

Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет
информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

Кафедра информатики

Дисциплина: Методы трансляции

ОТЧЁТ
по лабораторной работе
на тему

Синтаксический анализ

Выполнил
Студент гр. 053502
Шаргородский И.С.

Проверил
Ассистент кафедры информатики
Гриценко Н.Ю.

Минск 2023

СОДЕРЖАНИЕ

1 Цель работы	3
2 Краткие теоретические сведения.....	4
3 Демонстрация работы.....	5
3.1 Результаты работы	5
3.2 Синтаксические ошибки.....	6
4 Выводы	7
Приложение А (информационное)_Код программ.....	8

1 ЦЕЛЬ РАБОТЫ

Освоение работы с существующими синтаксическими анализаторами. Разработать свой собственный синтаксический анализатор, выбранного подмножества языка программирования.

Построить синтаксическое дерево.

Определить минимум 4 возможных синтаксических ошибки и показать их корректное выявление.

Основной целью работы является написание сценариев, которые задают синтаксические правила для выбранного подмножества языка.

2 КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Синтаксический анализ — это процесс сопоставления линейной последовательности лексем (слов, токенов) естественного или формального языка с его формальной грамматикой. Результатом обычно является дерево разбора (синтаксическое дерево). Обычно применяется совместно с лексическим анализом.

Синтаксический анализатор — это программа или часть программы, выполняющая синтаксический анализ.

Как правило, результатом синтаксического анализа является синтаксическое строение предложения, представленное либо в виде дерева зависимостей, либо в виде дерева составляющих, либо в виде некоторого сочетания первого и второго способов представления.

Таким образом на основе анализа выражений, состоящих из литералов, операторов и круглых скобок выполняется группирование токенов исходной программы в грамматические фразы, используемые для синтеза вывода.

3 ДЕМОНСТРАЦИЯ РАБОТЫ

3.1 Результаты работы

Рассмотрим результат синтаксического анализа тестовой программы (см. приложение А) программой-анализатором (см. рисунок 1)

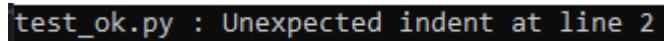
```
=
    num
    +
        int
            input
                Enter a number:
            int
                input
                    Enter a number:
=
    factorial
    +
        +
            2
            *
                2
                2
if
    <
        num
        0
    print
        Sorry, factorial does not exist for negative numbers
elif
    ==
        num
        0
    print
        The factorial of 0 is 1
else
    for
        in
            i
            range
                1
                num
            =
                factorial
                *
                    factorial
                    i
        print
            The factorial of
            num
            is
            factorial
```

Рисунок 1 – Таблица ключевых слов

3.2 Синтаксические ошибки

Ошибка неожиданного углубления – производится, когда анализатор встречает слово, которое находится, неожиданно, на новой глубине. Результат анализа ошибки представлен на рисунке 2. Входная программа:

```
num = int(input("Enter a number: ")) + int(input("Enter a number: "))  
    factorial = 2 + (2 * 2) + 2
```

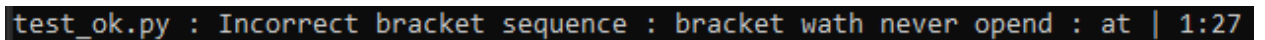


```
test_ok.py : Unexpected indent at line 2
```

Рисунок 2 – Пример ошибки неожиданного углубления

Ошибка скобочной последовательности – производится, когда анализатор встречает некорректную скобочную последовательность. Результат анализа ошибки представлен на рисунке 3. Входная программа:

```
factorial = 2 + (2 * 2) + 2)
```

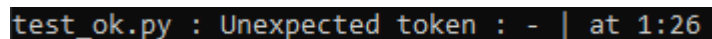


```
test_ok.py : Incorrect bracket sequence : bracket wath never open : at | 1:27
```

Рисунок 3 – Пример ошибки скобочной последовательности

Ошибка некорректного выражения – производится, когда анализатор встречает выражение, которое содержит операнды или операторы в некорректных местах. Результат анализа ошибки представлен на рисунке 4. Входная программа:

```
factorial = 2 + (2 * 2) + - 2
```

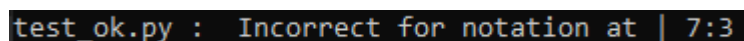


```
test_ok.py : Unexpected token : - | at 1:26
```

Рисунок 4 – Пример ошибки некорректного выражения

Ошибка некорректного определения for – производится, когда анализатор встречает некорректно определенный цикл for. Результат анализа ошибки представлен на рисунке 5. Входная программа:

```
for i < range(1,num):
```



```
test_ok.py : Incorrect for notation at | 7:3
```

Рисунок 5 – Пример ошибки незнакомого токена

4 ВЫВОДЫ

Таким образом, в ходе лабораторной работы было изучено понятие синтаксического анализа в теории трансляции. Был разработан собственный синтаксический анализатор выбранного подмножества языка программирования.

ПРИЛОЖЕНИЕ А
(информационное)
Код программ

1. Тестовая программа с ошибками

```
num = int(input("Enter a number: "))
factorial = 2 + (2 * 2) + - 2
if num < 0:
    print("Sorry, factorial does not exist for negative numbers")
elif num == 0:
    print("The factorial of 0 is 1")
else:
    for i < range(1j1,number + 1.1.2j):
        factorial = factorial*i
    print("The factorial of", num, "is", factorial)
```

2. Тестовая программа

```
num = int(input("Enter a number: "))
factorial = 2 + (2 * 2) + 2
if num < 0:
    print("Sorry, factorial does not exist for negative numbers")
elif num == 0:
    print("The factorial of 0 is 1")
else:
    for i in range(1,number + 1):
        factorial = factorial*i
    print("The factorial of", num, "is", factorial)
```