**Лабораторная работа №3.   
Синтаксический разбор с использованием метода рекурсивного спуска**

Студент ИУ7-22М Лахов А.Е. Вариант 1

**Задание**

Рассматривается грамматика выражений отношения с правилами:

<выражение> –>

<простое выражение> |

<простое выражение> <операция отношения> <простое выражение>

<простое выражение> –>

<терм> |

<знак> <терм> |

<простое выражение> <операция типа сложения> <терм>

<терм> –>

<фактор> |

<терм> <операция типа умножения> <фактор>

<фактор> –>

<идентификатор> |

<константа> |

( < простое выражение > ) |

**not** <фактор>

<операция отношения> –>

= | <> | < | <= | > | >=

<знак> –>

+ | -

<операция типа сложения> –>

+ | - | **or**

<операция типа умножения> –>

\* | / | **div** | **mod** | **and**

Дополнить грамматику блоком, состоящим из последовательности операторов присваивания. Вариант расширенной грамматики в стиле Алгол-Паскаль.

<программа> –>

<блок>

<блок> –>

**begin** <список операторов> **end**

<список операторов> –>

<оператор>; | <список операторов> <оператор>;

<оператор> –>

<идентификатор> := <выражение>

Данный вариант расширенной грамматики содержит левую рекурсию, которая должна быть устранена. Последнее правило расширенной грамматики было изменено: было добавлено двоеточие перед знаком «=».

**Преобразование исходной грамматики**

Сделаем преобразование исходной грамматики для ухода от левой рекурсии.

<выражение> –>

<простое выражение> |

<простое выражение> <операция отношения> <простое выражение>

<простое выражение> –>

<терм> <*простое выражение’* > |

<знак> <терм> <*простое выражение’* >

<*простое выражение’* > –>

<операция типа сложения> <терм> <*простое выражение’* > | ε

<терм> –>

<фактор> <*терм’* >

<*терм’* > –>

<операция типа умножения> <фактор> <*терм’* > | ε

<фактор> –>

<идентификатор> |

<константа> |

( < простое выражение > ) |

**not** <фактор>

<операция отношения> –>

= | <> | < | <= | > | >=

<знак> –>

+ | -

<операция типа сложения> –>

+ | - | **or**

<операция типа умножения> –>

\* | / | **div** | **mod** | **and**

<программа> –>

<блок>

<блок> –>

**begin** <список операторов> **end**

<список операторов> –>

<оператор>; <*список операторов’* >

<*список операторов’* > –>

<оператор>; <*список операторов’* > | ε

<оператор> –>

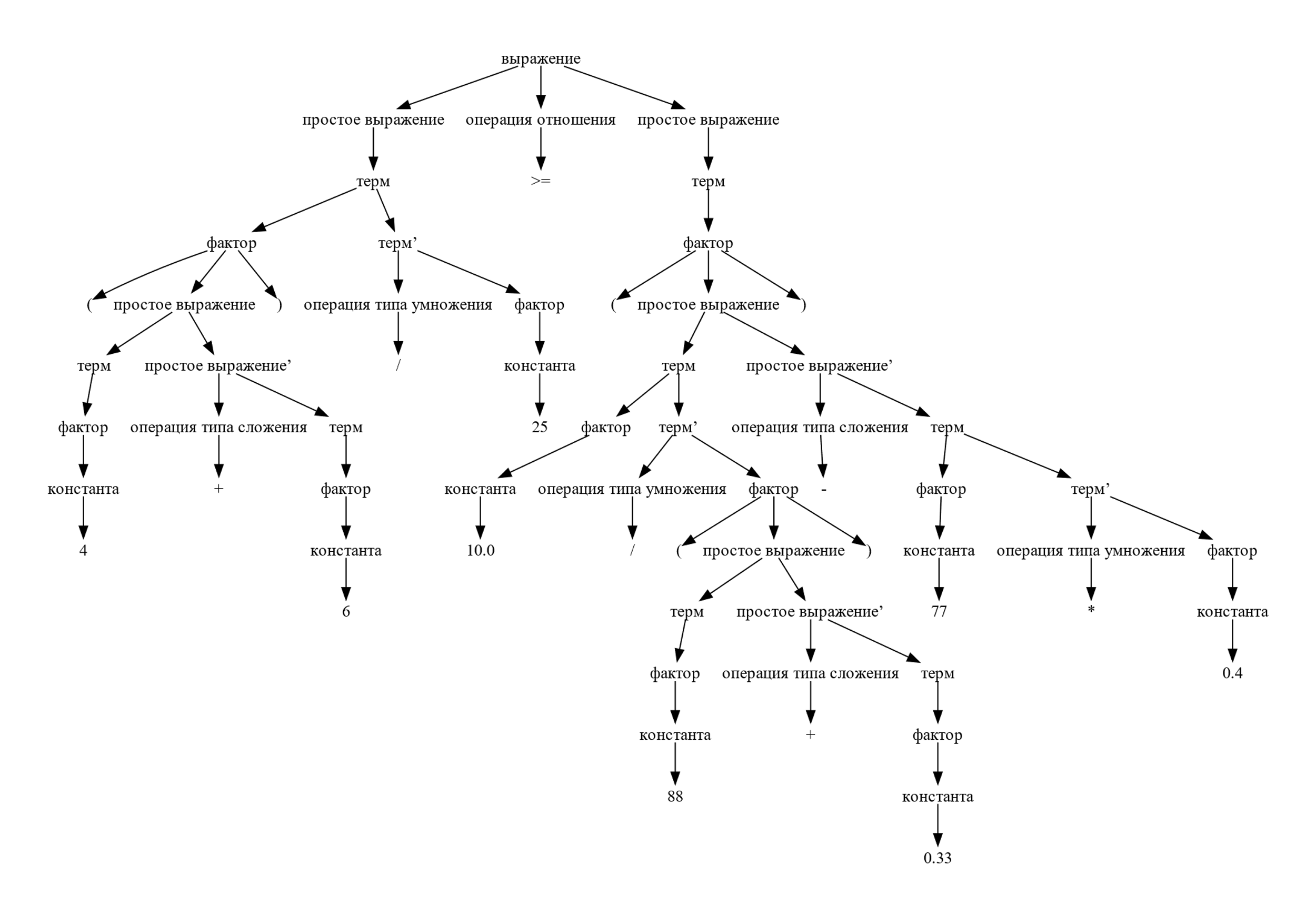
<идентификатор> := <выражение>

**Тестирование**

1) ввод:

*(4 + 6) / 25 >= (10.0/(88 + 0.33)-77\*0.4)*

Результат:



2) ввод:

*begin*

*a := 43.0 + 39;*

*val := (4 + 6) / 25 >= (10.0/(88 + 0.33)-77\*0.4)*

*end*

Результат:

