



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»

## Отчёт по лабораторной работе №11-12 по курсу

### «Функциональное и логическое программирование»

Тема Структурные программы на Prolog. Работа программы на Prolog.

Студент Сироткина П.Ю.

Группа ИУ7-66Б

Преподаватели Толпинская Н.Б., Строганов Ю.В.

Москва — 2022 г.

# Лабораторная работа №11

## Часть 1

Разработать свою программу - «Телефонный справочник». Протестировать работу программы.

```
1 domains
2     name, phone_number = symbol.
3
4 predicates
5     phone(name, phone_number)
6
7 clauses
8     phone("Polina", "111").
9     phone("Polina", "222").
10    phone("Anna", "333").
11    phone("Ivan", "444").
12    phone("Petr", "555").
13    phone("Petr", "666").
14
15 goal
16     %phone(Name, "111").
17     % Name = Polina
18     % 1 Solution
19
20     %phone("Polina", Phone_number).
21     % Phone_number = 111
22     % Phone_number = 222
23     % 2 Solutions
24
25     phone("Ivan", Phone_number).
26     % Phone_number = 444
27     % 1 Solution
```

## Часть 2

Составить программу – базу знаний, с помощью которой можно определить, например, множество студентов, обучающихся в одном ВУЗе и их телефоны. Студент может одновременно обучаться в нескольких ВУЗах. Привести примеры возможных вариантов вопросов и варианты ответов (не менее 3-х). Описать порядок формирования вариантов ответа.

Исходную базу знаний сформировать с помощью только фактов.

\*Исходную базу знаний сформировать, используя правила.

\*\*Разработать свою базу знаний (содержание произвольно).

(См. след. лист)

```

1 domains
2     student_id, university_id = integer.
3     name, phone_number, university_name = symbol.
4
5 predicates
6     student_info(student_id, name, phone_number)
7     university_info(university_id, university_name)
8     study(student_id, university_id)
9     students_from(university_id, name, phone_number)
10
11 clauses
12     student_info(0, "Polina", "111").
13     student_info(0, "Polina", "112").
14     student_info(1, "Anna", "222").
15     student_info(2, "Ivan", "333").
16
17     university_info(0, "BMSTU").
18     university_info(1, "MSU").
19     university_info(2, "HSE").
20
21     study(0, 1).
22     study(1, 0).
23     study(1, 1).
24     study(2, 2).
25
26     students_from(University_id, Name, Phone_number) :- student_info(
27         Student_id, Name, Phone_number), study(Student_id, University_id).
28
29 goal
30     students_from(1, Name, Phone_number).
31     % Name = Polina, Phone_number = 111
32     % Name = Polina, Phone_number = 112
33     % Name = Anna, Phone_number = 222
34     % 3 Solution
35
36     %students_from(University_id, Name, "333").
37     % University_id = 2, Name = Ivan
38     % 1 Solution
39
40     %students_from(University_id, "Anna", Phone_number).
41     % University_id = 0, Phone_number = 222
42     % University_id = 1, Phone_number = 222
43     % 2 Solutions

```

# Лабораторная работа №12

## Часть 1

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

- «Телефонный справочник»: Фамилия, №тел, Адрес – структура (Го- род, 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 (Фак- ты).

Используя правила, обеспечить возможность поиска:

1.
  - По № телефона найти: Фамилию, Марку автомобиля, Стоимость автомобиля (мо- жет быть несколько);
  - Используя сформированное в пункте а) правило, по № телефона найти: только Марку автомобиля (автомобилей может быть несколько).
2. 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 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 унифика- ции, выписать подстановку – наибольший общий унификатор, и соответствующие примеры термов.

```
1 domains
2     surname, phone = symbol.
3     city, street = symbol.
4     house, flat = integer.
5     address = address_struct(city, street, house, flat)
6     mark, color = symbol.
7     cost = integer.
8     bank = symbol.
9     account, sum = integer.
10
11 predicates
12     has_phone(surname, phone, address)
13     has_car(surname, mark, color, cost)
14     has_deposite(surname, bank, account, sum)
15
16     info_by_phone(phone, surname, mark, cost)
17     only_mark_by_phone(phone, mark)
18     info_by_surname_and_city(surname, city, street, bank, phone)
19
20 clauses
```

```

21 has_phone("Petrov", "111", address_struct("Moscow", "Gagarina",
22     1, 1)).
23 has_phone("Petrov", "222", address_struct("Moscow", "Gagarina",
24     1, 1)).
25 has_phone("Ivanov", "333", address_struct("Vladimir", "Lenina", 2,
26     2)).
27 has_phone("Volkov", "444", address_struct("Sevastopol", "Lenina", 2,
28     2)).
29 has_phone("Sidorov", "555", address_struct("Kazan", "Lomonosova",
30     3, 3)).
31
32 has_car("Petrov", "BMW", "black", 1000).
33 has_car("Ivanov", "Audi", "black", 1500).
34 has_car("Ivanov", "Audi", "white", 1500).
35 has_car("Volkov", "Lada", "white", 500).
36
37 has_deposit("Petrov", "Alpha", 0, 5000).
38 has_deposit("Ivanov", "Beta", 1, 5500).
39 has_deposit("Volkov", "Gamma", 2, 6000).
40 has_deposit("Sidorov", "Alpha", 3, 6500).
41 has_deposit("Sidorov", "Beta", 4, 7000).
42
43 info_by_phone(Phone, Surname, Mark, Cost) :- has_phone(Surname, Phone,
44     _), has_car(Surname, Mark, _, Cost).
45 only_mark_by_phone(Phone, Mark) :- info_by_phone(Phone, _, Mark, _).
46 info_by_surname_and_city(Surname, City, Street, Bank, Phone) :-
47     has_phone(Surname, Phone, address_struct(City, Street, _, _)),
48     has_deposit(Surname, Bank, _, _).
49
50 goal
51 %info_by_phone("444", Surname, Mark, Cost).
52 % Surname = Volkov, Mark = Lada, Cost = 500
53 % 1 Solution
54
55 %only_mark_by_phone("333", Mark).
56 % Mark = Audi
57 % Mark = Audi
58 % 2 Solutions
59
60 info_by_surname_and_city("Sidorov", "Kazan", Street, Bank, Phone).
61 % Street = Lomonosova, Bank = Alpha, Phone = 555
62 % Street = Lomonosova, Bank = Beta, Phone = 555
63 % 2 Solutions

```

Шаг	Сравниваемые термины, результат, подстановка, если есть	Дальнейшие действия
1	Сравнение info_by_phone("444", Surname, Mark, Cost) и has_phone("Petrov", "111", address_struct("Moscow", "Gagarina", 1, 1)). Унификация неуспешна.	Откат. Переход к следующему предложению.
2-14	...	...
15	Сравнение info_by_phone("444", Surname, Mark, Cost) и info_by_phone(Phone, Surname, Mark, Cost). Унификация успешна.  Подстановка: {Surname = Surname, Mark = Mark, Cost = Cost}.	Переход к терму правила.  Унификация has_phone(Surname, "444", _).
16	Сравнение has_phone(Surname, "444", _) и has_phone("Petrov", "111", address_struct("Moscow", "Gagarina", 1, 1)). Унификация неуспешна.	Откат. Переход к следующему предложению.
17-18	...	...
19	Сравнение has_phone(Surname, "444", _) и has_phone("Volkov", "444", address_struct("Sevastopol", "Lenina", 2, 2)). Унификация успешна.  Подстановка: {Surname = "Volkov", Mark = Mark, Cost = Cost}.	Переход к терму следующего правила.  Унификация has_car("Volkov", Mark, _, Cost).
20	Сравнение has_car("Volkov", Mark, _, Cost) и has_phone("Petrov", "111", address_struct("Moscow", "Gagarina", 1, 1)). Унификация неуспешна.	Откат. Переход к следующему предложению.
21-27	...	...
28	Сравнение has_car("Volkov", Mark, _, Cost) и has_car("Volkov", "Lada", "white", 500). Унификация успешна.  Подстановка: {Surname = "Volkov", Mark = "Lada", Cost = 500}.	Ответ найден. Сохранение подстановки {Surname = "Volkov", Mark = "Lada", Cost = 500}. Реконкретизация Mark и Cost. Переход к следующему предложению.
29	Сравнение has_car("Volkov", Mark, _, Cost) и has_deposite("Petrov", "Alpha", 0, 5000). Унификация неуспешна.	Откат. Переход к следующему предложению.
30-35	...	...
36	Сравнение has_car("Volkov", Mark, _, Cost) и info_by_surname_and_city(Surname, City, Street, Bank, Phone). Унификация неуспешна.	Достижение конца БЗ. Реконкретизация Surname.  Переход к следующему предложению отн. шага 19.
38	Сравнение has_phone(Surname, "444", _) и has_phone("Sidorov", "555", address_struct("Kazan", "Lomonosova", 3, 3)). Унификация неуспешна.	Откат. Переход к следующему предложению.
39-54	...	...
55	Сравнение has_phone(Surname, "444", _) и info_by_surname_and_city(Surname, City, Street, Bank, Phone). Унификация неуспешна.	Достижение конца БЗ. Резольвента пуста. Завершение работы.

Шаг	Сравниваемые термины, результат, подстановка, если есть	Дальнейшие действия
1	Сравнение only_mark_by_phone("333", Mark) и has_phone("Petrov", "111", address_struct("Moscow", "Gagarina", 1, 1)). Унификация неуспешна.	Откат. Переход к следующему предложению.
2-15	...	...
16	Сравнение only_mark_by_phone("333", Mark) и only_mark_by_phone(Phone, Mark). Унификация успешна.  Подстановка: {Mark = Mark}.	Переход к терму правила.  Унификация info_by_phone("333", _, Mark, _).
17	Сравнение info_by_phone("333", _, Mark, _) и has_phone("Petrov", "111", address_struct("Moscow", "Gagarina", 1, 1)). Унификация неуспешна.	Откат. Переход к следующему предложению.
18-30	...	...
31	Сравнение info_by_phone("333", _, Mark, _) и info_by_phone(Phone, Surname, Mark, Cost). Унификация успешна.  Подстановка: {Mark = Mark}	Переход к терму правила.  Унификация has_phone(Surname, "333", _).
32	Сравнение has_phone(Surname, "333", _) и has_phone("Petrov", "111", address_struct("Moscow", "Gagarina", 1, 1)). Унификация неуспешна.	Откат. Переход к следующему предложению.
33	...	...
34	Сравнение has_phone(Surname, "333", _) и has_phone("Ivanov", "333", address_struct("Vladimir", "Lenina", 2, 2)). Унификация успешна.	Переход к терму следующего правила.  Унификация has_car("Ivanov", Mark, _, Cost).
35	Сравнение has_car("Ivanov", Mark, _, Cost) и has_phone("Petrov", "111", address_struct("Moscow", "Gagarina", 1, 1)). Унификация неуспешна.	Откат. Переход к следующему предложению.
36-40	...	...
41	Сравнение has_car("Ivanov", Mark, _, Cost) и has_car("Ivanov", "Audi", "black", 1500). Унификация успешна.  Подстановка: {Mark = "Audi"}.	Ответ найден. Сохранение подстановки {Mark = "Audi"}. Реконкретизация Mark. Переход к следующему предложению.
42	Сравнение has_car("Ivanov", Mark, _, Cost) и has_car("Ivanov", "Audi", "black", 1500). Унификация успешна.  Подстановка: {Mark = "Audi"}.	Ответ найден. Сохранение подстановки {Mark = "Audi"}. Реконкретизация Mark. Переход к следующему предложению.
43	Сравнение has_car("Ivanov", Mark, _, Cost) и has_car("Volkov", "Lada", "white", 500). Унификация неуспешна.	Откат. Переход к следующему предложению.

Шаг	Сравниваемые термины, результат, подстановка, если есть	Дальнейшие действия
44-50	...	...
51	Сравнение has_car("Ivanov", Mark, _, Cost) и info_by_surname_and_city(Surname, City, Street, Bank, Phone). Унификация неуспешна.	Достижение конца БЗ. Реконкретизация Surname. переход к следующему шагу отн. шага шага 34.
52	Сравнение has_phone(Surname, "333", _) и has_phone("Volkov", "444", address_struct("Sevastopol", "Lenina", 2, 2)) Унификация неуспешна.	Откат. Переход к следующему предложению.
53-64	...	...
65	Сравнение has_phone(Surname, "333", _) и info_by_surname_and_city(Surname, City, Street, Bank, Phone). Унификация неуспешна.	Достижение конца БЗ. Резольвента пуста. Завершение работы.



Шаг	Сравниваемые термины, результат, подстановка, если есть	Дальнейшие действия
1	Сравнение info_by_surname_and_city("Sidorov", "Kazan", Street, Bank, Phone) и has_phone("Petrov", "111", address_struct("Moscow", "Gagarina", 1, 1)). Унификация неуспешна.	Откат. Переход к следующему предложению.
2-16	...	...
17	Сравнение info_by_surname_and_city("Sidorov", "Kazan", Street, Bank, Phone) и info_by_surname_and_city(Surname, City, Street, Bank, Phone). Унификация успешна.  Подстановка: {Street = Street, Bank = Bank, Phone = Phone}.	Переход к терму правила.  Унификация has_phone("Sidorov", Phone, address_struct("Kazan", Street, _ _)).
18	Сравнение has_phone("Sidorov", Phone, address_struct("Kazan", Street, _ _)) и has_phone("Petrov", "111", address_struct("Moscow", "Gagarina", 1, 1)). Унификация неуспешна.	Откат. Переход к следующему предложению.
19-21	...	...
22	Сравнение has_phone("Sidorov", Phone, address_struct("Kazan", Street, _ _)) и has_phone("Sidorov", "555", address_struct("Kazan", "Lomonosova", 3, 3)). Унификация успешна.  Подстановка: {Street = "Lomonosova", Bank = Bank, Phone = "555"}	Переход к терму следующего правила.  Унификация has_deposite("Sidorov", Bank, _ _).
23	Сравнение has_deposite("Sidorov", Bank, _ _) и has_phone("Petrov", "111", address_struct("Moscow", "Gagarina", 1, 1)). Унификация неуспешна.	Откат. Переход к следующему предложению.
24-34	...	...
35	Сравнение has_deposite("Sidorov", Bank, _ _) и has_deposite("Sidorov", "Alpha", 3, 6500). Унификация успешна.  Подстановка: {Street = "Lomonosova", Bank = "Alpha", Phone = "555"}	Ответ найден. Сохранение подстановки {Street = "Lomonosova", Bank = "Alpha", Phone = "555"}. Реконкретизация Bank. Переход к следующему предложению.
36	Сравнение has_deposite("Sidorov", Bank, _ _) и has_deposite("Sidorov", "Alpha", 3, 6500). Унификация успешна.  Подстановка: {Street = "Lomonosova", Bank = "Beta", Phone = "555"}	Ответ найден. Сохранение подстановки {Street = "Lomonosova", Bank = "Beta", Phone = "555"}. Реконкретизация Bank. Переход к следующему предложению.
37-39	...	...
40	Сравнение has_deposite("Sidorov", Bank, _ _) и info_by_surname_and_city(Surname, City, Street, Bank, Phone). Унификация неуспешна.	Достижение конца БЗ. Реконкретизация Phone. Переход к следующему шагу отн. шага 22.
41	Сравнение has_phone("Sidorov", Phone, address_struct("Kazan", Street, _ _)) и has_car("Petrov", "BMW", "black", 1000). Унификация неуспешна.	Откат. Переход к следующему предложению.
42-51	...	...
52	Сравнение has_phone("Sidorov", Phone, address_struct("Kazan", Street, _ _)) и info_by_surname_and_city(Surname, City, Street, Bank, Phone). Унификация неуспешна.	Достижение конца БЗ. Резольвента пуста. Завершение работы.

## Часть 2

Составить программу, объединив в ней информацию-знания (12.1). По Марке и Цвету автомобиля найти Фамилию, Город, Телефон и Банки, в которых владелец автомобиля имеет вклады. Владельца может быть несколько (не более трех), один и ни одного.

1. Для каждого из трех вариантов 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 пример (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»). Сделать вывод: одинаковы ли множество работ и объем в разных случаях.
3. Оформите 2 таблицы, демонстрирующие порядок работы алгоритма унификации во-проса и подходящего заголовка правила (для двух случаев из пункта 2) и укажите результаты его работы: ответ и побочный эффект.

```
1 domains
2     surname, phone = symbol.
3     city, street = symbol.
4     house, flat = integer.
5     address = address_struct(city, street, house, flat)
6     mark, color = symbol.
7     cost = integer.
8     bank = symbol.
9     account, sum = integer.
10
11 predicates
12     has_phone(surname, phone, address)
13     has_car(surname, mark, color, cost)
14     has_deposit(surname, bank, account, sum)
15
16     info_by_phone(phone, surname, mark, cost)
17     only_mark_by_phone(phone, mark)
18     info_by_surname_and_city(surname, city, street, bank, phone)
19
20     info_by_mark_and_color(mark, color, surname, city, phone, bank)
21
22 clauses
23     has_phone("Petrov", "111", address_struct("Moscow", "Gagarina",
24         1, 1)).
25     has_phone("Ivanov", "222", address_struct("Vladimir", "Lenina", 2,
26         2)).
```

```

25     has_phone("Sidorov", "333", address_struct("Kazan", "Lomonosova"
26         , 3, 3)).
27
28     has_car("Petrov", "BMW", "black", 500).
29
30     has_car("Ivanov", "Audi", "white", 1500).
31     has_car("Petrov", "Audi", "white", 1500).
32
33     has_deposit("Petrov", "Alpha", 0, 1000).
34     has_deposit("Ivanov", "Gamma", 1, 2000).
35     has_deposit("Sidorov", "Beta", 2, 3000).
36
37     info_by_phone(Phone, Surname, Mark, Cost) :- has_phone(Surname, Phone,
38         _), has_car(Surname, Mark, _, Cost).
39     only_mark_by_phone(Phone, Mark) :- info_by_phone(Phone, _, Mark, _).
40     info_by_surname_and_city(Surname, City, Street, Bank, Phone) :-
41         has_phone(Surname, Phone, address_struct(City, Street, _, _)),
42         has_deposit(Surname, Bank, _, _).
43
44     info_by_mark_and_color(Mark, Color, Surname, City, Phone, Bank) :-
45         has_car(Surname, Mark, Color, _), has_phone(Surname, Phone,
46             address_struct(City, _, _, _)), has_deposit(Surname, Bank, _, _).
47
48 goal
49     %info_by_mark_and_color("Lada", "white", Surname, City, Phone, Bank).
50     % No solutions
51
52     info_by_mark_and_color("BMW", "black", Surname, City, Phone, Bank).
53     % Surname = Petrov, City = Moscow, Phone = 111, Bank = Alpha
54     % 1 Solution
55
56     %info_by_mark_and_color("Audi", "white", Surname, City, Phone, Bank).
57     % Surname = Ivanov, City = Vladimir, Phone = 222, Bank = Gamma
58     % Surname = Petrov, City = Moscow, Phone = 111, Bank = Alpha
59     % 2 Solutions

```

Шаг	Сравниваемые термы, результат, подстановка, если есть	Дальнейшие действия
1	Сравнение info_by_mark_and_color("Lada", "white", Surname, City, Phone, Bank) и has_phone("Petrov", "111", address_struct("Moscow", "Gagarina", 1, 1)). Унификация неуспешна.	Откат. Переход к следующему предложению.
2-12	...	
13	Сравнение info_by_mark_and_color("Lada", "white", Surname, City, Phone, Bank) и info_by_mark_and_color(Mark, Color, Surname, City, Phone, Bank). Унификация успешна.  Подстановка: {Surname = Surname, City = City, Phone = Phone, Bank = Bank}	Переход к терму правила.  Унификация has_car(Surname, "Lada", "white", _)
14	Сравнение has_car(Surname, "Lada", "white", _) и has_phone("Petrov", "111", address_struct("Moscow", "Gagarina", 1, 1)). Унификация неуспешна.	Откат. Переход к следующему предложению.
14-26	...	...
27	Сравнение has_car(Surname, "Lada", "white", _) и info_by_mark_and_color(Mark, Color, Surname, City, Phone, Bank). Унификация неуспешна.	Достижение конца БЗ. Ответ не найден.

Шаг	Сравниваемые термины, результат, подстановка, если есть	Дальнейшие действия
1	Сравнение info_by_mark_and_color("BMW", "black", Surname, City, Phone, Bank) и has_phone("Petrov", "111", address_struct("Moscow", "Gagarina", 1, 1)). Унификация неуспешна.	Откат. Переход к следующему предложению.
2-12	...	...
13	Сравнение info_by_mark_and_color("BMW", "black", Surname, City, Phone, Bank) и info_by_mark_and_color(Mark, Color, Surname, City, Phone, Bank). Унификация успешна. Подстановка: {Surname = Surname, City = City, Phone = Phone, Bank = Bank}	Переход к терму правила.  Унификация has_car(Surname, "BMW", "black", _).
14	Сравнение has_car(Surname, "BMW", "black", _) и has_phone("Petrov", "111", address_struct("Moscow", "Gagarina", 1, 1)). Унификация неуспешна.	Откат. Переход к следующему предложению.
15-16	...	...
17	Сравнение has_car(Surname, "BMW", "black", _) и has_car("Petrov", "BMW", "black", 500). Унификация успешна. Подстановка: {Surname = "Petrov", City = City, Phone = Phone, Bank = Bank}	Переход к терму следующего правила.  Унификация has_phone("Petrov", Phone, address_struct(City, _, _))
18	Сравнение has_phone("Petrov", Phone, address_struct(City, _, _)) и has_phone("Petrov", "111", address_struct("Moscow", "Gagarina", 1, 1)). Унификация успешна. Подстановка: {Surname = "Petrov", City="Moscow", Phone = 111, Bank = Bank}	Переход к терму следующего правила.  Унификация has_deposite("Petrov", Bank, _, _).
19	Сравнение has_deposite("Petrov", Bank, _, _) и has_phone("Petrov", "111", address_struct("Moscow", "Gagarina", 1, 1)). Унификация неуспешна.	Откат. Переход к следующему предложению.
20-24	...	...
25	Сравнение has_deposite("Petrov", Bank, _, _) и has_deposite("Petrov", "Alpha", 0, 1000). Унификация успешна. Подстановка: {Surname = "Petrov", City="Moscow", Phone = 111, Bank = "Alpha"}	Ответ найден. Сохранение подстановки {Surname = "Petrov", City="Moscow", Phone = 111, Bank = "Alpha"} Реконкретизация Bank. Переход к следующему предложению.
26	Сравнение has_deposite("Petrov", Bank, _, _) и has_deposite("Ivanov", "beta", 1, 2000). Унификация неуспешна.	Откат. Переход к следующему предложению.
27-31	...	...
32	Сравнение has_deposite("Petrov", Bank, _, _) и info_by_mark_and_color(Mark, Color, Surname, City, Phone, Bank). Унификация неуспешна.	Достижение конца БЗ. Реконкретизация Surname, City, Phone. Переход к следующему шагу отн. шага 18.
33	Сравнение has_phone("Petrov", Phone, address_struct(City, _, _)) и has_phone("Ivanov", "222", address_struct("Vladimir", "Lenina", 2, 2)). Унификация неуспешна.	Откат. Переход к следующему предложению.
34-43	...	...

Шаг	Сравниваемые термины, результат, подстановка, если есть	Дальнейшие действия
44	Сравнение has_phone("Petrov", Phone, address_struct(City, _, _)) и info_by_mark_and_color(Mark, Color, Surname, City, Phone, Bank). Унификация неуспешна.	Достижение конца БЗ. Реконкретизация Surname. Переход к следующему предложению отн. шага 17.
45	Сравнение has_car(Surname, "BMW", "black", _) и has_car("Ivanov", "Audi", "white", 1500). Унификация неуспешна.	Откат. Переход к следующему предложению.
46-50	...	...
51	Сравнение has_car(Surname, "BMW", "black", _) и info_by_mark_and_color(Mark, Color, Surname, City, Phone, Bank). Унификация неуспешна.	Достижение конца БЗ. Резольвента пуста. Завершение работы.

Шаг	Сравниваемые термины, результат, подстановка, если есть	Дальнейшие действия
1	Сравнение info_by_mark_and_color("Audi", "white", Surname, City, Phone, Bank) и has_phone("Petrov", "111", address_struct("Moscow", "Gagarina", 1, 1)). Унификация неуспешна.	Откат. Переход к следующему предложению.
2-12	...	...
13	Сравнение info_by_mark_and_color("Audi", "white", Surname, City, Phone, Bank) и info_by_mark_and_color(Mark, Color, Surname, City, Phone, Bank). Унификация успешна.  Подстановка: {Surname = Surname, City = City, Phone = Phone, Bank = Bank}	Переход к терму правила.  Унификация has_car(Surname, "Audi", "white", _).
14	Сравнение has_car(Surname, "Audi", "white", _) и has_phone("Petrov", "111", address_struct("Moscow", "Gagarina", 1, 1)). Унификация неуспешна.	Откат. Переход к следующему предложению.
15-17	...	...
18	Сравнение has_car(Surname, "Audi", "white", _) и has_car("Ivanov", "Audi", "white", 1500). Унификация успешна.  Подстановка: {Surname = "Ivanov", City = City, Phone = Phone, Bank = Bank}	Переход к терму следующего правила.  Унификация has_phone("Ivanov", Phone, address_struct(City, _, _))
19	Сравнение has_phone("Ivanov", Phone, address_struct(City, _, _)) и has_phone("Petrov", "111", address_struct("Moscow", "Gagarina", 1, 1)). Унификация неуспешна.	Откат. Переход к следующему предложению.
20	Сравнение has_phone("Ivanov", Phone, address_struct(City, _, _)) и has_phone("Ivanov", "222", address_struct("Vladimir", "Lenina", 2, 2)). Унификация успешна.  Подстановка: {Surname = "Ivanov", City = "Vladimir", Phone = 222, Bank = Bank}	Переход к терму следующего правила.  Унификация has_deposit("Ivanov", Bank, _, _).
21	Сравнение has_deposit("Ivanov", Bank, _, _) и has_phone("Petrov", "111", address_struct("Moscow", "Gagarina", 1, 1)). Унификация неуспешна.	Откат. Переход к следующему предложению.
22-27	...	...
28	Сравнение has_deposit("Ivanov", Bank, _, _) и has_deposit("Ivanov", "Gamma", 1, 2000). Унификация успешна.  Подстановка: {Surname = "Ivanov", City = "Vladimir", Phone = 222, Bank = "Gamma"}	Ответ найден. Сохранение подстановки {Surname = "Ivanov", City = "Vladimir", Phone = 222, Bank = "Gamma"}. Реконкретизация Bank. Переход к следующему предложению.
29	Сравнение has_deposit("Ivanov", Bank, _, _) и has_deposit("Sidorov", "Beta", 2, 3000). Унификация неуспешна.	Откат. Переход к следующему предложению.
30-33	...	...
34	Сравнение has_deposit("Ivanov", Bank, _, _) и info_by_mark_and_color(Mark, Color, Surname, City, Phone, Bank). Унификация неуспешна.	Достижение конца БЗ. Реконкретизация Surname, City, Phone. переход к следующему предложению отн. шага 19.

Шаг	Сравниваемые термины, результат, подстановка, если есть	Дальнейшие действия
35	Сравнение has_phone("Ivanov", Phone, address_struct(City, _, _)) и has_car("Petrov", "BMW", "black", 500). Унификация неуспешна.	Откат. Переход к следующему предложению.
36-43	---	---
44	Сравнение has_phone("Ivanov", Phone, address_struct(City, _, _)) и info_by_mark_and_color(Mark, Color, Surname, City, Phone, Bank). Унификация неуспешна.	Достижение конца БЗ. Реконретизация Surname. Переход к следующему предложению отн. шага 18.
45	Сравнение has_car(Surname, "Audi", "white", _) и has_car("Petrov", "Audi", "white", 1500). Унификация успешна.  Подстановка: {Surname = "Ivanov", City = City, Phone = Phone, Bank = Bank}	Переход к терму следующего правила.  Унификация has_phone("Petrov", Phone, address_struct(City, _, _))
46	Сравнение has_phone("Petrov", Phone, address_struct(City, _, _)) и has_phone("Petrov", "111", address_struct("Moscow", "Gagarina", 1, 1)). Унификация успешна.  Подстановка: {Surname = "Petrov", City = "Moscow", Phone = "111", Bank = Bank}	Переход к терму следующего правила.  Унификация has_deposite("Petrov", Bank, _, _).
47	Сравнение has_deposite("Petrov", Bank, _, _) и has_phone("Petrov", "111", address_struct("Moscow", "Gagarina", 1, 1)). Унификация неуспешна.	Откат. Переход к следующему предложению.
47-52	---	---
53	Сравнение has_deposite("Petrov", Bank, _, _) и has_deposite("Petrov", "Alpha", 0, 1000). Унификация успешна.  Подстановка: {Surname = "Petrov", City = "Moscow", Phone = "111", Bank = "Alpha"}	Ответ найден. Сохранение подстановки {Surname = "Petrov", City = "Moscow", Phone = "111", Bank = "Alpha"}. Реконретизация Bank. Переход к следующему предложению.
54	Сравнение has_deposite("Petrov", Bank, _, _) и has_deposite("Ivanov", "Gamma", 1, 2000). Унификация неуспешна.	Откат. Переход к следующему предложению.
55-58	---	---
59	Сравнение has_deposite("Petrov", Bank, _, _) и info_by_mark_and_color(Mark, Color, Surname, City, Phone, Bank). Унификация неуспешна.	Достижение конца БЗ. Реконретизация Surname, City, Phone. переход к следующему предложению отн. шага 35.
60	Сравнение has_car(Surname, "Audi", "white", _) и has_deposite("Petrov", "Alpha", 0, 1000). Унификация неуспешна.	Откат. Переход к следующему предложению.
61-65	---	---
66	Сравнение has_car(Surname, "Audi", "white", _) и info_by_mark_and_color(Mark, Color, Surname, City, Phone, Bank). Унификация неуспешна.	Достижение конца БЗ. Резольвента пуста. Завершение работы.



Шаг	Сравниваемые термины, результат, подстановка, если есть	Дальнейшие действия
1	Сравнение info_by_mark_and_color("Audi", "white", Surname, City, Phone, Bank) и has_phone("Sidorov", "333", address_struct("Kazan", "Lomonosova", 3, 3)). Унификация неуспешна.	Откат. Переход к следующему предложению.
2-12	...	...
13	Сравнение info_by_mark_and_color("Audi", "white", Surname, City, Phone, Bank) и info_by_mark_and_color(Mark, Color, Surname, City, Phone, Bank). Унификация успешна.  Подстановка: {Surname = Surname, City = City, Phone = Phone, Bank = Bank}	Переход к терму правила.  Унификация has_car(Surname, "Audi", "white", _).
14	Сравнение has_car(Surname, "Audi", "white", _) и has_phone("Sidorov", "333", address_struct("Kazan", "Lomonosova", 3, 3)). Унификация неуспешна.	Откат. Переход к следующему предложению.
15-17	...	...
18	Сравнение has_car(Surname, "Audi", "white", _) и has_car("Ivanov", "Audi", "white", 1500). Унификация успешна.  Подстановка: {Surname = "Ivanov", City = City, Phone = Phone, Bank = Bank}	Переход к терму следующего правила.  Унификация has_phone("Ivanov", Phone, address_struct(City, _, _))
19	Сравнение has_phone("Ivanov", Phone, address_struct(City, _, _)) и has_phone("Sidorov", "333", address_struct("Kazan", "Lomonosova", 3, 3)). Унификация неуспешна.	Откат. Переход к следующему предложению.
21	Сравнение has_phone("Ivanov", Phone, address_struct(City, _, _)) и has_phone("Ivanov", "222", address_struct("Vladimir", "Lenina", 2, 2)). Унификация успешна.  Подстановка: {Surname = "Ivanov", City = "Vladimir", Phone = 222, Bank = Bank}	Переход к терму следующего правила.  Унификация has_deposit("Ivanov", Bank, _, _).
22	Сравнение has_deposit("Ivanov", Bank, _, _) и has_phone("Sidorov", "333", address_struct("Kazan", "Lomonosova", 3, 3)). Унификация неуспешна.	Откат. Переход к следующему предложению.
23-29	...	...
30	Сравнение has_deposit("Ivanov", Bank, _, _) и has_deposit("Ivanov", "Gamma", 1, 2000). Унификация успешна.  Подстановка: {Surname = "Ivanov", City = "Vladimir", Phone = 222, Bank = "Gamma"}	Ответ найден. Сохранение подстановки {Surname = "Ivanov", City = "Vladimir", Phone = 222, Bank = "Gamma"}. Реконкретизация Bank. Переход к следующему предложению.
31	Сравнение has_deposit("Ivanov", Bank, _, _) и info_by_phone(Phone, Surname, Mark, Cost). Унификация неуспешна.	Откат. Переход к следующему предложению.
31-34	...	...
35	Сравнение has_deposit("Ivanov", Bank, _, _) и info_by_mark_and_color(Mark, Color, Surname, City, Phone, Bank). Унификация неуспешна.	Достижение конца БЗ. Реконкретизация Surname, City, Phone. переход к следующему предложению отн. шага 21.

Шаг	Сравниваемые термины, результат, подстановка, если есть	Дальнейшие действия
36	Сравнение has_phone("Ivanov", Phone, address_struct(City, _, _)) и has_car("Petrov", "BMW", "black", 500). Унификация неуспешна.	Откат. Переход к следующему предложению.
37-44	---	---
45	Сравнение has_phone("Ivanov", Phone, address_struct(City, _, _)) и info_by_mark_and_color(Mark, Color, Surname, City, Phone, Bank). Унификация неуспешна.	Достижение конца БЗ. Реконкретизация Surname. Переход к следующему предложению отн. шага 18.
46	Сравнение has_car(Surname, "Audi", "white", _) и has_car("Petrov", "Audi", "white", 1500). Унификация успешна. Подстановка: {Surname = "Ivanov", City = City, Phone = Phone, Bank = Bank}	Переход к терму следующего правила.  Унификация has_phone("Petrov", Phone, address_struct(City, _, _))
47	Сравнение has_phone("Petrov", Phone, address_struct(City, _, _)) и has_phone("Petrov", "111", address_struct("Moscow", "Gagarina", 1, 1)). Унификация успешна. Подстановка: {Surname = "Petrov", City = "Moscow", Phone = "111", Bank = Bank}	Переход к терму следующего правила.  Унификация has_deposite("Petrov", Bank, _).
48	Сравнение has_deposite("Petrov", Bank, _) и has_phone("Petrov", "111", address_struct("Moscow", "Gagarina", 1, 1)). Унификация неуспешна.	Откат. Переход к следующему предложению.
49-54	---	---
55	Сравнение has_deposite("Petrov", Bank, _) и has_deposite("Petrov", "Alpha", 0, 1000). Унификация успешна. Подстановка: {Surname = "Petrov", City = "Moscow", Phone = "111", Bank = "Alpha"}	Ответ найден. Сохранение подстановки {Surname = "Petrov", City = "Moscow", Phone = "111", Bank = "Alpha". Реконкретизация Bank. Переход к следующему предложению.
56	Сравнение has_deposite("Petrov", Bank, _) и has_deposite("Ivanov", "Gamma", 1, 2000). Унификация неуспешна.	Откат. Переход к следующему предложению.
57-60	---	---
61	Сравнение has_deposite("Petrov", Bank, _) и info_by_mark_and_color(Mark, Color, Surname, City, Phone, Bank). Унификация неуспешна.	Достижение конца БЗ. Реконкретизация Surname, City, Phone. переход к следующему предложению отн. шага 35.
62	Сравнение has_car(Surname, "Audi", "white", _) и has_deposite("Petrov", "Alpha", 0, 1000). Унификация неуспешна.	Откат. Переход к следующему предложению.
63-67	---	---
68	Сравнение has_car(Surname, "Audi", "white", _) и info_by_mark_and_color(Mark, Color, Surname, City, Phone, Bank). Унификация неуспешна.	Достижение конца БЗ. Резольвента пуста. Завершение работы.