

# 编译原理实践第5次课

## SLR(0)分析算法

张昊 1927405160

[概述](#)

[编程环境说明](#)

[运行](#)

## 概述

使用 Python3 实现了SLR(0)分析算法，包括 基于SLR(0)分析算法构建可行前缀的DFA，基于DFA构建预测分析表，基于预测分析表结合栈进行语法解析。

给定文法：

```
E -> E + T
E -> T
T -> T * F
T -> F
F -> (E)
F -> id
```

## 编程环境说明

- 语言：Python3
- 文件编码：UTF-8
- 测试环境：Python 3.8.10

## 运行

项目源文件有三：`main.py` 主程序、`parser.py` 解析器类、`util.py` 工具类。

主程序接受两个参数，第一个参数是保存了文法的文本文件，其格式如下：

```
产生式头部 -> 产生式尾部
```

其中文法中不会出现中括号。

第二个参数是保存有输入字符串的文本文件，其中各词法单元之间使用空格隔开，且只有一行。

运行方法：

```
$ python3 main.py grammar.txt tokens.txt
```

其中 `grammar.txt` 文件如下：

```
E -> E + T
E -> T
T -> T * F
T -> F
F -> ( E )
F -> id
```

其中 `tokens.txt` 文件如下:

```
id + ( id + id ) * id
```

输出如下:

```
[E [E [T [F [id]]]] + [T [T [F ( [E [E [T [F [id]]]] + [T [F [id]]]] )]] * [F [id]]]]
```

```
PS D:\Code\compiler\slr> python main.py grammar.txt tokens.txt
[E [E [T [F [id]]]] + [T [T [F ( [E [E [T [F [id]]]] + [T [F [id]]]] )]] * [F [id]]]]
```

将该字符串形式生成树状图如下:

