



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 предмету: «Функциональное и логическое
программирование»*

Студент: Лаврова А. А.,
Группа: ИУ7-65Б
Преподаватель: Толпинская Н. Б.

Москва, 2020 г.

Практическая часть

Задание

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

- **«Телефонный справочник»:** Фамилия, Номер, Адрес – структура (Город, Улица, Номер дома, Номер 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!!!

Листинг программы:

```
domains
    surname = symbol.
    telephone = symbol.
    city = symbol.
    street = symbol.
    house = integer.
    address = address(city, street, house).

    model = symbol.
    color = symbol.
    price = integer.

    name_of_bank = symbol.
    account = integer.
    amount = integer.

predicates
    phonebook(surname, telephone, address).
    automobile(surname, model, color, price).
    bank(surname, name_of_bank, account, amount).

    find_by_number(telephone, surname, model, price).
    find_by_surname_city(surname, city, street, name_of_bank, telephone).
    search(model, color, surname, city, telephone, name_of_bank)

clauses
    phonebook("Ivanov", "8111111111", address("Moscow", "Street1", 1)).
    phonebook("Petrov", "8222222222", address("Moscow", "Street2", 2)).
    phonebook("Sidorov", "8333333333", address("Kazan", "Street3", 3)).
    phonebook("Vasechkin", "8444444444", address("Chelyabinsk", "Street4", 4)).
    phonebook("Lolaev", "8555555555", address("Zlatoust", "Street5", 5)).
    phonebook("Lolaev", "8666666666", address("Ekaterinburg", "Street6", 6)).

    automobile("Ivanov", "Kia", "red", 1550000).
    automobile("Ivanov", "Lada", "grey", 1000000).
    automobile("Vasechkin", "Land Rover", "black", 5000000).
    automobile("Sidorov", "Honda", "white", 2000000).
```

```

bank("Lolaev", "HomeCreditBank", 100, 100).
bank("Ivanov", "Sberbank", 200, 200).
bank("Vasechkin", "VTB", 300, 300).
bank("Lolaev", "Pochta Bank", 400, 400).

find_by_number(Telephone, Surname, Model, Price) :-
phonebook(Surname, Telephone, _), automobile(Surname, Model, _, Price).

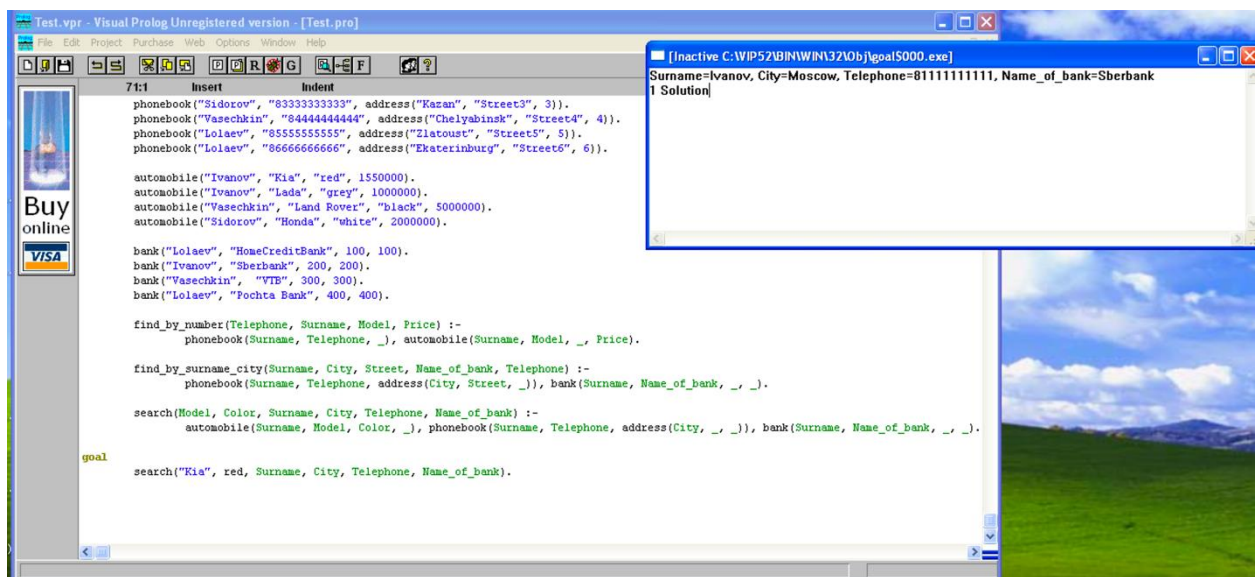
find_by_surname_city(Surname, City, Street, Name_of_bank, Telephone) :-
phonebook(Surname, Telephone, address(City, Street, _)), bank(Surname, Name_of_bank, _, _).

search(Model, Color, Surname, City, Telephone, Name_of_bank) :-
automobile(Surname, Model, Color, _), phonebook(Surname, Telephone, address(City, _, _)),
bank(Surname, Name_of_bank, _, _).

goal
    search("Kia", red, Surname, City, Telephone, Name_of_bank).

```

Пример работы программы:



Вопрос: Кто владеет красным Kia? (какая у него фамилия, в каком городе живет, какой у него телефон, в каком банке хранит сбережения)

Теоретическая часть

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

Факт (т. е. знание) – частный случай правила (*Правила – предложение вида A :*

– B_1, \dots, B_n , то, что находится слева от знака $:-$ является заголовком правила, то, что справа –

телом правила.), у него нет тела. То есть знание сформулировано в заголовке правила.

2. Что такое процедура?

Процедура – совокупность правил, заголовки которых одинаковы.

3. Сколько в БЗ текущего задания процедур?

4

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

Пример терма – это результат подстановки некоторых конкретных значений в предикат, частный случай предиката. Пример терма строится после того, как задан вопрос и хранится до окончания работы программы. Примеры строятся во время алгоритма унификации

5. Что такое наиболее общий пример?

S называется наиболее общим примером T1 и T2, 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.

Унификация - попарное сопоставление термов и попытка построить для них общий пример. Происходит двунаправленная передача параметров процедурам.

Двунаправленная передача параметров – это передача этих самых параметров из/в программу для дальнейшего использования (например, значение параметра, который нас интересует).

7. В каком случае запускается механизм отката?

Механизм отката запускается в случае, когда унификация завершается тупиковой ситуацией с откатом к предыдущему шагу.

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

Переменные – это терм, бывают:

именованные – обозначаются комбинацией символов латинского алфавита, цифр и символа подчеркивания, начинающейся с прописной буквы или символа подчеркивания;
анонимные – символ подчеркивания.

№1

search("Kia", red, Surname, City, Telephone, Name_of_bank).

№ шага	Сравниваемые термы; результат; подстановка, если есть	Дальнейшие действия: прямой ход или откат (к чему приводит?)
1	search("Kia", red, Surname, City, Telephone, Name_of_bank) и search(Model, Color, Surname, City, Telephone, Name_of_bank) Связываются через Model = "Kia" и color = red	search убирается из стека automobile(Surname, Model, Color, _) заносится в стек phonebook(Surname, Telephone, address(City, _, _)) заносится в стек bank(Surname, Name_of_bank, _, _) заносится в стек
2	automobile(Surname, "Kia", red, _) и automobile("Ivanov", "Kia", "red", 1550000). Связываются через Surname = "Ivanov"	automobile убирается из стека
3	phonebook("Ivanov", telephone, address(City, Street, house)) и phonebook("Ivanov", "8111111111", address("Moscow", "Street1", 1)) Связывается через Surname = "Ivanov", Telephone = "8111111111", City = "Moscow"	phonebook убирается из стека

4	<p>bank("Ivanov", name_of_bank, account, amount).</p> <p>и</p> <p>bank("Lolaev", "HomeCreditBank", 100, 100)</p> <p>Не унифицируется</p>	
5	<p>bank("Ivanov", name_of_bank, account, amount).</p> <p>и</p> <p>bank("Ivanov", "Sberbank", 200, 200)</p> <p>Связывается через Name_of_bank = "Sberbank"</p>	<p>bank убирается из стека</p> <p>Выводится: Ivanov, Moscow, "8111111111", "Sberbank"</p> <p>Name_of_bank развязывается</p>
6	... (все остальные факты bank не унифицируются)	
7		<p>Откат</p> <p>Все факты bank пройдены, в стек заносится последний вопрос</p> <p>phonebook(Surname, Telephone, address)</p> <p>Surname развязывается</p>
8	... (все остальные факты phonebook не унифицируются)	
9		<p>Откат</p> <p>Все факты phonebook пройдены, в стек заносится последний вопрос</p> <p>automobile</p> <p>Telephone развязывается</p>
10	... (все остальные факты automobile не унифицируются)	
11		<p>Откат</p> <p>Все факты automobile пройдены, в стек заносится последний вопрос</p> <p>automobile</p> <p>Model и color развязывается</p> <p>Система завершает работу</p>

№2

Если мы предполагаем, что система проходит всю базу знания от начала и до конца (так, про крайней мере, написано в методичке Натальи Борисовны), то количество сравнений от этого не изменится, так как для каждого вопроса мы попытаемся унифицировать элементы из базы знаний.

Может быть мы можем изменить количество сравнений, если система знает, где находится конец процедуры, и дальше конца области она не пройдет. Следовательно, от того, где находится процедура, зависит количество сравнений.

№3

search("Kia", red, Surname, City, Telephone, Name_of_bank).

шаг унификации	результатирующая ячейка	рабочее поле	пункт алгоритма	стек
0			1	search("Kia", red, Surname, City, Telephone, Name_of_bank). = search(Model, Color, Surname, City, Telephone, Name_of_bank)
1		search("Kia", red, Surname, City, Telephone, Name_of_bank). = search(Model, Color, Surname, City, Telephone, Name_of_bank) :- automobile(Surname, Model, Color, _), phonebook(Surname, Telephone, address(City, _, _)), bank(Surname, Name_of_bank, _, _).		Model = Kia Color = red automobile(Surname, Model, Color, _) phonebook(Surname, Telephone, address(City, _, _)) bank(Surname, Name_of_bank, _, _).
2	Model = Kia	Model = Kia ←	г	Color = red automobile(Surname, Model, Color, _)

				phonebook(Surname, Telephone, address(City, _, _)) bank(Surname, Name_of_bank, _, _).
3	Model = Kia Color = red	Color = red ←	г	automobile(Surname, Model, Color, _) phonebook(Surname, Telephone, address(City, _, _)) bank(Surname, Name_of_bank, _, _).
4	Model = Kia Color = red	automobile(Surname, "Kia", "red", _) automobile("Ivanov", "Kia", "red", 1550000). Унификация прошла успешно →	е	Surname = Ivanov Kia= Kia red = red phonebook(Surname, Telephone, address(City, _, _)) bank(Surname, Name_of_bank, _, _).
5	Model = Kia Color = red Surname = Ivanov	Surname = Ivanov ←	г	Model = Kia red = red phonebook(Surname, Telephone, address(City, _, _)) bank(Surname, Name_of_bank, _, _).
6	Model = Kia Color = red Surname = Ivanov	Kia = Kia	б	red = red phonebook(Surname, Telephone, address(City, _, _)) bank(Surname, Name_of_bank, _, _).
7	Model = Kia Color = red Surname = Ivanov	red = red	б	phonebook(Surname, Telephone, address(City, _, _)) bank(Surname, Name_of_bank, _, _).
8	Model = Kia Color = red	phonebook(Ivanov, Telephone, address(City, _, _))		Ivanov = Ivanov Telephone = "81111111111"

	Surname = Ivanov	phonebook("Ivanov", "8111111111", address("Moscow", "Street1", 1)). Унификация прошла успешно →		City = Moscow bank(Surname, Name_of_bank, _, _).
9	Model = Kia Color = red Surname = Ivanov	Ivanov = Ivanov	б	Telephone = "8111111111" City = Moscow bank(Surname, Name_of_bank, _, _).
10	Model = Kia Color = red Surname = Ivanov Telephone = "8111111111"	Telephone = "8111111111" ←	г	City = Moscow bank(Surname, Name_of_bank, _, _).
11	Model = Kia Color = red Surname = Ivanov Telephone = "8111111111" City = Moscow	City = Moscow ←	г	bank(Surname, Name_of_bank, _, _).
12	Model = Kia Color = red Surname = Ivanov Telephone = "8111111111" City = Moscow	bank(Ivanov, Name_of_bank, _, _). bank("Lolaev", "HomeCreditBank", 100, 100). Унификация не произошла!		bank(Surname, Name_of_bank, _, _).
13	Model = Kia Color = red Surname = Ivanov Telephone = "8111111111" City = Moscow	bank(Ivanov, Name_of_bank, _, _). bank("Ivanov", "Sberbank", 200, 200). Унификация произошла успешно! →		Ivanov = Ivanov Name_of_bank = Sberbank
14	Model = Kia Color = red	Ivanov = Ivanov	б	Name_of_bank = Sberbank

	Surname = Ivanov Telephone = "81111111111" City = Moscow			
15	Model = Kia Color = red Surname = Ivanov Telephone = "81111111111" City = Moscow	Name_of_bank = Sberbank ←	г	
16		Откат к шагу 13, перебираем факты bank		
17		bank(Ivanov, Name_of_bank, _, _). bank("Vasechkin", "VTB", 300, 300). Унификация не произошла!		
18		bank(Ivanov, Name_of_bank, _, _). bank("Lolaev", "Pochta Bank", 400, 400). Унификация не произошла!		
19		Откат к шагу 8, перебираем факты phonebook		bank(Surname, Name_of_bank, _, _).
20- 24		phonebook(Ivanov, Telephone, address(City, _, _)) ... Унификация не произошла!		bank(Surname, Name_of_bank, _, _).
25		Откат к шагу 1, перебираем факты automobile		phonebook(Surname, Telephone, address(City, _, _)) bank(Surname, Name_of_bank, _, _).
26- 30		search("Kia", red, Surname, City, Telephone, Name_of_bank). ... Унификация не произошла!		phonebook(Surname, Telephone, address(City, _, _)) bank(Surname, Name_of_bank, _, _).