

Objetivos: 1. Desenvolver a habilidade para representar e manipular objetos de dados estruturados.

2. Ilustrar Prolog como uma linguagem de consulta de banco de dados

Descrição: Um banco de dados pode ser representado naturalmente em Prolog como um conjunto de fatos. Por exemplo, um banco de dados sobre famílias pode ser representado de tal forma que cada família é descrita unicamente por uma cláusula. Cada família tem três componentes: marido, esposa e filhos. Como o número de filhos varia de família para família, os filhos podem ser representados como uma lista que é capaz de acomodar qualquer quantidade de pessoas. Cada pessoa, por sua vez, será representada por uma estrutura de quatro componentes: nome, sobrenome, data de nascimento e emprego. A informação de emprego corresponderá a 'desempregado' para representar o fato de que o indivíduo não está empregado. Por outro lado, esta informação deverá conter o nome da empresa e o salário recebido. Como exemplo, a cláusula a seguir descreve a família Silva

---

#### % BASE DE CONHECIMENTO

```
familia(
    pessoa(pedro,silva,data(1,abril,1930),empregado(cnn,1000)),
    pessoa(maria,silva,data(1,maio,1945),empregado(bnpp, 1230)),
    [
        pessoa(manoel,silva,data(10,setembro,1975),desempregado),
        pessoa(ana,silva,data(12,novembro,1981),empregado(bcc,800))
    ]
).

familia(
    pessoa(paulo,santos,data(1,janeiro,1955),empregado(abb,1200)),
    pessoa(ana,santos,data(1,maio,1955),desempregado),
    [
        pessoa(manoela,santos,data(10,setembro,1981),desempregado),
        pessoa(andre,santos,data(18,novembro,1978),empregado(bb,1100)),
        pessoa(carlos,santos,data(02,fevereiro,1981),empregado(cc,3200)),
    ]
).

familia(
    pessoa(eddard, stark, data(31, 03, 1948), morto),
    pessoa(catelyn, stark, data(11, 08, 1953), morto),
    [
        pessoa(robb, stark, data(09, 12, 1977), morto),
        pessoa(sansa, stark, data(01, 06, 1979), empregado(conselheira,
11000)),
        pessoa(arya, stark, data(06, 01, 1991), empregado(assasina, 8800)),
        pessoa(bran, stark, data(23, 10, 1993), empregado(vidente, 1)),
        pessoa(rickon, stark, data(29, 04, 1999), morto)
    ]
).
```

Nosso banco de dados então seria constituído por vários fatos semelhantes a este descrevendo todas as famílias que são de interesse para nosso programa. Prolog é uma linguagem muito adequada para recuperar a informação desejada de tal banco de dados. Algo interessante sobre Prolog é que podemos nos referir a objetos sem que seja necessário especificar todos os componentes destes objetos. Nós podemos indicar a estrutura de objetos que estamos interessados e deixar os demais componentes não especificados ou parcialmente especificados. A família Silva pode ser referenciada por:

```
familia(pessoa(_,silva,_,_),_,_).
```

Os caracteres especificados por \_ indicam variáveis anônimas cujo valor não estamos interessados. Por exemplo, podemos nos referir a família composta por três filhos pelo termo

familia(\_,\_,[\_,\_,\_])

Mais um exemplo: para encontrar todas as mulheres casadas que tem ao menos três filhos podemos formular a seguinte consulta:

?- familia(\_,pessoa(N,S,\_,\_),[\_,\_,\_]).

Observe por meio destes exemplos que podemos especificar objetos de interesse através de sua estrutura ao invés de seu conteúdo. Nós especificamos sua estrutura e deixamos seus argumentos como itens não especificados. Um conjunto de predicados pode ser construído visando facilitar o processo de interação com o banco de dados. Alguns destes úteis para o banco de dados relativo a famílias são:

marido(X) :- familia(X,\_,\_).

esposa(X) :- familia(\_,X,\_).

filhos(X) :- familia(\_,\_ ,Filhos),member(X,Filhos).

existe(Pessoa) :- marido(Pessoa); esposa(Pessoa); filhos(Pessoa).

dataDeNasc(pessoa(\_,\_ ,Data,\_),Data).

dt(Data):- existe(X),dataDeNasc(X,Data).

salario(pessoa(\_,\_ ,empregado(\_ ,S)),S).

salario(pessoa(\_,\_ ,desempregado),0).

Baseado nestes fatos e regras, defina regras que ao serem consultadas retorne:

1. Obter os nomes de todas as pessoas no banco de dados;

Entrada: ?- nomes(X).

Saída: X = pedro silva;

X= manojel silva;

2. Encontrar todos os filhos nascidos em 1981;

Entrada: ?- filhosBydt(X,1981).

Saída: X = carlos santos

3. Encontrar todas as esposas empregadas;  
Entrada: ?- esposaEmpregada(X).  
Saída: X = Maria Silva
4. Encontrar todas as pessoas desempregadas nascidas antes de 1962;  
Entrada: ?- pessoasDesempregada(X,1962).  
Saída: X= manoela santos
5. Encontrar as pessoas nascidas antes de 1950 com salário inferior a 5000;  
Entrada: ?- nascidoAntesSalario(X,1950,5000).  
Saída: X = <nome completo>
6. Encontrar os nomes das famílias (sobrenome) que tenham no mínimo três filhos;  
Entrada: ?- familiaMinimFilhos(X,3).  
Saída: X = <sobrenome da família>
7. Encontrar a renda familiar de uma certa família;  
Entrada: ?- rendaFamilia(X,R).  
Saída: X = <sobrenome da família>  
R = <Renda da família>
8. Encontrar todas as famílias que não possuam filhos;  
Entrada: ?- familiaSemFilhos(X).  
Saída: X = <nome da família>
9. Encontrar todos os filhos desempregados;  
Entrada : ?- FilhosDesempregados(X).  
Saída: X = <nome completo do filho>.
10. Encontrar todos os filhos cujos pais diferem em idade por no mínimo 15 anos.  
Entrada : paisIdade(X,15).  
Saída: X = <nome completo do filho>