

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.

Практическая часть.

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

- «**Телефонный справочник»:** Фамилия, №тел, Адрес <u>структура</u> (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).
```

SURNAME=Petrov, CITY=Kursk, PHONE=81111111111, BANK=Tinkoff 1 Solution

Порядок формирования ответа для примера 1:

№	Спорицироми за торми за примера 1.	Дальнейшие действия:
	Сравниваемые термы; результат;	1
шага	подстановка, если есть	прямой ход или откат
	find_by_car(SURNAME, CITY, PHONE,	Заносится в стек
	MARKA, COLOR, BANK)	find_by_car(SURNAME,
		CITY, PHONE, «LADA»,
	find_by_car(SURNAME, CITY, PHONE,	«GREEN», BANK)
	«LADA», «GREEN», BANK)	
	Раскрывается правило, то есть надо	Прямой ходб
	доказать каждое утверждение в теле	Заносится в стек
	правила (последовательно)	cars(SURNAME, CITY,
	cars(SURNAME, CITY, MARKA, COLOR,	«LADA», «GREEN», _)
	PRICE)	
	=	
	cars(SURNAME, CITY, «LADA»,	
	«GREEN», PRICE)	
	cars(SURNAME, CITY, «LADA»,	Определение отношения
	«GREEN»,)	найдено
	По функтору и списку аргументов ищется	
	определение отношения	
	cars("Petrov", "Moscow", "TOYOTA",	
	"RED", 256000)	
	Унификация cars(SURNAME, CITY,	Результат сравнения
	«LADA», «GREEN»,) и cars("Petrov",	термов false, прямой ход
	"Moscow", "TOYOTA", "RED", 256000)	термов таке, примон ход
	Унификация cars(SURNAME, CITY,	Результат сравнения
	«LADA», «GREEN»,) и	1 -
	· · · · · · · · · · · · · · · · · · ·	термов false, прямой ход
	cars("Krot", "Moscow", "TOYOTA", "RED",	
	256000)	D.
	Унификация cars(SURNAME, CITY,	Результат сравнения
	«LADA», «GREEN», _)	термов true
	И	S
	cars("Petrov", "Kursk", "LADA", "GREEN",	\Box
	100000)	Переход к следующему
		Ж гверждению в теле
		Жравила (прямой ход)
	Следующее утверждение	Заносится в стек
		info(SURNAME, PHONE,
	info(SURNAME, PHONE, address(CITY, _, _,	address(CITY, _, _, _))
	_))	
	D	0
	Раскрывается правило	Определение отношения
	info(SURNAME, PHONE, address(CITY, _, _,	найдено
	_))	
	<u> </u>	i .

-> info("Petrov", PHONE, address("Kursk", _, _, _))	
По функтору и списку аргументов ищется определение отношения (с самого начала)	
Унификация info("Petrov", PHONE, address("Kursk", _, _, _)) и	Результат сравнения термов false, прямой ход
info("Petrov", "88885353535", address("Moscow", "Saykin street", 10, 143))	
Унификация info("Petrov", PHONE, address("Kursk", _, _, _)) и	Результат сравнения термов true
info("Petrov", "8111111111", address("Kursk", "Non street", 1, 10)	8 Анонимные переменные фе связываются со
	ЗначениемПереход к следующемуутверждению в телешравила (прямой ход)
Следующее утверждение	Заносится в стек banks(SURNAME, CITY,
banks(SURNAME, CITY, BANK, _, _)	BANK, _, _)
Раскрывается правило	Определение отношения найдено
banks(SURNAME, CITY, BANK, _, _) ->	
banks("Petrov", "Kursk", BANK, _, _) По функтору и списку аргументов ищется определение отношения	
Унификация banks("Petrov", "Kursk", BANK, _, _)	Результат сравнения термов false, прямой ход
banks("Petrov", "Moscow", "Sberbank", 356, 50000)	
Унификация banks("Petrov", "Kursk", BANK, _, _)	Результат сравнения термов true T
banks("Petrov", "Kursk", "Tinkoff", 357, 9000)	Вывод результата, переход к следующей ктроке
Унификация banks("Petrov", "Kursk", BANK, _, _)	Результат сравнения термов false. В базе знаний больше нет
banks("Krot", "Moscow", "Rocket-bank", 20, 2000).	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:

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

T	
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 U Rереход к следующему Втверждению в теле Кравила (прямой ход)
Следующее утверждение	
info(SURNAME, PHONE, address(CITY, _, _, _, _,))	
Раскрывается правило info(SURNAME, PHONE, address(CITY, _, _, _, _, _)) ->	Определение отношения найдено Заносится в стек info("Petrov", PHONE, address("Moscow", _, _, _,))
info("Petrov", PHONE, address("Moscow", _, _, _, _, _)) По функтору и списку аргументов ищется	
определение отношения (с самого начала)	
Унификация info("Petrov", PHONE, address("Moscow", _, _, _)) и	Результат сравнения термов true 8
info("Petrov", "88885353535", address("Moscow", "Saykin street", 10, 143))	Анонимные переменные не связываются со вначением Переход к следующему этверждению в теле правила (прямой ход)
Следующее утверждение	правила (примон ход)
banks(SURNAME, CITY, BANK, ,)	
Раскрывается правило	Определение отношения
banks(SURNAME, CITY, BANK, _, _)	найдено Заносится в стек
b аПо функтору и списку аргументов ищется	b a n
определение отношения Унификация banks("Petrov", "Moscow", BANK, ,)	Pезультат сравнения термов true
banks("Petrov", "Moscow", "Sberbank", 356, 50000)	Вывод результата, переход к следующей строке
 	· · · · · · · · · · · · · · · · · · ·

Унификация banks("Petrov", "Moscow", 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("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 Rереход к следующему Втверждению в теле Жравила (прямой ход)
Следующее утверждение	(
info(SURNAME, PHONE, address(CITY, _, _, _, _))	
Раскрывается правило	Определение отношения
<pre>info(SURNAME, PHONE, address(CITY, _, _, _)) -></pre>	найдено Заносится в стек info("Krot", PHONE, address("Moscow", _, _, _,))
info("Krot", PHONE, address("Moscow", _, _, _, _,))	
По функтору и списку аргументов ищется	

определение отношения (с самого начала)	
Унификация info("Krot", PHONE, address("Moscow", _, _, _))	Результат сравнения термов false, прямой ход
И info("Petrov", "88885353535", address("Moscow", "Saykin street", 10, 143))	
Унификация info("Krot", PHONE, address("Moscow", _, _, _))	Результат сравнения термов false, прямой ход
И info("Petrov", "8111111111", address("Kursk", "Non street", 1, 10))	
Унификация info("Krot", PHONE, address("Moscow", _, _, _))	Результат сравнения термов true
И info("Krot", "89995353566", address("Moscow", "Nwot street", 5, 60))	"89995353566" Переход к следующему утверждению в теле правила (прямой ход)
Следующее утверждение	
banks(SURNAME, CITY, BANK, _, _)	Own a various and a strong and
Раскрывается правило banks(SURNAME, CITY, BANK, _, _) -> b	Определение отношения найдено Заносится в стек b
аПо функтору и списку аргументов ищется ппределение отношения	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",	Результат сравнения термов true
20, 2000)	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
	в оазе знании оольше 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.

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:

№	Сравниваемые термы; результат;	Дальнейшие действия:
шага	подстановка, если есть	прямой ход или откат
	find_by_car(SURNAME, CITY, PHONE,	Заносится в стек
	MARKA, COLOR, BANK)	find by car(SURNAME,
		CITY, PHONE, "Merc",
	find_by_car(SURNAME, CITY, PHONE, "Merc", "BLUE", BANK)	"BLUE", BANK)
	Раскрывается правило, то есть надо	Прямой ходб
	доказать каждое утверждение в теле	Заносится в стек
	правила (последовательно)	cars(SURNAME, CITY,
	cars(SURNAME, CITY, MARKA, COLOR,	"Merc", "BLUE",)
	PRICE)	7 7 2
	cars(SURNAME, CITY, "Merc", "BLUE",	
	cars(SURNAME, CITY, "Merc", "BLUE",	Определение отношения
		найдено
	По функтору и списку аргументов ищется	
	определение отношения	
	cars("Petrov", "Moscow", "TOYOTA",	
	"RED", 256000)	

Vнифик	ация cars(SURNAME, CITY,	Результат сравнения
_	"BLUE", _) u cars("Petrov",	термов false, прямой ход
	w", "TOYOTA", "RED", 256000)	
	ация cars(SURNAME, CITY,	Результат сравнения
"Merc",	"BLUE", _) и	термов false, прямой ход
cars("Kr	rot", "Moscow", "TOYOTA",	
	256000)	
Унифик	ация cars(SURNAME, CITY,	Результат сравнения
"Merc",	"BLUE", _)	термов false, прямой ход.
И		
`	etrov", "Kursk", "LADA", "GREEN",	
100000)		
X7 1	(CLIDNANCE CITY	D.
_	ация cars(SURNAME, CITY,	Результат сравнения
	"BLUE", _)	Tepmos true
И	rotovoli "Magaayyii "Magaii	SURNAME = "Krotova"
`	rotova", "Moscow", "Merc",	CITY = "Moscow" Переход к следующему
"BLUE"	', 100000)	
		утверждению в теле
		правила (прямой ход)
•	ощее утверждение info(SURNAME,	
PHONE	, address(CITY, _, _, _))	
info(SU	RNAME, PHONE, address(CITY, _,	Заносится в стек
,))		info("Krotova", PHONE,
_, _//		address("Moscow",, _, _, _))
=		
info("Kr	rotova", PHONE, address("Moscow",,	
_, _, _))	, individe a defense and individual	
_, _, _//		
Унифик	сация info("Krotova", PHONE,	Результат сравнения
	("Moscow",, _, _, _))	термов false, прямой ход.
,	Petrov", "88885353535",	
address("Moscow", "Saykin street", 10, 143))	
Унифик	сация info("Krotova", PHONE,	Результат сравнения
-	"Moscow", _, _, _, _))	термов false, прямой ход.
,	Petrov", "8111111111",	
address("Kursk", "Non street", 1, 10))	
Унифик	сация info("Krotova", PHONE,	Результат сравнения
	("Moscow",, _, _, _))	термов false, прямой ход.
		В базе знаний больше нет
,	Krot", "89995353566",	утверждений с заданным
address("Moscow", "Nwot street", 5, 60))	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

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

	правила (прямой ход)
Следующее утверждение	
info(SURNAME, PHONE, address(CITY, _, _, _, _,))	
Раскрывается правило info(SURNAME, PHONE, address(CITY, _, _, _)) -> info("Petrov", PHONE, address("Moscow", _, _, _))	Определение отношения найдено Заносится в стек info("Petrov", PHONE, address("Moscow", _, _, _))
По функтору и списку аргументов ищется определение отношения (с самого начала) 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, ,)	Определение отношения найдено Заносится в стек b а
По функтору и списку аргументов ищется определение отношения	n
Унификация banks("Petrov", "Moscow", BANK, _, _) и banks("Petrov", "Moscow", "Sberbank", 356, 50000) Унификация banks("Petrov", "Moscow", BANK, _, _)	Результат сравнения термов true Вывод результата, переход к следующей строке Результат сравнения термов false. Прямой ход
и banks("Petrov", "Kursk", "Tinkoff", 357, 9000) Унификация banks("Petrov", "Moscow",	Результат сравнения
BANK, _, _) и banks("Krot", "Moscow", "Rocket-bank",	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", "81111111111", 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 Rереход к следующему Утверждению в теле Жравила (прямой ход)
Следующее утверждение	
info(SURNAME, PHONE, address(CITY, _, _, _, _))	
Раскрывается правило info(SURNAME, PHONE, address(CITY, _, _, _)) ->	Определение отношения найдено Заносится в стек info("Krot", PHONE, address("Moscow", _, _, _,))
info("Krot", PHONE, address("Moscow", _, _, _, _))	
По функтору и списку аргументов ищется определение отношения (с самого начала)	

Унификация info("Krot", PHONE,	Результат сравнения
address("Moscow", _, _, _,))	термов false, прямой ход
И info("Petrov", "88885353535",	
address("Moscow", "Saykin street", 10,	
143))	
Унификация info("Krot", PHONE,	Результат сравнения
address("Moscow", _, _, _))	термов false, прямой ход
И info("Petrov", "8111111111", address("Kursk", "Non street", 1, 10))	
Унификация info("Krot", PHONE,	Результат сравнения
address("Moscow", _, _, _,))	термов true
И info("Krot", "89995353566",	Поромон и оночниому
address("Moscow", "Nwot street", 5, 60))	Переход к следующему утверждению в теле
	правила (прямой ход)
Следующее утверждение	привни (примен пед)
banks(SURNAME, CITY, BANK, ,)	
Раскрывается правило	Определение отношения
	найдено
banks(SURNAME, CITY, BANK, _, _)	Заносится в стек
->	ь
banks("Krot", "Moscow", BANK, _, _)	a
По функтору и списку аргументов ищется определение отношения	n
Унификация banks("Krot", "Moscow",	Результат сравнения
BANK, ,)	термов false, прямой ход
И banks("Petrov", "Moscow", "Sberbank",	
356, 50000)	
Унификация banks("Krot", "Moscow",	Результат сравнения
BANK, _, _)	термов false, прямой ход
И banks("Petrov", "Kursk", "Tinkoff", 357,	
9000) Унификация banks("Krot", "Moscow",	Результат сравнения
BANK, _, _)	термов true
И banks("Krot", "Moscow", "Rocket-bank",	
20, 2000)	Вывод результата, переход
	к следующей строке.
	В базе знаний больше нет
	утверждений с заданным
	именем. Откат, из стека
	Д
	В базе знаний больше нет
	утверждений с заданным
	именем. Откат, из стека
	достаётся info("Krot",
	Accidental mile (inter ,

		PHONE, address("Moscow", _, _, _))
Сле	едующее утверждение cars(SURNAME,	
"TC	ификация cars(SURNAME, CITY, DYOTA", "RED", _) и cars("Petrov", nrsk", "LADA", "GREEN", 100000)	Результат сравнения термов false, прямой ход
"TC	ификация cars(SURNAME, CITY, DYOTA", "RED", _) и cars("Krotova", oscow", "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", _)
Сте	ек пуст	Стек пуст, завершение цикла.

СПРАВОЧНИК – АВТОМОБИЛИ – БАНКИ

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

Раскрывается правило	Определение отношения найдено
т аскрывается правило	Заносится в стек
cars(SURNAME, CITY, MARKA, COLOR,	cars(«Petrov», «Moscow»,
	"TOYOTA", "RED",)
	, , , _,
cars(«Petrov», «Moscow», "TOYOTA",	
"RED", _)	
По функтору и списку аргументов ищется	
определение отношения	B
Унификация cars(«Petrov», «Moscow»,	Результат сравнения термов true
"TOYOTA", "RED", _)	Анонимные переменные не
И	связываются со значением
cars("Petrov", "Moscow", "TOYOTA",	Переход к следующему
"RED", 256000)	утверждению в теле правила
RED , 250000)	(прямой ход)
Следующее утверждение	
banks(SURNAME, CITY, BANK, _, _)	
Раскрывается правило	Определение отношения найдено
1 v.1-v.(CLIDNIAME CITY DANIZ	Заносится в стек
banks(SURNAME, CITY, BANK, _, _)	b
->	a
banks("Petrov", "Moscow", BANK, _, _)	n
По функтору и списку аргументов ищется определение отношения	k
Унификация banks("Petrov", "Moscow",	Результат сравнения термов true
BANK, _, _)	
H	Вывод результата, переход к
banks("Petrov", "Moscow", "Sberbank", 356, 50000)	следующей строке
Унификация banks("Petrov", "Moscow",	Результат сравнения термов false.
BANK, _, _)	В базе знаний больше нет
И	утверждений с заданным именем.
banks("Krot", "Moscow", "Rocket-bank",	Откат, из стека достаётся
20, 2000).	_,_)
Следующее утверждение	
Accept Manager UTOVOTA!	
cars(«Petrov», «Moscow», "TOYOTA",	
"RED", _)	
Унификация cars(«Petrov», «Moscow»,	Результат сравнения термов false.
"TOYOTA", "RED", _)	Прямой ход
M	
cars("Krot", "Moscow", "TOYOTA",	
"RED", 256000)	
Унификация cars(«Petrov», «Moscow»,	
"TOYOTA", "RED", _)	
И	

cars("Petrov", "Kursk", "LADA", "GREEN", 100000)	
Унификация cars(«Petrov», «Moscow», "TOYOTA", "RED", _) и cars("Krotova", "Moscow", "Merc", "BLUE", 100000)	Результат сравнения термов false. В базе знаний больше нет утверждений с заданным именем. Откат, из стека достаётся cars(«Petrov», «Moscow», "TOYOTA", "RED", _)
Следующее утверждение info(SURNAME, PHONE, address(CITY, _, ,))	
Унификация info(SURNAME, PHONE, address(CITY, _, _, _)) и info("Petrov", "8111111111", address("Kursk", "Non street", 1, 10))	Результат сравнения термов true S U Переход к следующему Переждению в теле правила
Станующое угрорудацие	Ж ірямой ход)
Следующее утверждение cars(SURNAME, CITY, "TOYOTA", "RED", _)	
Раскрывается правило cars(SURNAME, CITY, "TOYOTA", "RED", _) ->	Определение отношения найдено Заносится в стек cars(«Petrov», «Kursk», "TOYOTA", "RED", _)
cars(«Petrov», «Kursk», "TOYOTA", "RED", _) По функтору и списку аргументов ищется определение отношения	
Унификация cars(«Petrov», «Kursk», "TOYOTA", "RED", _) И cars("Petrov", "Moscow", "TOYOTA", "RED", 256000)	Результат сравнения термов false, прямой ход
Унификация cars(«Petrov», «Kursk», "TOYOTA", "RED", _)	Результат сравнения термов false, прямой ход
И cars("Krot", "Moscow", "TOYOTA", "RED", 256000)	
Унификация cars(«Petrov», «Kursk», "TOYOTA", "RED", _)	Результат сравнения термов false, прямой ход
И cars("Petrov", "Kursk", "LADA", "GREEN", 100000)	
Унификация cars(«Petrov», «Kursk», "TOYOTA", "RED", _)	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
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" РНОNE = "89995353566" СІТҮ = "Моѕсоw" Переход к следующему утверждению в теле правила (прямой ход)
Следующее утверждение	примон ходу
cars(SURNAME, CITY, "TOYOTA", "RED", _)	
Раскрывается правило cars(SURNAME, CITY, "TOYOTA", "RED", _)	Определение отношения найдено Заносится в стек cars(«Krot», «Moscow», "TOYOTA", "RED",)
-> cars(«Krot», «Moscow», "TOYOTA", "RED", _) По функтору и списку аргументов ищется определение отношения	, , <u>, _</u> ,
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, _, _)	7.7
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. О
Следующее утверждение cars(SURNAME, C	
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	Результирующая	Рабочее поле	Стек
унификац	ячейка		
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" Очистка поля	Стек пуст

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 арность (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 строитс япример?

Пусть $\theta = \{X_1 = t_1, X_2 = t_2, ..., X_n = t_n\}$ — 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.