**Determinar todas as instituicoes/prestadores a que um utente já recorreu**

Para determinar todos os prestadores a que um utente recorreu decidimos criar um predicado, que manipula o conhecimento de um identificador do utente e a lista com os prestadores recorridos. Neste predicado auxiliamo-nos no predicado **solucoes**, sendo que queríamos uma lista com termos: identificador do prestador, nome do prestador. A **questão** que colocávamos no **solucoes** teria de ser relativa aos cuidados do utente, sendo que também teríamos de identificar o nome do prestador, daí necessitarmos do predicado **prestador** na **questão**. Após obter esta lista, auxiliamo-nos ainda no predicado **multiConjunto** para obter o resultado, por forma a eliminar os repetidos e obter uma lista com pares de elementos e número de ocorrências desses elementos.

De seguida apresentamos o predicado **prestadoresRecorridosUtente**:

prestadoresRecorridosUtente: IdUtente, ListaPrestador -> {V,F}

prestadoresRecorridosUtente( IdU,R ) :- solucoes( ( P,NP ),( cuidado( \_,IdU,P,\_,\_,\_ ),prestador( P,NP,\_,\_ ) ),L ), multiConjunto( L,R ).

Para as instituições o raciocínio é semelhante e por isso apenas apresentamos em anexo.

**Determinar o total de custos dos cuidados de saúde por utente/especialidade/prestadores/datas**

Para determinar o total de custos de cuidados de saúde por especialidade decidimos criar um predicado, que manipula o conhecimento de uma especialidade e o total de custos. Neste predicado auxiliamo-nos no predicado **solucoes**, sendo que queríamos uma lista com termos: custo do cuidado. A **questão** que colocávamos no **solucoes** teria de ser relativa aos cuidados de saúde, sendo que também teríamos de identificar a especialidade do cuidado, daí necessitarmos do predicado **prestador** na **questão**. Após obter esta lista, auxiliamo-nos ainda no predicado **somaC** para obter o resultado, por forma a soma total dos custos.

De seguida apresentamos o predicado **totalCustoEspecialidade**:

Extensao do predicado totalCustoEspecialidade: Especialidade, Custo -> {V,F}

totalCustoEspecialidade( Esp,C ) :- solucoes( Custo,( cuidado(\_,\_,P,\_,Custo,\_),prestador(P,\_,Esp,\_) ),L ),

somaC( L,C ).