

# Lógica para Computação

## Programação em Prolog

### Domínio Uzumaki-Hyuga

Antônio Andson da Silva<sup>1</sup>

<sup>1</sup>Engenharia de Computação – Universidade Federal do Ceará (UFC)

Endereço: Av. José de Freitas Queiroz, 5003 – Cedro – Quixadá – Ceará - Brasil 63902-580

andsonsilva@alu.ufc.br

***Resumo.** No decorrer deste relatório vamos abordar sobre o Prolog (Programação Lógica) e utilizando o mesmo, iremos realizar uma atividade relacionada com uma árvore genealógica. O domínio dessa árvore é composto por membros da família Uzumaki-Hyuga. O aspecto desta atividade à ser destacado será as consultas a serem realizadas.*

## 1. Sobre o Prolog

Prolog (Programação Lógica) é uma linguagem de programação que se enquadra no paradigma de Programação em Lógica Matemática. O Prolog foi criado em 1972, na Universidade de Marseille, França. Desde então tem sido utilizada para aplicações de computação simbólica e voltada para as áreas de inteligência artificial e linguística computacional. Consiste numa linguagem puramente lógica, que pode ser chamada de Prolog puro, seu nome Prolog foi escolhido por Philippe Roussel com uma abreviação de "PROgrammation en LOGique".

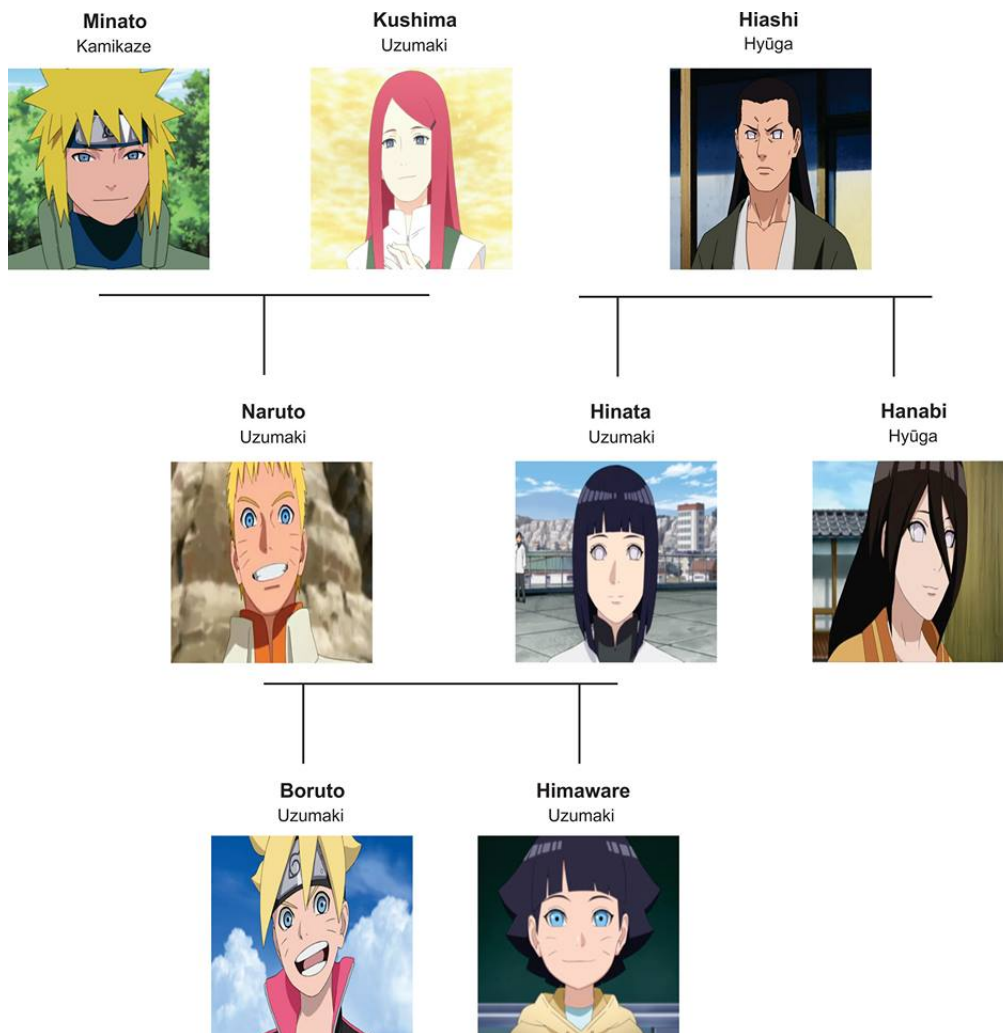
Sua história é bem peculiar, visto que o Prolog nasceu de um projeto que não tinha por foco a implementação de uma linguagem de programação, mas o processamento de linguagens naturais. Processamento de linguagem natural (PLN) é uma subárea da ciência da computação, inteligência artificial e da linguística que estuda os problemas da geração e compreensão automática de línguas humanas naturais. O Prolog é uma linguagem declarativa, significando que em vez de o programa estipular a maneira de chegar à solução, passo a passo, limita-se a fornecer uma descrição do problema que se pretende computar. Usa uma coleção base de dados de fatos e de relações lógicas (regras) que exprimem o domínio relacional do problema a resolver. Um programa pode rodar num modo interativo, a partir de consultas formuladas pelo usuário, usando a base de dados (fatos) e as regras relacionais, e o mecanismo de unificação para produzir a solução. O Prolog é baseado num subconjunto do cálculo de predicados de primeira ordem, o que é definido por cláusulas de Horn. A execução de um programa em Prolog é efetivamente a prova de um teorema por resolução de primeira ordem. Alguns conceitos fundamentais são unificação, recursão, e *backtracking*. *Backtracking* é um tipo de algoritmo que representa um refinamento da busca por força bruta, em que múltiplas soluções podem ser eliminadas sem serem explicitamente examinadas.

## 2. Desenvolvimento

Nessa sessão falaremos sobre o desenvolvimento de uma atividade/programa que utiliza-se a linguagem Prolog. A atividade consiste em três elementos básicos: fatos, regras e

consultas. Basicamente, vamos fazer consultas de membros de uma árvore genealógica, onde cada membro pode fazer parte de uma ou mais categorias. Por exemplo, o membro X pode estar na categoria pai como pode estar na categoria filho. No caso X é pai de um membro Y e filho de outro membro Z.

O domínio familiar que vamos abordar é o Uzumaki-Hyuga. A árvore genealógica desse domínio está ilustrado abaixo.



**Figura 1. Árvore Genealógica Uzumaki-Hyuga**

O programa contém o seguinte conjunto de fatos sobre este domínio familiar:

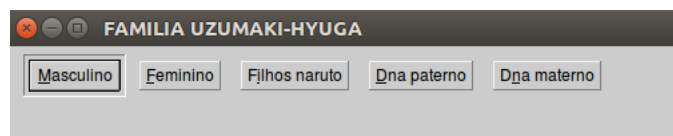
- feminino;
- masculino;
- paiOuMãe;
- conjugue.

E também contém o seguinte conjunto de regras:

- pai, mae;
- filhos, filho, filha;

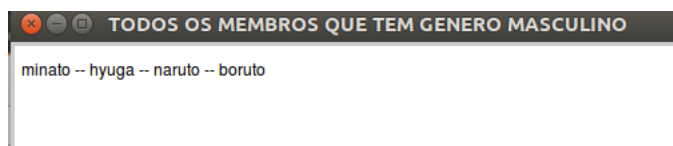
- irmãos, irmão, irmã;
- marido, esposa;
- tios, tio, tia;
- avoMOuavoF, avoM, avoF;
- netoOuNeta, neto, neta;
- primos, primo, prima.

As opções de consultas são: todos os membros que possui o gênero masculino, todos os membros que possui o gênero feminino, filhos de Naruto, DNA paterno e DNA materno. Ilustrado na figura Menu Principal.



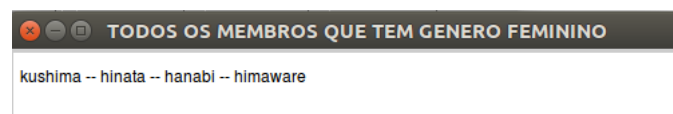
**Figura 2. Menu Principal**

Ao apertar o botão "Masculino", todos os membros do respectivo gênero são exibidos como mostra a figura 3.



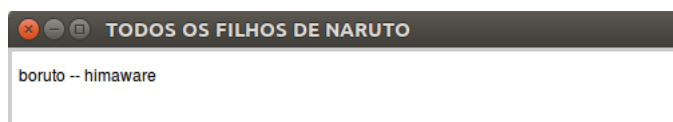
**Figura 3.**

Ao apertar o botão "Feminino", todos os membros do respectivo gênero são exibidos como mostra a figura 4.



**Figura 4.**

Ao apertar o botão "Filhos naruto", os membros Boruto e Himaware são exibidos como mostra a figura 5.



**Figura 5.**

Ao apertar o botão "Dna paterno", outra janela é aberta, seu modelo é ilustrado na figura abaixo. Essa nova janela tem a ideia de afirmar se um membro K é pai de um membro Z,

têm duas caixas de texto onde pode-se colocar o nome de membros. Uma caixa de texto é destinada para o pai e outra para o filho. Existe um botão resultado e ao apertar, ele retorna verdade se K for pai de Z e falso senão for. Outra nova janela é aberta para exibir o retorno.

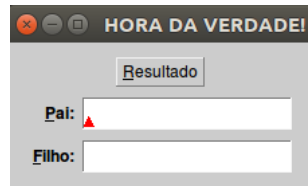


Figura 6. HORA DA VERDADE!

Na figura 7 e na figura 8 são exibidos os retornos, verdade e falso, respectivamente.

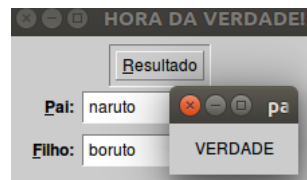


Figura 7.

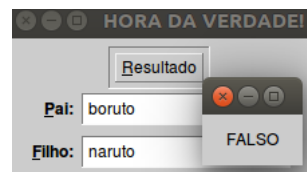


Figura 8.

Ao apertar o botão "Dna materno", acontece o processo semelhante ao apertar o botão "Dna paterno", a única diferença é que o "teste de DNA" é restrito para saber se um membro N é a mãe de outro membro Y.

### 3. Considerações Finais

Como podemos ver, o Prolog é uma linguagem muito poderosa, visto que seu paradigma de Programação em Lógica pode ser usada em diversas áreas, como por exemplo, inteligência artificial. Seus conceitos foram mostramos, mas vale lembrar que vimos apenas o básico do Prolog, existem muitas outras propriedades não abordadas nesse relatório, e muitas implementações diferentes da linguagem, cada um com o seu próprio padrão.

### Referências

- Bergin Jr, T. J. and Gibson Jr, R. G. (1996). *History of programming languages—II*. ACM.
- Palazzo, L. A. (1997). Introdução à programação.
- Wielemaker, J. and Anjewierden, A. (1992). Programming in xpc/prolog. *SWI, University of Amsterdam, Roetersstraat, 15*(1018):1992–2002.