|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ **Информатика и системы управления**

КАФЕДРА **ПРОГРАМНОЕ ОБЕСПЕЧЕНИЕ ЭВМ И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ (ИУ7)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.04.03 ПРОГРАММНАЯ ИНЖЕНЕРИЯ**

**Отчет**

|  |  |
| --- | --- |
| **По лабораторной работе №** | 11, 12, 13 |

**Дисциплина:** Функциональное и логическое программирование

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ7-62Б |  |  | Н.А. Гарасев |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | Н.Б.Толпинская |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2021

**Настройка TestGoal**

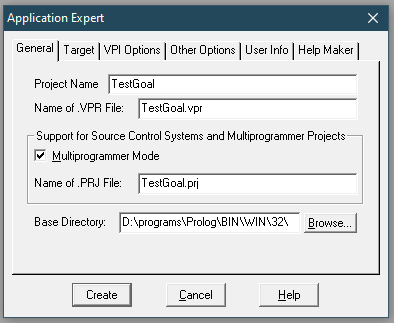


Рисунок 1. Создание нового проекта

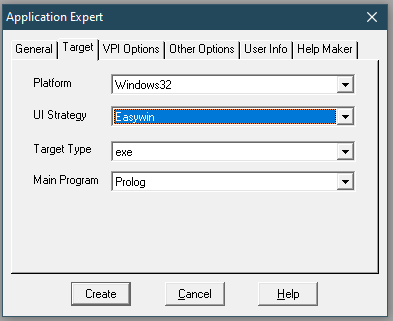


Рисунок 2. Настройка проекта

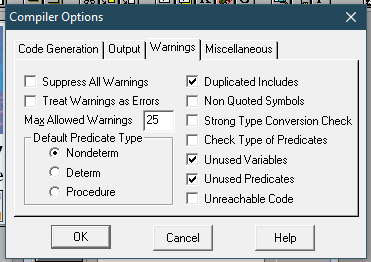


Рисунок 3. Настройка компилятора

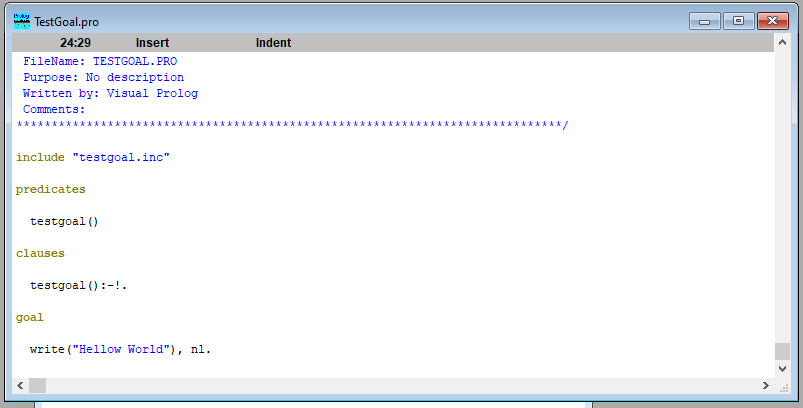


Рисунок 4. Программа Hello World

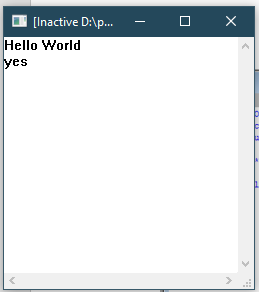


Рисунок 5. Результат программы

**Программа о студентах.**

domains

name, university = symbol.

predicates

study(name, university).

clauses

study("Garasev Nikita", "BMSTU").

study("Ivanov Gosha", "MGU").

study("Pavlov Nikita", "BMSTU").

study("Suchkov Alex", "BMSTU").

study("Kylikov Dmity", "BMSTU").

study("Brednev Ilia", "RGU").

goal

study(NAME, "BMSTU"), nl.

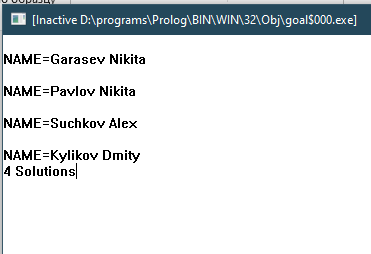


Рисунок 7. Результат программы

**Программа о студентах 2.**

domains

name, university = symbol.

faculty = symbol.

department = integer.

number = integer.

group = group(faculty, department, number).

predicates

study(name, group, university).

clauses

study("Garasev Nikita", group("IU", 7, 62), "BMSTU").

study("Garasev Nikita", group("M", 1, 61), "RGU").

study("Garasev Nikita", group("MT", 3, 23), "MGU").

study("Ivanov Gosha", group("MT", 3, 23), "MGU").

study("Pavlov Nikita", group("IU", 5, 62), "BMSTU").

study("Suchkov Alex", group("IU", 7, 61), "BMSTU").

study("Kylikov Dmity", group("IU", 7, 62), "BMSTU").

study("Brednev Ilia", group("M", 2, 61), "RGU").

goal

study("Garasev Nikita", GROUP, UNIVERSITY), nl.

%study(NAME, group("IU", 7, N), "BMSTU"), nl.

%study(NAME, GROUP, "RGU"), nl.

**Задание 3.**

Составить программу, т.е. модель предметной области – базу знаний, объединив в ней информацию – знания:

* «Телефонный справочник»: Фамилия, №тел, Адрес – структура (Город, Улица, №дома, №кв),
* «Автомобили»: Фамилия\_владельца, Марка, Цвет, Стоимость, и др.,
* «Вкладчики банков»: Фамилия, Банк, счет, сумма, др.

Владелец может иметь несколько телефонов, автомобилей, вкладов (Факты).

Используя правила, обеспечить возможность поиска

1. Первая часть

а) По № телефона найти: Фамилию, Марку автомобиля, Стоимость автомобиля (может быть несколько),

б) Используя сформированное в пункте а) правило, по № телефона найти: только Марку автомобиля (автомобилей может быть несколько),

1. Используя простой, не составной вопрос: по Фамилии (уникальна в городе, но в разных городах есть однофамильцы) и Городу проживания найти: Улицу проживания, Банки, в которых есть вклады и №телефона.

**Для задания 1 и задания 2:**

для одного из вариантов ответов, и для а) и для б), описать словесно порядок поиска ответа на вопрос, указав, как выбираются знания, и, при этом, для каждого этапа унификации, выписать подстановку – наибольший общий унификатор, и соответствующие примеры термов.

domains

surname = symbol.

phone = symbol.

city = symbol.

street = symbol.

house = integer.

flat = integer.

adress = adress(city, street, house, flat).

brand = symbol.

color = symbol.

price = integer.

bank = symbol.

account = symbol.

amount = integer.

predicates

person(surname, phone, adress).

car(surname, brand, color, price).

depositor(surname, bank, account, amount).

carByPhone(phone, surname, brand, price).

modelByPhone(phone, brand).

accountAndPhoneBySurnameAndCity(surname, city, street, phone, bank).

clauses

person("Garasev", "89273141395", adress("Moscow", "Kapotnoua", 19, 110)).

person("Kulikov", "89271939173", adress("Saransk", "Kovalenko", 7, 89)).

person("Kulikov", "89272030284", adress("Saransk", "Kovalenko", 7, 89)).

person("Chalyy", "892722822832", adress("Anadyr", "Morshovaya", 18, 322)).

person("Kulikov", "89273222280", adress("Moscow", "Kurskya", 25, 2)).

person("Kulikov", "89273222281", adress("Moscow", "Kurskya", 25, 2)).

car("Garasev", "BMW X6", "pink", 7500000).

car("Kulikov", "Audi A6", "white", 4500000).

car("Kulikov", "Audi A4", "black", 1500000).

car("Chalyy", "Bugatti", "gold", 7500000).

car("Chalyy", "Lada Vesta", "white", 500000).

depositor("Garasev", "Tinkoff", "22822", 7500).

depositor("Garasev", "VTB", "23822", 6500).

depositor("Kulikov", "Sberbank", "11111", 1500).

depositor("Kulikov", "Tinkoff", "22222", 3500).

depositor("Chalyy", "SelhozBank", "33333", 500).

carByPhone(Phone, Surname, Brand, Price) :- person(Surname, Phone,\_),

car(Surname, Brand, \_, Price).

modelByPhone(Phone, Brand) :- carByPhone(Phone, \_, Brand, \_).

accountAndPhoneBySurnameAndCity(Surname, City, Street, Phone, Bank) :-

person(Surname, Phone, adress(City, Street, \_, \_)), depositor(Surname, Bank, \_, \_).

goal

carByPhone("89271939173", Surname, Brand, Price), nl.

%modelByPhone("892722822832", Brand).

%accountAndPhoneBySurnameAndCity("Kulikov", "Saransk", Street, Phone, Bank), nl.

**Контрольные вопросы:**

**1. Что такое терм?**

Терм — это объект знаний в Prolog. Термы могут быть константами, переменными или составными термами. В свою очередь константы могут быть числовыми, символьными атомами или строками. Переменные могут быть именованными или анонимными. Составные термы используются для обозначения отношений между объектами, они объединяют отдельные элементы знаний в единый объект.

**2. Что такое предикат в матлогике (математике)?**

Предикат в матлогике — это функция, которая возвращает одно из двух значений: 0 («ложь») или 1 («Истина»).

**3. Что описывает предикат в Prolog?**

В Prolog предикат описывает отношение между объектами. Это отношение также как в математике может быть истинным.

**4. Назовите виды предложений в программе и приведите примеры таких предложений из Вашей программы. Какие предложения являются основными, а какие – не основными? Каковы: синтаксис и семантика (формальный смысл) этих предложений (основных и неосновных)?**

Виды предложений в программе:

* Факты — с их помощью записываются знания.  
  Пример: car(petrov, volvo, 1223122, red).
* Правила — это обобщенная запись знаний и условий, при которых знание является истиной.   
  Пример: carByPhone(Phone,Surname,Brand,Price):- person(Surname,Phone, \_ ),

car(Surname, Brand, Price,\_).

* Вопросы — используются для выяснения истинности какого-либо знания и

Пример: carByPhone("89271939173”, Surname, Brand, Price).

Основные предложения — это предложения, которые не содержат переменных.

Неосновные предложения — это предложения, которые содержат переменные в момент фиксации программы.

Синтаксически предложения записываются как

A:- B1,... , Bn.

А — это заголовок правила, а B1, ..., Bn — тело правила.

Правила состоят и из заголовка, и из тела. Факты — частный случай правил, состоят только из заголовка. Вопросы состоят только из тела.

5. **Каковы назначение, виды и особенности использования переменных в программе на Prolog? Какое предложение БЗ сформулировано в более общей – абстрактной форме: содержащее или не содержащее переменных?**

Переменные предназначены для передачи знаний.

Переменные могут быть именованные и анонимные.

Если переменная не имеет значения, то она называется свободной, иначе — конкретизированной. Анонимные переменные не могут быть связаны со значением. Анонимные переменные используются в случаях, когда необходимо использовать переменную, однако ее значение не существенно.

Именованные переменные уникальны в рамках одного предложения, т. е. в разных предложениях одно и то же имя переменной может использоваться для обозначения разных объектов. Анонимные переменные уникальны везде.

Все переменные без типовые, в процессе вычисления они могут связываться с любыми объектами.

Предложение, содержащее переменные сформулировано в более общей-абстрактной форме, так как несколько знаний могут подойти под одно предложение.

**6. Что такое подстановка?**

Подстановкой называется множество пар, вида: {xi = ti}, где x i – переменная, а ti – терм. Каждой переменной xi в соответствие ставится терм ti.

**7. Что такое пример терма? Как и когда строится? Как Вы думаете, система строит и хранит примеры?**

Пример терма А — это терм B такой, что существует такая подстановка θ, что B = Aθ.

Примеры термов строятся в процессе унификации, когда происходит сравнение двух термов, с помощью подстановки всех возможных значений из базы знаний. Пример хранится в памяти для продолжения доказательства. Он уничтожается, когда система заходит в тупик или на вопрос можно дать ответ «да».