Министерство науки и высшего образования РФ

Севастопольский государственный университет

Кафедра информатики и управления в технических системах

ОТЧЕТ

по лабораторной работе №2

ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ ПРОЛОГ

по дисциплине «Логическое программирование»

Выполнил:

Студент группы ИВТ/б 22-о

Черняев Н.Г.

Проверил:

Ткаченко К.С.

г. Севастополь 2019

**Цель работы**

Изучить основы синтаксиса языка Пролог, выработать навыки работы с интерактивной системой Пролог, научиться оформлять отношения между данными на языке Пролог на примере родственных отношений между членами семьи.

**Вариант № 23(24, 29, 4)**

Задание:

* Изобразить граф, иллюстрирующий описываемые родственные отношения.
* Составить программу, которая описывает родственные отношения.
* В качестве фактов описать унарные отношения: мужчина, женщина; и бинарные: состоят\_в\_браке, родитель\_ребенок. Записать правила, которые определяют следующие отношения: муж, жена, родитель, ребенок, сын, дочь, брат, сестра, дядя, тетя, бабушка, дедушка.
* Подобрать тестовые данные, проверяющие все полученные отношения. Тестовые данные должны содержать операции «И», «ИЛИ», «НЕ».
* Выполнить задания по вариантам:
  + определить предикат «муж папиной сестры»(24).
  + определить предикат «прадед»(29).
  + определить предикат «двоюродная сестра»(4).
* Использовать приложение PIE (Prolog Inference Engine - Машина Вывода Пролога), который включен в комплект системы программирования Visual Prolog для отладки программы «родственные отношения».

**Ход работы**

1. Изобразим граф, иллюстрирующий описываемые родственные отношения

|  |  |
| --- | --- |
| Условное обозначение | Отношение |
|  | Родитель - ребенок |
|  | Состоит в браке |
|  | Дядя/тетя – племянник/племянница |
|  | Брат/сестра |
|  | Двоюродный брат/сестра |

1. Составим программу, которая описывает родственные отношения(в качестве фактов описать унарные отношения: мужчина, женщина; и бинарные: состоят\_в\_браке, родитель\_ребенок. Записать правила, которые определяют следующие отношения: муж, жена, родитель, ребенок, сын, дочь, брат, сестра, дядя, тетя, бабушка, дедушка):

man(stepa).

man(bob).  
woman(zoya).  
woman(olya).  
woman(pam).  
man(tom).  
married(pam, tom).  
woman(liz).  
man(jim).  
married(liz, jim).  
woman(ann).  
man(vasya).  
man(petya).

child(olya, zoya).  
child(pam, zoya).  
child(ann, pam).  
child(vasya, pam).  
child(tom, bob).  
child(liz, bob).  
child(bob, stepa).  
child(petya, liz).  
  
wife(X,Y):-married(X,Y),woman(X).  %жена   
husband(X,Y):-married(X,Y),man(X). % муж   
child(C,Y):-married(X,Y),child(C,X),!. % Если является ребёнком одного из родителей то автоматически является ребёнком второго родителя   
doughter(X,Y):-child(X,Y),woman(X).    
son(X,Y):-child(X,Y),man(X).   
grandmother(G,C):-parent(X,C),parent(G,X),woman(G). %бабушка   
grandfather(G,C):-parent(X,C),parent(G,X),man(G).   %дедушка   
parent(D,X):-child(X,D). %родитель, если является ребёнком   
brother(A,Y):-parent(P,Y),parent(P,A),man(A).   %брат   
sister(X,Y):-child(X,P),child(Y,P),woman(X).  %сестра   
uncle(U,C):-parent(P,C),brother(U,P). %дядя   
aunt(A,C):-child(C,P),sister(A,P). %тётя

1. Подберем тестовые данные, проверяющие полученные отношения.

rr1:-parent(tom, ann),  
    brother(tom, liz),   
    aunt(pam, vasya),     
    grandfather(stepa, liz).



1. Выполним задания по вариантам
   1. определим предикат «муж папиной сестры»

husband\_of\_father\_sister(H,X):-married(Y,H),brother(B,Y),parent(B,X).

Дерево И/ИЛИ:

husband\_of\_father\_sister(H,X)

married(Y,H)

brother(B,Y)

parent(B,X)

Проверим его работу:

rr2:-husband\_of\_father\_sister(jim,vasya).

rr2.  
Trace: >> CALL:   rr2()  
Trace: >> CALL:   husband\_of\_father\_sister("jim","vasya")  
Trace: >> CALL:   married(Y$38,"jim")  
Trace: >> RETURN: married("liz","jim")  
Trace: >> CALL:   brother(B$39,"liz")  
Trace: >> CALL:   parent(P$40,"liz")  
Trace: >> CALL:   child("liz",P$40)  
Trace: >> RETURN: child("liz","bob")  
Trace: >> RETURN: parent("bob","liz")  
Trace: >> CALL:   parent("bob",B$39)  
Trace: >> CALL:   child(B$39,"bob")  
Trace: >> RETURN: child("tom","bob")  
Trace: >> RETURN: parent("bob","tom")  
Trace: >> CALL:   man("tom")  
Trace: >> RETURN: man("tom")  
Trace: >> RETURN: brother("tom","liz")  
Trace: >> CALL:   parent("tom","vasya")  
Trace: >> CALL:   child("vasya","tom")  
Trace: >> CALL:   married(X$41,"tom")  
Trace: >> RETURN: married("pam","tom")  
Trace: >> CALL:   child("vasya","pam")  
Trace: >> RETURN: child("vasya","pam")  
Trace: >> RETURN: child("vasya","tom")  
Trace: >> RETURN: parent("tom","vasya")  
Trace: >> RETURN: husband\_of\_father\_sister("jim","vasya")  
Trace: >> RETURN: rr2()  
True  
1 Solution

* 1. определить предикат «прадед»

grandgrandfather(G,X):-man(G),parent(G,Y),parent(Y,C),parent(C,X).

Дерево И/ИЛИ:

husband\_of\_father\_sister(H,X)

man(G)

parent(G,Y)

parent(C,X)

parent(Y,C)

Проверим его работу:

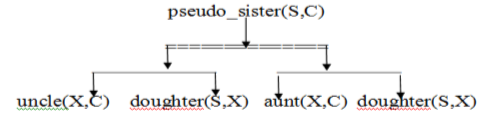
rr3:-grandgrandfather(stepa,vasya)

rr3.  
Trace: >> CALL:   rr3()  
Trace: >> CALL:   grandgrandfather("stepa","vasya")  
Trace: >> CALL:   man("stepa")  
Trace: >> RETURN: man("stepa")  
Trace: >> CALL:   parent("stepa",Y$35)  
Trace: >> CALL:   child(Y$35,"stepa")  
Trace: >> RETURN: child("bob","stepa")  
Trace: >> RETURN: parent("stepa","bob")  
Trace: >> CALL:   parent("bob",C$36)  
Trace: >> CALL:   child(C$36,"bob")  
Trace: >> RETURN: child("tom","bob")  
Trace: >> RETURN: parent("bob","tom")  
Trace: >> CALL:   parent("tom","vasya")  
Trace: >> CALL:   child("vasya","tom")  
Trace: >> CALL:   married(X$37,"tom")  
Trace: >> RETURN: married("pam","tom")  
Trace: >> CALL:   child("vasya","pam")  
Trace: >> RETURN: child("vasya","pam")  
Trace: >> RETURN: child("vasya","tom")  
Trace: >> RETURN: parent("tom","vasya")  
Trace: >> RETURN: grandgrandfather("stepa","vasya")  
Trace: >> RETURN: rr3()  
True  
1 Solution

* 1. определить предикат «двоюродная сестра»

pseudo\_sister(S,C):-uncle(X,C),doughter(S,X).  
pseudo\_sister(S,C):-aunt(X,C),doughter(S,X).

Дерево И/ИЛИ:



Проверим его работу:

rr4:-pseudo\_sister(ann,petya).

rr4.  
Trace: >> CALL:   rr4()  
Trace: >> CALL:   pseudo\_sister("ann","petya")  
Trace: >> CALL:   uncle(X$31,"petya")  
Trace: >> CALL:   parent(P$32,"petya")  
Trace: >> CALL:   child("petya",P$32)  
Trace: >> RETURN: child("petya","liz")  
Trace: >> RETURN: parent("liz","petya")  
Trace: >> CALL:   brother(X$31,"liz")  
Trace: >> CALL:   parent(P$33,"liz")  
Trace: >> CALL:   child("liz",P$33)  
Trace: >> RETURN: child("liz","bob")  
Trace: >> RETURN: parent("bob","liz")  
Trace: >> CALL:   parent("bob",X$31)  
Trace: >> CALL:   child(X$31,"bob")  
Trace: >> RETURN: child("tom","bob")  
Trace: >> RETURN: parent("bob","tom")  
Trace: >> CALL:   man("tom")  
Trace: >> RETURN: man("tom")  
Trace: >> RETURN: brother("tom","liz")  
Trace: >> RETURN: uncle("tom","petya")  
Trace: >> CALL:   doughter("ann","tom")  
Trace: >> CALL:   child("ann","tom")  
Trace: >> CALL:   married(X$34,"tom")  
Trace: >> RETURN: married("pam","tom")  
Trace: >> CALL:   child("ann","pam")  
Trace: >> RETURN: child("ann","pam")  
Trace: >> RETURN: child("ann","tom")  
Trace: >> CALL:   woman("ann")  
Trace: >> RETURN: woman("ann")  
Trace: >> RETURN: doughter("ann","tom")  
Trace: >> RETURN: pseudo\_sister("ann","petya")  
Trace: >> RETURN: rr4()  
True  
1 Solution

**Выводы**

В данной лабораторной работе были изучены основы синтаксиса языка Пролог, выработаны навыки работы с интерактивной системой Пролог, а также получены умения оформлять отношения между данными на языке Пролог на примере родственных отношений между членами семьи.