Министерство науки и высшего образования РФ

Севастопольский государственный университет

Кафедра информатики и управления в технических системах

ОТЧЕТ

по лабораторной работе №3

СТРУКТУРА ПРОГРАММЫ НА ЯЗЫКЕ ПРОЛОГ. ИНТЕРФЕЙС СРЕДЫ РАЗРАБОТКИ

по дисциплине «Логическое программирование»

Выполнил:

Студент группы ИВТ/б 22-о

Черняев Н.Г.

Проверил:

Ткаченко К.С.

г. Севастополь 2019

**Цель работы**

Изучить основные режимы работы интерфейса VisualProlog, структуру программы и ее разделы. Ознакомиться с понятием «рекурсия».

**Вариант № 23(24, 29, 4)**

Задание:

Выполнить задание согласно вариантам. Для выполнения работы воспользоваться средой VisualProlog, создав консольный проект. Использовать в проекте операции ввода/вывода.

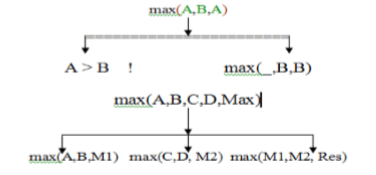
* Вариант 4 - Заданы четыре числа a, b, c и d. Определить максимальное из чисел
* Вариант 24 - Заданы прямая ax+by+c=0 и отрезок своими концами (x1, y1) и (x2, y2). Определить взаимное расположение отрезка и прямой: а) отрезок пересекает прямую; б) отрезок лежит на прямой; в) отрезок по одну сторону от прямой.
* Вариант 29 - Заданы прямые a1\*x+b1\*x+c=0 и a2\*x+b2\*x+c=0. Определить их взаимное расположение: а) пересекаются; б) параллельны; в) совпадают

**Ход работы**

**Задание по варианту 4.**

Программа нахождения максимального из четырёх чисел.

Дерево И/ИЛИ:



Описание структуры предикатов

max : (integer A,integer B,integer C, integer D, integer Res) procedure (i,i,i,i,o).

A,B,C,D – исходные числа, целые;

Res – результат, максимальное из четырёх.

max : (integer A, integer B, integer Res) procedure(i,i,o).

A,B – исходные числа, целые;

Res – максимальное из двух.

Текст программы:

class predicates

max : (integer A,integer B,integer C, integer D, integer Res) procedure (i,i,i,i,o).

max : (integer A, integer B, integer Res) procedure(i,i,o).

clauses

classinfo("L2", "4.0").

max(A,B,C,D,Res):- max(A,B,M1),max(C,D,M2),max(M1,M2,Res). max(A,B,A):- A>B,!.

max(\_,B,B).

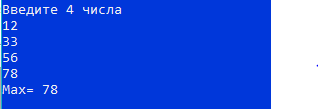
run():- console::init(),stdio::write("Введите 4 числа"), stdio::nl, max(stdio::read(),stdio::read(),stdio::read(),stdio::read(), Max), stdio::write("Мах= ",Max), std io::nl.

end implement main

goal

mainExe::run(main::run).

Результат выполнения программы:



**Задание по варианту 24.**

Программа, которая определяет взаимное расположение отрезка и прямой:

а) отрезок пересекает прямую;

б) отрезок лежит на прямой;

в) отрезок по одну сторону от прямой.

Дерево И/ИЛИ:

solve(A1,B1,C1,X1,Y1,X2,Y2,"Отрезок лежит на прямой")

A1=-(Y2-Y1)

B1= X2-X1

C1=((-Y1)\*(X2-X1)-(-X1)\*(Y2-Y1))

solve(\_,\_,\_,\_,\_,\_,\_,"Отрезок пересекает прямую")

solve(A1,B1,C1,X1,Y1,X2,Y2,"Отрезок по одну сторону от прямой")

A1=-(Y2-Y1)

B1= X2-X1

C1<>((-Y1)\*(X2-X1)-(-X1)\*(Y2-Y1))

Описание структуры предикатов

solve : (integer A1,integer B1,integer C1, integer X1, integer Y1, integer X2, integer Y2, string Ans)

A1,B1,C1,X1,Y1,X2,Y2 – исходные коэффициенты прямых, целые;

Ans – ответ, описывает как прямая и отрезок взаимно расположены.

Текст программы:

class predicates

 solve : (integer A1,integer B1,integer C1, integer X1, integer Y1, integer X2, integer Y2, string Ans) procedure (i,i,i,i,i,i,i,o).

clauses

classinfo("L2", "4.0").

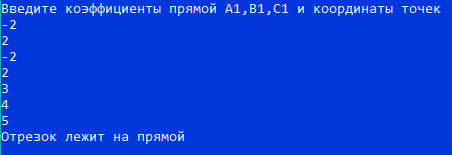
solve(A1,B1,C1,X1,Y1,X2,Y2,"Отрезок лежит на прямой"):- A1=-(Y2-Y1),B1=X2-X1,C1=((-Y1)\*(X2-X1)-(-X1)\*(Y2-Y1)),!.      
solve(A1,B1,C1,X1,Y1,X2,Y2,"Отрезок по одну сторону от прямой"):- A1=-(Y2-Y1),B1=X2-X1,C1<>((-Y1)\*(X2-X1)-(-X1)\*(Y2-Y1)),!.  
solve(\_,\_,\_,\_,\_,\_,\_,"Отрезок пересекает прямую").  
                 
   run():- console::init(),stdio::write("Введите коэффициенты прямой А1,В1,С1 и координаты точек"), stdio::nl,solve(stdio::read(),stdio::read(),stdio::read(),stdio::read(),stdio::read(),stdio::read(),stdio::read(), Max), stdio::write(Max), stdio::nl.

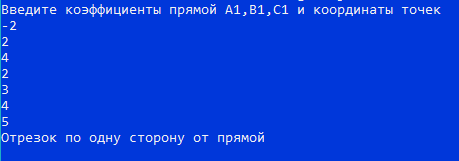
end implement main

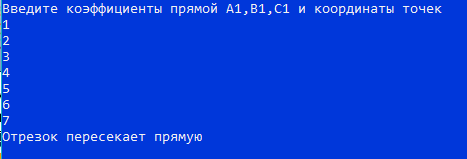
goal

mainExe::run(main::run).

Результат выполнения программы:







**Задание по варианту 29.**

Программа, которая определяет взаимное расположение 2 прямых:

а) пересекаются;

б) параллельны;

в) совпадают.

Дерево И/ИЛИ:

solve : (A1,B1,C1,A2,B2,C2,"Совпадают")

A1=A2

B1=B2

C1=C2

solve : (\_,\_,\_,\_,\_,\_,"Пересекаются ")

solve : (A1,B1,C1,A2,B2,C2," Параллельны ")

A1=A2

B1=B2

C1<>C2

Описание структуры предикатов

solve : (integer A1,integer B1,integer C1, integer A2, integer B2, integer C2, string Ans) procedure (i,i,i,i,i,i,o).

A1,B1,C1,A2,B2,C2 – исходные коэффициенты прямых, целые;

Ans – ответ, описывает как прямые взаимно расположены.

Текст программы:

class predicates

solve : (integer A1,integer B1,integer C1, integer A2, integer B2, integer C2, string Ans) procedure (i,i,i,i,i,i,o).

clauses

classinfo("L2", "4.0").

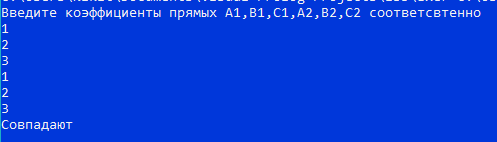
solve(A1,B1,C1,A2,B2,C2,"Совпадают"):- A1=A2,B1=B2,C1=C2,!.      
solve(A1,B1,C1,A2,B2,C2,"Параллельны"):- A1=A2,B1=B2,C1<>C2,!.  
solve(\_,\_,\_,\_,\_,\_,"Пересекаются").  
                 
   run():- console::init(),stdio::write("Введите коэффициенты прямых А1,В1,С1,А2,В2,С2 соответсвтенно"), stdio::nl,solve(stdio::read(),stdio::read(),stdio::read(),stdio::read(),stdio::read(),stdio::read(), Max), stdio::write(Max), stdio::nl.

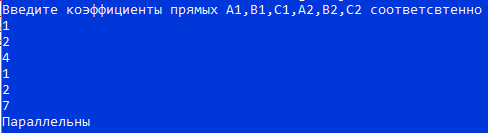
end implement main

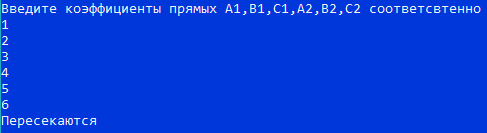
goal

mainExe::run(main::run).

Результат выполнения программы:







**Выводы**

Вданной лабораторной работе были изучены основные режимы работы интерфейса VisualProlog, структура программы и ее разделы. Были получены знания о понятии «рекурсия».