# 词法分析程序设计报告

## 需求分析

开发环境：Windows10 & WSL (Ubuntu 20.04.1 LTS) & VSCode & Remote – WSL & g++ (Ubuntu 9.3.0-10ubuntu2) 9.3.0

功能需求：设计语法分析程序

1. 由LL1文法自动构造预测分析表
2. 通过LL1预测分析表解析句子
3. 通过给定的LR分析表解析句子

## 总体设计

1. 数据结构设计：
   1. using Terminator = std::string;
   2. using NonTerminator = std::string;
   3. using Symbol = std::string;
   4. using State = size\_t;
   5. using SententialForm = std::vector<Symbol>;
   6. using Production = std::pair<NonTerminator, SententialForm>;
   7. using LRTable = std::vector<std::unordered\_map<Symbol, std::string>>;
   8. using LL1Table = std::unordered\_map<NonTerminator, std::unordered\_map<Terminator, Production>>;
2. 总体结构设计：
   1. LL1TableCreator类由给定产生式构造LL1预测分析表
   2. LL1Parser类通过LL1预测分析表解析句子
   3. LRParser类通过LR预测分析表解析句子
3. 用户接口设计：
   1. LL1TableCreator(const std::vector<Production> &productions,

                    const NonTerminator &startNonTerminator);

通过给定的产生式和起始非终结符初始化LL1TableCreator

* 1. LL1Table getLL1Table() const;

获得LL1预测分析表

* 1. const NonTerminator getStartNonTerminator() const;

获得起始非终结符

* 1. LL1Parser(const LL1Table &table, const NonTerminator &startNonT

erminator);

通过给定的LL1预测分析表和起始非终结符初始化语法分析器

* 1. LRParser(const LRTable &table, const std::vector<Production> &pr

oductions);

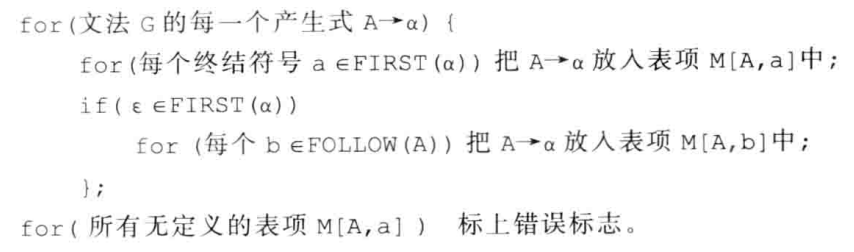
通过给定的LR预测分析表和产生式初始化语法分析器

* 1. bool parse(std::string sentence);

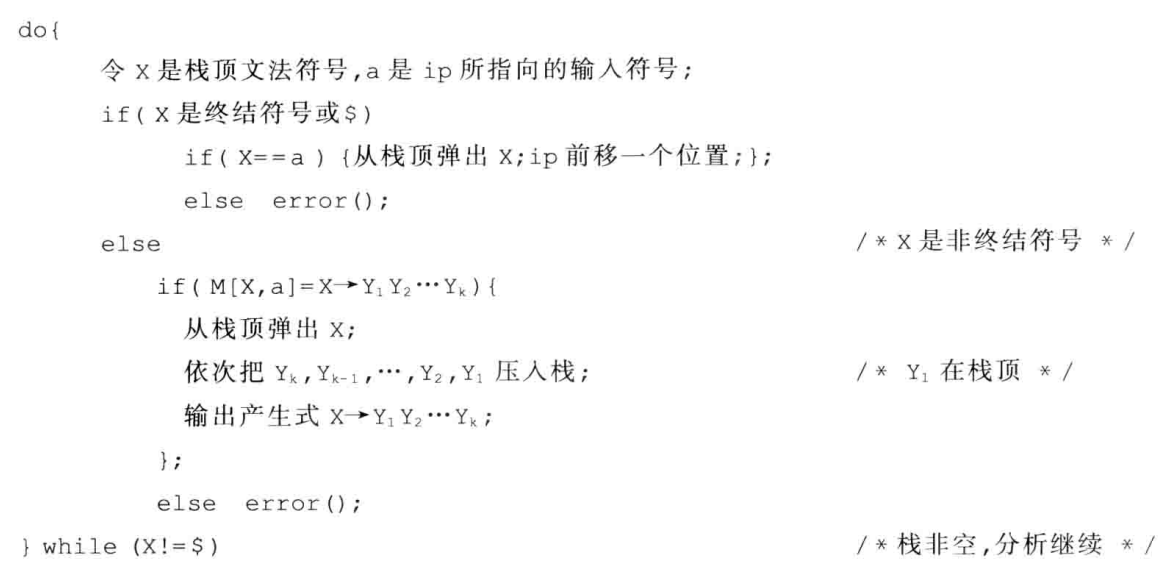
进行语法分析

## 详细设计

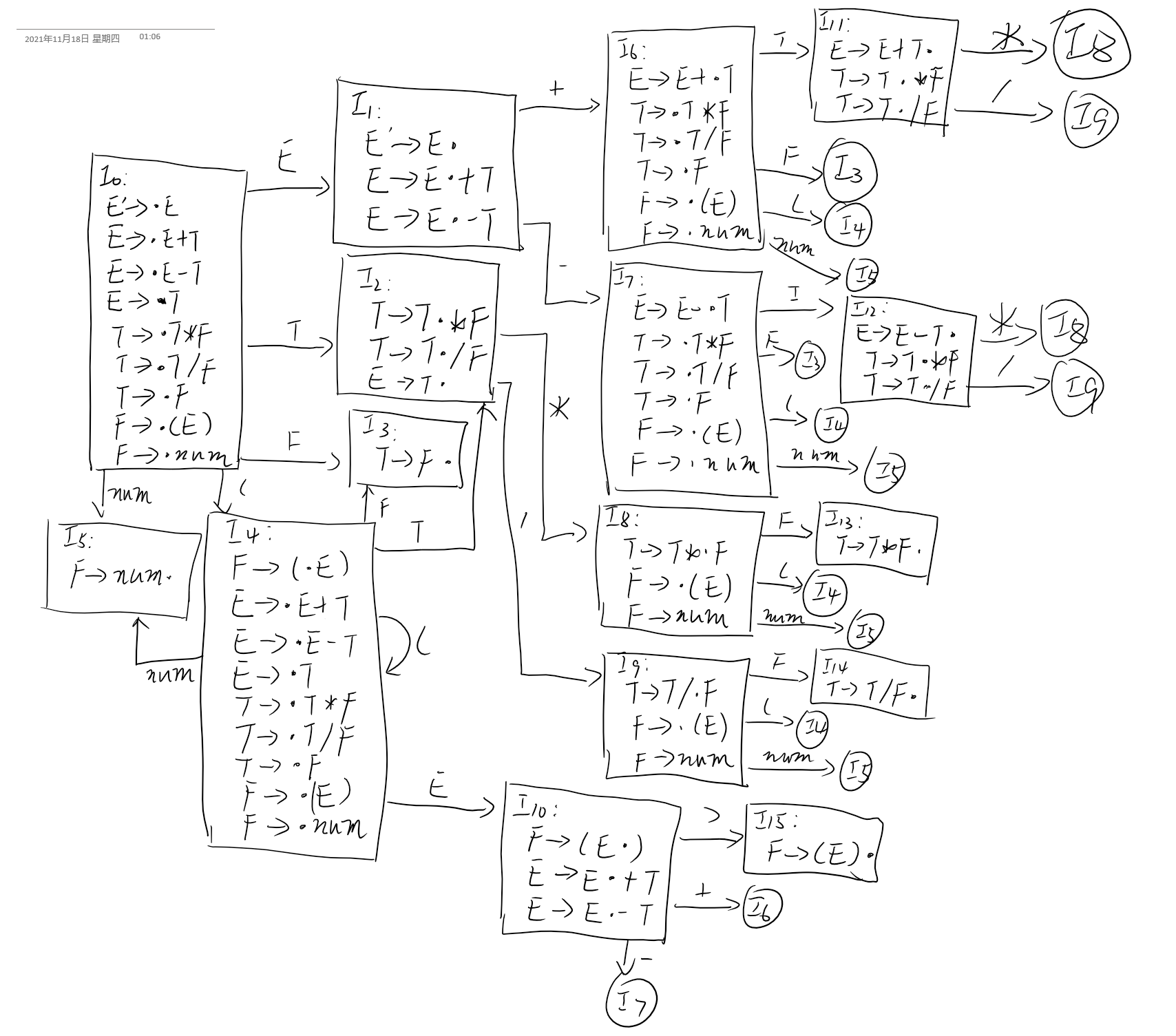
1. LL1TableCreator



1. LL1Parser



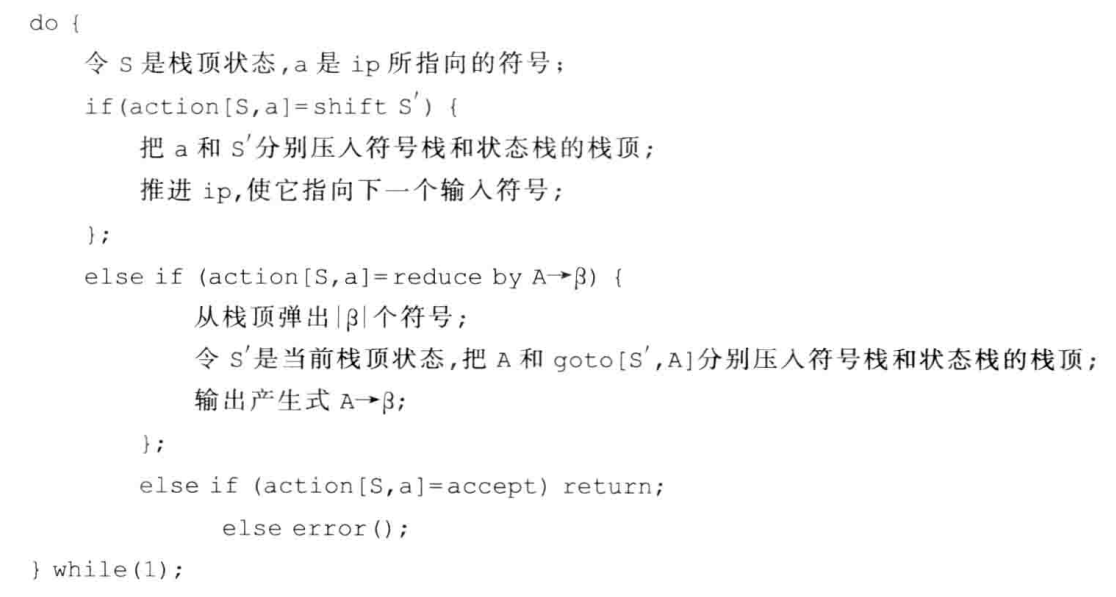
1. LRParser
   1. 识别该文发所有活前缀的DFA



* 1. 该文法的LR分析表
     1. E’->E
     2. E->E+T
     3. E->E-T
     4. E->T
     5. T->T\*F
     6. T->T/F
     7. T->F
     8. F->(E)
     9. F->num

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| STATUS | ACTION | | | | | | | | GOTO | | |
| + | - | \* | / | ( | ) | num | $ | E | T | F |
| 0 |  |  |  |  | s4 |  | s5 |  | 1 | 2 | 3 |
| 1 | s6 | s7 |  |  |  |  |  | acc |  |  |  |
| 2 | r3 | r3 | s8 | s9 | r3 | r3 | r3 | r3 |  |  |  |
| 3 | r6 | r6 | r6 | r6 | r6 | r6 | r6 | r6 |  |  |  |
| 4 |  |  |  |  | s4 |  | s5 |  | 10 | 2 | 3 |
| 5 | r8 | r8 | r8 | r8 | r8 | r8 | r8 | r8 |  |  |  |
| 6 |  |  |  |  | s4 |  | s5 |  |  | 11 | 3 |
| 7 |  |  |  |  | s4 |  | s5 |  |  | 12 | 3 |
| 8 |  |  |  |  | s4 |  | s5 |  |  |  | 13 |
| 9 |  |  |  |  | s4 |  | s5 |  |  |  | 14 |
| 10 | s6 | s7 |  |  |  | s15 |  |  |  |  |  |
| 11 | r1 | r1 | s8 | s9 | r1 | r1 | r1 | r1 |  |  |  |
| 12 | r2 | r2 | s8 | s9 | r2 | r2 | r2 | r2 |  |  |  |
| 13 | r4 | r4 | r4 | r4 | r4 | r4 | r4 | r4 |  |  |  |
| 14 | r5 | r5 | r5 | r5 | r5 | r5 | r5 | r5 |  |  |  |
| 15 | r7 | r7 | r7 | r7 | r7 | r7 | r7 | r7 |  |  |  |

* 1. LR分析程序



## 源程序清单

源程序：

include/LL1TableCreator.h

src/LL1TableCreator.cpp

include/LL1Parser.h

src/LL1Parser.cpp

include/LRParser.h

src/LRParser.cpp

main.cpp

构建：mkdir bin && cd bin && cmake .. && make

运行：./Parser

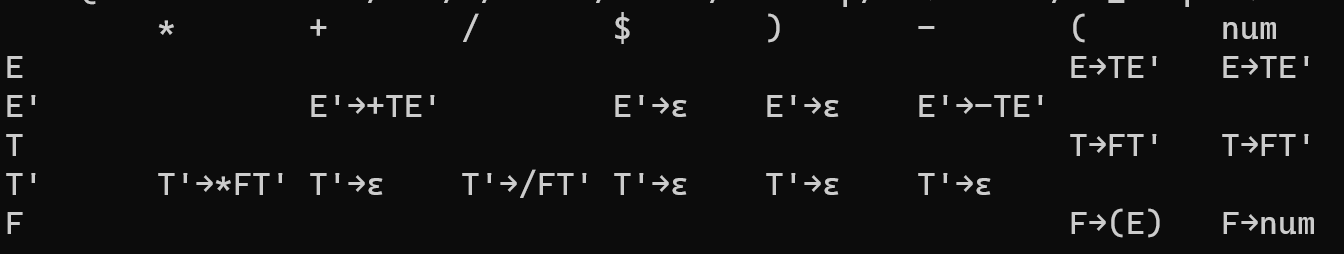
## 程序测试

测试环境：Windows10 WSL (Ubuntu 20.04.1 LTS)

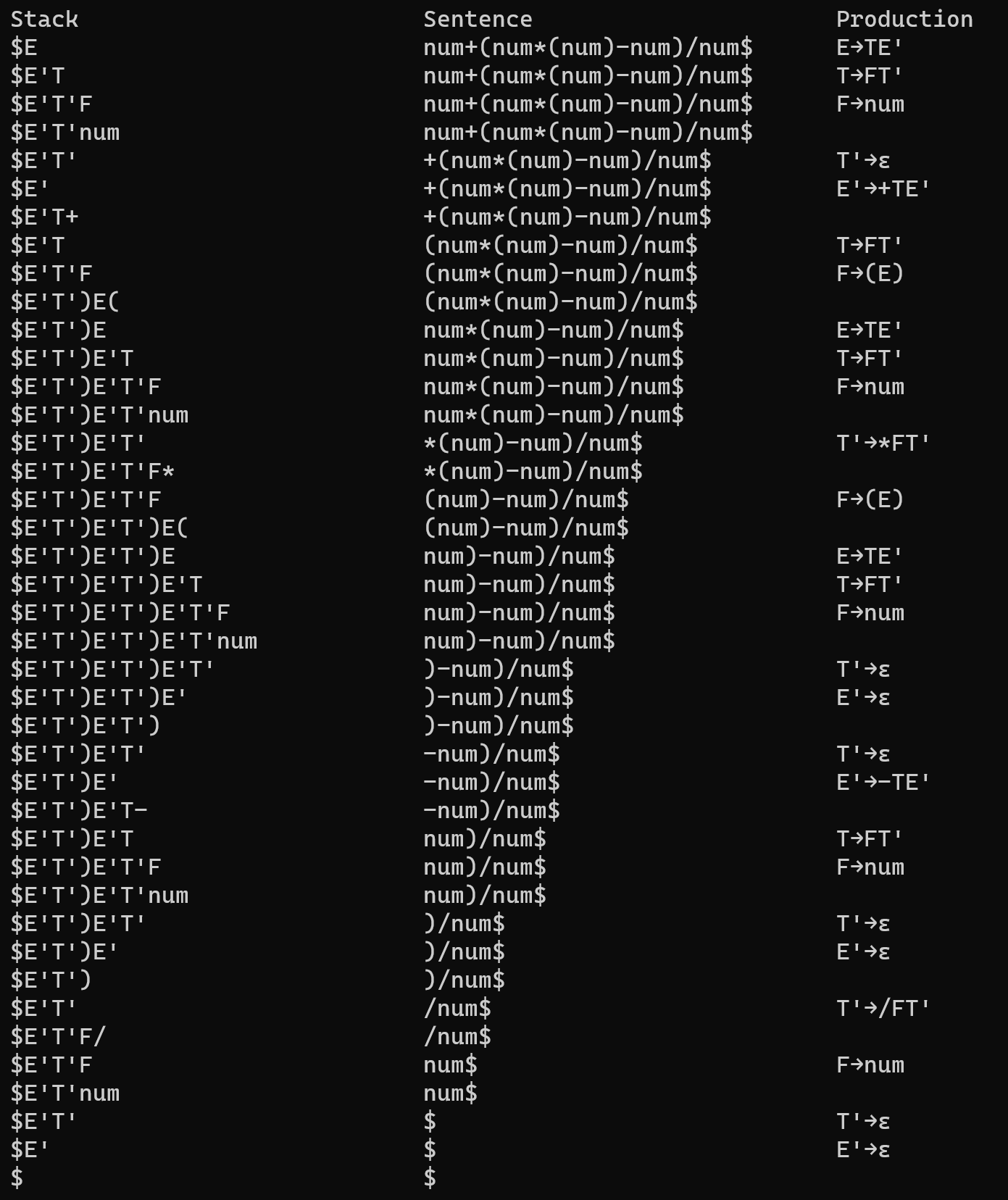
测试的功能：LL1预测分析表构造，LL1分析，LR分析

测试结果：

LL1预测分析表：



LL1分析：



LR分析：

