

Um divertido exercício de representação de conhecimento¹

Descrição

Nos anos 1930s, um cantor estadunidense chamado Dwight Latham, leu um livro chamado “Wit and Humor of the Age”, de 1883, do qual o famoso escritor Mark Twain é co-autor. Nele, Latham encontrou um parágrafo em que Twain demonstra que é possível para uma pessoa se tornar o seu próprio avô. Isso é possível através de uma combinação improvável, porém legal, de casamentos, na qual uma pessoa se torna padrasto de sua madrasta, ou seja, desconsiderando os “drastos”, ele se torna seu próprio avô. Em 1947, juntamente com Moe Jaffe, Latham transformou essa improvável estória em música². Você pode ver uma performance dessa música [neste link](#), ou procurar por outros exemplos no Youtube.

A letra da música segue abaixo, com uma tradução livre à direita.

<p>I'm my Own Grandpa By Dwight Latham and Moe Jaffe Performed by Ray Stevens</p> <p>Many, many years ago When I was twenty-three I was married to a widow Who was pretty as can be This widow had a grown-up daughter Who had hair of red My father fell in love with her And soon the two were wed</p> <p>This made my dad my son-in-law And it really changed my life For my daughter was my mother 'Cause she was my father's wife To complicate the matter Even though it brought me joy I soon became the father Of a bouncing baby boy</p> <p>My little baby then became A brother-in-law to Dad And so became my uncle Though it made me very sad For if he was my uncle Then that also made him brother Of the widow's grown-up daughter</p>	<p>Eu sou meu próprio avô Por Dwight Latham e Moe Jaffe Interpretado por Ray Stevens</p> <p>Muitos, muitos anos atrás Quando eu tinha vinte e três anos Eu era casado com uma viúva Que era tão bonita quanto possível Esta viúva tinha uma filha crescida Que tinha cabelos ruivos Meu pai se apaixonou por ela E logo os dois se casaram</p> <p>Isso fez de meu pai meu filho³ E de fato mudou minha vida Pois minha filha era minha mãe Porque ela era a esposa de meu pai. Para complicar o assunto Embora isso realmente me tenha trazido alegria Eu logo me tornei o pai De um menino saltitante</p> <p>Este pequeno bebê então se tornou Um irmão do papai E assim se tornou meu tio Embora isso me tenha entristecido muito Pois se ele era meu tio Então isso também o fez irmão Da filha adulta da viúva</p>
---	---

1 Inspirado em um exercício da disciplina [CS 173](#), da Universidade de Rochester.

2 Fonte: [Wikipedia](#)

3 “son in law”. O termo mais adequado em português seria genro. No entanto vamos tratar as relações “in law” da mesma forma que as biológicas, por isso preferimos traduzir como filho. No seu programa você também deve fazer com que o predicado “son” também reconheça os “in-law”, assim como os demais predicados. Ou seja, também tratar padrasto e madrasta como pai e mãe, respectivamente, bem como “filho-drasto”, “genro” como filho, e “filha-drasta”, “nora” como filha e assim por diante (exemplo: o marido da irmã da minha mãe também é meu tio etc.)

<p>Who is of course my step-mother</p> <p>My father's wife then had a son Who kept them on the run And he became my grandchild For he was my daughter's son My wife is now my mother's mother And it makes me blue Because although she is my wife She's my grandmother too</p> <p>Now if my wife is my grandmother Then I'm her grandchild And every time I think of it It nearly drives me wild For now I have become The strangest case you ever saw As husband of my grandmother I am my own grandpa</p> <p>Chorus I'm my own grandpa I'm my own grandpa It sounds funny I know But it really is so Oh, I'm my own grandpa</p>	<p>Que, é claro, é minha madrasta</p> <p>A esposa de meu pai então teve um filho Que os manteve na correria E ele se tornou meu neto Pois ele era filho de minha filha Minha esposa é agora a mãe de minha mãe. E isso me deixa triste Porque embora ela seja minha esposa Ela é minha avó também</p> <p>Agora, se minha esposa é minha avó Então eu sou seu neto E toda vez que penso nisso Isso quase me deixa louco Porque agora eu me tornei O caso mais estranho que você já viu Como marido de minha avó Eu sou meu próprio avô</p> <p>Refrão Eu sou meu próprio avô Eu sou meu próprio avô Parece engraçado, eu sei. Mas realmente é assim Oh, eu sou meu próprio avô</p>
--	--

Como você pôde perceber, a música conta uma história que utiliza relações de família, do tipo, “eu era casado com uma viúva”, “essa mulher tinha uma filha” etc. Ou seja, **fatos** relativos a família do narrador da música. A maioria desses fatos estão em primeira pessoa. A nossa versão Prolog para esses fatos está listada abaixo. A música também faz algumas afirmações (por exemplo, “minha esposa é minha avó”), que decorrem dos fatos. Essas afirmações decorrem dos fatos graças a **regras** que definem relações familiares. Também há informações irrelevantes na música, como a cor de cabelo da filha da viúva etc.

Sua primeira missão é escrever essas regras, de tal maneira que, juntamente com os fatos, possam ser verificadas as afirmações do compositor. Ou seja, queremos que ao final, o seu programa responda ‘true’ à consulta:

```
son(dad, i), daughter(redhair, i), mother(redhair, i),
brother(baby, dad), uncle(baby, i), brother(baby,
redhair), mother(redhair, i), grandchild(onrun, i), mother(widow,
redhair), grandmother(widow, i), grandchild(i,
widow), grandfather(i, i).
```

Note que nenhum dos termos da consulta será um fato do código do programa, eles devem ser deduzidos a partir da aplicação das regras que você irá escrever (de uso geral, e não específicas para a situação familiar em questão). A versão do professor adiciona 22 linhas aos fatos abaixo. Não necessariamente esse é o número ótimo, cito apenas para vocês terem uma ideia do pequeno volume do trabalho.

Como dito na nota de rodapé e anteriormente, seu programa deve tratar os “in-laws” da mesma forma que as relações biológicas. Então meu genro é meu filho, assim como minha madrasta é minha mãe etc.

Seguem-se os fatos iniciais:

```
child(redhair, widow).
child(i, dad).
child(onrun, dad).
child(baby, i).
```

```
male(i).
male(dad).
male(onrun).
male(baby).
```

```
female(redhair).
female(widow).
```

```
spouse(i, widow).
spouse(widow, i).
spouse(dad, redhair).
spouse(redhair, dad).
```

Lembre-se, você não deve acrescentar mais nenhum fato, esses são suficientes para o nosso caso. Você escreverá apenas regras, por exemplo, `brother(X, Y)` será uma regra que você escreverá para dizer quando um X é irmão de um Y. Você deve fazer com que uma pessoa não seja irmão ou irmã dela própria, como vimos no exemplo na aula (apenas para essas relações).

Uma vez que seu programa esteja respondendo às consultas corretamente, ele é uma nova fonte de conhecimento para descobrir novas relações que nem foram citadas na música! **Faça no seu programa também um predicado `uncles(Resp)` que retorne uma lista `Resp` dos pares (X, Y), sem repetições, tais que `uncle(X, Y)`, ou seja, X é tio de Y⁴. Explique no relatório o caminho que Prolog utiliza para encontrar os pares, o modo debug (trace) pode lhe ser útil.**

Para finalizar, **escreva todos os fatos e regras do seu programa na sintaxe da lógica de predicados**. Não esqueça dos quantificadores!

Entrega

Seu grupo, de no máximo 3 pessoas, deve entregar um único arquivo .zip contendo:

- Arquivo .pl do projeto, para ser executado pelo SWI Prolog. Comente o código explicando-o, e também coloque como comentários os nomes e logins de todos os integrantes do grupo.
- Arquivo .pdf com o relatório do projeto.
 - O relatório deve conter: Descrição do Problema, Destaques da Implementação com Exemplos de Execução (o de `uncles(Resp)` bem como outros que achar interessante destacar), Versão em Lógica de Predicados, Conclusão, Bibliografia utilizada.
- Pontuação: Programa (6,0); Relatório (4,0).

4 Dica: pesquise sobre o predicado built-in `setof`