Университет ИТМО

Факультет программной инженерии и компьютерной техники

Системы искусственного интеллекта

Лабораторная работа №1

### ProLog

Выполнил: Анищенко А.А.

Группа P33113

Преподаватель: Болдырева Е.А.

Санкт-Петербург

2020 г.

### Цель работы

Создать описание генеалогического дерева на языке ProLog.

### Схема дерева

Изображение выглядит как монитор, экран, телевидение, контроль

Автоматически созданное описание

### Листинг метода

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  108  109  110  111  112  113  114  115  116  117  118  119  120  121  122  123  124  125  126  127 | **parent**(anischenko\_alexey, anischenko\_anatoliy).  **parent**(anischenko\_julia, anischenko\_anatoliy).  **parent**(anischenko\_alexey, anischenko\_timofey).  **parent**(anischenko\_julia, anischenko\_timofey).  **parent**(anischenko\_vitaliy, anischenko\_alexey).  **parent**(anischenko\_olga, anischenko\_alexey).  **parent**(anischenko\_vitaliy, anischenko\_igor).  **parent**(anischenko\_olga, anischenko\_igor).  **parent**(kukushkin\_michail, anischenko\_julia).  **parent**(kukushkina\_tatiana, anischenko\_julia).  **parent**(kukushkin\_pavel, kukushkin\_michail).  **parent**(kukushkina\_vera, kukushkin\_michail).  **parent**(kukushkin\_pavel, tsygonyaeva\_elena).  **parent**(kukushkina\_vera, tsygonyaeva\_elena).  **parent**(drobov\_aleksandr, kukushkina\_tatiana).  **parent**(drobova\_nina, kukushkina\_tatiana).  **parent**(drobov\_aleksandr, ilyicheva\_luda).  **parent**(drobova\_nina, ilyicheva\_luda).  **parent**(tsygonyaev\_yuriy, tsygonyaev\_alexey).  **parent**(tsygonyaeva\_elena, tsygonyaev\_alexey).  **parent**(tsygonyaev\_yuriy, tsygonyaev\_vitaliy).  **parent**(tsygonyaeva\_elena, tsygonyaev\_vitaliy).  **parent**(tsygonyaev\_alexey, tsygonyaev\_ivan).  **parent**(tsygonyaeva\_tatiana, tsygonyaev\_ivan).  **parent**(tsygonyaev\_vitaliy, tsygonyaev\_michail).  **parent**(tsygonyaeva\_anna, tsygonyaev\_michail).  **parent**(tsygonyaev\_vitaliy, tsygonyaev\_sergey).  **parent**(tsygonyaeva\_anna, tsygonyaev\_sergey).  **male**(anischenko\_alexey).  **male**(anischenko\_anatoliy).  **male**(anischenko\_timofey).  **male**(anischenko\_igor).  **male**(anischenko\_vitaliy).  **male**(kukushkin\_michail).  **male**(kukushkin\_pavel).  **male**(drobov\_aleksandr).  **male**(tsygonyaev\_yuriy).  **male**(tsygonyaev\_alexey).  **male**(tsygonyaev\_vitaliy).  **male**(tsygonyaev\_ivan).  **male**(tsygonyaev\_michail).  **male**(tsygonyaev\_sergey).  **female**(anischenko\_julia).  **female**(anischenko\_olga).  **female**(kukushkina\_tatiana).  **female**(kukushkina\_vera).  **female**(drobova\_nina).  **female**(ilyicheva\_luda).  **female**(tsygonyaeva\_elena).  **female**(tsygonyaeva\_tatiana).  **female**(tsygonyaeva\_anna).  **spouse**(anischenko\_alexey, anischenko\_julia).  **spouse**(anischenko\_olga, anischenko\_vitaliy).  **spouse**(kukushkin\_michail, kukushkina\_tatiana).  **spouse**(drobov\_aleksandr, drobova\_nina).  **spouse**(kukushkin\_pavel, kukushkina\_vera).  **spouse**(tsygonyaeva\_elena, tsygonyaev\_yuriy).  **spouse**(tsygonyaev\_alexey, tsygonyaeva\_tatiana).  **spouse**(tsygonyaeva\_anna, tsygonyaev\_vitaliy).  **spouse**(X, Y) :-  **spouse**(Y, X).  **predecessor**(X, Y) :- **parent**(X, Y).  **predecessor**(X, Y) :- **parent**(Z, Y), **predecessor**(X, Z).  **relatives**(X, Y) :-  X \= Y,  (  **predecessor**(X, Y);  **predecessor**(Z, X), **predecessor**(Z, Y)  ).  **relatives**(X, Y) :-  **spouse**(Z, Y),  **relatives**(X, Z).  **relatives**(X, Y) :-  **relatives**(Y, X).  **grandparent**(X, Y) :- **parent**(Z, Y), **parent**(X, Z).  **great\_grandparent**(X, Y) :- **grandparent**(Z, Y), **parent**(X, Z).  **sibling**(X, Y) :- **parent**(Z, X), **parent**(Z, Y), X \= Y.  **uncle\_or\_aunt**(X, Y) :- **sibling**(X, Z), **parent**(Z, Y).  **great\_uncle\_or\_aunt**(X, Y) :- **uncle\_or\_aunt**(X, Z), **parent**(Z, Y).  **wife**(X, Y) :- **spouse**(X, Y), **female**(X).  **husband**(X, Y) :- **spouse**(X, Y), **male**(X).  **father**(X, Y) :- **male**(X), **parent**(X, Y).  **mother**(X, Y) :- **female**(X), **parent**(X, Y).  **grandfather**(X, Y) :- **male**(X), **grandparent**(X, Y).  **grandmother**(X, Y) :- **female**(X), **grandparent**(X, Y).  **great\_grandfather**(X, Y) :- **male**(X), **great\_grandparent**(X, Y).  **great\_grandmother**(X, Y) :- **female**(X), **great\_grandparent**(X, Y).  **son**(X, Y) :- **male**(X), **parent**(Y, X).  **daughter**(X, Y) :- **female**(X), **parent**(Y, X).  **grandson**(X, Y) :- **male**(X), **grandparent**(Y, X).  **granddaughter**(X, Y) :- **female**(X), **grandparent**(Y, X).  **great\_grandson**(X, Y) :- **male**(X), **great\_grandparent**(Y, X).  **great\_granddaughter**(X, Y) :- **female**(X), **great\_grandparent**(Y, X).  **brother**(X, Y) :- **male**(X), **sibling**(X, Y).  **sister**(X, Y) :- **female**(X), **sibling**(X, Y).  **uncle**(X, Y) :- **male**(X), **uncle\_or\_aunt**(X, Y).  **aunt**(X, Y) :- **female**(X), **uncle\_or\_aunt**(X, Y).  **nephew**(X, Y) :- **male**(X), **uncle\_or\_aunt**(Y, X).  **niece**(X, Y) :- **female**(X), **uncle\_or\_aunt**(Y, X).  **great\_uncle**(X, Y) :- **male**(X), **great\_uncle\_or\_aunt**(X, Y).  **great\_aunt**(X, Y) :- **female**(X), **great\_uncle\_or\_aunt**(X, Y).  **great\_nephew**(X, Y) :- **male**(X), **great\_uncle\_or\_aunt**(Y, X).  **great\_niece**(X, Y) :- **female**(X), **great\_uncle\_or\_aunt**(Y, X).  **first\_cousin**(X, Y) :-  X \= Y,  not(**sibling**(X, Y)),  **grandparent**(Z, X),  **grandparent**(Z, Y).  **second\_cousin**(X, Y) :-  X \= Y,  not(**sibling**(X, Y)),  not(**first\_cousin**(X, Y)),  **great\_grandparent**(Z, X),  **great\_grandparent**(Z, Y).  **first\_cousin\_once\_removed**(X, Y) :-  (**first\_cousin**(Z, X), **parent**(Z, Y));  (**first\_cousin**(Z, Y), **parent**(Z, X)). |

### Пример работы

Изображение выглядит как снимок экрана

Автоматически созданное описаниеЗдесь всё сработало, абсолютно правильно.

Изображение выглядит как снимок экрана

Автоматически созданное описаниеА вот здесь кажется, что допущена ошибка, хотя на самом деле это не так. Просто используемый интерпретатор ProLog’а не обрабатывает равные предикаты. Таким образом каждая связь встречается 2 раза (Это происходит из-за того, что у каждого брата/сестры по два родителя. По-другому реализовывать данную функцию нельзя, так как если, у брата/сестры может быть только один общий родитель или, например, в таблице может не быть одного из родителей). В частности, из-за этого, не работает рекурсия. Но данная программа также была запущена в компиляторе, где всё отработало абсолютно правильно.

Как видно, здесь программа тоже отработала правильно.

Вывод

Выполнив данную лабораторную работу, я изучил основы языка ProLog и создал с его помощью генеалогическое дерево. С помощью языка ProLog можно описывать логические процессы, строить взаимосвязи и решать различные логические задачи (некоторые из которых решаются значительно проще чем на других языках программирования).