|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **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»

**ОТЧЕТ**

*к лабораторной работе №14*

*По курсу: «Функциональное и логическое программирование»*

Студент ИУ7-62Б

Блохин Д.М.

Преподаватели

Толпинская Н.Б.

Строганов Ю.В.

*Москва, 2021 г.*

**Задание:**

Составить базу знаний, объединив в ней информацию – знания:

* **«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. Владельцев может быть несколько (не более 3-х), один и ни одного.

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!!!

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.

**Вопросы:**

**В какой части правила сформулировано знание?**В заголовке правила находится знание о предметной области, а в теле правила содержится условия истинности этого знания.

**Что такое процедура?**Процедурой называется совокупность правил, заголовки которых имеют одно и то же имя и одну и ту же арность, описывают одно определенное отношение.

**Сколько в БЗ текущего задания процедур?**

4

**Что такое пример терма, это частный случай терма, пример?**Терм В называется примером терма А, если существует такая подстановка Θ, что В=АΘ, где АΘ – результат применения подстановки к терму.   
**Как и когда строится?**

Примеры строятся во время алгоритма унификации.

**Что такое общий пример?**Терм С называется общим примером термов А и В, если существуют такие подстановки Θ1 и Θ2, что С = А Θ1 и С=В Θ2.

**Назначение и результат работы алгоритма унификации.**

Унификация – механизм логического вывода. Результат – подстановка.

**Что значит двунаправленная передача параметров при работе алгоритма унификации, поясните на примере одного из случаев пункта 3.**info\_by\_car(Car\_brand, Car\_color, Surname, City, TeleTelephone, Bank\_name) :- car\_owner(Surname, about\_car(Car\_brand, Car\_color, \_, \_)),

Telephone\_record(Surname, TeleTelephone, address(City, \_, \_, \_)), bank\_owner(Surname, dep\_info(Bank\_name, \_, \_)).  
Если в info\_by\_car задать только TeleTelephone, то Surname будет найдено в Telephone\_record и использовано в car\_owner и bank\_owner.

**В каком случае запускается механизм отката?**

В случае тупиковой ситуации при попытке сопоставить два терма.

**Виды и назначение переменных в Prolog. Примеры из задания.**

* Именованные – есть имя – комбинация символов (Car\_color).
* Анонимные – нет имени – символ подчеркивания ( \_ ).

**Почему использованы те или другие переменные (примеры из задания)?**

Анонимные переменные используются, если не нужно связывать ее со значением (не нужно заносить в результирующую ячейку, например), А именованные, если нужно связывать и заносить в результирующую ячейку.

**Текст программы:**

domains

surname, teleTelephone, city, street = symbol.

home\_number, flat\_number = integer.

address = address(city,

street,

home\_number,

flat\_number).

car\_brand, car\_color = symbol.

car\_price, car\_year = integer.

about\_car = about\_car(car\_brand,

car\_color,

car\_price,

car\_year).

bank\_name = symbol.

bank\_account, bank\_amount = integer.

dep\_info = dep\_info(bank\_name,

bank\_account,

bank\_amount).

predicates

Telephone\_record(surname, teleTelephone, address).

car\_owner(surname, about\_car).

bank\_owner(surname, dep\_info).

info\_by\_car(car\_brand, car\_color, surname, city, teleTelephone, bank\_name).

clauses

Telephone\_record("Gerasimenko", "85886893800", address("Moscow", "Auf", 17, 33)).

Telephone\_record("Kryat", "85886893892", address("Moscow", "Kolokol", 12, 33)).

Telephone\_record("Popov", "89932110022", address("Minsk", "Moskovskaya", 94, 11)).

Telephone\_record("Blohin", "89265291378", address("Moscow", "Varsh", 52, 9)).

Telephone\_record("Blohin", "89255223228", address("Moscow", "Varsh", 52, 9)).

Telephone\_record("Blohin", "89115291378", address("Saratov", "Moskovskaya", 5, 2)).

car\_owner("Paklin",

about\_car("Cooper", "Red", 2000001, 2012)).

car\_owner("Kryat",

about\_car("Mitsubishi", "Grey", 289731, 2006)).

car\_owner("Blohin",

about\_car("Opel", "Grey", 290000, 2007)).

car\_owner("Blohin",

about\_car("Audi", "Red", 2900000, 2015)).

bank\_owner("Kolosov",

dep\_info("Sber", 1, 100000)).

bank\_owner("Kolosov",

dep\_info("VTB", 5, 200000)).

bank\_owner("Kryat",

dep\_info("VTB", 1, 210000)).

bank\_owner("Popov",

dep\_info("Rocket", 2, 10000000)).

bank\_owner("Blohin",

dep\_info("Gazprom", 4, 400000)).

info\_by\_car(Car\_brand, Car\_color, Surname, City, TeleTelephone, Bank\_name) :- car\_owner(Surname, about\_car(Car\_brand, Car\_color, \_, \_)),

Telephone\_record(Surname, TeleTelephone, address(City, \_, \_, \_)), bank\_owner(Surname, dep\_info(Bank\_name, \_, \_)).

goal

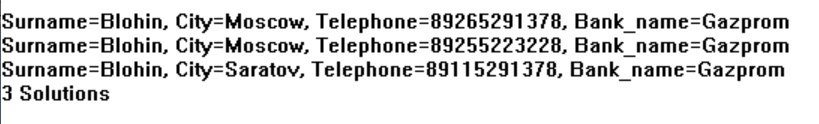
info\_by\_car("Audi", "Red", Surname, City, TeleTelephone, Bank\_name).

%info\_by\_car("Mitsubishi", "Grey", Surname, City, TeleTelephone, Bank\_name).

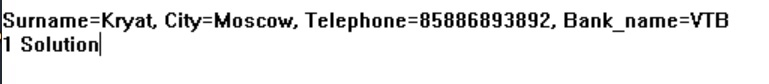
%info\_by\_car("Mitsubishi", "White", Surname, City, TeleTelephone, Bank\_name).

**Результаты работы программы:**

some:



one:



none:



**Задание №1** порядок поиска ответа на вопрос**:**

**Для нескольких владельцев:**

Вопрос: info\_by\_car("Audi", “Red”, Surname, City, TeleTelephone, Bank\_name)

|  |  |  |
| --- | --- | --- |
| № шага | Сравниваемые термы; результат; подстановка, если есть | Дальнейшие действия: прямой ход или откат (к чему приводит?) |
| 1 | Сравнение терма  info\_by\_car("Audi", "Red", Surname, City, TeleTelephone, Bank\_name)  С каждым термом из набора  Telephone\_record(…)  car\_owner(…)  bank\_owner(…) | После каждого сравнения переход к анализу следующего предложения. |
| 2 | Сравниваемые термы:  T1 = info\_by\_car(Car\_brand, Car\_color, Surname, City, TeleTelephone, Bank\_name)  T2 =  info\_by\_car("Audi", "Red", Surname, City, TeleTelephone, Bank\_name)  Подстановка: Car\_brand = "Audi", Car\_color = “Red” | Прямой ход |
| 3 | Сравниваем  car\_owner(Surname, about\_car(“Audi”, “Red”, \_, \_)) с каждым термом из набора  Telephone\_record(…) | После каждого сравнения переход к анализу следующего предложения. |
| 4 | Сравниваем термы  T1 = car\_owner("Paklin",  about\_car("Cooper", "Red", 2000001, 2012))  и  T2 =  car\_owner(Surname, about\_car(“Audi”, “Red”, \_, \_)) | Переход к следующему предложению |
| 5 | Сравниваем термы  T1 = car\_owner("Kryat",  about\_car("Mitsubishi", "Grey", 289731, 2006))  и  T2 =  car\_owner(Surname, about\_car(“Audi”, “Red”, \_, \_)) | Переход к следующему предложению |
| 6 | Сравниваем термы  T1 = car\_owner("Blohin",  about\_car("Opel", "Grey", 290000, 2007))  и  T2 =  car\_owner(Surname, about\_car(“Audi”, “Red”, \_, \_)) | Переход к следующему предложению |
| 7 | Сравниваем термы  T1 = car\_owner("Blohin",  about\_car("Opel", "Grey", 290000, 2007))  и  T2 =  car\_owner(Surname, about\_car(“Audi”, “Red”, \_, \_))  Подстановка Surname = “Blohin” | Прямой ход |
| 8 | Сравниваем  Telephone\_record(“Blohin”, TeleTelephone, address(City, \_, \_, \_)) с каждым термом из набора  Telephone\_record(…)  пока не будет сравниваться c Telephone\_record("Blohin", "89265291378", address("Moscow", "Varsh", 52, 9)). | После каждого сравнения переход к анализу следующего предложения. |
| 9 | Сравниваем термы  T1 = Telephone\_record("Blohin", "89265291378", address("Moscow", "Varsh", 52, 9))  и  T2 =  Telephone\_record(“Blohin”, TeleTelephone, address(City, \_, \_, \_))  Подстановка TeleTelephone = "89265291378", City = “Moscow” | Прямой ход |
| 10 | Сравниваем  bank\_owner(“Blohin”, dep\_info(Bank\_name, \_, \_)) с каждым термом из набора  Telephone\_record(…)  car\_owner(…) | После каждого сравнения переход к анализу следующего предложения. |
| 14 | Сравниваем термы  T1 = bank\_owner("Kolosov",  dep\_info("Sber", 1, 100000))  и  T2 =  bank\_owner(“Blohin”, dep\_info(Bank\_name, \_, \_)) | Переход к следующему предложению |
| 15 | Сравниваем термы  T1 = bank\_owner("Kolosov",  dep\_info("VTB", 5, 200000))  и  T2 =  bank\_owner(“Blohin”, dep\_info(Bank\_name, \_, \_)) | Переход к следующему предложению |
| 16 | Сравниваем термы  T1 = bank\_owner("Popov",  dep\_info("Rocket", 2, 10000000))  и  T2 =  bank\_owner(“Blohin”, dep\_info(Bank\_name, \_, \_)) | Переход к следующему предложению |
| 17 | Сравниваем термы  T1 = bank\_owner("Blohin",  dep\_info("Gazprom", 4, 400000))  и  T2 =  bank\_owner(“Blohin”, dep\_info(Bank\_name, \_, \_))  Подстановка Bank\_name = “Gazprom” | **Решение найдено**  **Output:Surname = “Blohin”**  **City = “Moscow”**  **TeleTelephone = “89265291378”**  **Bank\_name = “Gazprom”** |
| 18 | Сравниваем термы  T1 = info\_by\_car(Car\_brand, Car\_color, Surname, City, TeleTelephone, Bank\_name)и  T2 =  bank\_owner(“Blohin”, dep\_info(Bank\_name, \_, \_)) | Возврат к шагу 9 |
| 19 | Сравниваем термы  T1 = Telephone\_record("Blohin", "89255223228", address("Moscow", "Varsh", 52, 9))  и  T2 =  Telephone\_record(“Blohin”, TeleTelephone, address(City, \_, \_, \_))  Подстановка TeleTelephone = "89255223228", City = “Moscow” | Прямой ход |
| 20 | ------ | Повторение шагов 10-16 |
| 21 | Сравниваем термы  T1 = bank\_owner("Blohin",  dep\_info("Gazprom", 4, 400000))  и  T2 =  bank\_owner(“Blohin”, dep\_info(Bank\_name, \_, \_))  Подстановка Bank\_name = “Gazprom” | **Решение найдено**  **Output:Surname = “Blohin”**  **City = “Moscow”**  **TeleTelephone = “**89255223228**”**  **Bank\_name = “Gazprom”** |
| 22 | Сравниваем термы  T1 = info\_by\_car(Car\_brand, Car\_color, Surname, City, TeleTelephone, Bank\_name)и  T2 =  bank\_owner(“Blohin”, dep\_info(Bank\_name, \_, \_)) | Возврат к шагу 19 |
| 23 | Сравниваем термы  T1 = Telephone\_record("Blohin", "89115291378", address("Saratov", "Moskovskaya", 5, 2))  и  T2 =  Telephone\_record(“Blohin”, TeleTelephone, address(City, \_, \_, \_))  Подстановка TeleTelephone = "89115291378", City = “Saratov” | Прямой ход |
| 24 | -------- | Повторение шагов 10-16 |
| 25 | T1 = bank\_owner("Blohin",  dep\_info("Gazprom", 4, 400000))  и  T2 =  bank\_owner(“Blohin”, dep\_info(Bank\_name, \_, \_)) | **Решение найдено**  **Output:Surname = “Blohin”**  **City = “Saratov”**  **TeleTelephone = “**89115291378**”**  **Bank\_name = “Gazprom”** |
| 26 | Сравниваем термы  T1 = info\_by\_car(Car\_brand, Car\_color, Surname, City, TeleTelephone, Bank\_name)и  T2 =  bank\_owner(“Blohin”, dep\_info(Bank\_name, \_, \_)) | Возвращение к шагу 23 |
| 27 | Сравнение  термы  Telephone\_record(“Blohin”, TeleTelephone, address(City, \_, \_, \_)) с каждым оставшимся термом | После каждого сравнения переход к анализу следующего предложения и возвращение к шагу 3 |
| 28 | Сравнение  термы  car\_owner(Surname, about\_car(“Audi”, “Red”, \_, \_)) с каждым оставшимся термом | После каждого сравнения переход к анализу следующего предложения и возвращение к шагу 2 |
| 29 | -------- | Откат. Все предложения проанализированы. Решение есть. Работа завершается |

**Задание №2** порядок поиска ответа на вопрос при разных порядках следования**:**

Если считать, что система знает где в базе знаний начинается область тех или иных процедур и не делает лишних проверок, сравнивая функторы и арность с самого начала базы знаний, то, если поменять расположение процедур местами, количество сравнений не изменится.

Если же считать, что система полностью проходит базу знаний от начала до конца, то количество сравнений также не изменится. Т.к. для каждого вопроса будет попытка унификации с каждым элементом базы знаний.

Количество сравнений изменится только лишь в том случае, если считать, что система не знает где начало процедуры и дальше конца области этой процедуры не пойдет. В таком случае, расположение процедуры, с которой требуется наибольшее число попыток унификации, в начале БЗ, уменьшит количество шагов.

Количество сравнений может изменится от порядка занесения в стек (см. Рис. 1, 2).

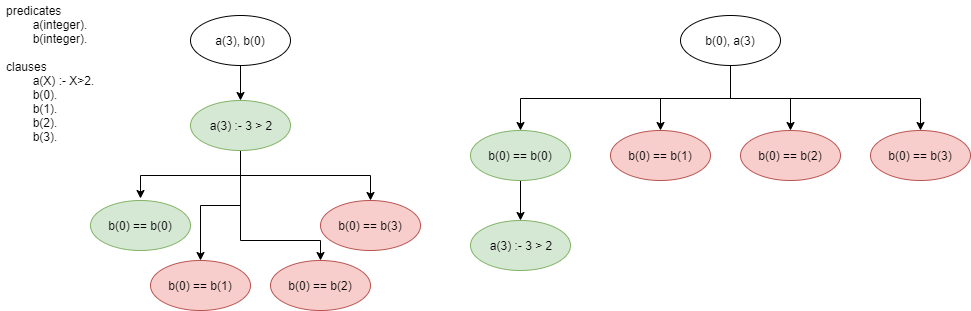


Рис. 1 – порядок не повлиял на количество сравнений

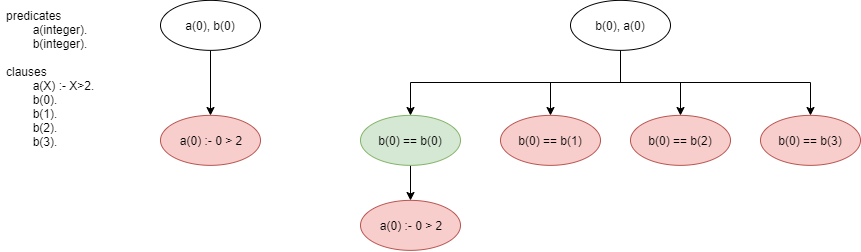


Рис. 2 – порядок повлиял на количество сравнений

**Задание №3** таблицы работы алгоритма унификации:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| шаг унифи ка ции | результирующая ячейка | рабочее поле | пункт алгоритма | стек |
| 0 |  |  | 1. | Info\_by\_car ("Audi", “Red, Surname, City, TeleTelephone, Bank\_name) =  Info\_by\_car (Car\_brand, Car\_color, Surname, City, TeleTelephone, Bank\_name) |
| 1 |  | Info\_by\_car ("Audi", “Red, Surname, City, TeleTelephone, Bank\_name) =  Info\_by\_car (Car\_brand, Car\_color, Surname, City, TeleTelephone, Bank\_name):- car\_owner(Surname, about\_car(Car\_brand, Car\_color, \_, \_)), Telephone\_record(Surname, TeleTelephone, address(City, \_, \_, \_)), bank\_owner(Surname, dep\_info(Bank\_name, \_, \_)).  -----🡪 | е | Car\_brand = "Audi" Car\_color = “Red”  Car\_owner(Surname about\_car(Car\_brand, Car\_color, \_, \_)) Telephone\_record(Surname, TeleTelephone, address(City, \_, \_, \_)) dep\_info(Surname, Bank\_name, \_, \_). |
| 2 | Car\_brand = "Audi" | Car\_brand = "Audi"  🡨----- | г | Color = “Red”  Car\_owner(Surname about\_car(“Audi”, Car\_color, \_, \_)) Telephone\_record(Surname, TeleTelephone, address(City, \_, \_, \_)) dep\_info(Surname, Bank\_name, \_, \_). |
| 3 | Car\_brand = "Audi",  Car\_color = “Red” | Car\_color = “Red” 🡨----- | г | Car\_owner(Surname about\_car(“Audi”, “Red”, \_, \_)) Telephone\_record(Surname, TeleTelephone, address(City, \_, \_, \_)) dep\_info(Surname, Bank\_name, \_, \_). |
| 4 | Car\_brand = "Audi",  Car\_color = “Red” | Пытаемся запустить алг. унификации для car\_owner("Paklin", about\_car("Cooper", "Red", 2000001, 2012)).= car\_owner(Surname, about\_car("Audi", "Red”, \_, \_)).  В итоге неудачно. |  | Telephone\_record(Surname, TeleTelephone, address(City, \_, \_, \_)) dep\_info(Surname, Bank\_name, \_, \_). |
| 5 | Car\_brand = "Audi",  Car\_color = “Red” | Пытаемся запустить алг. унификации для car\_owner("Kryat", about\_car("Mitsubishi", "Grey", 289731, 2006)) = car\_owner(Surname, about\_car("Audi", "Red”, \_, \_)).  В итоге неудачно. |  | Telephone\_record(Surname, TeleTelephone, address(City, \_, \_, \_)) dep\_info(Surname, Bank\_name, \_, \_). |
| 6 | Car\_brand = "Audi",  Car\_color = “Red” | Пытаемся запустить алг. унификации для car\_owner("Blohin", about\_car("Opel", "Grey", 290000, 2007)) = car\_owner(Surname, about\_car("Audi", "Red”, \_, \_)).  В итоге неудачно. |  | Telephone\_record(Surname, TeleTelephone, address(City, \_, \_, \_)) dep\_info(Surname, Bank\_name, \_, \_). |
| 7 | Car\_brand = "Audi",  Car\_color = “Red” | Пытаемся запустить алг. унификации для car\_owner("Blohin", about\_car("Audi", "Red", 2900000, 2015)) = car\_owner(Surname, about\_car("Audi", "Red”, \_, \_)).  --🡪 |  | Telephone\_record(Surname, TeleTelephone, address(City, \_, \_, \_)) dep\_info(Surname, Bank\_name, \_, \_). |
| 8 | Car\_brand = "Audi",  Car\_color = “Red” | car\_owner("Blohin", about\_car("Audi", "Red", 2900000, 2015)) = car\_owner(Surname, about\_car("Audi", "Red”, \_, \_)). |  | Telephone\_record(Surname, Telephone, address(City, \_, \_, \_)) dep\_info(Surname, Bank\_name, \_, \_). |
| 7 | Car\_brand = "Audi",  Color = “Red” Surname = “Blokhin” | Surname = “Blokhin”  🡨----- | г | "Audi" = "Audi" “red” = “red”  Telephone\_record(“Blokhin”, Telephone, address(City, \_, \_, \_)) dep\_info(“Blokhin”, Bank, \_, \_). |
| 8 | Model = "Audi",  Color = «Red» Last\_name = “Blokhin” | "Audi" = "Audi" | б | «Red» = «Red»  Telephone\_record(“Blokhin”, Telephone, address(City, \_, \_, \_)) dep\_info(“Blokhin”, Bank, \_, \_). |
| 9 | Model = "Audi",  Color = «Red» Last\_name = “Blokhin” | «Red» = «Red» | б | Telephone\_record(“Blokhin”, Telephone, address(City, \_, \_, \_)) dep\_info(“Blokhin”, Bank, \_, \_). |
| 10 | Model = "Audi",  Color = «Red» Last\_name = “Blokhin” | Пытаемся запустить алг. унификации для Telephone\_record(“Blokhin”, Telephone, address(City, \_, \_, \_)) = Telephone\_record("Blokhin", "89265291378", address("Moscow", "Varsh", 52, 9)) -----🡪 |  | Telephone\_record(“Blokhin”, Telephone, address(City, \_, \_, \_)) = Telephone\_record("Blokhin", "89265291378", address("Moscow", "Postal", 1, 2)) dep\_info(“Blokhin”, Bank, \_, \_). |
| 11 | Model = "Audi",  Color = «Red» Last\_name = “Blokhin” | Telephone\_record(“Blokhin”, Telephone, address(City, \_, \_, \_)) = Telephone\_record("Blokhin", 89265291378", address("Moscow", "Varsh", 52, 9)) -----🡪 | е | “Blokhin” = “Blokhin” Telephone = 89265291378" City = “Moscow”  dep\_info(“Blokhin”, Bank, \_, \_). |
| 12 | Model = "Audi",  Color = «Red» Last\_name = “Blokhin” | “Blokhin” = “Blokhin” | б | Telephone = "89265291378" City = “Moscow”  dep\_info(“Blokhin”, Bank, \_, \_). |
| 13 | Model = "Audi",  Color = «Red» Last\_name = “Blokhin” Telephone = "89265291378" | Telephone = "89265291378" 🡨----- | г | City = “Moscow”  dep\_info(“Blokhin”, Bank, \_, \_). |
| 14 | Model = "Audi",  Color = «Red» Last\_name = “Blokhin” Telephone = "89265291378 City = “Moscow” | City = “Moscow” 🡨----- | г | dep\_info(“Blokhin”, Bank, \_, \_). |
| 15 | -||- | Пытаемся запустить алг. унификации для dep\_info(“Blokhin”, Bank, \_, \_) = dep\_info("Blokhin", "RNCB", private, 999999999) -----🡪 |  | dep\_info(“Blokhin”, Bank, \_, \_) = dep\_info("Blokhin", "RNCB", private, 999999999) |
| 16 | -||- | Bank\_owner(“Blokhin”, Bank, \_, \_) = dep\_info("Blokhin", "Gazprom", private, 999999999) -----🡪 | е | “Blokhin” = “Blokhin” Bank = “RNCB” |
| 17 | -||- | “Blokhin” = “Blokhin” | б | Bank = “RNCB” |
| 18 | Model = "Audi",  Color = «Red» Last\_name = “Blokhin” Telephone = "89265291378 City = “Moscow” Bank = “RNCB” | Bank = “RNCB” 🡨----- | г |  |
| 19 |  | Не прошло и года как стек опустел. В рез. ячейке подстановка. Ну что, готовы? Откатываемся. (к 15) |  |  |
| 20 | Model = "Audi",  Color = «Red» Last\_name = “Blokhin” Telephone = "89265291378 City = “Moscow” | Пытаемся запустить алг. унификации для dep\_info(“Blokhin”, Bank, \_, \_) = dep\_info("Autumn", "Sberbank", public, 10).  В итоге неудачно. |  |  |
| 21 |  | Пытаемся запустить алг. унификации для dep\_info(“Blokhin”, Bank, \_, \_) = dep\_info("Summer", "Raiffeisen", public, 10000).  В итоге неудачно. |  |  |
|  |  | investorы закончились. Откатываемся к 10. |  |  |
| 22 | Model = "Audi",  Color = «Red» Last\_name = “Blokhin” | Пытаемся запустить алг. унификации для Telephone\_record(“Blokhin”, Telephone, address(City, \_, \_, \_)) = Telephone\_record("Blokhin", "+000", address("SPB", "Neva", 2, 4)) -----🡪 |  | Telephone\_record(“Blokhin”, Telephone, address(City, \_, \_, \_)) = Telephone\_record("Blokhin", "+000", address("SPB", "Neva", 2, 4)) dep\_info(“Blokhin”, Bank, \_, \_). |
| 23-32 |  | Тут все аналогично предыдущыму  (шаги 11-20) случаю |  |  |
| 33 | Model = "Audi",  Color = «Red» Last\_name = “Blokhin” Telephone = "+000 City = “SPB” Bank = “RNCB” | Стек опустел. В рез. ячейке подстановка. |  |  |
| 34 | Model = "Audi",  Color = «Red» Last\_name = “Blokhin | Откат к 22. Пытаемся запустить алг. унификации для Telephone\_record(“Blokhin”, Telephone, address(City, \_, \_, \_)) = Telephone\_record("Summer", "+345", address("Sevastopol", "Anotherstrt", 4, 8)) в итоге неудачно. |  | dep\_info(“Blokhin”, Bank, \_, \_). |
| 35 | Model = "Audi",  Color = «Red» Last\_name = “Blokhin | Пытаемся запустить алг. унификации для Telephone\_record(“Blokhin”, Telephone, address(City, \_, \_, \_)) = Telephone\_record("Autumn", "+333", address("Moscow", "Postal", 8, 16)). в итоге неудачно. |  | dep\_info(“Blokhin”, Bank, \_, \_). |
| 36 | Model = "Audi",  Color = «Red» | Откат к 5. Пытаемся запустить алг. унификации для car("Blokhin", "Audi", «Red», 150) = car("Summer", "Apple", green, 1000000). в итоге неудачно. |  | address(City, \_, \_, \_)) dep\_info(Last\_name, Bank, \_, \_). |