



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 №16-17*

*По курсу: «Функциональное и логическое  
программирование»*

**Темы:** «Использование правил в программе на Prolog», «Формирование эффективных программ на Prolog»

Студент: Якуба Д.В.

Группа: ИУ7-63Б

Преподаватели: Толпинская Н. Б.,

Строганов Ю. В.

Москва, 2021 г.

# Практическая часть

## Лабораторная работа 16.

Задание. Создать базу знаний: «ПРЕДКИ», позволяющую наиболее эффективным способом (за меньшее количество шагов, что обеспечивается меньшим количеством предложений БЗ – правил), и используя разные варианты (примеры) одного вопроса, определить (указать: какой вопрос для какого варианта):

1. По имени субъекта определить всех его бабушек (предки 2-го колена);
2. По имени субъекта определить всех его дедушек (предки 2-го колена);
3. По имени субъекта определить всех его бабушек и дедушек (предки 2-го колена);
4. По имени субъекта определить его бабушку по материнской линии (предки 2-го колена);
5. По имени субъекта определить его бабушку и дедушку по материнской линии (предки 2-го колена).

Минимизировать количество правил и количество вариантов вопросов. Использовать конъюнктивные правила и простой вопрос.

Для одного из вариантов ВОПРОСА и конкретной БЗ составить таблицу, отражающую конкретный порядок работы системы, с объяснениями:

- очередная проблема на каждом шаге и метод ее решения,
- каково новое текущее состояние резольвенты, как получено,
- какие дальнейшие действия? (запускается ли алгоритм унификации? Каких термов? Почему этих?),
- вывод по результатам очередного шага и дальнейшие действия.

Так как резольвента хранится в виде стека, то состояние резольвенты требуется отображать в столбик: вершина – сверху! Новый шаг надо начинать с нового состояния резольвенты!

```
domains
    sex = symbol.
    name = string.
    parent = parent(name, sex).

predicates
    isParentOf(parent, name).
    ancestors(name, name, name, name, name).

clauses
    isParentOf(parent("Alexey Romanov", m), "Morgan Romanov").
    isParentOf(parent("Alexandra Romanova", w), "Morgan Romanov").

    isParentOf(parent("Helen Romanova", w), "Alexey Romanov").
    isParentOf(parent("Vasiliy Romanov", m), "Alexey Romanov").

    isParentOf(parent("Klavdiya Portak", w), "Helen Romanova").
    isParentOf(parent("Evheny Portak", m), "Helen Romanova").

    isParentOf(parent("Valentina Romanova", w), "Vasiliy Romanov").
    isParentOf(parent("Elisey Romanov", m), "Vasiliy Romanov").

    isParentOf(parent("Anastasia Grishkovec", w), "Alexandra Romanova").
    isParentOf(parent("Gennadiy Sehenswurdigkeiten", m), "Alexandra Romanova").

    isParentOf(parent("Irina Grishkovec", w), "Anastasia Grishkovec").
    isParentOf(parent("Sergey Grishkovec", m), "Anastasia Grishkovec").

    isParentOf(parent("Klavdiya Sehenswurdigkeiten", w), "Gennadiy
Sehenswurdigkeiten").
    isParentOf(parent("Dmitry Sehenswurdigkeiten", m), "Gennadiy
Sehenswurdigkeiten").
```

```

ancestors(ChildName, MothersMother, MothersFather, FathersMother, FathersFather)
:-
    isParentOf(parent(MothersName, w), ChildName),
isParentOf(parent(FathersName, m), ChildName),
    isParentOf(parent(MothersMother, w), MothersName),
isParentOf(parent(MothersFather, m), MothersName),
    isParentOf(parent(FathersMother, w), FathersName),
isParentOf(parent(FathersFather, m), FathersName).

goal
    % ancestors("Morgan Romanov", MothersMotherName, _, FathersMotherName, _)
    % Point 1
    % MothersMotherName=Anastasia Grishkovec,
    % FathersMotherName=Helen Romanova

    % ancestors("Morgan Romanov", _, MothersFatherName, _, FathersFatherName)
    % Point 2
    % MothersFatherName=Gennadiy Sehenswurdigkeiten,
    % FathersFatherName=Vasiliiy Romanov

    % ancestors("Alexey Romanov", MothersMotherName, MothersFatherName,
FathersMotherName, FathersFatherName)
    % Point 3
    % MothersMotherName=Klavdiya Portak,
    % MothersFatherName=Evheny Portak,
    % FathersMotherName=Valentina Romanova,
    % FathersFatherName=Elisey Romanov

    % ancestors("Alexey Romanov", MothersMotherName, _, _, _)
    % Point 4
    % MothersMotherName=Klavdiya Portak

    ancestors("Alexandra Romanova", MothersMotherName, MothersFatherName, _, _)
    % Point 5
    % MothersMotherName=Irina Grishkovec,
    % MothersFatherName=Sergey Grishkovec
.

```

ancestors("Alexey Romanov", MothersMotherName, \_, \_, \_)

№ шаг а	Состояние резольвенты, и вывод: дальнейшие действия (почему?)	Для каких термов запускается алгоритм унификации: T1=T2 и каков результат (и подстановка)	Дальнейшие действия: прямой ход или откат (к чему приводит?)
1	Ancestors("Alexey Romanov", MothersMotherName, _, _, _)  Резольвента непуста, запуск алгоритма унификации для подцели, извлекаемой из стека	Ancestors("Alexey Romanov", MothersMotherName, _, _, _) = isParentOf(parent("Alexey Romanov", m), "Morgan Romanov").  Унификация неуспешна (несовпадение функторов)	Откат, переход к следующему предложению
2-14	...	...	...
15	Ancestors("Alexey Romanov", MothersMotherName, _, _, _)  Резольвента непуста, запуск алгоритма унификации для подцели, извлекаемой из стека	Ancestors("Alexey Romanov", MothersMotherName, _, _, _) = ancestors(ChildName, MothersMother, MothersFather, FathersMother, FathersFather) Унификация успешна  Подстановка: {ChildName="Alexey Romanov", MothersMother=MothersMotherName}	Формируется новое состояние резольвенты. Прямой ход
16	isParentOf(parent(MothersName, w), "Alexey Romanov"),	isParentOf(parent(MothersName, w), "Alexey Romanov") =	Откат, переход к следующему предложению

	isParentOf(parent(_, m), "Alexey Romanov"), isParentOf(parent(MothersMotherName, w), MothersName), isParentOf(parent(_, m), MothersMotherName), isParentOf(parent(_, w), _), isParentOf(parent(_, m), _)  Резольвента не пуста, запуск алгоритма унификации для подцели, извлекаемой из стека	isParentOf(parent("Alexey Romanov", m), "Morgan Romanov") Унификация не успешна (несовпадение термов)	
17	...	...	...
18	isParentOf(parent(MothersName, w), "Alexey Romanov"), isParentOf(parent(_, m), "Alexey Romanov"), isParentOf(parent(MothersMotherName, w), MothersName), isParentOf(parent(_, m), MothersMotherName), isParentOf(parent(_, w), _), isParentOf(parent(_, m), _)  Резольвента не пуста, запуск алгоритма унификации для подцели, извлекаемой из стека	isParentOf(parent(MothersName, w), "Alexey Romanov") = isParentOf(parent("Helen Romanova", w), "Alexey Romanov") Унификация успешна  Подстановка: {ChildName="Alexey Romanov", MothersMother=MothersMotherName, MothersName="Helen Romanova"}	Формируется новое состояние резольвенты, прямой ход
19	isParentOf(parent(_, m), "Alexey Romanov"), isParentOf(parent(MothersMotherName, w), "Helen Romanova"), isParentOf(parent(_, m), MothersMotherName), isParentOf(parent(_, w), _), isParentOf(parent(_, m), _)  Резольвента не пуста, запуск алгоритма унификации для подцели, извлекаемой из стека	Ни одна переменная в терме isParentOf(parent(_, m), "Alexey Romanov") конкретизации не подлежит	Формируется новое состояние резольвенты
20	isParentOf(parent(MothersMotherName, w), "Helen Romanova"), isParentOf(parent(_, m), MothersMotherName), isParentOf(parent(_, w), _), isParentOf(parent(_, m), _)  Резольвента не пуста, запуск алгоритма унификации для подцели, извлекаемой из стека	isParentOf(parent(MothersMotherName, w), "Helen Romanova") = isParentOf(parent("Alexey Romanov", m), "Morgan Romanov") Унификация не успешна (несовпадение термов)	Откат, переход к следующему предложению
21-23	...	...	...
24	isParentOf(parent(MothersMotherName, w), "Helen Romanova"), isParentOf(parent(_, m), MothersMotherName), isParentOf(parent(_, w), _), isParentOf(parent(_, m), _)  Резольвента не пуста, запуск алгоритма унификации для подцели, извлекаемой из стека	isParentOf(parent(MothersMotherName, w), "Helen Romanova") = isParentOf(parent("Klavdiya Portak", w), "Helen Romanova") Унификация успешна  Подстановка: {ChildName="Alexey Romanov", MothersMotherName="Klavdiya Portak", MothersName="Helen Romanova"}	Формируется новое состояние резольвенты, прямой ход
25	isParentOf(parent(_, m), "Klavdiya Portak"), isParentOf(parent(_, w), _), isParentOf(parent(_, m), _)  Резольвента не пуста, запуск алгоритма унификации для подцели, извлекаемой из стека	Ни одна переменная в терме isParentOf(parent(_, m), "Klavdiya Portak") конкретизации не подлежит	Формируется новое состояние резольвенты, прямой ход

26	<code>isParentOf(parent(_, w), _), isParentOf(parent(_, m), _)</code>  Резольвента непуста, запуск алгоритма унификации для подцели, извлекаемой из стека	Ни одна переменная в терме <code>isParentOf(parent(_, w), _)</code> , <code>isParentOf(parent(_, m), _)</code> не подлежит конкретизации	Формируется новое состояние резольвенты, прямой ход
27	Резольвента пуста	Подстановка: <code>{ChildName="Alexey Romanov", MothersMotherName="Klavdiya Portak", MothersName="Helen Romanova"}</code>	Резольвента пуста  Вывод: <code>MothersMotherName=Klavdiya Portak</code>  Новая подстановка: <code>{ChildName="Alexey Romanov", MothersMother=MothersMotherName, MothersName="Helen Romanova"}</code>  Откат относительно пункта 24
28	<code>isParentOf(parent(MothersMotherName, w), "Helen Romanova"), isParentOf(parent(_, m), MothersMotherName), isParentOf(parent(_, w), _), isParentOf(parent(_, m), _)</code>  Резольвента непуста, запуск алгоритма унификации для подцели, извлекаемой из стека	<code>isParentOf(parent(MothersMotherName, w), "Helen Romanova") = isParentOf(parent("Evheny Portak", m), "Helen Romanova").</code>  Унификация не успешна (несовпадение термов)	Откат, переход к следующему предложению
29-36	...	...	...
37	<code>isParentOf(parent(MothersMotherName, w), "Helen Romanova"), isParentOf(parent(_, m), MothersMotherName), isParentOf(parent(_, w), _), isParentOf(parent(_, m), _)</code>  Резольвента непуста, запуск алгоритма унификации для подцели, извлекаемой из стека	<code>isParentOf(parent(MothersMotherName, w), "Helen Romanova") = ancestors(ChildName, MothersMother, MothersFather, FathersMother, FathersFather)</code>  Унификация не успешна (несовпадение функторов)	Конец БЗ достигнут. Откат относительно пункта 18  Новая подстановка: <code>{ChildName="Alexey Romanov", MothersMother=MothersMotherName}</code>
38	<code>isParentOf(parent(MothersName, w), "Alexey Romanov"), isParentOf(parent(_, m), "Alexey Romanov"), isParentOf(parent(MothersMotherName, w), MothersName), isParentOf(parent(_, m), MothersMotherName), isParentOf(parent(_, w), _), isParentOf(parent(_, m), _)</code>  Резольвента непуста, запуск алгоритма унификации для подцели, извлекаемой из стека	<code>isParentOf(parent(MothersName, w), "Alexey Romanov") = isParentOf(parent("Vasiliy Romanov", m), "Alexey Romanov")</code>  Унификация не успешна (несовпадение термов)	Откат, переход к следующему предложению
39-48	...	...	...
49	<code>isParentOf(parent(MothersName, w), "Alexey Romanov"), isParentOf(parent(_, m), "Alexey Romanov"), isParentOf(parent(MothersMotherName, w), MothersName), isParentOf(parent(_, m), MothersMotherName), isParentOf(parent(_, w), _), isParentOf(parent(_, m), _)</code>  Резольвента непуста, запуск алгоритма унификации для подцели, извлекаемой из стека	<code>isParentOf(parent(MothersName, w), "Alexey Romanov") = ancestors(ChildName, MothersMother, MothersFather, FathersMother, FathersFather)</code> Унификация не успешна (несовпадение функторов)	Конец БЗ достигнут. Откат относительно пункта 15  Подстановки более нет

50	Ancestors("Alexey Romanov", MothersMotherName, _, _, _)  Резольвента непушта, но достигнут конец БЗ, завершение работы системы	-	Резольвента непушта, но достигнут конец БЗ, завершение работы системы
----	---	---	---

## Лабораторная работа 17.

Задание. В одной программе написать правила, позволяющие найти

1. Максимум из двух чисел:
  - a. Без использования отсечения;
  - b. С использованием отсечения;
2. Максимум из трех чисел:
  - a. Без использования отсечения;
  - b. С использованием отсечения.

Убедиться в правильности результатов. Для каждого случая из пункта 2 обосновать необходимость всех условий тела. Для одного из вариантов ВОПРОСА и каждого варианта задания 2 составить таблицу, отражающую конкретный порядок работы системы.

Так как резольвента хранится в виде стека, то состояние резольвенты требуется отображать в столбик: вершина – сверху! Новый шаг надо начинать с нового состояния резольвенты!

Требуется ответить на вопрос: «За счет чего может быть достигнута эффективность работы системы?»

```

predicates
  maxOfTwo(real, real, real).
  maxOfTwoCut(real, real, real).
  maxOfThree(real, real, real, real).
  maxOfThreeCut(real, real, real, real).

clauses
  maxOfTwo(FVar, SVar, FVar) :- FVar >= SVar.
  maxOfTwo(FVar, SVar, SVar) :- FVar < SVar.

  maxOfTwoCut(FVar, SVar, FVar) :- FVar >= SVar, !.
  maxOfTwoCut(_, SVar, SVar).

  maxOfThree(FVar, SVar, TVar, FVar) :- FVar >= SVar, FVar >= TVar.
  maxOfThree(FVar, SVar, TVar, SVar) :- SVar >= FVar, SVar >= TVar.
  maxOfThree(FVar, SVar, TVar, TVar) :- TVar >= SVar, TVar >= FVar.

  maxOfThreeCut(FVar, SVar, TVar, FVar) :- FVar >= SVar, FVar >= TVar, !.
  maxOfThreeCut(_, SVar, TVar, SVar) :- SVar >= TVar, !.
  maxOfThreeCut(_, _, TVar, TVar).

goal
  maxOfTwo(1.2, 1.333, Max); % 1.333
  maxOfTwo(1.999, -2, Max); % 1.999
  maxOfTwo(1.999, 1.999, Max); % 1.999, 1.999

  maxOfTwoCut(1.2, 1.333, Max); % 1.333
  maxOfTwoCut(1.999, -2, Max); % 1.999
  maxOfTwoCut(1.999, 1.999, Max); % 1.999

  maxOfThree(1.1, 1.2, 1.3, Max); % 1.3
  maxOfThree(1.3, 1.2, 1.1, Max); % 1.3
  maxOfThree(1.3, 1.1, 1.2, Max); % 1.3
  maxOfThree(1.2, 1.1, 1.3, Max); % 1.3
  maxOfThree(1.2, 1.2, 1.3, Max); % 1.3
  maxOfThree(1.2, 1.3, 1.2, Max); % 1.3

```

```

maxOfThree(1.3, 1.2, 1.2, Max); % 1.3
maxOfThree(111, 111, 111, Max); % 111 111 111

maxOfThreeCut(1.1, 1.2, 1.3, Max); % 1.3
maxOfThreeCut(1.3, 1.2, 1.1, Max); % 1.3
maxOfThreeCut(1.3, 1.1, 1.2, Max); % 1.3
maxOfThreeCut(1.2, 1.1, 1.3, Max); % 1.3
maxOfThreeCut(1.2, 1.2, 1.3, Max); % 1.3
maxOfThreeCut(1.2, 1.3, 1.2, Max); % 1.3
maxOfThreeCut(1.3, 1.2, 1.2, Max); % 1.3
maxOfThreeCut(111, 111, 111, Max) % 111
.

```

Обоснование.

```

maxOfThree(FVar, SVar, TVar, FVar) :- FVar >= SVar, FVar >= TVar.
maxOfThree(FVar, SVar, TVar, SVar) :- SVar >= FVar, SVar >= TVar.
maxOfThree(FVar, SVar, TVar, TVar) :- TVar >= SVar, TVar >= FVar.

```

Условия в теле первого правила изложены для определения факта того, что первое переданное число больше двух других чисел или равно им. Условия в теле второго правила изложены из факта того, что второе переданное число больше двух других чисел или равно им. Условия в теле третьего правила изложены из факта того, что третье переданное число больше двух других чисел или равно им.

```

maxOfThreeCut(FVar, SVar, TVar, FVar) :- FVar >= SVar, FVar >= TVar, !.
maxOfThreeCut(_, SVar, TVar, SVar) :- SVar >= TVar, !.
maxOfThreeCut(_, _, TVar, TVar).

```

Условия в теле первого правила изложены для определения факта того, что первое переданное число больше или равно двух других чисел. В случае, если условие не выполняется, FVar – точно меньше второго переданного значения SVar, поэтому в теле второго правила определяется, больше или равно значению SVar значения TVar. В случае, если условие не выполняется, TVar – наибольшее из трёх чисел. В случае выполнения условия первого или второго правила смысла далее идти по предложениям процедуры нет, так как искомое значение уже было найдено, поэтому применяется системный предикат отсечения.

maxOfThree(1.1, 1.2, 1.3, Max)

№ шага	Состояние резольвенты, и вывод: дальнейшие действия (почему?)	Для каких термов запускается алгоритм унификации: T1=T2 и каков результат (и подстановка)	Дальнейшие действия: прямой ход или откат (к чему приводит?)
1	maxOfThree(1.1, 1.2, 1.3, Max)  Резольвента непуста, запуск алгоритма унификации для подцели, извлекаемой из стека	maxOfThree(1.1, 1.2, 1.3, Max) = maxOfTwo(FVar, SVar, FVar) Унификация неуспешна (несовпадение функторов)	Откат, переход к следующему предложению
2-4	...	...	...
5	maxOfThree(1.1, 1.2, 1.3, Max)  Резольвента непуста, запуск алгоритма унификации для подцели, извлекаемой из стека	maxOfThree(1.1, 1.2, 1.3, Max) = maxOfThree(FVar, SVar, TVar, FVar) :- FVar >= SVar, FVar >= TVar Унификация успешна  Подстановка: {FVar=1.1, SVar=1.2, TVar=1.3, FVar=Max }	Прямой ход, формируется новое состояние резольвенты
6	FVar >= SVar, FVar >= TVar  Резольвента непуста, выбор подцели, извлекаемой из стека	Сравнение связанных переменных: FVar >= SVar. Результат: ложь	Откат к предыдущему состоянию резольвенты  Подстановки более нет

7	<code>maxOfThree(1.1, 1.2, 1.3, Max)</code>  Резольвента непуста, запуск алгоритма унификации для подцели, извлекаемой из стека	<code>maxOfThree(1.1, 1.2, 1.3, Max)</code> <code>= maxOfThree(FVar, SVar, TVar, SVar) :- SVar &gt;= FVar, SVar &gt;= TVar</code> Унификация успешна  Подстановка: {FVar=1.1, SVar=1.2, TVar=1.3, SVar=Max }	Прямой ход, формируется новое состояние резольвенты
8	<code>SVar &gt;= FVar,</code> <code>SVar &gt;= TVar</code>  Резольвента непуста, выбор подцели, извлекаемой из стека	Сравнение связанных переменных: <code>SVar &gt;= FVar</code> Результат: истина	Прямой ход, формируется новое состояние резольвенты
9	<code>SVar &gt;= TVar</code>  Резольвента непуста, выбор подцели, извлекаемой из стека	Сравнение связанных переменных: <code>SVar &gt;= TVar</code> Результат: ложь	Откат к предыдущему состоянию резольвенты  Подстановки более нет
10	<code>maxOfThree(1.1, 1.2, 1.3, Max)</code>  Резольвента непуста, запуск алгоритма унификации для подцели, извлекаемой из стека	<code>maxOfThree(1.1, 1.2, 1.3, Max)</code> <code>= maxOfThree(FVar, SVar, TVar, TVar) :- TVar &gt;= SVar, TVar &gt;= FVar</code> Унификация успешна  Подстановка: {FVar=1.1, SVar=1.2, TVar=1.3, TVar=Max}	Прямой ход, формируется новое состояние резольвенты
11	<code>TVar &gt;= SVar,</code> <code>TVar &gt;= FVar</code>  Резольвента непуста, выбор подцели, извлекаемой из стека	Сравнение связанных переменных: <code>TVar &gt;= SVar</code> Результат: истина	Прямой ход, формируется новое состояние резольвенты
12	<code>TVar &gt;= FVar</code>  Резольвента непуста, выбор подцели, извлекаемой из стека	Сравнение связанных переменных: <code>TVar &gt;= FVar</code> Результат: истина	Прямой ход, формируется новое состояние резольвенты
13	Резольвента пуста	Подстановка: {FVar=1.1, SVar=1.2, TVar=1.3, Max=TVar}	Резольвента пуста  Вывод: Max=1.3  Подстановки более нет  Откат относительно пункта 10
14	<code>maxOfThree(1.1, 1.2, 1.3, Max)</code>  Резольвента непуста, запуск алгоритма унификации для подцели, извлекаемой из стека	<code>maxOfThree(1.1, 1.2, 1.3, Max)</code> <code>= maxOfThreeCut(FVar, SVar, TVar, FVar) :- FVar &gt;= SVar, FVar &gt;= TVar, !.</code> Унификация неуспешна (несовпадение функторов)	Откат, переход к следующему предложению
15	...	...	...
16	<code>maxOfThree(1.1, 1.2, 1.3, Max)</code>  Резольвента непуста, запуск алгоритма унификации для подцели, извлекаемой из стека	<code>maxOfThree(1.1, 1.2, 1.3, Max)</code> <code>= maxOfThreeCut(_, _, TVar, TVar)</code> Унификация неуспешна (несовпадение функторов)	Конец БЗ достигнут, завершение работы системы



maxOfThreeCut(1.2, 1.3, 1.2, Max)

№ шага	Состояние резольвенты, и вывод: дальнейшие действия (почему?)	Для каких термов запускается алгоритм унификации: T1=T2 и каков результат (и подстановка)	Дальнейшие действия: прямой ход или откат (к чему приводит?)
1	maxOfThreeCut(1.2, 1.3, 1.2, Max)  Резольвента непуста, запуск алгоритма унификации для подцели, извлекаемой из стека	maxOfThreeCut(1.2, 1.3, 1.2, Max) = maxOfTwo(FVar, SVar, FVar) Унификация неуспешна (несовпадение функторов)	Откат, переход к следующему предложению
2-7	...	...	...
8	maxOfThreeCut(1.2, 1.3, 1.2, Max)  Резольвента непуста, запуск алгоритма унификации для подцели, извлекаемой из стека	maxOfThreeCut(1.2, 1.3, 1.2, Max) = maxOfThreeCut(FVar, SVar, TVar, FVar) :- FVar >= SVar, FVar >= TVar, ! Унификация успешна  Подстановка: {FVar=1.2, SVar=1.3, TVar=1.2, FVar=Max}	Прямой ход, формируется новое состояние резольвенты
9	FVar >= SVar, FVar >= TVar, !  Резольвента непуста, выбор подцели, извлекаемой из стека	Сравнение связанных переменных: FVar >= SVar Результат: ложь	Откат к предыдущему состоянию резольвенты, подстановки более нет
10	maxOfThreeCut(1.2, 1.3, 1.2, Max)  Резольвента непуста, запуск алгоритма унификации для подцели, извлекаемой из стека	maxOfThreeCut(1.2, 1.3, 1.2, Max) = maxOfThreeCut(_, SVar, TVar, SVar) :- SVar >= TVar, ! Унификация успешна  Подстановка: {FVar=1.2, SVar=1.3, TVar=1.2, SVar=Max}	Прямой ход, формируется новое состояние резольвенты
11	SVar >= TVar !  Резольвента непуста, выбор подцели, извлекаемой из стека	Сравнение связанных переменных: SVar >= TVar Результат: истина	Прямой ход, формируется новое состояние резольвенты
12	!  Резольвента непуста, выбор подцели, извлекаемой из стека	Встречен системный предикат отсечения	Решение найдено, резольвента пуста.  Вывод: Max=1.3  Завершение работы системы