



**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»**

Лабораторная работа № 14.

По курсу «Функциональное и логическое программирование».

Студент: Власова Е. В.

Группа: ИУ7-64Б.

Преподаватели: Толпинская Н. Б.

Строганов Ю. В.

2020 г.

Практическая часть.

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.

Листинг программы:

```
domains
home = address(CITY, STREET, HOUSE, FLAT)
SURNAME = string
PHONE = string
CITY, STREET = string
```

```

HOUSE, FLAT = integer
MARKA, COLOR = string
PRICE = integer
BANK = string
NUMBER, MONEY = integer
predicates
    info(SURNAME, PHONE, home)
    cars(SURNAME, CITY, MARKA, COLOR, PRICE)
    banks(SURNAME, CITY, BANK, NUMBER, MONEY)
    find_by_car(SURNAME, CITY, PHONE, MARKA, COLOR, BANK)
clauses
    info("Petrov", "88885353535", address("Moscow", "Saykin street", 10, 143)).
    info("Petrov", "81111111111", address("Kursk", "Non street", 1, 10)).
    info("Krot", "89995353566", address("Moscow", "Nwot street", 5, 60)).
    %2 owners
    cars("Petrov", "Moscow", "TOYOTA", "RED", 256000). %PERSON 1
    cars("Krot", "Moscow", "TOYOTA", "RED", 256000). %PERSON 3
    %1 owner
    cars("Petrov", "Kursk", "LADA", "GREEN", 100000). %PERSON 2
    %0 owners
    cars("Krotova", "Moscow", "Merc", "BLUE", 100000).

    banks("Petrov", "Moscow", "Sberbank", 356, 50000). %PERSON 1
    banks("Petrov", "Kursk", "Tinkoff", 357, 9000). %PERSON 2
    banks("Krot", "Moscow", "Rocket-bank", 20, 2000). %PERSON 3

    find_by_car(SURNAME, CITY, PHONE, MARKA, COLOR, BANK):-
cars(SURNAME, CITY, MARKA, COLOR, _),
info(SURNAME, PHONE, address(CITY, _, _)),
banks(SURNAME, CITY, BANK, _, _).
goal
    find_by_car(SURNAME, CITY, PHONE, "TOYOTA", "RED", BANK).

```

Задание 1.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.

```

find_by_car(SURNAME, CITY, PHONE, MARKA, COLOR, BANK) :- cars(SURNAME,
CITY, MARKA, COLOR, _),
info(SURNAME, PHONE,
address(CITY, _, _)), banks(SURNAME, CITY, BANK, _, _).

```

1. У машины один владелец:

```

Goal
find_by_car(SURNAME, CITY, PHONE, "LADA", "GREEN", BANK).

```

[Inactive D:\ph\BIN\WIN\32\Obj\goal\$000.exe]

SURNAME=Petrov, CITY=Kursk, PHONE=8111111111, BANK=Tinkoff

1 Solution

Порядок формирования ответа для примера 1:


№ шага	Сравниваемые термы; результат; подстановка, если есть	Дальнейшие действия: прямой ход или откат
	<code>find_by_car(SURNAME, CITY, PHONE, MARKA, COLOR, BANK)</code> = <code>find_by_car(SURNAME, CITY, PHONE, «LADA», «GREEN», BANK)</code>	Заносится в стек <code>find_by_car(SURNAME, CITY, PHONE, «LADA», «GREEN», BANK)</code>
	Раскрывается правило, то есть надо доказать каждое утверждение в теле правила (последовательно) <code>cars(SURNAME, CITY, MARKA, COLOR, PRICE)</code> = <code>cars(SURNAME, CITY, «LADA», «GREEN», PRICE)</code>	Прямой ход Заносится в стек <code>cars(SURNAME, CITY, «LADA», «GREEN», _)</code>
	<code>cars(SURNAME, CITY, «LADA», «GREEN», _)</code> По функтору и списку аргументов ищется определение отношения <code>cars("Petrov", "Moscow", "TOYOTA", "RED", 256000)</code>	Определение отношения найдено
	Унификация <code>cars(SURNAME, CITY, «LADA», «GREEN», _)</code> и <code>cars("Petrov", "Moscow", "TOYOTA", "RED", 256000)</code>	Результат сравнения термов false, прямой ход
	Унификация <code>cars(SURNAME, CITY, «LADA», «GREEN», _)</code> и <code>cars("Krot", "Moscow", "TOYOTA", "RED", 256000)</code>	Результат сравнения термов false, прямой ход
	Унификация <code>cars(SURNAME, CITY, «LADA», «GREEN», _)</code> и <code>cars("Petrov", "Kursk", "LADA", "GREEN", 100000)</code>	Результат сравнения термов true S U Переход к следующему утверждению в теле правила (прямой ход)
	Следующее утверждение <code>info(SURNAME, PHONE, address(CITY, _, _))</code>	Заносится в стек <code>info(SURNAME, PHONE, address(CITY, _, _))</code>
	Раскрывается правило <code>info(SURNAME, PHONE, address(CITY, _, _))</code>	Определение отношения найдено

	-> info("Petrov", PHONE, address("Kursk", _, _, _)) По функтору и списку аргументов ищется определение отношения (с самого начала)	
	Унификация info("Petrov", PHONE, address("Kursk", _, _, _)) и info("Petrov", "88885353535", address("Moscow", "Saykin street", 10, 143))	Результат сравнения термов false, прямой ход
	Унификация info("Petrov", PHONE, address("Kursk", _, _, _)) и info("Petrov", "81111111111", address("Kursk", "Non street", 1, 10))	Результат сравнения термов true 8 Анонимные переменные не связываются со значением Переход к следующему утверждению в теле правила (прямой ход)
	Следующее утверждение banks(SURNAME, CITY, BANK, _, _)	Заносится в стек banks(SURNAME, CITY, BANK, _, _)
	Раскрывается правило banks(SURNAME, CITY, BANK, _, _) -> banks("Petrov", "Kursk", BANK, _, _) По функтору и списку аргументов ищется определение отношения	Определение отношения найдено
	Унификация banks("Petrov", "Kursk", BANK, _, _) и banks("Petrov", "Moscow", "Sberbank", 356, 50000)	Результат сравнения термов false, прямой ход
	Унификация banks("Petrov", "Kursk", BANK, _, _) и banks("Petrov", "Kursk", "Tinkoff", 357, 9000)	Результат сравнения термов true Т Вывод результата, переход к следующей строке
	Унификация banks("Petrov", "Kursk", BANK, _, _) и banks("Krot", "Moscow", "Rocket-bank", 20, 2000).	Результат сравнения термов false. В базе знаний больше нет утверждений с заданным именем. Откат, из стека достаётся

		banks(SURNAME, CITY, BANK, _, _)
	Унификация info("Petrov", PHONE, address("Kursk", _, _, _)) и info("Krot", "89995353566", address("Moscow", "Nwot street", 5, 60))	Результат сравнения термов false. В базе знаний больше нет утверждений с заданным именем. Откат, из стека достаётся info(SURNAME, PHONE, address(CITY, _, _, _))
	Унификация cars(SURNAME, CITY, «LADA», «GREEN», _) и cars("Krotova", "Moscow", "Merc", "BLUE", 100000)	Результат сравнения термов false. В базе знаний больше нет утверждений с заданным именем. Откат, из стека достаётся с
	Стек пуст	Стек пуст, завершение программы

2. У машины несколько владельцев:

goal
find_by_car(SURNAME, CITY, PHONE, "TOYOTA", "RED", BANK).

 [Inactive D:\ph\BIN\WIN\32\Obj\goal\$000.exe]

SURNAME=Petrov, CITY=Moscow, PHONE=88885353535, BANK=Sberbank
SURNAME=Krot, CITY=Moscow, PHONE=89995353566, BANK=Rocket-bank
2 Solutions|

Порядок формирования ответа для примера 2:

№ шага	Сравниваемые термы; результат; подстановка, если есть	Дальнейшие действия: прямой ход или откат
	find_by_car(SURNAME, CITY, PHONE, MARKA, COLOR, BANK) = find_by_car(SURNAME, CITY, PHONE, "TOYOTA", "RED", BANK)	Заносится в стек find_by_car(SURNAME, CITY, PHONE, "TOYOTA", "RED", BANK)
	Раскрывается правило, то есть надо доказать каждое утверждение в теле правила (последовательно) cars(SURNAME, CITY, MARKA, COLOR, PRICE) = cars(SURNAME, CITY, "TOYOTA", "RED", _)	Прямой ход Заносится в стек cars(SURNAME, CITY, "TOYOTA", "RED", _)
	cars(SURNAME, CITY, "TOYOTA", "RED", _) По функтору и списку аргументов ищется	Определение отношения найдено

	определение отношения cars("Petrov", "Moscow", "TOYOTA", "RED", 256000)	
	Унификация cars(SURNAME, CITY, "TOYOTA", "RED", _) и cars("Petrov", "Moscow", "TOYOTA", "RED", 256000)	Результат сравнения термов true S C Переход к следующему утверждению в теле правила (прямой ход)
	Следующее утверждение info(SURNAME, PHONE, address(CITY, _, _, _))	
	Раскрывается правило info(SURNAME, PHONE, address(CITY, _, _, _)) -> info("Petrov", PHONE, address("Moscow", _, _, _)) По функтору и списку аргументов ищется определение отношения (с самого начала)	Определение отношения найдено Заносится в стек info("Petrov", PHONE, address("Moscow", _, _, _))
	Унификация info("Petrov", PHONE, address("Moscow", _, _, _)) и info("Petrov", "88885353535", address("Moscow", "Saykin street", 10, 143))	Результат сравнения термов true 8 Анонимные переменные не связываются со значением Переход к следующему утверждению в теле правила (прямой ход)
	Следующее утверждение banks(SURNAME, CITY, BANK, _, _)	
	Раскрывается правило banks(SURNAME, CITY, BANK, _, _) -> b По функтору и списку аргументов ищется определение отношения	Определение отношения найдено Заносится в стек b a n k
	Унификация banks("Petrov", "Moscow", BANK, _, _) и banks("Petrov", "Moscow", "Sberbank", 356, 50000)	Результат сравнения термов true Вывод результата, переход к следующей строке


	Унификация <code>banks("Petrov", "Moscow", BANK, _, _)</code> и <code>banks("Krot", "Moscow", "Rocket-bank", 20, 2000)</code> .	Результат сравнения термов <code>false</code> . В базе знаний больше нет утверждений с заданным именем. Откат, из стека достаётся <code>banks("Petrov",</code>
	Следующее утверждение <code>info(SURNAME, PHONE, address(CITY, _, _, _))</code>	
	Унификация <code>info("Petrov", PHONE, address("Moscow", _, _, _))</code> и <code>info("Petrov", "8111111111", address("Kursk", "Non street", 1, 10))</code>	Результат сравнения термов <code>false</code> . Прямой ход
	Унификация <code>info("Petrov", PHONE, address("Moscow", _, _, _))</code> и <code>info("Krot", "89995353566", address("Moscow", "Nwot street", 5, 60))</code>	Результат сравнения термов <code>false</code> . В базе знаний больше нет утверждений с заданным именем. Откат, из стека достаётся <code>info("Petrov", PHONE, address("Moscow", _, _, _))</code>
	Следующее утверждение <code>cars(SURNAME, CITY, "TOYOTA", "RED", PRICE)</code>	
	Унификация <code>cars(SURNAME, CITY, "TOYOTA", "RED", _)</code> и <code>cars("Krot", "Moscow", "TOYOTA", "RED", 256000)</code>	Результат сравнения термов <code>true</code> S U Переход к следующему утверждению в теле правила (прямой ход)
	Следующее утверждение <code>info(SURNAME, PHONE, address(CITY, _, _, _))</code>	
	Раскрывается правило <code>info(SURNAME, PHONE, address(CITY, _, _, _))</code> <code>-></code> <code>info("Krot", PHONE, address("Moscow", _, _, _))</code> По функтору и списку аргументов ищется	Определение отношения найдено Заносится в стек <code>info("Krot", PHONE, address("Moscow", _, _, _))</code>

	определение отношения (с самого начала)	
	<p>Унификация info("Krot", PHONE, address("Moscow", _, _, _))</p> <p>И info("Petrov", "88885353535", address("Moscow", "Saykin street", 10, 143))</p>	<p>Результат сравнения термов false, прямой ход</p>
	<p>Унификация info("Krot", PHONE, address("Moscow", _, _, _))</p> <p>И info("Petrov", "81111111111", address("Kursk", "Non street", 1, 10))</p>	<p>Результат сравнения термов false, прямой ход</p>
	<p>Унификация info("Krot", PHONE, address("Moscow", _, _, _))</p> <p>И info("Krot", "89995353566", address("Moscow", "Nwot street", 5, 60))</p>	<p>Результат сравнения термов true "89995353566"</p> <p>Переход к следующему утверждению в теле правила (прямой ход)</p>
	Следующее утверждение banks(SURNAME, CITY, BANK, _, _)	
	<p>Раскрывается правило</p> <p>banks(SURNAME, CITY, BANK, _, _)</p> <p>-></p> <p>b</p> <p>По функтору и списку аргументов ищется определение отношения</p>	<p>Определение отношения найдено</p> <p>Заносится в стек</p> <p>b</p> <p>a</p> <p>n</p> <p>k</p>
	<p>Унификация banks("Krot", "Moscow", BANK, _, _)</p> <p>И banks("Petrov", "Moscow", "Sberbank", 356, 50000)</p>	<p>Результат сравнения термов false, прямой ход</p>
	<p>Унификация banks("Krot", "Moscow", BANK, _, _)</p> <p>И banks("Petrov", "Kursk", "Tinkoff", 357, 9000)</p>	<p>Результат сравнения термов false, прямой ход</p>
	<p>Унификация banks("Krot", "Moscow", BANK, _, _)</p> <p>И banks("Krot", "Moscow", "Rocket-bank", 20, 2000)</p>	<p>Результат сравнения термов true</p> <p>Вывод результата, переход к следующей строке.</p> <p>В базе знаний больше нет утверждений с заданным именем. Откат, из стека</p> <p>д</p> <p>о</p>
		<p>В базе знаний больше нет утверждений с заданным именем. Откат, из стека достаётся info("Krot",</p>

		PHONE, address("Moscow", _, _, _))
	Следующее утверждение cars(SURNAME, C	
	Унификация cars(SURNAME, CITY, "TOYOTA", "RED", _) и cars("Petrov", "Kursk", "LADA", "GREEN", 100000)	Результат сравнения термов false, прямой ход
	Унификация cars(SURNAME, CITY, "TOYOTA", "RED", _) и cars("Krotova", "Moscow", "Merc", "BLUE", 100000)	Результат сравнения термов false, прямой ход. В базе знаний больше нет утверждений с заданным именем. Откат, из стека достаётся cars(SURNAME, CITY, "TOYOTA", "RED", _)
	Стек пуст	Стек пуст, завершение цикла.

3. У машины нет владельца (он отсутствует в базе):

find_by_car(SURNAME, CITY, PHONE, "Merc", "BLUE", BANK).

 [Inactive D:\ph\BIN\WIN\32\Obj\goal\$000.exe]

No Solution

Порядок формирования ответа для примера 1:

№ шага	Сравниваемые термы; результат; подстановка, если есть	Дальнейшие действия: прямой ход или откат
	find_by_car(SURNAME, CITY, PHONE, MARKA, COLOR, BANK) = find_by_car(SURNAME, CITY, PHONE, "Merc", "BLUE", BANK)	Заносится в стек find_by_car(SURNAME, CITY, PHONE, "Merc", "BLUE", BANK)
	Раскрывается правило, то есть надо доказать каждое утверждение в теле правила (последовательно) cars(SURNAME, CITY, MARKA, COLOR, PRICE) = cars(SURNAME, CITY, "Merc", "BLUE", _)	Прямой ход Заносится в стек cars(SURNAME, CITY, "Merc", "BLUE", _)
	cars(SURNAME, CITY, "Merc", "BLUE", _) По функтору и списку аргументов ищется определение отношения cars("Petrov", "Moscow", "TOYOTA", "RED", 256000)	Определение отношения найдено

	Унификация cars(SURNAME, CITY, "Merc", "BLUE", _) и cars("Petrov", "Moscow", "TOYOTA", "RED", 256000)	Результат сравнения термов false, прямой ход
	Унификация cars(SURNAME, CITY, "Merc", "BLUE", _) и cars("Krot", "Moscow", "TOYOTA", "RED", 256000)	Результат сравнения термов false, прямой ход
	Унификация cars(SURNAME, CITY, "Merc", "BLUE", _) и cars("Petrov", "Kursk", "LADA", "GREEN", 100000)	Результат сравнения термов false, прямой ход.
	Унификация cars(SURNAME, CITY, "Merc", "BLUE", _) и cars("Krotova", "Moscow", "Merc", "BLUE", 100000)	Результат сравнения термов true SURNAME = "Krotova" CITY = "Moscow" Переход к следующему утверждению в теле правила (прямой ход)
	Следующее утверждение info(SURNAME, PHONE, address(CITY, _, _, _))	
	info(SURNAME, PHONE, address(CITY, _, _, _)) = info("Krotova", PHONE, address("Moscow", _, _, _))	Заносится в стек info("Krotova", PHONE, address("Moscow", _, _, _))
	Унификация info("Krotova", PHONE, address("Moscow", _, _, _)) И info("Petrov", "88885353535", address("Moscow", "Saykin street", 10, 143))	Результат сравнения термов false, прямой ход.
	Унификация info("Krotova", PHONE, address("Moscow", _, _, _)) И info("Petrov", "81111111111", address("Kursk", "Non street", 1, 10))	Результат сравнения термов false, прямой ход.
	Унификация info("Krotova", PHONE, address("Moscow", _, _, _)) И info("Krot", "89995353566", address("Moscow", "Nwot street", 5, 60))	Результат сравнения термов false, прямой ход. В базе знаний больше нет утверждений с заданным именем. Откат, из стека достаётся info("Krotova",

		PHONE, address("Moscow",,,_,_,_))
	Следующее утверждение cars(SURNAME, CITY, "Merc", "BLUE", _)	В базе знаний больше нет утверждений с заданным именем. Откат, из стека достаётся cars(SURNAME, CITY, "Merc", "BLUE", _)
	Стек пуст	Стек пуст, завершение программы

Задание 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?

АВТОМОБИЛИ – СПРАВОЧНИК - БАНКИ

№ шага	Сравниваемые термы; результат; подстановка, если есть	Дальнейшие действия: прямой ход или откат
	find_by_car(SURNAME, CITY, PHONE, MARKA, COLOR, BANK) = find_by_car(SURNAME, CITY, PHONE, "TOYOTA", "RED", BANK)	Заносится в стек find_by_car(SURNAME, CITY, PHONE, "TOYOTA", "RED", BANK)
	Раскрывается правило, то есть надо доказать каждое утверждение в теле правила (последовательно) cars(SURNAME, CITY, MARKA, COLOR, _) = cars(SURNAME, CITY, "TOYOTA", "RED", _)	Прямой ход Заносится в стек cars(SURNAME, CITY, "TOYOTA", "RED", _)
	cars(SURNAME, CITY, "TOYOTA", "RED", _) По функтору и списку аргументов ищется определение отношения cars("Petrov", "Moscow", "TOYOTA", "RED", 256000)	Определение отношения найдено
	Унификация cars(SURNAME, CITY, "TOYOTA", "RED", _) и cars("Petrov", "Moscow", "TOYOTA", "RED", 256000)	Результат сравнения термов true S C Переход к следующему утверждению в теле

		правила (прямой ход)
	Следующее утверждение info(SURNAME, PHONE, address(CITY, _, _))	
	Раскрывается правило info(SURNAME, PHONE, address(CITY, _, _)) -> info("Petrov", PHONE, address("Moscow", _, _)) По функтору и списку аргументов ищется определение отношения (с самого начала)	Определение отношения найдено Заносится в стек info("Petrov", PHONE, address("Moscow", _, _, _))
	Унификация info("Petrov", PHONE, address("Moscow", _, _, _)) и info("Petrov", "88885353535", address("Moscow", "Saykin street", 10, 143))	Результат сравнения термов true Анонимные переменные не связываются со значением Переход к следующему утверждению в теле правила (прямой ход)
	Следующее утверждение banks(SURNAME, CITY, BANK, _, _)	
	Раскрывается правило banks(SURNAME, CITY, BANK, _, _) -> banks("Petrov", "Moscow", BANK, _, _) По функтору и списку аргументов ищется определение отношения	Определение отношения найдено Заносится в стек b a n k
	Унификация banks("Petrov", "Moscow", BANK, _, _) и banks("Petrov", "Moscow", "Sberbank", 356, 50000)	Результат сравнения термов true Вывод результата, переход к следующей строке
	Унификация banks("Petrov", "Moscow", BANK, _, _) и banks("Petrov", "Kursk", "Tinkoff", 357, 9000)	Результат сравнения термов false. Прямой ход
	Унификация banks("Petrov", "Moscow", BANK, _, _) и banks("Krot", "Moscow", "Rocket-bank",	Результат сравнения термов false. В базе знаний больше нет утверждений с заданным

	20, 2000).	именем. Откат, из стека достаётся banks("Petrov",
	Следующее утверждение info(SURNAME, PHONE, address(CITY, _, _, _))	
	Унификация info("Petrov", PHONE, address("Moscow", _, _, _)) и info("Petrov", "8111111111", address("Kursk", "Non street", 1, 10))	Результат сравнения термов false. Прямой ход
	Унификация info("Petrov", PHONE, address("Moscow", _, _, _)) и info("Krot", "89995353566", address("Moscow", "Nwot street", 5, 60))	Результат сравнения термов false. В базе знаний больше нет утверждений с заданным именем. Откат, из стека достаётся info("Petrov", PHONE, address("Moscow", _, _, _))
	Следующее утверждение cars(SURNAME, CITY, "TOYOTA", "RED", PRICE)	
	Унификация cars(SURNAME, CITY, "TOYOTA", "RED", _) и cars("Krot", "Moscow", "TOYOTA", "RED", 256000)	Результат сравнения термов true S U Переход к следующему утверждению в теле Правила (прямой ход)
	Следующее утверждение info(SURNAME, PHONE, address(CITY, _, _, _))	
	Раскрывается правило info(SURNAME, PHONE, address(CITY, _, _, _)) -> info("Krot", PHONE, address("Moscow", _, _, _)) По функтору и списку аргументов ищется определение отношения (с самого начала)	Определение отношения найдено Заносится в стек info("Krot", PHONE, address("Moscow", _, _, _))

	Унификация info("Krot", PHONE, address("Moscow", _, _, _)) И info("Petrov", "88885353535", address("Moscow", "Saykin street", 10, 143))	Результат сравнения термов false, прямой ход
	Унификация info("Krot", PHONE, address("Moscow", _, _, _)) И info("Petrov", "81111111111", address("Kursk", "Non street", 1, 10))	Результат сравнения термов false, прямой ход
	Унификация info("Krot", PHONE, address("Moscow", _, _, _)) И info("Krot", "89995353566", address("Moscow", "Nwot street", 5, 60))	Результат сравнения термов true Переход к следующему утверждению в теле правила (прямой ход)
	Следующее утверждение banks(SURNAME, CITY, BANK, _, _)	
	Раскрывается правило banks(SURNAME, CITY, BANK, _, _) -> banks("Krot", "Moscow", BANK, _, _) По функтору и списку аргументов ищется определение отношения	Определение отношения найдено Заносится в стек b a n k
	Унификация banks("Krot", "Moscow", BANK, _, _) И banks("Petrov", "Moscow", "Sberbank", 356, 50000)	Результат сравнения термов false, прямой ход
	Унификация banks("Krot", "Moscow", BANK, _, _) И banks("Petrov", "Kursk", "Tinkoff", 357, 9000)	Результат сравнения термов false, прямой ход
	Унификация banks("Krot", "Moscow", BANK, _, _) И banks("Krot", "Moscow", "Rocket-bank", 20, 2000)	Результат сравнения термов true Вывод результата, переход к следующей строке. В базе знаний больше нет утверждений с заданным именем. Откат, из стека д о
		В базе знаний больше нет утверждений с заданным именем. Откат, из стека достаётся info("Krot",

		PHONE, address("Moscow", _, _, _))
	Следующее утверждение cars(SURNAME, C	
	Унификация cars(SURNAME, CITY, "TOYOTA", "RED", _) и cars("Petrov", "Kursk", "LADA", "GREEN", 100000)	Результат сравнения термов false, прямой ход
	Унификация cars(SURNAME, CITY, "TOYOTA", "RED", _) и cars("Krotova", "Moscow", "Merc", "BLUE", 100000)	Результат сравнения термов false, прямой ход. В базе знаний больше нет утверждений с заданным именем. Откат, из стека достаётся cars(SURNAME, CITY, "TOYOTA", "RED", _)
	Стек пуст	Стек пуст, завершение цикла.

СПРАВОЧНИК – АВТОМОБИЛИ – БАНКИ

№ шага	Сравниваемые термы; результат; подстановка, если есть	Дальнейшие действия: прямой ход или откат
	find_by_car(SURNAME, CITY, PHONE, MARKA, COLOR, BANK) = find_by_car(SURNAME, CITY, PHONE, "TOYOTA", "RED", BANK)	Заносится в стек find_by_car(SURNAME, CITY, PHONE, "TOYOTA", "RED", BANK)
	Раскрывается правило, то есть надо доказать каждое утверждение в теле правила (последовательно) info(SURNAME, PHONE, home) = info(SURNAME, PHONE, address(CITY, _, _))	Прямой ход Заносится в стек info(SURNAME, PHONE, address(CITY, _, _))
	Унификация info(SURNAME, PHONE, address(CITY, _, _)) и info("Petrov", "88885353535", address("Moscow", "Saykin street", 10, 143))	Результат сравнения термов true S U R Анонимные переменные не связываются со значением Переход к следующему утверждению в теле правила (прямой ход)
	Следующее утверждение cars(SURNAME, CITY, MARKA, COLOR, _)	

	<p>Раскрывается правило</p> <p><code>cars(SURNAME, CITY, MARKA, COLOR, _)</code> <code>=</code> <code>cars(«Petrov», «Moscow», "TOYOTA", "RED", _)</code> По функтору и списку аргументов ищется определение отношения</p>	<p>Определение отношения найдено</p> <p>Заносится в стек</p> <p><code>cars(«Petrov», «Moscow», "TOYOTA", "RED", _)</code></p>
	<p>Унификация <code>cars(«Petrov», «Moscow», "TOYOTA", "RED", _)</code></p> <p>и</p> <p><code>cars("Petrov", "Moscow", "TOYOTA", "RED", 256000)</code></p>	<p>Результат сравнения термов true</p> <p>Анонимные переменные не связываются со значением</p> <p>Переход к следующему утверждению в теле правила (прямой ход)</p>
	<p>Следующее утверждение</p> <p><code>banks(SURNAME, CITY, BANK, _, _)</code></p>	
	<p>Раскрывается правило</p> <p><code>banks(SURNAME, CITY, BANK, _, _)</code> <code>-></code> <code>banks("Petrov", "Moscow", BANK, _, _)</code> По функтору и списку аргументов ищется определение отношения</p>	<p>Определение отношения найдено</p> <p>Заносится в стек</p> <p>b a n k s</p>
	<p>Унификация <code>banks("Petrov", "Moscow", BANK, _, _)</code></p> <p>и</p> <p><code>banks("Petrov", "Moscow", "Sberbank", 356, 50000)</code></p>	<p>Результат сравнения термов true</p> <p>Вывод результата, переход к следующей строке</p>
	<p>Унификация <code>banks("Petrov", "Moscow", BANK, _, _)</code></p> <p>и</p> <p><code>banks("Krot", "Moscow", "Rocket-bank", 20, 2000)</code>.</p>	<p>Результат сравнения термов false.</p> <p>В базе знаний больше нет утверждений с заданным именем.</p> <p>Откат, из стека достаётся <code>_, _</code></p>
	<p>Следующее утверждение</p> <p><code>cars(«Petrov», «Moscow», "TOYOTA", "RED", _)</code></p>	
	<p>Унификация <code>cars(«Petrov», «Moscow», "TOYOTA", "RED", _)</code></p> <p>и</p> <p><code>cars("Krot", "Moscow", "TOYOTA", "RED", 256000)</code></p>	<p>Результат сравнения термов false.</p> <p>Прямой ход</p>
	<p>Унификация <code>cars(«Petrov», «Moscow», "TOYOTA", "RED", _)</code></p> <p>и</p>	

	<code>cars("Petrov", "Kursk", "LADA", "GREEN", 100000)</code>	
	<p>Унификация <code>cars(«Petrov», «Moscow», "TOYOTA", "RED", _)</code></p> <p>и</p> <p><code>cars("Krotova", "Moscow", "Merc", "BLUE", 100000)</code></p>	<p>Результат сравнения термов false.</p> <p>В базе знаний больше нет утверждений с заданным именем.</p> <p>Откат, из стека достаётся <code>cars(«Petrov», «Moscow», "TOYOTA", "RED", _)</code></p>
	Следующее утверждение <code>info(SURNAME, PHONE, address(CITY, _, _))</code>	
	<p>Унификация <code>info(SURNAME, PHONE, address(CITY, _, _))</code> и <code>info("Petrov", "8111111111", address("Kursk", "Non street", 1, 10))</code></p>	<p>Результат сравнения термов true</p> <p>S</p> <p>U</p> <p>Переход к следующему утверждению в теле правила (прямой ход)</p>
	Следующее утверждение <code>cars(SURNAME, CITY, "TOYOTA", "RED", _)</code>	
	<p>Раскрывается правило</p> <p><code>cars(SURNAME, CITY, "TOYOTA", "RED", _)</code></p> <p>-></p> <p><code>cars(«Petrov», «Kursk», "TOYOTA", "RED", _)</code></p> <p>По фактору и списку аргументов ищется определение отношения</p>	<p>Определение отношения найдено</p> <p>Заносится в стек</p> <p><code>cars(«Petrov», «Kursk», "TOYOTA", "RED", _)</code></p>
	<p>Унификация <code>cars(«Petrov», «Kursk», "TOYOTA", "RED", _)</code></p> <p>И <code>cars("Petrov", "Moscow", "TOYOTA", "RED", 256000)</code></p>	Результат сравнения термов false, прямой ход
	<p>Унификация <code>cars(«Petrov», «Kursk», "TOYOTA", "RED", _)</code></p> <p>И <code>cars("Krot", "Moscow", "TOYOTA", "RED", 256000)</code></p>	Результат сравнения термов false, прямой ход
	<p>Унификация <code>cars(«Petrov», «Kursk», "TOYOTA", "RED", _)</code></p> <p>И <code>cars("Petrov", "Kursk", "LADA", "GREEN", 100000)</code></p>	Результат сравнения термов false, прямой ход
	Унификация <code>cars(«Petrov», «Kursk», "TOYOTA", "RED", _)</code>	Результат сравнения термов false, прямой ход

	И cars("Krotova", "Moscow", "Merc", "BLUE", 100000)	Вывод результата, переход к следующей строке. В базе знаний больше нет утверждений с заданным именем. О т
	Следующее утверждение info(SURNAME, PHONE, address(CITY, _, _))	
	Унификация info(SURNAME, PHONE, address(CITY, _, _)) и info("Krot", "89995353566", address("Moscow", "Nwot street", 5, 60))	Результат сравнения термов true SURNAME = "Krot" PHONE = "89995353566" CITY = "Moscow" Переход к следующему утверждению в теле правила (прямой ход)
	Следующее утверждение cars(SURNAME, CITY, "TOYOTA", "RED", _)	
	Раскрывается правило cars(SURNAME, CITY, "TOYOTA", "RED", _) -> cars(«Krot», «Moscow», "TOYOTA", "RED", _) По функтору и списку аргументов ищется определение отношения	Определение отношения найдено Заносится в стек cars(«Krot», «Moscow», "TOYOTA", "RED", _)
	Унификация cars(«Krot», «Moscow», "TOYOTA", "RED", _) И cars("Petrov", "Moscow", "TOYOTA", "RED", 256000)	Результат сравнения термов false, прямой ход
	Унификация cars(«Krot», «Moscow», "TOYOTA", "RED", _) И cars("Krot", "Moscow", "TOYOTA", "RED", 256000)	Результат сравнения термов true Переход к следующему утверждению в теле правила (прямой ход)
	Следующее утверждение banks(SURNAME, CITY, BANK, _, _)	
	Раскрывается правило banks(SURNAME, CITY, BANK, _, _) -> banks("Krot", "Moscow", BANK, _, _) По функтору и списку аргументов ищется определение отношения	Определение отношения найдено Заносится в стек b a n k s

	Унификация banks("Krot", "Moscow", BANK, _, _) И banks("Petrov", "Moscow", "Sberbank", 356, 50000)	Результат сравнения термов false, прямой ход
	Унификация banks("Krot", "Moscow", BANK, _, _) И banks("Petrov", "Kursk", "Tinkoff", 357, 9000)	Результат сравнения термов false, прямой ход
	Унификация banks("Krot", "Moscow", BANK, _, _) И banks("Krot", "Moscow", "Rocket-bank", 20, 2000)	Результат сравнения термов true Вывод результата, переход к следующей строке. В базе знаний больше нет утверждений с заданным именем. О т
	Следующее утверждение cars(SURNAME, С	
	Унификация cars(«Krot», «Moscow», "TOYOTA", "RED", _) и cars("Petrov", "Kursk", "LADA", "GREEN", 100000)	Результат сравнения термов false, прямой ход
	Унификация cars(SURNAME, CITY, "TOYOTA", "RED", _) и cars("Krotova", "Moscow", "Merc", "BLUE", 100000)	Результат сравнения термов false, прямой ход. В базе знаний больше нет утверждений с заданным именем. Откат, из стека достаётся cars(«Krot», «Moscow», "TOYOTA", "RED", _) В базе знаний больше нет утверждений с заданным именем Из стека достаётся info(SURNAME, PHONE, address(CITY, _, _, _))
	Стек пуст	Стек пуст, завершение цикла.

Вывод: Prolog просматривает все предложения последовательно, сравнивая текущий терм со всеми в разделе clauses. Поэтому не зависимо от последовательности предикатов, объём всегда будет одинаков.

Порядок работы алгоритма унификации вопроса и подходящего заголовка правила
Унификация терма find_by_car(SURNAME, CITY, PHONE, MARKA, COLOR, BANK) и find_by_car(SURNAME, CITY, PHONE, "TOYOTA", "RED", BANK)

Шаг унификации	Результирующая ячейка	Рабочее поле	Стек
ии			

			find_by_car(SURNAME , CITY, PHONE, MARKA, COLOR, BANK) find_by_car(SURNAME , CITY, PHONE, "TOYOTA", "RED", BANK)
		find_by_car(SURNAME , CITY, PHONE, MARKA, COLOR, BANK) = find_by_car(SURNAME , CITY, PHONE, "TOYOTA", "RED", BANK)	MARKA = "TOYOTA" COLOR = "RED"
	MARKA = "TOYOTA"	MARKA = "TOYOTA"	COLOR = "RED"
	COLOR = "RED"	COLOR = "RED"	Стек пуст
		Очистка поля	

Порядок работы алгоритма унификации вопроса и подходящего заголовка правила
 Унификация терма cars(SURNAME, CITY, "TOYOTA", "RED", PRICE) и cars("Petrov", "Moscow", "TOYOTA", "RED", 256000)

Шаг унификации	Результирующая ячейка	Рабочее поле	Стек
			cars(SURNAME, CITY, "TOYOTA", "RED", PRICE) = cars("Petrov", "Moscow", "TOYOTA", "RED", 256000)
		cars(SURNAME, CITY, "TOYOTA", "RED", _) = cars("Petrov", "Moscow", "TOYOTA", "RED", 256000)	"Petrov" ="Moscow" "TOYOTA"="TOYOTA" "RED"="RED"
	"Petrov"	"Petrov"	CITY="Moscow" "TOYOTA"="TOYOTA" "RED"="RED"

	"Petrov" CITY="Moscow"	CITY="Moscow"	"TOYOTA"="TOYOTA" "RED"="RED"
	"Petrov" CITY="Moscow"	"TOYOTA"="TOYOTA"	"RED"="RED"
	"Petrov" CITY="Moscow"	"RED"="RED"	Стек пуст
		Очистка поля	Стек пуст

Теоретическая часть.

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

Знания о предметной области выражаются на языке Пролог в виде предложений, называемых утверждениями (clauses).

2. Что такое процедура?

Процедурой называется совокупность правил, заголовки которых имеют одно и то же имя и одну и ту же аргументность (местность), т.е. это совокупность правил, описывающих одно определенное отношение

3. Сколько в БЗ текущего задания процедур?

Три. «Телефонный справочник», «Автомобили», «Вкладчики банков»

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

Пусть $\theta = \{X_1 = t_1, X_2 = t_2, \dots, X_n = t_n\}$ – подстановка, тогда результат применения подстановки к терму обозначается: $A\theta$. Применение подстановки заключается в замене каждого вхождения переменной X_i на соответствующий терм. Терм B называется примером терма A , если существует такая подстановка θ , что $B = A\theta$.

В процессе выполнения программы – система, используя встроенный алгоритм унификации, пытается обосновать возможность истинности вопроса, строя подстановки и примеры термов (вопроса и формулировки знания), используя базу знаний.

5. Что такое наиболее общий пример?

Терм C называется общим примером термов A и B , если существуют такие подстановки θ_1 и θ_2 , что $C = A\theta_1$ и $C = B\theta_2$

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 двух термов – это основной шаг доказательства. В процессе работы система выполняет большое число унификаций. Унификация – операция, которая позволяет формализовать процесс логического вывода.

С помощью алгоритма унификации происходит двунаправленная передача параметров процедурам. Например, из внешнего мира в программу для дальнейшего использования или из программы во внешний мир – значения интересующего нас параметра.

7. В каком случае запускается механизм отката?

Откат дает возможность получить много решений в одном вопросе к программе.

Во всех точках программы, где существуют альтернативы, в стек заносятся точки возврата.

Если впоследствии окажется, что выбранный вариант не приводит к успеху, то осуществляется откат к последней из имеющихся в стеке точек программы, где был выбран один из альтернативных вариантов.

Выбирается очередной вариант, программа продолжает свою работу. Если все варианты в точке уже были использованы, то регистрируется неудачное завершение и осуществляется переход на предыдущую точку возврата, если такая есть.

При откате все связанные переменные, которые были означены после этой точки, опять освобождаются.

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

При поступлении вопроса с переменной в Пролог-систему. Например:

info("Mark", X).

X – переменная, входящая в вопрос, изначально является неконкретизированной.

Пролог просматривает базу данных в поисках факта, сопоставимого с вопросом. Если неконкретизированная переменная появляется в качестве одного из аргументов, то Пролог считает, что такой аргумент сопоставим с любым другим аргументом, находящимся в том же факте. При обнаружении такого факта переменная X становится конкретизированной, обозначая объект, являющийся вторым аргументом найденного факта.

info(SURNAME, PHONE, address(CITY, _, _, _))

Это относится только к именованным переменным. Анонимные переменные не могут быть связаны со значением. Используются, если нас не интересует значение данного параметра.

Если составные термы, факты, правила и вопросы не содержат переменных, то они

называются основными. Составные термы, факты, правила и вопросы в момент фиксации в программе могут содержать переменные, тогда они называются неосновными.