Федеральное государственное бюджетное образовательное учреждение высшего

образования

«Чувашский государственный университет им. И.Н. Ульянова»

Кафедра вычислительной техники

Лабораторная работа № 6

**Динамические базы данных**

Вариант 9

Выполнил: Иванов В.С.

студент группы ИВТ-41-22

Проверил: кандидат технических наук

Обломов Игорь Александрович

Чебоксары, 2024

Цель работы: изучение основ динамической базы данных в прологе, ее создание.

Основные теоретические сведения.

Программная среда Пролога состоит из двух частей:

* программного модуля Пролога;
* *базы данных* (БД) Пролога.

Программный модуль включает в себя интерпретатор (компилятор), драйвер ввода-вывода и набор системных процедур. Пользователь обычно не имеет возможности изменять программный модуль.

База данных Пролога содержит набор основных, заранее определенных предикатов. Кроме того, резервируется пространство для добавления пользовательских предикатов, представляющих собой программу на Прологе. Механизм добавления, удаления и обновления утверждений пользователя классифицируется следующим образом:

1. добавление и удаление утверждений;
2. добавление утверждений из файла.

Пролог программа представляется реляционной БД, то есть как описание некоторого множества отношений. Описание отношений в ней присутствует в явном виде (факты) или в неявном виде (правила).

Предикаты для работы с БД таковы: assert, asserta, assertz, assert\_in, asserta\_in, retract, retractall. Имена предикатов и их количество зависит от реализации языка.

Предикат assert(<the fact>) – всегда успешен, а в качестве своего побочного эффекта добавляет факт к текущей базе данных. Предикат retract(<the fact>) – приводит к противоположному эффекту: удаляет сопоставимый c <the fact> факт из базы данных.

Следует отметить, что добавленный факт будет присутствовать в текущей БД до конца текущего сеанса работы. Однако он будет «не видимым» для пользователя, т. е. не будет отображаться в рабочем окне. Для того чтобы факт появился в рабочем окне, необходимо использовать предикат assert\_in.

Задание.

9. Работник отдела кадров милиции при приеме сотрудника в карточке, которая заводится на каждого сотрудника, должен указать: ФИО, год рождения, пол, образование, воинское звание. Каждые два года офицер повышается в звании, если за это время не было взысканий. После выслуги в 25 лет работник отправляется на пенсию. Требуется получение из БД следующих сведений: список офицеров, уходящих на пенсию в наступающем году; список офицеров представленных на повышение звания; список офицеров, принятых на работу с начала текущего года.

Код программы:

rab("Pe Pe Pe",1977,male,mid,"general") .

rab("A A A",2002,male,hight,"copter") .

%rab(FIO, birth year, sex, education, rank).

rab("Kozh Kir Yur", 2004, male, hight, "captain").

rab("Me Me Me", 1912, male, mid, "general").

rab("Pu Pu Pu", 1986, male, mid, "general").

rab("Ma Ma Ma", 2000, male,hight, "sergeant").

%punishments

punishment("Kozh Kir Yur", "genocide").

has\_punishment(FIO) :-

punishment(FIO, \_).

rank\_up(FIO, Birth, Sex, Education, Rank):-

rab(FIO, Birth, Sex, Education, Rank),

CurYear = 2024,

ServYears is CurYear - Birth - 22,

ServYears >2,

0 is ServYears mod 2,

not(has\_punishment(FIO)).

next\_year\_on\_pensia(FIO, Birth, Sex, Education, Rank):-

rab(FIO, Birth, Sex, Education, Rank),

CurYear = 2024,

ServYears is CurYear - Birth - 22,

ServYears >= 25.

hired\_in\_year(FIO, Birth, Sex, Education, Rank):-

rab(FIO, Birth, Sex, Education, Rank),

CurYear = 2024,

ServYears is Birth + 22,

ServYears =:= CurYear.

?- hired\_in\_year(FIO, Birth, Sex, Education, Rank),write("Hired in year:"), nl, write(FIO), nl.

?- rank\_up(FIO, Birth, Sex, Education, Rank), write("Rank up:"), nl, write(FIO), nl.

?-next\_year\_on\_pensia(FIO, Birth, Sex, Education, Rank), write("Next year on pensia:"), nl, write(FIO), nl.

Вывод: изучил основы динамической базы данных в прологе, создал БД.