# 设计分工：

组长学号及姓名：20184317彭明均

分工：文法设计、数据结构设计、语法分析（LL1）、基于DAG的中间代码优化、部分目标代码生成（if，while等翻译框架）

组员1学号及姓名：20184454谢一帆

分工：文法设计、符号表组织、部分目标代码生成（符号表相关的，目标代码前定义数据，数据代码堆栈段分配，活动记录）

组员2学号及姓名：20184540梁俊涛

分工：文法设计、中间代码生成、部分目标代码生成（逻辑运算、赋值、输入输出）、分析栈

组员3学号及姓名：20184390杨子江

分工：文法设计、词法分析、部分目标代码生成（算术运算）

# 文法：

1.    <函数定义> -><类型><标识符> ( <参数声明> ) { <函数块> }

2.    <类型> ->int | float | char | void | $

3.    <因式> -> ( <表达式> ) | <标识符> | <数字>  |<字符>

4.    <表达式> -><因子><项>

5.    <因子> -><因式><因式递归>

6.    <因式递归> -> \* <因式><因式递归> | / <因式><因式递归> | $

7.    <项> -> + <因子><项> | - <因子><项> | $

8.    <参数声明> -><声明><声明闭包> | $

9.    <声明> -><类型><标识符><赋初值>  |<标识符><赋初值>

10.    <赋初值> -> = <右值> | $

11.    <右值> -><表达式>

12.    <声明闭包> -> , <声明><声明闭包> | $

13.    <函数块> -><声明语句闭包><函数块闭包>

14.    <声明语句闭包> -><声明语句><声明语句闭包> | $

15.    <声明语句> -><声明> ;

16.    <函数块闭包> -><赋值函数><函数块闭包> | <while循环><函数块闭包> | <条件语句><函数块闭包> | <函数返回><函数块闭包> |<cout语句><函数块闭包>|<cin语句><函数块闭包>| $

17.    <赋值函数> -><标识符><赋值或函数调用>

18.    <赋值或函数调用> -> = <右值> ; | ( <参数列表> ) ;

19.    <参数列表> -><参数><参数闭包>

20.    <参数闭包> -> , <参数><参数闭包> | $

21.    <参数> -><标识符> | <数字> | <字符串>

22.    <While循环>->while(<逻辑表达式>){<函数块>}

23.    <逻辑表达式> -><表达式><逻辑运算符><表达式>

24.    <逻辑运算符> -> < | > | == |>=|<=

25.    <条件语句> -> if ( <逻辑表达式> ) { <函数块> } <否则语句>

26.    <否则语句> -> else { <函数块> } | $

27.    <函数返回> -> return <因式> ;

28.    <cout语句>-> cout << <标识符>; | cout<< <数字>; | cout<< <字符>;

29.    <cin语句>-> cin>> <标识符>;

# 实现基本功能：

实现int， float， char和string的数据类型，实现的操作有基本赋值，函数定义， if -else语句，输入输出语句，基本算术运算，判断运算，while循环。

最后生成可汇编的汇编语言文件。