|  |  |
| --- | --- |
| 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 «Программное обеспечение ЭВМ и информационные технологии»

**ОТЧЕТ**

*к лабораторной работе №14*

*По курсу: «Функциональное и логическое программирование»*

Студентка ИУ7-65Б

Оберган Т.М.

Преподаватели

Толпинская Н.Б.

Строганов Ю.В.

*Москва, 2020 г.*

**Задание:**

Составить 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). В разных городах есть однофамильцы, в одном городе – фамилия уникальна. Владельцев может быть несколько (не более 3-х), один и ни одного.

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!!!

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.

**Вопросы:**

**В какой части правила сформулировано знание?**В заголовке находится знание о предметной области, а в теле содержится условия истинности этого знания.

**Что такое процедура?**Процедурой называется совокупность правил, заголовки которых имеют одно и то же имя и одну и ту же арность, описывают одно определенное отношение.

**Сколько в БЗ текущего задания процедур?**

4

**Что такое пример терма, это частный случай терма, пример?**Терм В называется примером терма А, если существует такая подстановка Θ, что В=АΘ, где АΘ – результат применения подстановки к терму.   
**Как и когда строится?**

Примеры строятся во время алгоритма унификации.

**Что такое общий пример?**Терм С называется общим примером термов А и В, если существуют такие подстановки Θ1 и Θ2, что С = А Θ1 и С=В Θ2.

**Назначение и результат работы алгоритма унификации.**

Унификация – механизм логического вывода. Результат – подстановка.

**Что значит двунаправленная передача параметров при работе алгоритма унификации, поясните на примере одного из случаев пункта 3.**car\_owner(Model, Color, Lname, City, Phone, Bank) :-

car(Lname, Model, Color, \_),

phone\_book(Lname, Phone, address(City, \_, \_, \_)),

investors(Lname, Bank, \_, \_).  
Видимо имеется ввиду, что если в car\_owner задать только Phone, то Lname будет найдено в phone\_book и использовано в car и investors.

**В каком случае запускается механизм отката?**

В случае тупиковой ситуации при попытке сопоставить два терма.

**Виды и назначение переменных в Prolog. Примеры из задания.**

* Именованные – есть имя – комбинация символов (Color).
* Анонимные – нет имени – символ подчеркивания ( \_ ).

**Почему использованы те или другие переменные (примеры из задания)?**

Анонимные переменные используются, если не нужно связывать ее со значением (не нужно заносить в результирующую ячейку, например).

**Текст программы:**

domains

lname, phone, city, street = symbol.

building, apartment = integer.

model, color, bank, account = symbol.

price, amount = integer.

address = address(city, street, building, apartment).

predicates

phone\_book(lname, phone, address).

car(lname, model, color, price).

investors(lname, bank, account, amount).

car\_owner(model, color, lname, city, phone, bank).

clauses

phone\_book("Winter", "+123", address("Moscow", "Postal", 1, 2)).

phone\_book("Winter", "+000", address("SPB", "Neva", 2, 4)).

phone\_book("Summer", "+345", address("Sevastopol", "Anotherstrt", 4, 8)).

phone\_book("Autumn", "+333", address("Moscow", "Postal", 8, 16)).

car("Winter", "IDK", red, 200000).

car("Winter", "SmallOne", black, 150).

car("Summer", "Apple", green, 1000000).

investors("Winter", "RNCB", private, 999999999).

investors("Autumn", "Sberbank", public, 10).

investors("Summer", "Raiffeisen", public, 10000).

car\_owner(Model, Color, Lname, City, Phone, Bank) :-

car(Lname, Model, Color, \_),

phone\_book(Lname, Phone, address(City, \_, \_, \_)),

investors(Lname, Bank, \_, \_).

goal

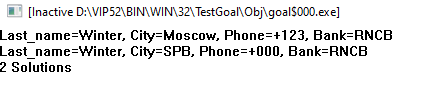
%car\_owner("SmallOne", black, Last\_name, City, Phone, Bank). % some

%car\_owner("Apple", green, Last\_name, City, Phone, Bank). % one

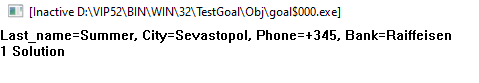
%car\_owner("SmallOne", red, Last\_name, City, Phone, Bank). % none

**Результаты работы программы:**

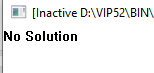
some:



one:



none:



**Задание №1** порядок поиска ответа на вопрос**:**

Некоторые нижние подчеркивания в таблице почему-то заменились на пробел.

**Для нескольких владельцев:**

Вопрос: car\_owner("SmallOne", black, Last\_name, City, Phone, Bank)

|  |  |  |
| --- | --- | --- |
| № шага | Сравниваемые термы; результат; подстановка, если есть | Дальнейшие действия: прямой ход или откат (к чему приводит?) |
| 1 | car\_owner(Model, Color, Lname, City, Phone, Bank) и  car\_owner("SmallOne", black, Last\_name, City, Phone, Bank) сравнимы.  Связываются: Model =”SmallOne” Color=black | car\_owner убирается из стека.  Помещаем в стек  car(Lname, Model, Color, \_) и phone\_book(Lname, Phone address(City, \_, \_, \_)) и investors(Lname, Bank, \_, \_) |
| 2 | car(Lname, “SmallOne”, black, \_) и car("Winter", "IDK", red, 200000) не унифицируемы. |  |
| 3 | car(Lname, “SmallOne”, black, \_) и car("Winter", "SmallOne", black, 150) унифицируемы  Lname = “Winter” | car убирается из стека |
| 4 | phone\_book(“Winter”, Phone address(City, \_, \_, \_)) и phone\_book("Winter", "+123", address("Moscow", "Postal", 1, 2)) унифицируемы  Phone = ”+123” City = “Moscow” | phone\_book убирается из стека |
| 5 | investors(“Winter”, Bank, \_, \_) и investors("Winter", "RNCB", private, 999999999) унифицируемы  Bank = “RNCB” | investors убирается из стека  стек пуст  **Выводится** Lname, City, Phone, Bank: “Winter”, “Moscow”, “+123”, “RNCB”  Откат. В стек возвращается investors. Bank развязывается. |
| 6 | investors(“Winter”, Bank, \_, \_) и investors("Autumn", "Sberbank", public, 10) не унифицируемы |  |
| 7 | investors(“Winter”, Bank, \_, \_) и investors("Summer", "Raiffeisen", public, 10000) не унифицируемы | Дошли до конца investors  Откат. В стек возвращается phone\_book. Phone, City развязываются. |
| 8 | phone\_book(“Winter”, Phone address(City, \_, \_, \_)) и phone\_book("Winter", "+000", address("SPB", "Neva", 2, 4)) унифицируемы  Phone = “+000” City = “SPB” | phone\_book убирается из стека |
| 9, 10, 11 | Аналогично 5, 6, 7 | **Выводится** Lname, City, Phone, Bank: “Winter”, “SPB”, “+000”, “RNCB” |
| 12, 13 | phone\_book(“Winter”, Phone address(City, \_, \_, \_)) не унифицируется с оставшимися phone\_book | Дошли до конца phone\_book.  Откат. В стек возвращается car |
| 14 | car(Lname, “SmallOne”, black, \_) и car("Summer", "Apple", green, 1000000) не унифицируемы. | Дошли до конца car.  Откат.  В стек возвращается car\_owner |
| 15 |  | Дошли до конца car\_owner, программа завершается |

Я устала расписывать таблички, надеюсь подойдет такое словесное описание..:  
**Для одного владельца:**

Вопрос: car\_owner("Apple", green, Last\_name, City, Phone, Bank)

Точно также зайдем в car\_owner, потом

в car будем искать зеленую Apple, пройдемся по всем, найдя подходящий последний факт car  
car("Summer", "Apple", green, 1000000).

в phone\_book будем искать людей с фамилией “Summer”, найдем на 3 сравнении, свяжем телефон и город  
phone\_book("Summer", "+345", address("Sevastopol", "Anotherstrt", 4, 8))

ищем investors “Summer”, пройдя по всем, найдем подходящий последний  
investors("Summer", "Raiffeisen", public, 10000)  
**выводится** Summer, Sevastopol,“+345”, “Raiffeisen”  
investors закончились

досматриваем оставшиеся phone\_book, не находим

phone\_book закончились

car закончились

car\_owner закончились  
Программа завершилась.

**Ни одного владельца:**

Вопрос: car\_owner("SmallOne", red, Last\_name, City, Phone, Bank)

Находим car\_owner  
car\_owner(Model, Color, Lname, City, Phone, Bank); Model=“SmallOne”, Color=red

Ищем унифицируемые с car(Lname, “SmallOne”, red, \_)  
 Прошлись по всем car, не нашли унифицируемых, откатываемся.

car\_owner закончились.

Программа завершилась, выведя **NoSolution** т.к. не было найдено решения.

**Задание №2** порядок поиска ответа на вопрос при разных порядках следования**:**

Если считать, что система знает где в базе знаний начинается область тех или иных процедур и не делает лишних проверок, сравнивая функторы и арность с самого начала базы знаний, то, если поменять расположение процедур местами, количество сравнений не изменится.

Если же считать, что система полностью проходит базу знаний от начала до конца, то количество сравнений также не изменится. Т.к. для каждого вопроса будет попытка унификации с каждым элементом базы знаний.

Количество сравнений изменится только лишь в том случае, если считать, что система не знает где начало процедуры и дальше конца области этой процедуры не пойдет. В таком случае, расположение процедуры, с которой требуется наибольшее число попыток унификации, в начале БЗ, уменьшит количество шагов.

Количество сравнений может изменится от порядка занесения в стек (см. Рис. 1, 2).

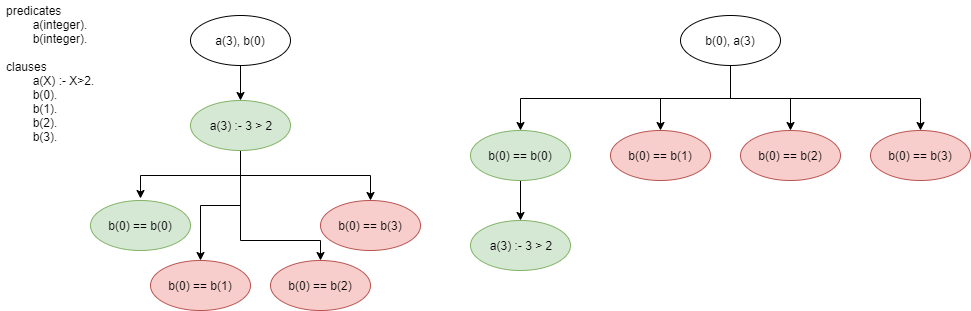


Рис. 1 – порядок не повлиял на количество сравнений

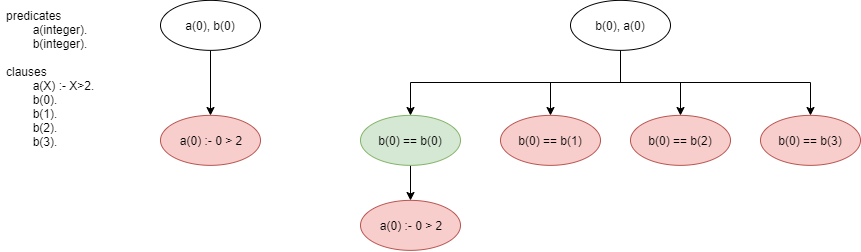


Рис. 2 – порядок повлиял на количество сравнений

**Задание №3** таблицы работы алгоритма унификации:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| шаг унифи ка ции | результирующая ячейка | рабочее поле | пункт алгоритма | стек |
| 0 |  |  | 1. | car\_owner("SmallOne", black, Last\_name, City, Phone, Bank) =  car\_owner(Model, Color, Lname, City, Phone, Bank) |
| 1 |  | car\_owner("SmallOne", black, Last\_name, City, Phone, Bank) =  car\_owner(Model, Color, Lname, City, Phone, Bank) :- car(Lname, Model, Color, \_), phone\_book(Lname, Phone, address(City, \_, \_, \_)), investors(Lname, Bank, \_, \_).  -----🡪 | е | Model = "SmallOne" Color = black  car(Last\_name, Model, Color, \_) phone\_book(Last\_name, Phone, address(City, \_, \_, \_)) investors(Last\_name, Bank, \_, \_). |
| 2 | Model = "SmallOne" | Model = "SmallOne"  🡨----- | г | Color = black  car(Last\_name, "SmallOne",Color, \_) phone\_book(Last\_name, Phone, address(City, \_, \_, \_)) investors(Last\_name, Bank, \_, \_). |
| 3 | Model = "SmallOne",  Color = black | Color = black 🡨----- | г | car(Last\_name, "SmallOne", black, \_) phone\_book(Last\_name, Phone, address(City, \_, \_, \_)) investors(Last\_name, Bank, \_, \_). |
| 4 | Model = "SmallOne",  Color = black | Пытаемся запустить алг. унификации для car("Winter", "IDK", red, 200000) = car(Last\_name, "SmallOne", black, \_)  В итоге неудачно. |  | phone\_book(Last\_name, Phone, address(City, \_, \_, \_)) investors(Last\_name, Bank, \_, \_). |
| 5 | Model = "SmallOne",  Color = black | Пытаемся запустить алг. унификации для car("Winter", "SmallOne", black, 150) = car(Last\_name, "SmallOne", black, \_)  -----🡪 |  | car("Winter", "SmallOne", black, 150) = car(Last\_name, "SmallOne", black, \_) phone\_book(Last\_name, Phone, address(City, \_, \_, \_)) investors(Last\_name, Bank, \_, \_). |
| 6 | Model = "SmallOne",  Color = black | car("Winter", "SmallOne", black, 150) = car(Last\_name, "SmallOne", black, \_) -----🡪 | е | Last\_name = “Winter” "SmallOne" = "SmallOne" black = black  phone\_book(Last\_name, Phone, address(City, \_, \_, \_)) investors(Last\_name, Bank, \_, \_). |
| 7 | Model = "SmallOne",  Color = black Last\_name = “Winter” | Last\_name = “Winter”  🡨----- | г | "SmallOne" = "SmallOne" black = black  phone\_book(“Winter”, Phone, address(City, \_, \_, \_)) investors(“Winter”, Bank, \_, \_). |
| 8 | Model = "SmallOne",  Color = black Last\_name = “Winter” | "SmallOne" = "SmallOne" | б | black = black  phone\_book(“Winter”, Phone, address(City, \_, \_, \_)) investors(“Winter”, Bank, \_, \_). |
| 9 | Model = "SmallOne",  Color = black Last\_name = “Winter” | black = black | б | phone\_book(“Winter”, Phone, address(City, \_, \_, \_)) investors(“Winter”, Bank, \_, \_). |
| 10 | Model = "SmallOne",  Color = black Last\_name = “Winter” | Пытаемся запустить алг. унификации для phone\_book(“Winter”, Phone, address(City, \_, \_, \_)) = phone\_book("Winter", "+123", address("Moscow", "Postal", 1, 2)) -----🡪 |  | phone\_book(“Winter”, Phone, address(City, \_, \_, \_)) = phone\_book("Winter", "+123", address("Moscow", "Postal", 1, 2)) investors(“Winter”, Bank, \_, \_). |
| 11 | Model = "SmallOne",  Color = black Last\_name = “Winter” | phone\_book(“Winter”, Phone, address(City, \_, \_, \_)) = phone\_book("Winter", "+123", address("Moscow", "Postal", 1, 2)) -----🡪 | е | “Winter” = “Winter” Phone = "+123" City = “Moscow”  investors(“Winter”, Bank, \_, \_). |
| 12 | Model = "SmallOne",  Color = black Last\_name = “Winter” | “Winter” = “Winter” | б | Phone = "+123" City = “Moscow”  investors(“Winter”, Bank, \_, \_). |
| 13 | Model = "SmallOne",  Color = black Last\_name = “Winter” Phone = "+123" | Phone = "+123" 🡨----- | г | City = “Moscow”  investors(“Winter”, Bank, \_, \_). |
| 14 | Model = "SmallOne",  Color = black Last\_name = “Winter” Phone = "+123 City = “Moscow” | City = “Moscow” 🡨----- | г | investors(“Winter”, Bank, \_, \_). |
| 15 | -||- | Пытаемся запустить алг. унификации для investors(“Winter”, Bank, \_, \_) = investors("Winter", "RNCB", private, 999999999) -----🡪 |  | investors(“Winter”, Bank, \_, \_) = investors("Winter", "RNCB", private, 999999999) |
| 16 | -||- | investors(“Winter”, Bank, \_, \_) = investors("Winter", "RNCB", private, 999999999) -----🡪 | е | “Winter” = “Winter” Bank = “RNCB” |
| 17 | -||- | “Winter” = “Winter” | б | Bank = “RNCB” |
| 18 | Model = "SmallOne",  Color = black Last\_name = “Winter” Phone = "+123 City = “Moscow” Bank = “RNCB” | Bank = “RNCB” 🡨----- | г |  |
| 19 |  | Не прошло и года как стек опустел. В рез. ячейке подстановка. Ну что, готовы? Откатываемся. (к 15) |  |  |
| 20 | Model = "SmallOne",  Color = black Last\_name = “Winter” Phone = "+123 City = “Moscow” | Пытаемся запустить алг. унификации для investors(“Winter”, Bank, \_, \_) = investors("Autumn", "Sberbank", public, 10).  В итоге неудачно. |  |  |
| 21 |  | Пытаемся запустить алг. унификации для investors(“Winter”, Bank, \_, \_) = investors("Summer", "Raiffeisen", public, 10000).  В итоге неудачно. |  |  |
|  |  | investorы закончились. Откатываемся к 10. |  |  |
| 22 | Model = "SmallOne",  Color = black Last\_name = “Winter” | Пытаемся запустить алг. унификации для phone\_book(“Winter”, Phone, address(City, \_, \_, \_)) = phone\_book("Winter", "+000", address("SPB", "Neva", 2, 4)) -----🡪 |  | phone\_book(“Winter”, Phone, address(City, \_, \_, \_)) = phone\_book("Winter", "+000", address("SPB", "Neva", 2, 4)) investors(“Winter”, Bank, \_, \_). |
| 23-32 |  | Тут все аналогично предыдущыму  (шаги 11-20) случаю |  |  |
| 33 | Model = "SmallOne",  Color = black Last\_name = “Winter” Phone = "+000 City = “SPB” Bank = “RNCB” | Стек опустел. В рез. ячейке подстановка. |  |  |
| 34 | Model = "SmallOne",  Color = black Last\_name = “Winter | Откат к 22. Пытаемся запустить алг. унификации для phone\_book(“Winter”, Phone, address(City, \_, \_, \_)) = phone\_book("Summer", "+345", address("Sevastopol", "Anotherstrt", 4, 8)) в итоге неудачно. |  | investors(“Winter”, Bank, \_, \_). |
| 35 | Model = "SmallOne",  Color = black Last\_name = “Winter | Пытаемся запустить алг. унификации для phone\_book(“Winter”, Phone, address(City, \_, \_, \_)) = phone\_book("Autumn", "+333", address("Moscow", "Postal", 8, 16)). в итоге неудачно. |  | investors(“Winter”, Bank, \_, \_). |
| 36 | Model = "SmallOne",  Color = black | Откат к 5. Пытаемся запустить алг. унификации для car("Winter", "SmallOne", black, 150) = car("Summer", "Apple", green, 1000000). в итоге неудачно. |  | address(City, \_, \_, \_)) investors(Last\_name, Bank, \_, \_). |

Спасибо за внимание.