|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **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**

**по Лабораторной работе №3**

**по курсу**

**«Проектирование компиляторов»**

**Тема**

**«Синтаксический разбор с использованием метода рекурсивного спуска»**

**Вариант 1**

Студент \_\_\_ИУ7-21М\_\_\_\_ \_\_\_\_\_Карпухин А.С.\_\_\_\_

(Группа) (И.О.Фамилия)

Преподаватель \_\_\_\_Ступников А.А.\_\_\_\_

(И.О.Фамилия)

*2021 г.*

**Задание**

**Задачи работы:**

1. Дополнить грамматику блоком, состоящим из последовательности операторов присваивания.
2. Для модифицированной грамматики написать программу нисходящего синтаксического анализа с использованием метода рекурсивного спуска.

**Подготовка грамматики**

Листинг 1. Исходная грамматика

|  |
| --- |
| <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  <идентификатор> -> “[\_a-zA-Z][\_a-zA-Z0-9]{0,30}”  <константа> -> “-?[0-9]+|\".\*\"|\'.\*\'” |

Листинг 2. Дополненная грамматика

|  |
| --- |
| <программа> ->  <блок>  <блок> ->  { <список операторов> }  <список операторов> ->  <оператор> <хвост>  <хвост> ->  ; <оператор> <хвост> | ε  <оператор> ->  <идентификатор> = <выражение> |  <блок>  <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  <идентификатор> -> “[\_a-zA-Z][\_a-zA-Z0-9]{0,30}”  <константа> -> “-?[0-9]+|\".\*\"|\'.\*\'” |

Листинг 3. Дополненная грамматика без левой рекурсии

|  |
| --- |
| <программа> ->  <блок>  <блок> ->  { <список операторов> }  <список операторов> ->  <оператор> <хвост>  <хвост> ->  ; <оператор> <хвост> | ε  <оператор> ->  <идентификатор> = <выражение> |  <блок>  <выражение> ->  <простое выражение> |  <простое выражение> <операция отношения> <простое выражение>  <простое выражение> ->  <терм> |  <знак> <терм> |  <терм> <простое выражение 1> |  <знак> <терм> <простое выражение 1>  <простое выражение 1> ->  <операция типа сложения> <терм>  <операция типа сложения> <терм> <простое выражение 1>  <терм> ->  <фактор> |  <фактор> <терм 1>  <терм 1> ->  <операция типа умножения> <фактор> |  <операция типа умножения> <фактор> <терм 1>  <фактор> ->  <идентификатор> |  <константа> |  ( < простое выражение > ) |  not <фактор>  <операция отношения> ->  == | <> | < | <= | > | >=  <знак> ->  + | -  <операция типа сложения> ->  + | - | or  <операция типа умножения> ->  \* | / | div | mod | and  <идентификатор> -> “[\_a-zA-Z][\_a-zA-Z0-9]{0,30}”  <константа> -> “-?[0-9]+|\".\*\"|\'.\*\'” |

**Пример работы программы**

Листинг 4. Входные данные для разбора

|  |
| --- |
| {  f = t + r - ( a and r ) ;  {  w = - q or ( + t )  }  } |

Дерево, полученное в результате разбора методом рекурсивного спуска с откатами для приведенного в листинге 4 кода, изображено на рисунке 1.

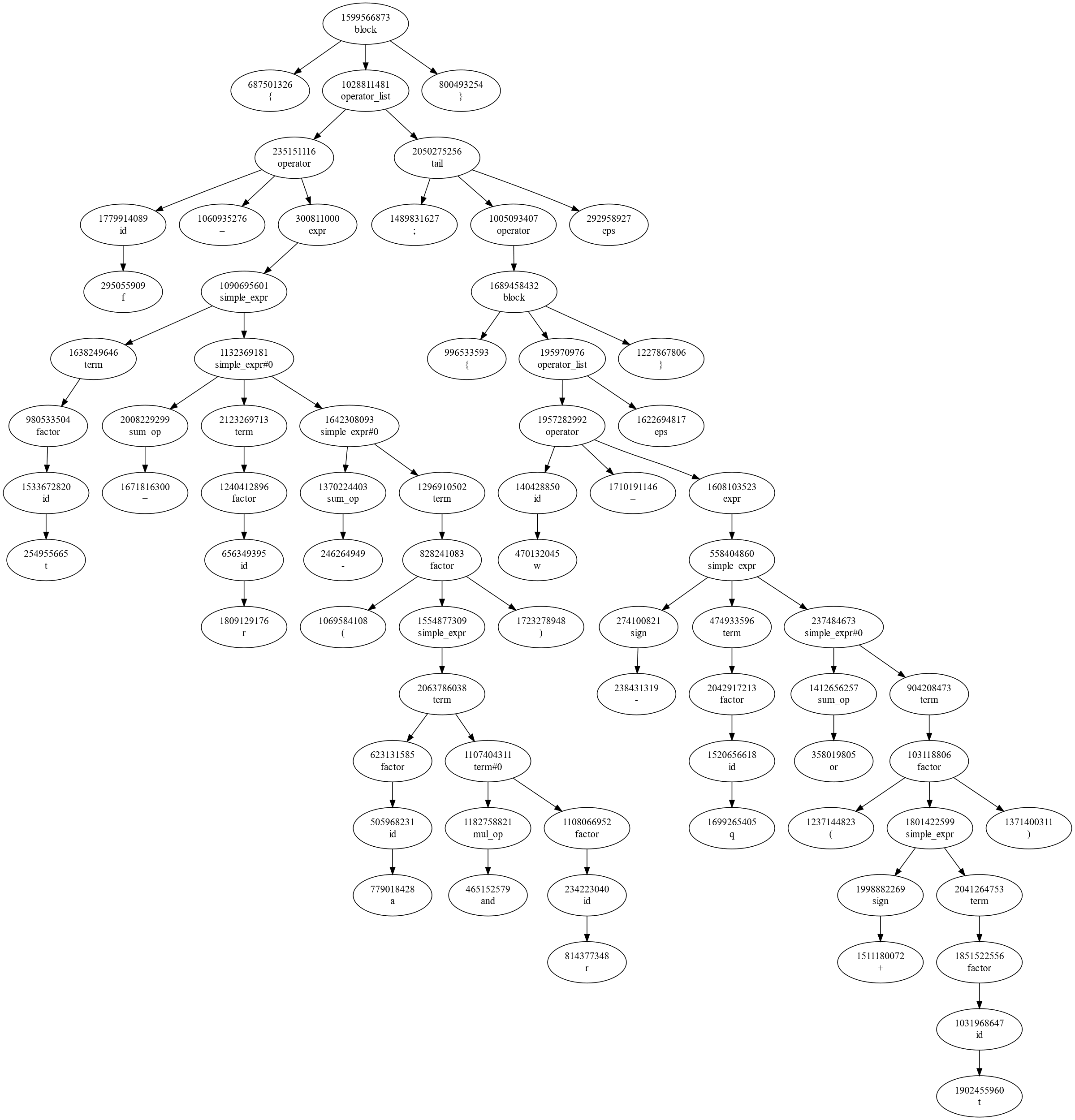


Рисунок 1 – Дерево разбора, построенное для листинга 4.

**Заключение**

В ходе выполнения работы были изучены методы синтаксического анализа, в частности, метод рекурсивного спуска, применяемый в данной работе.

Для исходной грамматики была построена новая, дополняющая исходную конструкциями, описывающими блоки с последовательностями присваивания в стиле языка Си. Из полученной грамматики была также удалена левая рекурсия.

На основе описания рассмотренного метода была реализована программа для синтаксического анализа методом рекурсивного спуска с откатами для полученной дополненной грамматики без левой рекурсии.