Лексический анализатор

Выполнил: Каланов Андрей, ПС-31

**Назначение программы:**

Лексический анализ — процесс аналитического разбора входной последовательности символов на распознанные группы — лексемы, с целью получения на выходе идентифицированных последовательностей, называемых «токенами».   
Токенизаторы классифицируют лексемы по различным типам («идентификатор, оператор», «часть речи» и т. п.). Лексический анализ используется в компиляторах и интерпретаторах языков программирования.

Анализатор распознает лексемы в исходном коде и выдает их список.

**Вход:**

На вход программе посылается input и output .txt файлы, название файлов может быть любым. В input файле подается текст программы.

**Выход:**

На выходе получается список токенов.

**Структура токена:**

Лексема, тип лексемы, номер строки в тексте, номер позиции в тексте.

**Список токенов:**

1. Ошибка: неизвестный токен
2. Идентификатор: буквы, “\_”, цифры. Идентификатор не может начинаться с цифры
3. Целые числа: целое число (7)
4. Числа с фиксированной длинной: десятичное число (5.4)
5. Числа с плавающей длинной: десятичное число со степенью (5.0E-6)
6. Символ (char): ’’
7. Строки (string): "”
8. Массивы: "[]"
9. Двоичная система счисления: '0', '1'
10. Восьмеричная система счисления: '0', '1', '2', '3', '4', '5', '6', '7'
11. Шестнадцатеричная система счисления: '0', '1', '2', '3', '4', '5', '6', '7', '8', '9', 'A', 'B', 'C', 'D', 'E', 'F'
12. Операторы (ключевые слова языка): "if", "else", "for", "while", "do", "break", "continue", "return", "read", "write", "main", "void", "int", "float", "double", "char", "string", "bool"
13. Арифметические операции: '+', '-', '\*', '/', '=', '%'
14. Операции сравнения: "==", "!=", ">", "<", ">=", "<=", "+=", "-=", "\*=", "/=", "%=", "&=", "|=", "^=", "<<=", ">>="
15. Логические операции: "&&", "||", "!"
16. Битовые операции: "~", "&", "|", "^", "<<", ">>"
17. Скобки: '{', '}', '(', ')'
18. Комментарии: "//", "/\*", "\*/"
19. Разделители: ',', ';', ':'