



**«Московский государственный технический
университет имени Н.Э. Баумана»
(МГТУ им. Н.Э. Баумана)**

Факультет «Информатика и системы управления»
Кафедра «Программное обеспечение ЭВМ и информационные технологии»

Отчет по лабораторной работе №14
по курсу:
«Функциональное и Логическое программирование»

Студент группы ИУ7-63Б: Фурдик Н. О.
(Фамилия И.О.)

Преподаватель: Толпинская Н. Б., Строганов Ю. В.
(Фамилия И.О.)

Оглавление

Постановка задачи	2
Листинг программы	3
Описание порядка поиска ответов	4
Ответы на вопросы	9
Список литературы	13

Постановка задачи

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

- 1) «Телефонный справочник»: Фамилия, №тел, Адрес – структура (Город, Улица, №дома, №кв);
- 2) «Автомобили»: Фамилия владельца, Марка, Цвет, Стоимость, и др;
- 3) «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

Ниже представлен листинг программы:

```
domains

addr = ad(symbol town, symbol street, integer building, integer apartment)

predicates

person(symbol l_name, symbol ph_number, addr)
car(symbol owner_lname, symbol brand, symbol color, integer cost, integer year, symbol town)
bank_client(symbol client_lname, symbol bank_name, symbol type, integer amount_of_money, integer behaviour, symbol town)

get_stuff(symbol car_color, symbol car_brand, symbol lname, symbol town, symbol ph_num, symbol bank)

clauses

person("klyuge", "+7912571026", ad("volgograd", "varshavshkoe sh.", 52, 83)).
person("toporova", "+79165072034", ad("saint—petersburg", "varshavshkoe sh.", 100, 1151)).
person("zayceva", "+79138767153", ad("kasimov", "lermontovskiy pr.", 83, 578)).
person("mishkina", "+79165329056", ad("moscow", "antonova", 53, 1294)).
person("klyuge", "+79127675454", ad("moscow", "venevskaya", 85, 947)).
person("nikitin", "+7913336524", ad("kiev", "pushkina", 86, 973)).
person("nikitin", "+79146106984", ad("ryazan", "varshavshkoe sh.", 72, 27)).
person("toporov", "+7913733594", ad("saint—petersburg", "varshavshkoe sh.", 47, 1321)).
person("zaycev", "+7912486657", ad("volgograd", "lermontovskiy pr.", 86, 668)).
person("mishkin", "+79154045900", ad("kiev", "pushkina", 25, 278)).

car("zayceva", "geely", "white", 6409914, 2015, "kasimov").
car("nikitin", "mitsubishi", "green", 42502483, 2011, "ryazan").
car("klyuge", "mercedes", "blue", 19367655, 2013, "moscow").
car("toporov", "geely", "black", 59107666, 2011, "saint—petersburg").
car("toporova", "mitsubishi", "green", 12836628, 2018, "saint—petersburg").
car("mishkin", "mercedes", "black", 33637427, 2015, "kiev").
car("mishkin", "mitsubishi", "green", 3009291, 2011, "kiev").
car("zayceva", "ford", "blue", 36681813, 2012, "kasimov").
car("nikitin", "geely", "black", 58439837, 2011, "ryazan").
car("zaycev", "geely", "blue", 61712885, 2011, "kiev").

bank_client("mishkina", "tinkoff", "payment", 2439027, 8, "moscow").
bank_client("konev", "sberbank", "payment", 63838628, 4, "kasimov").
bank_client("toporova", "tinkoff", "savings", 60815175, 9, "saint—petersburg").
bank_client("klyuge", "tinkoff", "savings", 1768099, 9, "moscow").
bank_client("zayceva", "tinkoff", "savings", 53406609, 0, "kasimov").
bank_client("nikitin", "alphabank", "savings", 14555497, 8, "kiev").
bank_client("klyuge", "tinkoff", "credit", 34218309, -1, "volgograd").
bank_client("toporova", "sberbank", "credit", 37214889, 10, "saint—petersburg").
bank_client("toporova", "sberbank", "savings", 51895925, 10, "saint—petersburg").
bank_client("nikitin", "sberbank", "payment", 3318136, 8, "toporova").

get_stuff(Car_color, Car_brand, Lname, Town, Ph_num, Bank) :-car(Lname, Car_brand, Car_color, _, _, Town),
    person(Lname, Ph_num, ad(Town, _, _, _)),
```

```
bank_client(Lname, Bank, _, _, Town).
```

```
goal
```

```
%last name, town, ph number, bank
```

```
get_stuff(Car_color, Car_brand, Lname, Town, Ph_num, Bank), Car_brand = "mitsubishi", Car_color = "green", nl.
```

```
%
```

```
%Car_color=green, Car_brand=mitsubishi, Lname=toporova, Town=saint-petersburg, Ph_num=+79165072034, Bank=tinkoff
```

```
%Car_color=green, Car_brand=mitsubishi, Lname=toporova, Town=saint-petersburg, Ph_num=+79165072034, Bank=sberbank
```

```
%Car_color=green, Car_brand=mitsubishi, Lname=toporova, Town=saint-petersburg, Ph_num=+79165072034, Bank=sberbank
```

```
%3 Solutions
```

Листинг 1: Задания 1 и 2

Описание порядка поиска ответов

Ниже представлен вывод при трех ответах. Для меньшего числа ответов алгоритм будет выглядеть аналогично.

При разных порядках следования в БЗ процедур и знаний в них, множество и объем работ будут одинаковы, поскольку алгоритму поиска в любом случае придется дойти до конца БЗ.

Таблица 1: Задания 1 и 2. Вывод при трех ответах

№ ша- га	Сравниваемые термины; подстановка, если есть	Результат	Дальнейшие действия: прямой ход или откат (к чему приводит?)
1	Подстановка (связывание) Car_brand = 'mitsubishi', Car_color = 'green' в запрос get_stuff(Car_color, Car_brand, Lname, Town, Ph_num, Bank)	get_stuff('green', 'mitsubishi', Lname, Town, Ph_num, Bank)	прямой ход
2	Поиск совпадений и подстановка (связывание) 'mitsubishi', 'green' в соответствующее правило get_stuff(Car_color, Car_brand, Lname, Town, Ph_num, Bank)	get_stuff('green', 'mitsubishi', Lname, Town, Ph_num, Bank)	прямой ход
3	Подстановка Car_brand = 'mitsubishi', Car_color = 'green' в car(Lname, Car_brand, Car_color, _, _, Town)	Создается пример car(Lname, 'mitsubishi', 'green', _, _, Town)	прямой ход
4	Поиск car по примеру	car('zayceva', 'geely', 'white', 6409914, 2015, 'kasimov'), не подходит	прямой ход
5	Поиск car по примеру	...	прямой ход
6	Поиск car по примеру	car('toporova', 'mitsubishi', 'green', 12836628, 2018, 'saint-petersburg'), Lname = 'toporova', Town = 'saint-petersburg'	прямой ход
7	Подстановка в person(Lname, Ph_num, ad(Town, _, _, _)) Lname = 'toporova', Town = 'saint-petersburg'	Создается пример person('toporova', Ph_num, ad('saint-petersburg', _, _, _))	прямой ход
8	Поиск person по примеру	person('klyuge', '+7912571026', ad('volgograd', 'varshavshkoe sh.', 52, 83)), не подходит	откат
9	Поиск person по примеру	...	откат
10	Поиск person по примеру	person('toporova', '+79165072034', ad('saint-petersburg', 'varshavshkoe sh.', 100, 1151)), Lname = 'toporova' и Ph_num = '+79165072034'	прямой ход
11	Подстановка в bank_client(Lname, Bank, _, _, _, Town) Lname = 'toporova', Bank = Bank, Town = 'saint-petersburg'	Создается пример bank_client('toporova', Bank, _, _, _, 'saint-petersburg')	прямой ход

12	Поиск bank_client по примеру	bank_client('mishkina', 'tinkoff', 'payment', 2439027, 8, 'moscow')., не подходит	откат
13	Поиск bank_client по примеру	...	откат
14	Поиск bank_client по примеру	bank_client('toporova', 'tinkoff', 'savings', 60815175, 9, 'saint-petersburg'), Bank = 'tinkoff'	прямой ход
15	Общий пример найден, вывод	Car_color=green, Car_brand=mitsubishi, Lname=toporova, Town=saint-petersburg, Ph_num=+79165072034, Bank=tinkoff	откат, снова ищем по bank_client
16	Поиск bank_client по примеру	...	откат
17	Поиск bank_client по примеру	bank_client('toporova', 'sberbank', 'credit', 37214889, 10, 'saint-petersburg')., Bank = 'sberbank'	прямой ход
18	Общий пример найден, вывод	Car_color=green, Car_brand=mitsubishi, Lname=toporova, Town=saint-petersburg, Ph_num=+79165072034, Bank=sberbank	откат, снова ищем по bank_client
19	Поиск bank_client по примеру	...	откат
20	Поиск bank_client по примеру	bank_client('toporova', 'sberbank', 'savings', 51895925, 10, 'saint-petersburg'), Bank = 'sberbank'	прямой ход
21	Общий пример найден, вывод	Car_color=green, Car_brand=mitsubishi, Lname=toporova, Town=saint-petersburg, Ph_num=+79165072034, Bank=sberbank	откат, снова ищем по bank_client и ничего не находим. Откатываемся до person
22	Поиск person по примеру	...	ничего не находим. Откатываемся до car
23	Поиск car по примеру	...	ничего не находим, однако мы уже дошли до конца БЗ. Конец

Таблица 2: Задание 3. Алгоритм унификации

шаг унификации	результатирующая ячейка	рабочее поле	пункт алгоритма	стек
0				get_stuff('green', 'mitsubishi', Lname, Town, Ph_num, Bank) = get_stuff(Car_color, Car_brand, Lname, Town, Ph_num, Bank)
1		get_stuff('green', 'mitsubishi', Lname, Town, Ph_num, Bank) = get_stuff(Car_color, Car_brand, Lname, Town, Ph_num, Bank)		Car_color = 'green', Car_brand = 'mitsubishi', Lname = Lname, Ph_num = Ph_num, Bank = Bank
2	Car_color = 'green', Car_brand = 'mitsubishi'	Car_color = 'green', Car_brand = 'mitsubishi', Lname = Lname, Ph_num = Ph_num, Bank = Bank		Lname = Lname, Town = Town, Ph_num = Ph_num, Bank = Bank
3	Car_color = 'green', Car_brand = 'mitsubishi'	Lname = Lname, Town = Town		car(Lname, 'mitsubishi', 'green', _, _, Town); Ph_num = Ph_num, Bank = Bank
4	Car_color = 'green', Car_brand = 'mitsubishi', Lname = Lname, Town = Town	car(Lname, 'mitsubishi', 'green', _, _, Town)		car('toporova', 'mitsubishi', 'green', 12836628, 2018, 'saint-petersburg'); Ph_num = Ph_num, Bank = Bank
5	Car_color = 'green', Car_brand = 'mitsubishi', Lname = 'toporova', Town = 'saint-petersburg'	car('toporova', 'mitsubishi', 'green', 12836628, 2018, 'saint-petersburg'), Lname = 'toporova', Town = 'saint-petersburg'		Ph_num = Ph_num, Bank = Bank
6	Car_color = 'green', Car_brand = 'mitsubishi', Lname = 'toporova', Town = 'saint-petersburg'	Ph_num = Ph_num		person('toporova', Ph_num, ad('saint-petersburg', _, _, _)); Bank = Bank
7	Car_color = 'green', Car_brand = 'mitsubishi', Lname = 'toporova', Town = 'saint-petersburg', Ph_num = Ph_num	person('toporova', Ph_num, ad('saint-petersburg', _, _, _))		person('toporova', '+79165072034', ad('saint-petersburg', 'varshavshkoe sh.', 100, 1151)); Bank = Bank

8	Car_color = 'green', Car_brand = 'mitsubishi', Lname = 'toporova', Town = 'saint- petersburg', Ph_num = '+79165072034'	person('toporova', '+79165072034', ad('saint- petersburg', 'varshavshkoe sh.', 100, 1151)).		Bank = Bank
9	Car_color = 'green', Car_brand = 'mitsubishi', Lname = 'toporova', Town = 'saint- petersburg', Ph_num = '+79165072034'	Bank = Bank		bank_client(Lname, Bank, __, __, __, Town)
10	Car_color = 'green', Car_brand = 'mitsubishi', Lname = 'toporova', Town = 'saint- petersburg', Ph_num = '+79165072034', Bank = Bank	bank_client('toporova', Bank, __, __, __, 'saint-petersburg')		bank_client('toporova', 'tinkoff', 'savings', 60815175, 9, 'saint-petersburg')
11	Car_color = 'green', Car_brand = 'mitsubishi', Lname = 'toporova', Town = 'saint- petersburg', Ph_num = '+79165072034', Bank = 'tinkoff'	bank_client('toporova', 'tinkoff', 'savings', 60815175, 9, 'saint-petersburg'), Bank = 'tinkoff'		
Вывод:	подстановка	Т.к. стек пуст – успех и в рез. ячейке - подстановка		

Ответы на вопросы

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

Знание формулируется в правой части правила. Оно содержит в себе некую информацию, которая является истиной.

- 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. Процедура описывает одно отношение, которое нельзя сформулировать одним предложением.

- 3) **Сколько в БЗ текущего задания процедур?**

В БЗ текущего задания четыре процедуры.

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

Терм B называется примером терма A , если существует такая подстановка такая, что $B = A\theta$.

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

Пример. Имеется БЗ, в которой есть факт $g(1, a, b)$. Базе Знаний подаётся запрос $g(x, y, z)$. Система находит общий пример и только тогда утверждение из Базы Знаний может считаться доказательством истинности.

- 5) **Что такое наиболее общий пример?**

Терм S называется более общим чем терм T , если T является примером S , а S не является примером T . S называется наиболее общим примером T_1 и T_2 , если S такой их общий пример, который является более общим по отношению к любому другому их примеру.

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

Алгоритм встроен в систему и для своей работы использует три области

памяти:

- стек: хранит равенства, унификацию которых надо доказать;
- рабочее поле: равенство вытащенное из стека, которое сейчас доказываем;
- результирующая ячейка памяти: накапливает наиболее общий унификатор;
- флаг - переменная 'неудача': равна 1, если возникла тупиковая ситуация иначе 0.

шаг: начало: Сопоставляем $T_1=T_2$

занести в стек $T_1=T_2$

положить неудача=0

пока стек не пуст – **ЦИКЛ:**

- считать из стека в рабочую область очередное равенство $S=T$
 - обработать считанное по правилам:
 - если S и T несовпадающие константы, то неудача=1, и выход из цикла
 - если одинаковые константы то следующий шаг цикла
 - если S переменная и T терм содержащий S , то неудача=1, и выход из цикла
 - если S переменная и T терм НЕ содержащий S , то отыскать в стеке и в результирующей ячейке все вхождения S и заменить на T . Добавить в результирующую ячейку равенство $S=T$. Следующий шаг цикла
 - если S и T составные термы с разными функторами или разными арностями, то неудача=1, выход из цикла
 - если S и T составные термы с одинаковыми функторами и арностью:
 $S = f(s_1s_2..s_m); T = f(t_1t_2...t_m)$, то занести в стек равенство $S_1 = T_1, S_2 = T_2...S_m = T_m$.
 - очистить рабочее поле
- **конец цикла**

если неудача = 1 то унификация невозможна

если неудача = 0 то унификация успешна, результирующая ячейка содержит наиболее общий унификатор

конец.

Таким образом, 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?

Для обоснования истинности вопроса, не всегда бывает возможным подобрать нужное знание в принципе, или не всегда это можно выполнить однозначно. Кроме этого, не очевиден принцип и алгоритм (последовательность перебора) выбора знания для использования. Часто при выборе знания и человеку и системе необходимо оценить содержательный смысл – семантику утверждения, что на формальном уровне однозначно выполнить сложно. Следует заметить, что недетерминизм поиска ответа – это базовый принцип работы естественного, искусственного интеллекта и системы Prolog. Таким образом, в сложных ситуациях – при высокой степени неопределенности, и человек и система Prolog совершают ошибки. Для того, что бы такая ошибка выбора знания для системы Prolog была не критична, в систему встроен механизм отката (на момент последнего сделанного выбора) и отмена принятого решения (выполненной конкретизации).

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

Переменные начинаются с заглавной буквы или с символа подчёркивания и могут быть анонимными или именованными (анонимная переменная обозначается нижним подчеркиванием). Говорят, что переменная может быть связана с некоторым значением или оставаться независимой. Именованные переменные уникальны в рамках одного предложения. Анонимная переменная уникальна всегда. Переменные предназначены для передачи значений «во времени и в пространстве».

```
bank_client(Lname, Bank, __, __, __, Town)
```

Здесь использовались именные переменные `Lname`, `Bank`, `Town` поскольку нам были важны значения, которые принимают соответствующие термы. Поскольку оставшиеся поля никак не влияли на результат, вместо них использовались анонимные переменные.

Литература

1. Толпинская Н.Б. - Курс лекций по "Функциональному и Логическому программированию"[Текст], Москва 2019 год.
2. Анатолий Адаменко, Андрей Кучуков. Логическое программирование и Visual Prolog (с CD). — СПб.: БХВ-Петербург, 2003. — 990 с. — ISBN 5-94157-156-9.