**Задания по дисциплине «Рекурсивно-логическое программирование»**

**Задание 1. Определение отношений в программе**

летает("синица"). % факты

летает("лебедь").

летает("аэроплан").

животное("лебедь").

животное("синица").

животное("тигр").

птица("пингвин").

птица("страус").

птица(X):- животное(X), летает(X). % правило

?- птица(A). % запрос

1. Какие элементы принадлежат отношению «птица», определенному в программе «Птицы»?

2. Запустите приложение PIE, поместите в него программу «Птицы» и найдите ответ на запрос к программе. Для этого откройте новый файл, поместите в него текст программы (факты и правила), сделайте активным окно с программой и выберите команду меню Engine > Reconsult. В окне Dialog введите запрос птица(A).

**Задание 2. Определение отношений в программе**

Файл family.txt (база данных)

clauses

parent("Иван", "Мария").

parent("Анна", "Мария").

parent("Мария", "Павел").

parent("Мария", "Петр").

parent("Мария", "Елизавета").

spouse("Иван", "Анна").

spouse("Павел", "Юлия").

male("Иван").

male("Павел").

male("Петр").

female("Мария").

female("Анна").

female("Елизавета").

female("Юлия").

Файл main.pro

class facts - relatives

parent: (string Родитель, string Ребенок).

spouse: (string Муж, string Жена).

male: (string).

female: (string).

class predicates

father: (string Отец, string Ребенок) nondeterm anyflow.

mother: (string Мать, string Ребенок) nondeterm (o,o).

clauses

father(X, Y):-

parent(X, Y),

male(X).

mother(X, Y):-

parent(X, Y),

female(X).

run():-

init(),

file::consult("family.txt", relatives),

father(X, Y),

write("отец - ", X, ", ребенок - ", Y), nl,

fail;

mother(X, Y),

write("мать - ", X, ", ребенок - ", Y), nl,

fail;

if father("Иван", "Петр") then

write("\nИван является отцом Петра")

else

write("\nИван не является отцом Петра")

end if,

\_ = readLine().

Задайте следующие вопросы программе:

1. Частный простой запрос (является ли Иван родителем Петра?)

2. Общий простой запрос (найти всех мужчин)

3. Конъюнктивный составной общий запрос (найти сыновей Марии).

4. Простой общий запрос с отбором информации при помощи анонимной переменной (кто из женщин замужем?)

5. Дизъюнктивный составной общий запрос (найти всех персон)

Определите в программе о родственных отношениях следующие бинарные отношения:

* внук, внучка, дедушка и бабушка;
* ребенок как обратное к отношению родитель;
* сын, дочь, сестра и брат;
* супруги как симметричное замыкание отношения супруг.

Проверить работу программы несколькими запросами.

**Задание 3. Определение отношений в программе**

Напишите программу, которая с помощью бинарных отношений владелец, животное и цвет, хранящих пары хозяин – кличка, кличка – животное и кличка – окрас, соответственно, описывает следующие сведения. Майкл владеет рыжим котом, Сьюзен шоколадным, Дэн и Пит владеют серыми котами. Билл имеет собаку серого окраса, а Бетти шоколадного. Все животные имеют уникальные клички (придумайте их самостоятельно). Составьте запросы к программе:

* найти владельцев серых котов;
* найти животного Билла и цвет этого животного;
* найти животных, которыми владеют Бетти и Майкл;
* найти владельцев животных шоколадного окраса.

**Задание 4. Машина вывода Пролога**

publication = book(author, string Название, edition Издание);

magazine(string Название, integer Номер, integer Год).

author = author(string Фамилия, string Имя, string Отчество).

edition = edition(string Место, string Издательство, integer Год).

class facts

library: (publication).

clauses

library(magazine("Компьютерра", 2, 2009)).

library(magazine("Наука и жизнь", 11, 2012)).

library(book(author("Чехов", "Антон", "Павлович"),

"Избранное", edition("Москва", "АСТ, Астрель", 2003))).

library(book(author("Великова", "Людмила", "Викторовна"),

"Русский язык", edition("Москва", "МЦНМО", 2003))).

run():-

% Что есть в библиотеке?

library(X),

write(X), nl,

fail;

% Названия книг, изданных в 2003 году

library(book(\_, Title, edition(\_, \_, 2003))),

write(Title), nl,

fail;

\_ = readLine().

Добавьте в программу "Библиотека" новые факты, содержащие сведения о книгах и журналах. Используя анонимные переменные, найдите ответы на запросы:

1. За какие годы имеются журналы в библиотеке?
2. Найдите книги, изданные в Москве или в Санкт-Петербурге.
3. Какие журналы за позапрошлый год имеются в библиотеке? (Текущий год определяется с помощью предиката getDate).
4. Найдите самую старую по году издания литературу.

Проведите трассировку запросов.

**Задание 5. Управление перебором. Отсечение**

Жители острова A, B и С, один из которых всегда говорил правду, другой всегда лгал, а третий был хитрецом — иногда говорил правду, а иногда лгал, сообщили о себе следующее:

A: "Я хитрец";

B: "Да, A хитрец";

C: "Я не хитрец".

Напишите программу на Visual Prolog и определите, кто из них кем был на самом деле.

**Задание 6. Внутренняя база данных**

Создайте базу данных "Мое расписание", в которой хранятся факты для описания отношения расписание вида:

расписание(время(пт, 1),

предмет("Интеллектуальные системы", лекция),

преп("Аншаков О.М.", профессор), место(416, 2)).

Факты должны содержать следующие сведения: день недели и номер пары; предмет — название и вид занятия (лекционное и т. д.); фамилия и инициалы преподавателя, его должность; аудитория и корпус.

Найдите ответы на следующие вопросы.

1. Кто ведет "Английский язык"?
2. Когда и где проходят занятия по курсу "Экспертные системы"?
3. В каких аудиториях проводятся занятия по вторникам и средам?
4. В какие дни недели ровно три пары?
5. По какие предметам лабораторные занятия ведет только лектор.

**Задание 7. Рекурсия**

Для заданного натурального n вычислите значение 1/sin 1 + 1/(sin 1 + sin 2) + … + 1/(sin 1 + … + sin n).

**Задание 8. Списки. Полиморфизм**

Проверьте, является ли список палиндромом.

**Задание 9. Деревья**

Найдите количество терминальных вершин произвольного дерева, составьте и выведите список этих вершин.

**Задание 10. Арифметика и строки**

Напишите программу, которая переводит

1. римские числа в арабские;
2. арабские числа в римские.

**Задание 11. Игры**

Реализуйте игру в 37 пользователя с компьютером. Игроки по очереди называют одно из чисел от 1 до 5, названные числа суммируются. Нельзя повторять число, которое назвал противник на предыдущем ходе. Выигрывает участник, получивший ровно 37 очков или вынудивший противника превысить это число.

**Задание 12. Экспертная система.**

Создайте экспертную систему по выбору направления обучения для абитуриента.