



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 № 7**

**По дисциплине «Функциональное и логическое программирование»**

**Студент: Тимонин А. С.**

**Группа ИУ7-626**

**Преподаватель Толпинская Н. Б.**

Москва.  
2020 г.

## Практическая часть

## Ответить на вопросы:

- Какое первое состояние резольвенты?
- В каком случае система запускает алгоритм унификации? (т.е. Как эту необходимость на формальном уровне распознает система?)
- Каково назначение использования алгоритма унификации?
- Каков результат работы алгоритма унификации?
- В каких пределах программы переменные уникальны?
- Как применяется подстановка, полученная с помощью алгоритма унификации?
- Как изменяется резольвента?
- В каких случаях запускается механизм отката?

## В одной программе написать правила, позволяющие найти

- [illegible]

Убедиться в правильности результатов.

**Для каждого случая пункта 2 обосновать необходимость всех условий тела.**

Для одного из вариантов **ВОПРОСА** и каждого варианта задания **2** составить таблицу, отражающую конкретный порядок работы системы:

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

## Листинг 1. Реализация программы

```
domains
    num1, num2, num3, max = integer.

predicates
    nondeterm maxOf2(num1, num2, max).
```

```

nondeterm maxOf2c(num1, num2, max).

nondeterm maxOf3(num1, num2, num3, max).
nondeterm maxOf3c(num1, num2, num3, max).

```

## clauses

```

% Bez otsecheniya dlya 2 chisel
maxOf2(Num1, Num2, Max) :-
    Num1 > Num2, Max = Num1.
maxOf2(Num1, Num2, Max) :-
    Num1 <= Num2, Max = Num2.

% S otsecheniem dlya 2 chisel
maxOf2c(Num1, Num2, Max) :-
    Num1 > Num2, Max = Num1, !.
maxOf2c(_, Num2, Max) :-
    Max = Num2.

% Bez otsecheniya dlya 3 chisel
maxOf3(Num1, Num2, Num3, Max) :-
    Num1 >= Num2, Num1 >= Num3, Max = Num1.
maxOf3(Num1, Num2, Num3, Max) :-
    Num2 >= Num1, Num2 >= Num3, Max = Num2.
maxOf3(Num1, Num2, Num3, Max) :-
    Num3 >= Num1, Num3 >= Num2, Max = Num3.

% S otsecheniem dlya 3 chisel
maxOf3c(Num1, Num2, Num3, Max) :-
    Num1 >= Num2, Num1 >= Num3, Max = Num1, !.
maxOf3c(_, Num2, Num3, Max) :-
    Num2 >= Num3, Max = Num2, !.
maxOf3c(_, _, Num3, Max) :-
    Max = Num3.

```

## goal

```

%
% FOR 2 NUMBERS
%

write("Max2 s otsecheniem\n"),
write("max(2, 1): "),
maxOf2c(2, 1, Max);

write("max(5, 10): "),
maxOf2c(5, 10, Max);

write("max(5, 10): "),
maxOf2c(5, 10, Max);

write("\n\nMax2 bez otsecheniya\n"),
write("max(3, 2): "),
maxOf2(3, 2, Max);

```

```
write("max(21, -21): "),
maxOf2(21, -21, Max);

write("max(6, 11): "),
maxOf2(6, 11, Max).

%
% FOR 3 NUMBERS
%

write("Max3 s otsecheniem\n"),
write("max(2, 1, 9): "),
maxOf3c(2, 1, 9, Max);

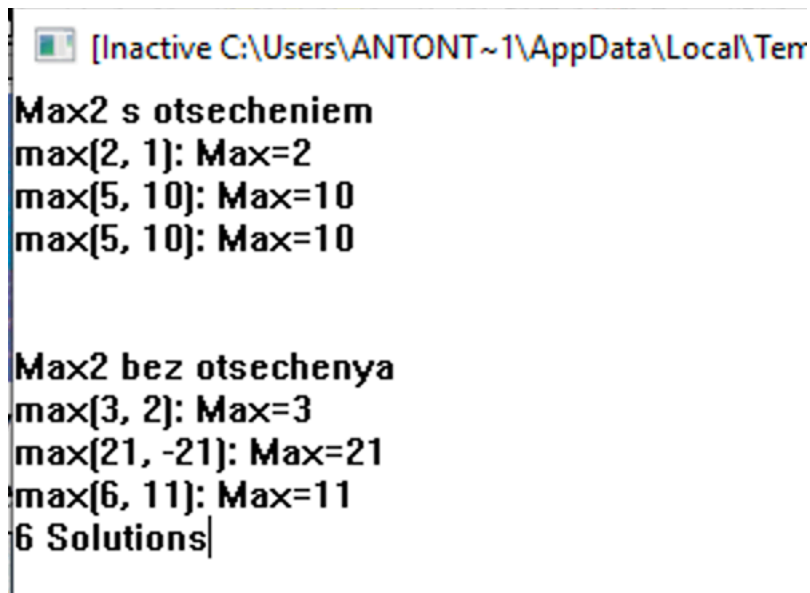
write("max(5, 10, 5): "),
maxOf3c(5, 10, 5, Max);

write("max(10, 5, 1): "),
maxOf3c(10, 5, 1, Max);

write("\n\nMax3 bez otsecheniya\n"),
write("max(3, 2, 2): "),
maxOf3(3, 2, 2, Max);

write("max(21, -21, 0): "),
maxOf3(21, -21, 0, Max);

write("max(6, 11, 19): "),
maxOf3(6, 11, 19, Max).
```



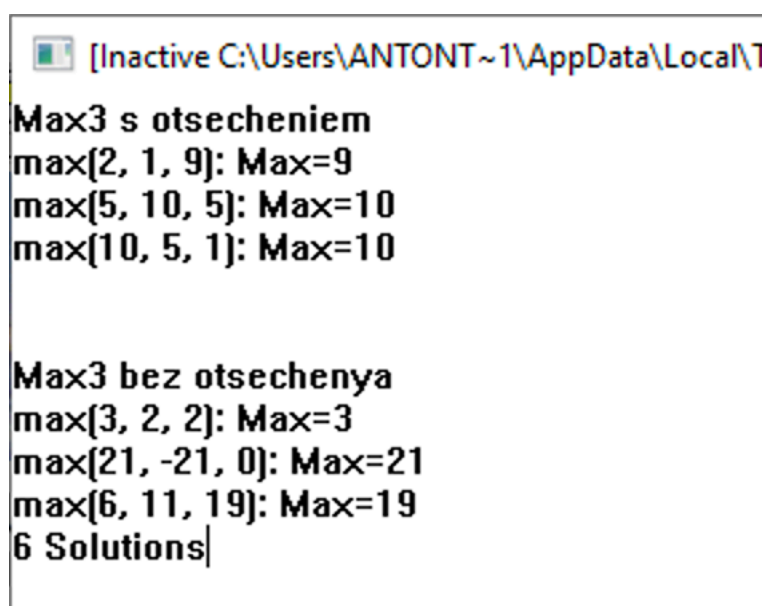
```
[Inactive C:\Users\ANTONT~1\AppData\Local\Temp]

Max2 s otsecheniem
max{2, 1}: Max=2
max{5, 10}: Max=10
max{5, 10}: Max=10

Max2 bez otsechenya
max{3, 2}: Max=3
max{21, -21}: Max=21
max{6, 11}: Max=11
6 Solutions|
```

Рисунок 1. Поиск максимального числа, среди двух чисел

```
maxOf2(num1, num2, max).
maxOf2c(num1, num2, max).
```



```
[Inactive C:\Users\ANTONT~1\AppData\Local\Temp]

Max3 s otsecheniem
max{2, 1, 9}: Max=9
max{5, 10, 5}: Max=10
max{10, 5, 1}: Max=10

Max3 bez otsechenya
max{3, 2, 2}: Max=3
max{21, -21, 0}: Max=21
max{6, 11, 19}: Max=19
6 Solutions|
```

Рисунок 2. Поиск максимального числа, среди трех чисел

```
maxOf3(num1, num2, num3, max).
maxOf3c(num1, num2, num3, max).
```

## Формирование ответа

Таблица 1.  $\text{maxOf3c}(2, 1, 9, \text{Max})$ .

| № шага | Состояние резольвенты, и вывод: дальнейшие действия (почему?)                                       | Для каких термов запускается алгоритм унификации: $T1=T2$ и каков <b>результат</b> (и подстановка)   | Дальнейшие действия: прямой ход или откат (почему и к чему приводит?) |
|--------|---|--|---|
| 1      | $\text{maxOf3c}(2, 1, 9, \text{Max})$   | Подстановка $\text{num1} = 2, \text{num2} = 1, \text{num3} = 9, \text{Max} = \text{Max}$<br>$\text{maxOf3c}(2, 1, 9, \text{Max})$<br>$\text{maxOf3c}(\text{num1}, \text{num2}, \text{num3}, \text{Max})$ | Прямой ход  |
| 2      | $\text{Num1} \geq \text{Num2}$<br>$\text{Num1} \geq \text{Num3}$<br>$\text{Max} = \text{Num1}$<br>! | Проверка<br>$2 \geq 1$<br>Да   | Прямой ход  |
| 3      | $\text{Num1} \geq \text{Num3}$<br>$\text{Max} = \text{Num1}$<br>!                                   | Проверка<br>$2 \geq 9$<br>Нет  | Обратный ход  |
| 4      | $\text{maxOf3c}(2, 1, 9, \text{Max})$   | Подстановка $\text{num2} = 1, \text{num3} = 9, \text{Max} = \text{Max}$<br>$\text{maxOf3c}(2, 1, 9, \text{Max})$<br>$\text{maxOf3c}(\_, \text{num2}, \text{num3}, \text{Max})$                           | Прямой ход  |
| 5      | $\text{Num2} \geq \text{Num3}$<br>$\text{Max} = \text{Num2}$<br>!                                   | Проверка<br>$1 \geq 9$<br>Нет  | Обратный ход  |
| 6      | $\text{maxOf3c}(2, 1, 9, \text{Max})$   | Подстановка $\text{num3} = 9, \text{Max} = \text{Max}$<br>$\text{maxOf3c}(2, 1, 9, \text{Max})$<br>$\text{maxOf3c}(\_, \_, \text{num3}, \text{Max})$   | Прямой ход  |
| 7      | $\text{Max} = \text{Num3}$<br>!   | <b>Подстановка:</b> $\text{Max} = 9$   | Прямой ход  |
| 8      | !   | <b>Результат:</b> $\text{Max} = 9$   | Обратный ход  |

Таблица 2.  $\text{maxOf3}(6, 11, 19, \text{Max})$ .

| № шага | Состояние резольвенты, и вывод: дальнейшие действия (почему?)                                  | Для каких термов запускается алгоритм унификации: $T1=T2$ и каков <b>результат</b> (и подстановка)  | Дальнейшие действия: прямой ход или откат (почему и к чему приводит?) |
|--------|--|---|---|
| 1      | $\text{maxOf3}(6, 11, 19, \text{Max})$   | Подстановка $\text{num1} = 6, \text{num2} = 11, \text{num3} = 19, \text{Max} = \text{Max}$<br>$\text{maxOf3}(6, 11, 19, \text{Max})$<br>$\text{maxOf3c}(\text{num1}, \text{num2}, \text{num3}, \text{Max})$ | Прямой ход  |
| 2      | $\text{Num1} \geq \text{Num2}$<br>$\text{Num1} \geq \text{Num3}$<br>$\text{Max} = \text{Num1}$ | Проверка<br>$6 \geq 11$<br>Нет  | Обратный ход  |

|    |  |  |              |
|----|--|--|--------------|
| 3  | maxOf3(6, 11, 19, Max)                     | Подстановка num1 = 6, num2 = 11,<br>num3 = 19, Max = Max<br>maxOf3(6, 11, 19, Max)<br>maxOf3c(num1, num2, num3, Max) | Прямой ход   |
| 4  | Num2 >= Num1<br>Num2 >= Num3<br>Max = Num2 | Проверка<br>11 >= 6<br>Да  | Прямой ход   |
| 5  | Num2 >= Num3<br>Max = Num2                 | Проверка<br>11 >= 19<br>Нет  | Обратный ход |
| 6  | maxOf3(6, 11, 19, Max)                     | Подстановка num1 = 6, num2 = 11,<br>num3 = 19, Max = Max<br>maxOf3(6, 11, 19, Max)<br>maxOf3c(num1, num2, num3, Max) | Прямой ход   |
| 7  | Num3 >= Num1<br>Num3 >= Num2<br>Max = Num3 | Проверка<br>19 >= 6<br>Да  | Прямой ход   |
| 8  | Num3 >= Num2<br>Max = Num3                 | Проверка<br>19 >= 11<br>Да   | Прямой ход   |
| 9  | Max = Num3                                 | <b>Подстановка:</b> Max = 19   | Прямой ход   |
| 10 | Пусто                                      | <b>Результат:</b> Max = 19   | Обратный ход |

## Теоретическая часть

### Какое первое состояние резольвенты?

Простой вопрос

### В каком случае система запускает алгоритм унификации? (т.е. Как эту необходимость на формальном уровне распознает система?)

Если резольвента не пуста – запускается алгоритм унификации, а если – пуста, это значит, что получен один, однократный ответ «Да» на поставленный вопрос, после чего включается механизм отката, в попытке найти другое решение с помощью другого знания. При этом, БЗ просматривается сверху вниз.

### Каково назначение использования алгоритма унификации?

Алгоритм унификации связывает переменные из вопроса со значениями, с параметрами, которые содержатся в правилах и фактах.

### Каков результат работы алгоритма унификации?

Унификация – операция, которая позволяет формализовать процесс логического вывода. Алгоритм унификации сопоставляет подцель с заданной переменной.

```
1 goal
2     a = man("Anton", Surname).
3     a = man(Name, "Timonin")
```

Например, если в строке 2, в переменной a Name сопоставится с «Anton», а в строке 3, переменная Surname сопоставится с «Timonin». Если бы в строке 2 мы поставили бы фамилию, например, «Konin», тогда строка 3 выдала бы ошибку, так как «Konin» не равно «Timonin». Это связано с тем, что когда параметр в какой-либо переменной занят, он перестает сопоставлять переменные, а начинает их сравнивать.

### В каких пределах программы переменные уникальны?



Именованные переменные уникальны в рамках предложения, а анонимная переменная – любая уникальна.

### **Как применяется подстановка, полученная с помощью алгоритма унификации?**

В ходе алгоритма унификации может быть два варианта: унифицируемые термы успешны или нет. Если они успешны:

1. T1 и T2 - одинаковые константы;
2. T1 – не конкретизированная переменная, а T2 - константа или составной терм, не содержащий в качестве аргумента T1. Тогда унификация успешна, причем T1 конкретизируется значением T2.

Тогда применяется подстановка и переменные связываются со значениями.

### **Как изменяется резольвента?**

Преобразования резольвенты выполняются с помощью редукции.

Редукцией цели G с помощью программы P называется замена цели G телом того правила из P, заголовок которого унифицируется с целью

### **В каких случаях запускается механизм отката?**

Когда в программе возможен выбор нескольких вариантов, Пролог заносит в стек точку возврата, для последующего отката по этой точке возврата.

Пролог унифицирует выбранный вариант, если унификация прошла успешно, тогда пролог подготавливает ответ, и далее по точке возврата происходит унификация с другими вариантами. Если пролог не видит дальнейшие варианты, которые он мог бы проунифицировать, тогда по точке возврата программа возвращается на еще более раннюю стадию.