



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 №17

*По предмету: «Функциональное и логическое
программирование»*

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

Москва, 2020 г

Задание

В одной программе написать правила, позволяющие найти

- 1. Максимум из двух чисел** а) без использования отсечения,
в) с использованием отсечения;
- 2. Максимум из трех чисел** а) без использования отсечения,
в) с использованием отсечения;

Листинг:

```
domains
```

```
    num = integer.
```

```
predicates
```

```
    max_of_2_a(num, num, num).
```

```
    max_of_2_b(num, num, num).
```

```
    max_of_3_a(num, num, num, num).
```

```
    max_of_3_b(num, num, num, num).
```

```
clauses
```

```
    max_of_2_a(A, B, A):- A >= B.
```

```
    max_of_2_a(A, B, B):- B > A.
```

```
    max_of_2_b(A, B, A):- A > B, !.
```

```
    max_of_2_b(_, B, B).
```

```
    max_of_3_a(A, B, C, A):- A >= B, A >= C.
```

```
    max_of_3_a(A, B, C, B):- B >= A, B >= C.
```

```
    max_of_3_a(A, B, C, C):- C >= B, C >= A.
```

```
    max_of_3_b(A, B, C, A):- A >= B, A >= C, !.
```

```
    max_of_3_b(_, B, C, B):- B >= C, !.
```

```
    max_of_3_b(_, _, C, C).
```

goal

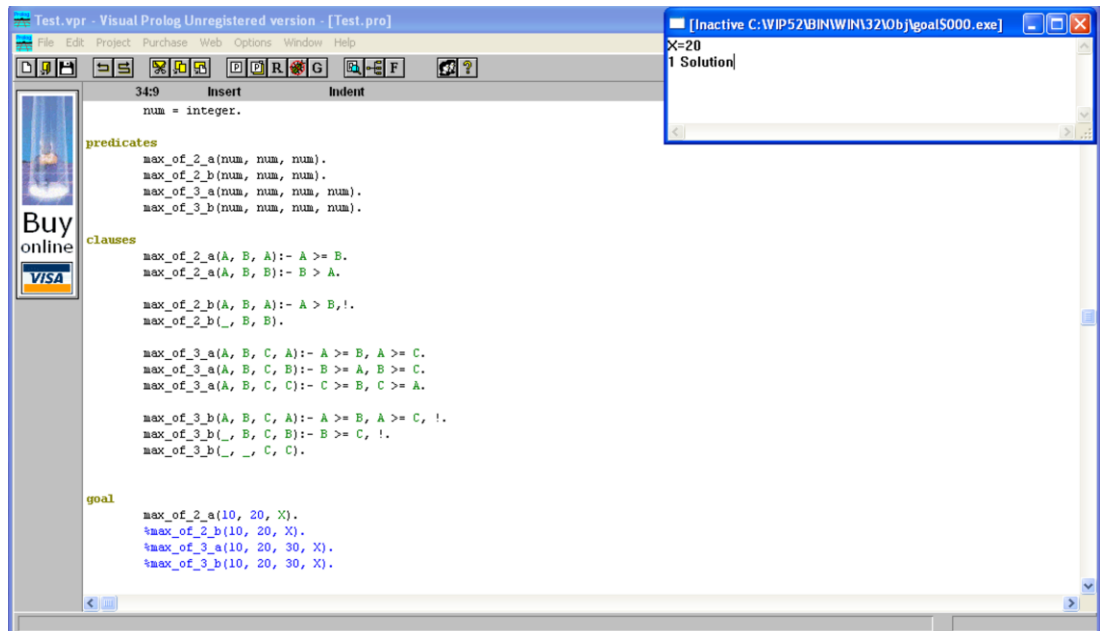
%max_of_2_a(10, 20, X).

%max_of_2_b(10, 20, X).

%max_of_3_a(100, 20, 30, X).

max_of_3_b(100, 20, 30, X).

Результат работы программы:



```
Test.vpr - Visual Prolog Unregistered version - [Test.pro]
File Edit Project Purchase Web Options Window Help
34:9 Insert Indent
num = integer.

predicates
    max_of_2_a(num, num, num).
    max_of_2_b(num, num, num).
    max_of_3_a(num, num, num, num).
    max_of_3_b(num, num, num, num).

clauses
    max_of_2_a(A, B, A):- A >= B.
    max_of_2_a(A, B, B):- B > A.

    max_of_2_b(A, B, A):- A > B, !.
    max_of_2_b(_, B, B).

    max_of_3_a(A, B, C, A):- A >= B, A >= C.
    max_of_3_a(A, B, C, B):- B >= A, B >= C.
    max_of_3_a(A, B, C, C):- C >= B, C >= A.

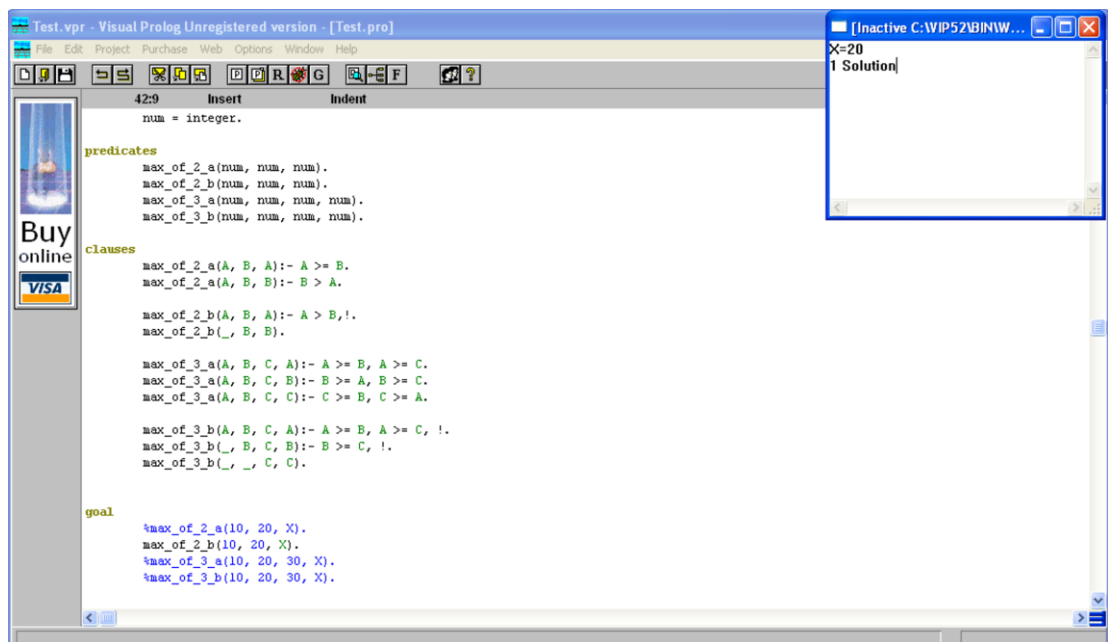
    max_of_3_b(A, B, C, A):- A >= B, A >= C, !.
    max_of_3_b(_, B, C, B):- B >= C, !.
    max_of_3_b(_, _, C, C).

goal
    max_of_2_a(10, 20, X).
    %max_of_2_b(10, 20, X).
    %max_of_3_a(10, 20, 30, X).
    %max_of_3_b(10, 20, 30, X).
```

[Inactive C:\WP52\BIN\WIN32\Obj\goal5000.exe]

X=20
1 Solution

Задание №1 (а)



```
Test.vpr - Visual Prolog Unregistered version - [Test.pro]
File Edit Project Purchase Web Options Window Help
42:9 Insert Indent
num = integer.

predicates
    max_of_2_a(num, num, num).
    max_of_2_b(num, num, num).
    max_of_3_a(num, num, num, num).
    max_of_3_b(num, num, num, num).

clauses
    max_of_2_a(A, B, A):- A >= B.
    max_of_2_a(A, B, B):- B > A.

    max_of_2_b(A, B, A):- A > B, !.
    max_of_2_b(_, B, B).

    max_of_3_a(A, B, C, A):- A >= B, A >= C.
    max_of_3_a(A, B, C, B):- B >= A, B >= C.
    max_of_3_a(A, B, C, C):- C >= B, C >= A.

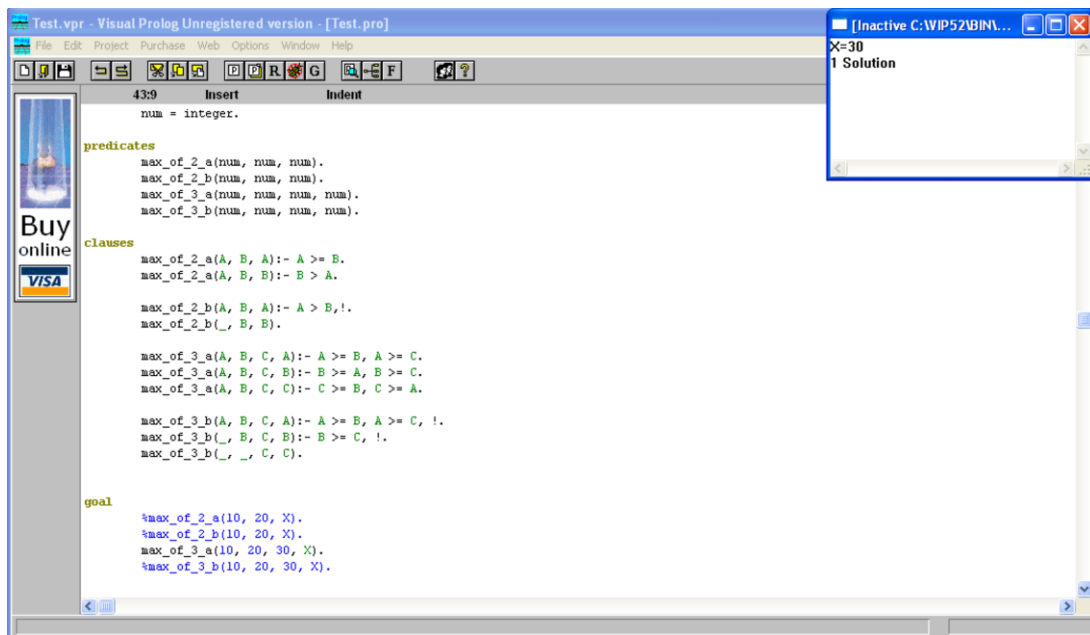
    max_of_3_b(A, B, C, A):- A >= B, A >= C, !.
    max_of_3_b(_, B, C, B):- B >= C, !.
    max_of_3_b(_, _, C, C).

goal
    %max_of_2_a(10, 20, X).
    max_of_2_b(10, 20, X).
    %max_of_3_a(10, 20, 30, X).
    %max_of_3_b(10, 20, 30, X).
```

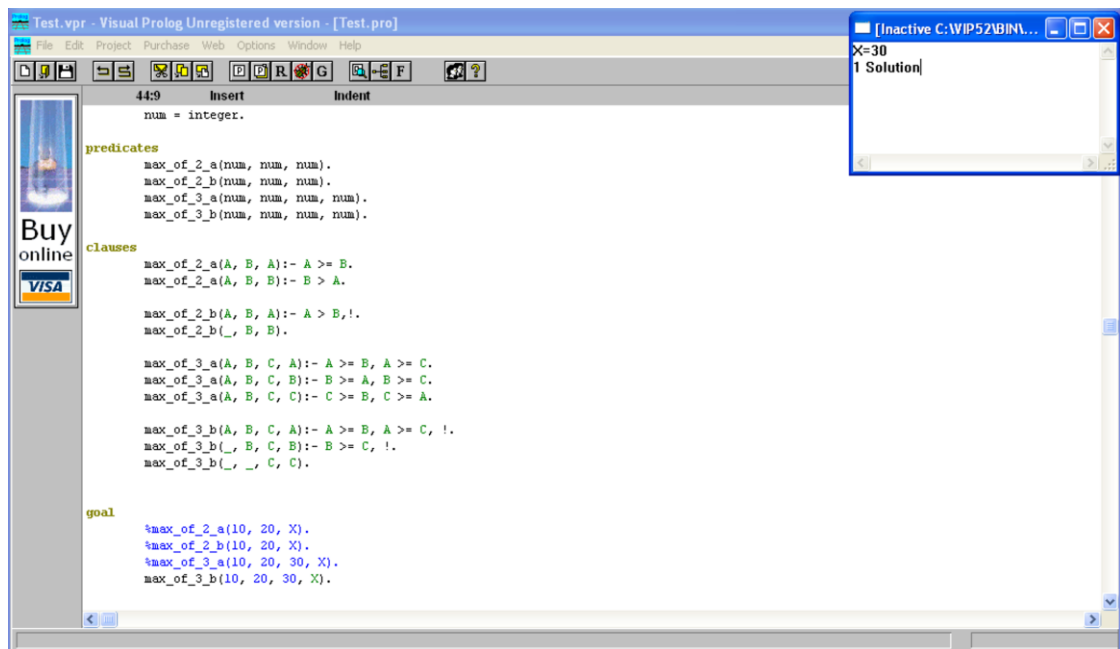
[Inactive C:\WP52\BIN\WIN32\Obj\goal5000.exe]

X=20
1 Solution

Задание №1 (б)



Задание №2 (а)



Задание №2 (б)

Объяснение задания №2 (пункт «а»):

1) $\text{max_of_3_a}(A, B, C, A):- A \geq B, A \geq C.$

Если A – самое большое число из всех 3-х. Также они все могут быть равны, либо $A = B$, либо $A = C$.

2) $\text{max_of_3_a}(A, B, C, B):- B \geq A, B \geq C.$

Если B – самое большое число. Также B может быть равен C .

3) $\text{max_of_3_a}(A, B, C, C):- C \geq B, C \geq A.$

Если C – самое большое число.

Объяснение задания №2 (пункт «б»):

1) $\text{max_of_3_b}(A, B, C, A) :- A \geq B, A \geq C, !.$

Если A – больше или равно B или C .

2) $\text{max_of_3_b}(_, B, C, B) :- B \geq C, !.$

Если B больше C , но также может B может быть равен A .

3) $\text{max_of_3_b}(_, _, C, C).$

Оставшиеся случаи.

Таблица для задания №2 (пункт «а»):

$\text{max_of_3_a}(100, 20, 30, X).$

№ 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 (почему и к чему приводит?)
1	$\text{max_of_3_a}(100, 20, 30, X)$	$T1 = \text{max_of_3_a}(100, 20, 30, X)$ $T2 = \text{max_of_3_a}(A, B, C, A)$ Произошла унификация	Прямой ход
2	$100 \geq 20$ $100 \geq 30$ $(A = 100$ $B = 20$ $C = 30)$	$100 \geq 20$ Верное утверждение	Прямой ход
3	$100 \geq 30$ $(A = 100$ $B = 20$ $C = 30)$	$100 \geq 30$ Верное утверждение	Прямой ход
4	$(A = 100$ $B = 20$ $C = 30)$		Стек пуст. Вывод результата. Прямой ход.

5	max_of_3_a(100, 20, 30, X)	T1 = max_of_3_a(100, 20, 30, X) T2 = max_of_3_a(A, B, C, B) Произошла унификация	Прямой ход
6	20 >= 100 20 >= 30 (A = 100 B = 20 C = 30)	20 >= 100 Неверное утверждение	Прямой ход.
7	max_of_3_a(100, 20, 30, X)	T1 = max_of_3_a(100, 20, 30, X) T2 = max_of_3_a(A, B, C, C) Произошла унификация	Прямой ход
8	30 >= 100 30 >= 20 (A = 100 B = 20 C = 30)	30 >= 100 Неверное утверждение	Прямой ход.
			Конец БЗ. Завершение программы.

Таблица для задания №2 (пункт «б»):

max_of_3_b(100, 20, 30, X).

№ 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 (почему и к чему приводит?)
1	max_of_3_a(100, 20, 30, X)	T1 = max_of_3_a(100, 20, 30, X)	Прямой ход

		$T2 = \max_of_3_a(A, B, C, A)$ Произошла унификация	
2	$100 \geq 20$ $100 \geq 30$ ($A = 100$ $B = 20$ $C = 30$)	$100 \geq 20$ Верное утверждение	Прямой ход
3	$100 \geq 30$ ($A = 100$ $B = 20$ $C = 30$)	$100 \geq 30$ Верное утверждение	Прямой ход
4	($A = 100$ $B = 20$ $C = 30$)		Стек пуст. Вывод результата. Отсечение!
5			Завершение работы программы.

Вывод: с помощью отсечения можно достигнуть увеличения эффективности работы программы, так не выполняются лишние действия.

Теоретическая часть

1) Какое первое состояние резольвенты?

Первое состояние – это вопрос, который задается программе.

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

Унификация запускается для того, чтобы доказать какое-либо утверждение. В этом случае резольвента не пуста.

3) Каково назначение использования алгоритма унификации?

Унификация позволяет формализовать процесс логического вывода.

Назначение - поиск знания, которое является ответом на конкретный вопрос.

4) *Каков результат работы алгоритма унификации?*

Ответ «да» или «нет»

5) *В каких пределах программы переменные уникальны?*

Именованные переменные уникальны в пределах предложения.

Анонимные переменные уникальны в любом месте программы.

6) *Как применяется подстановка, полученная с помощью алгоритма унификации?*

Применение подстановки $\{X_1=T_1, \dots, X_n=T_n\}$ заключается в замене каждого вхождения переменной X_i на соответствующий терм T_i .

7) *Как изменяется резольвента?*

Резольвента меняется по принципу стека (т. е. берется верхняя подцель) во время доказательства утверждения. В ходе доказательства заполняется новая резольвента. Резольвента оказывается пустой при успешном завершении работы программы.

8) *В каких случаях запускается механизм отката?*

Откат происходит в случае тупиковой ситуации или в случае, если резольвента пуста.