', '1980-04-16', 'AldoCastro388@hotmail.com', '962967073', 'Rua Doutor Jose Regio Dume 4700-004 Braga',
utente('U9', 'Tiburcio Guerreiro', '08519097436', 'M', '1978-03-16', 'TiburcioGuerreiro342@outlook.com', '936810784', 'Rua do Cortinhal 4775-001 Ce
utente('U11', 'Florisbela Carvalho', '38557360298', 'F', '1969-02-01', 'FlorisbelaCarvalho1969@gmail.com', '910843987', 'Rua Dona Teresa Aldao 4800
utente('U14', 'Tatiano Carvalho', '39247286162', 'M', '1960-12-30', 'TatianoCarvalho@outlook.com', '921622230', 'Praceta do Viajante Azurem 4800-00
utente('U15', 'Indro Leal', '45346361911', 'M', '1949-07-08', 'IndroLeal1949@gmail.com', '929206797', 'Rua de Sao Lourenco da Ordem 4700-004 Braga',
utente('U17', 'Dinarda Matias', '29224473940', 'F', '1974-10-09', 'DinardaMatias1974@outlook.com', '915914158', 'Largo do Rego Dume 4700-008 Braga',
utente('U18', 'Marcilene Goncalves', '77919813012', 'M', '1989-12-04', 'MarcileneGoncalves305@hotmail.com', '924527670', 'Rua do Carvalhal Dume 470
utente('U19', 'Milena Vicente', '39280061650', 'F', '2004-06-04', 'MilenaVicente@gmail.com', '969590100', 'Travessa General Humberto Delgado 4760-0
utente('U13', 'Adelia Aamaral', '41527730111', 'F', '1944-09-26', 'email_desconhecido', '910419566', 'Rua Doutor Joao Afonso Almeida Azurem 4800-004 G
utente('U4', 'Cinderela Nogueira', '50775228700', 'F', '1950-12-08', 'CinderelaNogueira@outlook.com', 'tlf_desconhecido', 'Rua Paulo Vi 4700-004 Br
utente('U10', 'Jansenio Figueiredo', '02069412938', 'M', '1983-02-02', 'JansenioFigueiredo218@outlook.com', '914112112', 'morada_desconhecida', 'Comi
utente('U20', 'Delairo Lopes', '36606379205', 'M', '1956-05-23', 'email_inpossivel', '927819307', 'Rua Monte de Baixo 4705-001 Arentia', 'Ceriaoniali

true.

?- involucaoC(utente('U20',_____,_____,_____)).
true.

?- listing(utente(_____,_____,_____)).
:- dynamic utente/11.

utente('U0', 'Romulo Mota', '22337840788', 'M', '1936-04-22', 'RomuloMota@outlook.com', '962935227', 'Rua Francisco Augusto Alvina 4700-004 Braga',
utente('U1', 'Goncalo Melo', '65105592957', 'M', '1931-09-18', 'GoncaloMelo1991@hotmail.com', '938823950', 'Travessa 1 de Maio 4700-008 Braga', 'Ge
utente('U5', 'Apolo Aamaral', '39446661300', 'M', '1980-05-06', 'ApoloAamaral@hotmail.com', '926158761', 'Avenida da Liberdade 4760-001 Vila Nova de
utente('U7', 'Afonso Torres', '82144830813', 'M', '1992-12-24', 'AfonsoTorres@gmail.com', '911799897', 'Rua Jose Albino Costa e Silva Azurem 48
utente('U8', 'Aldo Castro', '89885876143', 'M', '1980-04-16', 'AldoCastro388@hotmail.com', '962967073', 'Rua Doutor Jose Regio Dume 4700-004 Braga',
utente('U9', 'Tiburcio Guerreiro', '08519097436', 'M', '1978-03-16', 'TiburcioGuerreiro342@outlook.com', '936810784', 'Rua do Cortinhal 4775-001 Ce
utente('U11', 'Florisbela Carvalho', '38557360298', 'F', '1969-02-01', 'FlorisbelaCarvalho1969@gmail.com', '910843987', 'Rua Dona Teresa Aldao 4800
utente('U14', 'Tatiano Carvalho', '39247286162', 'M', '1960-12-30', 'TatianoCarvalho@outlook.com', '921622230', 'Praceta do Viajante Azurem 4800-00
utente('U15', 'Indro Leal', '45346361911', 'M', '1949-07-08', 'IndroLeal1949@gmail.com', '929206797', 'Rua de Sao Lourenco da Ordem 4700-004 Braga',
utente('U17', 'Dinarda Matias', '29224473940', 'F', '1974-10-09', 'DinardaMatias1974@outlook.com', '915914158', 'Largo do Rego Dume 4700-008 Braga',
utente('U18', 'Marcilene Goncalves', '77919813012', 'M', '1989-12-04', 'MarcileneGoncalves305@hotmail.com', '924527670', 'Rua do Carvalhal Dume 470
utente('U19', 'Milena Vicente', '39280061650', 'F', '2004-06-04', 'MilenaVicente@gmail.com', '969590100', 'Travessa General Humberto Delgado 4760-0
utente('U13', 'Adelia Aamaral', '41527730111', 'F', '1944-09-26', 'email_desconhecido', '910419566', 'Rua Doutor Joao Afonso Almeida Azurem 4800-004 G
utente('U4', 'Cinderela Nogueira', '50775228700', 'F', '1950-12-08', 'CinderelaNogueira@outlook.com', 'tlf_desconhecido', 'Rua Paulo Vi 4700-004 Br
utente('U10', 'Jansenio Figueiredo', '02069412938', 'M', '1983-02-02', 'JansenioFigueiredo218@outlook.com', '914112112', 'morada_desconhecida', 'Comi

true.
```

Involução do Conhecimento

- Involução do Conhecimento Imperfeito
 - Conhecimento Imperfeito Incerto

Exemplo de involução de Conhecimento Imperfeito Incerto:

```
?- listing(excecao(_)).
excecao(staff(Id, IdC, N, _)) :-
    staff(Id, IdC, N, email_desconhecido).
excecao(utente('U12', 'Biana Andrade', '03294496759', 'F', '1982-05-02', 'BianaAndrade@hotmail.com', 932477556, 'Rua de Sao Martinho Dume 4700-008
excecao(utente('U12', 'Biana Andrade', '03294496759', 'F', '1982-05-02', 'BianaAndrade@hotmail.com', 932477556, 'Rua do Monte 4700-009 Braga', 'Pa
excecao(utente('U6', 'Zelia Abreu', '94334540382', 'F', '1995-06-19', 'ZeliaAbreu@hotmail.com', 966727220, 'Rua do Brasil 4775-001 Cambeses', 'Ele
excecao(utente('U6', 'Zelia Abreu', '94334540382', 'F', '1995-06-19', 'ZeliaAbreu@hotmail.com', 916779510, 'Rua do Brasil 4775-001 Cambeses', 'Ele
excecao(utente('U2', 'Bianca Moura', '84874182284', 'F', '1958-10-30', 'BiancaMoura244@gmail.com', 921591637, 'Rua de Sao Rosendo ( Bispo de Dume
excecao(utente('U2', 'Bianca Moura', '84874182284', 'F', '1958-10-30', 'BiancaMoura299@gmail.com', 921591637, 'Rua de Sao Rosendo ( Bispo de Dume
excecao(utente('U3', 'Vilar Nunes', '05052988794', 'M', '1976-02-17', 'VilarNunes1976@gmail.com', 968452327, 'Rua de Remelhe Dume 4700-008 Braga',
excecao(utente('U3', 'Vilar Nunes', '05052988794', 'F', '1976-02-17', 'VilarNunes1976@gmail.com', 968452327, 'Rua de Remelhe Dume 4700-008 Braga',
excecao(staff('S6', 'Almiro Soares', '65199252818', 'AlmiroSoares1991@gmail.com', 'C2')).
excecao(staff('S6', 'Almiro Soares', '65199252818', 'AlmiroSoares@gmail.com', 'C2')).

?- involucaoC(utente('U6', _____), conheImpImp).
true.

?- listing(excecao(_)).
excecao(utente('U12', 'Biana Andrade', '03294496759', 'F', '1982-05-02', 'BianaAndrade@hotmail.com', 932477556, 'Rua de Sao Martinho Dume 4700-008
excecao(utente('U12', 'Biana Andrade', '03294496759', 'F', '1982-05-02', 'BianaAndrade@hotmail.com', 932477556, 'Rua do Monte 4700-009 Braga', 'Pa
excecao(utente('U6', 'Zelia Abreu', '94334540382', 'F', '1995-06-19', 'ZeliaAbreu@hotmail.com', 916779510, 'Rua do Brasil 4775-001 Cambeses', 'Ele
excecao(utente('U2', 'Bianca Moura', '84874182284', 'F', '1958-10-30', 'BiancaMoura244@gmail.com', 921591637, 'Rua de Sao Rosendo ( Bispo de Dume
excecao(utente('U2', 'Bianca Moura', '84874182284', 'F', '1958-10-30', 'BiancaMoura299@gmail.com', 921591637, 'Rua de Sao Rosendo ( Bispo de Dume
excecao(utente('U3', 'Vilar Nunes', '05052988794', 'M', '1976-02-17', 'VilarNunes1976@gmail.com', 968452327, 'Rua de Remelhe Dume 4700-008 Braga',
excecao(utente('U3', 'Vilar Nunes', '05052988794', 'F', '1976-02-17', 'VilarNunes1976@gmail.com', 968452327, 'Rua de Remelhe Dume 4700-008 Braga',
excecao(staff('S6', 'Almiro Soares', '65199252818', 'AlmiroSoares1991@gmail.com', 'C2')).
excecao(staff('S6', 'Almiro Soares', '65199252818', 'AlmiroSoares@gmail.com', 'C2')).
```

Sistema de Inferência

- A necessidade da criação de um Sistema de Inferência deve-se ao facto de o nosso sistema de representação de conhecimento e raciocínio, quando questionado, não responder apenas verdadeiro ou falso. É agora possível obter resposta do tipo **desconhecido**.

```
si(Q, verdadeiro) :- Q.  
si(Q, falso) :- -Q.  
si(Q, desconhecido) :- nao(Q),  
                        nao(-Q).
```

Sistema de Inferência

- Uma possível crítica a este sistema de inferência é que ele é muito simples, pois apenas responde a uma questão de cada vez.
- Assim, decidimos aumentar a complexidade do nosso sistema de inferência por, numa primeira fase, **tornar possível responder a mais do que uma questão de cada vez**, isto é, responder a uma lista de questões, criando uma lista de respostas.

```
siLista([], []).  
siLista([Q|Qs], R) :- si(Q, X),  
                      siLista(Qs, Y),  
                      R = [X|Y].
```

Sistema de Inferência

- Responde a uma **conjunção** de questões

- **verdadeiro** se $\langle \forall i : 1 \dots n : X_i \text{ é verdadeiro} \rangle$
- **falso** se $\langle \exists i : 1 \dots n : X_i \text{ é falso} \rangle$
- **desconhecido** se $\langle \exists i : 1 \dots n : X_i \text{ é desconhecido} \rangle \wedge \langle \forall i : 1 \dots n : X_i \text{ não é falso} \rangle$

p	q	$p \wedge q$
verdadeiro	verdadeiro	verdadeiro
falso	—	falso
—	falso	falso
verdadeiro	desconhecido	desconhecido
desconhecido	verdadeiro	desconhecido
desconhecido	desconhecido	desconhecido

Sistema de Inferência

- Responde a uma **conjunção** de questões

```
siConjuncao([],_).  
siConjuncao([Q],R) :- si(Q, R).  
siConjuncao([Q|Qs], R) :- si(Q, R1),  
                           siConjuncao(Qs,R2),  
                           conjuncao(R1,R2,R).
```

```
conjuncao(verdadeiro, verdadeiro, verdadeiro).  
conjuncao(falso, _, falso).  
conjuncao(_, falso, falso).  
conjuncao(verdadeiro, desconhecido, desconhecido).  
conjuncao(desconhecido, verdadeiro, desconhecido).  
conjuncao(desconhecido, desconhecido, desconhecido).
```


Sistema de Inferência

- Responde a uma **disjunção** de questões

- **verdadeiro** se $\langle \exists i : 1 \dots n : X_i \text{ é verdadeiro} \rangle$
- **falso** se $\langle \forall i : 1 \dots n : X_i \text{ é falso} \rangle$
- **desconhecido** se $\langle \exists i : 1 \dots n : X_i \text{ é desconhecido} \rangle \wedge \langle \forall i : 1 \dots n : X_i \text{ não é verdadeiro} \rangle$

p	q	$p \vee q$
verdadeiro	—	verdadeiro
—	verdadeiro	verdadeiro
falso	falso	falso
falso	desconhecido	desconhecido
desconhecido	falso	desconhecido
desconhecido	desconhecido	desconhecido

Sistema de Inferência

- Responde a uma **disjunção** de questões

```
siDisjuncao([],_).  
siDisjuncao([Q],R) :- si(Q, R).  
siDisjuncao([Q|Qs], R) :- si(Q, R1),  
                           siDisjuncao(Qs, R2),  
                           disjuncao(R1, R2, R).
```

```
disjuncao(verdadeiro, _, verdadeiro).  
disjuncao(_, verdadeiro, verdadeiro).  
disjuncao(falso, falso, falso).  
disjuncao(falso, desconhecido, desconhecido).  
disjuncao(desconhecido, falso, desconhecido).  
disjuncao(desconhecido, desconhecido, desconhecido).
```

