|  |  |
| --- | --- |
| 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 №** | 1 14 |

**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

**Цель работы – получить навыки построения модели предметной области, разработки и оформления программы на Prolog, изучить принципы, логику формирования программы и отдельные шаги выполнения программы на Prolog**.

**Задачи работы**: приобрести навыки декларативного описания предметной области с использованием фактов и правил. Изучить способы использования термов, переменных, фактов и правил в программе на Prolog, принципы и правила сопоставления и отождествления, порядок унификации.

Задание на лабораторную работу.

Используя базу знаний, хранящую знания (лаб. 13):

**«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). В разных городах есть однофамильцы, в одном городе – фамилия уникальна.

Используя **конъюнктивное правило и простой вопрос**, обеспечить возможность поиска:

***По Марке и Цвету автомобиля найти Фамилию, Город, Телефон и Банки, в которых владелец автомобиля имеет вклады.*** Лишней информации не находить и не передавать!!!

Владельцев может быть несколько (не более 3-х), один и ни одного.

1. Для каждого из трех вариантов словесно подробно описать порядок формирования ответа (в виде таблицы). При этом, указать – отметить моменты очередного запуска алгоритма унификации и полный результат его работы. Обосновать следующий шаг работы системы. Выписать унификаторы – подстановки. Указать моменты, причины и результат отката, если он есть.

2. Для случая нескольких владельцев (2-х):

приведите примеры (таблицы) работы системы при разных порядках следования в БЗ процедур, и знаний в них: («Телефонный справочник», «Автомобили», «Вкладчики банков», или: «Автомобили», «Вкладчики банков», «Телефонный справочник»). Сделайте **вывод:** Одинаковы ли: множество работ и объем работ в разных случаях?

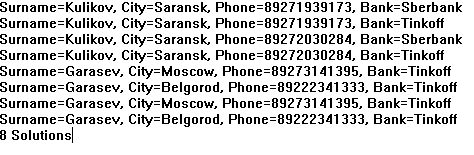
3. Оформите 2 таблицы, демонстрирующие порядок работы алгоритма унификации вопроса и подходящего заголовка правила (для двух случаев из пункта 2) и укажите результаты его работы: ответ и побочный эффект.

**Текст программы**

|  |
| --- |
| 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).  infoByBrandAndColor(brand, color, surname, city, phone, bank).    clauses  person("Garasev", "89273141395", adress("Moscow", "Kapotnoua", 19, 110)).  person("Garasev", "89222341333", adress("Belgorod", "Smolenskaya", 29, 112)).  person("Kulikov", "89271939173", adress("Saransk", "Kovalenko", 7, 89)).  person("Kulikov", "89272030284", adress("Saransk", "Kovalenko", 7, 89)).  person("Garasev", "89273141395", adress("Moscow", "Kapotnoua", 19, 110)).  person("Garasev", "89222341333", adress("Belgorod", "Smolenskaya", 29, 112)).  person("Chalyy", "892722822832", adress("Anadyr", "Morshovaya", 18, 322)).    car("Kulikov", "Audi", "white", 4500000).  car("Kulikov", "Audi", "black", 1500000).  car("Garasev", "BMW X6", "pink", 7500000).  car("Garasev", "Audi", "black", 1500000).  car("Chalyy", "Bugatti", "gold", 7500000).  car("Chalyy", "Lada Vesta", "white", 500000).    depositor("Kulikov", "Sberbank", "11111", 1500).  depositor("Kulikov", "Tinkoff", "22222", 3500).  depositor("Garasev", "Tinkoff", "22822", 7500).  depositor("Chalyy", "SelhozBank", "33333", 500).    infoByBrandAndColor(Brand, Color, Surname, City, Phone, Bank)  :- car(Surname, Brand, Color, \_),  person(Surname, Phone, adress(City, \_, \_, \_)),  depositor(Surname, Bank, \_ , \_). |
|  |

**Демонстрация работы программы**

**Goal** infoByBrandAndColor("Audi", "black", Surname, City, Phone, Bank).



**Goal** infoByBrandAndColor("Bugatti", "gold", Surname, City, Phone, Bank).



**Goal** infoByBrandAndColor("Bugatti", "black", Surname, City, Phone, Bank).



**Задание 1**

**Владельцев несколько**

*Вопрос infoByBrandAndColor("Audi", "black", SurnameQ, CityQ, PhoneQ, BankQ).*

|  |  |  |
| --- | --- | --- |
| **Шаг** | **Сравнимые термы; результат, подстановка (если есть)** | **Действие** |
| 0 | infoByBrandAndColor("Audi", "black", SurnameQ, CityQ, PhoneQ, BankQ).= person("Kulikov", "89271939173", adress("Saransk", "Kovalenko", 7, 89)).  Разные функторы  Унификация не успешна | **Текущее состояние резольвенты:**  *infoByBrandAndColor("Audi", "black", SurnameQ, CityQ, PhoneQ, BankQ).*  Откат, переход к следующему предложению. |
| ... | ... | ... |
| 5 | infoByBrandAndColor("Audi", "black", SurnameQ, CityQ, PhoneQ, BankQ).= car("Kulikov", "Audi", "white", 4500000).  Разные функторы  Унификация не успешна | Откат, переход к следующему предложению. |
| ... | ... | ... |
| 11 | infoByBrandAndColor("Audi", "black", SurnameQ, CityQ, PhoneQ, BankQ).= depositor("Kulikov", "Sberbank", "11111", 1500).  Разные функторы  Унификация не успешна | Откат, переход к следующему предложению. |
| ... | ... | ... |
| 15 | infoByBrandAndColor("Audi", "black", SurnameQ, CityQ, PhoneQ, BankQ).= infoByBrandAndColor(Brand, Color, Surname, City, Phone, Bank)  Унификация успешна  Θ = {"Audi" = Brand, "black" = Color, SurnameQ = Surname, CityQ = City, PhoneQ = Phone,  BankQ = Bank} | **Новое состояние резольвенты:**  *car(SurnameQ, "Audi",”black", \_,)*  *person(SurnameQ, PhoneQ, address(CityQ, \_, \_, \_))*  *depositor(SurnameQ, BankQ, \_, \_)* |
| 16 | car(SurnameQ, "Audi","black", \_, CityQ) = person("Kulikov", "89271939173", adress("Saransk", "Kovalenko", 7, 89)).  Разные функторы  Унификация не успешна | Откат, переход к следующему предложению. |
| ... | ... | ... |
| 21 | car(SurnameQ, "Audi", "black", \_, CityQ) = car("Kulikov", "Audi", "white", 4500000).  Различаются константы color  Унификация не успешна | **Новое состояние резольвенты:**  Откат, переход к следующему предложению. |
| 22 | car(SurnameQ, "Audi","black", \_, CityQ) = car("Kulikov", "Audi", "black", 1500000).    Унификация успешна  Θ = {"Audi" = Brand, "black" = Color, SurnameQ = "Kulikov", CityQ = City, PhoneQ = Phone, BankQ = Bank} | **Новое состояние резольвенты:**  *person(*"Kulikov"*, PhoneQ, address(*CityQ*, \_, \_, \_))*  *depositor(*"Kulikov"*, BankQ, \_, \_)* |
| 23 | person("Kulikov", PhoneQ, address(CityQ, \_, \_, \_))=  person("Kulikov", "89271939173", adress("Saransk", "Kovalenko", 7, 89)).  Унификация успешна  Θ = {"Audi" = Brand, “black" = Color, SurnameQ = "Kulikov", CityQ = “Saransk”, PhoneQ = “89271939173”, BankQ = Bank} | **Новое состояние резольвенты:**  *depositor(*"Kulikov"*, BankQ, \_, \_)* |
| 24 | depositor("Kulikov", BankQ, \_, \_) = person("Kulikov", "89271939173", adress("Saransk", "Kovalenko", 7, 89)).  Разные функторы  Унификация не успешна | Откат, переход к следующему предложению. |
| ... | ... | ... |
| 29 | depositor("Kulikov", BankQ, \_, \_) = car("Kulikov", "Audi", "white", 4500000).  Разные функторы  Унификация не успешна | Откат, переход к следующему предложению. |
| … | … | … |
| 36 | depositor("Kulikov", BankQ, \_, \_) *=* depositor("Kulikov", "Sberbank", "11111", 1500).  Унификация успешна  Θ = {"Audi" = Brand, "black" = Color, SurnameQ = "Kulikov", CityQ = “Saransk”, PhoneQ = “89271939173”, BankQ = “Sberbank”} | **Вывод:**  SurnameQ = "Kulikov", CityQ = “Saransk”, PhoneQ = “89271939173”, BankQ = “Sberbank”  Откат, переход к следующему предложению. |
| 37 | depositor("Kulikov", BankQ, \_, \_) *=* depositor("Kulikov", "Tinkoff", "22222", 3500).  Унификация успешна  Θ = {"Audi" = Brand, "black" = Color, SurnameQ = "Kulikov", CityQ = “Saransk”, PhoneQ = “89271939173”, BankQ = “Tinkoff”} | **Вывод:**  SurnameQ = "Kulikov", CityQ = “Saransk”, PhoneQ = “89271939173”, BankQ = “Tinkoff”  Откат, переход к следующему предложению. |
| 38 | depositor("Kulikov", BankQ, \_, \_) *=* depositor("Garasev", "Tinkoff", "22822", 7500).  Различаются константы surnameУнификация не успешна | Откат, переход к следующему предложению. |
| ... | ... | ... |
| 40 | depositor("Kulikov", BankQ, \_, \_) = infoByBrandAndColor(Brand, Color, Surname, City, Phone, Bank)  Разные главные функторы  Унификация не успешна | **Новое состояние резольвенты:**  *person(*"Kulikov"*, PhoneQ, address(*CityQ*, \_, \_, \_))*  *depositor(*"Kulikov"*, BankQ, \_, \_)* |
| 41 | person("Kulikov", PhoneQ, address(CityQ, \_, \_, \_))=  person("Kulikov", "89272030284", adress("Saransk", "Kovalenko", 7, 89)).  Унификация успешна  Θ = {"Audi" = Brand, "black" = Color, SurnameQ = "Kulikov", CityQ = “Saransk”, PhoneQ = “89272030284”, BankQ = Bank} | **Новое состояние резольвенты:**  *depositor(*"Kulikov"*, BankQ, \_, \_)* |
| 42 | depositor("Kulikov", BankQ, \_, \_) = person("Kulikov", "89271939173", adress("Saransk", "Kovalenko", 7, 89)).  Разные функторы  Унификация не успешна | Откат, переход к следующему предложению. |
| ... | ... | ... |
| 46 | depositor("Kulikov", BankQ, \_, \_) = car("Kulikov", "Audi", "white", 4500000).  Разные функторы  Унификация не успешна | Откат, переход к следующему предложению. |
| … | … | … |
| 52 | depositor("Kulikov", BankQ, \_, \_) *=* depositor("Kulikov", "Sberbank", "11111", 1500).  Унификация успешна  Θ = {"Audi" = Brand, "black" = Color, SurnameQ = "Kulikov", CityQ = “Saransk”, PhoneQ = “89272030284”, BankQ = “Sberbank”} | **Вывод:**  SurnameQ = "Kulikov", CityQ = “Saransk”, PhoneQ = “89272030284”, BankQ = “Sberbank”  Откат, переход к следующему предложению. |
| 53 | depositor("Kulikov", BankQ, \_, \_) *=* depositor("Kulikov", "Tinkoff", "22222", 3500).  Унификация успешна  Θ = {"Audi" = Brand, "black" = Color, SurnameQ = "Kulikov", CityQ = “Saransk”, PhoneQ = “89272030284”, BankQ = “Tinkoff”} | **Вывод:**  SurnameQ = "Kulikov", CityQ = “Saransk”, PhoneQ = “89272030284”, BankQ = “Tinkoff”  Откат, переход к следующему предложению. |
| 54 | depositor("Kulikov", BankQ, \_, \_) *=* depositor("Garasev", "Tinkoff", "22822", 7500).  Различаются константы surname  Унификация не успешна | Откат, переход к следующему предложению. |
| ... | ... | ... |
| 55 | depositor("Kulikov", BankQ, \_, \_) = infoByBrandAndColor(Brand, Color, Surname, City, Phone, Bank)  Разные функторы  Унификация не успешна | **Новое состояние резольвенты:**  *person(*"Kulikov"*, PhoneQ, address(*CityQ*, \_, \_, \_))*  *depositor(*"Kulikov"*, BankQ, \_, \_)* |
| 56 | person("Kulikov", PhoneQ, address(CityQ, \_, \_, \_))=  person("Garasev", "89273141395", adress("Moscow", "Kapotnoua", 19, 110)).  Различаются константы surname  Унификация не успешна | Откат, переход к следующему предложению. |
| … | … | … |
| 60 | person("Kulikov", PhoneQ, address(CityQ, \_, \_, \_))=  = car("Kulikov", "Audi", "white", 4500000).  Разные функторы  Унификация не успешна | Откат, переход к следующему предложению. |
| ... | ... | ... |
| 66 | person("Kulikov", PhoneQ, address(CityQ, \_, \_, \_))=  = depositor("Kulikov", "Sberbank", "11111", 1500).  Разные функторы  Унификация не успешна | Откат, переход к следующему предложению. |
| … | … | … |
| 70 | person("Kulikov", PhoneQ, address(CityQ, \_, \_, \_))=  = infoByBrandAndColor(Brand, Color, Surname, City, Phone, Bank)  Разные функторы  Унификация не успешна | **Текущее состояние резольвенты:**  *car(SurnameQ, "Audi","black", \_,)*  *person(SurnameQ, PhoneQ, address(CityQ, \_, \_, \_))*  *depositor(SurnameQ, BankQ, \_, \_)* |
| 71 | car(SurnameQ, "Audi","black", \_, CityQ) = car("Garasev", "BMW X6", "pink", 7500000).    Различаются константы color  Унификация не успешна | Откат, переход к следующему предложению. |
| 72 | car(SurnameQ, "Audi",”black", \_, CityQ) = car("Garasev", "Audi", "black", 1500000).    Унификация успешна  Θ = {"Audi" = Brand, "black" = Color, SurnameQ = "Garasev", CityQ = “Saransk”, PhoneQ = Phone, BankQ = Bank} | **Новое состояние резольвенты:**  *person(*"Garasev"*, PhoneQ, address(*CityQ*, \_, \_, \_))*  *depositor(*"Garasev"*, BankQ, \_, \_)* |
| 73 | *person(*"Garasev"*, PhoneQ, address(*CityQ*, \_, \_, \_))*  = person("Kulikov", "89271939173", adress("Saransk", "Kovalenko", 7, 89)).  Различаются константы surname  Унификация не успешна | Откат, переход к следующему предложению. |
| … | *…* | … |
| 75 | *person(*"Garasev"*, PhoneQ, address(*CityQ*, \_, \_, \_))*  = person("Garasev", "89273141395", adress("Moscow", "Kapotnoua", 19, 110))  Унификация успешна  Θ = {"Audi" = Brand, "black" = Color, SurnameQ = "Garasev", CityQ = “Moscow”, PhoneQ = "89273141395", BankQ = Bank} | **Новое состояние резольвенты:**  *depositor(*"Garasev"*, BankQ, \_, \_)* |
| 76 | depositor("Garasev", BankQ, \_, \_) = person("Kulikov", "89271939173", adress("Saransk", "Kovalenko", 7, 89)).  Разные функторы  Унификация не успешна | Откат, переход к следующему предложению. |
| ... | ... | ... |
| 81 | depositor("Garasev", BankQ, \_, \_) = car("Kulikov", "Audi", "white", 4500000).  Разные функторы  Унификация не успешна | Откат, переход к следующему предложению. |
| … | … | … |
| 87 | depositor("Garasev", BankQ, \_, \_) *=* depositor("Kulikov", "Sberbank", "11111", 1500).  Различаются константы surname  Унификация не успешна | Откат, переход к следующему предложению. |
| … | … | … |
| 89 | depositor("Garasev", BankQ, \_, \_) *=* depositor("Garasev", "Tinkoff", "22822", 7500).  Унификация успешна  Θ = {"Audi" = Brand, "black" = Color, SurnameQ = "Garasev", CityQ = “Moscow”, PhoneQ = "89273141395", BankQ = “Tinkoff”} | **Вывод:**  SurnameQ = "Garasev", CityQ = “Moscow”, PhoneQ = "89273141395", BankQ = “Tinkoff”  Откат, переход к следующему предложению. |
| 90 | depositor("Garasev", BankQ, \_, \_) *=* depositor("Chalyy", "SelhozBank", "33333", 500).  Различаются константы surname  Унификация не успешна | Откат, переход к следующему предложению. |
| 91 | depositor("Garasev", BankQ, \_, \_) = infoByBrandAndColor(Brand, Color, Surname, City, Phone, Bank)  Разные функторы  Унификация не успешна | **Новое состояние резольвенты:**  *person(*"Garasev"*, PhoneQ, address(*CityQ*, \_, \_, \_))*  *depositor(*"Garasev"*, BankQ, \_, \_)* |
| 92 | *person(*"Garasev"*, PhoneQ, address(*CityQ*, \_, \_, \_))*  = "Chalyy", "892722822832", adress("Anadyr", "Morshovaya", 18, 322)  Унификация успешна  Θ = {"Audi" = Brand, "black" = Color, SurnameQ = "Garasev", CityQ = “Moscow”, PhoneQ = "892722822832", BankQ = Bank} | **Новое состояние резольвенты:**  *depositor(*"Garasev"*, BankQ, \_, \_)* |
| 93 | depositor("Garasev", BankQ, \_, \_) = person("Kulikov", "89271939173", adress("Saransk", "Kovalenko", 7, 89)).  Разные функторы  Унификация не успешна | Откат, переход к следующему предложению. |
| ... | ... | ... |
| 99 | depositor("Garasev", BankQ, \_, \_) = car("Kulikov", "Audi", "white", 4500000).  Разные функторы  Унификация не успешна | Откат, переход к следующему предложению. |
| … | … | … |
| 105 | depositor("Garasev", BankQ, \_, \_) *=* depositor("Kulikov", "Sberbank", "11111", 1500).  Различаются константы surname  Унификация не успешна | Откат, переход к следующему предложению. |
| … | … | … |
| 107 | depositor("Garasev", BankQ, \_, \_) *=* depositor("Garasev", "Tinkoff", "22822", 7500).  Унификация успешна  Θ = {"Audi" = Brand, "black" = Color, SurnameQ = "Garasev", CityQ = “Moscow”, PhoneQ = "892722822832", BankQ = “Tinkoff”} | **Вывод:**  SurnameQ = "Garasev", CityQ = “Moscow”, PhoneQ = "892722822832", BankQ = “Tinkoff”  Откат, переход к следующему предложению. |
| 90 | depositor("Garasev", BankQ, \_, \_) *=* depositor("Chalyy", "SelhozBank", "33333", 500).  Различаются константы surname  Унификация не успешна | Откат, переход к следующему предложению. |
| 91 | depositor("Garasev", BankQ, \_, \_) = infoByBrandAndColor(Brand, Color, Surname, City, Phone, Bank)  Разные функторы  Унификация не успешна | **Новое состояние резольвенты:**  **Резольвента пуста.** |

**Ни одного владельца**

*Вопрос* infoByBrandAndColor *("Lada Vesta", "red", SurnameQ, CityQ, Phone\_numQ, BankQ).*

|  |  |  |
| --- | --- | --- |
| **Шаг** | **Сравнимые термы; результат, подстановка (если есть)** | **Действие** |
| 0 | infoByBrandAndColor(“Lada Vesta", "red", SurnameQ, CityQ, Phone\_numQ, BankQ) = person("Kulikov", "89271939173", adress("Saransk", "Kovalenko", 7, 89))  Разные функторы  Унификация не успешна | **Текущее состояние резольвенты:**  infoByBrandAndColor *("Lada Vesta", "red", SurnameQ, CityQ, Phone\_numQ, BankQ)*  Откат, переход к следующему предложению. |
| ... | ... | ... |
| 6 | infoByBrandAndColor (“Lada Vesta", "red", SurnameQ, CityQ, Phone\_numQ, BankQ) = car("Kulikov", "Audi", "white", 4500000)  Разные функторы  Унификация не успешна | Откат, переход к следующему предложению. |
| ... | ... | ... |
| 12 | infoByBrandAndColor (“Lada Vesta", "red", SurnameQ, CityQ, Phone\_numQ, BankQ) = depositor("Kulikov", "Sberbank", "11111", 1500")  Разные функторы  Унификация не успешна | Откат, переход к следующему предложению. |
| ... | ... | ... |
| 16 | infoByBrandAndColor (“Lada Vesta", "red", SurnameQ, CityQ, Phone\_numQ, BankQ) = infoByBrandAndColor (Brand, Color, SurnameQ, CityQ, PhoneQ, BankQ)  Унификация успешна  Θ = {"Lada Vesta" = Brand, "red" = Color, SurnameQ = Surname, CityQ = City, Phone\_numQ = Phone\_num,  BankQ = Bank} | **Смена состояния резольвенты**  **Новое состояние резольвенты:**  *car(SurnameQ,* "Lada Vesta"*,"red", \_)*  *person(SurnameQ, PhoneQ, address(CityQ, \_, \_, \_))*  *depositor(SurnameQ, BankQ, \_, \_)* |
| 17 | car(SurnameQ, "Lada Vesta","red", \_, CityQ) = person("Kulikov", "89271939173", adress("Saransk", "Kovalenko", 7, 89))  Разные функторы  Унификация не успешна | Откат, переход к следующему предложению. |
| ... | ... | ... |
| 23 | car(SurnameQ, "Lada Vesta", "red", \_) = car("Kulikov", "Audi", "white", 4500000)    Различаются константы типа brand  Унификация не успешна | Откат, переход к следующему предложению. |
| ... | ... | ... |
| 27 | car(SurnameQ, "Lada Vesta", "red", \_) = car("Chalyy", "Lada Vesta", "white", 500000)    Различаются константы типа color  Унификация не успешна | Откат, переход к следующему предложению. |
| 28 | car(SurnameQ, "Lada Vesta", "red", \_) = depositor("Kulikov", "Sberbank", "11111", 1500).  Разные функторы  Унификация не успешна | Откат, переход к следующему предложению. |
| ... | ... | ... |
| 32 | car(SurnameQ, "Lada Vesta", "red", \_) = infoByBrandAndColor(Brand, Color, Surname, City, Phone, Bank)  Разные функторы  Унификация не успешна | Так как цель car(SurnameQ, “Lada Vesta","red", \_, CityQ) не доказана, а все знания из базы знаний проверены, система завершает работу и  **Вывод:**  **No solution** |

**При рассмотрении таблицы, в которой изменен порядок следования БЗ можно заметить, что объем работ одинаков. Порядок следования не важен. Обход осуществляется по всем предложениям.**

**Ответы на теоретические вопросы**

**1. В какой части правила сформулировано знание? Это знание о чем, с формальной точки зрения?**

Знание сформулировано в заголовке правила как составной терм f(t1, …, tn). Это знание о том, что между аргументами (t1, tn) существует взаимосвязь, и эта взаимосвязь имеет имя f.

**2. Что такое процедура?**

Процедура - совокупность правил, заголовки которых имеют одно и то же имя и одну и ту же арность(местность), т. е. это совокупность правил, описывающих одно и то же отношение. Отношение, определяемое процедурой, называется предикатом.

**3. Сколько в БЗ текущего задания процедур?**

4 процедуры.

**4. Что такое пример терма, это частный случай терма, пример? Как строится пример?**

Пример терма А — это терм B такой, что существует такая подстановка θ, что B = Aθ. Пример терма — это частный случай терма, так как по определению является термом. Пример строится с помощью применения подстановки θ к терму А. Примеры термов строятся в процессе унификации, когда происходит сравнение двух термов, с помощью подстановки всех возможных значений из базы знаний.

**5. Что такое наиболее общий пример?**

Терм С является общим примером термов А и В, если существуют подстановки θ1 и θ2, такие, что С = А θ1 и С = В θ2. Терм S называется более общим чем терм Т, если T является примером S, а S не является примером T. Наиболее общий пример термов Т1 и Т2 — это такой их общий пример, который является более общим по отношению к любому другому из примеру.

**6. Назначение и результат работы алгоритма унификации. Что значит двунаправленная передача параметров при работе алгоритма унификации, поясните на примере одного из случаев пункта 3.**

Унификация — это алгоритм, который позволяет формализовать процесс доказательства. В процессе работы система проводит большое количество унификаций, чтобы проверить условия и прийти к логическому выводу, ответить на вопрос «Да» или «Нет». Результатом работы алгоритма унификации является вывод о сопоставимости термов, а также конкретизация переменных.

Двунаправленная передача параметров заключается в следующем: значения параметров могут передаваться из «внешнего мира» в программу для дальнейшего использования и из программы во «внешний мир».

**7. В каком случае запускается механизм отката?**

В процессе выполнения программы - система, используя встроенный алгоритм унификации, пытается найти знание и обосновать, с его помощью, возможность истинности вопроса, строя подстановки и примеры термов, используя базу знаний, найти такие значения переменных, при которых это удается, а значит, на поставленный вопрос можно дать ответ "Да". Возможно система "ошибается" в своих обоснованиях и возникает тупиковая ситуация, или, ответив на вопрос, пытается найти другой способ доказательства. Тогда включается механизм отката(отказа от последнего заключения и последних действий, сделанных системой) и выполняется ре- конкретизация переменных, конкретизация которых была выполнена на последнем шаге.

**8. Виды и назначение переменных в Prolog. Примеры из задания. Почему использованы те или другие переменные (примеры из задания)?**

Переменная:

* Именованная – обозначается комбинацией символов латинского алфавита, цифр и символа подчеркивания, начинающейся с прописной буквы или символа подчеркивания (X, A21, \_X)
* Анонимная - обозначается символом подчеркивания (\_)

Переменные предназначены для передачи значений «во времени и в пространстве». Переменные в факты и правила входят только с квантором всеобщности. А в вопросы переменные входят только с квантором существования. Именованные переменные уникальны в рамках одного предложения, анонимная переменная - любая уникальна. В разных предложениях может использоваться одно имя переменной для обозначения разных объектов. В процессе выполнения программы переменные могут связываться с различными объектами - конкретизироваться. Это относится только к именованным переменным. Анонимные переменные не могут быть связаны со значением.

Переменная конкретизирована, если имеется объект, который обозначает эта переменная. Переменная не конкретизирована, если еще не известно, что именно она обозначает. Конкретизация не связана с распределением памяти и присваиванием переменной значения. В логическом программировании все переменные рассматриваются как без- типовые, то есть в процессе вычисления любая переменная может быть связана с объектом произвольной природы.

**Пример из задания**

info\_by\_brand\_and\_color(Brand, Color, Surname, City, Phone, Bank) :- car(Surname, Brand, Color, \_), person(Surname, Phone, address(City, \_, \_, )), depositor(Surname, Bank, \_, \_).

Переменные Surname, Brand, Color, City, Phone, Bank - именованные переменные. \_ - анонимные переменные, они использованы в данном случае, чтобы термы были одной арности, однако их значение нам не важно. Значения именованных переменных важны.