

PROLOG Um programa PROLOG é composto basicamente de 3 coisas: fatos, regras e perguntas. Muitas vezes a pergunta é externa ao programa. Os fatos e regras são construídos usando-se cláusulas de Horn. Uma cláusula de Horn é uma disjunção de literais, com no máximo um literal positivo. Em lógica tal cláusula aparece como

$$\sim a \vee \sim b \vee \sim c \dots \vee e$$

Lembrando da equivalência em lógica ($\sim A \vee B \equiv A \rightarrow B$), tal cláusula corresponde a $a \wedge b \wedge c \dots \rightarrow e$, o que em linguagem comum corresponde a uma definição, como por exemplo: Se é um gato preto \wedge é charmoso \wedge é inteligente ENTÃO é o binho.

A cláusula de Horn acima ($\sim a \vee \sim b \vee \sim c \dots \vee e$) escreve-se em prolog ao contrário, assim:

$$e :- a, b, c, \dots$$

Note que na definição PROLOG a, b, c, \dots estão na sua versão positiva

Neste caso, a cláusula de Horn sem literais negativos é dita FATO. Com exatamente um literal positivo é dita REGRA. E, sem literal positivo é chamada de objetivo.

Uma fórmula de Horn é uma conjunção de cláusulas de Horn. A relevância das cláusulas de Horn através do princípio de resolução reside no fato de que a resolução de duas cláusulas de Horn é uma cláusula de Horn. O nome "Cláusula de Horn" é uma homenagem ao lógico Alfred Horn, que foi quem primeiro chamou a atenção para o valor destas cláusulas, em 1951, no artigo "On sentences which are true of direct unions of algebras".

A obrigação de usar cláusulas de Horn pode deixar mais pobre a representação lógica do problema, mas em compensação garante uma busca automatizada eficiente pelo Prolog.

Princípio da resolução

É a regra de inferência que produz a partir de duas cláusulas uma nova cláusula implicada por elas. A regra toma duas cláusulas contendo literais complementares e produz uma nova cláusula com todos os literais de ambas, excluídos os complementares. A cláusula produzida é chamada *resolvente* das duas iniciais. Quando usada em conjunto com um algoritmo de busca a regra de resolução ganha poder para decidir a satisfatibilidade de uma fórmula proposicional e a validade de uma sentença sob um conjunto de axiomas. A técnica usa a prova por contradição, e tem os seguintes passos:

1. deve-se incluir a sentença a ser provada NEGADA
2. Todas as premissas têm que estar conectadas por conjunções.
3. A regra de resolução é aplicada a todos os possíveis pares de cláusulas que contém literais complementares
4. A cláusula resultante é simplificada removendo-se literais repetidos e os literais complementares. A seguir ela é adicionada ao conjunto original de cláusulas para posteriores inferências.
5. Se uma cláusula vazia é encontrada, então a fórmula inteira é contraditória (ou não é satisfeita) e pode-se concluir que a conjectura inicial provém das premissas originais.
6. Se, ao contrário, uma cláusula vazia não pode ser encontrada, e a regra de resolução não pode ser aplicada para derivar mais cláusulas, então a conjectura original não é um teorema da base de conhecimento original.

Um exemplo simplório para mostrar o processo de resolução: Seja a seguinte base de conhecimento:

Todos os humanos são mortais
 Zequinha é humano

Pode-se afirmar que Zequinha é mortal? Veja-se a resolução:

- Fato: humano(zequinha)
- Regra: humano(X) \rightarrow mortal(X)
- Negação da afirmação: \sim mortal(Zequinha).
- Eliminando \rightarrow , fica \sim humano(X) \vee mortal(X)
- Neste ponto a base de conhecimento é:
 1. humano(zequinha)
 2. \sim humano(X) \vee mortal(X)
 3. \sim mortal(zequinha)
- Note que nas regras 2 e 3 é possível obter dois literais complementares, desde que se substitua X por zequinha (X/zequinha) na regra 2, e tem-se a regra 4:

$$\sim$$
 humano(X) \vee mortal(X) \oplus \sim mortal(zequinha) , que simplificados ficam:
 4. \sim humano(X)
- Agora juntando 1 e 4, desde que se substitua X por zequinha fica:
 5. humano(zequinha) \oplus \sim humano(zequinha), que evidentemente simplificados dão como resposta NIL (vazio).
 6. Com isso, está demonstrado o teorema proposto.

Para você fazer

A família Pirlimpimpim Suponha uma família cujos iniciadores foram Joao-Maria, que se casaram lá atrás no tempo. Eles tiveram 4 filhos: Carlos, Julia, Marta e Quico. Carlos casou-se com Suzana; Julia com Zé, Marta com Paulo e Quico ficou solteiro. Suzana tinha 2 irmãos: Alfredo e Carolina. O casal Carlos-Suzana teve os filhos Andre e Zélia. O casal Ze-Julia não teve filhos e o casal Paulo-Marta teve 3 filhos: Binho, Ulisses e Marcela. André se casou com Lia e Zélia se casou com o Chico. Binho casou com Lea e deste casamento vieram 3 filhos: Debora, Flora e Marcos. Afonso casou-se com debora e Marcos casou-se com Juliana. Flora não casou. Finalmente o casal Afonso-Debora teve 3 filhos: cecilio, almir e quica

1. Construa um desenho contendo a árvore genealógica parcial desta família (só as pessoas aqui citadas). Por questões de padronização, sempre escreva o casal no formato homem-mulher. Sublinhe na árvore as pessoas que estão em linha ascendente ou descendente, para identificar quem são os agregados. Antes de prosseguir, confira seu desenho com o do professor, para garantir resultados idênticos lá na frente.

2. Agora construa um módulo de nome FAMILI.PL, contendo os fatos que descrevem o desenho. Use os predicados:

```
pai(joao,carlos).
pai(joao,julia).
pai(joao,marta).
pai(joao,quico).
pai(carlos,andre).
pai(carlos,zelia).
pai(paulo,binho).
pai(paulo,ulisses).
pai(paulo,marcela).
pai(afonso,cecilio).
pai(afonso,almir).
pai(afonso,quica).
pai(binho,debora).
pai(binho,flora).
pai(binho,marcos).
```

depois defina mae, casado e irmao (irmao só para os que não se conhecem os pais, que são suzana, alfredo e carolina). Na definição de casado, para um casamento (por exemplo joao e maria) defina dois fatos, a saber: casado(joao,maria) e casado(maria,joao). Note a ausência de acentos. Há um total de mais de 50 fatos aqui.

3. A seguir, construa as regras, a saber:

irmao(X,Y):-pai(Z,X),pai(Z,Y),X\=Y.

tio(X,Y):-

tio(X,Y):-

cunhado(X,Y):-

cunhado(X,Y):-

oavo(X,Y):-

oavo(X,Y):-

aavo(X,Y):-

aavo(X,Y):-

sobrinho(X,Y):-

tioavo(X,Y):-

tioavo(X,Y):-

neto(X,Y):-

neto(X,Y):-

Obviamente se quiser criar novas regras, fique à vontade: genro, nora, primo, ascendente, parente, contraparente...

4. Finalmente, informe algumas questões perdidas: Medite sobre o resultado encontrado. Não esqueça de responder às perguntas com ; até achar um resultado lógico.

1. neto(X,paulo).

2. oavo(X,lea).

3. oavo(X,binho).

4. aavo(X,ulisses).

5. tio(X,andre).

6. cunhado(X,carlos).

7. sobrinho(X,julia).

8. irmao (X,almir).

9. neto(X,marta).

10. tioavo(X,quica).

