

编译原理实验一报告

代码功能

1. 识别基本的词法错误和语法错误
2. 能够正确识别八进制和十六进制数字
3. 能够正确识别指数类型的浮点数
4. 能够识别“//”和“/*...*/”形式的注释
5. 对于无语法错误和词法错误的代码，输出正确的语法树

代码结构

代码项目中包括了五个文件，分别是**main.c**, **tree.c**, **tree.h**, **project.l** 和 **syntax.y**

main.c 用于整个代码项目的main函数定义，获取执行命令行中的需要编译的文件名

tree.h 声明了语法树的节点结构，包含了语法树节点的类型，节点数值，所在行数等信息

tree.c 定义了创建Node的函数，以及递归输出节点的函数

project.l 定义了c--的词法

syntax.y 定义了c--的语法

Makefile 是用于项目的编译运行

代码运行

```
make clean //清除编译文件

make //生成编译文件

make test //对测试文件进行测试
```

在命令行中选定到当前文件夹后，依次输入上面的三个指令即可

测试文件需要在**Makefile**中更改，在下图中画线位置输入测试文件的路径即可

```
36 # 定义的一些伪目标
37 .PHONY: clean test
38 test:
39     ./parser test_code1.c
40
41 clean:
42     rm -f parser lex.yy.c syntax.tab.c syntax.tab.h syntax.output
43     rm -f $(OBSJS) $(OBSJS:.o=.d)
44     rm -f $(LFC) $(YFC) $(YFC:.c=.h)
45     rm -f $(OTHER)
46     rm -f *~
47
48
```